Cutting Through the Noise: Detecting Fallacies in Online Discourse with NLP

Allen Li
School of Information, UC Berkeley
allen_li@berkeley.edu

Arthur Kang
School of Information, UC Berkeley
arthurkang123@berkeley.edu

Skylar Wang
School of Information, UC Berkeley
skylarmwang@berkeley.edu

Abstract

Understanding and identifying logical fallacies online has become increasingly important towards fighting misinformation and improving media literacy. This paper explores different Natural Language Processing approaches, including fine-tuned transformer models like RoBERTa and LLaMA 3.1, prompt-based Large Language Models using Active Prompting, and a novel method incorporating Logical Structure Trees (LST), to determine which model/method is able to best classify fallacious sentences within a challenging out-of-domain dataset. Our experiments, evaluated against a Multinomial Naïve Bayes baseline, demonstrate that LSTs outperformed other methods by leveraging argument structure. While traditional fine-tuning showed improvements over the baseline, prompt-based methods like Active Prompting struggled with domain transfer and prompt overload. These findings emphasize the value of structural representations for effective fallacy detection and point out the continuing challenges of LLM generalization in nuanced reasoning tasks.

1 Introduction

Online platforms and social media have fundamentally reshaped how individuals interact with information and with one another. While these spaces encourage open dialogue, they can also facilitate the spread of flawed arguments. Logical fallacies — forms of reasoning that distort the truth and derail constructive conversation — are among those arguments that can take many shapes, from personal attacks to emotional appeals, and are often convincing enough to slip by unnoticed. Developing tools to classify logical fallacies is a key step towards improving media literacy.

Despite its relevance, **logical fallacy detection remains an underdeveloped task** in the NLP literature. Existing models often struggle to capture the nuanced reasoning required to distinguish between fallacy types. This project explores whether **prompt-only approaches**, such as **Active Prompting** and **Logical Structure Trees (LST)**, can match or outperform **fine-tuned transformer models**, including **RoBERTa** and **LLaMA 3.1 with LoRA adapters**, in classifying logical fallacies. As a baseline, we implement a simple yet interpretable **Multinomial Naïve Bayes classifier**, trained on fallacious sentences and their corresponding labels from the Logical Fallacy Understanding Dataset (LFUD) dataset. This setup allows us to evaluate whether prompt engineering alone can exceed performance competitive with supervised fine-tuning, and to what extent the added complexity and resource requirements of transformer-based models are warranted for this task.

2 Background

2.1 LOGIC and LogicClimate datasets

Jin et al. (2022) pioneered the task of logical fallacy detection, introducing the LOGIC and LogicClimate datasets to benchmark this new challenge. They demonstrated that while existing pre-trained large language models struggled with this task, their novel structure-aware classifier outperformed these baselines by incorporating logical argument structure. However, their approach also revealed key limitations, such as difficulties generalizing to more complex, journalistic texts and its reliance on a single logical form per fallacy type, which may oversimplify the diversity of reasoning patterns found in real-world discourse. To build upon this, we aim to enhance cross-domain generalizability for improved detection of nuanced fallacies across diverse textual styles.

2.2 Logical Fallacy Understanding Dataset (LFUD)

Li et al. (2024) introduced the LFUD dataset to assess LLMs' ability to identify and reason through twelve distinct types of logical fallacies. Each of the 804 examples was synthetically generated by prompting GPT-4 to produce fallacious arguments tied to 67 unique propositions. This design ensures the focus remains on reasoning patterns rather than superficial cues. We use LFUD as a core fine-tuning resource due to its structured format and clear interpretability.

2.3 Active Prompting

Diao et al. (2023) introduced Active Prompting as a strategy to boost LLM performance by focusing annotation efforts on the most uncertain examples — those

where the model is least confident. This approach not only reduces manual effort but also outperforms traditional chain-of-thought (CoT) methods across various reasoning tasks. Building on their success, we adapt Active Prompting for logical fallacy detection using LLaMA 3. By identifying uncertain examples, we construct targeted few-shot prompts that guide the model more effectively. We evaluate this method on the LogicClimate dataset to assess its impact on detecting nuanced fallacies across different domains.

2.4 Logical Structure Trees

Lei and Huang (2024) proposed Logical Structure Trees (LST) as a reasoning scaffold, demonstrating their ability to improve LLM classification accuracy and interpretability by explicitly representing logical relations in arguments. Given their lightweight and model-agnostic design, our work explores LSTs as a promising alternative to task-specific fine-tuning for enhancing logical fallacy classification by providing explicit structural input to the model.

3 Methods

To address the challenge of logical fallacy detection, we conducted a series of experiments comparing traditional fine-tuning techniques with prompt-based strategies, all evaluated against a well-defined baseline.

3.1 Data

To support our fine-tuning and experimentation, we constructed a multi-source training corpus of fallacious arguments paired with labeled fallacy types. At its core is the Logical Fallacy Understanding Dataset (LFUD) released by Li et al. (2024), which consists of 804 examples spanning twelve fallacy categories. Each example is derived from one of 67 unique propositions, statements that are either true or false. Because of its limited size and lexical variety, we augmented LFUD with two additional sources: (1) GPT-4-generated examples following the data augmentation framework outlined in Li et al. (2024), and (2) real-world annotated arguments from the LOGIC dataset (Jin et al., 2022). Both the LFUD and LOGIC datasets are publicly available on GitHub. These combined sources allow us to fine-tune both a Naïve Bayes baseline and transformer models (RoBERTa, LLaMA) on a diverse set of reasoning patterns—ranging from highly structured prompts to naturally occurring discourse—while testing our hypothesis that prompt-only methods can rival or outperform fine-tuned models.

To expand LFUD synthetically, we selected 33 new propositions from *A Concise Introduction to Logic (Hurley, 14th Ed.)* Each was used as a seed for generating twelve fallacious arguments—one per fallacy type—by prompting GPT-4 using structured templates adapted from Li et al. (2024). This process yielded 396 new examples, enriching the dataset with greater lexical diversity while preserving conceptual clarity and consistency in reasoning structure.

In addition to synthetic data, we integrated annotated fallacy examples from the LOGIC dataset (Jin et al., 2022), which contains over 2,000 annotated arguments drawn from naturally occurring text. Following Li et al. (2024), we excluded instances labeled as "equivocation" and "miscellaneous" due to their ambiguous semantics and lack of definitional clarity, thereby minimizing label noise. To align with the fallacy labels in LFUD's schema, we relabeled LOGIC's and LogicClimate's "fallacy of logic" instances as "deductive fallacy". Both labels refer to invalid reasoning patterns such as affirming the consequent or denying the antecedent. These preprocessing steps yielded a unified dataset of N=3,600 examples suitable for fine-tuning and evaluation.

To ensure that the models generalized to new reasoning contexts rather than memorizing specific propositions, we performed stratified splitting at the proposition level for the LFUD portion of the dataset. Specifically, we extracted the set of unique propositions and allocated 90% to the training set and 10% to the validation set, ensuring no proposition appeared in both. This approach mitigates lexical leakage and provides a more rigorous evaluation of the model's ability to classify fallacies across distinct scenarios. For the LOGIC dataset, which consists of naturally occurring arguments without propositions, we applied a standard random 90/10 train-validation split. The resulting LFUD and LOGIC subsets were then merged to form the final training and validation sets used for fine-tuning.

To assess out-of-domain generalization, we used the LogicClimate dataset (N=1285) introduced by Jin et al. (2022). Unlike the curated and synthetic examples used for fine-tuning, LogicClimate contains fallacy-labeled arguments extracted from real-world climate change discourse. This benchmark provides a realistic test of model robustness in unfamiliar domains. Evaluation metrics include overall accuracy and class-wise F1 scores across all twelve fallacy types.

3.2 Baseline Model

As a baseline for our experiments, we trained a Multinomial Naïve Bayes classifier using the combined LFUD and LOGIC dataset. This model offers a straightforward and interpretable benchmark against which more complex transformer-based approaches can be evaluated. Each fallacious sentence is paired with one of twelve fallacy type labels, allowing the classifier to learn basic associations between linguistic patterns and reasoning errors. To assess generalization, we tested the model on the LogicClimate dataset (Jin et al., 2022), which contains fallacy-labeled arguments drawn from real-world climate change discourse. This out-of-domain evaluation provides insight into the model's ability to handle unfamiliar contexts and reasoning styles.

3.3 Experiment 1: Fine-tuning RoBERTa

This experiment aimed to improve logical fallacy classification beyond a shallow baseline by fine-tuning a **RoBERTa-large** model. RoBERTa (Robustly Optimized BERT Pretraining Approach) was selected for its improved pre-training strategy over BERT (Bidirectional Encoder Representations from Transformers), specifically its removal of next-sentence pretraining and use of dynamic masking during training. These enhancements allow RoBERTa to better capture complex linguistic patterns, making it a strong candidate for reasoning-intensive tasks like fallacy classification.

Model Architecture and Hyperparameters: We used the `roberta-large` implementation from Hugging Face configured for sequence classification across twelve logical fallacy types. Tokenization was handled using the corresponding `RobertaTokenizer`. Input sequences were padded and truncated to a maximum length of 256 tokens, and each fallacy type was mapped to a unique numerical label for training. This setup enabled the model to learn fine-grained distinctions between reasoning errors while maintaining compatibility with our unified dataset.

3.4 Experiment 2a. Fine-tuning LLaMa 3.1 with short prompt

This experiment investigated the use of LLMs for logical fallacy detection, focusing on fine-tuning the `LLaMA 3.1 8B Instruct` model. LLMs were chosen for their advanced instruction-following capabilities, which may make them more effective at capturing nuanced reasoning patterns than traditional transformer models like RoBERTa. We conducted two rounds of fine-tuning, exploring the effects of prompt design and LoRA hyperparameter tuning on model performance. The first iteration emphasized a concise instruction-style prompt for fallacy classification.

Model Architecture and Hyperparameters: Using HuggingFace, we loaded the `LLaMA 3.1 8B Instruct` model via **unsloth**, applying 4-bit quantization to reduce memory usage during training. LoRA adapters were integrated into both the attention and feed-forward layers, initialized with the following parameters:

$$r = 16$$
$$lora\_alpha = 16$$
$$lora\_dropout = 0$$

where r is the LoRA rank. Input sequences were configured with $max\_seq\_length = 2048$, though actual training inputs were tokenized to 512 tokens, with padding and truncation applied as needed.

For fine-tuning, each example was split into a "prompt" and "target" pair. During tokenization, the prompt and target were explicitly concatenated to form a single input sequence, allowing the model to learn the completed sequence of the input question and expected output.

Prompt: "Classify the logical fallacy in this sentence. Sentence: <*sentence*>. Fallacy:"

Target: The corresponding fallacy label (e.g., "ad hominem", "false causality").

This format trained the model to generate the fallacy label directly following the prompt, reinforcing task alignment. We used a learning rate of 2e-4, the Adam optimizer, and trained for 1 epoch. To prevent overfitting, an EarlyStoppingCallback with a patience of 5 was applied, monitoring $eval\_loss$ every 10 steps and restoring the best-performing model at the end.

For evaluation on LogicClimate, the model was prompted using the same format as during training. Given the generative nature of LLMs, the raw outputs varied slightly in phrasing. To standardize predictions, we used the `rapidfuzz` module to map generated outputs to the closest valid fallacy label in the LogicClimate dataset.

3.5 Experiment 2b. Fine-tuning LLaMa 3.1 with longer prompt

Building on the initial short-prompt strategy, this second iteration seeks to boost LLaMA 3.1's performance by enriching the training context and refining LoRA adapter settings. Specifically, we embedded a list of twelve fallacy definitions and a list of valid fallacy types directly into the training prompts. This approach mirrors how humans benefit from explicit guidance and structured examples, and was designed to help the model develop more nuanced reasoning capabilities. By incorporating definitions and a closed set of choices, this setup aims to tackle the challenge of subtle fallacy differentiation, a known difficulty in the NLP literature.

Model Architecture and Hyperparameters: To improve expressiveness and reduce overfitting, We increased the LoRA rank r from 16 to 64, and set $lora\_alpha$ to 64. These adjustments enable richer parameter updates during fine-tuning, better suited to the complexity of long-form prompts. Additionally, we introduced a $lora\_dropout$ of 0.05 to mitigate overfitting. All other training hyperparameters — including learning rate and optimizer settings — remained consistent with the short-prompt experiment.

The full prompt structure is:

Prompt: "<*fallacy definitions*>. Classify the logical fallacy in this sentence. Sentence: <*sentence*>. Choose from: <*fallacy types*>. Answer:

As before, the target column contained only the correct fallacy label, and prompt-target pairs were concatenated during tokenization to preserve the full

input-output sequence.

3.6 Experiment 3: Active Prompting

This experiment evaluated the use of Active Prompting, adapted from Diao et al. (2023), to improve logical fallacy detection using LLaMA 3. Unlike traditional Chain-of-Thought (CoT) methods that rely on manually selected examples, Active Prompting applies uncertainty-based sampling to identify cases where the model is least confident. These examples are then annotated with the correct fallacy label and reasoning, and used to construct a tailored few-shot prompt, offering a more efficient and adaptive alternative to static prompting.

To identify high-uncertainty examples, we passed each example from the LFUD training set through the LLaMA 3 8B Instruct model using a zero-shot prompt:

Prompt: Identify the most likely logical fallacy in this statement: <*statement*>. Fallacies: <*fallacy types*>. Briefly explain, then name the fallacy.

For each example, we generated N=5 responses and computed three uncertainty metrics:

1. **Disagreement:** The percentage of unique fallacy labels among all predictions. Measures how much the model disagrees on fallacy types across different responses. Low disagreement indicates consistent predictions, while a high disagreement suggests the model is assigning a wide variety of labels.
2. **Entropy:** Creates a probability distribution over labels and computes the Shannon entropy, which quantifies uncertainty in the distribution. Entropy accounts for both the number and probability of different labels. Compared to disagreement, entropy is more sensitive to class imbalance and gives a probabilistic measure of uncertainty rather than just diversity.
3. **Reasoning consistency:** Measures the similarity of natural-language explanations by embedding them with a sentence transformer and computing the average pairwise cosine similarity. This metric captures uncertainty at the level of reasoning rather than just label agreement, which is useful when predictions share the same label but differ subtly in their justifications, the cases in which disagreement or entropy might be overlooked.

After calculating the uncertainty metrics, we ranked all training examples by each metric and selected the top 10 per category, resulting in 30 high-uncertainty examples. These were annotated with fallacy labels and brief explanations using GPT-4, reducing manual effort while maintaining annotation quality.

We then constructed a few-shot prompt by combining these annotated examples with the original zero-shot format:

Prompt: Here are some examples of confusing arguments, along with explanations of their logical fallacies. You can refer to them for guidance: <*examples_context*>. Now, analyze the next argument. Identify the most likely logical fallacy in this statement: <*statement*>. Fallacies: <*list of fallacy types*>. Name the fallacy, then briefly explain.

We tested four prompting conditions:

1. **Zero-shot prompting**: No examples provided in the prompt.
2. **Entropy-based Active Prompting**:Includes the top-10 most uncertain examples based on entropy.
3. **Disagreement-based Active Prompting**: Includes the top-10 examples with the most divergent predictions across responses.
4. **Inconsistency-based Active Prompting**: Includes the top-10 examples with low semantic similarity across explanations.

While our approach closely follows Diao et al. (2023), we made two key adaptations due to resource constraints and task-specific considerations.

1. **Annotation Strategy:** In the original paper, the most uncertain examples were manually annotated by human experts to ensure high-quality examples. Due to limited annotation resources, we used GPT-4 to label and explain the top-10 uncertain examples. While this reduced human effort, it may introduce variability in explanation quality.

2. **Choice of Uncertainty Metrics:** Diao et al. (2023) conducted a pilot study comparing four uncertainty metrics: disagreement, entropy, variance, and self-confidence. They found that disagreement, entropy, and variance outperformed self-confidence, which was found to be unreliable due to LLM overconfidence. Based on their findings, we similarly excluded self-confidence from our evaluation. Additionally, we introduced reasoning consistency as a substitute for variance to better capture semantic uncertainty in the model's explanations, which is especially useful when models produce identical labels with different justifications.

3.7 Experiment 4: Logical Structure Trees (LST)

To enhance fallacy detection in complex argumentative texts, we implemented the Logical Structure Tree (LST) method by incorporating explicit logical structure into the model's input representation. Each argument from the LogicClimate dataset was parsed using the constituency parser from the Stanza NLP toolkit, which provides a syntactic tree of the sentence. We then traversed this tree to identify the first matching logical connective—such as causal, contrastive, analogical, or

conditional—from a predefined set of discourse relations. This connective becomes the internal node of the LST, and the corresponding argument spans are recursively parsed to form child subtrees, resulting in a hierarchical representation of the argument.

Each internal node represents a logical relation, while the leaves correspond to minimal discourse units. Unlike prior work that transforms such trees into natural language, we retain the tree in its structured form (e.g., nested brackets or JSON-like hierarchy) and feed it directly into a reasoning-capable language model along with the original argument and task instruction. This allowed the model to process both the syntactic and logical structure of the input simultaneously. This approach aims to provide the model with a more explicit representation of the underlying logical flow of arguments, which is often implicitly captured by LLMs but can be made more robust through structured input.

4 Results and Discussion

Our experiments evaluated the performance of various models — including fine-tuned transformers and prompt-based LLMs — on logical fallacy detection, particularly in an out-of-domain context. Performance was measured by overall accuracy and class-wise F1 scores on the LogicClimate dataset.

As shown in Tables 1, 2, and 3, our Logical Structure Tree (LST) model, powered by DeepSeek LLM (R1), demonstrated the most significant performance improvement, achieving an overall accuracy of 28.2% on the LogicClimate dataset, substantially outperforming the Naïve Bayes baseline (12.76%), as well as the RoBERTa and LLaMA 3.1 models (25.45% and 23.47% respectively). This result supports our hypothesis that prompt-only methods, particularly those incorporating structured representations like LST, can outperform fine-tuned transformer models in identifying flawed reasoning, especially in out-of-domain contexts.

LST showed notable gains in categories like Ad Hominem (F1=0.56), Appeal to Emotion (F1=0.468), and a particularly high F1 of 0.667 for Circular Reasoning. This reinforces the idea that explicitly representing logical flow is critical for logic-driven fallacies.

However, LST's performance was less consistent for categories like Intentional Fallacy and Fallacy of Credibility, suggesting that some errors stem from limitations in modeling implicit intent or subtle contextual cues, which are not directly captured by structural parsing alone. The model's reliance on explicit logical flow, while beneficial for certain fallacy types, appears to hinder its ability to interpret more nuanced or context-dependent forms of flawed reasoning. This indicates a potential avenue for improvement by integrating semantic or pragmatic understanding alongside structural analysis.

Among the fine-tuned models, RoBERTa-large achieved the highest accuracy at 25.45% on the LogicClimate dataset, outperforming both the Naïve Bayes baseline and the LLaMA 3.1 configurations. RoBERTa-large showed particular effectiveness in identifying fallacies such as Fallacy of Credibility (F1 = 0.4136) and Appeal to Emotion (F1 = 0.3265). These results suggest that RoBERTa is well-suited to capturing semantic and contextual cues—especially those tied to emotional appeals, authority bias, or causal misattribution. However, its performance dropped for fallacies requiring more formal logical inference, such as Deductive Fallacy (F1 = 0.06), Ad Populum (F1 = 0.0725), and False Dilemma (F1 = 0.0678). This indicates that while RoBERTa can detect surface-level patterns, it may struggle to model the underlying logical structure that defines these more abstract fallacies.

For LLaMA 3.1, the short-prompt variant (accuracy = 23.27%) outperformed the long-prompt version (20.16%), suggesting that more context does not necessarily improve performance. The long prompts, which included detailed definitions and fallacy lists, may have overwhelmed the model, making it harder to focus on relevant reasoning patterns. This highlights the importance of prompt clarity and model capacity over sheer prompt length.

Class-wise, the short-prompt LLaMA 3.1 model performed well in categories like Appeal to Emotion (F1 = 0.3820), Fallacy of Credibility (F1 = 0.3296), and Ad Hominem (F1 = 0.2745), indicating some ability to capture common semantic signals. However, both LLaMA variants struggled with fallacies such as Ad Populum, Deductive Fallacy, and Fallacy of Extension, reinforcing the idea that models lacking explicit structural reasoning may falter when the flaw lies in the argument's logic rather than its language.

For Active Prompting, none of the prompting strategies surpassed the Naïve Bayes baseline. Accuracy scores were 9.9% (inconsistency-based), 10.9% (entropy-based), and 11.7% (disagreement-based). The overall performance of Active Prompting on LogicClimate can be attributed to several factors:

Unbalanced Data Distribution: The unbalanced number of annotated examples across fallacy types within the LogicClimate dataset likely hindered the model's ability to learn robust representations for less frequent fallacy categories.

Annotation Quality: The reliance on GPT-4o for annotation may have led to less precise explanations than human experts would provide, thereby impacting the quality of the ground truth labels used for training and evaluation.

Prompt Overload: Similar to the LLaMA 3.1 findings, the potential in overwhelming the model with long, dense few-shot prompts contributed to its diminished performance.

Domain Transfer Issues: Crucially, the examples selected from the in-domain LFUD dataset did not transfer well to the real-world climate arguments in LogicClimate. This indicates that the few-shot prompts may

Models/methods	Accuracy	Ad Hominem	Ad Populum	Appeal to Emotion	Circular Reasoning
Naïve Bayes Classifier (baseline)	0.1276	0.18	0.05	0.12	0.00
RoBERTa-large	0.2545	0.2353	0.0725	0.3265	0.1905
LLaMA 3.1 w/LoRA (short prompt)	0.2327	0.2745	0.0940	0.3820	0.1538
LLaMA 3.1 w/LoRA (long prompt)	0.2016	0.2697	0.0652	0.2389	0.2222
Active Prompting (Entropy based)	0.1089	0.2971	0.0143	0.1454	0.00
Active Prompting (Disagreement based)	0.1167	0.3684	0.0171	0.20	0.00
Active Prompting (Inconsistency based)	0.0988	0.2929	0.0496	0.1770	0.08
Logical Structure Tree (DeepSeek R1)	**0.2820**	**0.5600**	**0.2670**	**0.4680**	**0.6670**

Table 1: Accuracy, prompt style, and F1 scores for the first four fallacy types on LogicClimate.

Models/methods	Deductive Fallacy	Fallacy of Credibility	Fallacy of Extension	Fallacy of Relevance
Naïve Bayes Classifier (baseline)	0.04	0.12	0.05	0.10
RoBERTa-large	0.06	**0.4136**	**0.2727**	0.1610
LLaMA 3.1 w/LoRA (short prompt)	0.1250	0.3296	0.1629	0.1590
LLaMA 3.1 w/LoRA (long prompt)	0.1098	0.2969	0.1194	0.1302
Active Prompting (Entropy based)	0.00	0.1895	0.0865	0.1345
Active Prompting (Disagreement based)	0.084	0.2105	0.0535	0.1459
Active Prompting (Inconsistency based)	0.0217	0.1224	0.0556	0.0928
Logical Structure Tree (DeepSeek R1)	**0.1540**	0.3230	0.2500	**0.3080**

Table 2: F1 scores for the next four fallacy types on LogicClimate.

Models/methods	False Causality	False Dilemma	Faulty Generalization	Intentional Fallacy
Naïve Bayes Classifier (baseline)	0.16	0.00	0.18	0.09
RoBERTa-large	**0.2929**	0.0678	0.2080	**0.2890**
LLaMA 3.1 w/LoRA (short prompt)	0.2571	0.2069	0.1749	0.2543
LLaMA 3.1 w/LoRA (long prompt)	0.1845	0.0909	0.1709	0.2762
Active Prompting (Entropy based)	0.2698	0.1455	0.0842	0.0286
Active Prompting (Disagreement based)	0.1667	0.1429	0.1033	0.0229
Active Prompting (Inconsistency based)	0.1765	0.0926	0.1379	0.0173
Logical Structure Tree (DeepSeek R1)	0.2220	**0.4440**	**0.2570**	0.0000

Table 3: F1 scores for the remaining four fallacy types on LogicClimate.

have introduced biases or patterns that failed to align with the distinct language of the target domain, limiting the generalization capability of the Active Prompting strategies. This highlights the importance of domain alignment when utilizing few-shot learning paradigms.

5 Conclusion

In this paper, we explored the task of logical fallacy detection, a critical step towards promoting rational discourse and curbing misinformation online. Our findings show that while fine-tuned models like RoBERTa and LLaMA 3.1 outperform basic baselines, methods that explicitly leverage argument structure, such as Logical Structure Trees (LST), delivered stronger performance on out-of-domain data. By explicitly modeling logical relations, LSTs validated the intuition that structured reasoning enhances interpretability and robustness.

Still, fallacy detection remains a challenging task. Active Prompting revealed limitations in domain transfer and underscored the difficulty of capturing nuanced reasoning through few-shot examples alone. Performance gaps across fallacy types suggest that models need more than structural cues — they require deeper semantic understanding to handle subtler forms of flawed logic. These insights point to the need for hybrid approaches that combine symbolic structure with contextual awareness.

6 Project Contributions

Allen Li: LFUD, Logic, and LogicClimate Preprocessing methodology (generation of new LFUD data, splitting dataset by propositions, renaming fallacy labels, etc.), Naïve Bayes baseline, fine-tuning Llama 3.1 w/LoRA, RoBERTa-large.

Arthur Kang: Logical Structure Trees (LSTs) w/ DeepSeek R1

Skylar Wang: Active Prompting (Entropy, Disagreement, and Inconsistency based)

7 References

Shizhe Diao, Pengcheng Wang, Yong Lin, Rui Pan, Xiang Liu, and Tong Zhang. 2024. Active Prompting with Chain-of-Thought for Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1330–1350, Bangkok, Thailand. Association for Computational Linguistics.

Yanda Li, Dixuan Wang, Jiaqing Liang, Guochao Jiang, Qianyu He, Yanghua Xiao, and Deqing Yang. 2024. Reason from Fallacy: Enhancing Large Language Models' Logical Reasoning through Logical Fallacy Understanding. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3053–3066, Mexico City, Mexico. Association for Computational Linguistics.

Yuanyuan Lei and Ruihong Huang. 2024. Boosting Logical Fallacy Reasoning in LLMs via Logical Structure Tree. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13157–13173, Miami, Florida, USA. Association for Computational Linguistics.

Zhijing Jin, Abhinav Lalwani, Tejas Vaidhya, Xiaoyu Shen, Yiwen Ding, Zhiheng Lyu, Mrinmaya Sachan, Rada Mihalcea, and Bernhard Schoelkopf. 2022. Logical Fallacy Detection. *In Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7180–7198, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Patrick J. Hurley. 2024. *A Concise Introduction to Logic, 14e*. Wadsworth, Belmont, CA.

A Appendix

A.1 Long Prompt for LLaMA 3.1

Here are the definitions provided in the long instruction prompt for fine-tuning LLaMA 3.1 task with LoRA:

Faulty generalization: drawing a conclusion from one or a few cases.

False causality: assuming a causal link without evidence.

Circular reasoning: the claim supports itself.

Ad populum: assuming something is true because many believe it.

False dilemma: presenting limited options when more exist.

Fallacy of relevance: using unrelated information.

Ad hominem: attacking the person instead of the argument.

Appeal to emotion: relying on emotion over logic.

Fallacy of extension: exaggerating an argument to attack it.

Fallacy of credibility: appealing to authority without proof.

Intentional fallacy: misrepresenting intent instead of content.

Deductive fallacy: flawed logical structure.

A.2 LFUD+Logic dataset used for Fine-tuning

Input	Output
All flowers don't stay open forever. Roses are a type of plants. Therefore, all plants do not stay open forever.	Faulty Generalization
When I stop looking at them, the flowers close. Therefore, the flowers withered because I didn't look at them.	False Causality
Flowers do not bloom perennially because all flowers don't stay open forever.	Circular Reasoning
Most people think that all flowers don't stay open forever, therefore it must be true.	Ad Populum
"All flowers either perish eventually or they must not be legitimate flowers."	False Dilemma

Table 4: The first 5 rows of the LFUD + Logic dataset used for fine-tuning.

A.3 LogicClimate dataset used for Inference

Source Article	Logical Fallacy
In June last year, a severe heatwave claimed over 1,000 lives in Karachi, Pakistan. Severe drought caused food shortages for millions of people in Ethiopia, with a lack of rainfall resulting in "intense and widespread" forest fires in Indonesia that belched out a vast quantity of greenhouse gas.	Intentional Fallacy
Diminishing sea ice is causing major walrus herds to haul themselves out on to land. Arctic marine species, such as snailfish and polar cod, are being pushed out of the region by species coming from further south, attracted to the warming waters. A huge algal bloom off the west coast of North America harmed marine life and fisheries.	Intentional Fallacy
A landmark report from the United Nations' scientific panel on climate change paints a far more dire picture of the immediate consequences of climate change than previously thought and says that avoiding the damage requires transforming the world economy at a speed and scale that has "no documented historic precedent."	Fallacy of Credibility
The report "is quite a shock, and quite concerning," said Bill Hare, an author of previous I.P.C.C. reports and a physicist with Climate Analytics, a nonprofit organization. "We were not aware of this just a few years ago."	Fallacy of Credibility
The World Coal Association disputed the conclusion that stopping global warming calls for an end of coal use.	False Dilemma

Table 5: The first 5 rows of the LogicClimate dataset used for model inference.