Leveraging Large Language Models for Classifying Subjective Arguments in Public Discourse

Adina Dobrinoiu
Supervisor(s): Luciano Cavalcante Siebert, Amir Homayounirad
Enrico Liscio
[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Adina Dobrinoiu
Final project course: CSE3000 Research Project
Thesis committee: Luciano Cavalcante Siebert, Amir Homayounirad, Enrico Liscio, Jie Yang

Abstract

This study investigates the effectiveness of Large Language Models (LLMs) in identifying and classifying subjective arguments within deliberative discourse. Using data from a Participatory Value Evaluation (PVE) conducted in the Netherlands, this research introduces an annotation strategy for identifying arguments and extracting their premises. Then, the Llama 2 model is used to test three different prompting approaches: zero-shot, one-shot and few-shot. The performance is evaluated using the cosine similarity metric and later enhanced by introducing chain-of-thought prompting. The results show that zero-shot prompting unexpectedly outperforms one-shot and few-shot prompting, due to the LLM overfitting to the examples provided. Chain-of-thought prompting is shown to improve the argument identification task. The subjectivity of the annotation task is reflected by the low averaged pairwise F1 score between annotators, and the considerable variance in the number of data items marked by each annotator as not being arguments. The subjectivity of the task is further highlighted by a pairwise chain-of-thought prompting analysis, which shows that annotators with more similar annotations received more similar LLM responses.

1 Introduction

Public deliberation is a way in which citizens can exchange opinions and discuss problems in detail. It fosters respectful and thoughtful discussions on urgent, significant, controversial, and complex issues across the globe [1]. An important element in deliberative discourse is constituted by arguments, defined as social and verbal means used to address conflicts or differences between at least two parties [2]. Arguments involve advancing a claim, which can be an opinion or an assertion [2].

As defined by James Fishkin, deliberation is the process of 'weighing' competing considerations by means of discussion [3]. Fishkin states that the attainability and effectiveness of deliberation, both in theory and practice, is based on argument formalization and the participants' attitude towards the deliberation. He outlines three criteria for arguments within deliberative discourse. First, arguments should be 'informative', supported by appropriate and accurate facts. Second, arguments should be 'balanced' such that discussions also include contrary arguments. Third, arguments should be 'substantive', evaluated solely on their merits rather than the manner or source of presentation. Therefore, argument formalization plays an important role in creating a beneficial deliberative discourse, enabling informed, balanced, and substantive discussions.

However, subjectivity, which refers to an individual's feelings, opinions, or preferences [4], is an inherent challenge in public deliberation. The diversity of beliefs, backgrounds, and perspectives among those formalizing arguments introduces variance in how arguments are perceived and annotated. Despite the expectation of neutrality, moderators might influence discussions by summarizing arguments or advocating certain viewpoints [5].

Other difficulties associated with deliberation are the large volumes of data produced in such debates [1] and the low accuracy of results, partly attributed to low participation rates [1]. A solution would be leveraging LLMs to classify deliberative discourse elements. In this way, discourse moderators would be able to comprehend and analyse different viewpoints better, resulting in more accurate deliberation outcomes. Consequently, increased accuracy may incentivize greater citizen participation.

Building upon these observations, this study aims to explore the question: **"Can LLMs detect the subjective arguments that support different stances in a deliberation?"**.

In order to answer the main focus of the research, three subquestions were created:

- How can LLMs flag and classify subjective arguments in public discourse?
- What evaluation metrics can be used to assess the performance of LLMs in argument extraction?
- How do few-shot and zero-shot approaches compare, and what impact does adding chain-of-thought reasoning have on their performance?

Section 2 of this paper provides the background of the research, summarizing previous studies and highlighting key findings and research gaps. Section 3 highlights the dataset used for the research, along with the annotation process. Section 4 showcases the methodology. Section 5 outlines the experimental setup and the results of the experiments, while Section 6 discusses these findings. Section 7 displays the limitations and future work. Section 8 indicates the responsible research aspects associated with this study. Section 9 states the conclusions.

2 Related work

Argument mining is an area of research that is concerned with automatically extracting argumentative structures such as premises, conclusions and corresponding indicators, from given data entries.

In most existing literature, argument mining is approached as an unsupervised learning task. These methods often generate large corpora of arguments that could be hard to process for discourse mediators. For this, Bar-Haim et al. [6] proposed a base framework for summarizing the arguments in form of key points that encapsulate the main ideas, making it easier to digest and utilize the information.

Less research has been conducted on argument mining as an LLM task. Chen et al. [7] have evaluated the performance of language models in two types of tasks: argument mining and argument generation, as they argue these are the most important tasks in the computational argumentation field. For the argument mining task, they adhered to a standardised prompt structure which defines the task and the required output format. They used zero-shot and few-shot approaches

to examine the effectiveness of LLMs in directly performing argument-related tasks without any prior fine-tuning. Zero-shot prompting involves providing the model with no task-specific examples, relying on its pre-trained knowledge to generate responses [8]. In contrast, in few-shot prompting, the model is provided with one or more examples to guide its responses [8].

To augment the efficiency of zero-shot and few-shot learning, Wei et al. [9] advocate for the use of chain-of-thought prompting. According to them, this method involves providing a series of intermediate reasoning steps in the prompts given to the LLM. By doing so, the model's performance improves across various tasks, including arithmetic, commonsense and symbolic reasoning [9], category into which argument extraction fits. The results indicate that chain-of-thought prompting enables LLMs to decompose complex problems into manageable steps, leading to a better performance compared to standard prompting.

The gap in the literature regarding argument extraction concerns the role of subjectivity in performing this task. This research aims to address this gap by examining how subjectivity influences argument extraction. Specifically, it seeks to determine whether LLMs can effectively identify and classify subjective arguments within deliberative discourse.

3 Dataset

This research used the dataset from the Participatory Value Evaluation (PVE) conducted by Spruit and Mouter [10], aimed at supporting the municipality of Súdwest-Fryslân in the Netherlands in co-creating an energy policy. This PVE involved 1,376 participants who were asked to distribute 100 points among six policy options, described in Table 1, and provide textual motivations for their choices. The creation and details of this dataset are described in the study by Liscio et al. [11].

Policy option	Description
o_1	The municipality takes the lead and unburdens you
o_2	Inhabitants do it themselves
o_3	The market determines what is coming
o_4	Large-scale energy generation will occur in a small number of places
o_5	Betting on storage (Súdwest-Fryslân becomes the battery of the Netherlands)
o_6	Become a major energy supplier in the Netherlands

Table 1: Policy options and their descriptions [11]

Annotating the dataset

The method employed for annotating the dataset involved generating a set of rules for annotators to follow.

Firstly, the definition of an argument was established: "An argument is **a set of claims** in which one or more of them—**the premises**—are put forward so as to offer reasons for another claim, the **conclusion**" [12]. An argument may have several premises, or it may have only one [12].
Therefore, it can be concluded that an argument can be expressed using a mathematical formula:

$$argument = \sum_{i=1}^{n} premise_i + conclusion$$

Secondly, examples of premises and conclusions were provided to the annotators, as well as examples of text that were not arguments altogether.

Thirdly, the labelling style was indicated to the annotators.

An **example argument** would be "I am installing solar panels on top of the roof next year because I care about sustainability". In this argument, **the premise** is "I care about sustainability" and **the conclusion** is "I am installing solar panels on top of the roof next year because I care about sustainability". What this research was concerned with extracting out of this sentence was the phrase "I care about sustainability". Therefore, using this definition of argument, this research is concerned with **extracting the premise(s) of an argument** if the data entry is an argument or stating **None** if the data entry is not an argument.

4 Methodology

To address the detection of subjective arguments within deliberative discourse, this research leveraged the Llama 2 large language model [13] with the default prompt temperature setting[1].

The experiments in this research consisted of four steps, as highlighted in the next subsections: using a zero-shot LLM approach for extracting arguments, using a few-shot(one or more) LLM approach for extracting arguments, using a chain-of-thought approach for extracting arguments and applying the cosine similarity metric to assess and compare the results. The three mentioned prompting techniques were chosen to evaluate the performance of LLMs under different conditions. Zero-shot prompting was used to evaluate the model's base pre-training, as no task-specific examples are provided [8]. Few-shot prompting evaluates the model's ability to generate answers when provided with one or more examples to guide its responses [8]. As argument extraction is viewed as a step-by-step task in Subsection 3, the chain-of-thought reasoning approach was applied to both zero-shot and few-shot methods to determine if it enhances their performance.

4.1 Prompting strategy

The behaviour and responses of an LLM can be guided by different roles assigned in a chat. In this research, two roles

[1]The temperature parameter associated with a prompt influences the probability distribution of the output. Higher temperatures increase randomness, resulting in more diverse outputs [14]. While higher temperatures are often associated with increased creativity, there is no strict correlation proving this [14]. Instead, higher temperatures are linked to the novelty of the LLM's responses [14]. Therefore, because there is no clear association between a specific temperature range and increased creativity, this research used the default temperature setting of Llama 2 for all prompts.

are used: ‘system’ and ‘user’. The ‘system’ role is used to set the context and guidelines for the chat. It defines how the LLM should behave and respond throughout the interaction. The ‘user’ role expresses the requests sent to the LLM.

```
{"role": "system",
 "content": "You are an expert in identifying
     arguments and their premises in a text.
     An argument is a set of claims in which
     one or more of them, the premises, are
     put forward so as to offer reasons for
     another claim, the conclusion. A text
     that does not provide support or reasons
      for a conclusion is not an argument."}
```

Listing 1: System role of the basic prompt

The snippet in Listing 1 outlines the basic system role prompt used in the research. At first, the role of the LLM in the interaction is established, and then the definition of an argument is provided.

```
{"role": "user",
 "content": "Can you extract the words in this
      sentence that state the premise or
     premises of this text: {argument} or
     extract "None" if the text is not an
     argument.
     Return the following JSON with the
         premises found in the text:
        {
            "premise_1": "",
            ...
        }
    ."}
```

Listing 2: User role of the basic prompt

The snippet in Listing 2 showcases the basic user role prompt utilized in the research. At first, the requested action is addressed to the LLM in a concise manner. Then, the format of the output is indicated.

Zero-shot approach

In the zero-shot approach, the LLM model has to perform the request indicated in the user role relying solely on its base pertaining and the information present in the prompt. This approach was performed using two different prompts: the basic prompt showcased in Subsection 4.1 and an enhanced prompt that also includes the policy option described in Section 3.

One shot approach

In the one-shot approach, the model uses a single provided example as a reference to perform the task on new inputs.

The snippet in Listing 3 highlights the addition to the basic prompt:

```
{"role": "user",
 "content":(...) An example for this task is
     for input: {input}, the output is: {
     output}."}
```

Listing 3: One-shot addition to the user role basic prompt

Besides the prompt provided to the model in the zero-shot approach, an example of what result the model should output is provided. The example is chosen to be a data item that annotators labelled most similarly.

Few-shot approach

In the few-shot approach, the model uses three provided examples as a reference to perform the task on new inputs.

The snippet in Listing 4 highlights the addition to the basic prompt:

```
{"role": "user",
 "content": "An example for this task is for
     input: {input_1}, the output is: {
     output_1}. Another example for this task
      is for input: {input_2}, the output is:
      {output_2}. Another example for this
     task is for input: {input_3}, the output
      is: {output_3}."}
```

Listing 4: Few-shot addition to the user role basic prompt

The examples provided to the model are chosen using the following criteria: the first example is, as in the one-shot approach, a data item that annotators labelled most similarly, the next example is a data entry which annotators labelled most differently and the last example is a data item that all annotators labelled as being ’None’.

Chain-of-thought reasoning

In chain-of-thought reasoning, the LLM model is provided with a step-by-step approach to the problem. Additionally, the outputs of the examples provided in the one-shot and few-shot approaches are accompanied by step-by-step explanations of the decision-making process. This systematic breakdown allows the LLM to develop a more structured and coherent overview of the process.

```
{"role": "user",
 "content": "Let’s think step by step. Can you
      extract the words in this sentence that
      state the premise or premises of this
     text: {argument} or extract "None" if
     the text is not an argument.
 Step 1, determine if the text provides any
     support or reasons for the conclusion.
     If it does not provide support or
     reasons for the conclusion, return {"
     premise_1": "None"} and don’t execute
     the next steps.
 Step 2, If it does provide support or
     reasons, return the following JSON with
     the premises found in the text:
    {
        "premise_1": "",
         ...
    }."}
```

Listing 5: Chain-of-thought addition to the user role basic prompt

Listing 5 showcases the addition of chain-of-thought reasoning to the basic user role prompt. Initially, the LLM is introduced to the concept of systematic thinking by the first sentence “Let’s think step by step” [15]. Following this, two steps are introduced and logically connected to each other.

4.2 Cosine similarity

"Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them" [16]. It is a method often employed for determining text similarity.

In this paper, cosine similarity was utilized as follows and also illustrated in Figure 1:

- For each pair of (LLM response, annotator label) for an entry, the cosine similarity matrix was computed.
- The similarity value of elements with similarity below 0.2 was set to 0, indicating no significant similarity.
- The highest similarity value in the matrix was selected for each LLM response.
- The similarity values were averaged across the entire dataset.
- The similarity values were further averaged over 10 different runs.

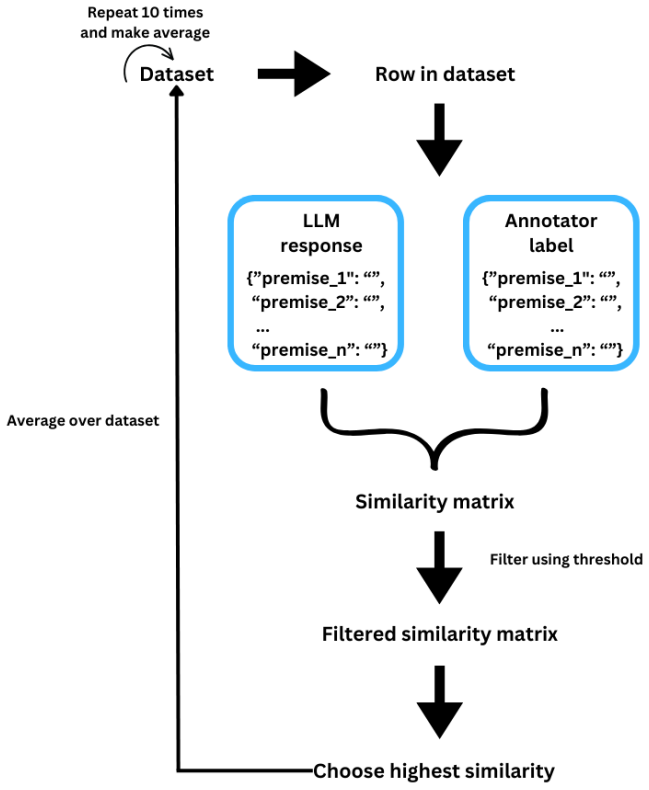

Figure 1: Cosine similarity evaluation metric

5 Results

This section highlights the experimental setup of this research and the results obtained from performing the experiments.

5.1 Experimental setup

The experiments of this research[2] were performed taking into consideration two types of dataset labels: individual annotator labels, as illustrated in Figure 2, and majority vote labels, as displayed in Figure 3.

The prompts mentioned in Section 4 were provided to the LLM with or without the policy option associated with the data entries for individual annotator labels. For majority labels, only the prompt with policy option was used.

Each prompt mentioned in Section 4 was executed 10 times for each type of label. Given that the prompt temperature was not set to 0, the LLM could produce different outputs for each of the 10 runs. This method prevented bias that could result from relying on a single set of results. Consequently, the final results represent the average of multiple LLM responses to the same prompt.

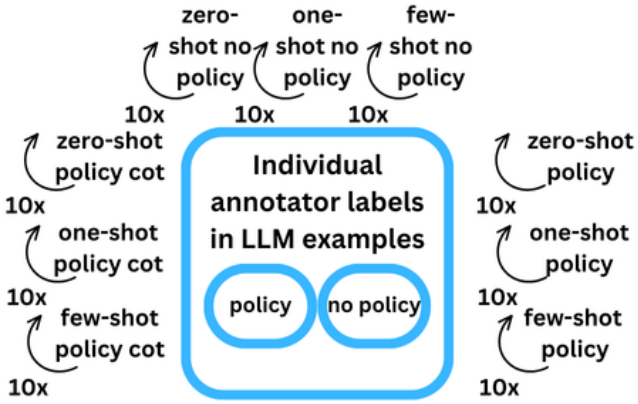

Figure 2: Individual annotator labels experiments, where cot = chain-of-thought

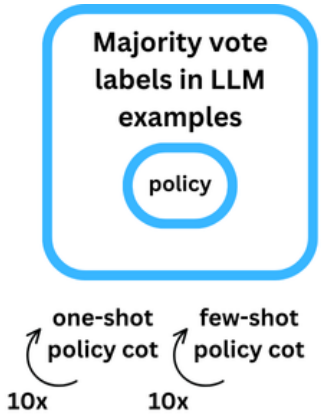

Figure 3: Majority vote labels experiments, where cot = chain-of-thought

5.2 Subjectivity of the annotation process

To address the issue of subjectivity in the annotation process, experiments have been conducted focusing on annotator agreement and the identification of non-arguments.

Agreement between annotators

To assess the agreement between annotators, various metrics were considered and evaluated. Initially, Kappa metrics were

[2] https://github.com/adina-dobrinoiu/Argument_Extraction

explored, known as a standard for interrater agreement [17]. The original Kappa index, introduced by Cohen in 1960, was designed to measure agreement between two annotators using nominal or categorical labels [18]. Fleiss extended this index to accommodate more than two annotators [19]. However, Kappa indices are not well-suited for tasks involving text span extraction [20], as they rely on the calculation of negative cases which cannot be defined for text spans, being sequences of words with variable lengths.

Therefore, an alternative metric was chosen: the pairwise F1 score. This metric computes agreement by treating one annotator's annotations as groundtruth and another's as predictions [21]. It has a range between 0 and 1, 0 being total disagreement and 1 being full agreement.

The resulting averaged pairwise F1 score for the labels provided by the 5 annotators of this research was **0.2447**. This number indicates a significant disagreement between the annotators, showcasing the subjective nature of the annotation task. It's important to note that disagreements do not imply incorrect annotations; rather, they reflect differences in how individuals interpret the argument extraction task.

Annotators	**1**	**2**	**3**	**4**	**5**
1	1	0.18	0.35	0.4	0.39
2	0.18	1	0.11	0.14	0.16
3	0.35	0.11	1	0.26	0.19
4	0.4	0.14	0.26	1	0.27
5	0.39	0.16	0.19	0.27	1

Table 2: The pairwise F1 score between the 5 different annotations of the dataset

Table 2 showcases the pairwise F1 score between the 5 different annotator labels of the dataset. What can be highlighted is the big difference between the highest agreement: **0.4**, between annotators 1 and 4, and the lowest agreement: **0.11**, between annotators 2 and 3. This illustrates that the argument extraction labelling task is subjective and prone to individual interpretation, even when annotators are given the same guidelines.

Non-arguments

Another subjective factor to be considered during the annotation process is whether the annotator believes the given text is an argument or not. As seen in Table 3, there is a big discrepancy between how many data items Annotator 2 considered not to be arguments compared to the rest of the annotators. Given the subjective nature of the argument identification task, it is important to acknowledge that Annotator 2's thought process is not necessarily incorrect relative to the others.

Ann. 1	**Ann. 2**	**Ann. 3**	**Ann. 4**	**Ann. 5**
13	35	13	18	22

Table 3: How many data entries each annotator marked as not being an argument, where Ann = annotator

Subjectivity in annotation can lead to variations in how data is labelled, which is evident from the differing judgments of the annotators. This variance highlights the importance of understanding individual annotators' perspectives and the criteria they use for classification. Even though all annotators have been given the same instructions for annotating the dataset, there are subjective factors such as feelings, opinions and preferences [4] that influence the annotation process.

5.3 Individual annotator labels

Tables 4 and 5 highlight the results of applying the average cosine similarity between the annotators' labelled data and the LLM responses. The key difference between the two tables is that in Table 4 the LLM is not provided with policy options related to the data, while in Table 5, the policy options are given. As expected, providing the LLM with more information regarding each data entry results in higher cosine similarity scores, which occurred in 8 out of the 15 cases. Although adding the policy option did not improve the results in all cases, it was included in the prompt for all subsequent computations to better resemble the annotation process, as annotators also had access to this information when performing the task.

Annotators	**LLM method - no policy**		
	Zero-shot	**One-shot**	**Few-shot**
Annotator 1	0.266	0.248	0.293
Annotator 2	0.063	0.017	0.086
Annotator 3	0.317	0.205	0.294
Annotator 4	0.169	0.187	0.241
Annotator 5	0.159	0.161	0.173

Table 4: LLM methods no policy

Annotators	**LLM method - with policy**		
	Zero-shot	**One-shot**	**Few-shot**
Annotator 1	0.297	0.243	0.261
Annotator 2	0.086	0.051	0.104
Annotator 3	0.283	0.239	0.259
Annotator 4	0.233	0.214	0.208
Annotator 5	0.161	0.147	0.139

Table 5: LLM methods with policy

When considering Table 5, an unexpected result was that the zero-shot approach generally outperformed the one-shot and few-shot approaches. This is attributed to the LLM overfitting to the provided examples in the one-shot and few-shot prompts, as seen in Table 6. Table 6 highlights the number of LLM outputs that match the example responses given in the prompt for each approach, indicating that the LLM overfit to the given examples.

After removing the faulty data entries, the cosine similarity scores were recalculated for the remaining data. The updated results presented in Table 7 showcase that there is no longer a significant score difference between the various approaches. The one-shot approach outperforms the zero-shot approach in 4 out of 5 cases. However, the few-shot approach still shows a worse performance compared to the zero-shot

Annotators	# of overfitting examples	
	One-shot	Few-shot
Annotator 1	15	3
Annotator 2	0	0
Annotator 3	25	7
Annotator 4	10	2
Annotator 5	12	6

Table 6: Number of LLM outputs that are the same as the examples provided in the prompt

and one-shot approaches. This could be caused by multiple examples generating a more subjective environment for the LLM, which could negatively alter the decision-making process. Another reason could be the inconsistencies between the examples provided.

Annotators	LLM method - removed data		
	Zero-shot	One-shot	Few-shot
Annotator 1	0.297	0.299	0.27
Annotator 2	0.086	0.051	0.104
Annotator 3	0.283	0.321	0.283
Annotator 4	0.233	0.26	0.217
Annotator 5	0.161	0.173	0.159

Table 7: LLM methods with removed faulty data entries

As mentioned in Section 3, the argument extraction task involves following multiple steps. To guide the LLM through these steps, chain-of-thought reasoning was used. The same approach of removing data items having outputs that overfit to the examples provided in the prompt was employed.

Ann.	LLM method - with or withoutchain-of-thought					
	Z	Z-C	O	O-C	F	F-C
1	0.297	0.233	0.299	0.225	0.27	0.245
2	0.086	0.374	0.051	0.147	0.104	0.174
3	0.283	0.318	0.321	0.361	0.283	0.311
4	0.233	0.234	0.26	0.222	0.217	0.207
5	0.161	0.271	0.173	0.168	0.159	0.143

Table 8: LLM methods with or without chain-of-thought, where Ann = annotator, Z = zero-shot, O = one-shot, F = few-shot, C = chain-of-thought

As shown in Table 8, applying chain-of-thought reasoning resulted in better outcomes in 8 out of the 15 cases. This approach was particularly effective for Annotator 2. When analyzing the LLM outputs, it appears that chain-of-thought prompting helps the LLM better recognize the possibility of there being no arguments in the text, leading it to label more data items as "None" compared to when this method is not used. As discussed in Subsection 5.2, Annotator 2 frequently labels data items as "None." Thus, the significant increase in cosine similarity score for Annotator 2 can be attributed to the LLM labelling more data items as "None." Therefore, chain-of-thought reasoning proves to be efficient for the task of argument identification. However, no clear trend has been identified regarding its efficiency for premise extraction.

5.4 Majority vote labels

For determining the majority vote labels, a threshold of three annotators has been applied. This means that only labels annotated by at least three annotators are considered when aggregating the labels. If the majority vote results in labels where one label is a subset of another, only the subset label is considered. If no label meets the threshold of being annotated by at least three annotators, the threshold is lowered by one, and the process is repeated.

Instead of individual annotator labels, these labels were used as example input for the LLM in the one-shot chain-of-thoughts and few-shot chain-of-thought approaches. The cosine similarity was used to evaluate the similarity between individual annotator labels and LLm responses.

Ann.	Individual labels vs majority vote			
	O	O-M	F	F-M
1	0.225	0.232	0.245	0.189
2	0.147	0.15	0.174	0.128
3	0.361	0.337	0.311	0.274
4	0.222	0.211	0.207	0.174
5	0.168	0.15	0.143	0.104

Table 9: LLM methods with policy and chain-of-thought, and with or without majority vote, where Ann = annotator, O = one-shot, F = few-shot, M = majority vote

For most annotators, using majority vote labels showed decreased performance compared to individual annotator labels. This result showcases the importance of the examples provided in the prompt of the LLM. With majority vote labels, the examples are not tailored to specific annotators, and, considering the significant disagreement between annotators highlighted in Subsection 5.2, a decrease in similarity between the LLM response and individual annotations is expected. This highlights the subjective nature of the task, as the LLM's outputs are less aligned with individual annotations when using majority vote labels for examples.

5.5 Pairwise chain-of-thought analysis

To further evaluate the subjectivity of the task, the LLM responses for the one-shot chain-of-thought and few-shot chain-of-thought approaches have been compared using pairwise annotator cosine similarity.

Table 10 showcases the cosine similarity score between the LLM responses in the one-shot chain-of-thought prompting approach. For the example provided to the LLM in the prompt, Annotators 1, 4, and 5 had the exact same label, while Annotator 3's label differed slightly. However, Annotator 2 had a completely different annotation compared to the rest. This difference is reflected in the lower similarity scores between Annotator 2 and the others, compared to the other combinations. This highlights the subjectivity of the task: when the LLM is provided with similar examples, it produces similar outputs, but when it is given a considerably different example, the output varies significantly.

Table 11 showcases the cosine similarity score between the LLM responses in the few-shot chain-of-thought prompting approach. Three examples have been provided to the LLM

Annotators	1	2	3	4	5
1	1	0.55	0.71	0.71	0.69
2	0.55	1	0.6	0.59	0.6
3	0.71	0.6	1	0.68	0.68
4	0.71	0.59	0.68	1	0.69
5	0.69	0.6	0.68	0.69	1

Table 10: The one-shot pairwise cosine similarity score between the different LLM responses for each annotator

Annotators	1	2	3	4	5
1	1	0.39	0.57	0.65	0.64
2	0.39	1	0.42	0.43	0.43
3	0.57	0.42	1	0.58	0.61
4	0.65	0.43	0.58	1	0.64
5	0.64	0.43	0.61	0.64	1

Table 11: The few-shot pairwise cosine similarity score between the different LLM responses for each annotator

in the prompt. The first example is the one associated with Table 10, described in the paragraph above. The second example had different labels from most of the annotators. The last example had 100% annotators agreement, with all of the annotators outputting the same label. The results in Table 11 show lower values compared to Table 10, especially for Annotator 2. This indicates that the examples provided in the prompt significantly influence the LLM's output. With more examples that diversify the annotation profile of each annotator, the similarity between LLM responses decreases.

Therefore, this comparison showcases that the argument extraction task is highly dependent on the examples provided in the prompt, making it a subjective task.

6 Discussion

The results highlighted in Section 5 of this research provide insights into the performance and subjectivity of LLMs in the task of argument extraction. This discussion section addresses these findings by showcasing the impact of the different experimental setups, the role of subjectivity in annotation, and the effectiveness of different prompting techniques.

6.1 Efectiveness of policy information

The experiments demonstrated that providing policy information to the LLM generally improves performance, as reflected by the higher cosine similarity scores in 8 out of 15 cases when compared to not providing such information. This suggests that additional context information assists the LLM in producing responses that better align with human annotators.

6.2 Performance of zero-shot, one-shot and few-shot approaches

Contrary to initial expectations, the zero-shot approach often outperformed both the one-shot and few-shot approaches. One cause was due to instances where the LLM overfit to the examples provided in the prompt, as shown in Table 6. After filtering out these faulty data entries, the performance of the one-shot approach surpassed the one of zero-shot. However, few-shot did not improve considerably. This could be attributed to the examples provided creating a more subjective environment for the LLM, resulting in increased randomness in its responses.

6.3 Chain-of-thought reasoning

Applying chain-of-thought reasoning showed mixed results. While it led to better outcomes in 8 out of 15 cases and was particularly effective for Annotator 2, it did not consistently improve performance for all annotators or prompt types. Chain-of-thought reasoning proved to be efficient for the task of argument identification, but it did not show clear improvements for premise extraction.

6.4 Majority vote labels

Using majority vote labels instead of individual annotator labels generally resulted in decreased performance. Knowing the significant disagreement between annotators, this finding showcases the subjectivity of the task by highlighting that LLM responses generated using majority vote labels as examples negatively influence the similarity with individual annotations.

6.5 Subjectivity

The low averaged pairwise F1 score between annotators of **0.2447** showcases significant subjectivity of the annotation task. The variance in annotations, even when being provided with the same instructions, reflects individual differences in interpreting and applying the rules. The subjectivity is further illustrated by the varying number of data entries marked as non-arguments by different annotators, with Annotator 2 marking significantly more entries as non-arguments compared to others.

The pairwise chain-of-thought cosine similarity analysis revealed that the LLM's responses were more consistent with annotators who had similar labels in the examples provided to the LLM. The lower similarity scores for Annotator 2, who had more different annotations, highlight the impact of subjective annotation on LLM performance.

7 Limitations and future work

Several limitations regarding the results highlighted in this paper must be acknowledged. Firstly, the results were limited to evaluating the performance of LLMs using only 50 data row annotations of the dataset. This relatively small sample size limits the generalization of the findings. Secondly, the nondeterministic nature of LLMs does not allow for the exact replication of results. Thirdly, LLMs showcased a tendency to overfit to the specific examples provided in the prompts.

To address these limitations and further advance the research, the following directions are proposed:

- Implementing automatic key point extraction [6] to generate a set of labels to be used for annotation. This would ensure better consistency among annotators and would transform the task into a classification task.
- Generating more annotations for the dataset to obtain universal results.

- Refining the prompts to mitigate the LLM overfitting tendencies.
- In case automatic key point extraction is not implemented, investigating alternative evaluation metrics, such as BERTScore [22], BLEU [23] or ROUGE [24].

8 Responsible Research

In this section, a reflection on the ethical considerations present in this research is conducted, and a discussion on the reproducibility of the methods is introduced.

8.1 Private dataset

Given that the dataset used for this research is private, several safety measures were taken to ensure there was no distribution of information from this dataset. Firstly, the dataset was not added to any university or personal repository. Secondly, the dataset entries were not provided to the browser version of any LLM model to prevent the private data from being used to train the models.

8.2 Annotators' privacy

Given that the dataset was annotated by five different members, each annotator's submission underwent anonymization to disassociate it from the specific annotator. This anonymization process was implemented to ensure that the labelled datasets could not be linked back to any particular annotator. It consisted of randomly assigning a number to each annotated dataset.

8.3 Reproducing the results

Due to the random nature of LLMs, the results of this study cannot be perfectly reproduced. Since the temperature is not set to zero, the Llama2 model is not deterministic, meaning that it is likely to produce different outputs when prompted with the same inputs at different times. However, to mitigate this variance, the results are obtained as an average of running the experiments 10 different times. This approach ensures that the results are distributed over multiple iterations, thereby minimizing the influence of any inherent bias within the LLM across a given iteration.

Another factor that might hinder reproducing the results is the annotator labels. When the same experiments are performed on other data annotations, the results will differ.

9 Conclusions

This paper investigated whether LLMs can effectively identify and classify subjective arguments within deliberative discourse.

First, an annotation strategy for the dataset created by Lisco et. al. [10] was proposed. Following this strategy, annotators first determined whether the data entry was an argument or not, and, if so, they extracted its premises.

Then, three prompting approaches were used to evaluate the performance of the Llama 2 model in argument premise extraction: zero-shot, one-shot and few-shot, with performance evaluated using cosine similarity. Each method was further enhanced with chain-of-thought reasoning and evaluated. It was noted that the LLM tended to overfit to the provided examples in one-shot and few-shot settings. As a contributing factor, zero-shot was unexpectedly performed better than few-shot and one-shot.

Chain-of-thought reasoning proved to be efficient for the task of argument identification, but it did not show clear improvements for premise extraction.

Subjectivity was highlighted in several ways. Firstly, the low averaged pairwise F1 score between annotators reflected variance in annotations, even when given the same instructions, indicating individual differences in interpreting and applying the rules. Secondly, there was a variance between how many data entries each annotator recognized as not being an argument. Thirdly, the pairwise chain-of-thought cosine similarity analysis revealed that the LLM's responses were more consistent with annotators who had similar labels in the examples provided to the LLM, showcasing the subjectivity of the task.

References

[1] R. Shortall, A. Itten, M. v. d. Meer, P. Murukannaiah, and C. Jonker, "Reason against the machine? future directions for mass online deliberation," *Frontiers in Political Science*, vol. 4, oct 2022. [Online]. Available: http://dx.doi.org/10.3389/fpos.2022.946589

[2] D. N. Walton, "What is reasoning? what is an argument?" *The Journal of Philosophy*, vol. 87, no. 8, pp. 399–419, 1990. [Online]. Available: http://www.jstor.org/stable/2026735

[3] J. Fishkin and R. Luskin, "Experimenting with a democratic ideal: Deliberative polling and public opinion." *Acta Politica*, vol. 40, pp. 284–298, 2005. [Online]. Available: https://doi.org/10.1057/palgrave.ap.5500121

[4] R. Siegesmund, "Subjectivity," in *The SAGE Encyclopedia of Qualitative Research Methods*, L. M. Given, Ed. Thousand Oaks, CA: SAGE, 2008, pp. 844–845. [Online]. Available: https://doi.org/10.4135/9781412963909

[5] P. Spada and J. Vreeland, "Who moderates the moderators?" *Journal of Public Deliberation*, vol. 9, 01 2013. [Online]. Available: https://doi.org/10.16997/jdd.165

[6] R. Bar-Haim, L. Eden, R. Friedman, Y. Kantor, D. Lahav, and N. Slonim, "From arguments to key points: Towards automatic argument summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 4029–4039. [Online]. Available: https://aclanthology.org/2020.acl-main.371

[7] G. Chen, L. Cheng, L. A. Tuan, and L. Bing, "Exploring the potential of large language models in computational argumentation," 2024. [Online]. Available: https://arxiv.org/html/2311.09022v2

[8] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, and et al., "Language models are few-shot learners," 2020. [Online]. Available: https://arxiv.org/abs/2005.14165

[9] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," 2023. [Online]. Available: https://arxiv.org/abs/2201.11903

[10] S. L. Spruit and N. Mouter, "1376 residents of súdwest-fryslân about the future energy policy of their municipality: the results of a consultation."

[11] E. Liscio, L. C. Siebert, C. M. Jonker, and P. K. Murukannaiah, "Value preferences estimation and disambiguation in hybrid participatory systems," 2024. [Online]. Available: https://arxiv.org/abs/2402.16751

[12] T. Govier, *A practical study of argument*. Cengage Learning, 2013, pp. 1-21.

[13] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, and et. al, "Llama 2: Open foundation and fine-tuned chat models," 2023. [Online]. Available: https://arxiv.org/abs/2307.09288

[14] M. Peeperkorn, T. Kouwenhoven, D. Brown, and A. Jordanous, "Is temperature the creativity parameter of large language models?" 2024. [Online]. Available: https://arxiv.org/abs/2405.00492

[15] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," 2023. [Online]. Available: https://arxiv.org/abs/2205.11916

[16] W. Gomaa and A. Fahmy, "A survey of text similarity approaches," *international journal of Computer Applications*, vol. 68, 04 2013. [Online]. Available: https://doi.org/10.5120/11638-7118

[17] N. Gisev, J. S. Bell, and T. F. Chen, "Interrater agreement and interrater reliability: Key concepts, approaches, and applications," *Research in Social and Administrative Pharmacy*, vol. 9, no. 3, pp. 330–338, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1551741112000642

[18] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960. [Online]. Available: https://doi.org/10.1177/001316446002000104

[19] J. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, pp. 378–, 11 1971. [Online]. Available: https://doi.org/10.1037/h0031619

[20] G. Hripcsak and A. S. Rothschild, "Agreement, the f-measure, and reliability in information retrieval," *J Am Med Inform Assoc*, vol. 12, no. 3, pp. 296–298, May-Jun 2005. [Online]. Available: https://doi.org/10.1197/jamia.M1733

[21] S. U. S. Chebolu, F. Dernoncourt, N. Lipka, and T. Solorio, "Oats: Opinion aspect target sentiment quadruple extraction dataset for aspect-based sentiment analysis," 2024. [Online]. Available: https://arxiv.org/abs/2309.13297

[22] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," 2020. [Online]. Available: https://arxiv.org/abs/1904.09675

[23] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "Bleu: a method for automatic evaluation of machine translation," 10 2002. [Online]. Available: https://doi.org/10.3115/1073083.1073135

[24] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," 01 2004, p. 10. [Online]. Available: https://aclanthology.org/W04-1013

A Use of Large Language Models

In this research project, ChatGPT[3] was solely used to improve and correct writing mistakes. For this, the following prompts were employed:

```
"Can you make this sentence/paragraph more
    clear: {text to be improved}?"

"Can you make this sentence/paragraph more
    concise: {text to be improved}?"

"Can you review this sentence/paragraph: {
    text to be improved}?"
```

These prompts were used for certain sentences and paragraphs of the paper to ensure clarity and correctness in the writing. All LLM responses were reviewed and only the satisfactory changes were incorporated into the paper.

[3] chat.openai.com