MultiLogicNMR(er): A Benchmark and Neural-Symbolic Framework for Non-monotonic Reasoning Tasks with Multiple Extensions

Anonymous ACL submission

Abstract

Non-monotonic reasoning refers to the fact that conclusions may be invalidated by new information. It is widely used in daily life and legal reasoning. An important concept in non-monotonic reasoning is that of an extension, which can be interpreted as a set of plausible conclusions. A default theory may have multiple extensions, and there are two reasoning modes – skeptical and credulous reasoning depending on whether to believe facts in all extensions or any one extension. In 2022, Xiu et al. proposed a non-monotonic reasoning dataset LogicNMR, which, however, only involves non-monotonic reasoning with single extensions. In this paper, we first synthesized a non-monotonic reasoning dataset with multiple extensions, MultiLogicNMR. To evaluate the generalization and robustness of models, we constructed two variants of the dataset with more extensions or more text diversity. In the experimental part, we systematically evaluate the prompting and fine-tuning methods and the results show that LLMs do not really understand non-monotonic reasoning. Finally, inspired by a symbolic solving algorithm, we propose a neural-symbolic framework, MultiLogicNMRer, for non-monotonic reasoning with multiple extensions. Experimental results show that the proposed MultiLogicNMRer significantly outperforms the prompting methods, even some fine-tuning methods. All the code and datasets are publicly available[1].

1 Introduction

Logical reasoning is a fundamental aspect of human intelligence, and plays a central role in intelligent systems for problem-solving and decision-making (Brachman and Levesque, 2004). Recently, large language models (LLMs) have made remarkable progress in NLP and related fields, demonstrating certain reasoning abilities (Wei et al., 2022).

[1]https://anonymous.4open.science/r/NMRer-C5B3

Facts: Toby is not noisy. Toby is not intelligent. Toby is not alert. Toby is important. Toby is not gorgeous.
Default Rules: If someoneA is not intelligent then he is delicious, unless he is not drab. If someoneA is delicious and not alert then he is huge, unless he is grieving. If someoneA is not noisy and delicious then he is different, unless he is dead. If someoneA is not gorgeous then he is grieving, unless he is huge.

The number of Extensions is: 2
E1: [Toby is delicious. Toby is grieving. Toby is different.]
E2: [Toby is delicious. Toby is huge. Toby is different.]

Question	Answer
Q1: Toby is delicious.	**A1**: True
Q2: Toby is not huge.	**A2**: Unknown
Q3: Toby is not different.	**A3**: False

Figure 1: A simplified example of skeptical reasoning mode in the MultiLogicNMR dataset.

Naturally, logical reasoning over natural language has received much attention in recent years. Various datasets have been proposed, such as the deductive reasoning datasets: RuleTaker (Clark et al., 2020), ProntoQA (Saparov and He, 2023), EntailmentBank (Dalvi et al., 2021), FOLIO (Han et al., 2022), the inductive reasoning datasets bAbI (Weston et al., 2016) and CLUTRR (Sinha et al., 2019), the abductive reasoning dataset AlphaNLI (Bhagavatula et al., 2020), and the reading comprehension datasets LogiQA (Liu et al., 2020) and ReClor (Yu et al., 2020). Various methods for logical reasoning over natural language have been explored, including fine-tuning methods (Zelikman et al., 2022), prompting methods (Kojima et al., 2022; Wei et al., 2022), and neural-symbolic methods (Hao et al., 2023; Olausson et al., 2023).

Non-monotonic reasoning (Lukaszewicz, 1990), also called defeasible reasoning, is one of the important reasoning modes in logic, and has been extensively studied in classical AI. Generally, non-monotonic reasoning refers to the fact that conclusions may be invalidated by new information. Most of what we learn about the world is in terms of default rules, properties that hold "in general",

but with exceptional cases. Reiter (1980) proposed default logic to formalize non-monotonic reasoning. Given a default theory, that is, a set of facts and default rules, the key to default reasoning is to find all extensions, where an extension is a set of plausible conclusions. For example, we can have a default theory: "John is a professor. John is a chair. A professor usually teaches. A chair usually does not teach." This default theory has two extensions: in one, John is a typical professor and so teaches; in the other, he is a typical chair and does not teach. There are two reasoning modes in default logic: skeptical reasoning only believes in common facts in all extensions, while credulous reasoning believes in facts in any one extension (van Harmelen et al., 2008). Thus, in the above example, by credulous reasoning, John teaches, which does not hold by skeptical reasoning. Non-monotonic reasoning has a wide range of application scenarios, such as reasoning based on conflict knowledge base (Liu et al., 2024), daily decision-making (Szalas, 2019) and legal reasoning (Calegari et al., 2019).

Despite its importance, non-monotonic reasoning based on LLMs has been under-explored. Preliminary investigations have proposed some non-monotonic reasoning benchmarks and used prompt-based and fine-tuning methods to evaluate LLMs' non-monotonic reasoning abilities. Through crowdsourcing, Rudinger et al. (2020) used NLI datasets to build a non-monotonic reasoning benchmark δ-NLI. However, δ-NLI entangles non-monotonic reasoning with commonsense reasoning. To disentangle the two, Xiu et al. (2022) synthesized a non-monotonic reasoning benchmark LogicNMR with explicit facts and rules. However, LogicNMR only involves non-monotonic reasoning with single extension. Recently, Parmar et al. (2024) proposed a comprehensive benchmark LogicBench, including non-monotonic reasoning. However, they focus on eight types of default rules, where each sample only contains a small number of default rules.

This paper explores LLM-based non-monotonic reasoning with multiple extensions. We first propose a benchmark MultiLogicNMR for such reasoning. Figure 1 shows a simplified sample in skeptical reasoning mode from the dataset. The question "Toby is delicious" appears in both extensions $E1$ and $E2$, so the answer is "True". However, the question "Toby is not huge" only appears in $E2$, so the answer is "Unknown". Secondly, we use MultiLogicNMR to explore the non-monotonic reasoning capabilities of multiple open-source and closed-source LLMs in skeptical and credulous reasoning. To evaluate the generalization of LLMs, we also constructed a dataset MultiLogicNMR_OOD involving more extensions. To evaluate the robustness of models, we generated the MultiLogicNMR_NL dataset by rewriting MultiLogicNMR using LLMs to increase its text diversity. Finally, inspired by a symbolic solving algorithm, we propose a neural-symbolic framework, MultiLogicNMRer, for non-monotonic reasoning with multiple extensions. The main idea is to plug into the symbolic framework various modules to use prompt-based LLMs to perform single-step reasoning.

The main contributions of this paper include the following points:

- Non-monotonic reasoning benchmarks with multiple extensions were generated based on automatic synthesis and LLMs. The benchmarks are used to systematically explore the non-monotonic reasoning capabilities of LLMs in skeptical and credulous reasoning modes, respectively.
- We propose a neural-symbolic framework, **MultiLogicNMRer**, for multiple-extension non-monotonic reasoning tasks. MultiLogicNMRer generates all the extensions to answer the question.
- Experiments show that LLMs do not truly master non-monotonic reasoning. However, the proposed MultiLogicNMRer significantly outperforms the prompting methods, even some fine-tuning methods.

2 Related Work

Preliminaries: Similar to LogicNMR (Xiu et al., 2022), we use default logic (DL) as the formal language for MultiLogicNMR in this work. The default rule is of the form: $\alpha : \beta_1, \beta_2, \ldots, \beta_m/\gamma$, where α, β_i and γ are formulas in first-order logic, α is called the prerequisite, $\beta_1, \beta_2, \ldots, \beta_m$ are the justifications, and γ is the conclusion. This default rule can be interpreted as: if the prerequisite fact α is believed, and each justification $\beta_1, \beta_2, \ldots, \beta_m$ can be consistently believed, then γ can be deduced. A default theory $Gamma$ can be denoted as $\Gamma = \langle D, W \rangle$, where D is the set of default rules and W is the set of facts. A set of sentences E is an extension of $\Gamma = \langle D, W \rangle$ iff for every sentence π, E satisfies $\pi \in E$ iff $W \cup \Delta \models \pi$, where $\Delta = \{\gamma | \alpha : \beta/\gamma \in D, \alpha \in E, \neg\beta \notin E\}$.

So, an extension E is the set of entailments of $\{W \cup \gamma\}$, where the γ are consequents from D. For example: given a default rule set $D = \{prof(x) : teaches(x) \,/\, teaches(x),\ chair(x) : \neg teaches(x) \,/\, \neg teaches(x)\}$ and a fact set $W = \{prof(J),\ chair(J)\}$, this default theory has two extensions $E_1 = \{prof(J),\ chair(J),\ teaches(J)\}$, $E_2 = \{prof(J),\ chair(J),\ \neg teaches(J)\}$.

Non-monotonic Reasoning over Natural Language: In addition to the three works discussed in the introduction, there are some other preliminary explorations into the non-monotonic reasoning capabilities of LLMs. Antoniou and Batsakis (2023) conducted a preliminary evaluation of the non-monotonic reasoning capabilities of the GPT3.5 on ten classic examples. However, GPT3.5 still has a big gap compared to symbolic solvers. Recently, Leidinger et al. (2024) focus on the non-monotonic reasoning capabilities of LLMs in the context of generics, they show that LLMs fail to reason consistently and robustly when adding supporting or unrelated exemplars. In addition, Kazemi et al. (2023) proposed a defeasible reasoning benchmark, BoardGameQA, which implements defeasible reasoning on conflicting knowledge bases through rule-based priority. However, the defeasibility in BoardGameQA is caused by conflicting rules. In this paper, the samples generated involve multiple default rules and may generate multiple extensions.

Prompt-based and fine-tuning methods are often used to evaluate the non-monotonic reasoning abilities of LLMs (Rudinger et al., 2020; Leidinger et al., 2024). The prompt-based method requires model to generate answers to questions given instructions, including standard zero-shot prompting (Kojima et al., 2022), few-shot prompting (Brown et al., 2020) and chain-of-thought (CoT) prompting (Wei et al., 2022). In addition, the fine-tuning method usually refers to fine-tuning model parameters in a fully supervised manner based on samples from downstream tasks (Xiu et al., 2022). There are some parameter-efficient fine-tuning methods proposed, such as adapter tuning (Hu et al., 2022) and so on.

Neural-symbolic logical solvers based on LLMs: Neural-symbolic frameworks have been widely used in logical reasoning tasks, mainly including those based on search and external solvers (Kautz, 2022; Xu et al., 2024). Search-based neural-symbolic methods usually use LLMs for single-step reasoning in classical search algorithms (Hong et al., 2022). For example, Hao et al. (2023) combined LLMs and Monte Carlo tree search (Browne and et. al., 2012) to achieve logical reasoning. The external-solver-based neural-symbolic framework use external solvers to solve problems in formal language translated by LLMs (Coppolillo et al., 2024; Olausson et al., 2023). For example, Pan et al. (2023) combined a symbolic solver and LLMs, providing an effective way to achieve faithful logical reasoning. However, the formal language translated by LLMs is prone to grammatical errors and information loss. In this paper, we build a neural-symbolic framework, MultiLogicNMRer, for non-monotonic reasoning based on a symbolic-solving algorithm.

3 Datasets Generation

In this paper, we first automatically synthesize the MultiLogicNMR dataset to investigate the non-monotonic reasoning capabilities of LLMs in both skeptical and credulous reasoning. The algorithm 1 in Appendix A.1 gives the process of generating the MultiLogicNMR. The construction of the MultiLogicNMR mainly includes the following steps:

Generating Default Theory: Constants and variables are referred to as terms, and $P(t_1, \ldots, t_n)$ denotes an atom where P is a n-ary predicate symbol and $t_1, \ldots, t_n$ are terms. A literal is an atom or the negation of an atom. We first randomly select predicates from the predicate pool to generate prerequisite literals, justification literals, and conclusion literals. It is required in generating the default rule that the α is the conjunction of at most two literals, the justifications is a set containing at most two literals β_i, and γ is a literal. Consistent with the predicate list in LogicNLI (Tian et al., 2021), the predicate pool contains unary and binary predicates. To ensure the default theory has multiple extensions, the literals in the prerequisite and conclusion of different default rules may overlap, and the literals in the justifications and conclusions of different default rules may negate each other. After generating all default rules, we randomly select entities from the entity pool to instantiate the prerequisite literals that are not included in the conclusion, thereby generating facts W.

Converting DL to ASP: Baral and Gelfond (1994) pointed out that when α is a conjunction of literals, $\beta_1, \beta_2, \ldots, \beta_m$ and γ are all literals, such a default rule can be translated into an answer set rule γ :- $\alpha, not \neg \beta_1, \ldots, not \neg \beta_m$. To ensure the

correctness of the answers, we convert the default rule into an ASP formulation and then call the ASP solver for reasoning.

Solving: To ensure the correctness of the answer of the questions, we use the symbolic solver (clingo[2]) to solve. First, all extensions of the default theory are generated using the symbolic solver and generating the correct answer to questions based on the extensions in the skeptical and the credulous reasoning, respectively. Given a default theory Γ and questions Q. Δ are all extensions. $E \in \Delta$ represents an extension. Equation 1 shows the answer to the question $A(\Gamma, Q)$ in the skeptical reasoning, and equation 2 shows the answer to the question $A(\Gamma, Q)$ in the credulous reasoning. The answer to the question may be True (T), False (F), and Unknown (M).

$$A(\Gamma, Q) = \begin{cases} T, \text{ if } \forall_E\, E \vdash Q \\ F, \text{ if } \forall_E\, E \vdash \neg Q \\ M, \text{ if } \exists_E\, E \nvdash Q, \exists_E\, E \nvdash \neg Q \end{cases} \quad (1)$$

$$A(\Gamma, Q) = \begin{cases} T, \text{ if } \exists_E\, E \vdash Q \\ F, \text{ if } \exists_E\, E \vdash \neg Q \\ M, \text{ if } \forall_E\, E \nvdash Q, E \nvdash \neg Q \end{cases} \quad (2)$$

Translate into Natural Language: Translate the default theory into natural language according to the template. For example, the rule "$\neg\, gorgeous(x)$:- $grieving(x), not\, huge(x)$." is translated into "If someoneA is grieving, then he is not gorgeous, unless he is huge.".

We first generated **MultiLogicNMR** datasets in skeptical and credulous reasoning modes, respectively. Each sample contains three questions with different answer labels. The number of samples on the number of extensions $E = \{1, 2, 3, 4, 5\}$ is equal. Figure 1 shows a simplified example from the MultiLogicNMR in skeptical reasoning. To measure the generalization of LLMs on the number of extensions, we also generated the dataset **MultiLogicNMR_OOD** with more default rules and extensions. The number of extensions in the MultiLogicNMR_OOD is $E = \{6, 8, 10, 12, 16\}$. In addition, to further evaluate the robustness of models across diverse natural languages, we use few-shot prompt-based GPT-4o to rewrite the test set of MultiLogicNMR, and the rewritten dataset is called **MultiLogicNMR_NL**. The statistical information of all datasets is shown in the Table 5.

[2]https://pypi.org/project/clyngor/

4 MultiLogicNMRer Framework

Inspired by the ASP symbolic solver (Gebser et al., 2012) in Appendix A.2, we propose a neural-symbolic framework for multiple extension non-monotonic reasoning, MultiLogicNMRer. Figure 2 shows the framework of MultiLogicNMRer, which includes six modules: Grounding, Upper and Lower Bound Initialization, Reduction, Reasoning, Selection, and Extraction Answer Module.

Grounding Module: The MultiLogicNMRer framework implements the grounding module with prompt-based LLMs. For example, the rule "If someoneA is handsome and not intelligent then he is delicious, unless he is not drab." is instantiated as "If Toby is handsome and not intelligent then Toby is delicious, unless Toby is not drab.".

Upper and Lower Bound Initialization Module: This module extracts facts from the rules through prompt-based LLMs to initialize upper and lower bound fact sets. Since an extension must contain original facts in the default theory, the lower bound fact set is initialized with the original facts. In addition, the facts extracted from the default theory are initialized to the upper bound fact set.

Reduction Module: The reduction module obtains the reduction rules generated by the default rules under the upper and lower bound fact sets, respectively. For a default rule $\alpha : \beta_1, \beta_2, \ldots, \beta_m / \gamma$, if none of the justification β_i appears in the lower bound set, the default rule can be reduced to α/γ. Otherwise, the rule cannot be reduced.

Reasoning Module: The reasoning module is to reason with the reduced rules under the lower and upper bound fact sets, respectively, thereby generating conclusions. Then, the upper and lower-bound fact sets are updated with the conclusions generated by reduced rules. Specifically, the conclusions generated by the reduced rules under the upper bound fact set should be included in the lower bound fact set. The upper bound fact set should only contain the conclusions generated by the reduced rules in the lower bound fact set.

Selection Module: When the updated lower bound fact set is still a subset of the upper bound fact set, select and remove a fact from the upper bound set, and then add it to the lower bound fact set. Then, the updated lower and upper bound facts are used to reason for the reduction rules. In the selection module, we use cosine similarity[3] to se-

[3]https://huggingface.co/sentence-transformers/distilbert-base-nli-stsb-mean-tokens

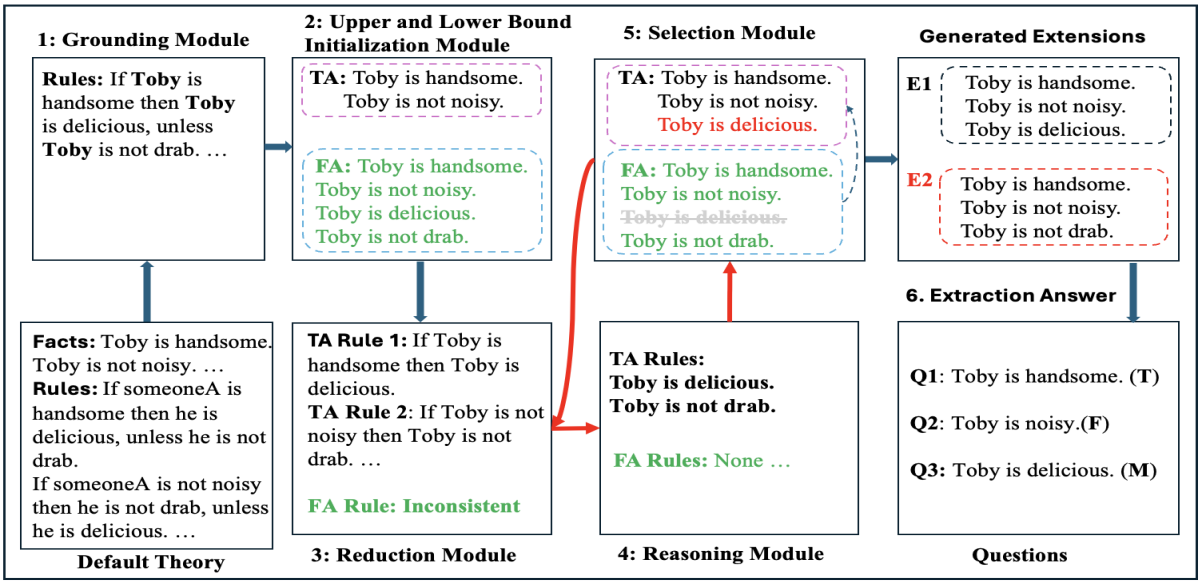

Figure 2: The framework of the proposed neural-symbolic method MultiLogicNMRer.

lect facts from the upper bound fact set. To reduce the search space, we prefer to choose facts that are relevant to the question.

Extraction Answer Module: To generate labels for the questions, we require the model to reason in skeptical and credulous reasoning modes, respectively, based on all generated extensions. The skeptical and credulous reasoning modes are shown in formula 1 and formula 2, respectively.

It is worth noting that all modules are not fine-tuned. Appendix A.4 shows all the prompts used in different modules of the MultiLogicNMRer.

5 Experiments

In the experimental part, we first use MultiLogicNMR to evaluate the model's non-monotonic reasoning capabilities under skeptical and credulous reasoning modes. Then, MultiLogicNMR_OOD and MultiLogicNMR_NL are utilized to evaluate the model's generalization and robustness.

5.1 Experimental Settings

We systematically evaluate the non-monotonic reasoning capabilities of closed-source LLMs (GPT-3.5-turbo (GPT3.5) (Brown et al., 2020), GPT-4o (OpenAI, 2023), DeepSeek-chat (Bi et al., 2024)) (DeepSeek) and open-source LLMs (Meta-LLAMA3-8B-Instruct (LLAMA3) (Touvron et al., 2023), gemma-7b-it[4] (Gemma), Mistral-7B-Instruct_v0.2 (Mistral) (Jiang et al., 2023)) in skeptical and credulous reasoning, respectively. We use prompt-based models and fine-tuned models as comparison methods. The prompt-based methods include standard zero-shot prompting, standard few-shot prompting, zero-shot CoT prompting, and symbolic solution strategy-based CoT prompting (Symbolic CoT) method. To ensure the reproducibility of experimental results, the temperature parameters of all models are set to 0. Appendix A.3 shows all prompt content of prompt-based methods. In addition, we use the LowRank Adaptation (LoRA) to fine-tune the open-source LLMs on the training set. All details of the fine-tuning experiments are described in Appendix A.5. Finally, we use the accuracy as the evaluation metric[5].

5.1.1 Results on MultiLogicNMR

Table 1 and Table 7 show the results of models on MultiLogicNMR in skeptical and credulous reasoning, respectively. First, the non-monotonic reasoning capabilities of standard zero-shot prompt-based LLMs are relatively limited. GPT-4o performs best, with the average accuracy reaching 56.7% and 59.8% in skeptical and credulous reasoning modes, respectively. However, the accuracy of LLAMA3 is only 39.7% and 41.2%. The results also show that closed-source LLMs still have more advantages than open-source LLMs. Second, compared to zero-shot prompting, the results of GPT-4o and DeepSeek are significantly improved under few-shot prompting. Next, The CoT prompts can effectively improve the non-monotonic reason-

[4]https://ai.google.dev/gemma?hl=zh-cn

[5]https://pypi.org/project/scikit-learn/

ing abilities of the models, especially the average accuracy of DeepSeek based on symbolic CoT in skeptical and credulous reasoning modes are 61.4% and 64.3%, respectively, which are much higher than the 40.2% and 39.5% accuracy under the standard zero-shot prompting. In addition, fine-tuning greatly improves the models' non-monotonic reasoning abilities. Specifically, the average accuracy of fine-tuned LLAMA3, Mistral, and Gemma models on all numbers of extensions in skeptical reasoning is 66.8%, 65.9%, and 68.3%, which are all higher than the prompt-based LLMs.

Table 1: Results of models on MultiLogicNMR in **skeptical** reasoning.

Model	Met	Test (Accuracy %)					
Extension	-hod	1	2	3	4	5	Avg
LLAMA3	Zero Shot	40.3	39.0	41.7	35.0	42.3	39.7
GPT3.5		41.3	46.7	44.6	45.6	49.6	45.6
DeepSeek		38.0	37.0	40.0	41.0	45.3	40.2
GPT-4o		57.0	58.3	56.3	53.6	58.3	56.7
LLAMA3	Few Shot	36.0	34.6	33.3	35.0	39.3	35.7
GPT3.5		39.3	36.7	33.3	30.3	40.6	36.1
DeepSeek		51.7	48.0	49.3	45.3	50.3	48.9
GPT-4o		62.0	60.0	58.7	58.3	57.3	59.3
GPT3.5	Zero Shot CoT	42.3	43.3	38.0	36.7	39.3	39.9
DeepSeek		55.0	55.6	59.3	59.7	62.0	58.3
GPT-4o		54.7	51.0	56.7	57.0	61.3	56.1
GPT3.5	Sym-bolic CoT	56.0	54.3	49.3	52.0	48.0	51.9
DeepSeek		63.3	59.3	62.0	60.3	62.0	61.4
GPT-4o		53.3	56.9	58.3	55.0	55.8	55.8
LLAMA3	Fine Tune	54.0	63.6	72.3	69.3	74.7	66.8
Mistral		62.6	62.3	68.6	67.0	69.0	65.9
Gemma		57.3	65.7	72.0	69.0	**77.3**	68.3
MLNMRer (LLAMA3)	Ours	71.0	**71.3**	72.3	70.3	76.3	72.3
MLNMRer (GPT3.5)		75.0	70.3	73.3	**73.7**	75.3	**73.5**
MLNMRer (DeepSeek)		**80.7**	71.2	**74.0**	66.7	72.3	73.0

5.2 Main Results

The proposed MultiLogicNMRer has a more significant improvement than the corresponding prompt-based basic models and is close to or even exceeds the results of the fine-tuned models. For example, when using DeepSeek as the base model, the average accuracy increases from 40.2% to 73.0% and from 39.5% to 74.5% in the skeptical and the credulous reasoning modes, respectively, exceeding the results of all prompt-based and fine-tuned methods. When LLAMA3 is used as the base model, the average accuracy in the skeptical reasoning mode is 72.3%, surpassing the average accuracy of the fine-tuned LLAMA3. These verify the effectiveness of MultiLogicNMRer. In addition, the results of MultiLogicNMRer on different numbers of extensions are consistent, which illustrates the stability and reliability of MultiLogicNMRer.

Table 2: Results of models on the out-of-domain dataset MultiLogicNMR_OOD in skeptical reasoning.

Model	Met	Test (Accuracy %)					
Extension	-hod	1	2	3	4	5	Avg
LLAMA3	Zero Shot	37.0	35.0	34.7	40.7	32.7	36.0
GPT3.5		43.0	40.3	39.0	41.0	43.3	41.3
DeepSeek		52.3	54.3	56.0	55.0	56.0	54.7
GPT-4o		48.7	52.0	43.3	49.7	41.0	46.9
GPT3.5	Few Shot	36.0	35.7	36.7	38.0	36.0	36.5
DeepSeek		60.7	63.1	59.3	59.7	60.9	60.8
GTP-4o		48.5	48.9	45.0	47.2	46.5	47.3
GPT3.5	Zero Shot CoT	39.0	34.3	33.3	34.7	33.0	34.8
DeepSeek		60.3	60.0	58.6	63.0	59.3	60.4
GPT-4o		57.0	54.0	55.7	56.3	59.0	56.4
GPT3.5	Sym-bolic CoT	36.3	44.7	43.7	41.3	44.3	42.1
DeepSeek		61.0	65.3	62.6	60.0	61.0	62.0
GPT-4o		55.6	52.3	51.0	52.3	53.3	52.9
LLAMA3	Fine Tune	68.3	**66.3**	68.0	**71.6**	**66.3**	**68.0**
Mistral		58.0	51.7	54.0	51.6	55.0	54.0
Gemma		53.3	56.0	59.0	56.7	59.0	57.2
MLNMRer (LLAMA3)	Ours	59.7	65.0	**72.3**	69.3	65.3	66.3
MLNMRer (GPT3.5)		65.3	54.3	56.0	60.0	53.3	57.8
MLNMRer (DeepSeek)		**67.5**	61.7	61.7	61.7	65.0	63.5

5.2.1 Generalization on Extensions Numbers

To explore the LLMs' generalization regarding the number of extensions, we evaluate the model's performance on the MultiLogicNMR_OOD.

Tables 2 and Table 8 show the results of the models based on the skeptical and credulous reasoning modes on the MultiLogicNMR_OOD, respectively. First, most of the prompt-based models produce slightly lower results than the models on MultiLogicNMR, indicating that the increase in the number of extensions raises the difficulty of non-monotonic reasoning to a certain extent. Second, we can similarly observe that the accuracy of the symbolic CoT prompt-based models is bet-

ter than other prompt-based models. In addition, compared with the results of the models on MultiLogicNMR, the fine-tuned model and the proposed MultiLogicNMRer have a certain degree of decline on the MultiLogicNMR_OOD, which further illustrates that the increase in the number of extensions can bring difficulty. However, the proposed MultiLogicNMRer still outperforms the prompt-based models, and its results even exceed the accuracy of the fine-tuned models. Specifically, in the skeptical reasoning mode, when GPT3.5 is used as the base model, the average accuracy of MultiLogicNMRer is 57.8%, while the average accuracy of the best symbolic CoT prompts-based GPT3.5 is only 42.1%. Similarly, in the credulous reasoning mode, the average accuracy of MultiLogicNMRer with LLAMA3 as the base model is 53.4%, while the accuracy of the fine-tuned LLAMA3 on MultiLogicNMR_OOD is 50.3%. These results further verify that the proposed neural-symbolic framework MultiLogicNMRer has a certain generalization ability on the number of extensions.

5.2.2 Robustness on Text Diversity

The diversity and complexity of the datasets automatically synthesized based on templates are relatively limited. To solve this problem, we generated the MultiLogicNMR_NL by rewriting the MultiLogicNMR. To evaluate the diversity of the dataset, we use two evaluation indicators, Remote Clique Score and Chamfer Distance Score (Li et al., 2023; Cox et al., 2021). The Remote Clique score is the average mean distance of an instance from other instances, while the Chamfer Distance Score is the average minimum distance of an instance from other instances. The higher scores mean higher diversity in the dataset. As shown in Table 4, the result of MultiLogicNMR_NL on the Remote Clique Score indicator is 0.412, while the Remote Clique Score of the original MultiLogicNMR is 0.341. The detailed process of generating the MultiLogicNMR_NL dataset is in Appendix A.1.

Table 3 and 9 give results of the models using the credulous reasoning and the skeptical reasoning modes on MultiLogicNMR_NL, respectively. Compared with the results of the models on MultiLogicNMR, the results of the models on MultiLogicNMR_NL have both declined to a large extent. For example, in credulous reasoning mode, the average accuracy of the symbolic CoT prompt-based GPT3.5 on MultiLogicNMR and MultiLogicNMR_NL is 56.1% and 44.1%, respectively. In addition, the average accuracy of the fine-tuned Mistral and Gemma models on the MultiLogicNMR_NL dataset dropped to 43.4% and 51.3%, respectively. These all show that the diversity of context will further increase the difficulty of non-monotonic reasoning. In addition, although the average accuracy of the proposed neural-symbolic method MultiLogicNMRer produced by the GPT3.5-based model has dropped to 52.1%, it is still higher than the 44.1% of the symbolic CoT prompt-based models. The experimental results also show that MultiLogicNMRer has certain advantages on text diversity datasets.

Table 3: Results of models on MultiLogicNMR_NL in **credulous** reasoning.

Model Extension	Method	Test (Accuracy %) 1	2	3	4	5	Avg
LLAMA3	Zero Shot	41.8	38.0	43.0	37.7	37.9	39.7
GPT3.5		39.1	35.3	34.0	39.1	38.6	37.2
DeepSeek		44.5	42.3	41.3	46.4	47.3	44.4
GPT-4o		43.5	37.0	48.3	46.3	42.7	43.5
GPT3.5	Few Shot	43.1	38.6	40.0	42.7	41.7	41.2
DeepSeek		48.8	51.0	50.0	51.2	52.0	50.6
GPT-4o		48.8	43.6	49.3	53.5	50.6	48.4
GPT3.5	Zero Shot CoT	41.5	38.7	42.7	40.4	46.6	41.9
DeepSeek		49.2	48.3	44.3	46.8	45.9	46.9
GPT-4o		46.8	49.0	45.7	44.8	47.7	46.8
GPT3.5	Symbolic CoT	40.4	45.3	43.3	47.0	44.3	44.1
DeepSeek		48.3	50.0	43.6	46.0	53.0	47.5
GPT-4o		49.8	46.0	46.3	51.8	48.9	48.6
LLAMA3	Fine Tune	54.2	59.6	62.7	59.3	63.1	59.7
Mistral		43.8	43.0	43.0	43.4	43.4	43.4
Gemma		41.8	50.0	51.7	53.2	59.7	51.3
MLNMRer (GPT3.5)	Ours	53.2	49.3	54.7	51.2	52.9	52.1
MLNMRer (DeepSeek)		51.5	45.3	44.3	51.5	50.8	48.6

5.3 Analysis

5.3.1 Ablation Study

We conducted an ablation study to verify the importance of the designed modules in the MultiLogicNMRer framework. In more detail, we created three variant models. The first model, denoted by MultiLogicNMRer (w/o grounding), is created by excluding the grounding module. The second model, denoted by MultiLogicNMRer (w/o reduction), is created by omitting the reduction module. The third model, denoted by MultiLogicNMRer

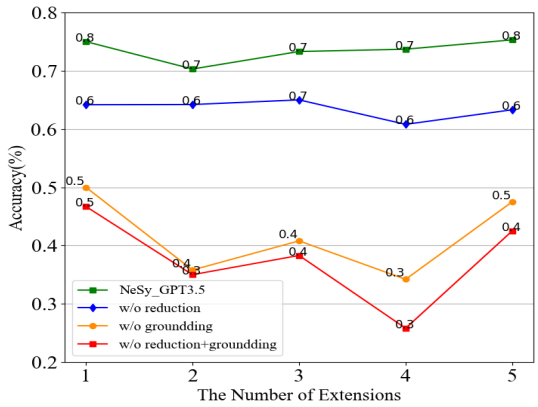

Figure 3: Results of Ablation on MultiLogicNMR dataset in skeptical reasoning.

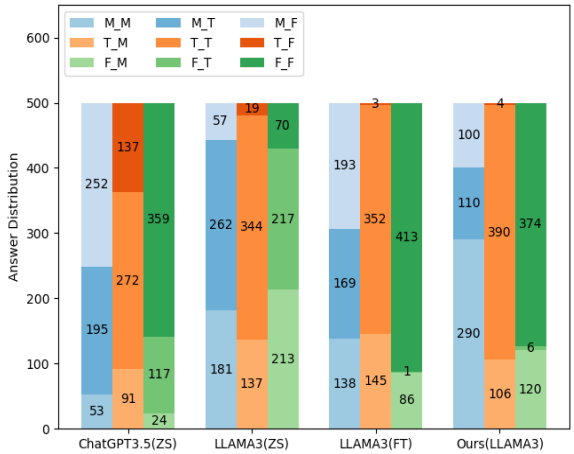

Figure 4: The distribution of answers generated by models in skeptical reasoning. The ZS and FT represent zero-shot prompting and fine-tuning, respectively.

(w/o reduction + grounding), is created by omitting both the reduction and grounding modules. Figure 3 illustrates the result of the ablation study. It can be observed that the performance of MultiLogicNMRer sharply drops when the grounding module is omitted, indicating that the grounding module is crucial for MultiLogicNMRer. The reason may be that the LLMs can't effectively find the correct upper and lower bound facts set of extension and perform reasoning based on ungrounded contexts. Furthermore, we can observe that the performance of MultiLogicNMRer decreases to a certain extent when the reduction module is omitted. We argue that this can be attributed to LLMs being able to directly reason on unreduced default rules.

5.3.2 Label Analysis

To analyze the challenges of LLMs on the MultiLogicNMR, Figure 4 shows the distribution of answers label generated by the models in skeptical reasoning. First, the results show that the zero-shot prompt-based GPT3.5 and LLAMA3 have the lowest accuracy for questions with "Unknown", especially GPT3.5, which only answers correctly 35/500. In addition, although the fine-tuning methods can further improve the model's accuracy on questions with "True" and "False", it still performs poorly on questions with "Unknown". The possible reason is that the models need to find all the extensions to answer questions with "Unknown" correctly in skeptical reasoning, and it is still challenging for LLMs to find all the extensions. Moreover, Figure 4 shows that MultiLogicNMRer based on LLAMA3 can correctly answer 290/500 questions with "Unknown" while maintaining high accuracy for questions with other labels. These results further illustrate that the MultiLogicNMRer has more advantages and effectiveness than the prompting and fine-tuning methods.

6 Conclusions

In this paper, we not only generate three datasets (MultiLogicNMR, MultiLogicNMR_OOD, and MultiLogicNMR_NL) but also propose a neural-symbolic framework, MultiLogicNMRer, for multiple-extension non-monotonic reasoning. Using the datasets, we systematically evaluate the non-monotonic reasoning abilities of prompt-based and fine-tuned LLMs in both skeptical and credulous reasoning modes. In particular, MultiLogicNMR_OOD and MultiLogicNMR_NL assess the models' generalization and robustness across text diversity, respectively. Our work reveals that the prompt-based LLMs still face significant challenges in non-monotonic reasoning, and even the fine-tuned LLMs do not understand non-monotonic reasoning. Nonetheless, not only does the performance of MultiLogicNMRer matches that of the fine-tuned models and even surpass some, but the reasoning process is also more reliable. The proposed neural-symbolic approach takes a new step towards achieving reliable non-monotonic reasoning over natural language. In the future, we are interested in exploring multiple-extension non-monotonic reasoning with more language diversity and complexity, and handling irrelevant information. We will also explore the application of non-monotonic reasoning to different real-world tasks.

7 Limitations

Although the proposed MultiLogicNMR and neural-symbolic framework MultiLogicNMRer can effectively explore and improve the non-monotonic reasoning abilities in LLMs, there are some limitations. First, we have only taken a small step towards building realistic non-monotonic reasoning with multiple extensions benchmark. Although we rewrote the MultiLogicNMR generated based on template translation using LLMs, which increased the diversity of the dataset to a certain extent, its diversity and complexity are limited compared to the non-monotonic reasoning benchmark in real scenarios. Second, humans can usually perform non-monotonic reasoning involving massive irrelevant information, while the generated dataset samples usually contain less irrelevant information. On one hand, this reduces the difficulty of model reasoning. On the other hand, it could reduce the impact of irrelevant information when evaluating a model. Moreover, although the basic model in the proposed neural-symbolic framework MultiLogicNMRer does not need to be fine-tuned, calling LLMs iteratively in multiple modules is necessary, which increases the cost of the MultiLogicNMRer framework in practical applications. However, the inference calculations increased by the solution strategy can effectively improve the model's performance, it is also a challenge to trade the performance and cost in models.

References

Grigoris Antoniou and Sotiris Batsakis. 2023. Defeasible reasoning with large language models - initial experiments and future directions. In *Proceedings of the 17th International Rule Challenge and 7th Doctoral Consortium @ RuleML+RR 2023 co-located with 19th Reasoning Web Summer School (RW 2023) and 15th DecisionCAMP 2023 as part of Declarative AI 2023, Oslo, Norway, 18 - 20 September, 2023*, volume 3485 of *CEUR Workshop Proceedings*.

Chitta Baral and Michael Gelfond. 1994. Logic programming and knowledge representation. *J. Log. Program.*, 19/20:73–148.

Chandra Bhagavatula, Ronan Le Bras, and et. al. 2020. Abductive commonsense reasoning. In *ICLR 2020*.

Xiao Bi, Deli Chen, and Guanting Chen et al. 2024. Deepseek LLM: scaling open-source language models with longtermism. *CoRR*, abs/2401.02954.

Ronald J. Brachman and Hector J. Levesque. 2004. *Knowledge Representation and Reasoning*.

Tom B. Brown, Benjamin Mann, and Nick Ryder et al. 2020. Language models are few-shot learners. In *NeurIPS 2020, December 6-12, 2020, virtual*.

Cameron B Browne and et. al. 2012. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43.

Roberta Calegari, Giuseppe Contissa, Francesca Lagioia, Andrea Omicini, and Giovanni Sartor. 2019. Defeasible systems in legal reasoning: A comparative assessment. In *Legal Knowledge and Information Systems - JURIX 2019: The Thirty-second Annual Conference, Madrid, Spain, December 11-13, 2019*, volume 322 of *Frontiers in Artificial Intelligence and Applications*, pages 169–174. IOS Press.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3882–3890.

Erica Coppolillo, Francesco Calimeri, Giuseppe Manco, Simona Perri, and Francesco Ricca. 2024. LLASP: fine-tuning large language models for answer set programming. *CoRR*, abs/2407.18723.

Samuel Rhys Cox, Yunlong Wang, Ashraf M. Abdul, Christian von der Weth, and Brian Y. Lim. 2021. Directed diversity: Leveraging language embedding distances for collective creativity in crowd ideation. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*, pages 393:1–393:35.

Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. Explaining answers with entailment trees. In *EMNLP 2021*, pages 7358–7370.

Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. 2012. *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Simeng Han, Hailey Schoelkopf, and et. al. 2022. FOLIO: natural language reasoning with first-order logic. *CoRR*, abs/2209.00840.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 8154–8173.

Ruixin Hong, Hongming Zhang, Xintong Yu, and Changshui Zhang. 2022. METGEN: A module-based entailment tree generation framework for answer explanation. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1887–1905.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

Albert Q. Jiang, Alexandre Sablayrolles, and Arthur Mensch et al. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Henry A. Kautz. 2022. The third AI summer: AAAI robert s. engelmore memorial lecture. *AI Mag.*, 43(1):93–104.

Mehran Kazemi, Quan Yuan, Deepti Bhatia, Najoung Kim, Xin Xu, Vaiva Imbrasaite, and Deepak Ramachandran. 2023. Boardgameqa: A dataset for natural language reasoning with contradictory information. In *NeurIPS 2023*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *NeurIPS 2022*.

Alina Leidinger, Robert van Rooij, and Ekaterina Shutova. 2024. Are llms classical or nonmonotonic reasoners? lessons from generics. In *ACL 2024 - Short Papers, Bangkok, Thailand, August 11-16, 2024*, pages 558–573.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023. Synthetic data generation with large language models for text classification: Potential and limitations. In *EMNLP 2023, Singapore, December 6-10, 2023*, pages 10443–10461.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In *IJCAI 2020*, pages 3622–3628.

Yantao Liu, Zijun Yao, Xin Lv, Yuchen Fan, Shulin Cao, Jifan Yu, Lei Hou, and Juanzi Li. 2024. Untangle the KNOT: interweaving conflicting knowledge and reasoning skills in large language models. In *LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 17186–17204.

Witold Lukaszewicz. 1990. *Non-monotonic reasoning - formalization of commonsense reasoning*. Ellis Horwood.

Theo Olausson, Alex Gu, Benjamin Lipkin, Cedegao E. Zhang, Armando Solar-Lezama, Joshua B. Tenenbaum, and Roger Levy. 2023. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *EMNLP 2023, Singapore, December 6-10, 2023*, pages 5153–5176.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Wang. 2023. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 3806–3824.

Mihir Parmar, Nisarg Patel, Neeraj Varshney, Mutsumi Nakamura, Man Luo, Santosh Mashetty, Arindam Mitra, and Chitta Baral. 2024. Towards systematic evaluation of logical reasoning ability of large language models. *arXiv preprint arXiv:2404.15522*.

Raymond Reiter. 1980. A logic for default reasoning. *Artif. Intell.*, 13(1-2):81–132.

Rachel Rudinger, Vered Shwartz, Jena D. Hwang, Chandra Bhagavatula, Maxwell Forbes, Ronan Le Bras, Noah A. Smith, and Yejin Choi. 2020. Thinking like a skeptic: Defeasible inference in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4661–4675.

Abulhair Saparov and He He. 2023. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *ICLR 2023*.

Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L. Hamilton. 2019. CLUTRR: A diagnostic benchmark for inductive reasoning from text. In *EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4505–4514.

Andrzej Szalas. 2019. Decision-making support using nonmonotonic probabilistic reasoning. In *Intelligent Decision Technologies 2019 - Proceedings of the 11th KES International Conference on Intelligent Decision Technologies (KES-IDT 2019), Volume 1, Malta, June 17-19, 2019*, volume 142 of *Smart Innovation, Systems and Technologies*, pages 39–51.

Jidong Tian, Yitian Li, Wenqing Chen, Liqiang Xiao, Hao He, and Yaohui Jin. 2021. Diagnosing the first-order logical reasoning ability through logicnli. In *EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3738–3747.

Hugo Touvron, Thibaut Lavril, and Gautier Izacard et al. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Frank van Harmelen, Vladimir Lifschitz, and Bruce W. Porter, editors. 2008. *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS 2022*.

Jason Weston, Antoine Bordes, Sumit Chopra, and Tomás Mikolov. 2016. Towards ai-complete question answering: A set of prerequisite toy tasks. In *ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Yeliang Xiu, Zhanhao Xiao, and Yongmei Liu. 2022. Logicnmr: Probing the non-monotonic reasoning ability of pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 3616–3626.

Jundong Xu, Hao Fei, Liangming Pan, Qian Liu, Mong-Li Lee, and Wynne Hsu. 2024. Faithful logical reasoning via symbolic chain-of-thought. *arXiv preprint arXiv:2405.18357*.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *ICLR 2020*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. Star: Bootstrapping reasoning with reasoning. In *NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

A Appendix

A.1 Datasets Generation

Algorithm 1 describes the generation process of the MultiLogicNMR dataset. Line 2 to 9 in Algorithm 1 show how to generate the default rules in each sample iteratively. Specifically, Lines 3, 5, and 6 in Algorithm 1 randomly select the prerequisite literal r_l^{pre}, justifications literal r_l^{jus}, and conclusion literal r_l^{con} of the default rule from the predicate pool P_P. In a default rule, we require that the prerequisite α is the conjunction of at most two literals r_l^{pre}, justifications is a set containing at most two literals r_l^{jus}, and the conclusion γ is a literal r_l^{con}. In addition, to be more capable of generating default theories with multiple extensions, the literals in the prerequisite and conclusion in different default rules may be the same, and the justifications and conclusions in different default rules may negate each other. After generating all the default rules, line 10 shows that we randomly select entities from the entity pool E_P to instantiate the prerequisite literals not included in the conclusion into facts W. In addition, we generate questions Q based on the conclusion literals r_l^{con} that are not included in the prerequisites. To generate answers to questions, we first call the symbolic solver[6] to generate all extensions of the default theory (line 12), and then use skeptical and credulous reasoning to reason about the questions based on the extensions to generate answers (line 13). Finally, we translate all facts W, and default rules R, questions Q, answers A, and extensions E into natural language based on the template (line 14).

To further evaluate the model's generalization for the number of extensions, we used the same synthesis method to generate the MultiLogicNMR_OOD with more extensions. The number of extensions in the MultiLogicNMR_OOD dataset is $E = [6, 8, 10, 12, 16]$. Since the diversity and complexity of natural language generated based on templates are limited, to solve this problem and evaluate the robustness of the model on diverse natural language datasets, we rewrote the test set in MultiLogicNMR using the GPT-4o model based on few-shot prompting. The dataset generated after rewriting called MultiLogicNMR_NL. An example from the MultiLogicNMR_NL dataset is shown in Box 2. Table 4 gives the evaluation results of the diversity of the original MultiLogicNMR and the MultiLogicNMR_NL dataset. The Remote Clique score is the average mean distance of an instance from other instances, and the Chamfer Distance Score is the average minimum distance of an instance from other instances. The higher scores mean greater differences between data instances, i.e., a higher diversity in the dataset. As shown in Table 4, the diversity of the rewritten MultiLogicNMR_NL dataset is higher than that of the MultiLogicNMR dataset.

Table 5 gives the statistical information of the generated datasets. Among them, compared to the MultiLogicNMR dataset, the samples in MultiLogicNMR_OOD have more rules and involve more extensions than MultiLogicNMR. They are used to measure the generalization of LLMs on non-monotonic reasoning with more extensions tasks.

Algorithm 1: MultiLogicNMR Dataset Generation Algorithm

Input: Predicates Pool P_P, Entity Pool E_P, Iterative Number T.
Output: Facts W, Default Rules R, Questions Q, Extensions E, Answers A.

1 Initialization: $iter = 1, T = 10, W = \varnothing, R = \varnothing, Q = \varnothing$;
2 **while** $iter \leq T$ **do**
3 **Generate prerequisite literal** r_l^{pre} based on predicate r_p^{pre} from P_P or R_P^{con};
4 $R_P^{pre} = R_P^{pre} \cup r_P^{pre}$;
5 **Generate justifications literal** r_l^{jus} based on predicate r_p^{jus} from P_P or R_P^{con}. $R_P^{jus} = R_P^{jus} \cup r_P^{jus}$;
6 **Generate conclusion literal** r_l^{con}, predicate r_p^{con} from P_P or R_P^{con}. $R_P^{con} = R_P^{jus} \cup r_P^{con}$;
7 **Generate default Rule** $R_i : \frac{R_l^{pre}:R_l^{jus}}{R_l^{con}}$, R_l^{pre} is the conjunction of at most two literals r_l^{pre}, R_l^{jus} at most two literals r_l^{jus}, and $R_l^{con} = r_l^{con}$;
8 $R = R \cup R_i, iter = iter + 1$;
9 **end**
10 **Generate fact literal** W_l^{fact} based on predicate $R_P^{fact} \in R_P^{pre} \setminus R_P^{con}, W = W \cup W_l^{fact}$;
11 **Generate question literal** R_l^{ques} based on predicate $R_P^{ques} \in R_P^{pre} \cap R_P^{con}, Q = Q \cup R_l^{ques}$;
12 **Generate default theory extensions** E through symbolic solvers[7]. ;
13 **Generate answers** A to questions Q based on the expansion E. ;
14 **Convert the facts, default rules and questions** into natural language ;

[6] https://pypi.org/project/clyngor/

Box 1: The few-shot prompting for the generating MultiLogicNMR_NL dataset

Task Description: Given natural language sentences, You need to regenerate new sentences based on the given sentences, and the newly generated sentences are required not to change the semantics of the original sentences.
The input and output format are: The input facts sentences are:“”.
The input rules sentences are:“”. The input questions sentences are: “”. The output facts sentences are: “”. The output rules sentences are:“”.
The output questions sentences are:“”.
For example: The input facts sentences are: “Hanna is nervous. Hanna is not fuzzy. Hanna is not quiet. Hanna is rainy. Hanna is not inquisitive. Hanna is dizzy. Hanna is courteous.”. The input rules sentences are: “If someoneA is nervous and gifted then he is boring, unless he is important. If someoneA is courteous and boring then he is oak, unless he is not historical or he is embarrassed. If someoneA is not inquisitive then he is practical, unless he is excited or happy. If someoneA is tense and excited then he is important, unless he is boring or he is not mean.”. The input questions sentences are: “Hanna is not boring. Hanna is oak. Hanna is not magnificent.”.
The output facts sentences are: “Hanna is experiencing a sense of unease and trepidation. Hanna is not particular or finicky in her preferences. Hanna is a lively and expressive person. Hanna is experiencing a spell of inclement weather. Hanna exhibits a lack of intellectual curiosity and a dearth of inquisitiveness. Hanna is experiencing some lightheadedness and feeling a bit disoriented. Hanna displays a charming and gracious demeanor in her interactions with others.”. The output rules sentences are: “If an individual possesses a combination of nervousness and exceptional gifts, they may come across as uninteresting, unless their presence holds significant importance. If an individual exhibits both courtesy and a tendency toward dullness, they can be likened to an oak tree, unless they lack any historical significance or feel ashamed in certain situations. If an individual is not inclined towards curiosity, then they are likely to embrace practicality, except in cases where they are filled with excitement or contentment. An individual experiences a combination of tension and excitement, which denotes their perceived significance, unless their demeanor is perceived as dull or unpleasant.”. The output questions sentences are: “Hanna may across as interesting. Hanna can be likened to an oak tree. Hanna is not magnificent.”.
Note that you only need to output the modified sentence.

Table 4: Comparison results between MultiLogicNMR_NL and MuLtiLogicNMR under diversity indicators.

Dataset	Mode	Remote Clique Score	Chamfer Distance Score
MultiLogicNMR	Skeptical	0.341	0.128
	Credulous	0.334	0.118
MultiLogicNMR_NL	Skeptical	0.412	0.151
	Credulous	0.412	0.150

Table 5: Statistical information for proposed datasets.

Dataset	Mode		#Num.	#Ques.	#F.Avg	#R.Avg	#Extensions (1:1:1:1:1)	Label (T:F:M)
MultiLogicNMR	Skeptical	Train	5000	15000	12	10	[1,2,3,4,5]	1:1:1
		Dev	500	1500				
		Test	500	1500				
	Credulous	Train	5000	15000				
		Dev	500	1500				
		Test	500	1500				
MultiLogicNMR_OOD	Skeptical	Test	500	1500	22	20	[6,8,10,12,16]	
	Credulous	Test	500	1500				
MultiLogicNMR_NL	Skeptical	Test	500	1500	12	10	[1,2,3,4,5]	
	Credulous	Test	500	1500				

The #Num. represents the number of samples in the generated dataset. The #Ques. represents the number of questions in generated dataset. #F. Avg represents the average number of facts in the dataset. The #R. Avg represents the average number of rules in the dataset, and #Extension represents the number of extensions.

Box 2: An Example from MultiLogicNMR_NL Dataset

Facts: Sinclair is free from confusion. Sinclair embodies purity. Sinclair is courageous. Sinclair is full of glamour.
Sinclair shows no reverence towards Grover. Sinclair exhibits a fierce nature. Sinclair possesses a reserved demeanor.
Sinclair maintains an optimistic outlook. Sinclair is likely. Sinclair is ordinary. Sinclair remains composed.
Sinclair is characterized by theatrical behavior. Sinclair does not mock Grover. Sinclair holds a fervent passion.
Default Rules: If an individual neither exhibits confusion nor holds a pessimistic view, they are free from regret, unless they are considered costly or not comforted.
An individual who is pure is financially oriented, unless they are tangible.
If an individual is both likely and glamorous, they do not grow tired, unless they lack enthusiasm.
Without regrets, an individual is cost effective, unless absent of regret or plentiful.
A passionate and financially oriented individual is plentiful, unless they are costly.
One who does not mock another and remains calm is free from anxiety, unless they are envious or abundant.
Lacking both super qualities and pessimism means one is comforted, unless they feel isolated or regretful.
Being courageous ensures one is not isolated, unless anonymity or lack of comfort prevails.
If an individual lacks respect for another yet possesses theatrical traits, they appear harsh, unless they are costly.
Combining fierceness with shyness results in a bad-tempered nature, unless the individual is passionate or cost-concerned.

The question is: Sinclair is exempt from regret. **The answer is:** Unknown.

The question is: Sinclair is not aligned with financial matters. **The answer is:** False.

The question is: Sinclair is not fatigued. **The answer is:** True.

A.2 Answer Set Solver

Algorithm 2 gives the symbolic solver for answer set programming. The idea of the proposed neural-symbolic framework MultiLogicNMRer is consistent with algorithm 2. In Algorithm 2, first, the input default theory is instantiated (line 16), and then the upper bound set U and the lower bound set L of the default theory are respectively calculated (line 17). Finally, the function $expand_p$ is called to update the upper and lower bound fact sets to generate the expansion (lines 1-7). Specifically, the lower bound set should contain the conclusions of the reduction rules under the upper bound set (line 5), while the upper bound set should only contain the conclusions of the reduction rules under the upper bound set (line 6). The lower bound is returned as an extension when the upper and lower bounds are consistent (line 11). If the updated lower bound set is included in the upper bound set, the extension search fails, and failure is returned (line 10); if the updated upper bound is still a superset of the lower bound, a random literal is selected from the upper bound set to be added to the lower bound set. The literal is deleted from the upper bound set. The upper and lower bounds set will be updated and searched again. All the upper and lower bound set pairs are searched, and all the extensions in the default rules are found.

It is worth noting that this symbolic solver only applies to normal logical program rules. On one hand, the default rules in MultiLogicNMR generated under specific constraints can be converted into equivalent logic programs; on the other hand, although the proposed MultiLogicNMR dataset involves classical negation $\neg$, it is impossible to include atoms and the negation of atomic in the same extension at the same time. Hence, the symbolic solver's solution idea still applies to the proposed non-monotonic reasoning benchmark MultiLogicNMR.

Algorithm 2: Classic ASP solving algorithm

1 $expand_p$(L,U) ;
2 repeat ;
3 $L' \leftarrow L$;
4 $U' \leftarrow U$;
5 $L \leftarrow L' \cup Cn(P^{U'})$;
6 $U \leftarrow U' \cap Cn(P^{L'})$;
7 Until ($L = L'$) or $L \nsubseteq U$;
8 $solver_p$(L, U) ;
9 $(L, U) \leftarrow expand_p(L, U)$;
10 if $L \nsubseteq U$ then failure ;
11 if $L = U$ then output L ;
12 else $a \leftarrow choose(U \setminus L)$;
13 $solve_p(L \cup \{a\}$, U);
14 $solve_p(L, U \setminus \{a\}$);
15 main() ;
16 $P \leftarrow ground(input)$;
17 init(L, U) ;
18 $solve_p$(L, U) ;

A.3 Standard Zero/Few Shot and CoT Prompts

Box 3: Zero-Shot Prompting in Skeptical Reasoning

Task Description: Given contexts and question, You need to generate answer labels for questions in a given context. The answers to the questions are labeled "True", "False" and "Unknown".
If the question can be inferred under all reasoning paths based on the context, and the negation of the question cannot be inferred under all reasoning paths based on the context, the answer label of the question is: "True";
If the negation of the question can be inferred under all reasoning path based on the context, and the question cannot be inferred under all reasoning path based on the context, the answer label of the question is: "False";
If the question and the negation of the question cannot be deduced under a certain reasoning path based on the context, the answer label of the question is: "Unknown". You must generate answer labels for the question.
The input format is: Context: " ". Question:" ".
The output format is: The answer label of the question is:" ".
Note that you only need to generate the answer label for the question without giving an explanation or justification. Please read the context carefully and answer the questions.

Box 4: Zero-Shot Prompting in Credulous Reasoning

Task Description: Given contexts and question, You need to generate answer labels for questions in a given context. The answers to the questions are labeled "True", "False" and "Unknown".
If the question can be inferred under a certain reasoning path based on the context, the answer label of the question is: "True";
If the negation of the question can be inferred under a certain reasoning path based on the context, the answer label of the question is: "False";
If the question and the negation of the question both cannot be deduced under all reasoning path based on the context, the answer label of the question is: "Unknown". You must generate answer labels for the question.
The input format is: Context: " ". Question: " ".
The output format is: The answer label of the question is: " ".
Note that you only need to generate the answer label for the question, without giving an explanation or justification. Please read the context carefully and answer the questions.

Box 5: Few-Shot Prompting in Skeptical Reasoning

Task Description: Given contexts and question, You need to generate answer labels for questions in a given context. The answers to the questions are labeled "True", "False" and "Unknown".
If the question can be inferred under all reasoning paths based on the context, and the negation of the question cannot be inferred under all reasoning paths based on the context, the answer label of the question is: "True";
If the negation of the question can be inferred under all reasoning path based on the context, and the question cannot be inferred under all reasoning path based on the context, the answer label of the question is: "False";
If the question and the negation of the question cannot be deduced under a certain reasoning path based on the context, the answer label of the question is: "Unknown". Each context has a question, you must generate answer labels for each question.
The input format is: Context: " ". Question:" ".
The output format is: The answer label of the question is:" ".
Example 1: Context: Basil is not innocent. Basil is not wooden. Basil is discreet. Basil is not petite. Basil is comprehensive. Basil is nutty. Basil is historical. ... If someoneA is historical then he is red, unless he is not lively or he is not big. If someoneA is nutty and steep then he is miniscule, unless he is not weary or he is outstanding. If someoneA is not petite then he is brave, unless he is sticky or he is psychological. If someoneA is not wooden and miniscule then he is psychological, unless he is brave. ...
If the question is: Basil is red. **Then the answer label for the question is:** "True";
If the question is: Basil is miniscule. **Then the answer label for the question is:** "Unknown";
If the question is: Basil is not ashamed. **Then the answer label for the question is:** "False".
Note that you only need to generate the answer label for the question, without giving an explanation or justification. Please read the context carefully and answer the questions.

Box 6: Few-Shot Prompting in Credulous Reasoning

Task Description: Given contexts and question, You need to generate answer labels for questions in a given context. The answers to the questions are labeled "True", "False" and "Unknown".
If the question can be inferred under a certain reasoning path based on the context, the answer label of the question is: "True"; If the negation of the question can be inferred under a certain reasoning path based on the context, the answer label of the question is: "False"; If the question and the negation of the question both cannot be deduced under all reasoning path based on the context, the answer label of the question is: "Unknown". Each context has three questions, You must generate answer labels for each question.
The input format is: Context: " ". Question:" ".
The output format is: The answer label of question is: " ".
Example 1: Context: Cecil is acceptable. Cecil is uptight. Cecil is not good tempered. Cecil is not severe. Cecil is not messy. Cecil is not self disciplined. Cecil is not logical. Cecil is not right. Cecil is careful.... If someoneA is not logical then he is not visible, unless he is not harsh. If someoneA is not messy and careful then he is not outstanding, unless he is not uptight. If someoneA is uptight and not severe then he is not successful, unless he is similar or he is not good. If someoneA is not visible then he is serious, unless he is not outstanding. If someoneA is not self disciplined then he is not fantastic, unless he is emotional or he is serious. ...
If the question is: Cecil is good. **Then the answer label for the question is:** "False"; **If the question is:** Cecil is not visible. **Then the answer label for the question is:** "Unknown"; **If the question is:** Cecil is similar. **Then the answer label for the question is:** "True".
Note that you only need to generate the answer label for the question, without giving an explanation or justification. Please read the context carefully and answer the questions.

Box 7: Zero-Shot COT Prompting in Skeptical Reasoning

Task Description: Given contexts and question, and the context consists of facts and default rules. You need first to generate all extensions based on the given context and then to answer the question according to the extensions. An extension is a set of non-contradictory facts and conclusions generated by rule-based reasoning. There may be multiple extensions in the context, and you need to find all the extensions. Next, the answer to the question is generated based on the generated extension.
If the question can be inferred under all extensions, and the negation of the question cannot be inferred under all extensions, the answer label of the question is: "True";
If the negation of the question can be inferred under all extensions, and the question cannot be inferred under all extensions, the answer label of the question is: "False";
If the question and the negation of the question cannot be deduced under a certain extension, the answer label of the question is: "Unknown". Each context has a question.
The input format is: Facts:" ", Default Rules:" ". Question:" ".
The output format is: The extensions are:" ". The answer label of the question is:" ".
You must generate all extensions and answer the question according to the extensions. Please read the context carefully. **Letś think step by step.**

Box 8: Zero-Shot COT Prompting in Credulous Reasoning

Task Description: Given contexts and question, the context consists of facts and default rules. You first need to generate all extensions based on the given context and then answer the question according to the extensions. An extension is a set of non-contradictory facts and conclusions generated by rule-based reasoning. There may be multiple extensions in the context, and you need to find all the extensions according to the above steps. Next, the answer to the question is generated based on the generated extension.
If the question can be inferred under a certain extension, the answer label of the question is: "True";
If the negation of the question can be inferred under a certain extension, the answer label of the question is: "False";
If the question and the negation of the question both cannot be deduced under all extension, the answer label of the question is: "Unknown". Each context has a question.
The input format is: Facts:" ", Default Rules:" ". Question:" ".
The output format is: The extensions are:" ". The answer label of the question is:" ". You must generate all extensions and answer the question according to the extensions. Please read the context carefully. "Letś think step by step."

Box 9: Symbolic COT Prompting in Sketpical Reasoning

Task Description: Given contexts and question, and the context consists of facts and default rules. You need to first generate all extensions based on given context, then to answer the question according to the extensions.
To generate an extension in the context, **firstly**, to generate the initial upper and lower bounds fact sets of extension based on the instantiated default rules, the lower bound fact set is initialized with the original facts. The upper bound fact set should also include all the facts extracted from the rules. **Next**, the upper and lower bounds facts are updated using the conclusions generated by the rules. If the updated lower bound fact set is still a subset of the upper bound fact set, a fact is selected from the upper bound fact set and added to the lower bound set. **Then**, the rules are inferred based on the updated upper and lower bound fact sets, and the upper and lower bound fact sets are updated again using the generated conclusions. Specifically, the conclusions generated by the rules under the upper bound fact set should be included in the lower bound fact set. In comparison, the upper bound fact set should only contain the conclusions generated by the reduced rules in the lower bound fact set. **Finally**, the process is iterated until the upper and lower bounds facts are consistent and an extension is found. There may be multiple extensions in the context, and you need to find all the extensions according to the above steps. **Next**, the answer to the question is generated based on the generated extension.
If the question can be inferred under all extensions, and the negation of the question cannot be inferred under all extensions, the answer label of the question is: "True";
If the negation of the question can be inferred under all extensions, and the question cannot be inferred under all extensions, the answer label of the question is: "False";
If the question and the negation of the question cannot be deduced under a certain extension, the answer label of the question is: "Unknown".
The input format is: Facts:"", Default Rules:"". Question:"". The output format is: The extensions are:"". The answer label of the question is "".
For example: Facts: Toby is not noisy. Toby is handsome. Default Rules: If someoneA is handsome then he is delicious, unless he is not drab. If someoneA is not noisy then he is not drab, unless he is delicious. The first step is to generate the upper and lower bounds of the extension. The lower bound fact set is "Toby is not noisy. Toby is handsome.", and the upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is delicious. Toby is not drab.". Then, the upper bound facts are updated using the conclusion "Toby is delicious. Toby is not drab." generated by the lower bound fact set based on the default rule. The new upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is delicious. Toby is not drab.". Similarly, since the default rule does not generate new facts when reasoning on the upper bound fact set, the lower bound fact set remains unchanged. At this time, since the lower bound fact set is a subset of the upper bound fact set, a fact is randomly selected from the upper bound fact set and added to the lower bound fact set. The new lower bound fact set is updated to "Toby is not noisy. Toby is handsome. Toby is delicious." The new upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is not drab.". Then the default rule is inferred based on the new upper and lower bound fact sets, and the process is iterated until the upper and lower bounds of the extension are consistent. Finally, the context can generate two extensions: The extension 1 are: " Toby is not noisy. Toby is handsome. Toby is delicious."; The extension 2 are " Toby is not noisy. Toby is handsome. Toby is not drab.".
If the question is: Toby is not noisy. Since all expansions can infer the query " Toby is not noisy.", so the answer label for the question is: True.
If the question is: Toby is not drab. Although extensions 2 can deduce the query "Toby is not drab ", extension 1 cannot deduce this query, so the answer label for the question is: Unknown;
If the question is: Toby is not handsome. Since all three extensions can lead to the query "Toby is handsome", so the answer label for the question is: False.
You must generate all extensions and answer the question according to the extensions. Please read the context carefully. **Letś think step by step.**

Box 10: Symbolic COT Prompting in Credulous Reasoning

Task Description: Given contexts and question, and the context consists of facts and default rules. You need to first generate all extensions based on given context, then to answer the question according to the extensions. To generate an extension in the context, **firstly**, to generate the initial upper and lower bounds fact sets of extension based on the instantiated default rules, the lower bound fact set is initialized with the original facts. The upper bound fact set should also include all the facts extracted from the rules. **Next**, the upper and lower bounds facts are updated using the conclusions generated by the rules. If the updated lower bound fact set is still a subset of the upper bound fact set, a fact is selected from the upper bound fact set and added to the lower bound set. **Then**, the rules are inferred based on the updated upper and lower bound fact sets, and the upper and lower bound fact sets are updated again using the generated conclusions. Specifically, the conclusions generated by the rules under the upper bound fact set should be included in the lower bound fact set. In comparison, the upper bound fact set should only contain the conclusions generated by the reduced rules in the lower bound fact set. **Finally**, the process is iterated until the upper and lower bounds facts are consistent and an extension is found. There may be multiple extensions in the context, and you need to find all the extensions according to the above steps. Next, the answer to the question is generated based on the generated extension.
If the question can be inferred under a certain extension, the answer label of the question is: "True";
If the negation of the question can be inferred under a certain extension, the answer label of the question is: "False";
If the question and the negation of the question both cannot be deduced under all extension, the answer label of the question is: "Unknown".
The input format is: Facts: "", Default Rules:"". Question:"". The output format is: The extensions are:"". The answer label of the question is:"".
For example: Facts: Toby is not noisy. Toby is handsome..... Default Rules: If someoneA is handsome then he is delicious, unless he is not drab. If someoneA is not noisy then he is not drab, unless he is delicious. The first step is to generate the upper and lower bounds of the extension. The lower bound fact set is "Toby is not noisy. Toby is handsome.", and the upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is delicious. Toby is not drab.". Then, the upper bound facts are updated using the conclusion "Toby is delicious. Toby is not drab." generated by the lower bound fact set based on the default rule. The new upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is delicious. Toby is not drab.". Similarly, since the default rule does not generate new facts when reasoning on the upper bound fact set, the lower bound fact set remains unchanged. At this time, since the lower bound fact set is a subset of the upper bound fact set, a fact is randomly selected from the upper bound fact set and added to the lower bound fact set. The new lower bound fact set is updated to "Toby is not noisy. Toby is handsome. Toby is delicious." The new upper bound fact set is "Toby is not noisy. Toby is handsome. Toby is not drab.". Then the default rule is inferred based on the new upper and lower bound fact sets, and the process is iterated until the upper and lower bounds of the extension are consistent. Finally, the context can generate two extensions: The extension 1 are: "Toby is not noisy. Toby is handsome. Toby is delicious."; The extension 2 are "Toby is not noisy. Toby is handsome. Toby is not drab.".
If the question is: Toby is not drab. Since the extension 2 can lead to the query " Toby is not drab ", so the answer label for the question is: True;
If the question is: Toby is intelligent. Since extensions 1 and 2 cannot deduce the query and the negation of query, so the answer label for the question is: Unknown;
If the question is: Toby is not handsome. Since the negation of query fact "Toby is handsome." can be derived in extensions, so the answer label for the question is: False.
You must generate all extensions and answer the question according to the extensions. Please read the context carefully. **Letś think step by step.**

A.4 MultiLogicNMRer Prompts

Box 11: Grounding Prompting in Grounding Module

Task Description: Given a set of facts and a rule, you need to instantiate the rules based on given facts. Instantiation requires the replacement of pronouns in rules with individuals from the fact.

Example 1: Godwin is not sour. Godwin is short. Godwin is scared. Godwin is wild. Godwin is expensive. Godwin is not bad. Godwin is not straightforward. Godwin is anxious. Godwin is not stubborn. Godwin is not zany. Godwin laugh Connor. Godwin esteem Connor. Godwin is not immediate. Godwin is persistent. The rule is: If someoneA laugh someoneB and he is not stubborn then he is old, unless he is not poor.
The output is: If Godwin laugh Connor and Godwin is not stubborn then Godwin is old, unless Godwin is not poor.

The output format is: The output is:“ ”.

Note that you need to output all instantiation rules.

Box 12: Fact Extraction Prompting in Upper and Lower Bound Initialization Module

Task Description: Given a set of facts and a rule, you need to extract all instantiated facts in the rule.

Example 1: The rule is: If Godwin laugh Connor and Godwin is not stubborn then Godwin is old, unless Godwin is not poor or Godwin is unhappy.
The output is: Godwin laugh Connor. Godwin is not stubborn. Godwin is old. Godwin is not poor. Godwin is unhappy.

The output format is: The output is: “ ”.

Note that you only need to output all instantiated facts in the rule, do not print the contents of the prompt, and don’t output the same facts repeatedly.

Box 13: Split Rule Prompting in Reduction Module

Task Description: Given a rule, The rule format is: If A then B, unless C. The A is the prerequisite, the B is the conclusion, and the C is the justification. You need to output all prerequisite, conclusions, and justifications in this rule.

Example 1: The rule is: If Brice is emotional then Brice is beige, unless Brice is sufficient.
The output is: prerequisite: "Brice is emotional.", conclusion:"Brice is beige. ", justification: "Brice is sufficient.".

Example 2: The rule is: If Cadman is historical and Cadman is emotional then Cadman is swift, unless Cadman is smart or Cadman is happy.
The output is: prerequisite: "Cadman is historical. Cadman is emotional.", conclusion: "Cadman is swift."; justification: "Cadman is smart. Cadman is happy. ".

The output format is: The output is: prerequisite: " ", conclusion:" ", justification: " ".

Box 14: Reasoning Prompting in Reasoning Module

Task Description: Given facts and a rule. You need to reason about the rules based on facts. The rule format is usually: If A then B. The A is the prerequisite, the B is the conclusion. If the prerequisite A is in the facts, you can deduce conclusion B. If the prerequisite A is not in the facts, then you can not deduce the conclusion B, so your output is: None.

Example 1: The input facts are: Godwin is not sour.Godwin is short.Godwin is scared. Godwin is wild. Godwin is expensive. Godwin is not bad. Godwin is not straightforward. Godwin is anxious. Godwin is not sour. Godwin is not zany. Godwin laugh Connor. Godwin esteem Connor. Godwin is immediate. Godwin is persistent. The rules are: If Godwin is not sour and immediate then Godwin is not lovely.
The output is: Godwin is not lovely.

Example 2: The facts are: Juliana is not old. Juliana is not anxious. Juliana is asleep. Juliana is giant. Juliana is not short. Juliana is comfortable. Juliana is not fearless. Juliana is aggressive. Juliana is not hot. Juliana is not southern. Juliana is not technical. Juliana is not educational. Juliana is not octagonal. Juliana is low. Juliana is not poor. The rule is: If someoneA is not short and not low then Juliana is persistent.
The output is: None.

The output format is: The output is:" ".

Note that you only need to output rule conclusions that can be inferred, not facts and reasoning processes.

Box 15: Extraction Answer in Skeptical Reasoning Mode

Dask Description: Given the extensions and question, and each extension consists of facts. you need to answer the questions according to the given extensions.
If the question can be inferred under all extensions, and the negation of the question cannot be inferred under all extensions, the answer label of the question is: "True".
If the negation of the question can be inferred under all extensions, and the question cannot be inferred under all extensions, the answer label of the question is: "False".
If the question and the negation of the question cannot be deduced under a certain extension, the answer label of the question is: "Unknown".
The input format is: The extension 1 are:"". The extension 2 are:"". The question is:"". The output format is: The answer is:"".
For example: The extension 1 are: "Magnus and Malcolm is spurn. Magnus is difficult. Magnus is arrogant. Magnus is nasty. Magnus is dangerous. Magnus is important. Magnus is vast. Magnus is not handsome.". The extension 2 are: "Magnus is not important. Magnus is vast. Magnus is dramatic. Magnus is not handsome. Magnus is poor. Magnus is sensitive.".
The question is: "Magnus is not important". The answer is: "Unknown".
The question is: "Magnus is vast". The answer is: "True".
The question is: "Magnus is handsome.". The answer is: "False".
Note that you only need to generate the answer label for the question without giving an explanation or justification. Please read all extensions carefully and answer the question.

Box 16: Extraction Answer in Credulous Reasoning Mode

Dask Description: Given the extensions and question, and each extension consists of facts. you need to answer the questions according to the given extensions.
If the question can be inferred based on a certain facts set, the answer label of the question is "True";
If the negation of the question can be inferred based on a certain facts set, the answer label of the question is "False";
If the question and the negation of the question both cannot be deduced under all facts set, the answer label of the question is "Unknown".
The input format is: The extension 1 are:"". The extension 2 are:"". The question is:"". The output format is: The answer is:"".
For example: The extension 1 are: "Magnus and Malcolm is spurn. Magnus is difficult. Magnus is arrogant. Magnus is nasty. Magnus is dangerous. Magnus is important. Magnus is vast. Magnus is handsome.". The extension 2 are: "Magnus is not important. Magnus is vast. Magnus is dramatic. Magnus is not handsome. Magnus is poor. Magnus is sensitive.".
The question is: "Magnus is happiness.". The answer is: "Unknown".
The question is: "Magnus is important". The answer is: "True". The question is: "Magnus is not dramatic.". The answer is: "False".
Note that you only need to generate the answer label for the question without giving an explanation or justification. Please read all extensions carefully and answer the question.

A.5 Experimental Setting and Results

We use the LoRA fine-tuning method to fine-tune the open-source LLMs (LLAMA3-8B-Instruct[8], gemma-7b-it[9], Mistral-7B-Instruct_v0.2[10]), respectively. The parameters of the fine-tuned model are shown in Table 6. All fine-tuning experiments are completed on a single NVIDIA 4090 GPU based on the unsloth[11] framework.

Table 6: Fine-tuning parameters of open-source LLMs.

Parameter	Value
per_device_train_batch_size	4
gradient_accumulation_steps	4
warmup_steps	10
max_steps	100
weight_decay	0.01
optim	Adamw_8bit
seed	3407

Table 7: Results of models on MultiLogicNMR in **credulous** reasoning.

Model Extension	Method	Test (Accuracy %)					
		1	2	3	4	5	Avg
LLAMA3	Zero Shot	36.7	41.6	43.0	42.3	42.3	41.2
GPT3.5		45.3	44.3	47.3	47.3	49.8	46.8
DeepSeek		41.0	42.0	39.6	36.3	38.3	39.5
GPT-4o		59.3	56.0	58.3	62.6	62.7	59.8
LLAMA3	Few Shot	33.0	35.3	36.0	37.0	33.7	34.9
GPT3.5		43.3	40.6	39.6	42.6	41.6	41.6
DeepSeek		55.6	51.0	52.6	52.0	55.3	53.3
GPT-4o		60.0	57.0	66.0	62.0	65.0	62.0
GPT3.5	Zero Shot CoT	40.6	45.0	42.6	45.0	43.6	43.4
DeepSeek		58.0	59.0	59.0	61.7	59.3	59.4
GPT-4o		57.3	52.7	54.3	50.7	52.0	53.4
GPT3.5	Symbolic CoT	58.7	53.7	55.7	57.0	55.3	56.1
DeepSeek		63.3	66.0	63.7	62.7	66.0	64.3
GPT-4o		52.5	44.1	51.7	54.1	57.5	52.0
LLAMA3	Fine Tune	55.6	66.7	71.7	69.6	73.7	67.4
Mistral		53.3	62.0	65.6	59.6	65.6	61.3
Gemma		59.3	66.3	73.0	**71.7**	73.3	68.7
MLNMRer (LLAMA3)	Ours	73.3	62.7	64.0	67.3	65.7	66.7
MLNMRer (GPT3.5)		75.7	69.9	72.3	69.3	58.7	69.2
MLNMRer (DeepSeek)		**79.6**	**74.3**	**74.3**	70.0	**74.0**	**74.5**

[8] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

[9] https://huggingface.co/google/gemma-7b-it

[10] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

[11] https://github.com/unslothai/unsloth

Table 8: Results of models on the MultiLogicNMR_OOD in **credulous** reasoning.

Model Extension	Met -hod	Test (Accuracy %) 1	2	3	4	5	Avg
LLAMA3	Zero Shot	36.7	34.0	32.6	38.6	35.0	35.4
GPT3.5		39.0	40.3	38.3	39.6	40.0	39.5
DeepSeek		44.0	43.0	45.3	40.3	47.3	44.0
GPT-4o		44.7	53.0	43.7	42.0	39.0	44.5
GPT3.5	Few Shot	39.7	35.0	35.3	37.7	35.7	36.7
DeepSeek		55.1	60.1	60.0	53.3	56.4	57.0
GPT-4o		43.4	46.4	47.7	49.0	50.3	47.1
GPT3.5	Zero Shot CoT	35.0	37.8	39.0	36.3	39.3	37.5
DeepSeek		56.0	61.3	59.4	60.7	57.8	59.2
GPT-4o		49.0	44.3	46.7	47.0	48.7	47.1
GPT3.5	Sym-bolic CoT	46.7	47.6	45.3	43.3	48.3	46.2
DeepSeek		61.3	62.3	62.0	60.3	60.7	61.3
GPT-4o		53.7	51.0	53.7	53.3	52.7	52.8
LLAMA3	Fine Tune	49.6	50.7	52.7	48.6	49.6	50.3
Mistral		34.0	46.7	39.3	44.0	40.0	40.8
Gemma		43.3	46.3	46.0	47.0	51.3	46.8
MLNMRer (LLAMA3)	Ours	54.7	52.7	51.0	53.6	55.0	53.4
MLNMRer (GPT3.5)		46.7	47.7	45.3	44.7	51.0	47.1
MLNMRer (DeepSeek)		52.0	52.3	48.0	54.3	53.3	52.4

Table 9: Results of models on the MultiLogicNMR_NL in skeptical reasoning.

Model Extension	Met -hod	Test (Accuracy %) 1	2	3	4	5	Avg
LLAMA3	Zero Shot	35.3	40.7	34.0	34.0	35.0	35.8
GPT3.5		30.3	33.6	36.7	35.0	39.1	34.9
DeepSeek		38.7	43.0	42.0	40.3	45.7	41.9
GPT-4o		43.0	38.7	47.3	41.0	45.6	43.1
GPT3.5	Few Shot	39.3	39.7	38.3	37.0	41.7	39.2
DeepSeek		45.6	45.3	47.3	43.3	47.4	45.8
GPT-4o		46.0	44.3	50.0	45.6	49.1	47.0
GPT3.5	Zero Shot CoT	36.0	37.7	36.7	35.7	38.7	36.9
DeepSeek		50.3	48.0	50.7	51.3	52.8	50.6
GPT-4o		48.3	49.3	51.3	52.3	52.5	50.8
GPT3.5	Sym-bolic CoT	36.3	44.7	43.7	41.3	44.3	42.1
DeepSeek		50.3	51.0	54.0	53.3	54.0	52.5
GPT-4o		48.0	44.6	45.0	47.0	48.1	46.5
LLAMA3	Fine Tune	52.7	53.6	58.3	59.0	60.6	56.8
Mistral		43.8	46.0	48.6	45.3	46.4	46.1
Gemma		36.3	38.0	37.0	39.0	41.1	38.3
MLNMRer (GPT3.5)	Ours	38.7	38.7	41.0	40.3	42.1	40.1
MLNMRer (DeepSeek)		52.0	47.7	52.6	42.3	44.1	47.7

A.6 Analysis and Case Study

Figure 5 also shows the distribution of answers generated by different methods in credulous reasoning, and a similar conclusion can be drawn as in Figure 5 in skeptical reasoning. Questions with Unknown are still very challenging to prompting and fine-tuning methods. At the same time, the proposed neural symbolic framework MultiLogic-NMRer has significantly improved the performance of questions with Unknown. To further explain this phenomenon, an example of the answer and explanation generated by GPT-4o in skeptical reasoning is shown in Box 17. According to the experimental results, it can be found that the model gave the correct answer to questions 1 and 3 with True and False answers, and GPT-4o generated a reasonable and correct reasoning path, respectively, which corresponds to a certain extension generated in the context. However, the model made a wrong prediction for question 2 with the answer Unknown. First, according to the model's explanation, the answer generated for this question should be False, not True. This shows that the GPT-4o has inconsistencies in the reasoning process. In addition, the model explanation only contains one extension, so it cannot correctly implement non-monotonic reasoning.

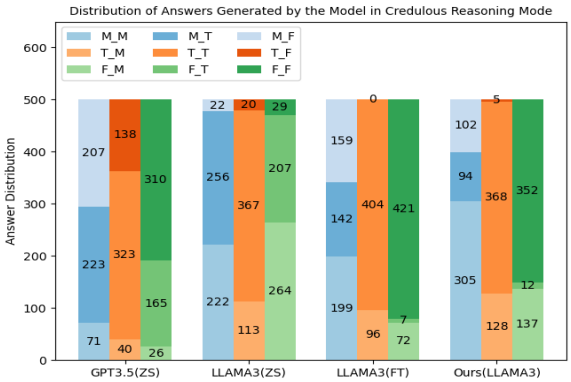

Figure 5: The distribution of answers generated by models in skeptical reasoning. The ZS represents the zero-shot prompt-based model and FT represents the fine-tuned model.

Box 17: An Example of Reasoning by the GPT-4o in Skeptical Reasoning Mode

Facts: Toby is not noisy. Toby is not intelligent. Toby is handsome. Toby is not alert. Toby is honest. Toby is important. Toby is young. Toby is not pessimistic. Toby scorn Benjamin. Toby is passionate. Toby is not gorgeous. Toby is scared.
Default Rules: If someoneA is handsome and not intelligent then he is delicious, unless he is not drab. If someoneA is delicious and not alert then he is huge, unless he is grieving or he is not nasty. If someoneA is not pessimistic and scared then he is not foreign,unless he is not sensitive or he is not bossy. If someoneA is not noisy and delicious then he is different, unless he is dead. If someoneA is not gorgeous and honest then he is grieving, unless he is huge. If someoneA is different then he is not numerous, unless he is not confused or he is not financial. If someoneA scorn someoneB and someoneA is not grieving then he is not financial, unless he is not distinct or he is not proud. If someoneA is important then he is not distinct, unless he is not financial. If someoneA is passionate and not huge then he is not proud, unless he is not financial or he is not spotless. If someoneA is not numerous and young then he is pleasant, unless he is not financial.

Question 1: Toby is delicious. **Answer 1:** True. **Generate Answer 1:** True
Explanation 1: Toby is handsome and not intelligent, which triggers the rule that if someoneA is handsome and not intelligent then he is delicious unless he is not drab. Since there is no evidence in the context indicating that Toby is not drab, we conclude that Toby is delicious.

Question 2: Toby is not huge. **Answer 2:** Unknown. **Generate Answer 2:** True
Explanation 2: Toby is handsome and not intelligent, making him delicious. Toby is delicious and not alert, with no mention of him grieving or being nasty, so he is huge. Hence, being huge implies he cannot be not huge, so the statement Toby is not huge is False, making the question "Toby is not huge" answered as True for the opposite.

Question 3: Toby is not different. **Answer 3:** False. **Generate Answer 3:** False
Explanation 3: Toby is not noisy. Toby is handsome and not intelligent, which makes him delicious. Being delicious and not alert makes Toby huge. Furthermore, still being delicious makes Toby different because he is not dead, thus confirming that Toby is different. Therefore, the statement "Toby is not different" is false.