Natural Language Reasoning in Large Language Models: Analysis and Evaluation

Anonymous ACL submission

Abstract

Despite the growing focus on reasoning in Large Language Models (LLMs), particularly through techniques like Chain-of-Thought prompting, there remains limited analysis on whether these models are really reasoning or if performance improvements are mainly due to the context added to the prompt. Furthermore, there is a lack of advanced evaluation tasks assessing natural language reasoning in generative models. This paper addresses a gap in the study of reasoning in LLMs by presenting the first large-scale evaluation of their unconstrained natural language reasoning capabilities based on natural language argumentation. As a result, three main contributions are produced: (i) the formalisation of a new strategy designed to evaluate argumentative reasoning understanding in LLMs: argument-component selection; (ii) the creation of the Argument Reasoning Tasks (ART) dataset, a new benchmark based on argument structures for natural language reasoning; and (iii) an extensive experimental analysis involving four different models, pointing out consistently the important limitations of LLMs on natural language reasoning tasks.

1 Introduction

The question of whether Large Language Models (LLMs) can perform reasoning is a thorny one. Not only have there been a wide range of studies exploring the issue (and coming to wildly different conclusions), but techniques such as Chain of Thought prompting (CoT) (Wei et al., 2022) and multi-hop Question-Answering (Yang et al., 2018; Zhu et al., 2024), that purport to place reasoning at the forefront of LLM interaction, have generated remarkable performance enhancements and demanding challenge tasks (Chu et al., 2024). Coupled with high profile marketing touting LLM reasoning capabilities[1] and anecdotal evidence of both spectacular success and spectacular failure, it is no wonder there has been such an explosion of work in trying to fairly assess reasoning competence in LLMs (Miao et al., 2020; Cobbe et al., 2021; Patel et al., 2021; Talmor et al., 2021; Geva et al., 2021; Mirzadeh et al., 2024; Han et al., 2024; Valmeekam et al., 2024; Mehrafarin et al., 2024; Paruchuri et al., 2024; Tyagi et al., 2024a; Samadarshi et al., 2024; Shiri et al., 2024; Han et al., 2024). To date, however, all of this work has focused on artificial, synthetic, toy problems such as arithmetic, logic puzzles, and theorem-proving. Though a focus on such toy problems offers an opportunity to carefully control variability under laboratory conditions, it also risks seriously misrepresenting LLM performance with respect to realistic human reasoning. Even (Guan et al., 2023), who demonstrate deep weaknesses in current LLM capacity, rest their argument on classical planning, a very narrow and tightly constrained type of reasoning. What is required is a vocabulary, a model, a dataset and a set of tasks that also cover natural, in-situ human reasoning, as it is expressed in language. This is the domain of argumentation theory (van Eemeren et al., 2014), and our goal in this paper is to leverage recent results in the area to equip us with the tools to assess LLM performance in realistic settings.

We address this significant challenge by presenting the first large-scale evaluation of the natural language reasoning capabilities of LLMs based on natural language argumentation. This paper has, therefore, the three following main contributions: (i) we formalise a new task that we define as argument-component selection which is designed to evaluate argumentative reasoning in LLMs; (ii) we create and release publicly the Argument Reasoning Tasks (ART) dataset, a new benchmark for argumentation reasoning consisting of 112,212 multiple-choice questions covering a total of sixteen different tasks addressing structural aspects of argumentation; and (iii) we present a complete set of experiments in-

[1] openai.com/index/learning-to-reason-with-llms/

volving four open- and closed-weight models as well as a thorough analysis of the observed results.

2 Related Work

As pointed out in recent work, CoT reasoning can also be achieved without any prompt engineering, by just modifying the greedy decoding strategy to explore alternative top-k decoding paths (Wang and Zhou, 2024). This finding, despite being presented as that the models reason intrinsically, it can also be interpreted as meaning that the models do not reason at all, they just have different alternative most likely sequence paths learnt from the training data, and tuning the input prompt (or adapting the decoding) allows to select a different decoding path than the one that provides the direct answer. Following this important finding, rather than focusing on how the output of the model is decoded, the focus should be on studying the model's ability to generalise and keep this behaviour labelled as "reasoning" when addressing problems of different nature and involving more complex and realistic reasoning than the ones that are commonly studied in the literature[2].

This aspect has been discussed in recent work (Valmeekam et al., 2024; Wu et al., 2024), where the reported results show that with minimal variations of the *standard* versions of the tasks included in the most popular benchmarks used for reasoning, the performance of LLMs drops significantly. These findings challenge the claims that LLMs can do reasoning and that it allows them to improve their performance in a broad range of tasks.

In addition to CoT and (multi-hop) QA-related tasks, other datasets and tasks to evaluate reasoning in LLMs have been proposed in the past. ProofWriter (Tafjord et al., 2021) presents a dataset to evaluate deductive logical reasoning through formal logic problems. COPA (Roemmele et al., 2011) and its multilingual version XCOPA (Ponti et al., 2020) are two datasets created to evaluate causal reasoning by providing situations and asking to select the most likely outcome to happen according to a cause-effect relationship. In this same direction, SWAG (Zellers et al., 2018) and HellaSWAG (Zellers et al., 2019) introduce two datasets to evaluate commonsense reasoning inference featuring adversarially generated scenarios in which models need to determine the most plausible option. The aNLI (Bhagavatula et al., 2019) dataset is proposed to investigate abductive reasoning. Again, the models are challenged to identify plausible outcomes for incomplete information scenarios. FOLIO (Han et al., 2024) consists of a collection of first order logic statements to evaluate the reasoning capabilities of LLMs. The models are asked to determine the truth values of a set of conclusions given some premises which are presented in both, natural language and first-order logic statements. From the reported results, it is possible to observe how LLMs struggle to solve this task. Finally, it is also worth mentioning other recent approaches, which have proposed the assessment of the reasoning capacities of LLMs based on games such as Minesweeper, grid puzzles, Sudoku or crosswords among others (Li et al., 2024; Tyagi et al., 2024b; Shah et al., 2024; Saha et al., 2024).

We can observe, however, that despite being focused on reasoning-related tasks, none of them address the problem of non-constrained reasoning in natural language (e.g., in argumentation), which is a fundamental aspect for evaluating and understanding the actual natural language reasoning capabilities of LLMs.

3 Theoretical Background

Arguments combine premises and conclusions to create complex reasoning structures. Argument theory distinguishes different structural combinations of premises and conclusions: serial (premises and/or conclusions are supported by premises themselves) (Beardsley, 1950), linked (multiple premises support a conclusion together in a combined inferential step) (Thomas, 1973), convergent (multiple premises independently support the conclusion), and divergent (same premise supports more than one conclusion). With these four types of argument structure, it is possible to analyse and understand argumentation in similar ways as multi-hop and CoT reasoning are commonly studied.

An important challenge in the evaluation of LLMs on argumentation skills is to ensure that the reasoning capacities are assessed instead of the dialogue generation abilities. Their training enables LLMs to create credible textual output based on probable token combinations. In an argument continuation task without sufficient limitations, the model will produce a probable continuation based on the input text. In this case, it is difficult to evaluate the appropriateness of the created continuation.

[2] And that, therefore, will most likely be also included in the training data of the latest versions of the popular LLMs

Our approach solves the evaluation challenge by introducing a multiple-choice task, a setup similar to the ones LSAT tests already used to measure the reasoning abilities of LLMs in logic games (Malik, 2024). By asking for one or more elements from a complex argumentative graph structure, the model needs to identify the correct continuation among a choice of options from the same argumentative context. This requires the ability to follow and reconstruct an implicit reasoning path. Tasks targeting larger chunks of argumentative elements require a model to choose an appropriate sub-structure as continuation, which demands deep understanding of necessary intermediate reasoning steps, similar to complex multi-hop Q&A tasks.

4 Method

Aimed at providing, for the first time, a method to consistently evaluate the natural language reasoning capabilities of LLMs in argumentation, we formulate argument-component selection, consisting of a series of sixteen different argumentative reasoning tasks grouped into four different types of argumentative structures. This way, the proposed method allows us to evaluate the natural language reasoning capabilities of LLMs by asking them to build and reconstruct natural language arguments.

4.1 Task Formulation

An argument is represented as a structure consisting of a sequence of argument components $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$ and the relations of inference and conflict between them $\mathcal{R} = \{\vdash, \multimap\}, \mathcal{R} : A \times A$. Our proposed tasks leverage the argumentative context, $\mathcal{C}$ to predict or generate a required argument component. To facilitate automatic evaluation and ensure consistency, we restrict the proposed tasks to selecting missing components from a predefined set of options. This constraint is crucial as open-ended generation poses challenges for evaluation, given that multiple valid components could fulfil the argument structure. By limiting the options, we allow the model to focus on identifying the most appropriate components while enabling reliable evaluation against gold-standard answers. We therefore define a series of Argumentative Reasoning Tasks as argument-component selection problems, where the model must identify the correct argument component(s) from a set of candidates to meet some set of structural argumentative criteria. The task requires filling the missing components of specific substructures while considering the entire argument as context. Accordingly, the model is provided with an argument as a context $\mathcal{C}$, a partially specified argument substructure (with missing argument components), and a candidate set $\mathcal{U} = \{u_1, u_2, \ldots, u_k\}$, which includes the correct answer $\hat{u}$. The objective is to select the correct missing component $\hat{u}$ by evaluating the candidates for their relevance and alignment with the given context $\mathcal{C}$. This process is formalized as:

$$\hat{u} = \arg\max_{u \in \mathcal{U}} \text{score}(u \mid \mathcal{C}),$$

where $\text{score}(u \mid \mathcal{C})$ measures the semantic and structural fit of the candidate u within the argument substructure. The next section outlines the instantiation of the argument-component selection formulation into the argumentative reasoning tasks included in our proposed evaluation method.

4.2 Argumentive Reasoning Tasks (ART)

We design a series of sixteen tasks based on four different types of argument structures: serial, linked, convergent, and divergent argument. Aimed at easing its understanding, a visual representation of the designed tasks can be found in Appendix A.

4.2.1 Serial Reasoning

In serial argument, an argument relation of inference ($\vdash$) is applied sequentially. The model is tasked with identifying a conclusion, premise, or intermediate step based on the argument component(s) and the entire argument as a context. It includes the following six tasks:

One-hop Conclusion. With a single instance of an argument relation of inference ($\vdash$), between a premise, α and a conclusion $\hat{\beta}$, a set is created of alternative potential conclusions, $\{\beta_1, \ldots, \beta_n\}$ (which when taken together with $\hat{\beta}$ is referred to together as the set B), from which the model must select. Treating the model as a function, f, the inputs are a set of argument components, plus a set of context, $\mathcal{C}$. The argument components in this case are the premise α, and the fact that the role to be played by the model's selection is as the conclusion of a $\vdash$ relation that has α as its premise. This role is expressed in the input by a metavariable X. The model's result is a binding of X to $\hat{\beta}$, one of the elements of B. Formally, given $B = \{\beta_0, \beta_1, \ldots, \beta_n\}$, $f(\{\alpha, \alpha \vdash X\}, \mathcal{C}) = \{X : \hat{\beta}\}$, where $\hat{\beta} \in B$.

One-hop Premise. The task in this case is to identify a premise $\hat{\alpha}$ given a conclusion β, where

$\hat{\alpha}$ supports β ($\hat{\alpha} \vdash \beta$), from a set of alternative potential premises $A = \{\alpha_0, \alpha_1, \ldots, \alpha_n\}$, which includes the target premise $\hat{\alpha}$. Given the inputs β, and the fact that the model's role is to select a premise for a ($\vdash$) relation with β as its conclusion (denoted as Y), along with a set of context $\mathcal{C}$, the model f outputs the selected premise. The model's result is a binding of Y to $\hat{\alpha}$, one of the elements of A. Formally, given $A = \{\alpha_0, \alpha_1, \ldots, \alpha_n\}$, $f(\{\beta, Y \vdash \beta\}, \mathcal{C}) = \{Y : \hat{\alpha}\}$, where $\hat{\alpha} \in A$.

Two-hop Conclusion. In a two-hop argument, there are two sequential inference relations, ($\vdash$). The first is between a premise α and an intermediate conclusion β, and the second is between β and a final conclusion $\hat{\gamma}$. A set of alternative potential conclusions, $\{\gamma_0, \gamma_1, \ldots, \gamma_n\}$ (referred to together with $\hat{\gamma}$ as the set D), is created from which the model must select in the given context $\mathcal{C}$. Formally, given $D = \{\gamma_0, \gamma_1, \ldots, \gamma_n\}$: $f(\{\alpha, \beta, \alpha \vdash \beta, \beta \vdash X\}, \mathcal{C}) = \{X : \hat{\gamma}\}, \quad \text{where } \hat{\gamma} \in D$.

Two-hop Premise. Following a similar formalisation, with two sequential argument relations of inference, ($\vdash$), the first between a premise, $\hat{\alpha}$, and an intermediate conclusion, β, and the second between β and a final conclusion, γ, a set is created of alternative potential premises, $\{\alpha_0, \alpha_1, \ldots, \alpha_n\}$ (referred to together with $\hat{\alpha}$ as the set A), from which the model must select in the given context $\mathcal{C}$. Formally, given $A = \{\alpha_0, \alpha_1, \ldots, \alpha_n\}$,$f(\{X \vdash \beta, \beta, \beta \vdash \gamma, \gamma\}, \mathcal{C}) = \{X : \hat{\alpha}\}, \quad \text{where } \hat{\beta} \in A$.

One-Intermediate Conclusion. Similarly, intermediate conclusion involves two sequential argument relations of inference, ($\vdash$). The first is between a premise α and an intermediate conclusion $\hat{\beta}$, and the second is between $\hat{\beta}$ and a final conclusion γ. A set of alternative potential intermediate conclusions, $\{\beta_0, \beta_1, \ldots, \beta_n\}$ (referred to together with $\hat{\beta}$ as the set B), is created from which the model must select in the given context $\mathcal{C}$. Formally, given $B = \{\beta_0, \beta_1, \ldots, \beta_n\}$, $f(\{\alpha, X \vdash \gamma, \gamma\}, \mathcal{C}) = \{X : \hat{\beta}\}, \quad \text{where } \hat{\beta} \in B$.

Two-Intermediate Conclusions. Two intermediate conclusions involve three sequential argument relations of inference, ($\vdash$). The first is between a premise α and the first intermediate conclusion $\hat{\beta}$, the second is between $\hat{\beta}$ and the second intermediate conclusion $\hat{\gamma}$, and the third is between $\hat{\gamma}$ and the final conclusion ω. Given the context $\mathcal{C}$, the model selects $\hat{\beta}$ and $\hat{\gamma}$ from a set of alternative potential intermediate conclusions, $B = \{\beta_0, \beta_1, \ldots, \beta_n\}$ and $U = \{\gamma_0, \gamma_1, \ldots, \gamma_n\}$, respectively. Formally, given B, U, $f(\{\alpha, \omega, \alpha \vdash X, X \vdash Y, Y \vdash \omega\}, \mathcal{C}) = \{X : \hat{\beta}\}, \{Y : \hat{\gamma}\}, \quad \text{where } \hat{\beta} \in B \text{ and } \hat{\gamma} \in U.$

4.2.2 Linked Reasoning

In a linked argument, there exists a support relation where a conclusion β is supported by a premise α in combination with another premise θ. It involves the following variants.

One Linked Premise. Given the context $\mathcal{C}$, the aim of this task is to identify the premise $\hat{\theta}$, such that $\alpha \wedge \hat{\theta} \vdash \beta$ holds, from a set of alternative potential linked premises, $Z = \{\theta_0, \theta_1, \ldots, \theta_n\}$. Formally, the relation is expressed as, $f(\{\alpha, \beta, \alpha \wedge X \vdash \beta\}, \mathcal{C}) = \{X : \hat{\theta}\}, \quad \text{where } \hat{\theta} \in Z$.

Two Linked Premises. Similarly, in two linked premise, given the context $\mathcal{C}$, the task is to identify both premises $\hat{\alpha}$ and $\hat{\theta}$, such that $\hat{\alpha} \wedge \hat{\theta} \vdash \beta$, from alternative potential linked premises, $A = \{\alpha_0, \alpha_1, \ldots, \alpha_n\}$ and $Z = \{\theta_0, \theta_1, \ldots, \theta_m\}$. Formally, the relation is expressed as, $f(\{\beta, X \wedge Y \vdash \beta\}, \mathcal{C}) = \{(X, Y) : (\hat{\alpha}, \hat{\theta})\}, \quad \text{where } \hat{\alpha} \in A,\ \hat{\theta} \in Z$.

Linked Reasoning Conclusion. Finally, this task aims to identify the conclusion $\hat{\beta}$, such that $\alpha \wedge \theta \vdash \hat{\beta}$ holds, from a set of alternative potential conclusions, $B = \{\beta_0, \beta_1, \ldots, \beta_n\}$, in the given context $\mathcal{C}$. Formally, the relation is expressed as, $f(\{\alpha, \theta, \alpha \wedge \theta \vdash X\}, \mathcal{C}) = \{X : \hat{\beta}\}, \quad \text{where } \hat{\beta} \in B$.

4.2.3 Convergent Reasoning

In a convergent argument, multiple premises (α, θ) independently support a conclusion β. It includes the following variants.

One Convergent Premise. The task is to identify a premise $\hat{\alpha}$ that independently supports β, given the conclusion β and the other premise θ that also independently supports β in the context $\mathcal{C}$. The model selects $\hat{\alpha}$ from a set of alternative potential premises, $A = \{\alpha_0, \alpha_1, \ldots, \alpha_n\}$. Formally, the relation is expressed as, $f(\{\theta, \beta, X \vdash \beta\}, \mathcal{C}) = \{X : \hat{\alpha}\}, \quad \text{where } \hat{\alpha} \in A$.

Two Convergent Premises. Two convergent premises identifies both $\hat{\alpha}$ and $\hat{\theta}$, such that each independently supports β in the given context $\mathcal{C}$. The premises $\hat{\alpha}$ and $\hat{\theta}$ are selected from the sets of alternative potential premises $\{\alpha_0, \alpha_1, \ldots, \alpha_n\}$ and $T = \{\theta_0, \theta_1, \ldots, \theta_m\}$, respectively. Formally, the relation is expressed as, $f(\{\beta, X \vdash \beta, Y \vdash \beta\}, \mathcal{C}) = \{(X, Y) : \hat{\alpha}, \hat{\theta}\}, \quad \text{where } \hat{\alpha} \in A \text{ and } \hat{\theta} \in T$.

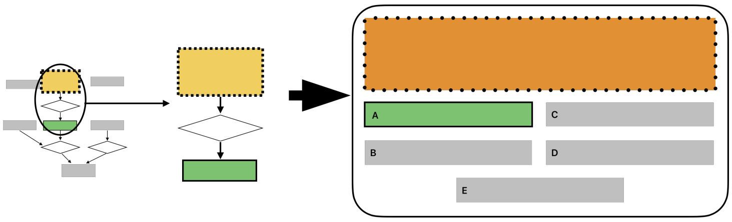

Figure 1: Illustration of the data processing for ART. In the argument graph on the left, substructures of the target task are identified (sub-graph in the middle). Based on these, multiple-choice questions as displayed on the right are created, where the question (orange dotted outlines) includes the argumentative component together (yellow square-outlines) with the relevant context (grey) from the complete graph in the left. The correct answer choice (green full outlines) of the identified argumentative structure is presented alongside with incorrect options sampled from all other nodes in the graph (grey).

Convergent Reasoning Conclusion. Final, this task identifies the conclusion $\hat{\beta}$, which is independently supported by the two premises α and θ. Given the premises α, θ and the context $\mathcal{C}$, the model must select a conclusion $\hat{\beta}$ from a set of potential conclusions B = $\{\beta_0, \beta_1, \ldots, \beta_m\}$. Formally, the relation is expressed as, $f(\{\alpha, \theta, \alpha \vdash X, \theta \vdash X\}, \mathcal{C}) = \{X : \hat{\beta}\}$, where $\hat{\beta} \in B$.

Alternative Hop. Given a premise α, an intermediate conclusion β, and a final conclusion ω, each with their respective argument relations of inference, ($\vdash$), the aim is to find an alternative reasoning chain that leads to ω. This chain should involve an alternative premise $\hat{\theta}$ that supports an intermediate conclusion $\hat{\gamma}$, which in turn leads to the final conclusion ω. Specifically, the model must identify an alternative $\hat{\theta}$ such that $\hat{\theta} \vdash \hat{\gamma}$ and $\hat{\gamma} \vdash \omega$ in the given context $\mathcal{C}$. Formally, let $Z = \{\theta_0, \theta_1, \ldots, \theta_n\}$ be the set of potential alternative premises and $U = \{\gamma_0, \gamma_1, \ldots, \gamma_n\}$ the set of potential intermediate conclusions. The model's task is then to find $\hat{\theta} \in Z$ and $\hat{\gamma} \in U$ that satisfy the relation. This is expressed as, $f(\{\alpha, \beta, \omega, \alpha \vdash \beta, \beta \vdash \omega, X \vdash Y, Y \vdash \omega\}, \mathcal{C}) = \{X : \hat{\theta}\}, \{Y : \hat{\gamma}\}$, where $\hat{\theta} \in Z$ and $\hat{\gamma} \in U$. Here, $\hat{\theta}$ is the selected alternative premise from the set Z and $\hat{\gamma}$ is the selected intermediate conclusion from the set U, such that $\theta \vdash \gamma$ and $\gamma \vdash \omega$ holds true.

4.2.4 Divergent Reasoning

In divergent argument, one premise supports multiple conclusions. It involves the following variants.

One Divergent Reasoning Conclusion. This task identifies one of the conclusions $\hat{\beta}$, $\hat{\gamma}$, which is supported by the premise α. Given the premise α, and one of the conclusions γ and the context $\mathcal{C}$, the model selects $\hat{\beta}$ from a set of potential conclusions $\{B = \beta_0, \beta_1, \ldots, \beta_m\}$. Formally, the relation is expressed as, $f(\{\alpha, \alpha \vdash X, \alpha \vdash \gamma\}, \mathcal{C}) = \{(X : \hat{\gamma}\}$, where $\hat{\beta} \in B$.

Two Divergent Reasoning Conclusions. This task identifies two conclusions $\hat{\beta}$ and $\hat{\gamma}$, both of which are supported by the premise α. Given the premise α and context $\mathcal{C}$, the model selects $\hat{\beta}$ and $\hat{\gamma}$ from a set of potential conclusions $\{B = \beta_0, \beta_1, \ldots, \beta_m\}$ and $Z = \{\gamma_0, \gamma_1, \ldots, \gamma_n\}$. Formally, the relation is expressed as, $f(\{\alpha, \alpha \vdash X, \alpha \vdash Y\}, \mathcal{C}) = \{(X, Y) : \{\hat{\beta}, \hat{\gamma}\}$, where $\hat{\beta} \in B$ and $\hat{\gamma} \in Z$.

Divergent Reasoning Premise. Given the conclusions β and γ within a context $\mathcal{C}$, the model selects $\hat{\alpha}$ from a set of potential premises $A = \{\alpha_1, \ldots, \alpha_n\}$, such that $\hat{\alpha}$ supports both β and γ. Formally, this relation is defined as, $f(\{X \vdash \beta, X \vdash \gamma\}, \mathcal{C}) = \{X : \hat{\alpha}\}$, where $\hat{\alpha} \in A$.

4.3 Data

To create a robust and comprehensive evaluation, we incorporate seven corpora spanning diverse domains and argumentative contexts, covering both monologue and dialogue structures. The corpora include MTC (Peldszus and Stede, 2015), AAEC (Stab and Gurevych, 2017), CDCP (Park and Cardie, 2018), ACSP (Lauscher et al., 2018), AbstRCT (Mayer et al., 2020), US2016 (Visser et al., 2020), and QT30 (Hautli-Janisz et al., 2022).

MTC consists of short argumentative texts originally in German and translated into English, annotated according to Freeman's macro-structural theory of argumentation, with argument relations categorized as supports and attacks. **AAEC** comprises persuasive student essays annotated at the discourse level, identifying argument components (claims and premises) and their argumentative relations as supports and attacks. **CDCP** is a corpus

Tasks		MTC	AAEC	CDCP	ACSP	AbstRCT	US2016	QT30
Type	**Variants**							
Serial	**1H-C**	290	4841	1033	5789	2288	3379	6488
	1H-P	290	4841	1033	5789	2288	3379	6488
	2H-C	57	3279	348	759	327	1009	1118
	2H-P	57	3279	348	759	327	1009	1118
	Int-C	57	3279	348	759	327	1009	1118
	2-Int-C	3	569	89	80	8	249	787
Linked	**1L-P**	17	-	64	-	-	180	511
	2L-P	17	-	64	-	-	180	511
	LR-C	17	-	64	-	-	180	511
Convergent	**1C-P**	96	4735	763	2024	1899	1129	397
	2C-P	96	4735	763	2024	1899	1129	397
	CR-C	96	4735	763	2024	1899	1129	397
	AH	57	3279	348	759	327	1009	1118
Divergent	**1DR-C**	-	-	11	184	48	106	386
	2DR-C	-	-	11	184	48	106	386
	DR-P	-	-	11	184	48	106	386

Table 1: Updated statistics of task types for each dataset. The task variants are defined as follows: 1H-C (One-hop Conclusion), Int-C (Intermediate Conclusion), 2H-P (Two-hop Premise), 2-Int-C (Two-Intermediate Conclusions); 1L-P (One Linked Premise), 2L-P (Two Linked Premises), LR-C (Linked Reasoning Conclusion); 1C-P (One Convergent Premise), 2C-P (Two Convergent Premises), CR-C (Convergent Reasoning Conclusion), AH (Alternative Hop); 1DR-C (One Divergent Reasoning Conclusion), 2DR-C (Two Divergent Reasoning Conclusions) and DR-P (Divergent Reasoning Premise).

of user comments on the Consumer Debt Collection Practices (CDCP) rule, annotated with argumentative structure. It includes two types of support relations, categorised as Reason and Evidence which are consolidated into a single support relation. **ACSP** is a corpus of scientific publications in the field of computer graphics, annotated for argumentative relations, including supports, contradictions, and semantic equivalence. **ABSTRCT** is a corpus of abstracts from randomized controlled trials across various medical domains, annotated to identify argument components and their relations. These relations include support, attack, and partial-attack. **US2016** includes transcripts of debates from the 2016 US presidential election (primary and general) and related Reddit discussions. Annotated using Inference Anchoring Theory (IAT), it captures argumentation and dialogue structures with relations categorised as supports, attacks, and rephrases. Finally, **QT30** contains transcripts from the UK's *Question Time*, a political talk show, also annotated with IAT to identify supports, attacks, and rephrases.

4.4 Data Processing

For each of the sixteen tasks included in our method, we systematically navigate through the argument structures available in the seven corpora, extracting all substructures that conform to the task specifications presented above. The resulting multiple-choice questions are organized into an input set and a corresponding target answer. The input set comprises the involved types of argumentative relations and their corresponding components (excluding the target correct answer), alongside the concatenation of all the sentences surrounding the argument component as the context (C). Four alternative incorrect answer options are randomly selected from other arguments outside of the identified argument substructure. For tasks instantiating the argument-component selection formulation, if multiple correct answers are present in an argument, only one correct answer is included in the list of options, while other correct answers are excluded from the pool of incorrect options. For serial reasoning task types, any reasoning chain involving linked arguments is excluded. This exclusion ensures that the substructure adequately captures the logic of the chain, as partial chains that involve only one argument component do not fully represent the structure of linked reasoning. Figure 1 summarises the data processing steps, in which complex and large argument graphs are converted into five-option multiple-choice questions.

As a result of this process, we present the Argumentative Reasoning Tasks (ART) dataset[3]. The ART dataset consists of a total of 112,212 multiple-choice questions following the sixteen task definitions, which can also be easily implemented as prompts as exemplified in Appendix C. Table 1 depicts the number of questions divided by task and corpora that make up our dataset.

[3]The dataset will be publicly released after the acceptance of this paper under a CC BY-NC-SA 4.0 license.

Dataset	Model	Size	Argument-Component Selection Serial	Linked	Convergent	Divergent
AAEC	**Qwen 2.5**	7B	23.78 ± 13.52	-	10.85 ± 11.50	-
		72B	35.59 ± 13.49	-	18.95 ± 19.37	-
	Llama 3.1	8B	12.23 ± 9.87	-	4.15 ± 3.62	-
		70B	38.77 ± 8.12	-	16.08 ± 20.25	-
	Mistral	7B	29.82 ± 14.12	-	10.4 ± 13.46	-
	GPT	GPT-4o	49.83 ± 17.37	-	35.78 ± 21.50	-
MTC	**Qwen 2.5**	7B	0.2 ± 0.21	-	1.75 ± 2.04	-
		72B	19.51 ± 16.29	-	2.6 ± 3.40	
	Llama 3.1	8B	0.16 ± 0.16	-	1.05 ± 1.50	-
		70B	8.53 ± 11.71	-	5.46 ± 4.56	-
	Mistral	7B	0.16 ± 0.26	-	0.9 ± 1.53	-
	GPT	GPT-4o	49.73 ± 24.36	-	11.36 ± 11.54	-
CDCP	**Qwen 2.5**	7B	29.97 ± 14.84	35.38 ± 25.32	17.45 ± 20.45	0.86 ± 0.80
		72B	50.28 ± 21.52	51.28 ± 16.59	24.68 ± 28.54	1.2 ± 0.61
	Llama 3.1	8B	10.33 ± 7.95	9.23 ± 12.21	5.85 ± 6.66	0.4 ± 0.4
		70B	40.71 ± 17.94	49.74 ± 21.88	21.47 ± 28.40	0.93 ± 0.53
	Mistral	7B	22.97 ± 12.18	12.82 ± 14.94	8.85 ± 12.64	0.26 ± 0.46
	GPT	GPT-4o	65.06 ± 13.41	68.87 ± 14.93	44.94 ± 30.31	7.33 ± 2.52
AbstRCT	**Qwen 2.5**	7B	11.46 ± 6.28	-	14.4 ± 18.73	0.933 ± 0.90
		72B	33.96 ± 19.27	-	29.40 ± 33.71	$1.46 \pm .070$
	Llama 3.1	8B	4.7 ± 3.30	-	8.9 ± 7.01	0.4 ± 0.4
		70B	19.05 ± 19	-	$11.12.86$	1.33 ± 0.80
	Mistral	7B	10.0 ± 5.77	-	6.35 ± 9.19	0.33 ± 0.41
	GPT	GPT-4o	48.61 ± 28.90	-	34.48 ± 29.19	11.4 ± 3.13
ACSP	**Qwen 2.5**	7B	37.13 ± 19.03	-	16.05 ± 15.38	9.13 ± 6.77
		72B	47.31 ± 23.55	-	25.07 ± 15.23	12.8 ± 6.43
	Llama 3.1	8B	12.3 ± 8.25	-	4.5 ± 4.94	2.4 ± 2.42
		70B	39.64 ± 13.76	-	12.433 ± 18.07	8.86 ± 6.10
	Mistral	7B	26.66 ± 13.47	-	12.4 ± 14.18	5.86 ± 5.08
	GPT	GPT-4o	90.47 ± 7.34	-	86.38 ± 3.16	41.45 ± 14.34
US2016	**Qwen 2.5**	7B	34.12 ± 19.53	30.55 ± 19.37	20.45 ± 21.46	7.6 ± 5.4
		72B	49.53 ± 27.61	48.33 ± 18.86	30.34 ± 25.69	10.53 ± 6.26
	Llama 3.1	8B	14.41 ± 6.34	11.66 ± 8.67	9.9 ± 12.58	2.86 ± 2.71
		70B	45.51 ± 25.65	45.18 ± 21.37	26.39 ± 26.64	8.06 ± 5.98
	Mistral	7B	37.95 ± 20.51	20.18 ± 17.84	12.8 ± 15.21	4.53 ± 3.70
	GPT	GPT-4o	58.47 ± 12.94	53.03 ± 9.32	45.85 ± 15.12	37.78 ± 17.21
QT30	**Qwen 2.5**	7B	31.40 ± 16.96	20.76 ± 18.63	11.4 ± 11.10	20 ± 18.11
		72B	42.45 ± 20.84	45.50 ± 16.24	20.33 ± 17.02	29.0 ± 15.77
	Llama 3.1	8B	9.99 ± 5.15	11.50 ± 10.26	5.8 ± 4.48	12.33 ± 13.52
		70B	36.21 ± 15.94	43.38 ± 20.84	18.10 ± 16.59	23.16 ± 17.40
	Mistral	7B	33.98 ± 17.96	20.76 ± 18.63	6.2 ± 8.22	12.4 ± 11.78
	GPT	GPT-4o	53.62 ± 23.80	53.04 ± 18.66	46.69 ± 21.44	41.65 ± 15.34

Table 2: Macro averaged F1-scores and standard deviations for the argument-component selection tasks.

5 Experiments

5.1 Experimental Setup

We evaluate the performance of state-of-the-art models, including Qwen 2.5 (Yang et al., 2024), Llama 3.1 (Touvron et al., 2023), Mistral (Jiang et al., 2023), GPT-4 (Achiam et al., 2023), and o1[4], across a range of complex reasoning tasks in both few-shot and zero-shot settings. The specific prompt templates and model hyperparameters, including temperature, top-p sampling, and inference steps, are detailed in the Appendix B for reproducibility and transparency. For evaluating the models on the ART multiple-choice reasoning tasks, we evaluate model performance using macro averaged F1-score. The code and dataset are available at `https://github.com/ANONYMOUS` (anonymous).

5.2 Results and Discussion

Table 2 reports the macro averaged F1-scores and their standard deviations for each model and type of argument structure. The fine-grained results considering each of the ART tasks independently has been included in Appendix D. Having the random chance baseline (i.e., 20%, one correct answer out of five options) as a reference, we can observe how language models could not consistently provide the correct answers for the ART tasks. This implies that LLMs may not effectively reason or comprehend argumentative reasoning, even if their generated texts resemble reasoning in appearance, as is often observed in text generation tasks. These patterns, which can be mistaken for reasoning ability, result from the model's capacity to produce fluent text rather than from an actual ability to parse or evaluate arguments.

Across all tasks, GPT variants standout, showing better performance compared to the other models. On average GPT-4o achieves 54.38 ± 25.30, 49.52 ± 22.96, 52.60 ± 26.53 and 27 ± 10.52 F1-score on serial, linked, convergent and divergent types of argument structure respectively. Qwen, Mistral, and Llama models' poor performance was consistent across the board. Despite showing a better performance than others, we can also observe higher standard deviations in the GPT-4o results (growing as the performance increases), meaning that there is a big difference in performance between simple and complex versions of the same type of argument structure task (e.g., **1H-C**, **1H-P** versus **2H-C**, **2H-C**).

This observation can be generalised to the rest of the models, which also show significant performance variations across task types and corpora. Generally the models struggle with task types involving argument substructures as the right answers (i.e., **2-Int-C** and **AH**), showing a lower performance than the random baseline. Notably, with the exception of GPT-4o, all other models, regardless of their size, performed near zero F1-score when tasked with selecting alternative reasoning hops (AH) and two intermediate conclusions (2-Int-C). For instance, Qwen 2.5:70B achieves 4.25 ± 4.92, 3.47 ± 4.72 in **AH** and **2-Int-C**, respectively. This highlights a significant limitation in handling complex reasoning structures, even for larger model architectures.

The results for GPT-4o on ACSP constitute significant outliers with a macro-average F1-score of 90.47, 86.38, and 41.45 for serial, linked and divergent types of argument structure respectively. The same model achieves 53.62, 53.04, 41.65 on QT30 for serial, linked and divergent types of argument

[4]`https://openai.com/index/learning-to-reason-with-llms/`

structure respectively. These results would, in principle, mean that GPT-4o is capable of effectively parse and understand natural language reasoning structures in scientific publications. After a deeper analysis on the data we observed, however, that on average ACSP has 324 argument components per argumentative context C, while US2016, QT30, AAEC, MTC, ABstRACT, and CDCP involve 17, 15, 15, 5, 7 and 26, respectively. Since the incorrect answers are randomly selected from C, ACSP provides larger space of candidate answers involving more semantically diverse and distant sets of answers. This allows to distinguish the correct answer by only focusing on semantic features of the text.

5.3 Sensitivity Study

In addition to the discussion about the results achieved by LLMs on ART, we have also analysed the models' sensitivity to variations in settings including model size and prompt template.

Model Size. The assessment of model sizes compares the 70B and 405B parameter versions of Llama 3.1, as well as GPT-4o vs o1-Preview. The parameter sizes for GPT-4o and o1-Preview are undisclosed, but according to OpenAI's release notes, o1-Preview is designed to handle more complex reasoning tasks compared to GPT-4o. Table 3 reports the results of this comparative study on the **2C-P** task, which, as highlighted in the previous results, is among the most challenging. This task requires the correct answer to include two argument components. The results of the model size sensitivity study show that the performance improves with the model size[5]. These findings indicate that the improvement of the task scales with the size of the model. This improvement, however, is still far from claiming a successful performance on the task. Scaling, therefore, seems not to be a solution to problems involving complex reasoning in natural language, having the 405B version of the Llama 3.1 model performing worse than a random baseline. Even o1-preview, a model that has been described as reasoning model to solve hard problems, cannot effectively identify the two correct premises in a convergent argument.

Prompt Template. Finally, we also investigate the influence of the prompt phrasing on the model performance by testing another independently developed prompt. The two prompts were created by two different authors of this paper without being able to see each other's prompt, having only available the formal definition of the selected tasks (i.e., **2H-C**, **2L-P**, **1C-P**, and **DR-C**) presented in Section 4. Table 4 reports the results from this study, showing a very similar performance on both prompts, meaning that the phrasing of the prompt used in our experiment does neither harm nor boost the model performance for the multiple-choice argument-component selection task.

[5] Under the assumption that o1-preview is the largest model tested.

Llama 3.1		**GPT**	
70B	**405B**	**gpt-4o**	**o1-preview**
9.98	18.73	32.18	41.96

Table 3: Sensitivity to model size across different architectures and variants (**2C-P**).

Model	**Prompt-1**	**Prompt-2**
Llama 3.1:70B	16.01	15.40
Mistral	7.25	7.09
Qwen 2.5:72B	16.29	14.61
GPT-4o	34.32	35.78

Table 4: Sensitivity to Prompt: Performance of models on Prompt-1 and Prompt-2 (**2H-C**,**2L-P**, **1CP** and **2DR-C**).

6 Conclusion

In this paper, we push forward the boundaries of knowledge on the reasoning capabilities of LLMs, a controversial and widely debated topic in the last years. We do so by asking a simple yet relevant research question, can LLMs parse and understand argumentative reasoning structures? Given that argumentation is the natural way of reasoning in natural language, if LLMs can reason, they should be able to parse, understand, and build natural language arguments.

From our results, we can observe that not only LLMs are not capable of understanding argumentative reasoning structures (let's not forget that this means reasoning in natural language), but also that in some cases where a slightly more challenging argumentative structure is used, they perform worse than a random baseline. Our findings, therefore highlight the needs of developing challenging tasks to evaluate natural language reasoning, and also question the reasoning capabilities of LLMs, as it has been recently suggested in the literature.

Limitations

Due to limitations in the compute budget, this work assesses very large / expensive models like the 405B parameter version of Llama and o1 only on a limited subset of ART multiple-choice questions. Nevertheless, the reported results indicate important trends, revealing that despite showing a slight increase in performance, they are still not capable of addressing tasks involving complex reasoning.

Further, this paper focuses on the multiple-choice task setup, assuming that this setup does not harm the performance of the model. Future work may investigate the influence of the task setup on the performance, comparing multiple-choice with less guided open answer setups.

Acknowledgments

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Monroe Curtis Beardsley. 1950. *Practical Logic*. Englewood Cliffs, NJ, Prentice-Hall.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2019. Abductive commonsense reasoning. *arXiv preprint arXiv:1908.05739*.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2024. Navigate through enigmatic labyrinth a survey of chain of thought reasoning: Advances, frontiers and future. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1173–1203.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Lin Guan, Karthik Valmeekam, Sarath Sreedharan, and Subbarao Kambhampati. 2023. Leveraging pre-trained large language models to construct and utilize world models for model-based task planning. In *Advances in Neural Information Processing Systems*, volume 36, pages 79081–79094. Curran Associates, Inc.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Wenfei Zhou, James Coady, David Peng, Yujie Qiao, Luke Benson, Lucy Sun, Alexander Wardle-Solano, Hannah Szabó, Ekaterina Zubova, Matthew Burtell, Jonathan Fan, Yixin Liu, Brian Wong, Malcolm Sailor, Ansong Ni, Linyong Nan, Jungo Kasai, Tao Yu, Rui Zhang, Alexander Fabbri, Wojciech Maciej Kryscinski, Semih Yavuz, Ye Liu, Xi Victoria Lin, Shafiq Joty, Yingbo Zhou, Caiming Xiong, Rex Ying, Arman Cohan, and Dragomir Radev. 2024. FOLIO: Natural language reasoning with first-order logic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22017–22031. Association for Computational Linguistics.

Annette Hautli-Janisz, Zlata Kikteva, Wassiliki Siskou, Kamila Gorska, Ray Becker, and Chris Reed. 2022. Qt30: A corpus of argument and conflict in broadcast debate. In *Proceedings of the 13th Language Resources and Evaluation Conference*, pages 3291–3300. European Language Resources Association (ELRA).

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Anne Lauscher, Goran Glavaš, and Simone Paolo Ponzetto. 2018. An argument-annotated corpus of scientific publications. In *Proceedings of the 5th Workshop on Argument Mining*, pages 40–46.

Yinghao Li, Haorui Wang, and Chao Zhang. 2024. Assessing logical puzzle solving in large language models: Insights from a minesweeper case study. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 59–81. Association for Computational Linguistics.

Saumya Malik. 2024. Lost in the logic: An evaluation of large language models' reasoning capabilities on lsat logic games. *arXiv preprint*.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *ECAI 2020*, pages 2108–2115. IOS Press.

Houman Mehrafarin, Arash Eshghi, and Ioannis Konstas. 2024. Reasoning or a semblance of it? a diagnostic study of transitive reasoning in llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11647–11662.

Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing english math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 975–984.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2024. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*.

Joonsuk Park and Claire Cardie. 2018. A corpus of erulemaking user comments for measuring evaluability of arguments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Akshay Paruchuri, Jake Garrison, Shun Liao, John Hernandez, Jacob Sunshine, Tim Althoff, Xin Liu, and Daniel McDuff. 2024. What are the odds? language models are capable of probabilistic reasoning. *arXiv preprint arXiv:2406.12830*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094.

Andreas Peldszus and Manfred Stede. 2015. An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon*, volume 2, pages 801–815.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. Xcopa: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI spring symposium series*.

Soumadeep Saha, Sutanoya Chakraborty, Saptarshi Saha, and Utpal Garain. 2024. Language models are crossword solvers. *arXiv preprint*.

Prisha Samadarshi, Mariam Mustafa, Anushka Kulkarni, Raven Rothkopf, Tuhin Chakrabarty, and Smaranda Muresan. 2024. Connecting the dots: Evaluating abstract reasoning capabilities of llms using the new york times connections word game. *arXiv preprint arXiv:2406.11012*.

Kulin Shah, Nishanth Dikkala, Xin Wang, and Rina Panigrahy. 2024. Causal language modeling can elicit search and reasoning capabilities on logic puzzles. *arXiv preprint*.

Fatemeh Shiri, Xiao-Yu Guo, Mona Far, Xin Yu, Reza Haf, and Yuan-Fang Li. 2024. An empirical analysis on spatial reasoning capabilities of large multimodal models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21440–21455.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. 2021. Proofwriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3621–3634.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2021. Commonsenseqa 2.0: Exposing the limits of AI through gamification. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Stephen Naylor Thomas. 1973. *Practical reasoning in natural language*. Englewood Cliffs, Prentice-Hall.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Nemika Tyagi, Mihir Parmar, Mohith Kulkarni, Aswin Rrv, Nisarg Patel, Mutsumi Nakamura, Arindam Mitra, and Chitta Baral. 2024a. Step-by-step reasoning to solve grid puzzles: Where do llms falter? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19898–19915.

Nemika Tyagi, Mihir Parmar, Mohith Kulkarni, Aswin Rrv, Nisarg Patel, Mutsumi Nakamura, Arindam Mitra, and Chitta Baral. 2024b. Step-by-step reasoning to solve grid puzzles: Where do llms falter? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19898–19915. Association for Computational Linguistics.

Karthik Valmeekam, Kaya Stechly, Atharva Gundawar, and Subbarao Kambhampati. 2024. Planning in strawberry fields: Evaluating and improving the planning and scheduling capabilities of lrm o1. *arXiv preprint arXiv:2410.02162*.

Frans H. van Eemeren, Bart Garssen, Erik C. W. Krabbe, A. Francisca Snoeck Henkemans, Bart Verheij, and Jean H. M. Wagemans. 2014. *Handbook of Argumentation Theory*. Springer.

Jacky Visser, Barbara Konat, Rory Duthie, Marcin Koszowy, Katarzyna Budzynska, and Chris Reed. 2020. Argumentation in the 2016 us presidential elections:

annotated corpora of television debates and social media reaction. *Language Resources and Evaluation*, 54(1):123–154.

Xuezhi Wang and Denny Zhou. 2024. Chain-of-thought reasoning without prompting. *arXiv preprint arXiv:2402.10200*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2024. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1819–1862.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Andrew Zhu, Alyssa Hwang, Liam Dugan, and Chris Callison-Burch. 2024. Fanoutqa: A multi-hop, multi-document question answering benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 18–37.

A Task Visualisation

To simplify the understanding of the task formalisations in Section 4, Figure 2 depicts a sub-set of tasks from the ART dataset including **1H-C**, **1Int-C**, **2H-P**, **2H-C**, **1DR-C**, **2DR-C**, **AH**, and **2Int-C**.

B Hyper-Parameters

We utilize the LLaMA 3.1 model in its 8B, 70B, and 405B configurations (Touvron et al., 2023), accessed through the `Ollama` library[6]. Additionally, the 7B configuration of the Mistral model (Jiang et al., 2023) is employed, also via the `Ollama` library[7]. Furthermore, we use the 7B and 72B versions of the Qwen 2.5 model[8], accessed through the `Ollama` library[9]. For GPT variants, we rely on the API provided by OpenAI for interacting with the GPT-4o and o1-Preview models. Across the models we use default parameters including the temperature and top_k predictions. We do not perform any finetuning and only apply prompting to off-the-shelf models.

C Prompt Templates

Aimed at improving the transparency and reproducibility of the results reported in this paper, Table 21 contains the templates of the prompts that we used for the different tasks included in ART.

D Complete Results

This appendix section contains the fine-grained results of the LLMs on the sixteen tasks included in ART.

D.1 Serial

- One-hop conclusion (**1H-C**): Table 5.
- One-hop premise (**1H-P**): Table 6.
- Two-hop conclusion (**2H-C**): Table 7.
- Two-hop premise (**2H-P**): Table 8.
- Intermediate conclusion (**Int-C**): Table 9.
- Two intermediate conclusions (**2-Int-C**): Table 10.

D.2 Linked

- One linked premise (**1L-P**): Table 11.
- Two linked premises (**2L-P**): Table 12.
- Linked reasoning conclusion (**LR-C**): Table 13.

[6] https://ollama.com/library/llama3.1
[7] https://ollama.com/library/mistral
[8] https://github.com/QwenLM/Qwen
[9] https://ollama.com/library/qwen2.5

D.3 Convergent

- One convergent premise (**1C-P**): Table 14.
- Two convergent premises (**2C-P**): Table 15.
- Convergent reasoning conclusion (**CR-C**): Table 16.
- Alternative Hop (**AH**): Table 17.

D.4 Divergent

- One divergent reasoning conclusion (**1DR-C**): Table 18.
- Two divergent reasoning conclusions (**2DR-C**): Table 19.
- Divergent reasoning premise (**DR-P**): Table 20.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	68.90	74.34	98.46	74.98	62.21	69.12	69.66
llama3.1:70b	34.80	24.60	37.80	43.60	0.40	34.80	-
llama3.1:8b	19.00	5.60	17.40	15.80	0.20	15.83	14.81
mistral	42.40	17.80	29.80	33.10	0.60	50.27	54.33
qwen2.5:72b	43.14	59.90	54.41	67.83	33.33	55.21	69.19
qwen2.5:7b	32.40	17.40	40.20	40.80	0.40	50.27	54.33

Table 5: 1H-C

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	68.12	71.34	93.41	71.87	58.12	68.90	68.23
llama3.1:70b	42.4	15.2	51.6	48.4	0.4	49.01	67.6
llama3.1:8b	25	8.2	17.8	22.4	0.4	10.733	10.218
mistral	27.6	10	36.4	28.6	0	40.25	47.2
qwen2.5:72b	34.79	35.64	52.96	62.02	33.33	52.06	63.60
qwen2.5:7b	37.2	14.2	48	42.4	0.4	42.6	47.65

Table 6: 1H-P

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	25.46	37.62	87.747	53.623	61.66	48.83	54.83
llama3.1:70b	51.00	36.00	44.40	32.40	0.40	35.60	35.00
llama3.1:8b	18.00	8.00	22.00	9.80	0.20	14.20	15.40
mistral	20.50	11.40	29.20	26.40	0.40	32.80	32.20
qwen2.5:72b	45.00	33.00	54.60	37.60	0.40	38.20	32.00
qwen2.5:7b	33.00	12.00	44.60	32.60	0.40	30.20	22.20

Table 7: 2H-C.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	41.00	39.10	79.31	70.43	33.33	54.43	56.08
llama3.1:70b	35.59	38.11	40.97	58.84	33.33	42.44	59.62
llama3.1:8b	8.00	4.60	10.20	9.40	0.00	9.5707	20.00
mistral	24.80	9.00	29.00	22.60	0.00	32.46	35.76
qwen2.5:72b	32.20	38.11	50.98	57.10	33.33	49.64	62.81
qwen2.5:7b	17.60	9.40	33.60	25.80	0.00	32.46	35.76

Table 8: 2H-P.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	55.88	69.24	97.50	76.16	33.33	71.67	66.66
llama3.1:70b	30.07	49.01	49.54	52.17	16.67	47.57	60.10
llama3.1:8b	3.40	1.80	6.40	4.60	0.00	11.90	10.84
mistral	33.80	11.80	35.60	24.80	0.00	47.60	57.00
qwen2.5:72b	38.46	37.13	69.70	64.93	16.67	57.46	68.79
qwen2.5:7b	21.80	15.80	55.20	36.00	0.00	32.40	43.59

Table 9: Int-C.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	39.65	0	86.42	43.33	0	8.78	35.40
Llama 3.1 70B	0.53	0	13.58	8.89	0	2.16	5.24
Llama 3.1 8B	0	0	0	0	0	0.13	0
Mistral	0.70	0	0	2.22	0	0.51	1.21
Qwen 2.5 72B	7.89	0	1.23	12.22	0	2.16	0.81
Qwen 2.5 7B	0.70	0	1.23	2.22	0	0.51	1.21

Table 10: 2-Int-C.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	–	–	–	79.87	–	65.23	73.81
llama3.1:70b	–	–	–	64.62	–	57.54	58.89
llama3.1:8b	–	–	–	4.62	–	13.49	17.22
mistral	–	–	–	9.23	–	25.60	26.67
qwen2.5:7b	–	–	–	49.23	–	39.68	43.89
qwen2.5:72b	–	–	–	58.46	–	53.77	57.22

Table 11: 1L-P

	AAEC	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	–	–	–	51.87	–	31.56	58.12
llama3.1:70b	–	–	–	24.62	–	19.44	20.56
llama3.1:8b	–	–	–	0	–	0.40	1.67
mistral	–	–	–	0	–	0.20	0
qwen2.5:7b	–	–	–	6.15	–	9.33	8.33

Table 12: 2L-P

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**Microtext**	**QT30**	**US2016**
GPT-4o	–	–	–	74.87	–	62.34	57.23
llama3.1:70b	–	–	–	60.00	–	53.17	56.11
llama3.1:8b	–	–	–	23.08	–	20.63	16.11
mistral	–	–	–	29.23	–	36.51	33.89
qwen2.5:7b	–	–	–	50.77	–	39.29	39.44
qwen2.5:72b	–	–	–	63.08	–	55.95	61.11

Table 13: LR-C.

Model	**AAEC**	**ABstRACT**	**ACSP**	**CDCP**	**MTC**	**QT30**	**US2016**
GPT-4o	44.68	23.44	89.45	46.00	3.46	54.98	51.35
llama3.1:70b	15.80	18.80	22.40	17.80	1.40	18.60	29.20
llama3.1:8b	4.40	14.00	6.40	6.40	1.00	7.00	9.40
mistral	9.00	2.40	22.60	7.20	0.20	5.60	18.00
qwen2.5:72b	22.20	21.20	34.60	24.80	1.60	27.00	37.00
qwen2.5:7b	16.60	12.40	26.80	20.80	1.80	14.60	27.20

Table 14: 1C-P

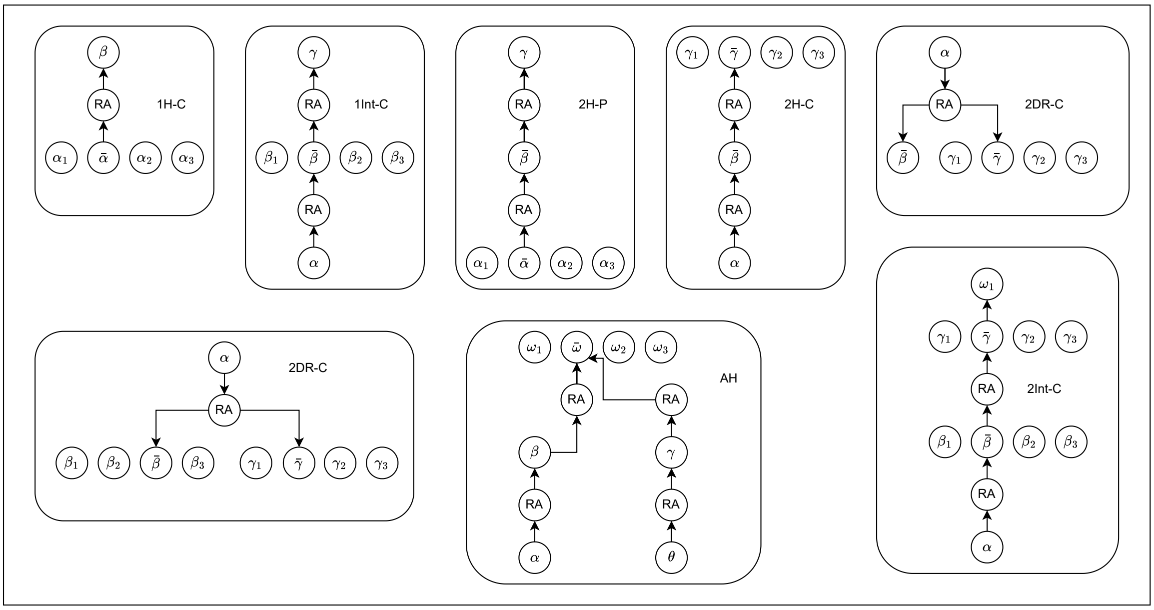

Figure 2: Illustration of selected task types highlighting serial, linked, convergent, and divergent argument structures. The figure includes task types involving single argument components, two argument components, and substructures such as alternative reasoning and two intermediate conclusions.

Model	AAEC	ABstRACT	ACSP	CDCP	MTC	QT30	US2016
GPT-4o	20.88	16.90	85.63	18.85	17.65	28.39	36.99
llama3.1:70b	8.34	14.58	3.90	5.50	9.80	13.82	13.98
llama3.1:8b	3.40	6.60	1.00	2.00	0.00	5.40	2.40
mistral	2.80	3.00	0.40	1.00	0.20	2.80	1.20
qwen2.5:72b	9.00	15.40	10.60	5.40	1.20	11.20	12.80
qwen2.5:7b	2.60	3.80	6.00	4.00	0.60	5.60	7.00

Table 15: 2C-P

Model	AAEC	ABstRACT	ACSP	CDCP	MTC	QT30	US2016
GPT-4o	61.898	78.09	88.34	87.13866121	24.34	73.13	64.45
llama3.1:70b	40.2	78.60	33.1	62.6	6.6	40	62.2
llama3.1:8b	8.8	15	10.6	15	3.1	10.8	27.8
mistral	29.8	20	26.6	27.2	3.1	18.6	32
qwen2.5:72b	44.6	78.6	41.4	65	7.6	40.8	63.8
qwen2.5:7b	24.2	41.4	31.4	45	4.6	25.4	47.6

Table 16: CR-C.

Model	AAEC	ABstRACT	ACSP	CDCP	MTC	QT30	US2016
GPT-4o	15.68	19.51	82.12	27.79	0	30.19	30.63
llama3.1:70b	0	0	0.2	0	0	0	0.2
llama3.1:8b	0	0	0	0	0	0	0
mistral	0	0	0	0	0	0	0
qwen2.5:72b	0	2.44	13.69	3.52	0	2.36	7.79
qwen2.5:7b	0	0	0	0	0	0	0

Table 17: AH.

	AAEC	ABstRACT	ACSP	CDCP	Microtext	QT30	US2016
LLAMA3.1:70B	-	1.6	9.2	1	-	23	9.2
LLAMA3.1:8B	-	0.4	2	0.4	-	10.2	3.2
Mistral	-	0.2	8.8	0	-	10	4.6
Qwen2.5	-	1	12	1	-	16.8	7.6
Qwen2.5:72B	-	2.2	18	1.4	-	35.6	11.6
GPT-4o	-	15.4	52.34	9.34	-	51.34	48.34

Table 18: DR-C.

Model	AAEC	ABstRACT	ACSP	CDCP	MTC	QT30	US2016
llama3.1:70b	-	0.2	2.6	0.4	-	6	1.6
llama3.1:8b	-	0	0.2	0	-	0	0
mistral	-	0	0	0	-	2	0.8
qwen2.5	-	0	1.4	0	-	4.8	2.2
qwen2.5:72b	-	0.8	5.6	0.6	-	11	3.8
GPT-4o	-	7.8	26.45	4.56	-	24.56	17.23

Table 19: 2DR-C

Model	AAEC	ABstRACT	ACSP	CDCP	MTC	QT30	US2016
LLAMA3.1:70B	-	1.6	14.8	1.4	-	40.8	13.4
LLAMA3.1:8B	-	0.8	5.0	0.8	-	26.8	5.4
Mistral	-	0.8	8.8	0.8	-	25.2	8.2
Qwen2.5	-	1.8	14.0	1.6	-	40.4	13.0
Qwen2.5:72B	-	1.4	14.8	1.8	-	40.4	16.2
GPT-4o	-	10.8	45.6	8.32	-	48.5	46.23

Table 20: DR-P.

Task Type	Prompt
1H-C	A one-hop argument involves a single inference step where a **Premise** directly supports **Conclusion**. Consider the following argument: '{argument}'. Given the **Premise: '{premise}'**, your task is to identify which of the following options represents the **Conclusion** that is directly supported by the **Premise**.
1H-P	A one-hop argument consists of a single inference step where a **Premise** directly supports **Conclusion**. Consider the following argument: '{argument}'. Given the **Conclusion: '{conclusion}'**, your task is to identify which of the following options can serve as the **Premise** that supports this **Conclusion**.
1Int-C	A two-hop serial argument involves two inference steps: a **Premise** supports **Conclusion 1** (the intermediate conclusion), and **Conclusion 1** further supports a final **Conclusion 2** in a chain. Consider the following argument: '{argument}'. Given the **Premise: '{premise}'**, your task is to identify which of the following options can serve as **Conclusion 1** that connects the **Premise** to **Conclusion 2: '{conclusion_2}'**.
2H-C	A two-hop serial argument involves two inference steps: a **Premise** supports **Conclusion 1** (the intermediate conclusion), and **Conclusion 1** further supports a final **Conclusion 2** in a chain. Consider the following argument: '{argument}'. Given the **Premise: '{premise}'** which supports **Conclusion 1: '{conclusion_1}'**, your task is to identify which of the following options can serve as the final **Conclusion 2** that is further supported by **Conclusion 1**.
2H-P	A two-hop serial argument involves two inference steps: a **Premise** supports **Conclusion 1** (the intermediate conclusion), and **Conclusion 1** further supports a final **Conclusion 2** in a chain. Consider the following argument: '{argument}'. Given **Conclusion 2: '{conclusion_2}'** which is supported by **Conclusion 1: '{conclusion_1}'**, your task is to identify which of the following options can serve as the **Premise** that supports **Conclusion 1**.
2Int-C	A three-hop serial argument involves three inference steps: a **Premise** supports **Conclusion 1**, **Conclusion 1** supports **Conclusion 2**, and **Conclusion 2** further supports **Conclusion 3** in a chain. Consider the following argument: '{argument}'. Given the **Premise: '{premise}'**, your task is to identify which one of the following options represents **Conclusion 1** that is logically supported by the **Premise**, and which one represents **Conclusion 2** that is supported by **Conclusion 1**, such that **Conclusion 2** further supports **Conclusion 3: '{conclusion_3}'** in the chain. The missing argument components must logically align with the provided context, ensuring that **Conclusion 1** is supported by the **Premise**, **Conclusion 2** is supported by **Conclusion 1**, and **Conclusion 3** is supported by **Conclusion 2**.
1L-P	In a linked argument, a **conclusion** is supported jointly by multiple premises (**Premise 1**, **Premise 2**). Consider the following argument: '{argument}'. Given the **Premise 1: '{premise_1}'**, your task is to identify which of the following options represents the **Premise 2** that, when used jointly with **Premise 1**, directly supports the **conclusion**: '{conclusion}'.
2L-P	In two linked premises, a **conclusion** is supported jointly by **Premise 1** and **Premise 2**. Consider the following argument: '{argument}'. Identify which one of the following represents **Premise 1** and **Premise 2**, from the given set of alternatives, jointly supporting the **conclusion**: '{conclusion}'.
LR-C	In a linked reasoning argument, a conclusion is supported jointly by **Premise 1** and **Premise 2**. Consider the following argument: '{argument}'. Given the **Premise 1: '{premise_1}'** and **Premise 2: '{premise_2}'**, your task is to identify which one of the following options represents the **Conclusion** that is jointly supported by **Premise 1** and **Premise 2**.
1C-P	In a Convergent argument, a **conclusion** is independently supported by multiple premises (**Premise 1**, **Premise 2**). Consider the following argument: '{argument}'. Given the **Premise 1: '{premise_1}'**, your task is to identify which of the following options represents the **Premise 2** that also independently supports the **Conclusion**: '{conclusion}'.
2C-P	In a Convergent argument, a **conclusion** is independently supported by **Premise 1** and **Premise 2**. Consider the following argument: '{argument}'. Identify which one of the following represents **Premise 1** and **Premise 2**, from the given set of alternatives, independently supporting the **Conclusion**: '{conclusion}'.
CR-C	In a Convergent reasoning argument, a **Conclusion** is independently supported by **Premise 1** and **Premise 2**. Consider the following argument: '{argument}'. Given the **Premise 1: '{premise_1}'** and **Premise 2: '{premise_2}'**, your task is to identify which one of the following options represents the **Conclusion** that is independently supported by **Premise 1** and **Premise 2**.
1DR-C	In divergent reasoning, a **Premise** supports multiple **Conclusions** (**Conclusion 1** and **Conclusion 2**). Consider the following argument: '{argument}'. Given the **Premise: '{premise}'**, and **Conclusion 1: '{conclusion_1}'**, your task is to identify which one of the following options represents the **Conclusion 2** that is also supported by the **Premise**.
2DR-C	In divergent reasoning, a **Premise** supports multiple **Conclusions** (**Conclusion 1** and **Conclusion 2**). Consider the following argument: '{argument}'. Given the **Premise: '{premise}'**, your task is to identify which one of the following represents **Conclusion 1** and **Conclusion 2**, from the given set of alternatives, that are supported by the **Premise** in the provided argument.
DR-P	In divergent reasoning, a **Premise** supports multiple **Conclusions** (**Conclusion 1** and **Conclusion 2**). Consider the following argument: '{argument}'. Given the **Conclusion 1: '{conclusion_1}'** and **Conclusion 2: '{conclusion_2}'**, your task is to identify the **Premise** that supports both **Conclusion 1** and **Conclusion 2** in the provided argument.

Table 21: Task Types and Corresponding Prompts.