LogicNMR: Probing the Non-monotonic Reasoning Ability of Pre-trained Language Models

Yeliang Xiu[1], **Zhaohao Xiao**[2]*, **Yongmei Liu**[1]*
[1]Dept. of Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[2]School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou 510006, China
xiuyliang@mail2.sysu.edu.cn; xiaozhanhao@gpnu.edu.cn; ymliu@mail.sysu.edu.cn

Abstract

The logical reasoning capabilities of pre-trained language models have recently received much attention. As one of the vital reasoning paradigms, non-monotonic reasoning refers to the fact that conclusions may be invalidated with new information. Existing work has constructed a non-monotonic inference dataset δ-NLI and explored the performance of language models on it. However, the δ-NLI dataset is entangled with commonsense reasoning. In this paper, we explore the pure non-monotonic reasoning ability of pre-trained language models. We build a non-monotonic reasoning benchmark, named LogicNMR, with explicit default rules and iterative updates. In the experimental part, the performance of popular language models on LogicNMR is explored from the perspectives of accuracy, generalization, proof-based traceability and robustness. The experimental results show that even though the fine-tuned language models achieve an accuracy of more than 94.4% on LogicNMR, they perform unsatisfactorily, with a significant drop, in generalization and proof-based traceability.

1 Introduction

Non-monotonic reasoning, also called defeasible reasoning, is one of the important reasoning modes in logic, which has been extensively studied in classical AI. The term non-monotonic reasoning was first introduced by Minsky (1975). Generally, non-monotonic reasoning refers to the fact that conclusions may be invalidated with new information (Lukaszewicz, 1990). The research on non-monotonic reasoning in traditional AI mainly focuses on formalizing non-monotonic reasoning via different logics, such as default logic (Reiter, 1980), circumscription (McCarthy, 1980), and Autoepistemic Logic (Moore, 1983).

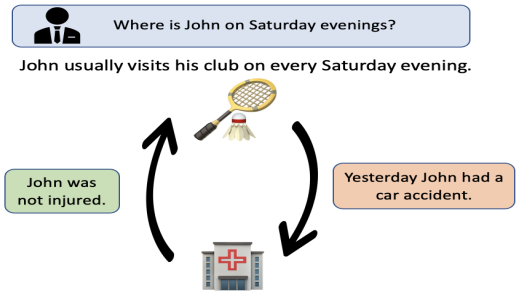

Figure 1: An example of non-monotonic reasoning in everyday life.

Non-monotonic reasoning is widespread in everyday life and plays a crucial role in both daily decision-making (Benferhat et al., 2000; Szalas, 2019) and legal reasoning (Lawsky, 2017). Most of what we learn about the world is in terms of generics, properties that hold "in general", but with exceptional cases. When we say "birds can fly", we mean "birds can usually fly", and there are exceptional cases such as ostrich or wounded birds. Such rules are called default rules. Figure 1 shows a typical example of non-monotonic reasoning. Suppose we desire to find John on Saturday evening and know he usually visits his club every Saturday evening. We thus infer that he will be at his club. However, if we are told John had a car accident yesterday, we would conclude he will likely be in a hospital. Then if we get to know John was not injured, we redraw the conclusion he visits his club. This example illustrates the dynamic nature that a context is constantly updated with new information and queried.

Recently, whether pre-trained language models truly have logical reasoning abilities has received extensive attention. Although pre-trained language models have made significant progress on many natural language understanding tasks, such as knowledge-based question answering (Lv et al., 2020) and com-

*Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 3616 - 3626
December 7-11, 2022

Table 1: An example from the δ-NLI dataset.

Premise:	Old man crafting something in his workshop.
Hypothes:	An old man is working.
Update:	The man is wearing pajamas and is chuckling.
Type:	strengthener / weakener [Answer: W]

Table 2: A simplified example from (Clark et al., 2020).

Facts:	Arthur is a bird.
Rules:	If someone is a bird and not abnormal then they can fly. If someone is a bird and wounded then they are abnormal.
Query:	Arthur can fly. True/false? [Answer: T]

monsense reasoning (Bhagavatula et al., 2020), etc, some research has shown that the prediction of pre-trained language models is easily affected by spurious correlations (Kaushik and Lipton, 2018; Jiang and Bansal, 2019), so it is difficult to judge the logical reasoning abilities of the evaluated models.

It is still in the preliminary research stage to probe whether pre-trained language models have non-monotonic reasoning mechanisms. Rudinger et al. (2020) construct a non-monotonic inference dataset δ-NLI through crowdsourcing based on three existing datasets. For δ-NLI, the authors develop a classification and generation task, and demonstrate that the classification task is easily solved by pretrained language models, but the generation task is much more challenging. Table 1 is an example of the classification task. However, δ-NLI entangles non-monotonic reasoning with commonsense reasoning. For instance, to solve the above example, we need the commonsense knowledge "people usually wear pyjamas when they are resting".

To disentangle deductive reasoning from commonsense reasoning, and explore pure deductive reasoning capabilities of pre-trained language models, Clark et al. (2020) introduce a synthetic dataset with explicit rules and facts, and show that fine-tuned language models perform well on this dataset. Table 2 is a simplified example from this dataset. In their work, rules have the semantics of logic programs with negation (Apt et al., 1988). Thus they make the closed-world assumption (CWA) (Reiter, 1977): unless an atomic sentence is known to be true, it can be assumed to be false. In the above example, all the facts and rules needed for reasoning are explicitly given. Since we cannot infer that Arthur is abnormal, we assume he is not abnormal, hence we deduce that he can fly. But if we are also given Arthur is wounded, we would withdraw that he can fly. So CWA is only a special case of non-monotonic reasoning.

In this paper, inspired by the research methodology of (Clark et al., 2020), different from the research problem of (Rudinger et al., 2020), we explore the pure non-monotonic reasoning abilities of pre-trained language models, disentangling from commonsense reasoning. We propose LogicNMR, a non-monotonic reasoning benchmark with three distinguished features. First, each context is given by explicit facts and default rules such as "a bird can fly unless he is wounded". So we handle explicit non-monotonic reasoning rather than implicit one by δ-NLI, and we deal with non-monotonic reasoning in a more general way than CWA. Second, each context is repeatedly updated with new facts and queried. This is in line with the phenomenon that human constantly receive new information and redraw conclusions. Third, the labels of the dataset are automatically generated by resorting to a formal non-monotonic reasoning solver and hence guaranteed with correctness. The non-monotonic reasoning ability in pre-trained language models are explored from accuracy, generalization, proof-based traceability and robustness. The experimental results reveal that even though the fine-tuned language models achieve a high accuracy on the in-distribution samples in LogicNMR, they perform unsatisfactorily, with a significant drop, in generalization and proof-based traceability.

2 Related Work

Many benchmarks involve logical reasoning, but entangled with commonsense reasoning, On one hand, natural language inference (NLI) is to determine the inference relation between two

texts, including entailment, contradiction, or neutral. For example, Bowman et al. (2015) present a significant NLI benchmark SNLI, a collection of 570k English sentence pairs. Richardson et al. (2020) explore the symbolic reasoning and monotonic reasoning abilities in pre-trained language models through semantic fragments. On the other hand, in machine reading comprehension, LogicQA (Liu et al., 2020) and ReClor (Yu et al., 2020) are two popular multiple-choice datasets involving complex logical reasoning, such as deductive reasoning and abductive reasoning. Li et al. (2022) and Xu et al. (2022) construct relationship graphs by extracting the basic units in the context, and then combine pre-trained language models and graph neural networks to solve LogicQA and ReClor.

Following (Rudinger et al., 2020), several other works focus on non-monotonic reasoning. To further explain why the updated information can cause the credibility of the original conclusion to change, Brahman et al. (2021) use distant supervision to generate reasons for δ-NLI. Madaan et al. (2021a) generate influence graphs through transfer learning to effectively improve performance on defeasible reasoning tasks. Madaan et al. (2021b) propose a model that can simulate thinking about question scenarios based on influence graphs to enhance the performance of defeasible reasoning tasks.

The work of Clark et al. (2020) initiated a line of research to explore logical reasoning capabilities in language models. Tian et al. (2021) present the LogicNLI benchmark for first-order logical reasoning and propose proof-based traceability to more effectively evaluate the logical reasoning abilities of language models. Saeed et al. (2021) propose a dataset RuleBERT with rules with probability in order to teach pre-trained language models to reason on soft Horn rules. Dalvi et al. (2021) introduce a dataset EntailmentBank with explanations in the form of entailment trees.

3 Default Logic and ASP

In this work, we choose Reiter (1980)'s default logic, one of the major formalisms for non-monotonic reasoning, as the logic underlying LogicNMR. A default rule is in the form of $\alpha : \beta_1, \beta_2, \ldots, \beta_m/\gamma$, where α, β_i and γ are formulas in first-order logic, α is called the prerequisite, $\beta_1, \beta_2, \cdots, \beta_m$ the justifications, and γ the conclusion. The interpretation of the default rule is that if you can infer α, and $\beta_1, \beta_2, \cdots, \beta_m$ are consistent, then infer γ. A default theory is a pair $T = \langle W, D \rangle$, where W is a set of facts, which are first-order sentences, and D is a set of default rules. For example, a default theory T_0 consists of $W_0 = \{fresh(A), afraid(A)\}$ and $D_0 = \{fresh(x) : \neg afraid(x)/cute(x)\}, fresh(x) : afraid(x)/\neg worried(x)\}$. A set of sentences E is an extension of $T = \langle W, D \rangle$ iff for every sentence ϕ, $\phi \in E$ iff $W \cup \Delta \models \phi$, where Δ is the set of γ s.t. $\alpha : \beta_1, \beta_2, \ldots, \beta_m/\gamma$ is in D, $\alpha \in E$, and $\neg\beta_i \notin E$ for $i = 1, \ldots, m$. Here $\models$ denotes the logic entailment relation for first-order logic. For example, T_0 has a unique extension $E_0 = W_0 \cup \{\neg worried(A)\}$.

In this paper, to reduce the complexity of the default theory, we make a set of restrictions. Variables and constants are terms, and $P(t_1, ..., t_n)$ is an atom when P is an n-ary predicate symbol and $t_1, .., t_n$ are terms. A literal is an atom or the negation of an atom. W is restricted to be a set of literals. α is the conjunction of at most two literals, there are at most two justifications, each justification is a literal, and γ is a literal. In addition, we require that: for each default theory, for any justification of any default rule, its negation does not appear as the conclusion of any other default rule. It is easy to show that under the above restrictions, each theory T has a unique extension, written $E(T)$. Then for any sentence ϕ, we write $T \vdash \phi$ if $\phi \in E(T)$. We define $Ans(T, \phi)$ as follows:

$$Ans(T, \phi) = \begin{cases} T, & \text{if } T \vdash \phi \text{ and } T \nvdash \neg\phi \\ F, & \text{if } T \nvdash \phi \text{ and } T \vdash \neg\phi \\ M, & \text{if } T \nvdash \phi \text{ and } T \nvdash \neg\phi \end{cases}$$

Chen et al. (2010) show that in the propositional case, each default theory is equivalent to an answer set program. Answer Set Programming (ASP) (Brewka et al., 2011) is an approach to declarative programming with efficient solvers, such as Clyngor[1]. Under our restrictions, a default theory T can be converted to an equivalent answer set program $\delta(T)$ as

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

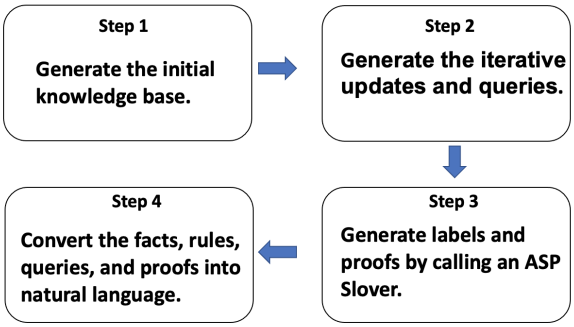

Figure 2: Pipeline of LogicNMR sample generation

follows: each default rule $\alpha : \beta_1, \beta_2, ..., \beta_m/\gamma$ is converted to an ASP rule

$$\gamma \leftarrow \alpha, \text{not } \neg\beta_1, \text{not } \neg\beta_2, \cdots, \text{not } \neg\beta_m.$$

Then the unique extension for T can be computed by computing the answer set for $\delta(T)$ by resolving to an ASP solver such as Clyngor.

4 LogicNMR Benchmark

To probe pure non-monotonic reasoning abilities in language models, we generate a non-monotonic reasoning benchmark LogicNMR with explicit facts and default rules and iterative updates and queries. Our dataset is available at https://github.com/sysulic/LogicNMR.

Overview of Dataset Generation. Figure 2 gives an overview of the generation process for a sample in LogicNMR. First, we generate an initial knowledge base (KB) containing default rules and facts. Then, we generate the iterative updates and queries. Next, we generate the label and associated proof for each update and query. Finally, we convert the initial KB, updates, queries and proofs into synthetic English using simple templates.

The predicate pool for LogicNMR includes unary and binary predicates, where the unary ones are 529 adjective words from (Tian et al., 2021), and the binary ones are 46 adjective words that describe relationships between subjects. Each sample is restricted to a subject, which is a name.

Figure 3 shows a LogicNMR sample represented with formulas. Here, T refers to the initial KB where there is a single fact and multiple default rules. U_i is the new fact for the i-th update, and Q_i is the query for the i-th update. For each query, when the label is T or F, a proof is a sequence of intermediate necessary conclusions during the reasoning process for the query.

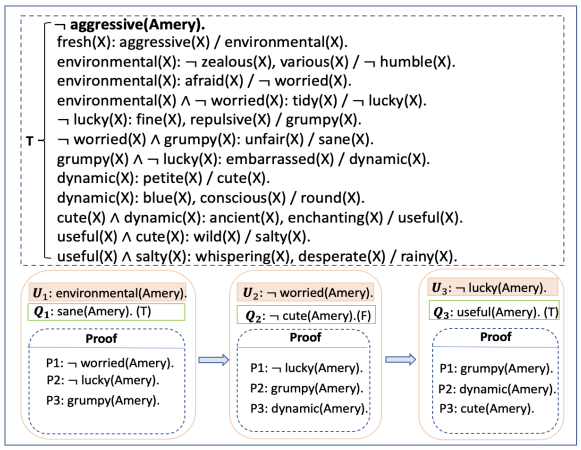

Figure 3: An example in the LogicNMR dataset is represented by the default logic.

In the following, we give detailed descriptions for each step of Figure 2.

Initial KB Generation. To generate the first default rule, we randomly select predicates from the predicate pool as the predicates for the prerequisite, justifications, and conclusion. We then add negations to the prerequisite, justification, and conclusion atoms with a probability of 0.5. Next, for every new default rule, the prerequisite literals are randomly selected from the existing conclusion literals, the justification literals are randomly generated and different from the prerequisite literals, and the conclusion literal is randomly generated so that it is different from the prerequisite literal and its negation does not appear in justifications for existing default rules. Finally, we generate the initial facts by instantiating prerequisite literals or the negation of justification literals of default rules with the unique subject. For example, in Figure 3, $\neg aggressive(Amery)$ is generated by negating the justification literal of the first default rule.

Iterative Updates and Queries Generation. Each LogicNMR sample is updated for five times. The updates are generated from the prerequisite and justification literals of default rules. If generated from prerequisite literals, the update is the instantiation of the prerequisite literal; if generated from justification literals, the update is the instantiation of the negation of the justification literal. The queries

are generated from the instantiation of conclusion literals of default rules and being negated with a probability of 0.5.

Task Definition. As shown in Figure 3, a sample in the LogicNMR dataset is a triple (T, U, S), where T is the initial KB, $U = \langle U_1, ..., U_5 \rangle$ is the sequence of updates, and $Q = \langle Q_1, ..., Q_5 \rangle$ is the sequence of queries. For $i = 1, \ldots, 5$, we let $T_i = T \cup \{U_j \mid 1 \leq j \leq i\}$. The task is to decide the answer $A = \langle A_1, ..., A_5 \rangle$ where $A_i = Ans(T_i, Q_i)$.

Labeling and Proofs. To compute $Ans(T_i, Q_i)$, we convert T_i into an answer set program $\delta(T_i)$, call Clyngor to compute the unique answer set, and then check if Q_i or $\neg Q_i$ is in the answer set. If Q_i or $\neg Q_i$ is in the answer set, we produce a sequence of proofs for it, where each proof is the conclusion of the default rules applied in the reasoning chain. For example, for Q_1 in Figure 3, with U_1, using the third default rule, we get $\neg worried(Amery)$; then using the fourth default rule, we get $\neg lucky(Amery)$; next, using the fifth default rule, we get $grumpy(Amery)$; finally, we obtain Q_1.

Conversion into English. We convert the initial KB, updates, queries, and proofs into English by using simple templates. For example, the default rule "$dynamic(X)$: $petite(X)/cute(X)$" is translated into "If someone is dynamic then he is cute, unless he is petite".

Dataset Statistics. Table 3 shows the statistical information of LogicNMR. The training, validation, and test sets in LogicNMR contain 5k, 2k, and 2k samples, respectively. The number of default rules in the initial KB is at most 12, and the number of initial facts is at most 2. Avg.Length and Max.Length represent the average and maximum number of words in the initial KB, respectively. To explore robustness of language models on LogicNMR, for each sample, there are six irrelevant facts and six irrelevant default rules. To reduce the bias caused by label imbalance, we require the labels in the dataset to be balanced. Also, the predicate pools for the training, validation, and test sets are different to avoid answering queries according to the correlations among predicates.

Table 3: Statistics of LogicNMR.

Data	Statistics	Train	Val	Test
LogicNMR	#Samples	5000	2000	2000
	#Queries	25000	10000	10000
	Avg.Length	199	198	198
	Max.Length	230	228	230
	#Initial Rules	12		
	#Initial Facts	1 or 2		
	#Updated Facts	5		
	#Irrelevant Facts	6		
	#Irrelevant Rules	6		
	#Subjects	363	100	100
	#Predicates	364	105	106
	Labels	F:T:M=1:1:1		

5 Experiments

In this section, by following (Tian et al., 2021), we explore the non-monotonic reasoning ability of pre-trained language models in terms of accuracy, generalization, proof-based traceability, and robustness, respectively, based on LogicNMR.

5.1 Experimental Settings

In this paper, we would like to investigate three mainstream pre-trained language models: BERT-large (Devlin et al., 2019) , RoBERTa-large (Liu et al., 2019), and GPT2 (Radford et al., 2019). The hyperparameters in such language models are shown in Table 4.

Table 4: The hyperparameter settings.

Paras.	BERT	RoBERTa	GPT2
batch size	32	24	32
learning rate	1e-5	1e-5	1e-5
decay rate	0.01	0.01	0.01
epochs	20	20	50
optimizer	ADAMW	ADAMW	ADAMW

5.2 Experimental Results

5.2.1 Accuracy

Table 5 shows the accuracy results of RoBERTa, BERT, and GPT2 models on the LogicNMR dataset. It is not difficult to find that the language models achieve a high accuracy on answering queries in the LogicNMR dataset after fine-tuning. Generally, RoBERTa has the top performance. With the number of updates

Table 5: The accuracy results of three language models on LogicNMR.

Models	Accuracy(%)				
	U=1	U=2	U=3	U=4	U=5
RoBERTa	99.6	**98.6**	**98.1**	**96.8**	**96.3**
BERT	**99.7**	98.4	95.9	94.9	94.7
GPT2	99.2	97.2	97.1	96.6	94.4

U represents the number of updates to the KB.

increasing, all language models yield only a slight drop. The high accuracy on answering non-monotonic reasoning queries looks like that the language models have already mastered the non-monotonic reasoning ability after fine-tuning. However, it is also possible that they only perform well because of their strong fitting ability on the in-distribution samples. To further probe whether the language models master the ability of non-monotonic reasoning, we still need to evaluate how they performs in terms of generalization and proof-based traceability on LogicNMR.

5.2.2 Generalization

In this paper, the generalization is used to measure whether the model truly understands non-monotonic reasoning, i.e., assesses how a model performs on the out-of-distribution samples. In CLUTRR (Sinha et al., 2019), the metric of generalization is measured by training a model on samples with inference depths less than or equal to K and then testing it on samples with an inference depth greater than K. However, different from monotonic reasoning in first-order logic, non-monotonic reasoning pays attention to the dynamicness of the knowledge bases. In other words, to evaluate whether a model masters the non-monotonic reasoning ability, we need to test its performance varying different updates. More formally, if a model has learned non-monotonic reasoning on a knowledge base with K updates, it should also be effective on knowledge bases with different numbers of updates. In this way, not only we make sure that samples in the tthe raining set are balanced in size, but also we can independently see how well models generalize in terms of different updates. Therefore, we define the generalization metric ofa model over the update number, noted Avg*, as its average accuracy on the samples with updates number $U \neq K$ with being trained on the samples with the update number $U = K$.

Table 6: The generalization results of the three language models on LogicNMR.

Model	U	Test(Accuracy(%))					Avg*
		U=1	U=2	U=3	U=4	U=5	
Ro-BERTa	U=1	**99.6**	60.1	50.0	54.6	53.2	54.4
	U=2	98.1	**98.6**	83.0	58.2	50.0	72.3
	U=3	81.7	94.5	**98.1**	93.7	75.4	86.3
	U=4	69.3	90.0	96.7	**96.8**	94.2	87.5
	U=5	57.5	61.4	89.6	96.3	**96.3**	76.2
BERT	U=1	**99.7**	64.5	54.3	55.4	51.5	56.4
	U=2	90.9	**98.4**	76.8	52.4	51.3	67.8
	U=3	64.6	66.9	**95.9**	69.6	52.1	63.3
	U=4	59.7	63.1	72.6	**94.9**	74.7	67.5
	U=5	43.7	45.6	53.4	67.1	**94.7**	52.4
GPT2	U=1	**99.2**	46.9	57.1	52.2	50.3	51.6
	U=2	28.5	**97.2**	73.2	64.7	60.3	56.6
	U=3	59.7	74.8	**97.1**	80.6	67.5	70.6
	U=4	60.7	64.7	88.1	**96.6**	94.6	77.0
	U=5	44.7	49.3	59.3	82.9	**94.4**	59.0

Avg* is the average accuracy on $U \neq K$.

Table 6 shows the generalization results of the language models for the number of updates on LogicNMR. First, the generalization performance of the RoBERTa is the best among the three models. Specifically, throughout the whole LogicNMR, the average generalization metric Avg* of RoBERTa, BERT, and GPT2 are 75.3%, 61.5%, and 62.9%, respectively. However, for each model at each update, its Avg* is lower than its average accuracy shown in Table 5. Second, the more significant difference between the number of updates of the testing set and the number of updates of the training set, the worse the generalization performance of the language models. For example, RoBERTa trained on the samples with update number $U = 3$ has average accuracies of 93.7% and 75.4% on the samples with $U = 4$ and $U = 5$, respectively. It reflects that the difference on the distributions of the samples is a significant challenge for language models, causing a unsatisfying performance on generalization.

5.2.3 Proof-based Traceability

The notion of proof-based traceability to evaluate whether a model can infer the correct answer according to the right reasoning path, which yields two metrics: a proof-based accu-

Table 7: The proof-based traceability of language models for updates number U on LogicNMR.

Model	U	Test(Accuracy(%))											
		U=1		U=2		U=3		U=4		U=5		Avg*	
		P-AC	P-EM	P-AC	P-EM	P-AC	P-EM	P-AC	P-EM	P-AC	P-EM	P-AC*	P-EM*
RoBERTa	U=1	**99.1**	**99.0**	57.8	55.6	53.8	50.2	59.7	58.1	59.8	56.1	57.7	55.0
	U=2	98.0	97.9	**99.4**	**99.0**	71.3	68.7	44.6	39.8	38.8	30.7	63.2	59.3
	U=3	99.4	99.2	98.8	98.4	**99.0**	**98.5**	96.0	93.2	87.2	82.7	95.3	93.3
	U=4	97.9	97.5	97.8	97.3	98.2	**97.8**	**97.8**	97.1	95.3	92.2	97.3	96.2
	U=5	54.2	20.1	69.6	36.6	84.7	64.9	92.9	83.1	**96.4**	**92.6**	75.3	51.1
BERT	U=1	**99.5**	**99.3**	59.3	55.3	50.2	41.1	49.7	38.1	48.9	39.3	52.0	42.9
	U=2	94.8	93.8	**98.3**	**97.7**	69.3	68.1	50.6	45.6	49.2	44.1	65.9	62.9
	U=3	96.3	91.7	96.6	93.8	**97.1**	**95.8**	64.0	61.7	44.5	39.4	75.3	71.6
	U=4	85.1	65.3	90.3	78.3	**95.9**	**93.0**	94.6	91.8	62.1	52.1	83.3	72.2
	U=5	40.7	13.5	48.6	15.8	63.3	27.6	77.9	48.5	**91.2**	**83.2**	57.6	26.3
GPT2	U=1	**99.6**	**99.5**	60.9	56.9	73.0	72.9	66.0	65.4	67.7	65.1	66.9	65.0
	U=2	0.47	0.21	**93.3**	**85.1**	82.8	80.9	69.4	65.3	63.2	59.5	53.9	51.5
	U=3	76.8	66.9	92.8	87.8	**98.1**	**96.5**	80.7	75.5	64.8	56.5	78.8	71.7
	U=4	75.7	53.9	87.4	74.1	95.3	92.1	**99.1**	**97.7**	96.3	93.9	88.7	78.5
	U=5	23.6	6.34	41.6	11.9	59.7	24.6	76.0	45.9	**91.3**	**82.7**	50.2	22.2

P-AC* is the average proof-based accuracy on the samples with $U \neq K$ and P-EM* is the average proof-based exact match result on the samples with $U \neq K$.

racy (P-AC) and a proof-based exact match (P-EM) (Yang et al., 2018; Tian et al., 2021). As there are some samples whose query or its negation has no proof, i.e., those queries labelled as and "M", we remove them from the testing samples. Also, since the samples using only one default rule in the inference process have no intermediate proof, such samples will be ignored. P-AC is the ratio of the samples that the model correctly answers the query with proofs on the testing samples, and P-EM is the ratio of the samples hat the model correctly predicts all proofs on the testing samples.

Table 7 shows the results of the language models on the in-domain and out-of-domain datasets, respectively. For the in-distribution samples, the average P-AC of RoBERTa, BERT, and GPT2 are 98.3%, 96.1%, 96.3%, and the average P-EM is 97.3%, 93.5%, 92.3%, respectively. Unsurprisingly, the three language models all achieve an excellent performance in terms of the proof-based traceability on in-distribution samples, as they perform well in terms of the accuracy. On the other hand, for out-of-distribution samples, the average P-AC* of RoBERTa, BERT, and GPT2 models are 77.7%, 66.8%, and 67.6%, and the average P-EM* are 70.9%, 55.2%, and 57.8%, respectively. It shows that the language models perform worse on out-of-distribution samples. The gap of the performances between in-distribution and out-of-distribution samples indicates that the three language models cannot generalize their reasoning ability to out-of-distribution samples, further suggesting that it is suspicious if we say the language models have already mastered the non-monotonic reasoning ability.

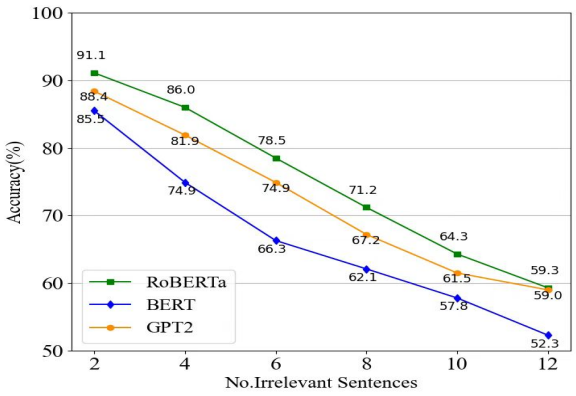

Figure 4: Robustness analysis to irrelevant sentences on $U = 5$.

5.2.4 Robustness

We also evaluate the robustness of the language models to irrelavant sentences. Only one irrelevant fact and one irrelevant default rule are

Update	Knowledge Base	Proofs	Query
U = 3	Amery is not aggressive. Amery is environmental. Amery is not afraid. **Amery is not worried**. If someoneA is fresh then he is environmental, unless he is not aggressive. If someoneA is environmental then he is not humble, unless he is zealous or he is not various. If someoneA is environmental then he is not worried, unless he is not afraid. If someoneA is environmental and not worried then **he is not lucky**, unless he is not tidy. If someoneA is not lucky then **he is grumpy**, unless he is not fine or he is not repulsive. If someoneA is not worried and grumpy then he is sane ,unless he is not unfair. If someoneA is grumpy and not lucky then **he is dynamic**, unless he is not embarrassed. If someoneA is dynamic then **he is cute**, unless he is not petite. If someoneA is dynamic then he is round, unless he is not blue or he is not conscious. If someoneA is cute and dynamic then he is useful, unless he is not ancient or he is not enchanting.	P1: **He is not lucky**. (√) P2: **He is grumpy**. (√) P3: **He is dynamic.** (√)	**Amery is not cute**. (√)
U = 4	Amery is not aggressive. Amery is environmental. Amery is not afraid. Amery is not worried. **Amery is fine.** If someoneA is fresh then he is environmental, unless he is not aggressive. If someoneA is environmental then he is not humble, unless he is zealous or he is not various. If someoneA is environmental then he is not worried, unless he is not afraid. If someoneA is environmental and not worried then **he is not lucky**, unless he is not tidy. If someoneA is not lucky then **he is grumpy**, unless he is not fine or he is not repulsive. If someoneA is not worried and grumpy then he is sane, unless he is not unfair. If someoneA is grumpy and not lucky then **he is dynamic**, unless he is not embarrassed. If someoneA is dynamic then **he is cute**, unless he is not petite. If someoneA is dynamic then he is round, unless he is not blue or he is not conscious. If someoneA is cute and dynamic then he is useful, unless he is not ancient or he is not enchanting.	P1: **He is not lucky**. (√) P2: **He is grumpy**. (×) P3: **He is dynamic**. (√)	**Amery is not cute**. (√)
U = 5	Amery is not aggressive. Amery is environmental. Amery is not afraid. Amery is not worried. Amery is fine. **Amery is grumpy.** If someoneA is fresh then he is environmental, unless he is not aggressive. If someoneA is environmental then he is not humble, unless he is zealous or he is not various. If someoneA is environmental then he is not worried, unless he is not afraid. If someoneA is environmental and not worried then **he is not lucky**, unless he is not tidy. If someoneA is not lucky then he is grumpy, unless he is not fine or he is not repulsive. If someoneA is not worried and grumpy then he is sane, unless he is not unfair. If someoneA is grumpy and not lucky then **he is dynamic** , unless he is not embarrassed. If someoneA is dynamic then **he is cute**, unless he is not petite. If someoneA is dynamic then he is round, unless he is not blue or he is not conscious. If someoneA is cute and dynamic then he is useful, unless he is not ancient or he is not enchanting.	P1: **He is not lucky.** (×) P2: **He is dynamic**. (√)	**Amery is not cute.** (×)

Figure 5: An example about RoBERTa from the LogicNMR benchmark. In "Query" column, ✓ represents RoBERTa answers the query correctly.

added to the knowledge base each time. Figure 4 shows the robustness analysis to irrelevant sentences on the samples with $U = 5$. When the number of irrelevant facts and default rules in the knowledge base increases, the performance of the language models decreases rapidly. The reason for the poor robustness of the models should be that the pattern of the generated samples is simple, and the language models only make predictions by association match. It further suggests that the language models by no means totally master the non-monotonic reasoning ability after finely tuning on a large number of samples about non-monotonic reasoning.

5.3 Case Study

Figure 5 shows an analysis of RoBERTa on some sample. RoBERTa is trained on the dataset with $U = 2$. The bold black sentences in fact represent updated facts currently added to the knowledge base. The solid underlined ones in proofs are the sentences that were predicted correctly by the model, and the dotted underlined ones are the sentences in proofs that were predicted wrong by the model.

In this example, after adding the new fact to the knowledge base for the third time, RoBERTa still predicts all proofs correctly. However, after the fourth update, although the model answers the query correctly, the proof P2 is predicted incorrectly, indicating that the model does not exactly recover the proofs of the query. After the fifth update, the query and its proofs are predicted incorrectly. The above case shows that as the number of updates to the knowledge base increases, the performance of the language model is getting worse. Even the query is answered correctly by the language model, in fact it is not obtained via a correct reasoning procedure by the language model.

6 Conclusions

In this paper, we construct a synthetic non-monotonic reasoning benchmark, LogicNMR, with explicit facts and rules, to capture the iterative update on the knowledge base. We probe whether the pre-trained language models have truly mastered the non-monotonic reasoning ability. The experimental results show that even though the fine-tuned language models all achieve a high accuracy, they perform worse on generalization, proof-based traceability and robustness to irrelevant information. Consequently, we cannot give a positive answer to the

research problem whether the language models master the non-monotonic reasoning ability. It suggests us to explore a better approach to take advantage of the language models to conduct non-monotonic reasoning tasks.

7 Limitations

Although we construct a dataset to probe the non-monotonic reasoning ability of language models and conduct some experiments, we have to admit that there are still some limitations. First, only three language models are used in this paper. More language models with different architectures should be evaluated. Second, the synthetic rules of LogicNMR are too strong. We will relax some restrictions of generating rules, such as query extraction way. Third, we limit the default theory to only one extension to reduce reasoning complexity, resulting in simpler non-monotonic inference patterns. A future work is to probe non-monotonic reasoning ability in a more general and systematic way, such as by allowing plural extensions.

Acknowledgements

This paper is supported by the Natural Science Foundation of China under Grant Nos. 62076261 and 61906216.

References

Krzysztof R. Apt, Howard A. Blair, and Adrian Walker. 1988. Towards a theory of declarative knowledge. In Jack Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann.

Salem Benferhat, Didier Dubois, Hélène Fargier, Henri Prade, and Regis Sabbadin. 2000. Decision, nonmonotonic reasoning and possibilistic logic. In *Logic-based artificial intelligence*, pages 333–358.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Wen-tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642.

Faeze Brahman, Vered Shwartz, Rachel Rudinger, and Yejin Choi. 2021. Learning to rationalize for nonmonotonic reasoning with distant supervision. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12592–12601.

Gerhard Brewka, Thomas Eiter, and Miroslaw Truszczynski. 2011. Answer set programming at a glance. *Commun. ACM*, 54(12):92–103.

Yin Chen, Hai Wan, Yan Zhang, and Yi Zhou. 2010. dl2asp: Implementing default logic via answer set programming. In *Logics in Artificial Intelligence - 12th European Conference, JELIA 2010, Helsinki, Finland, September 13-15, 2010. Proceedings*, volume 6341 of *Lecture Notes in Computer Science*, pages 104–116.

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.

Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. Explaining answers with entailment trees. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 7358–7370.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2726–2736.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? A critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, Brussels, Belgium, October 31 - November 4*, pages 5010–5015.

Sarah Lawsky. 2017. *Nonmonotonic Logic and Rule-Based Legal Reasoning*. Ph.D. thesis, UC Irvine.

Xiao Li, Gong Cheng, Ziheng Chen, Yawei Sun, and Yuzhong Qu. 2022. Adalogn: Adaptive logic graph network for reasoning-based machine reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 7147–7161.

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 *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3622–3628.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

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

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI 2020*, volume 34, pages 8449–8456.

Aman Madaan, Dheeraj Rajagopal, Niket Tandon, Yiming Yang, and Eduard H. Hovy. 2021a. Could you give me a hint ? generating inference graphs for defeasible reasoning. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 5138–5147.

Aman Madaan, Niket Tandon, Dheeraj Rajagopal, Peter Clark, Yiming Yang, and Eduard H. Hovy. 2021b. Think about it! improving defeasible reasoning by first modeling the question scenario. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6291–6310.

John McCarthy. 1980. Circumscription—a form of non-monotonic reasoning. *Artificial intelligence*, 13(1-2):27–39.

Marvin Minsky. 1975. A framework for representing knowledge. reprinted in the psychology of computer vision, p. winston.

Robert C. Moore. 1983. Semantical considerations on nonmonotonic logic. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence. Karlsruhe, FRG, August 1983*, pages 272–279.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Raymond Reiter. 1977. On closed world data bases. In *Logic and Data Bases, Symposium on Logic and Data Bases, Centre d'études et de recherches de Toulouse, France, 1977*, Advances in Data Base Theory, pages 55–76, New York. Plemum Press.

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

Kyle Richardson, Hai Hu, Lawrence Moss, and Ashish Sabharwal. 2020. Probing natural language inference models through semantic fragments. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI 2020*, volume 34, pages 8713–8721.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4661–4675.

Mohammed Saeed, Naser Ahmadi, Preslav Nakov, and Paolo Papotti. 2021. Rulebert: Teaching soft rules to pre-trained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 1460–1476.

Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L. Hamilton. 2019. CLUTRR: A diagnostic benchmark for inductive reasoning from text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, 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 *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing,EMNLP 2021*, pages 3738–3747.

Fangzhi Xu, Jun Liu, Qika Lin, Yudai Pan, and Lingling Zhang. 2022. Logiformer: A two-branch graph transformer network for interpretable logical reasoning. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 1055–1065.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. 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, EMNLP 2018, Brussels, Belgium, October 31 - November 4*, pages 2369–2380.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.