Courtroom-LLM: A Legal-Inspired Multi-LLM Framework for Resolving Ambiguous Text Classifications

Sangkeun Jung
Chungnam National University /
99, Daehak-ro, Yuseong-gu,
Daejeon 34134,
Republic of Korea
hugmanskj@gmail.com

Jeesu Jung*
Chungnam National University /
99, Daehak-ro, Yuseong-gu,
Daejeon 34134,
Republic of Korea
jisu.jung5@gmail.com

Abstract

In this research, we introduce the Courtroom-LLM framework, a novel multi-LLM structure inspired by legal courtroom processes, aiming to enhance decision-making in ambiguous text classification scenarios. Our approach simulates a courtroom setting within LLMs, assigning roles similar to those of prosecutors, defense attorneys, and judges, to facilitate comprehensive analysis of complex textual cases. We demonstrate that this structured multi-LLM setup can significantly improve decision-making accuracy, particularly in ambiguous situations, by harnessing the synergistic effects of diverse LLM arguments. Our evaluations across various text classification tasks show that the Courtroom-LLM framework outperforms both traditional single-LLM classifiers and simpler multi-LLM setups. These results highlight the advantages of our legal-inspired model in improving decision-making for text classification.

1 Introduction

Text classification is a core task in Natural Language Processing (NLP), playing a crucial role in various applications. Despite recent advancements in classification performance due to the development of Large Language Models (LLMs), significant challenges remain. This study focuses on developing methods to effectively utilize LLMs for text classification without additional training.

The immediate application of LLMs holds significant importance in many real-world scenarios, primarily due to the frequent lack of time or resources for fine-tuning models on large datasets. Consequently, there is a need to develop structured methodologies that can maximize the use of pre-trained LLM knowledge while achieving high classification performance. This requires designing an innovative framework that effectively leverages

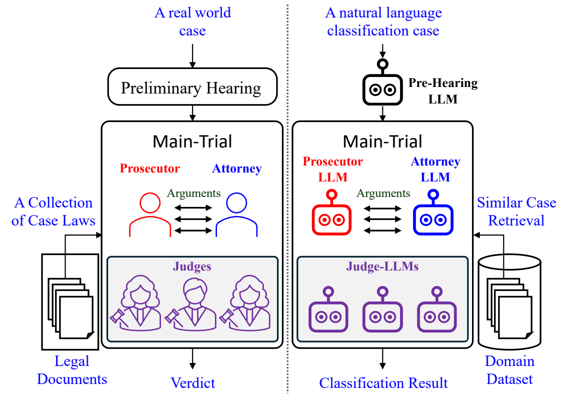


Figure 1: Comparison of the traditional courtroom system with our Courtroom-LLM framework.

LLM capabilities, rather than simply increasing model size.

In this context, a key challenge in text classification is the handling of *difficult-to-classify instances*. These cases include borderline examples where distinguishing between categories is unclear, texts with complex contexts or subtle nuances, and ambiguous cases where even experts may disagree (Brodley and Friedl, 1999). Similar to the *hard examples* described by (Bengio et al., 2009), such instances often produce inconsistent or low-confidence results in existing classification models. This issue becomes particularly critical in real-world applications, where reliable classification is essential. Thus, evaluating how well a training-free method using LLMs can manage these difficult cases is a crucial aspect of this research.

To address the challenges of text classification, particularly in handling complex and *ambiguous* cases, this study proposes a structured approach that combines both collaboration and competition among multiple LLMs. Inspired by the processes found in a legal courtroom, where roles such as prosecutors, defense attorneys, and judges engage in argumentation and deliberation, we have developed the **Courtroom-LLM framework**. In this setup (Figure 1), LLMs are assigned roles analo-

*Corresponding author

Proceedings of the 31st International Conference on Computational Linguistics, pages 7367–7385
January 19–24, 2025.

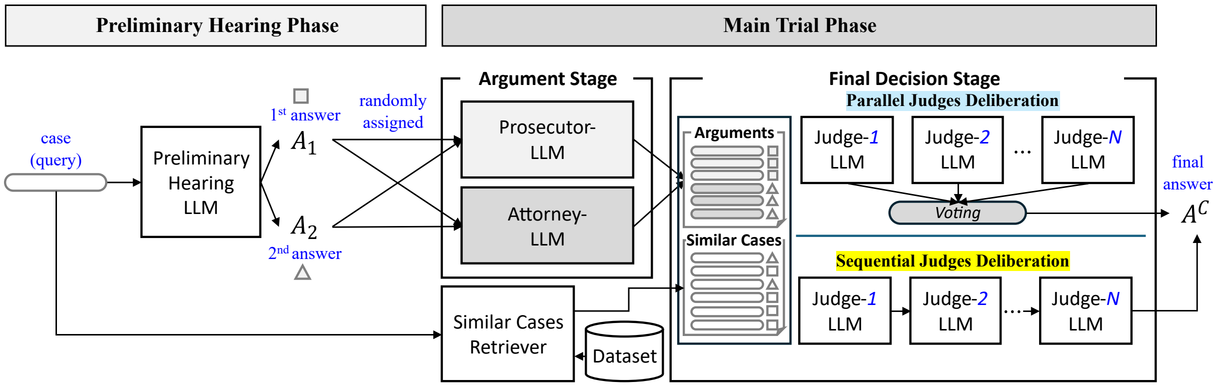


Figure 2: Overall architecture of Courtroom-LLM framework.

gous to these courtroom participants: prosecutors and defense attorneys engage in competitive argumentation, while judges collaboratively deliberate to reach a balanced decision. This structured approach allows for step-by-step reasoning, ensuring diverse perspectives are considered, leading to well-supported and interpretable outcomes, particularly for *difficult-to-classify* cases.

Through extensive experiments, we demonstrate that the Courtroom-LLM framework not only excels in handling ambiguous and difficult-to-classify cases but also shows superior performance in general classification tasks up to 150% performance gain than single model. The structured, courtroom-inspired approach leads to more robust and interpretable decision-making, ensuring that the competitive and collaborative dynamics between LLMs enhance classification accuracy across various scenarios. Importantly, this improvement is achieved without requiring additional model training, highlighting the potential of Courtroom-LLM as a versatile and powerful solution for a wide range of text classification challenges in real-world applications.

2 Related Works

Recent efforts to improve LLMs include enhancing input prompts for precision, enriching queries with context, and considering changes to LLM structures for more accurate responses.

One of the most extensively studied research directions is *prompt engineering*, which has become crucial across various tasks. Innovations in this field involve adding sequential and systematic prompts that guide response generation and optimizing the order of prompts to improve results(Mao et al., 2023). Significant advancements include the use of chain-of-thought(CoT) reasoning(Feng et al., 2024), providing *step-by-step* or *take-a-deep-breath* instructions(Shaikh et al., 2023; Yang et al., 2023), and abstracting initial queries to derive meaningful prompt blocks(Zheng et al., 2023).

To enhance LLMs' decision accuracy, recent approaches have included *additional information*, such as through retrieval functionalities. This supplementary information often comes from external search engines or internal databases(Lewis et al., 2020), employing techniques like Vector Databases(Vector DBs)(Pan et al., 2024) or search algoritm as BM25(Yu et al., 2023). Vector DBs enable efficient similarity searches and information retrieval, while CoT breaks complex problems into step-by-step considerations, enhancing the transparency of AI decision-making.

The decision accuracy is especially needed for problems that are easily distorted or ambiguous. Approaches such as filtering out easy data by using the intersection between models (Brodley and Friedl, 1999) or evaluating based on the confidence in the correct data have been utilized(Bengio et al., 2009). Concurrently, researchers are developing frameworks to assess the factual accuracy of LLMs, addressing the critical need for reliability in AI-generated information (Yang et al., 2024; Laban et al., 2023).

Research on varying *LLM connection structures* includes methods like querying multiple LLMs(Li et al., 2024) and refining the answers through post-processing, simulating real-world debates among LLMs to converge on a consensus(Yao et al., 2023; Pi et al., 2022), assigning specific roles to LLMs to gather varied responses(Suzgun and Kalai, 2024), and inducing more refined tasks through LLM cooperation or competition(Lazaridou et al., 2016).

Our study intersects the realms of prompt engineering, supplementary information provision, and exploration of LLM connection structures. By emulating a real-world courtroom system with LLMs, our research adopts an advanced approach to exploring connection structures and naturally incorporates prompt engineering by deriving materials for the final decision-making LLM from the arguments of prosecutors and attorneys. To our knowledge, this is the first attempt to implement a courtroom system through LLMs.

From an application perspective, this study is specifically focused on NLP classification tasks. Comparable approaches employed LLMs or unsupervised learning methods for classification tasks(Sun et al., 2023; Arora et al., 2022).

3 The Courtroom-LLM Framework

The Courtroom-LLM is a text classification framework inspired by legal processes, designed to enhance accuracy and fairness in handling complex, ambiguous cases. It employs a multi-LLM architecture that simulates courtroom roles, leveraging both *competition* and *collaboration* to improve decision-making.

The framework operates in two phases:

1. **Preliminary Hearing Phase:** The *Preliminary Hearing LLM (PH-LLM)* conducts an initial assessment and proposes two classification options. For reliable decisions, it uses a stable, high-performing model.
2. **Main Trial Phase:**
   (a) **Argument Stage:** The *Prosecutor-LLM* defends the PH-LLM's classification, while the *Attorney-LLM* argues for an alternative. Both LLMs can use lighter models focused on generating arguments.
   (b) **Final Decision Stage:** The *Judge-LLM(s)* synthesize these arguments to make the final classification. We experiment with different models to evaluate how size and type affect decision quality.

Figure 2 illustrates the overall architecture of our framework.

3.1 Preliminary Hearing Phase

The Preliminary Hearing Phase serves as the foundation of the Courtroom-LLM process and is led by the PH-LLM. The primary objective of this phase is to conduct an initial analysis of the input text and identify potential classification options.

In a courtroom setting, legal cases often revolve around two opposing viewpoints, each of which is presented for consideration. Similarly, in the context of text classification, the PH-LLM reduces the possible outcomes to *two* candidate classes, allowing the subsequent phases to focus on these two options and weigh them against each other. This reduction is essential to streamline the decision-making process, ensuring a binary choice in the Main Trial Phase.

The PH-LLM operates as follows:

- **Text Analysis**: The PH-LLM examines the input text, extracting key features and identifying critical themes.
- **Initial Classification**: Based on this analysis, the PH-LLM proposes the most likely classification (A_1) and an alternative (A_2).
- **Handover to Main Trial**: The classifications A_1 and A_2 are then passed to the Prosecutor-LLM and Attorney-LLM for further argumentation in the Main Trial Phase.

This initial phase plays a critical role by laying the groundwork for the argumentation and decision-making stages that follow. At its core, the PH-LLM operates much like a traditional single-LLM classifier based on standard prompt-driven techniques. The preparatory prompt for PH-LLM's initial decision-making is presented in Appendix A.1.

3.2 Main Trial – Argument Stage

The Argument Stage in the Main Trial leverages competitive dynamics by simulating opposing roles. The Prosecutor-LLM and Attorney-LLM engage in an adversarial process, each randomly assigned to represent either classification outcome (A_1 or A_2) determined by the PH-LLM. This randomization ensures unbiased argumentation for both classifications.

Each LLM constructs arguments emphasizing features of the text that align with their assigned classification, underscoring why it should be considered correct. They may use specific textual evidence or examples to support their case, while also critically examining the opposing classification and pointing out its potential weaknesses.

This adversarial interaction ensures that both classifications are rigorously evaluated before moving to the next stage. The arguments constructed by both LLMs are then passed on to the Judge-LLM(s) for collaborative decision-making in the next phase. Appendix A.2 shows an example of a prompt designed for this purpose, guiding the creation of arguments within these limits.

3.3 Main Trial - Final Decision Stage

The Final Decision Stage is where the Judge-LLM(s) engage in **collaborative decision-making** to synthesize the arguments presented during the Argument Stage and reach a final classification decision. This stage can be implemented using one of two approaches: the *Parallel Judge Method* or the *Sequential Judge Method*.

In the Parallel Judge Method, multiple Judge-LLMs **independently** assess the arguments at the same time. Each Judge-LLM reviews the arguments from the Prosecutor-LLM and Attorney-LLM, and after completing their assessments, the final decision is made by *majority vote*. This method promotes diverse perspectives by allowing each judge to evaluate the arguments without influence from others.

In the Sequential Judge Method, multiple Judge-LLMs evaluate the arguments one by one. Each Judge-LLM considers the conclusions of the previous Judge-LLM, building upon those insights to form their own judgment. This process encourages **cumulative** reasoning, where each subsequent judge adds depth to the final decision.

The key benefits of this stage include: In the parallel judge method, each Judge-LLM independently evaluates the arguments, ensuring diverse perspectives and reducing bias from any single viewpoint. In the sequential judge method, judges build on each other's deliberations, fostering collaboration that enhances decision reliability. By combining both independent assessments and collaborative refinement, the framework ensures thorough and balanced final classifications.

Appendix A.3 shows an example of a prompt designed for judge-LLMs, providing collected precedents, and arguments of the attorney and prosecutor for a better judgment.

3.4 Similar Cases Retriever

The Similar Cases Retriever is an auxiliary module of the Courtroom-LLM framework, enhancing decision-making by retrieving relevant examples from the domain dataset. Inspired by the use of legal precedents in real courtrooms, it provides Judge-LLM(s) with similar cases to ensure consistency and accuracy in classifications. This approach adapts the concept of referencing precedents to the domain of text classification, supplying the model with contextually relevant examples to inform current decisions.

The Similar Cases Retriever works by:

1. **Text Embedding**: The input text and all texts in the dataset are converted into vector representations using text embedding techniques.
2. **Similarity Calculation**: Cosine similarity is calculated between the input text and the dataset texts to determine their closeness.
3. **Selection of Similar Cases**: The top *N* most similar cases, based on similarity scores, are selected as few-shot examples.
4. **Provision of Results**: These selected cases are then made available to the Judge-LLM(s) as few-shot examples to inform the decision-making process.

By referencing these few-shot examples from similar past cases, the Judge-LLM(s) can make more informed decisions, grounded in historical examples that closely resemble the current case. The retrieved example can be found in Appendix C.1.

3.5 Bias Prevention and Fairness Enhancement

The Courtroom-LLM framework includes several methods to reduce bias and make sure the decision-making process is fair. One way to do this is by using an *argument length limitation*, which means that each LLM (like the Prosecutor-LLM or Attorney-LLM) can only provide a certain amount of information. This helps keep the arguments clear and prevents any one LLM from overwhelming the Judge-LLM(s) with too much information.

Another important feature is the use of multiple Judge-LLMs, either one after another (in sequence) or all at once (in parallel). This ensures that no single Judge-LLM has too much power or influence over the final decision. By involving multiple judges, the framework encourages a mix of perspectives, which helps make the decision more balanced and less likely to be biased.

To enhance fairness, the framework randomly assigns the two most likely classifications (A_1 and A_2) determined by the PH-LLM to either the Prosecutor or Attorney. This randomization prevents bias from consistently assigning the "best" classification

Subset	Data name	Label	Original Size (Sampled rate)
Natural Language Understanding	RTE(Wang et al., 2019)	Entailment, Non-entailment	277 (100%)
	BoolQ(Clark et al., 2019)	yes, no	2,370 (21.09%)
Natural Language Inference	QNLI(Wang et al., 2019)	Entailment, Non-entailment	5,460 (9.15%)
	ANLI(Nie et al., 2020) R1	entailment, neutral, contradiction	1,000 (50.00%)
Classification	Emotion(Saravia et al., 2018)	sadness, joy, love, anger, fear, surprise	2,000 (25.00%)

Table 1: Dataset summary. For evaluation, we randomly selected 500 samples from each dataset, except for RTE which had fewer than 500 samples in total. We utilized the validation sets for RTE, BoolQ, and QNLI, while for ANLI and emotion, we used the test sets.

	Model name	Size
Closed	GPT-4o	200B
	Gemini-1.5-Flash	10B
Open	LLaMA-3.1	8B

Table 2: Judge-LLM model size

to a particular role. It ensures both sides argue their positions equally, regardless of which classification they're defending, promoting a balanced deliberation process without favoring one classification over the other based on initial assignment.

4 Experiments

In this section, we present a comprehensive series of experiments designed to evaluate the effectiveness of the Courtroom-LLM framework across various NLP classification tasks. Our experiments aim to:

- Quantify the performance gains achieved by the Courtroom-LLM structure compared to single LLM and simple multi-LLM approaches.
- Compare the efficacy of sequential versus parallel judge deliberation within the Courtroom structure.
- Examine how model size and type influence the framework's effectiveness.
- Assess the framework's performance on ambiguous classification cases, demonstrating its robustness in challenging scenarios.

4.1 Experimental Setup

To comprehensively evaluate our Courtroom-LLM framework, we conducted experiments across a diverse range of NLP datasets and utilized various LLM configurations.

4.1.1 Datasets

We selected a variety of classification datasets widely recognized within the NLP community for their relevance and challenge. Table 1 summarizes the characteristics of these datasets.

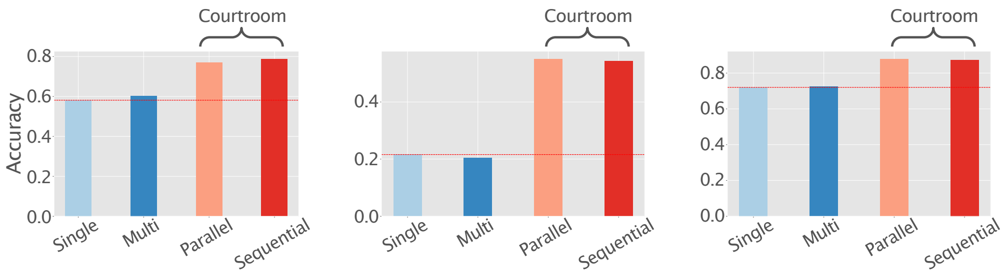
4.1.2 Model Configurations

For our Courtroom-LLM framework, we carefully selected different LLMs for each role to optimize performance and efficiency:

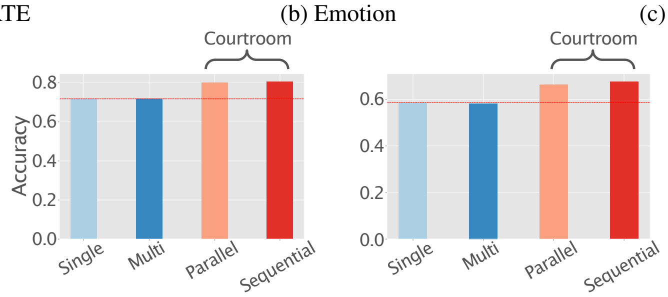
- **Preliminary Hearing LLM:** We used GPT-4o, known for its stable and consistently high classification performance, to ensure reliable initial classification.
- **Prosecutor and Attorney LLMs:** We employed LLaMA3.1-8b, an easily accessible open-source LLM, for generating arguments for and against the initial classification.
- **Judge LLMs:** To thoroughly investigate the impact of model size and type on the final decision quality, we experimented with a wide range of models, including both open-source and closed-source LLMs. The models we examined include GPT-4o(et al, 2024c), Gemini-1.5-Flash(et al, 2024b), and LLaMA 3.1 8B(et al, 2024a). Table 2 shows the Judge-LLMs' parameter size.

4.1.3 Implementation Details

To construct the similar case retriever, we utilized the embeddings of the en_core_web_sm model from spaCy (Honnibal et al., 2020). For implementing PH-LLM, Prosecutor-LLM, Attorney-LLM, and Judge-LLM, the temperature for the model was fixed at 0.5 to balance creativity and coherence in the generated responses.

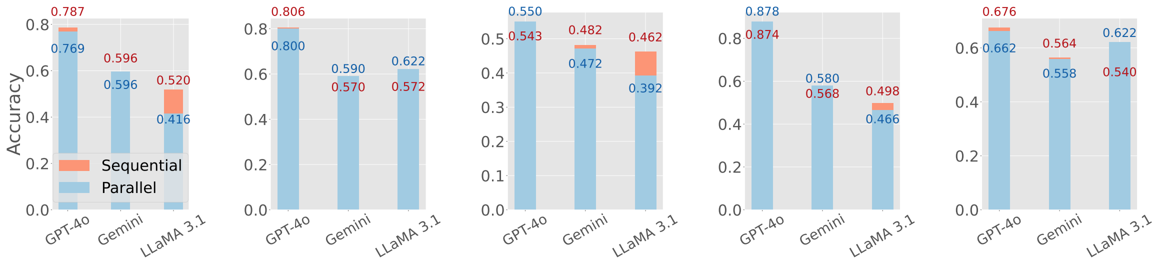


(a) RTE (b) Emotion (c) QNLI



(d) BoolQ (e) ANLI R1

Figure 3: Accuracy of Courtroom-LLM: comparing initial PH-LLM decisions with final outcomes. The graph shows the performance derived 1-shot prediction using LLaMA 3.1 for Attorney and Prosecutor LLM and GPT-4o as the PH-LLM, and Judge-LLM. ‘Single’ is the single model prediction, ‘Multi’ is the majority voting of multiple model predictions. ‘Parallel’ is the Courtroom-LLM Parallel structure, ‘Sequential’ is the Courtroom-LLM Sequential Structure.



(a) RTE (b) BoolQ (c) Emotion (d) QNLI (e) ANLI R1

Figure 4: Performance gap between the two structures of the Courtroom-LLM framework: Parallel and Sequential. GPT-4o was used as the PH-LLM, while LLaMA 3.1 8B was used as the Attorney-LLM and Prosecutor-LLM. For the Judge-LLM, three models—GPT-4o, Gemini-1.5-Flash, and LLaMA 3.1 8B—were used to make 1-shot predictions.

4.2 Performance Gains with Courtroom Structure

Our experimental results demonstrate the efficacy of the proposed Courtroom-LLM in enhancing performance. Through experiments, we quantified the extent of performance improvement achieved by implementing the novel approach. Furthermore, our analysis revealed specific conditions under which the Courtroom-LLM yields particularly significant performance gains. In our current experimental setup and model conditions, we provide that 1-shot examples consistently demonstrated the best performance as a guideline.

4.2.1 Baseline vs. Courtroom

Comparatively, the sequential judge setup significantly surpasses the single-LLM-based classification across the board. Performance improvements ranged from a 150% increase in the Emotion domain to a 11% enhancement in the QNLI task compared to baseline models. Figure 3 presents how the Courtroom-LLM approach consistently excels in classification tasks across different datasets. Limited by space, we presented only configurations with 1-shot predictions, but other setups also demonstrate enhanced performance; see Appendix B for full results.

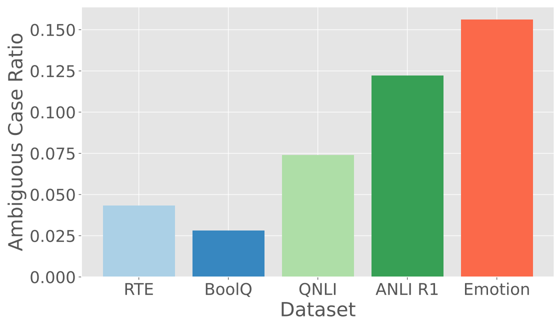


Figure 5: The proportion of ambiguous cases in each dataset. The NLU datasets (RTE, BoolQ) have the lowest ratio of ambiguous cases, while the proportion increases as we move towards NLI (QNLI, ANLI R1) and multi-label classification tasks (Emotion).

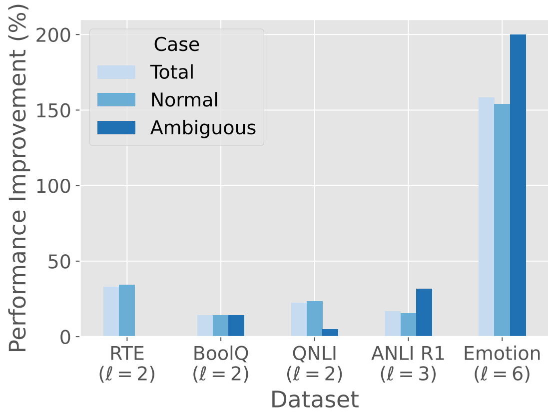


Figure 6: Performance improvement (%) when applying the Courtroom Sequential structure compared to a single-model (1-shot) prediction using GPT-4o as Judge-LLM, shown for total (combined normal and ambiguous cases), normal, and ambiguous cases across different datasets. ℓ represents the number of classification labels. The RTE dataset shows zero improvement for ambiguous cases due to the near absence of such cases.

4.2.2 Parallel vs. Sequential Judges

Two methods for structuring judges were evaluated in the Courtroom-LLM framework: *Parallel*, where judges form opinions independently, and *Sequential*, where each judge's decision is influenced by the previous judgement. Our experiments show the sequential judge structure excels in most scenarios, consistently outperforming other methods.

For the RTE task, using the Sequential approach instead of Parallel resulted in performance gains of up to 2.48%. Similarly, in the QNLI task, improvements of up to 2.45% were observed. The Emotion task showed even more significant gains, with performance increasing by 4.46%.

These improvements were especially notable in *multi-label* classification tasks compared to binary classification, and became more apparent as the number of few-shot examples increased. This suggests that incorporating opposing opinions and their respective decisions generally leads to better outcomes, similar to judicial decision-making processes. The trend is clearly illustrated in Figure 4. For each task discussed in this paper, examples demonstrating the performance improvement of the Courtroom-LLM framework over single-LLM approaches can be found in Appendix C.2.

4.2.3 Performance Analysis on Ambiguous Classification Cases

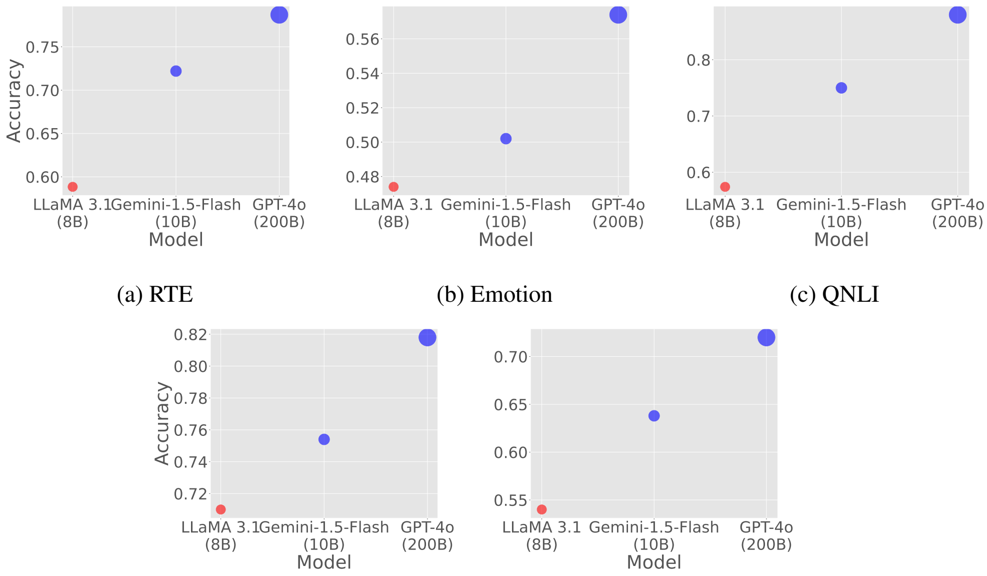
As a form of post-analysis, we conducted performance comparisons between ambiguous and normal examples.

In this study, we define an *ambiguous* example as a case where the PH-LLM produces *inconsistent* classification results over N iterations. Specifically, if the top classification result varies even once across five classification attempts ($N = 5$), we classify the case as *ambiguous*. In contrast, examples with consistent results across all iterations are considered *normal*. Using this criterion, we analyzed the distribution of ambiguous and normal cases across the datasets utilized in our experiments. The distribution of these cases is illustrated in Figure 5.

The experimental results show that ambiguous cases are more difficult for single models to resolve compared to simpler ones. In these challenging scenarios, the Courtroom-LLM delivers significant performance gains. For example, in the Emotion task, accuracy improved by up to 200% for ambiguous cases. This improvement is approximately 1.3 higher than that observed for normal cases, highlighting the framework's strength in handling more complex problems. The tendency was particularly strong in the ANLI and Emotion tasks, where ambiguous cases accounted for more than 10%. This trend is shown in Figure 6. For a description of the different Judge-LLM models used, refer to Appendix D.

4.3 Performance Analysis Across Model Sizes and Types

In our experiments, we applied the Courtroom-LLM framework to both open and closed LLMs. The framework consistently outperformed simple majority voting across all models. In addition to model size, we observed differences in performance between open and closed LLMs. Notably,



(a) RTE (b) Emotion (c) QNLI

(d) BoolQ (e) ANLI R1

Figure 7: The maximum accuracy of each LLM applied to the Courtroom-LLM framework Sequential structure. The larger the model, the greater the diameter of the node proportionally. Generally, larger models tend to achieve higher performance. Blue dots show the closed model performance, and red dot shows the open model performance.

among smaller models, the closed LLM Gemini outperformed the open LLM LLaMA 3.1, despite their similar size. This highlights the influence of both model architecture and openness on performance within the Courtroom-LLM. Figure 7 shows the sequential accuracy for each LLM size.

5 Discussion

While employing multiple LLMs for binary or multi-label classification may seem expensive initially, the long-term benefits outweigh the costs. As on-device LLMs become more feasible and the cost of using LLMs decreases, this collaborative approach could prove more cost-effective than traditional fine-tuning.

Additionally, the Courtroom-LLM framework demonstrated improved performance over single LLMs by leveraging existing models without requiring additional training, making it a resource-efficient solution.

Our experiments showed the sequential approach consistently outperforms alternatives. However, due to cost limitations, we used the same judge-LLM sequentially. Future work should explore how varying the abilities and order of judge-LLMs could further enhance decision-making and overall performance.

Currently, the large-scale models used are costly and may seem excessive for the problem at hand. However, as LLM costs decline in the near future, the proposed collaborative and competitive framework is poised to demonstrate its full impact and utility.

6 Conclusion

In this research, we introduced the Courtroom-LLM framework, an innovative approach inspired by legal courtroom procedures designed to enhance performance in ambiguous text classification tasks. By simulating roles analogous to prosecutors, defense attorneys, and judges, this multi-LLM architecture effectively balances collaborative and competitive dynamics to improve classification accuracy, particularly in challenging or borderline cases.

Our evaluations across diverse NLP classification tasks consistently demonstrate that the structured, debate-like setting of the Courtroom-LLM framework significantly outperforms traditional single-LLM classifiers and basic multi-LLM systems. The Sequential Judge approach, in particular, proved most effective, showcasing its ability to process complex reasoning step-by-step and deliver well-grounded decisions. Our experiments revealed that the Courtroom-LLM framework is especially beneficial in managing hard-to-classify instances, offering not only improved classification accuracy but also clear, explainable decision-making.

Looking ahead, this research opens new avenues for leveraging structured multi-LLM collaboration in NLP. Future work could explore applying this framework to other tasks, such as sequential labeling or generative processes, further broadening its impact on language processing technologies.

Limitations

The Courtroom-LLM framework, despite its effectiveness in NLP classification, presents certain limitations:

1. **Scope of Application**: The current setup is designed for text-classification, derived from debates between prosecutor-LLM and defense attorney-LLM. Expanding this framework to accommodate generative NLP tasks and sequential labeling scenarios remains a challenge for future development.

2. **Handling of Neutral Labels**: The framework shows limitations in accurately classifying '*neutral*' labels in tasks like natural language inference, indicating a need for improved model sensitivity to nuanced classifications.

Future enhancements to the Courtroom-LLM framework should aim to address these limitations, broadening its applicability and efficiency in diverse NLP tasks.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155857, Artificial Intelligence Convergence Innovation Human Resources Development (Chungnam National University)) and by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(No. 2022R1F1A1071047).

References

Simran Arora, Avanika Narayan, Mayee F. Chen, Laurel Orr, Neel Guha, Kush Bhatia, Ines Chami, Frederic Sala, and Christopher Ré. 2022. Ask me anything: A simple strategy for prompting language models. *Preprint*, arXiv:2210.02441.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Carla E Brodley and Mark A Friedl. 1999. Identifying mislabeled training data. *Journal of artificial intelligence research*, 11:131–167.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of NAACL-HLT 2019*.

Abhimanyu Dubey et al. 2024a. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Gemini Team et al. 2024b. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

OpenAI et al. 2024c. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. 2024. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Philippe Laban, Wojciech Kryscinski, Divyansh Agarwal, Alexander Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. 2023. SummEdits: Measuring LLM ability at factual reasoning through the lens of summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9662–9676, Singapore. Association for Computational Linguistics.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2016. Multi-agent cooperation and the emergence of (natural) language. *arXiv preprint arXiv:1612.07182*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. 2024. More agents is all you need. *Preprint*, arXiv:2402.05120.

Junyu Mao, Stuart E. Middleton, and Mahesan Niranjan. 2023. Do prompt positions really matter? *Preprint*, arXiv:2305.14493.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

James Jie Pan, Jianguo Wang, and Guoliang Li. 2024. Survey of vector database management systems. *The VLDB Journal*, pages 1–25.

Xinyu Pi, Qian Liu, Bei Chen, Morteza Ziyadi, Zeqi Lin, Qiang Fu, Yan Gao, Jian-Guang Lou, and Weizhu Chen. 2022. Reasoning like program executors. *arXiv preprint arXiv:2201.11473*.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Omar Shaikh, Hongxin Zhang, William Held, Michael Bernstein, and Diyi Yang. 2023. On second thought, let's not think step by step! bias and toxicity in zero-shot reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4454–4470, Toronto, Canada. Association for Computational Linguistics.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023. Text classification via large language models. *arXiv preprint arXiv:2305.08377*.

Mirac Suzgun and Adam Tauman Kalai. 2024. Meta-prompting: Enhancing language models with task-agnostic scaffolding. *Preprint*, arXiv:2401.12954.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In the Proceedings of ICLR.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Jiuding Yang, Hui Liu, Weidong Guo, Zhuwei Rao, Yu Xu, and Di Niu. 2024. Reassess summary factual inconsistency detection with large language model. In *Proceedings of the 1st Workshop on Towards Knowledgeable Language Models (KnowLLM 2024)*, pages 27–31, Bangkok, Thailand. Association for Computational Linguistics.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Wenhao Yu, Zhihan Zhang, Zhenwen Liang, Meng Jiang, and Ashish Sabharwal. 2023. Improving language models via plug-and-play retrieval feedback. *Preprint*, arXiv:2305.14002.

Prompt for Preliminary Hearing

Choose the most appropriate label from the provided label list for the sentence.
Sentence:
sentence1: Security forces were on high alert after an election campaign in which more than 1,000 people, including seven election candidates, have been killed.
sentence2: Security forces were on high alert after a campaign marred by violence.
Label: entailment, non-entailment

Model Output (A_1)

entailment

Figure 8: Example of the preparatory prompt used in PH-LLM's initial decision-making process.

Prompt for Argument Stage (Prosecutor-LLM)

Please explain the context within these two sentence about the label *Entailment* ← This part will be *not-entailment* in case of an attorney-LLM.
The result should be one sentence.
Sentence:
sentence1: Security forces were on high alert after an election campaign in which more than 1,000 people, including seven election candidates, have been killed.
sentence2: Security forces were on high alert after a campaign marred by violence.

Model Output (Argument of prosecutor-LLM)

The context within these two sentences is that security forces were on high alert due to a campaign, either an election campaign or a campaign marred by violence.

Figure 9: Example prompts for generating arguments by the prosecutor- or attorney-LLM. The highlighted label part requesting explanation to the prosecutor and attorney are different respectively.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023. Take a step back: evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*.

A Prompt for Courtroom-LLM framework

In this section, we introduce the real prompt of each role in Courtroom-LLM framework.

A.1 Preliminary Hearing Phase

For Preliminary Hearing(PH), we use a general prompt format. The input consists of sentences, and candidate labels are provided alongside it. The PH phase is conducted in a zero-shot. Figure 8

Prompt for Final Decision Stage

There are two opposite arguments about the two sentences. Referring to precedents, which one do you think is correct about this?

Sentence:
sentence1: Security forces were on high alert after an election campaign in which more than 1,000 people, including seven election candidates, have been killed.
sentence2: Security forces were on high alert after a campaign marred by violence.
Label: entailment, non-entailment

|attorney|
Both sentences describe a situation where security forces were on high alert due to a violent campaign, with the first sentence specifically mentioning the deaths of over 1,000 people and seven election candidates.
|prosecutor|
The label non-entailment refers to the fact that the second sentence does not necessarily imply the same level of violence as mentioned in the first sentence.

|precedent|
Case: Entailment
text: sentence1: As a result, peptic ulcer disease has been transformed from a chronic, frequently disabling condition to one that can be cured by a short regimen of antibiotics and other medicines.
sentence2: Antibiotics are used against peptic ulcer.
reason: The label of "entailment" is appropriate for this pair of sentences because sentence 1 implies or suggests that peptic ulcer disease can now be cured by a short regimen of antibiotics and other medicines. Sentence 2 directly states that antibiotics are used against peptic ulcer.
Case: Non-entailment
Sentence:
…

Model Output (Decision of a judge-LLM)

Based on the precedents, the correct label for the sentence pair is *entailment*.

Figure 10: Example prompts for Judge-LLM in decision making using parallel and sequential judges deliberation.

illustrates the input prompt and its output example.

A.2 Main Trial – Argument Stage

To generate the claim sentences for the argument, we provide the input sentence along with the label specified in the PH stage as input to the model. Figure 9 shows the prompt for the Attorney-LLM and Prosecutor-LLM.

A.3 Main Trial – Final Decision Stage

For the Final Decision, the model receives a total of three types of text: first, the input sentence and candidate labels; second, the claims generated in the Argument Stage; and lastly, the precedents. In the case of a sequential structure, the judge's argument is also included as input. Figure 10 illustrates the input prompt and its output example.

B Overall Accuracy

We experimented with the performance of few-shot 1 to 3 examples and judge-LLM configurations of 1, 3, and 5, using the data employed in the paper to validate the methodology. We conducted experiments for single-LLM, multiple-LLM without the Courtroom-LLM framework, and two versions of applying our framework (parallel and sequential judges). Table 3, Table 5, and Table 4 displays the performance of datasets for natural language understanding, natural language inference, and text classification task.

C Examples

C.1 Examples of Similar Case Retriever

Table 6 shows the example of the retrieval input and result comparing with random searching.

C.2 Examples of Judge LLM

In this section, we present the formatted context input and corresponding outputs for the actual judge-LLM. We provide the input forms for the RTE dataset in natural language understanding task, the ANLI R1 dataset in natural language inference task, and the Emotion dataset in text classification task, along with the outputs of single-LLM, multiple *N*-LLM, and Courtroom-LLM(parallel judges), and Courtoom-LLM(sequential judges). The inputs for RTE, ANLI R1, Emotion datasets are shown in Table 7, Table 9, and Table 11. The outputs are shown in Table 8, Table 10, and Table 12. While there have been no alterations to the actual input data, redundant information overlapping with the actual datasets has been condensed in the respective tables.

D Performance Improvement on Ambiguous Case

In this section, we present the performance improvement on ambiguous and normal case when applying the Courtroom-LLM Sequential structure

Task	LLM	Structure	fewshot		
			0	1	2
RTE	GPT-4o	Single	0.661	0.603	0.671
		Multiple	0.632	0.567	0.671
		Parallel	**0.751**	0.769	0.769
		Sequential	0.744	**0.787**	**0.783**
	Gemini-1.5-Flash	Single	0.708	0.722	0.632
		Multiple	**0.726**	**0.733**	**0.643**
		Parallel	0.491	0.596	0.596
		Sequential	0.538	0.596	0.614
	LLaMA 3.1 8B	Single	**0.588**	0.494	0.578
		Multiple	0.588	0.494	0.570
		Parallel	0.537	0.415	0.565
		Sequential	0.491	**0.520**	**0.580**
BoolQ	GPT-4o	Single	0.794	0.718	0.752
		Multiple	**0.834**	0.712	0.746
		Parallel	0.668	0.800	0.802
		Sequential	0.654	**0.806**	**0.818**
	Gemini-1.5-Flash	Single	0.738	0.684	0.704
		Multiple	**0.766**	**0.684**	**0.742**
		Parallel	0.518	0.590	0.514
		Sequential	0.464	0.570	0.500
	LLaMA 3.1 8B	Single	**0.710**	0.582	0.623
		Multiple	0.710	0.572	**0.623**
		Parallel	0.636	**0.636**	0.605
		Sequential	0.548	0.572	0.580

Table 3: Natural language understanding task accuracy comparison on RTE(Wang et al., 2019) and BoolQ(Clark et al., 2019) dataset: **Bold** indicates the highest accuracy within each structure category. Parallel N-LLMs use N independent LLMs for classification, finalized by majority voting.

Task	LLM	Structure	few-shot		
			0	1	2
Emotion	GPT-4o	Single	0.198	0.204	0.252
		Multiple	0.224	0.176	0.236
		Parallel	0.538	0.550	0.560
		Sequential	**0.562**	**0.572**	**0.574**
	Gemini-1.5-Flash	Single	0.302	0.220	0.183
		Multiple	0.286	0.218	0.174
		Parallel	0.478	0.472	0.444
		Sequential	**0.500**	**0.482**	**0.502**
	LLaMA 3.1 8B	Single	0.288	0.198	0.186
		Multiple	0.288	0.198	0.186
		Parallel	0.198	0.392	0.371
		Sequential	**0.342**	**0.462**	**0.474**

Table 4: Classification task accuracy on Emotion(Saravia et al., 2018) datasets: **Bold** indicates the highest accuracy within each structure category. Parallel N-LLMs use N independent LLMs for classification, finalized by majority voting.

Task	LLM	Structure	fewshot		
			0	1	2
QNLI	GPT-4o	Single	0.744	0.724	0.734
		Multiple	0.760	0.710	0.748
		Parallel	0.808	**0.878**	0.862
		Sequential	**0.828**	0.874	**0.880**
	Gemini-1.5-Flash	Single	0.738	0.702	0.656
		Multiple	**0.756**	**0.694**	**0.674**
		Parallel	0.514	0.580	0.576
		Sequential	0.500	0.568	0.538
	LLaMA 3.1 8B	Single	0.434	**0.568**	0.516
		Multiple	0.434	**0.568**	0.516
		Parallel	0.510	0.466	**0.562**
		Sequential	**0.574**	0.498	0.560
ANLI	GPT-4o	Single	0.704	0.582	0.626
		Multiple	**0.760**	0.594	0.682
		Parallel	0.656	0.662	0.668
		Sequential	0.686	**0.676**	**0.692**
	Gemini-1.5-Flash	Single	0.624	0.580	0.522
		Multiple	**0.676**	**0.626**	0.534
		Parallel	0.546	0.558	0.548
		Sequential	0.564	0.564	**0.562**
	LLaMA 3.1 8B	Single	0.446	0.376	0.122
		Multiple	0.446	0.376	0.338
		Parallel	0.504	0.502	0.506
		Sequential	**0.508**	**0.540**	**0.549**

Table 5: Natural language inference task accuracy comparison on QNLI(Wang et al., 2019) and ANLI(Nie et al., 2020): **Bold** indicates the highest accuracy within each structure category. Parallel N-LLMs use N independent LLMs for classification, finalized by majority voting.

input text
question: What came into force after the new constitution was herald? sentence: As of that day, the new constitution heralding the Second Republic came into force.
randomly selected example
question: Who originally hosted Who Wants to Be a Millionaire for ABC? sentence: Hosted throughout its ABC tenure by Regis Philbin, the program became a major ratings success throughout its initial summer run, which led ABC to renew Millionaire as a regular series, returning on January 18, 2000.
selected example using similar cases retriever
question: When was the new constitution promulgated? sentence: As of that day, the new constitution heralding the Second Republic came into force.

Table 6: Selected few-shot case examples of QNLI dataset using random selection and similar cases retriever. Highlighted words show the similar context, using similar cases retriever.

Context

Sentence:
sentence1: Eric Harris and Dylan Klebold, seniors at the suburban Denver school, ...
sentence2: 13 persons were killed by two students in 1999.
Label: entailment,non-entailment

Arguments

|prosecutor| $\leftarrow A_1$
The label entailment is that the event described in sentence 2 is the same as the massacre described in sentence 1 where Eric Harris and Dylan Klebold killed a teacher and 12 students, representing the violent destruction of the perception of schools as safe havens.
|attorney| $\leftarrow A_2$
The label ""non-entailment"" refers to the fact that sentence 2 does not fully capture the magnitude and impact of the event described in sentence 1, which involved the killing of a teacher, the injuring of numerous individuals, and the shattering of the perception of schools as safe places.

Precedents

Case: entailment
text:
sentence1: Rotorua has banned criminals with five or more dishonesty convictions ...
sentence2: The Central Business District (CBD) is part of Rotorua. reason: The label of 'entailment' is appropriate for this sentence pair because sentence 2 directly follows from and is implied by sentence 1. In sentence 1, it is mentioned that criminals with five or more dishonesty convictions are banned from entering the Central Business District (CBD) of Rotorua. Sentence 2 simply states that the Central Business District (CBD) is part of Rotorua, which is a logical consequence of the information provided in sentence 1. Therefore, sentence 2 can be inferred from sentence 1, indicating an entailment relationship between the two sentences.
Case: non-entailment
text:
sentence1: The court in Angers handed down sentences ranging from four months suspended to 28 years for, ...
sentence2: Franck V. comes from Angers.
reason: The reason for labeling the sentence as 'non-entailment' is that sentence 2 does not necessarily follow or logically derive from sentence 1. While sentence 1 provides information about Franck V.'s involvement in a sex ring in Angers, sentence 2 simply states that Franck V. comes from Angers. There is no direct or implied connection between Franck V.'s origin and his involvement in the sex ring mentioned in sentence 1. Therefore, sentence 2 does not necessarily follow from sentence 1, leading to the label of 'non-entailment'.

Table 7: Natural language understanding task input example of judge-LLM on RTE dataset. The precedent size is 1, and A_1 label is 'entailment' and A_2 is 'non-entailment'. All the data included in the natural language understanding task share the same input format.

Answer Label
entailment
Single-LLM Output
non-entailment
Parallel *N*-LLM Output
non-entailment
Parallel *N*-LLM Output list
[non-entailment , non-entailment , non-entailment , non-entailment , non-entailment]
Courtroom(Parallel Judges)
[non-entailment , non-entailment , non-entailment, non-entailment, non-entailment]
Courtroom(Sequential Judges) Output
entailment
Courtroom(Sequential Judges) Output list
[non-entailment, non-entailment, entailment, non-entailment, entailment]
Courtroom(Sequential Judges) Responses
|Judge 0| The correct label for this sentence pair is "non-entailment." |Judge 1| The correct label for this sentence pair is "non-entailment." |Judge 2| |Judge 2| The correct label for this sentence pair is "entailment." |Judge 3| The correct label for this sentence pair is "non-entailment." |Judge 4| The correct label for this sentence pair is "entailment."

Table 8: Natural language understanding task output example of judge-LLM input on RTE dataset. The precedent size is 1, and A_1 label is ‘entailment’ and A_2 is ‘non-entailment’.

compared to single model. Figure 11 shows the performance changes of the Courtroom-LLM Sequential structure compared to single-model (1-shot) predictions for total, normal, and ambiguous cases across datasets, with Judge-LLM variations, GPT-4o, Gemini-1.5-Flash, and LLaMA 3.1 8B.

Context
premise: Helena Sukov (]) (born 23 February 1965) is a former professional tennis player from the Czech Republic. hypothesis: Helena Sukova enjoyed attending professional football games. Label: entailment,neutral,contradiction
Arguments
|prosecutor| $\leftarrow A_1$ The hypothesis that Helena Sukova enjoyed attending professional football games cannot be determined from the given premise. |attorney| $\leftarrow A_2$ The hypothesis that Helena Sukova enjoyed attending professional football games cannot be supported or refuted based on the given information about her career as a professional tennis player.
Precedents
Case: entailment text: premise: Judy Tegart Dalton is a retired professional tennis player... hypothesis: Judy and player Margaret Court played on the same doubles team in tournaments. reason: The reason for labeling the sentence as 'entailment' is that the hypothesis is supported by the information provided in the premise. The premise states that Judy Tegart Dalton won nine Grand Slam doubles titles, and five of those titles were with Margaret Court. Therefore, it can be inferred that Judy and Margaret Court played on the same doubles team in tournaments. Case: neutral text: premise: Alexandra Lendon Bastedo was a British actress, best known for her role as secret agent Sharron Macready ... hypothesis: Alexandra Lendon Bastedo despised her role as Sharron Macready. reason: The label of the sentence is 'neutral' because it simply states a fact about Alexandra Lendon Bastedo, without expressing any positive or negative sentiment. The sentence mentions her role as Sharron Macready in the 1968 British espionage/science fiction adventure series "The Champions" and states that she is best known for it. It does not provide any opinion or judgment about her feelings towards the role. Case: contradiction text: premise: This is a list of cities in Azerbaijan. Azerbaijan is a country in the South Caucasus region ... hypothesis: Azerbaijan has over 100 cities and 4,620 villages. reason: The label of the sentence is 'contradiction' because the hypothesis states that Azerbaijan has over 100 cities and 4,620 villages, which contradicts the information presented in the premise. The premise clearly states that Azerbaijan has a total of 77 cities, including 12 Federal-level cities, and provides the specific numbers for smaller "rayon"-class cities, urban-type settlements, and villages. Therefore, the hypothesis contradicts the information given in the premise.

Table 9: Natural language inference task input example of judge-LLM on ANLI R1 dataset. The precedent size is 1, and A_1 label is 'neutral' and A_2 is 'contradiction'. All the data included in the natural language inference task share the same input format.

<table>
<tr><th>Answer Label</th></tr>
<tr><td>Neutral</td></tr>
<tr><td>Single-LLM Output</td></tr>
<tr><td>Contradiction</td></tr>
<tr><td>Parallel N-LLM Output</td></tr>
<tr><td>Contradiction</td></tr>
<tr><td>Parallel N-LLM Output list</td></tr>
<tr><td>[Contradiction, Contradiction, Contradiction, Contradiction, Contradiction]</td></tr>
<tr><td>Courtroom(Parallel Judges)</td></tr>
<tr><td>[Neutral, Contradiction, Neutral, Neutral, Contradiction]</td></tr>
<tr><td>Courtroom(Sequential Judges) Output</td></tr>
<tr><td>Neutral</td></tr>
<tr><td>Courtroom(Sequential Judges) Output list</td></tr>
<tr><td>[Contradiction, Neutral, Neutral, Neutral, Neutral]</td></tr>
<tr><td>Courtroom(Sequential Judges) Responses</td></tr>
<tr><td>|Judge 0| Based on the precedents, the correct label for the sentence "The hypothesis that Helena Sukova enjoyed attending professional football games cannot be determined from the given premise" is contradiction.
|Judge 1| Based on the precedents, the correct label for the sentence "The hypothesis that Helena Sukova enjoyed attending professional football games cannot be determined from the given premise" is neutral.
|Judge 2| Based on the precedents, the correct label for the sentence is neutral.
|Judge 3| Based on the precedents, the correct label for the sentence is neutral.
|Judge 4| Based on the precedents, the correct label for the sentence " is neutral.</td></tr>
</table>

Table 10: Natural language inference task output example of judge-LLM on ANLI R1 dataset. The precedent size is 1, and A_1 label is 'neutral' and A_2 is 'contradiction'.

Context

Sentence: i jest i feel grumpy tired and pre menstrual which i probably am but then again its only been a week and im about as fit as a walrus on vacation for the summer
Label: sadness,joy,love,anger,fear,surprise

Arguments

|prosecutor| $\leftarrow A_1$
The speaker is feeling grumpy, tired, and possibly premenstrual, which may contribute to their sadness, but they also acknowledge that it has only been a week and they are not in the best physical shape.
|attorney| $\leftarrow A_2$
The speaker is expressing their current state of feeling grumpy, tired, and premenstrual, possibly attributing it to their hormonal cycle, but also acknowledging that it has only been a week since their last period and they are not physically fit.

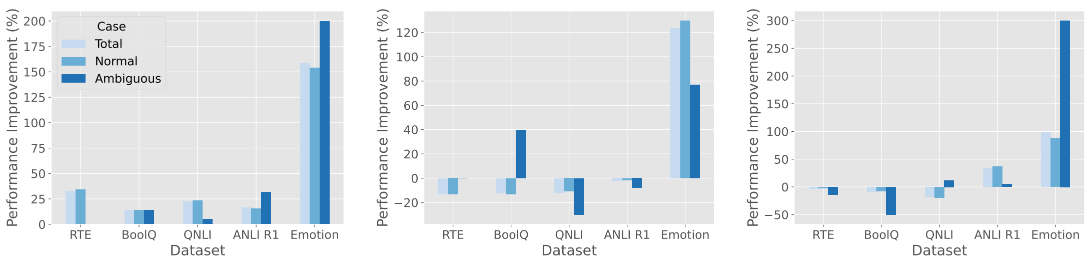
Precedents

Case: sadness
text: im pretty sure and its been about a week and a half so although im feeling kind of betrayed and disillusioned by men at the moment everythings okay
reason: The label of "sadness" is assigned to this sentence because the person expresses feeling betrayed and disillusioned by men. These negative emotions indicate a sense of sadness or disappointment. Despite stating that everything is okay, the initial emotions conveyed suggest a underlying feeling of sadness.
Case: anger
text: im not condoning terrorist action but you feel so furious and powerless
reason: The label of 'anger' is appropriate for this sentence because the speaker expresses a strong emotion of fury. The use of the word "furious" indicates a high level of anger and frustration. Additionally, the feeling of powerlessness further emphasizes the intensity of the speaker's anger.

Table 11: Text classification task input example of judge-LLM on Emotion dataset. The precedent size is 1, and A_1 label is 'sadness' and A_2 is 'anger'.

Answer Label
anger
Single-LLM Output
suprise
Parallel *N*-LLM Output
anger
Parallel *N*-LLM Output list
[suprise, anger, anger, anger, anger]
Courtroom(Parallel Judges)
[sadness, anger, sadness, sadness, sadness]
Courtroom(Sequential Judges) Output
anger
Courtroom(Sequential Judges) Output list
[anger, sadness, anger, anger, anger]
Courtroom(Sequential Judges) Responses
|Judge 0| The correct label for the given sentence is "anger." |Judge 1| The correct label for the given sentence is "sadness." |Judge 2| The correct label for the given sentence is "anger." |Judge 3| The correct label for the given sentence is "anger." |Judge 4| The correct label for the given sentence is "anger."

Table 12: Text classification task output example of judge-LLM on Emotion dataset. The precedent size is 1, and A_1 label is 'sadness' and A_2 is 'anger'.



(a) GPT-4o (b) Gemini-1.5-Flash (c) LLaMA 3.1 8B

Figure 11: Performance change (%) of the Courtroom-LLM Sequential structure over single-model (1-shot) prediction for total, normal, and ambiguous cases across datasets. GPT-4o serves as PH-LLM, LLaMA 3.1 8B as Attorney- and Prosecutor-LLM, while GPT-4o, Gemini-1.5-Flash, and LLaMA 3.1 8B are used as Judge-LLM.