RBG-AI at FadeIT: Prompted LLMs with Label Abstraction for Logical Fallacy Detection

Meenakshi[1,†], Jairam R[1,*,†], Reshma U[1,†], Barathi Ganesh HB[2] and Michal Ptaszynski[2]

[1]RBG AI Research, RBG.AI, SREC Incubation Center, Coimbatore, Tamil Nadu 641022, India
[2]Text Information Processing Lab, Kitami Institute of Technology, Kitami, Hokkaido 090-0015, Japan

Abstract

This paper describes the submission of RBG-AI to the FadeIT shared task at EVALITA 2026, addressing the identification of logical fallacies within Italian social media discussions related to migration, climate change, and public health. The task involves significant complications due to the fine-grained taxonomy of fallacies, their long-tailed label distributions, and the inter-annotator disagreement. In order to perform well under these limitations, we suggest a unified few-shot, prompt-based paradigm based on instruction-tuned large language models, with no task-specific fine-tuning. The proposed method simultaneously delivers the assistance of two complementary subtasks: (i) the identification of the type of fallacy at the sentence level by multi-label classification, and (ii) the marking of the fallacy span at the token level with BIO. We propose a label hierarchy to merge the semantically similar fallacies into larger groups. The output of this imposed space is more reliable in case of low-supervision and subjective annotation conditions. The signal design accompanied by carefully selected in-context examples is intended to be the most helpful in case of rare fallacies, multi-label instances, and annotator variation. A systematic study on various instruction-tuned models under identical settings revealed that Meta-LLaMA-3.1-8B-Instruct provided the most robust performance on both subtasks. The empirical results showed that combining label abstraction with constrained prompting greatly improved the prediction consistency, specially for low-frequency fallacies and fragmented spans. The results point out that prompt engineering and structuring of label space are crucial levers for effective fallacy detection in low-resource and high-ambiguity environments and form a practical alternative to conventional supervised fine-tuning.

Keywords

Logical fallacy detection, Prompt-based learning, Instruction-tuned language models, Annotation uncertainty, Label abstraction

1. Introduction

A fallacy is commonly defined as a type of reasoning that is persuasive but lacks logical validity [1, 2]. These forms of argumentation are relatively common in public discourse and could potentially influence opinions, when they are deliberately or mistakenly copied. For example, false dilemma arguments limit the number of alternatives, simplify complex problems, and commonly appear in political discourse [3], advertising [4], and social media platforms such as X (Twitter) [5]. They are also common in propaganda. Recently, fallacious reasoning has found a particularly significant role in polarized arguments in society, including the 2016 Brexit referendum and community discussions about COVID-19 vaccines [6, 7, 8]. This reflects the growing need for automatic fallacy detection systems in NLP to mitigate the problem of misinformation and improve critical thinking [9]. Notwithstanding the growing interest in fallacious reasoning, the problem of fallacy detection remains difficult. Most current approaches are fragmented and concentrate primarily on a specific genre or a few types of fallacies. There is no commonly accepted classification scheme [10], and fallacies can also appear jointly or overlap in the same text [11], although most datasets are simplified to a single label per segment. The subjectivity in annotation further complicates the task, as several plausible labels may exist for the same content [12].

EVALITA 2026: 9th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Feb 26 – 27, Bari, IT
*Corresponding author.
†These authors contributed equally.
meenakshi@rbg.ai (Meenakshi); jairam@rbg.ai (J. R); reshma.u@rbg.ai (R. U); hbbg.jp@gmail.com (B. G. HB); michal@mail.kitami-it.ac.jp (M. Ptaszynski)
0009-0000-5935-9593 (J. R); 0000-0002-7172-5821 (R. U); 0000-0002-1150-2773 (B. G. HB); 0000-0002-1910-9183 (M. Ptaszynski)

The FadeIT shared task, a part of EVALITA 2026, targets these problems. It targets the fallacy detection task in Italian social media discussions on migration, climate change, and public health, using the FAINA dataset [13]. The dataset provides span-level annotations with high granularity, allows overlapping fallacies, and keeps separate annotations for different annotators [14]. In this paper, we investigate the fallacy detection performance of instruction-tuned large language models, meta-llama/Meta-Llama-3.1-8B-Instruct, in a few-shot learning scenario. By leveraging well-designed prompts and representative examples, we assess the model's capacity to learn fallacious patterns with a high degree of annotation subjectivity. The source code for implementation and the results are publicly available at https://github.com/rbg-research/FadeIT-EVALITA-2026.

2. Related Works

Recent breakthroughs in large language models (LLMs) have reignited interest in applying these models for reasoning-intensive and argumentation-related tasks, such as logical fallacy detection. Previous research on models such as GPT-3 showed great success in natural language understanding and logical inference. This suggested their potential in reasoning applications [15, 16, 17, 18]. However, these models struggled to maintain coherence over longer chains of reasoning and frequently failed on multi-step inference [19]. More recently, developments with GPT-4 introduced chain-of-thought prompting to prompt step-by-step reasoning with improved outcomes for deductive reasoning performance [20]. However, recent studies have also shown that large language models continue to struggle with complex reasoning processes and are prone to error accumulation [21]. Simultaneously, the field of LLM has changed rapidly. Both closed-source and open-source models continue to deliver competitive performance [22]. Recent open-source models such as LLaMA and Qwen2.5 have shown strong results across reasoning and language understanding tasks, which narrows the gap with closed-source models [23, 24]. Other researches have also investigated the use of few-shot and prompt-based learning strategies. The researches examined the role of instruction tuning, model size, and prompt design in task adaptation without task-specific fine-tuning [25, 26]. The results of these studies stress the need for prompt engineering as an effective way to adapt LLMs to specialized reasoning tasks.

In the context of logical fallacy identification, the existing literature has primarily employed supervised transformer models. These models tend to rely on a single correct ground truth annotation, which fails to capture the subjective nature of fallacy annotation. To overcome this problem, the FAINA dataset was proposed in the research work [27]. The proposed FAINA dataset maintains annotator disagreement and supports overlapping span annotations for a total of twenty fallacy categories. In addition, it proposes an evaluation setting that supports multiple plausible gold standards and partial span alignment. The baseline experiments conducted on the FAINA dataset make it clear that multi-label and multi-task transformer models are effective reference baselines for both sentence-level and span-level fallacy detection tasks. Based on the above developments, the recent literature attempts to apply prompt-based LLMs for fallacy detection. This is especially useful in low-resource settings and ambiguous annotation settings. Few-shot prompting provides a flexible alternative to the existing fine-tuning approach, which assists the model in adapting through structured instructions and a few examples, instead of requiring a large amount of labeled data.

3. Methodology

Our goal is to address the problem of logical fallacy detection through a unified prompt-based framework that can handle both subtasks like sentence-level classification and token-level span identification. Our proposed method is motivated by three challenges associated with the problem: (i) the lack of annotated data, (ii) the subjective nature of fallacy interpretation, and (iii) the disagreement among annotators. To address these challenges, we propose a few-shot learning framework based on instruction-following large language models. The overview of our proposed method is shown in Figure 1.

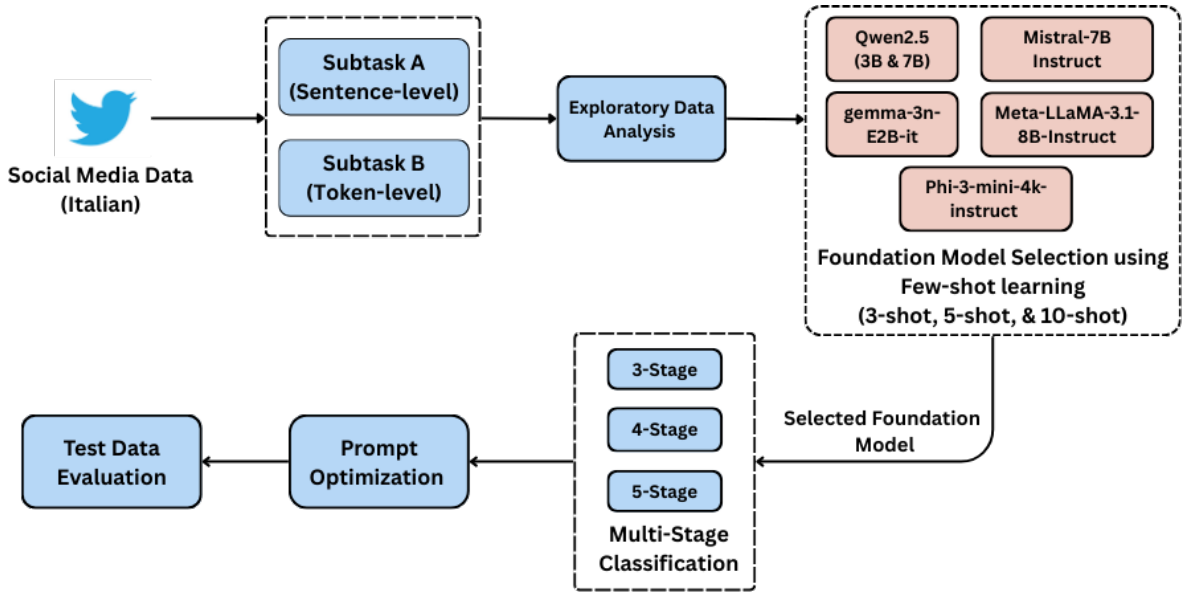

Figure 1: Overview of the proposed few-shot fallacy detection framework across both subtasks.

3.1. Task Formulation and Data Characteristics

The problem was split into two related sub-problems (subtask A and subtask B), which processed the same input texts but generated different output formats. In subtask A, fallacy detection, as a part of the FadeIT shared task [28], is formulated as a sentence-level multi-label classification problem. One sentence may demonstrate multiple types of fallacies. On the other hand, in subtask B, the fallacy detection problem is framed as a token-level sequence labeling problem. It involves detecting fallacy spans using the BIO tagging scheme. In contrast, subtask B treats fallacy detection as a token-level sequence labeling task. This requires identifying fallacy spans using the BIO tagging scheme. Although both the subtasks operates in same semantic content, they impose different modeling requirements. The sentence-level classification primarily focuses on global contextual reasoning and modeling the co-occurrence of multiple fallacy labels. In contrast, sequence labeling emphasizes accurate boundary detection and structural consistency at the token-level. Furthermore, each instance is independently annotated by two expert annotators, which naturally leads to valid disagreement in both label selection and span boundaries. The evaluation framework, following the guidelines of the official task explicitly allows for disagreements rather than forcing a single consensual annotation.

Additionally, we proceeded with an exploratory data analysis (EDA) to gain insights into the dataset characteristics. At the instance level, the files containing tab-separated values for subtask A were examined in order to check the integrity of the textual fields and annotation columns. For the purpose of clarifying the different fallacy types and topical co-occurrences, we checked the distribution of the labels assigned by the two annotators (labels_a1 and labels_a2), and the corresponding post_topic_keywords. In particular, we focused on the number of posts per topic to identify the most common subjects captured in the dataset. This task, along with data verification, also helped us identify one instance with missing value. For subtask B, we converted the token-level data released in CoNLL format into a structured dataframe for systematic analysis. This enabled us to conduct consistency checks at both token and span levels, including counting of BIO-tagged fallacy labels, analyzing span lengths, and classifying fallacy spans by topic. Overlapping or nested spans were given special attention as they are permissible under the annotation scheme and cause difficulties in sequence labeling.

The exploratory phases in combination provided a thorough view of the dataset's most important and least important features, which then influenced the selection of model and prompt design. The analysis disclosed this is a challenging aspect for models rather than a data problem, which may influence the model performance. A more elaborate quantitative discussion of these data characteristics is presented

in Section 4.1.

3.2. Few-shot Prompt-Based Modeling Framework

In this work, task adaptation is achieved through prompt engineering and example selection, without fine-tuning the model or updating task-specific parameters. We adopt a few-shot learning framework and test 3-shot, 5-shot, and 10-shot scenarios to analyze the effects of varying lengths of prompts on the stability of predictions. The concrete prompt templates for each experimental configuration are listed in the public repository (https://github.com/rbg-research/FadeIT-EVALITA-2026). The exemplars are selected to achieve maximum informational diversity, including but not limited to rare categories of fallacies, multi-label instances, and differences caused by different annotators. In this prompt-based method, we design different prompt templates for each subtask according to their respective output formats. For subtask A, the prompts instruct the model to predict one or more fallacy labels from a predefined set. For subtask B, the model is asked to point out the fallacy at the token level. The predicted results are then transformed into BIO formatted tags in CoNLL format via a post-processing step that relies on span alignment and fuzzy matching. The implementation of span alignment, fuzzy matching thresholds, and logic for BIO conversion are demonstrated in the released code repository. Despite the difference in the output format of the two sub-tasks, the two templates share the same structural framework, which includes: task descriptions, definitions of labels, and output constraints. This makes the structure compact and consistent, which is helpful for the model to reason in a stable manner across the two sub-tasks without influencing the performance of the specific task.

3.3. Model Selection and Label Grouping Strategy

To investigate the potential of prompt-based few-shot learning for fallacy detection, we chose a broad range of instruction-tuned LLMs for evaluation. The choice of the models was informed by the following factors: (i) public availability and replicability, (ii) suitability for instruction tuning for few-shot prompting, and (iii) the model's ability to support robust reasoning. The models chosen for evaluation based on the factors mentioned above include Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, Mistral-7B-Instruct, Gemma-3n-E2B, Phi-3-mini, and Meta-LLaMA-3.1-8B-Instruct. The performance of all the models was evaluated under the same experimental conditions. This included the use of a common prompting template and example selection. The performance of the models was evaluated based on the following factors: (i) the coherence of multi-label predictions, (ii) the resistance to inter-annotator variability, and (iii) accuracy over spans. The best model was then selected for evaluation based on multi-stage classification.

Apart from model selection, a hierarchical grouping of labels was adopted to address the difficulties posed by the problem. The difficulties included high class imbalance and semantic overlap between the categories of fallacies. Rather than predicting on the fine-grained fallacy categories, we aggregated similar fallacies into more coarse categories based on semantic similarity and corpus-level frequency distributions (high, medium, and low frequency labels). This served to alleviate prediction ambiguity and constrain the prediction space in few-shot prompting. The label hierarchy was defined at different levels of granularity to obtain three, four, and five group settings. For each setting, the prompt explicitly constrained predictions to the predefined set of labels. This not only constrains the decision space but also enhances discrimination performance, particularly for the less numerous classes. We treated these settings as different experimental conditions and evaluated them equally for the two subtasks. This allows for a direct evaluation of the effect of label granularity on prediction stability and interpretability with the same modeling framework.

3.4. Evaluation Protocol and Prompt Refinement

As per the official task guidelines, each model produced its output separately for each annotator. The decision to use the same predicted label for both annotators is left to the participants. The participants are encouraged to model different annotators and use different predicted labels for each of them, taking

into account the variability and, at the same time, removing a source of bias introduced by the different but still valid opinions of the annotators.

The official train-dev split, provided by the organizers, included 80% of the Faina dataset for experimentation, with 20% held aside for validation. The gold-standard annotations were produced for the validation set, which enabled the direct computation of common evaluation metrics such as precision, recall, and F1-score. The validation set was used only for (i) optimizing repetition prompts, (ii) comparing model configurations, and (iii) choosing labeling strategies. The official test set was entirely held out and not accessed at any point during model development, to avoid any information leakage. The prompt candidates were subjected to a systematic evaluation that included both quantitative measures and qualitative error analysis. A key factor in prompt selection was its robustness, consistency in predictions, and adherence to the annotation guidelines during both subtasks.

4. Results and Discussions

4.1. Dataset Characteristics and Implications for Modeling

Analyzing the distributions of the labels, some critical structural issues in the FadeIT dataset can be identified. Figure 2 and 3 illustrate the distribution of the fallacy label of subtask A based on the annotation of the annotator a1 and annotator a2.

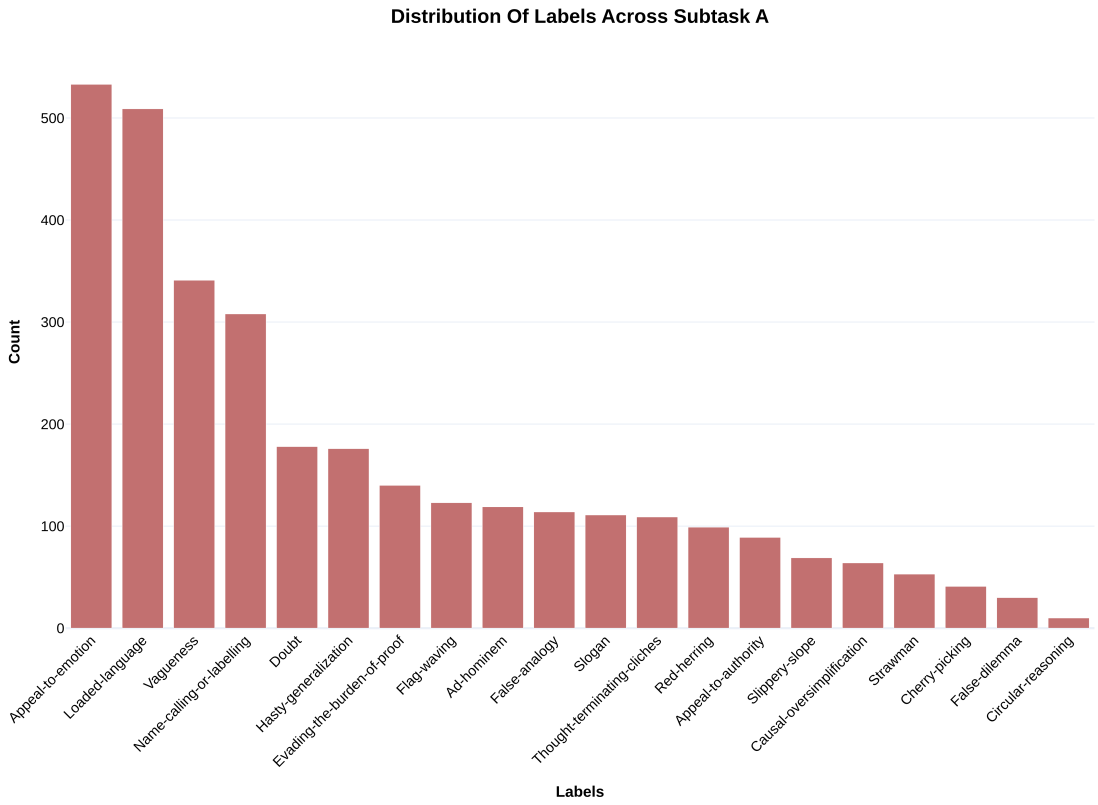

Figure 2: Label distribution for subtask A based on annotator a1, showing a strongly skewed distribution with a few dominant fallacy categories and many low-frequency classes.

The distribution is long-tailed. Some fallacies, like Appeal to Emotion, Loaded Language, and Vagueness, are more common. On the other hand, some logically subtle fallacies, such as Cherry Picking and False Dilemma, are less common. This is directly linked to the model performance in few-shot learning, where common labels are more prominent in the predictions, and less common ones are less represented in the prompt examples. The multi-label setting of subtask A makes this situation worse since multiple overlapping fallacies can be conveyed in a single sentence, making the decision

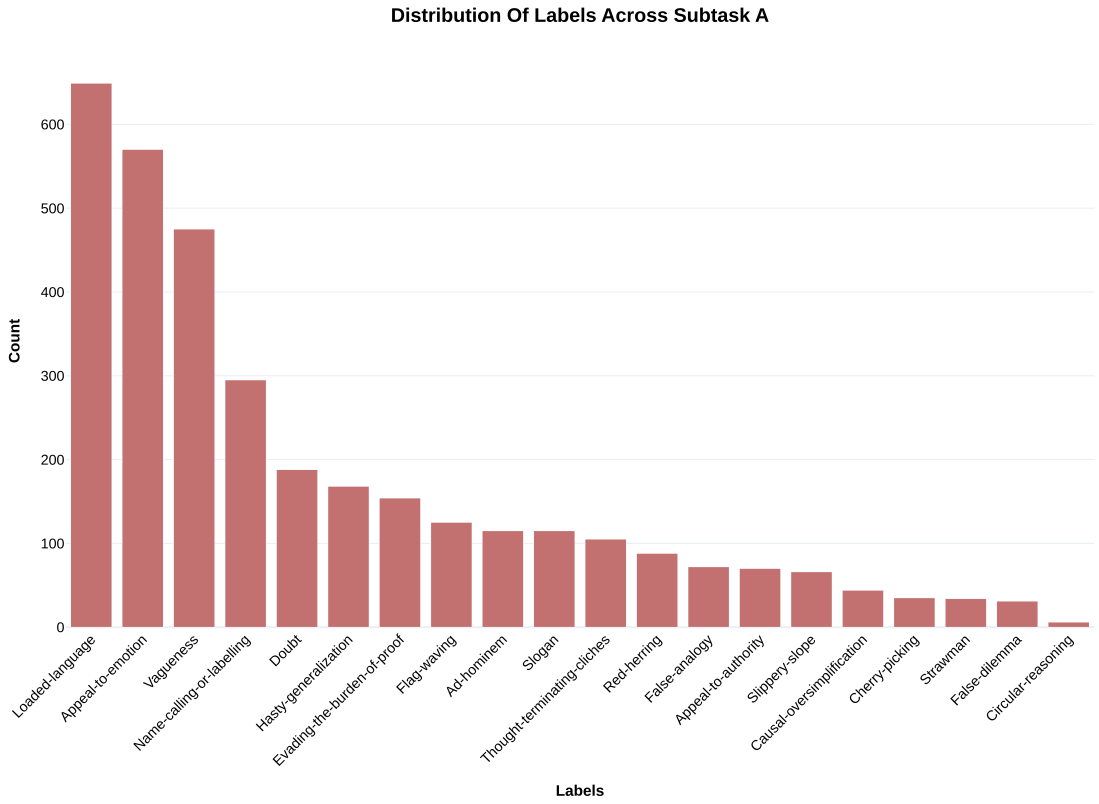

Figure 3: Label distribution for subtask A based on annotator a2, showing a strongly skewed distribution with a few dominant fallacy categories and many low-frequency classes.

boundary even more unclear. These characteristics limit the effectiveness of unconstrained multi-class classification and motivate the use of structured label-grouping strategies to stabilize the predictions.

Figure 4 and 5 shows the token-level label distribution of subtask B, excluding the overwhelmingly dominant O tag. Excluding this tag, the remaining distribution is still quite imbalanced, with most labeled tokens falling into a small minority of labels. The BIO tagging scheme exacerbates sparsity by doubling the number of fallacy categories into Band I-versions. Furthermore, span lengths vary widely across different types of fallacies. Some fallacies have short lexical cues, while others have longer argumentative spans. All these factors together contribute to the continued difficulty of token-level fallacy detection compared to sentence-level classification and the necessity of strong structural constraints in few-shot prompting.

4.2. Comparative Performance of Instruction-Tuned Models

The analysis involving the instruction-tuned language models clearly indicates a level of performance ladder achieved through the subtasks. From Table 1, the larger models, which have stronger instruction following, always outperform the smaller models, which indicates that the fallacy detection task involving subjectivity and label ambiguity benefit from larger capacity.

Meta-LLaMA-3.1-8B-Instruct scored the highest micro F1-metrics for both subtasks A and B. It demonstrated a better balance between correct positive and negative predictions (precision and recall) than any other model. The advantage of its performance is mainly evident in those situations where recall plays an important role. It is implied that the model is able to recognize fallacious reasoning which is beyond the most frequently found categories to a larger extent than the other models. On the other hand, smaller models like Qwen2.5-3B and Phi-3-mini are prone to producing unstable predictions during few-shot sessions often missing low-frequency labels or generating torn span outputs. Even though Gemma-3n-E2B-it participates actively in the competition of sentence-level classification it still

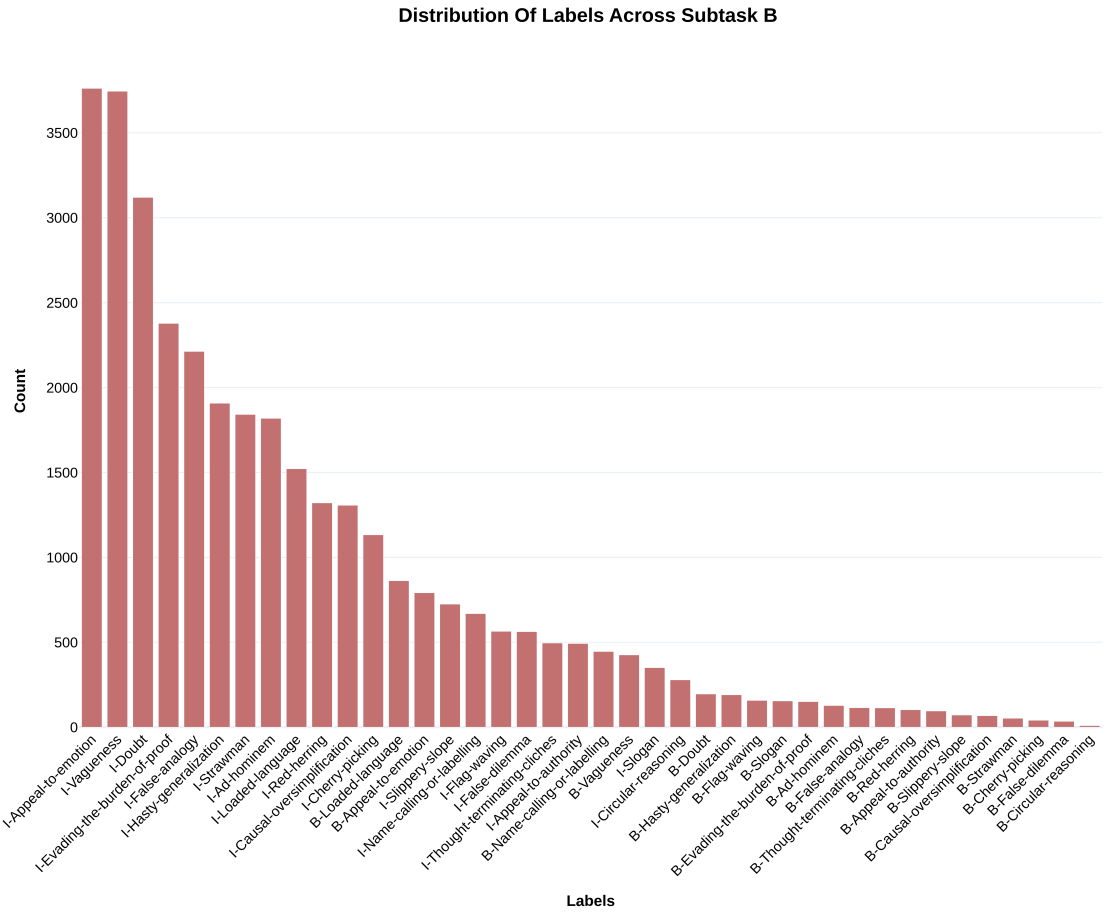

Figure 4: Distribution of BIO-tagged fallacy labels for subtask B (Annotator a1) after removing the O tag, illustrating substantial class imbalance at the token level.

Table 1
Performance comparison of instruction-tuned LLMs on Subtask A and Subtask B on the validation set (20% of train-dev split)

Model	Subtask A (micro F1)	Subtask B (micro F1, soft eval.)
Qwen2.5-3B-Instruct	13.71	2.41
Qwen2.5-7B-Instruct	17.77	7.63
Mistral-7B-Instruct-v0.3	21.39	10.11
gemma-3n-E2B-it	28.71	11.57
Meta-LLaMA-3.1-8B-Instruct	**32.68**	**13.68**
Phi-3-mini-4k-instruct	21.68	7.56

loses some points in span-level performance which indicates it is still having difficulties with fine-grained boundary detection. The findings suggest that detecting fallacies goes beyond just recognizing surface-level language features. Rather, it requires the capacity to combine annotation-specific conventions with abstract argumentative structures. As a further step, the models' precision-recall behavior is analyzed and the trade-off between the average precision and recall of each task is visualized in Figure 6.

The model's behavior concerning the precision-recall trade-off is very different for the two subtasks, as shown in Figure 6. In Subtask A, the models with higher capacity occupy the upper region of the trade-off area and thus sustain higher recalls with almost no compromise in precision or just proportional decrease, which explains their higher F1 scores. On the other hand, the smaller models' behavior is scattered more, reflecting the unstable decision boundaries under the few-shot prompting condition that they are seeing. The mentioned phenomenon is even stronger in Subtask B, where

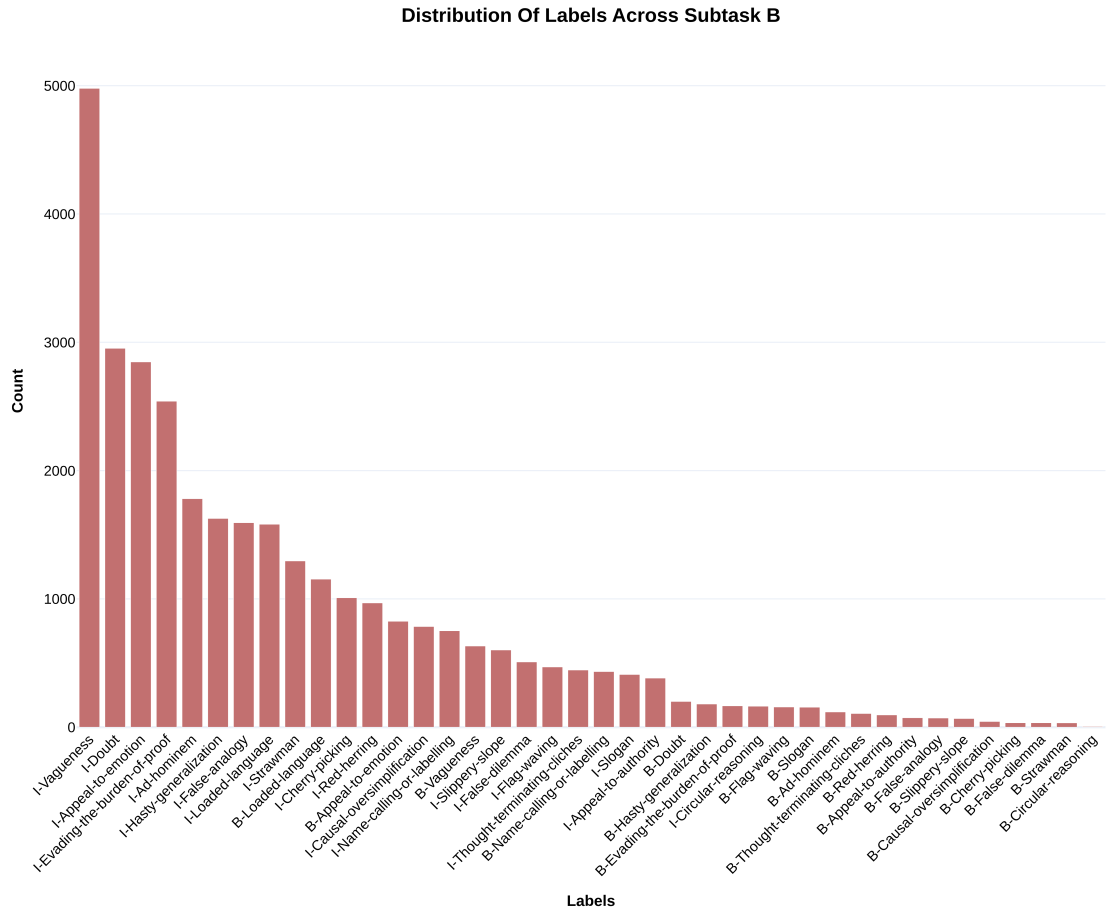

Figure 5: Distribution of BIO-tagged fallacy labels for Subtask B (Annotator a2) after removing the O tag, illustrating substantial class imbalance at the token level.

the overall recall is still low because of the sparsity and fragmentation of the span-level annotations. However, the larger instruction-tuned models give rise to better-calibrated predictions which is a sign that the effective span detection relies not only on lexical sensitivity but also on the ability to impose structural consistency within the limited output formats.

4.3. Effect of Hierarchical Label Granularity

The impact of different levels of hierarchical label abstraction on model behavior was evaluated by analyzing the subtasks through few-shot prompting, focusing mainly on stability, interpretability, and consistency of predictions in the face of annotation subjectivity and extreme class imbalance.

Considering Subtask A, dedicated to sentence-level multi-label classification, the three-stage label configuration (as described in section 3.3), where fallacies were abstracted into three groups based on corpus-level distribution and semantic relatedness attained the highest performance overall (Table 2). Grouping labels into coarser categories increased recall due to enlarging the decision space to some extent, but it very often resulted in the lowering of precision where the prediction was too generic and hence, wrong. On the contrary, the finer-grained configurations created more confusion in the multi-label situations, especially when the semantically related fallacies occurred in the same sentence. The three-stage abstraction was able to provide a balanced trade-off by limiting the output space just enough to be able to encompass overlapping fallacy types, thereby resulting in more stable and interpretable predictions. Subtask B involved identifying fallacies through BIO-based sequence labeling, and the three-stage and four-stage configurations provided similar overall performance. The four-stage configuration obtained a slight increase in soft span evaluation, mainly for short and contiguous

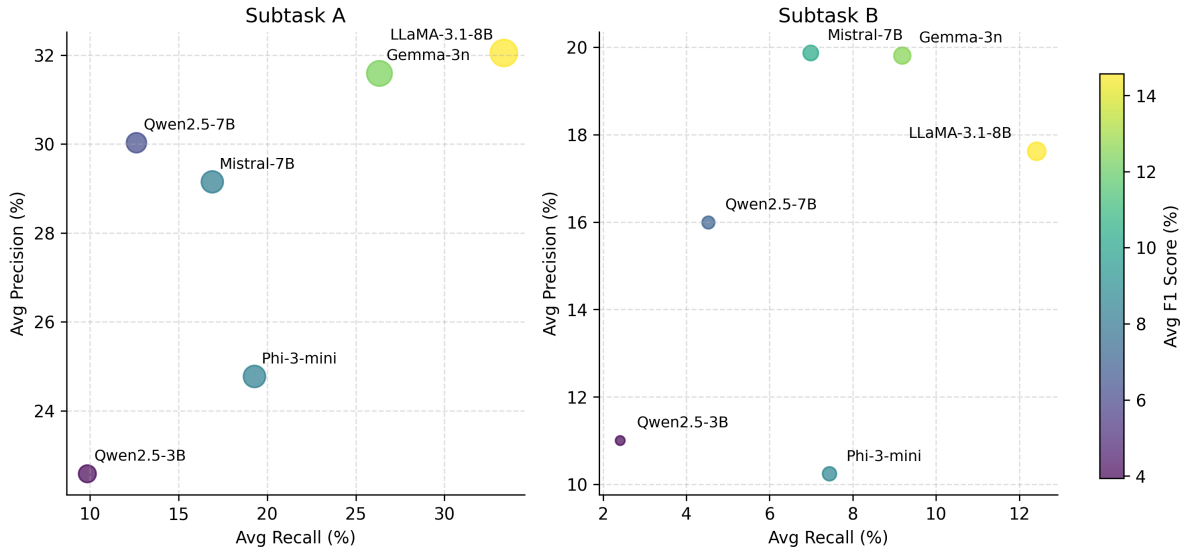

Figure 6: Precision-recall trade-off across instruction-tuned language models for Subtask A (sentence-level multi-label classification) and Subtask B (token-level span detection).

segments. Nonetheless, these advancements were not uniform across fallacy types and they were linked to a rise in fluctuation in span boundary predictions. Considering the scarcity and unevenness associated with token-level annotations, finer labels did not necessarily lead to an overall performance boost in a systematic manner.

Table 2
Effect of multistage label classification on Subtask A and Subtask B

Multi-Stage Setting	Subtask A (micro F1)	Subtask B (micro F1, soft eval.)
3-Stage	**39.03**	20.41
4-Stage	36.10	**21.77**
5-Stage	33.58	21.14

Both subtasks were considered collectively, and hence the three-stage configuration was ultimately selected as the final setting. The decision for the latter was based on the fact that although the finer abstractions provided slight advantages related to specific tasks, the overall reduction of methodological complexity and the configuration-induced biases through the use of the same label structure across the subtasks were worth it. Thus, the decision favored robustness and cross-task consistency instead of performance gains on isolated tasks. Overall, the results show that label abstraction at a moderate level is more beneficial than fine-grained partitioning in few-shot fallacy detection. Limiting the label space has a positive effect on the stability and reliability of inference, particularly in low-resource situations where there is a lot of variability in the annotations and long-tailed label distributions.

4.4. Sensitivity to Prompt Design

Prompt design is shaking out to be one of the most important factors that affect the performance of the model, particularly for subtask A. The left side of Figure 7 shows the effects of various prompt expressions on precision, recall, and micro F1-score. The prompts with too few output constraints tended to raise their recall by overpredicting the fallacy labels. The prompts that were too strict reduced the predictions of valid multi-labels, leading to a reduction in recall.

The most informative prompts were those with task instructions, definitions of label terms, and output constraints simultaneously. This approach is beneficial in ensuring that the model makes decisions within a well-defined region of the decision space, thus providing more reliable predictions

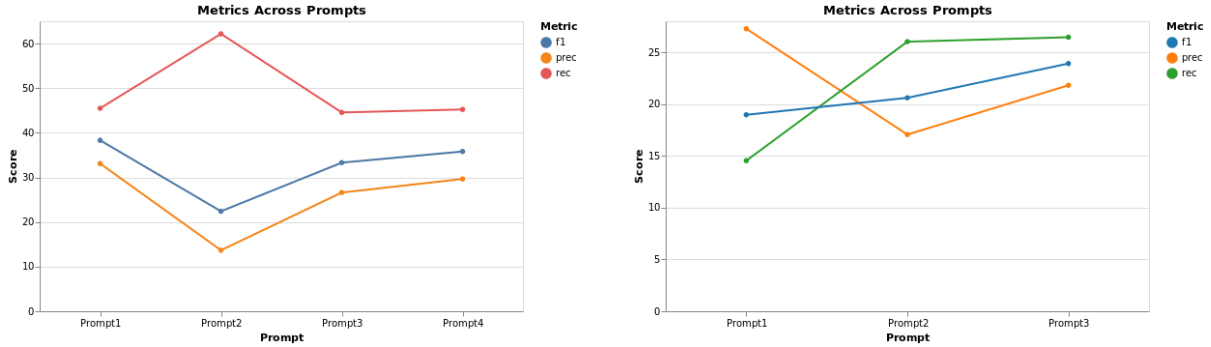

Figure 7: Impact of prompt formulation on model performance across subtasks: (left) precision, recall, and micro F1-score for sentence-level fallacy detection in subtask A; (right) micro F1-score trends for token-level fallacy detection in subtask B under soft span evaluation.

on validation sets. Of all the prompt designs considered, Prompt 1 demonstrated the smallest variability in the precision-recall curve. Hence, it is selected for further experiments.

For subtask B, the right side of Figure 7 indicates that the sensitivity to the prompt is lower in comparison; nonetheless, it is still a significant consideration. The span-level predictions are highly dependent on the accuracy of the BIO tagging task instructions and the token alignment constraints. The prompts emphasizing structural consistency and formatting rules result in more consistent spans and reduced errors in boundary fragmentation. These findings confirm the notion that prompt engineering is a core aspect of the modeling procedure rather than a secondary consideration for additional gains in few-shot fallacy detection. The full specification of all considered prompt designs, including Prompt 1 and its variations, can be found in the released code repository for complete reproducibility.

4.5. Discussion of Empirical Findings

The experimental results show that, in the FadeIT scenario, the most important factor influencing the detection of logical fallacies is the quality of the characteristics' fit. This is more important than new design features. Moreover, problems such as disagreement between annotators, unbalanced label distributions, and overlapping semantics between fallacy categories make it difficult to apply supervised learning techniques. These factors amplify the challenge further when they are operating in low-resource scenarios. The results indicate that it is very important to keep in mind the issue of subjectivity in annotation and the scarcity of annotated data when designing models aimed at achieving robust performance.

For both subtasks, the main reason for the improvements was the implementation of techniques that helped diminishing uncertainty at the inference stage. Hierarchical label abstraction proved to be very successful in narrowing down the output space, hence limiting the influence of common fallacies and also improving the detection of those categories which are less represented in the dataset. Among the three-label configurations that were experimented with, the one which provided the most consistent compromise was the one that did not break down into too many categories. This is because making too many finely grained categories became a source of confusion during in-context learning and on the contrary, making overly coarse groups resulted in the problem of increased false positives.

The prompt engineering technique proved to be a fundamental part of the modeling process. Using prompts with clear instructions, simple label definitions, and tight output constraints led to more consistent predictions than those made with unconstrained formulations. This was especially evident in sentence-level multi-label classification. Additionally, using structurally constrained prompts was key in lowering the number of errors caused by boundary misclassifications in token-level BIO-based span detection.

The model's capacity was another key factor that influenced its robustness. The larger instruction-tuned models, especially Meta-LLaMA-3.1-8B-Instruct, demonstrated more consistent reasoning, im-

proved generalization to low-frequency fallacies, and more coherent span predictions. Finally, the combination of multi-stage label grouping and exemplar driven prompting served as efficient strategies, which enabled stable inference under few-shot settings. Overall, these results emphasize the prompt centric, label structured paradigm as a strong alternative to task-specific fine-tuning for fallacy detection.

4.6. Official Test Set Results

Table 3
Official test set results on the FadeIT shared task

Run	Subtask A (Avg. micro F1)	Subtask B (Avg. micro F1, soft eval.)
I	38.57	16.17
II	**42.07**	24.31
III	40.00	**27.71**

Table 3 shows the official results on the FadeIT test set, which comprises 20% of the Faina dataset, as evaluated by the task organizers. All three submitted Runs uses the chosen Meta-LLaMA-3.1-8B-Instruct model under a 3-shot prompting setup, varying only in their label configuration strategy. For Run I, a flat label configuration was used, where the full fallacy labels was shown simultaneously to the model within a single prompt, allowing the model to predict directly over all labels without any hierarchical abstraction. Run II uses the proposed three-stage hierarchical label grouping strategy (Section 3.3), in which fallacies were partitioned into three subsets based on semantic abstraction and corpus-level frequency distributions. In this setting, the model was invoked three times per instance, each time restricted to one predefined subset of labels. Only fallacies from the active subset of labels were regarded as legitimate outputs, and the final label set was created by combining the predictions from the three invocation. Run III employed the same approach but with a four-stage grouping configuration, consisting of four consecutive model calls per case. As evident from Table 3, among the submitted Runs, Run II performed the best on average micro-F1 for Subtask A. Simultaneously, Run III reported the best results on Subtask B when evaluating soft spans. The above results demonstrate that although slightly more refined grouping can help with token-level span detection, moderate abstraction of labels helps with sentence-level multi-label prediction.

5. Conclusion

The current study examines the process of logical fallacy identification by means of two interrelated tasks. The research involves sentence categorization via multi-labeling and token-level span identification in a few-shot prompt-based learning setting. The findings of the research demonstrate the difficulties of logical fallacy identification. For instance, the task is challenging even without the presence of complicating factors such as class imbalance, sparsity, and overlap of annotations, as well as diverse interpretations of fallacies. In both tasks, larger language models that received training instructions performed better in terms of reasoning. They generated more coherent multi-label predictions and demonstrated higher consistency at the span level. The proposed multi-stage label grouping strategy worked well in handling long-tailed label distributions. The strategy achieved a balance between detail and stability in decision-making. Reducing uncertainty by using a smaller label set and designing appropriate prompts is critical in improving model robustness. This is especially the case in resource-scarce settings where task instructions have a significant impact on model behavior.

Future studies could investigate techniques for label abstraction. These would enable the granularity of the labels to be varied depending on the complexity of the context or the confidence level of the prediction made by the model. In addition, the use of large language models could be integrated with structured sequence labeling modules to improve performance and semantic consistency in span-based

fallacy identification. Extending the study to a multilingual and cross-domain setting would be valuable but would pose a challenge in understanding the process of generalization.

Declaration on Generative AI

During the preparation of this work, the author(s) used Generative AI in order to: Grammar and spelling check. After using these tool(s) the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] C. L. Hamblin, Fallacies, Advanced Reasoning Forum, 2022.

[2] C. W. Tindale, Fallacies and argument appraisal, Cambridge University Press, 2007.

[3] O. Balalau, R. Horincar, From the stage to the audience: Propaganda on reddit, in: EACL 2021-16th Conference of the European Chapter of the Association for Computational Linguistics, 2021.

[4] V. Danciu, et al., Manipulative marketing: persuasion and manipulation of the consumer through advertising, Theoretical and Applied Economics 21 (2014) 19–34.

[5] F. Macagno, Argumentation profiles and the manipulation of common ground. the arguments of populist leaders on twitter, Journal of Pragmatics 191 (2022) 67–82.

[6] F. Zappettini, The brexit referendum: How trade and immigration in the discourses of the official campaigns have legitimised a toxic (inter) national logic, in: " Brexit" as a Social and Political Crisis, Routledge, 2021, pp. 23–39.

[7] S. A. Elsayed, O. Abu-Hammad, A. B. Alolayan, Y. S. Eldeen, N. Dar-Odeh, Fallacies and facts around covid-19: the multifaceted infection, Journal of Craniofacial Surgery 31 (2020) e643–e644.

[8] G. Da San Martino, S. Yu, A. Barrón-Cedeno, R. Petrov, P. Nakov, Fine-grained analysis of propaganda in news articles, in: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), 2019, pp. 5636–5646.

[9] U. Ecker, J. Roozenbeek, S. Van Der Linden, L. Q. Tay, J. Cook, N. Oreskes, S. Lewandowsky, Misinformation poses a bigger threat to democracy than you might think, Nature 630 (2024) 29–32.

[10] H. Hansen, Fallacies. en zalta, 2020.

[11] Z. Jin, A. Lalwani, T. Vaidhya, X. Shen, Y. Ding, Z. Lyu, M. Sachan, R. Mihalcea, B. Schoelkopf, Logical fallacy detection, arXiv preprint arXiv:2202.13758 (2022).

[12] B. Plank, The'problem'of human label variation: On ground truth in data, modeling and evaluation, arXiv preprint arXiv:2211.02570 (2022).

[13] A. Ramponi, S. Tonelli, FadeIT at EVALITA 2026: Overview of the fallacy detection in italian social media texts task, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[14] A. Ramponi, A. Daffara, S. Tonelli, Fine-grained fallacy detection with human label variation, arXiv preprint arXiv:2502.13853 (2025).

[15] J. Chen, Z. Liu, X. Huang, C. Wu, Q. Liu, G. Jiang, Y. Pu, Y. Lei, X. Chen, X. Wang, et al., When large language models meet personalization: Perspectives of challenges and opportunities, World Wide Web 27 (2024) 42.

[16] M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili, et al., Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects, Authorea preprints 1 (2023) 1–26.

[17] M. Zhang, J. Li, A commentary of gpt-3 in mit technology review 2021, Fundamental Research 1 (2021) 831–833.

[18] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, J. Gao, Large language models: A survey, arXiv preprint arXiv:2402.06196 (2024).

[19] J. Yang, H. Jin, R. Tang, X. Han, Q. Feng, H. Jiang, S. Zhong, B. Yin, X. Hu, Harnessing the power of llms in practice: A survey on chatgpt and beyond, ACM Transactions on Knowledge Discovery from Data 18 (2024) 1–32.

[20] T. Gou, B. Zhang, Z. Sun, J. Wang, F. Liu, Y. Wang, J. Wang, Rationality of thought improves reasoning in large language models, in: International Conference on Knowledge Science, Engineering and Management, Springer, 2024, pp. 343–358.

[21] Y. Tong, D. Li, S. Wang, Y. Wang, F. Teng, J. Shang, Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning, arXiv preprint arXiv:2403.20046 (2024).

[22] K. He, R. Mao, Q. Lin, Y. Ruan, X. Lan, M. Feng, E. Cambria, A survey of large language models for healthcare: from data, technology, and applications to accountability and ethics, Information Fusion 118 (2025) 102963.

[23] B. Hui, J. Yang, Z. Cui, J. Yang, D. Liu, L. Zhang, T. Liu, J. Zhang, B. Yu, K. Lu, et al., Qwen2. 5-coder technical report, arXiv preprint arXiv:2409.12186 (2024).

[24] A. Yang, B. Zhang, B. Hui, B. Gao, B. Yu, C. Li, D. Liu, J. Tu, J. Zhou, J. Lin, et al., Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement, arXiv preprint arXiv:2409.12122 (2024).

[25] G. I. Winata, A. Madotto, Z. Lin, R. Liu, J. Yosinski, P. Fung, Language models are few-shot multilingual learners, arXiv preprint arXiv:2109.07684 (2021).

[26] D. Huang, X. Fu, X. Yin, H. Pen, Z. Wang, Automating maritime risk data collection and identification leveraging large language models, in: 2024 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2024, pp. 433–439.

[27] A. Ramponi, A. Daffara, S. Tonelli, Fine-grained fallacy detection with human label variation, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 762–784. doi:10.18653/v1/2025.naacl-long.34.

[28] H. Al-Omari, M. Abdullah, O. AlTiti, S. Shaikh, Justdeep at nlp4if 2019 task 1: Propaganda detection using ensemble deep learning models, in: Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda, 2019, pp. 113–118.