Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text

Siyuan Wang,* Wanjun Zhong, Duyu Tang, Zhongyu Wei, Zhihao Fan, Daxin Jiang,
Ming Zhou, Nan Duan

Motivation

- Large scale pre-trained models for logical reasoning mainly focus on word-level semantics of text while struggling to capture symbolic logic. In this paper, we propose to understand logical symbols and expressions in the text to arrive at the answer.

Dataset

In rheumatoid arthritis, the body' s immune system misfunctions by attacking healthy cells in the joints causing the release of a hormone that in turn causes pain and swelling. This hormone is normally activated only in reaction to injury or infection. A new arthritis medication will contain a protein that inhibits the functioning of the hormone that causes pain and swelling in the joints.

The statements above, if true, most strongly support which one of the following conclusions?

"Unlike aspirin and other medications that reduce pain and swelling and that are currently available, the new medication would repair existing cell damage that had been caused by rheumatoid arthritis."

"A patient treated with the new medication for rheumatoid arthritis could sustain a joint injury without becoming aware of it."

"Joint diseases other than rheumatoid arthritis would not be affected by the new medication."

"The benefits to rheumatoid arthritis sufferers of the new medication would outweigh the medication's possible harmful side effects."

Baseline

- Given four options, four concatenated sequences are constructed to calculate four scores, and the one with the highest score is chosen as the answer. Specifically, the concatenated sequence is formulated as *[CLS]* c *[SEP]* q || o *[SEP]*, where c is the context and q || o is the concatenation of the question and each option. The representations of the special token *[CLS]* in the four sequences are fed into a linear layer with a softmax function to get the probability distribution of options as P ({o1, o2, o3, o4}|c, q). The cross entropy loss is calculated as:

$$\mathcal{L}_A = -\sum \log P(o_a|c, q)$$

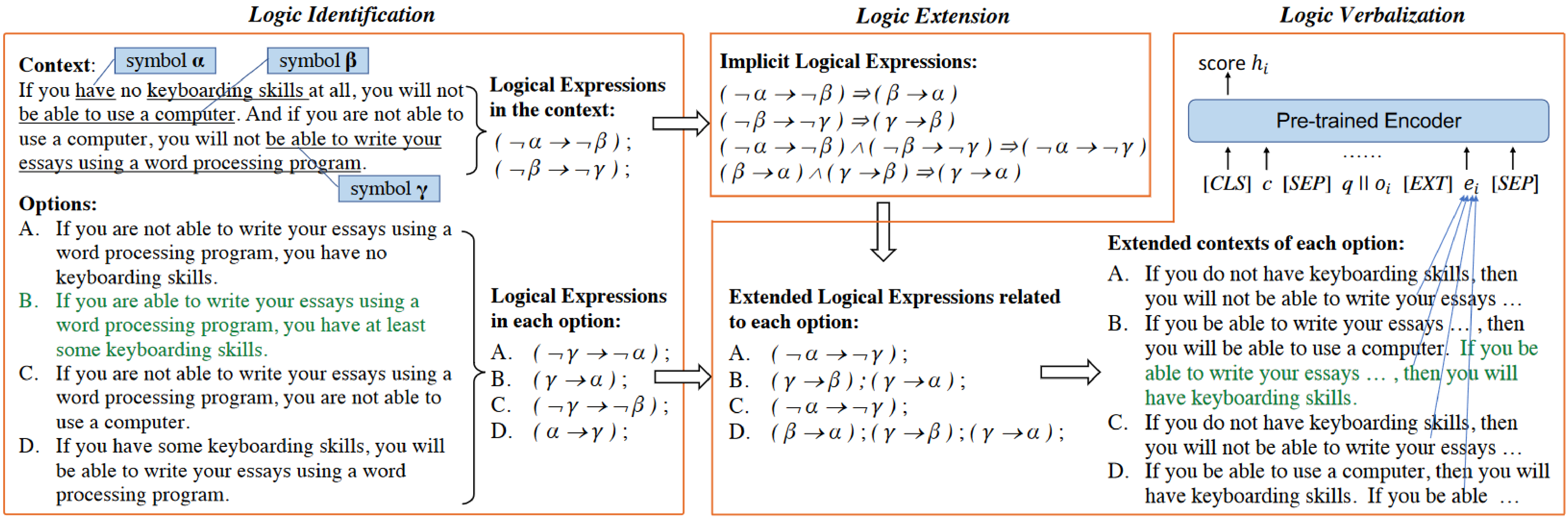
Logic Identification

(1) $\{\alpha, \beta, \gamma, ...\}$: the logical symbols, which are the basic constituents in the context to constitute the logical expressions, such as the "have keyboarding skills" in Figure 2.

(2) $\{\neg, \rightarrow\}$: the logical connectives set. $\neg$ means the negation operation upon a specific logical symbol and $\rightarrow$ acts as a conditional relationship between two logical symbols.

(3) $\{(\alpha \rightarrow \beta), ...\}$: the logical expressions which are composed of logical symbols and connectives. $(\alpha \rightarrow \beta)$ means that α is the condition of β.

Model



Logic Identification
Context:
symbol α
symbol β
If you have no keyboarding skills at all, you will not be able to use a computer. And if you are not able to use a computer, you will not be able to write your essays using a word processing program.
symbol γ
Logical Expressions in the context:
(¬α → ¬β) ;
(¬β → ¬γ) ;
Options:
A. If you are not able to write your essays using a word processing program, you have no keyboarding skills.
B. If you are able to write your essays using a word processing program, you have at least some keyboarding skills.
C. If you are not able to write your essays using a word processing program, you are not able to use a computer.
D. If you have some keyboarding skills, you will be able to write your essays using a word processing program.
Logical Expressions in each option:
A. (¬γ → ¬α) ;
B. (γ → α) ;
C. (¬γ → ¬β) ;
D. (α → γ) ;
Logic Extension
Implicit Logical Expressions:
(¬α → ¬β) ⇒ (β → α)
(¬β → ¬γ) ⇒ (γ → β)
(¬α → ¬β) ∧ (¬β → ¬γ) ⇒ (¬α → ¬γ)
(β → α) ∧ (γ → β) ⇒ (γ → α)
Extended Logical Expressions related to each option:
A. (¬α → ¬γ) ;
B. (γ → β) ; (γ → α) ;
C. (¬α → ¬γ) ;
D. (β → α) ; (γ → β) ; (γ → α) ;
Logic Verbalization
score h_i
Pre-trained Encoder
......
[CLS] c [SEP] q || o_i [EXT] e_i [SEP]
Extended contexts of each option:
A. If you do not have keyboarding skills, then you will not be able to write your essays …
B. If you be able to write your essays … , then you will be able to use a computer. If you be able to write your essays … , then you will have keyboarding skills.
C. If you do not have keyboarding skills, then you will not be able to write your essays …
D. If you be able to use a computer, then you will have keyboarding skills. If you be able …

Model

logic	$(\neg\alpha \rightarrow \neg\gamma)$
template	If do not α, then will not γ.
extended context	If you do not have keyboarding skills, then you will not be able to write your essays using a word processing program.

- Logic Verbalization
  - A logical expression is regarded as related to an option if it has the same logical symbols with the option judged by the text overlapping, and whether a negation connective exists also needs to be considered.
  - Then we transform all logical expressions related to the option at symbolic space into natural language by filling them into a template and concatenate them into a sentence.
  - We feed extended contexts into the pre-trained model to match the options and predict the answer. We reformulate the input sequence as *[CLS]* c *[SEP]* q || o *[EXT]* e *[SEP]* for encoding and feed the *[CLS]* representation into a classification layer to get each option's score and find the most appropriate answer.

Logic-Driven Data Augmentation

- Logic-Driven Contrastive Learning

  In our multiple-choice question answering setting, we alter the score function from measuring the similarity between two representations to calculating the score that the question can be solved by the correct answer under a given context:

  $$s'(c^+, q, o_a) \gg s'(c^-, q, o_a)$$

  The contrastive loss can be formulated as a classification loss for predicting the most plausible context that supports the answer:

  $$\mathcal{L}_C = -\sum \log \frac{\exp(s'(+))}{\exp(s'(+)) + \exp(s'(-))}$$

Logic-Driven Data Augmentation

Aware of symbolic logical expressions, we can construct logical negative samples including negative contexts that are literally similar but logical dissimilar to the positive one.

In the logic-driven data augmentation algorithm, our framework is trained with a combined loss as L = LA + LC.

$(context, question, answer)$

$\downarrow$ ***Logic Identification***

$(\alpha \rightarrow \beta), (\beta \rightarrow \gamma), \ldots$

$\downarrow$ ***Randomly delete, reverse or negate a logical expression***

delete	$(\beta \rightarrow \gamma), \ldots$
reverse	$(\beta \rightarrow \alpha), (\beta \rightarrow \gamma), \ldots$
negate	$(\alpha \rightarrow \neg\beta), (\beta \rightarrow \gamma), \ldots$ $(\neg\alpha \rightarrow \beta), (\beta \rightarrow \gamma), \ldots$

$\downarrow$ ***Logic Verbalization***

$(context^{-}, question, answer)$

Results

Model	Test	EASY	HARD
GPT*	45.4	73.0	23.8
GPT-2*	47.2	73.0	27.0
BERT*	49.8	72.0	32.3
XLNet*	56.0	75.7	40.5
RoBERTa*	55.6	75.5	40.0
$\text{LReasoner}_{\text{RoBERTa}}$	62.4	**81.4**	47.5
ALBERT	66.5	76.6	58.6
$\text{LReasoner}_{\text{ALBERT}}$	**70.7**	81.1	62.5
Human Performance*	63.0	57.1	**67.2**

Model	Val	Test	EASY	HARD
RoBERTa	62.6	55.6	75.5	40.0
+ CE	65.2	58.3	78.6	42.3
+ DA	65.8	61.0	80.9	45.4
+ CE + DA	66.2	62.4	81.4	47.5

Model	Test	EASY	HARD
RoBERTa (w/o CLR)	55.6	75.5	40.0
RoBERTa (w/ CLR-RS)	58.2	79.3	41.6
RoBERTa (w/ CLR-RD)	58.9	78.9	43.2
RoBERTa (w/ CLR-L)	61.0	80.9	45.4

MERIt: Meta-Path Guided Contrastive Learning for Logical Reasoning

Fangkai Jiao† , Yangyang Guo§∗, Xuemeng Song† , Liqiang Nie

Motivation

- Previous studies either employ graph-based models to incorporate prior knowledge about logical relations, or introduce symbolic logic into neural models through data augmentation. These methods, however, heavily depend on annotated training data, and thus suffer from overfitting and poor generalization problems due to the dataset sparsity.
- To address these two problems, in this paper, we propose MERIt, a MEta-path guided contrastive learning method for logical ReasonIng of text, to perform self-supervised pre-training on abundant unlabeled text data

Meta-Path Logic Reasoning

Context: Economist: **(1)** A country's rapid emergence from an economic recession (r_1) requires **(2)** substantial new investment in that country's economy. Since **(3)** people's confidence in the economic policies of their country (r_2) is a precondition for **(2)** any new investment, **(4)** countries that put collective goals before individuals' goals (r_3) cannot **(1)** emerge quickly from an economic recession.

Question:

Which one of the following, if assumed, enables the economist's conclusion to be properly drawn?

Options:

A. People in **(4)** countries that put collective goals before individuals' goals (r_4) lack **(3)** confidence in the economic policies of their countries.

B. A country's economic policies are the most significant factor determining whether that country's economy will experience a recession.

C. If the people in a country that puts individuals' goals first are willing to make new investments in their country's economy, their country will emerge quickly from an economic recession.

D. No new investment occurs in any country that does not emerge quickly from an economic recession.

Answer: A

Logic Structure: (4) $\xrightarrow{r_4}$ (3) $\xrightarrow{r_2}$ (2) $\xrightarrow{\bar{r}_1}$ (1) $\Leftrightarrow$ (4) $\xrightarrow{r_3}$ (1)

As shown in Figure 1, given a context containing a series of logical variables $\{v_1, v_2, \cdots, v_n\}$, and the relations between them, the logical reasoning objective is to judge whether a triplet $\langle v_i, r_{i,j}, v_j \rangle$ in language, where $r_{i,j}$ is the relation between v_i and v_j, can be inferred from the context through a reasoning path:

$$\langle v_i, r_{i,j}, v_j \rangle \leftarrow (v_i \xrightarrow{r_{i,i+1}} v_{i+1} \cdots \xrightarrow{r_{j-1,j}} v_j). \quad (2)$$

The equation is also referred to *symbolic logic rules* (Clark et al., 2020; Liu et al., 2021).

Given an entity-level knowledge graph, where the nodes refer to entities and edges are the relations among them, the meta-path connecting two target entities $\langle e_i, e_j \rangle$ can be given as,

$$e_i \xrightarrow{r_{i,i+1}} e_{i+1} \xrightarrow{r_{i+1,i+2}} \cdots e_{j-1} \xrightarrow{r_{j-1,j}} e_j, \quad (3)$$

where $r_{i,j}$ denotes the relation between entities e_i and e_j. The meta-path in the entity-level knowledge graph are often employed as a particular data structure expressing the relation between two indirectly connected entities (Zeng et al., 2020a; Xu et al., 2021).

From Logical Reasoning to Meta-Path

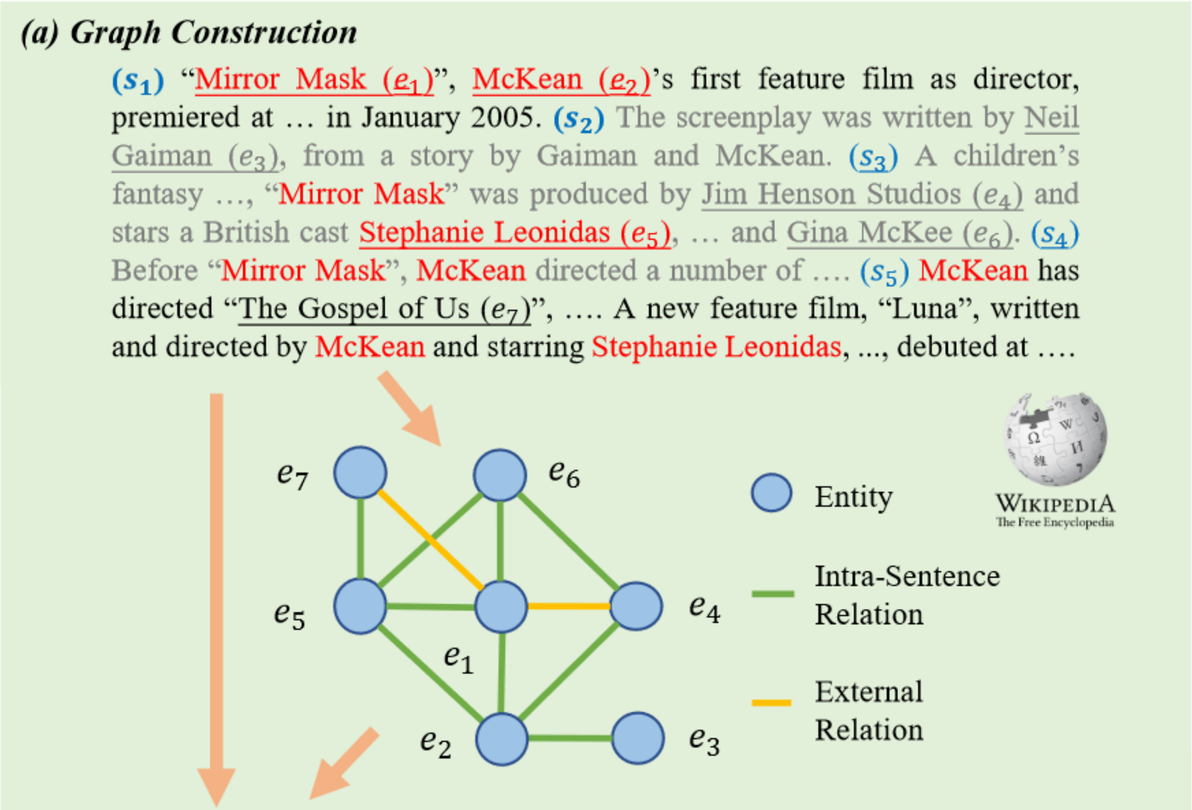
- It is non-trivial to derive such instance pairs from large-scale unlabeled corpus like Wikipedia due to the redundant constituents, e.g., nouns and predicates. In order to address it, we propose to take the entities contained in unlabeled text as logical variables.

$$\langle\, e_i, r_{i,j}, e_j \,\rangle \leftarrow (e_i \xrightarrow{r_{i,i+1}} e_{i+1} \cdots \xrightarrow{r_{j-1,j}} e_j)$$

- In order to aid the logical consistency conditioned on entities to be established, we posit an assumption that *under the same context (in the same passage), the definite relation between a pair of entities can be inferred from the contextual indirect one, or at least not logically contradict to it*.

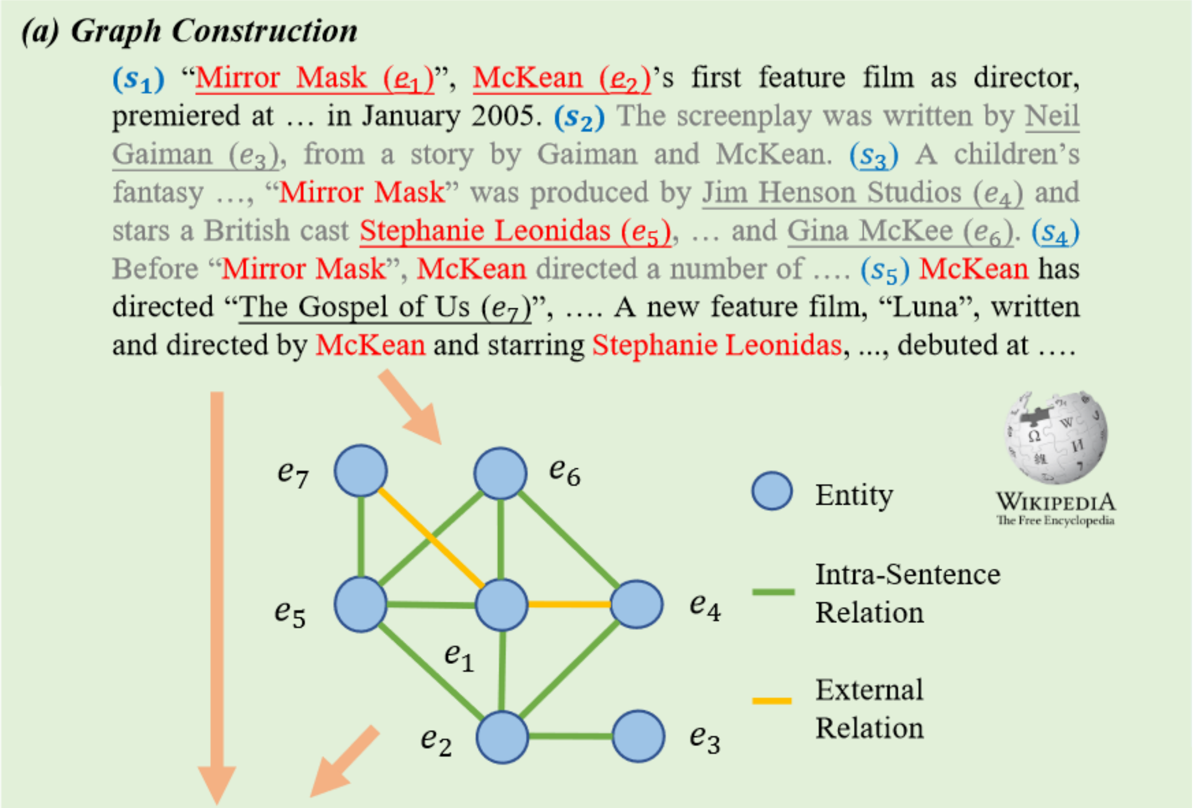
From Logical Reasoning to Meta-Path

- Taking the passage in Figure 2 as an example, it can be concluded from the sentences s1 and s5 that, the director McKean has cooperated with Stephanie Leonidas. Therefore, the logic is consistent between {s1, s5} and s3. This can be viewed as a weaker constraint than the original one in Equation 2 for logical consistency, yet it can be further enhanced by constructing negative candidates violating logics.



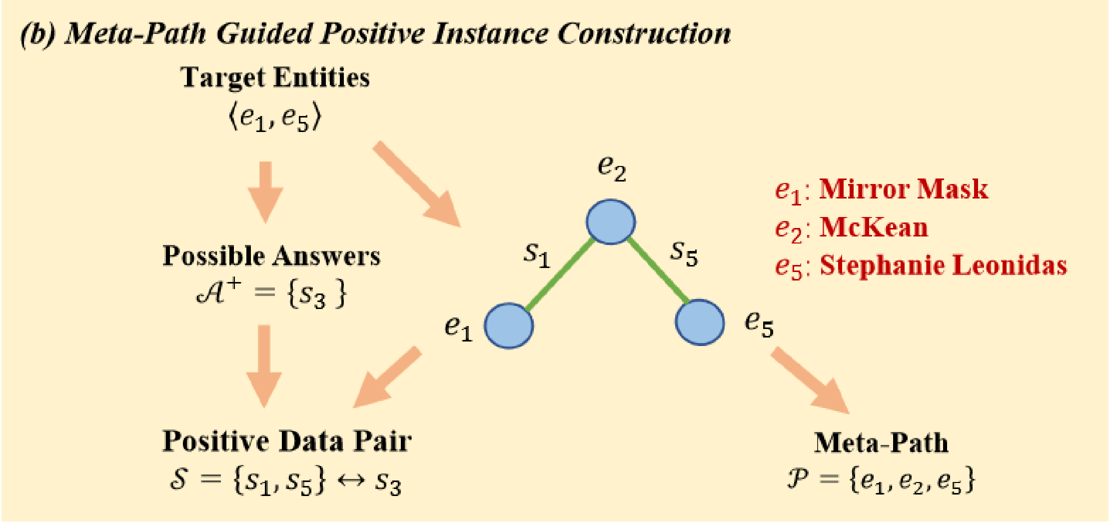
From Logical Reasoning to Meta-Path

Motivated by this, given an arbitrary document $\mathcal{D} = \{s_1, \cdots, s_m\}$, where s_i is the i-th sentence, we can first build an entity-level graph, denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of entities contained in $\mathcal{D}$ and $\mathcal{E}$ denotes the set of relations between entities. Notably, to comprehensively capture the relations among entities, we take into account both the external relation from the knowledge graph and the intra-sentence relation. As illustrated in Figure 2 (a), there will be an intra-sentence relation between two entities if they are mentioned in a common sentence. Thereafter, we can derive the pre-training instance pairs according to the meta-paths extracted from the graph, which will be detailed in the following subsections.



Meta-Path Guided Positive Instance Construction

In particular, as shown in Figure 2 (b), given an entity pair $\langle e_i, e_j \rangle$, we denote the collected answer candidates as $\mathcal{A}^+$, and then we use Depth-First Search (Tarjan, 1972) to find a meta-path linking them on $\mathcal{G}$, following Equation 3. Thereafter, the context sentences $\mathcal{S}$ corresponding to the answer candidates in $\mathcal{A}^+$ are derived by retrieving those sentences undertaking the intra-sentence relations during the search algorithm. Finally, for each answer candidate $a \in \mathcal{A}^+$, the pair $(\mathcal{S}, a)$ is treated as a positive context-answer pair to facilitate our contrastive learning. The details of positive instance generation algorithm are described in Appendix A.

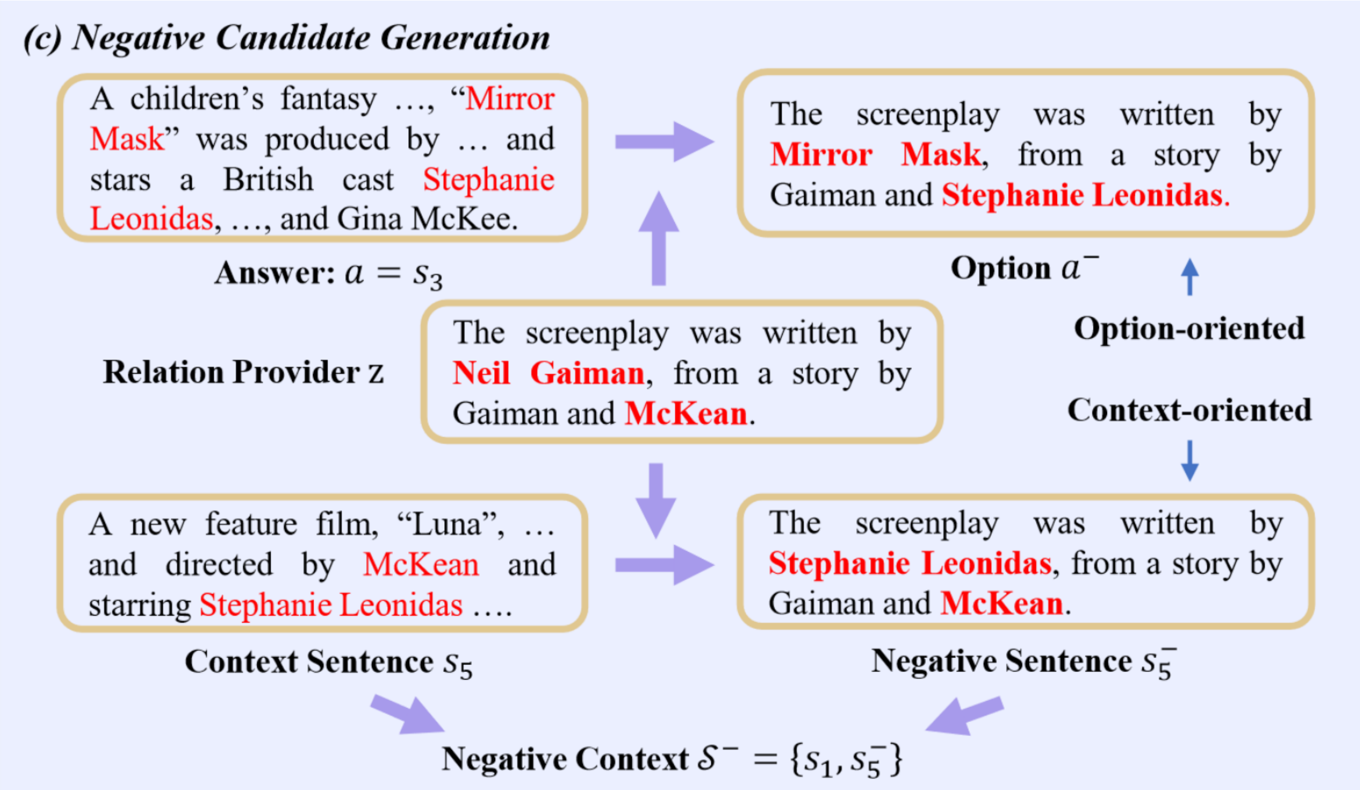


Negative Instance Generation

- In order to obtain the negative instances (i.e., negative context-option pairs) where the option is not logically consistent with the context, the most straightforward way is to randomly sample the sentences from different documents. However, this approach could lead to trivial solutions by simply checking whether the entities involved in each option are the same as those in the given context.
- In the light of this, we resort to directly breaking the logical consistency of the positive instance pair by modifying the relation rather than the entities in the context or the option, to derive the negative instance pair.

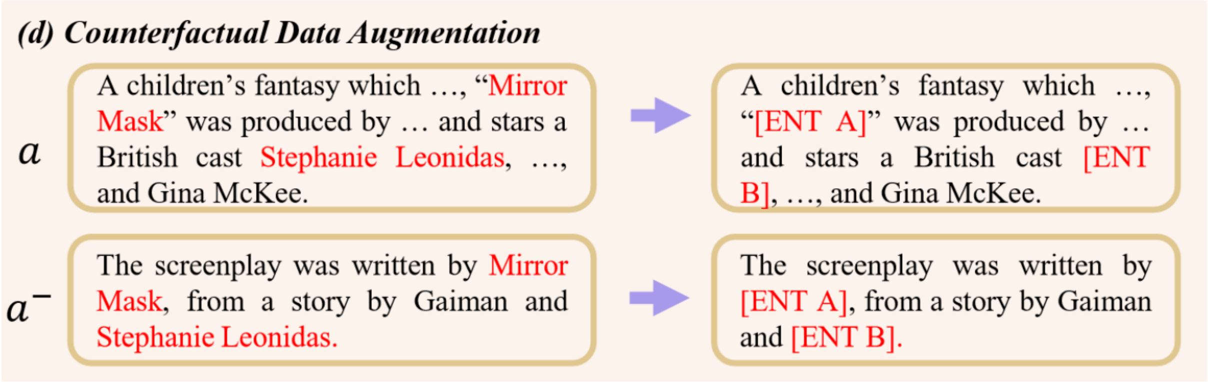
Negative Instance Generation

- We devise two negative instance generation methods: the **context-oriented** and the **option-oriented** method, focusing on generating negative pairs by modifying the relations involved in the context S and answer a of the positive pair, respectively.



Counterfactual Data Augmentation

- In particular, since the correct answer is from a natural sentence and describes a real world fact, while the negative option is synthesized by entity replacement, which may conflict with the commonsense knowledge. As a result, the pretrained language model tends to identify the correct option directly by judging its factuality rather than the logical consistency with the given context.



Contrastive Learning based Pre-training

- As discussed in previous subsection, there are two contrastive learning schemes: option-oriented CL and context-oriented CL.
- Let $\mathcal{A}^-$ be the set of all constructed negative options with respect to the correct option *a*. The option-oriented CL can be formulated as:

$$\mathcal{L}_{\mathrm{OCL}} = L(\mathcal{S}, a, \mathcal{A}^-).$$

- In addition, given $\mathcal{C}^-$ as the set of all generated negative contexts corresponding to *S*, the objective of context-oriented CL can be written as:

$$\mathcal{L}_{\mathrm{CCL}} = L(a, \mathcal{S}, \mathcal{C}^-).$$

- To avoid the catastrophic forgetting problem, we also add the MLM objective during pre-training and the final loss is:

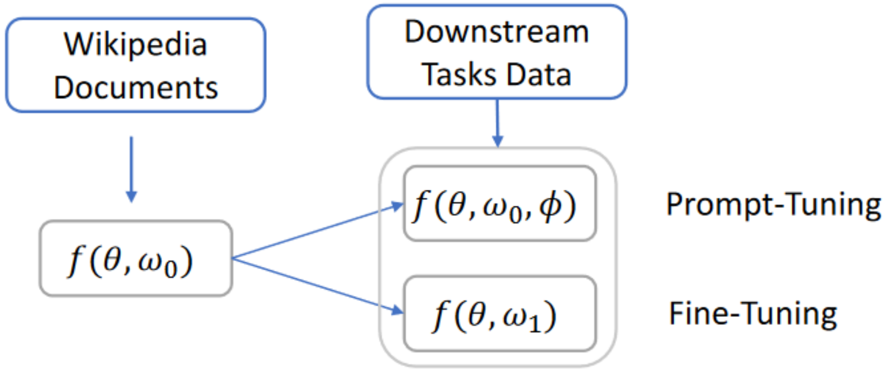
$$\mathcal{L} = \mathcal{L}_{\mathrm{OCL}} + \mathcal{L}_{\mathrm{CCL}} + \mathcal{L}_{\mathrm{MLM}}.$$

Fine-tuning

- During the fine-tuning stage, to approach the task of MCQA, we adopt the following loss function:

$$\mathcal{L}_{\text{QA}} = -\log \frac{\exp f(P, Q, O_y)}{\sum_i \exp f(P, Q, O_i)},$$

- As for the fine-tuning stage, we employ two schemes including simple fine-tuning and prompt-tuning.

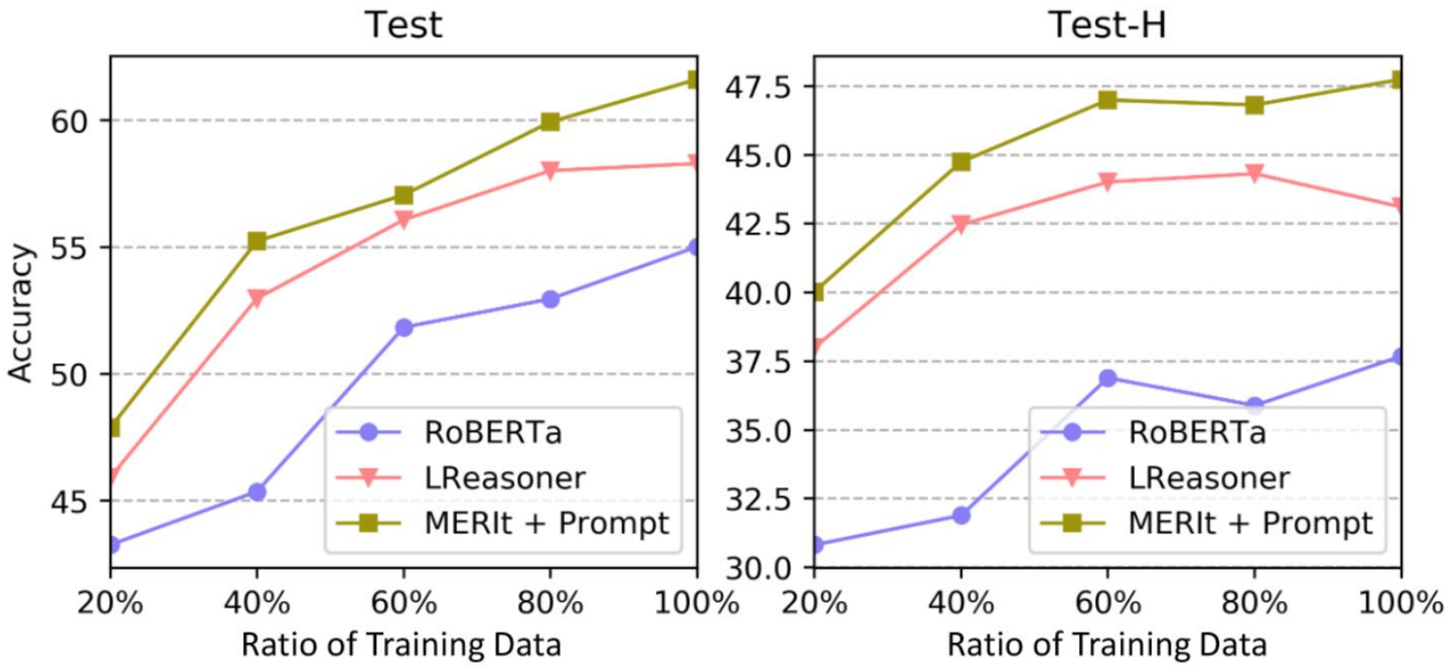


Experiments

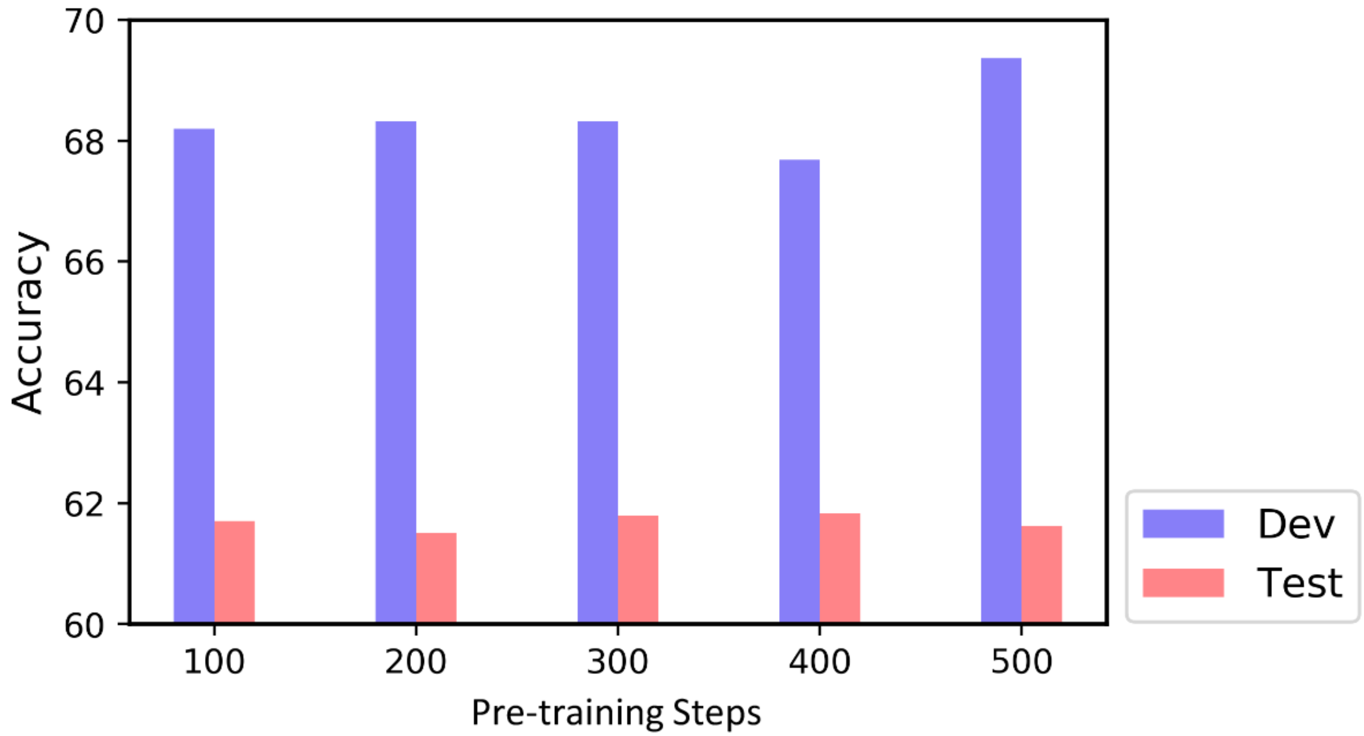
Model / Dataset	ReClor				LogiQA	
	Dev	Test	Test-E	Test-H	Dev	Test
RoBERTa	62.6	55.6	75.5	40.0	35.0	35.3
DAGN	65.2	58.2	76.1	44.1	35.5	38.7
DAGN (Aug)	65.8	58.3	75.9	44.5	36.9	39.3
LReasoner (RoBERTa)‡	64.7	58.3	77.6	43.1	—	—
Focal Reasoner	66.8	58.9	77.1	44.6	**41.0**	40.3
MERIt	66.8	59.6	78.1	45.2	40.0	38.9
MERIt + LReasoner	67.4	60.4	78.5	46.2	—	—
MERIt + Prompt	**69.4**	**61.6**	79.3	**47.8**	39.9	**40.7**
MERIt + Prompt + LReasoner	67.3	61.4	**79.8**	46.9	—	—
ALBERT	69.1	66.5	76.7	58.4	38.9	37.6
MERIt (ALBERT)	74.2	70.1	81.6	61.0	43.7	**42.5**
MERIt (ALBERT) + Prompt	**74.7**	**70.5**	**82.5**	**61.1**	**46.1**	41.7
max						
LReasoner (RoBERTa)	66.2	62.4	81.4	47.5	38.1	40.6
MERIt	67.8	60.7	79.6	45.9	**42.4**	41.5
MERIt + Prompt	**70.2**	**62.6**	80.5	**48.5**	39.5	**42.4**
LReasoner (ALBERT)	73.2	70.7	81.1	62.5	41.6	41.2
MERIt (ALBERT)	73.2	71.1	83.6	61.3	43.9	**45.3**
MERIt (ALBERT) + Prompt	**75.0**	**72.2**	**82.5**	**64.1**	**45.8**	43.8

Model	Dev	Test	Test-E	Test-H
DeBERTa-v2-xlarge	76.7	71.0	83.8	60.9
MERIt (DeBERTa-v2-xlarge)	**78.0**	**73.1**	**86.2**	**64.4**
DeBERTa-v2-xxlarge	78.3	75.3	84.0	68.4
MERIt (DeBERTa-v2-xxlarge)	**80.6**	**78.1**	**84.6**	**72.9**

Experiments



Experiments



Ablation Studies

Model	Dev	Dev (P.)	Test	Test (P.)
MERIt	66.8	69.4	59.6	61.6
- DA	63.0	64.5	57.9	59.8
+ DA^2	65.3	67.8	**60.2**	61.3
+ DA^3	66.2	68.0	59.3	**61.9**
- Option-oriented CL	63.8	65.4	58.9	61.5
- Context-oriented CL	64.0	66.5	58.8	60.2
- Meta-Path	64.8	65.1	58.0	60.8