Self-Consistent Narrative Prompts on Abductive Natural Language Inference

Chunkit Chan[1], Xin Liu[1], Tszho Chan[1], Jiayang Cheng[1], Yangqiu Song[1], Ginny Y. Wong[2], Simon See[2]
[1]Department of Computer Science and Engineering, HKUST, Hong Kong SAR, China
[2]NVIDIA AI Technology Center (NVAITC), NVIDIA, Santa Clara, USA
{ckchancc, xliucr, zchencj, jchengaj, yqsong}@cse.ust.hk
{gwong, ssee}@nvidia.com

Abstract

Abduction has long been seen as crucial for narrative comprehension and reasoning about everyday situations. The abductive natural language inference (αNLI) task has been proposed, and this narrative text-based task aims to infer the most plausible hypothesis from the candidates given two observations. However, the inter-sentential coherence and the model consistency have not been well exploited in the previous works on this task. In this work, we propose a prompt tuning model α-**PACE**[1], which takes self-consistency and inter-sentential coherence into consideration. Besides, we propose a general self-consistent framework that considers various narrative sequences (e.g., linear narrative and reverse chronology) for guiding the pre-trained language model in understanding the narrative context of input. We conduct extensive experiments and thorough ablation studies to illustrate the necessity and effectiveness of α-**PACE**. The performance of our method shows significant improvement against extensive competitive baselines.

1 Introduction

Abductive reasoning aims to find the most plausible explanation based on incomplete observations (Peirce, 1974). Abduction has long been seen to be essential for understanding narratives (Hobbs et al., 1993) and reasoning about everyday situations (Andersen, 1973). Bhagavatula et al. (2020) investigated the language-based abduction in narrative texts and introduced the abductive natural language inference (αNLI) benchmark, which is a multiple-choice question answering task for identifying the most likely explanation among two hypotheses based on two observations. One example is illustrated in Figure 1, where "O_1" and "O_2" are two observations. Abductive reasoning is identifying a possible hypothesis (either H_1 or H_2) that can best explain the consequences by evaluating and comparing the plausibility of these two hypotheses.

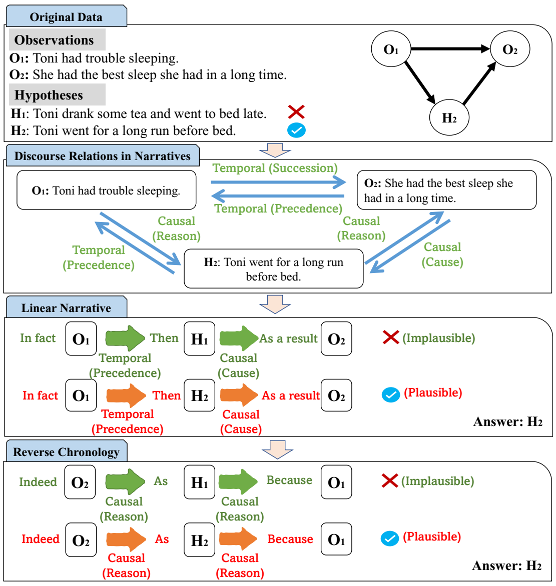

Figure 1: A data example from αNLI and its corresponding narrative sequences, including linear narrative and reverse chronology. Two sequences explain the same narrative example seamlessly by utilizing the discourse connectives (i.e., " in fact," " then," " as a result,").

Traditional works on the αNLI task focus on ranking the hypotheses among "H_1" and "H_2" (Zhu et al., 2020; Li et al., 2021a) or incorporating the knowledge from various sources into pre-trained language models, such as general commonsense knowledge (Mitra et al., 2019; Du et al., 2021) and social commonsense knowledge (Paul et al., 2020). However, one crucial piece of information, i.e., the inter-sentential coherence and the consistency of the model, has yet to be investigated and explored.

These prior studies often concatenate the observations and hypotheses as the model input, ignoring the coherence between sentences and their inter-sentential relations in this narrative-based task. Nar-

[1]The source code is available at https://github.com/HKUST-KnowComp/Alpha-PACE

Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1040–1057
November 1–4, 2023. ©2023 Association for Computational Linguistics

rative, as a semiotic representation of a sequence of events meaningfully connected in a temporal and causal way (Ryan et al., 2007; Onega and Landa, 2014), intrinsically encodes the information required for abductive reasoning that makes it logical, sensible, and coherent. For instance, Figure 1 illustrates that the relation connected from "H_2" to "O_2" is a causal relation emphasized by a discourse connective "as a result" and provides the extra causal information needed for pre-trained language models (PLMs) to comprehend these observations and hypotheses in depth. Furthermore, the consistency of a model, a highly desirable characteristic for a model in natural language processing, refers to the invariant in behavior despite meaning-preserving alterations in its input. Prior research has highlighted the significance of mode consistency and revealed that language models could exhibit inconsistencies in various contexts, including conversation, explanation generation, and factual knowledge extraction (Adiwardana et al., 2020; Camburu et al., 2020; Elazar et al., 2021). These inconsistencies may result in output variability and local optimality. Since the prompt tuning-based method reduces the model variability by freezing the pre-trained model without altering the representations, we propose a self-consistent prompt tuning model that considers the inter-sentential coherence in the αNLI task.

We have noticed that Wang et al. (2022) proposed a self-consistent framework (i.e., sample-and-marginalize method) that focuses on the answer consistency among diverse reasoning paths. It relies on an individual prompt to sample various outputs and perform majority voting to address the inconsistency issue that language models suffer. However, this method may not be an optimal method for the αNLI task as it does not take the narrative sequences into account. A narrative usually describes the sequence of events in various narrative orders, utilizing different inter-sentential relations. In particular, people can understand the same narrative context by utilizing alternative narrative sequences instead of *linear narrative*, such as *nonlinear narrative* and *reverse chronology*. For example, Figure 1 shows that both linear narrative and reverse chronology can explain the same narrative seamlessly by employing discourse connectives. In this linguistic phenomenon, two narrative sequences with different description orders emphasize different partial information about these events, while expressing the same narrative context. When applying machine learning for abductive reasoning, with context sequences being different, the performance of models can vary as pre-trained language models interpret the context information from diverse perspectives and extents.

In this paper, we attempt to imitate the cognitive process of narrative understanding, and propose a general self-consistent framework to facilitate a PLM understanding of the narrative context based on the above linguistic phenomenon considering different narrative sequences. For each narrative sequence, we design a specific prompt template to distinguish the difference in narrative order while still incorporating inter-sentential coherence and self-consistency.

Our contributions are summarized as follows:

1. This work is the first to consider inter-sentential coherence and self-consistency through the prompt tuning method in the task.
2. We propose a general self-consistent framework based on the linguistic phenomenon that allows various narrative sequences for undertaking abductive reasoning.
3. We conduct extensive experiments and thorough ablation studies to illustrate the necessity and effectiveness of the specific prompt template and general self-consistent framework. The results support our claims and the success of our proposed model.

2 Related Work

Abductive Reasoning Abduction has long been thought necessary for comprehending narrative (Hobbs et al., 1993) and reasoning about everyday events (Andersen, 1973). Most earlier research has concentrated on formal logic-based abductive reasoning (Levesque, 1989; Ng and Mooney, 1990; Paul, 1993). However, the rigidity of formal logic restricts its application in the field of NLP. Hence, Bhagavatula et al. (2020) developed a language-based abductive reasoning task to help with this, and they developed baselines that adopt the pre-trained language models (i.e., BERT (Devlin et al., 2019)) under their probabilistic framework. To solve this task, Paul et al. (2020) proposed a multi-head knowledge attention approach to enhance RoBERTa (Liu et al., 2019) by incorporating the structured social commonsense knowledge generated from COMET (Bosselut et al., 2019). Du et al. (2021) employed a latent variable to acquire commonsense knowledge from the event graph and

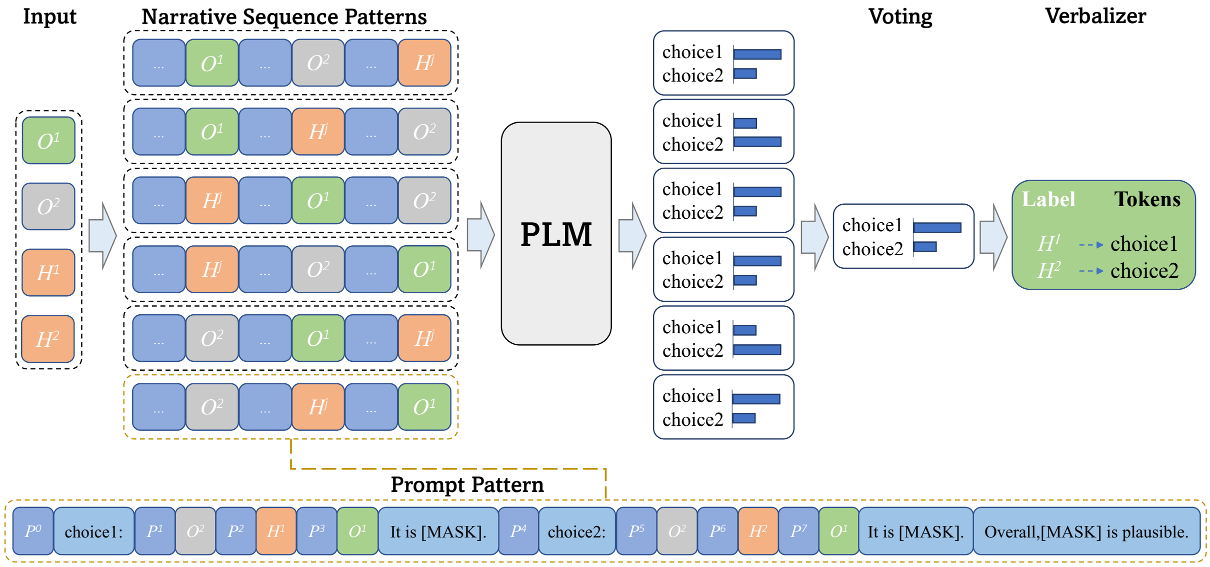
Input
Narrative Sequence Patterns
PLM
Voting
Verbalizer
choice1
choice2
Label
Tokens
H1 --> choice1
H2 --> choice2
Prompt Pattern
P0 choice1: P1 O2 P2 H1 P3 O1 It is [MASK]. P4 choice2: P5 O2 P6 H2 P7 O1 It is [MASK]. Overall,[MASK] is plausible.

Figure 2: The general self-consistent narrative prompt framework for considering varying narrative sequences. Two observations (O^1, O^2) and a pair of hypotheses (H^1, H^2) are permuted as six different sequence patterns, where the corresponding task-specific self-consistent prompt pattern includes two prefix prompts P^0, P^4, six cloze prompts $P^1, P^2, P^3, P^5, P^6, P^7$, and the manual template "It is [MASK]." and "Overall, [MASK] is plausible." The majority voting results align to label predictions finally.

enhance the pre-trained language model RoBERTa. Apart from incorporating commonsense knowledge to tackle this task, Zhu et al. (2020) reformulated the αNLI task as a ranking task using a learning-to-ranking framework to rank candidate hypotheses. Li et al. (2021a) proposed an interactive language model that groups the correct and incorrect hypotheses instead of ranking these hypotheses and adopts joint softmax focal loss for this αNLI task. However, prior works did not exploit model consistency and various narrative sequences in this task.

Prompt Tuning By relaxing the constraint that prompts token embedding to be the natural language, Li and Liang (2021) and Hambardzumyan et al. (2021) proposed combining a PLM's input token embeddings with additional continuous vectors. Some studies (Lester et al., 2021; Qin and Eisner, 2021; Li and Liang, 2021) proposed only tuning continuous prompts, while some works (Han et al., 2021; Zhong et al., 2021; Liu et al., 2021b; Chan et al., 2023b) explore combining discrete prompts and continuous prompts. They tune the embedding of these additional continuous vectors, and the parameters of PLMs are frozen in their task. In our work, we also adopt this strategy, but we focus on utilizing this approach to investigate the model consistency and the narrative coherence information underlying various observations and hypotheses in this task.

3 α-PACE

In order to explore the inter-sentential coherence and model consistency for the abductive natural language inference (αNLI) task, we propose the **a**bductive self-consistent **P**rompt tuning model on n**A**tural language inferen**CE** task (α-**PACE**).

3.1 Problem Definition

Abduction is to infer the most reasonable explanation for incomplete observations (Peirce, 1974). In the αNLI task (Bhagavatula et al., 2020), given two observations O_i^1 and O_i^2, we choose the most plausible hypothesis among H_i^1 and H_i^2:

$$H_i^* = \arg\max_{H_i^j} P\left(H_i = H_i^j \mid O_i^1, O_i^2\right), \quad (1)$$

where H_i^* is the most reasonable hypothesis, and i indicates i-th instance of the dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ where $x_i = \{O_i^1, O_i^2, H_i^1, H_i^2\}$. We will omit the index i without causing ambiguity in the following part.

3.2 T5 Foundation Model

T5 (Raffel et al., 2020), an encoder-decoder model, has been pre-trained on a multi-task mixture of unsupervised and supervised tasks. The unsupervised denoising training task focused on training this model to predict consecutive masked spans of tokens. For instance, the input “She had the best sleep she had in a long time.” was corrupted as “She <X> the best sleep she had in a <Y>.” The target output was “<X> had <Y> long time </s>” </s> is the eos_token. The supervised pre-trained task required the model to perform a sequence-to-sequence input-output mapping with the instruction of a task prefix (e.g., “translate German to English:” or “summarize:”). However, discovering the specific textual prefix token was arduous and required enormous human effort. To overcome this issue, prefix tuning (Li and Liang, 2021) and prompt tuning (Lester et al., 2021) methods were proposed, which relaxed the constraint of discrete textual tokens to continuous and tunable ones.

3.3 Self-Consistent Prompt Tuning Model

To predict the hypothesis H_i^* for each instance x_i, we employed a human-tailored template $\mathcal{T}(\cdot)$ transforming the data instances x_i to the prompt input $\tilde{x}_i = \mathcal{T}(x_i)$, and a verbalizer $\mathrm{V}(\cdot)$ is utilized to map a set of words to class labels. Figure 2 illustrates the architecture of α-**PACE**.

3.3.1 Task-Specific Self-Consistent Prompt

The meticulously devised template, crafted to investigate inter-sentential coherence and self-consistency, comprises essential discrete tokens, masked tokens, and learnable continuous tokens.

Inter-Sentential Coherence For the inter-sentential coherence, we concatenate the O_i^1, H_i^1, O_i^2 and O_i^1, H_i^2, O_i^2 as two sentence sequences S_i^1 and S_i^2, instead of directly connect the O_i^1, O_i^2, H_i^1, and H_i^2 together as the model input. These two sequences are to help PLMs easily capture the coherence information inherently in various sequences. Moreover, Chatman (1980) explains that the story is the context of narrative (the what of the narrative), and the discourse is the form of narrative (the how). Specifically, the discourse is the means by which the narrative content is expressed (Chatman, 1980; Tomaščíková, 2009). Therefore, by adding discourse connectives between two events, the pre-trained language model can understand the narrative context more easily and enhance the inter-sentential coherence. Nevertheless, employing diverse discourse connectives for each data instance presents a formidable challenge. Hence, we insert the continuous tunable prompt tokens to represent the discourse connectives between each sentence (i.e.,O_i^1, H_i^1, O_i^2 and O_i^1) to learn the coherence information between these sentences. Since some discourse connectives naturally start before the first sentence (such as “since” and “although”), we assign the continuous tunable prompt tokens before the first sentence of the sentence sequence. We follow Liu et al. (2021a) to name the continuous prompts in our method. The continuous prompts are denoted as $\{\mathrm{P}^k \in \mathbb{R}^{p^k \times d} | k = 0, 1, \cdots, 7\}$, where the P^0 and P^4 serve as the prefix prompt to learn the instruction guiding the model to perform the αNLI task by following Lester et al. (2021). Other prompt tokens correspond to the cloze prompt between two different sentences (or before the first sentence) utilized to represent the discourse connectives to learn the coherence information between sentences. p^k is the length of the k-th prompt.

Class Label	First [MASK]	Second [MASK]	Third [MASK]
H^1	plausible	not plausible	choice1
H^2	not plausible	plausible	choice2

Table 1: The label word set on αNLI task.

Self-Consistent Prompt For the self-consistency of model output, three [MASK] tokens are included: a [MASK] combined with the discrete token “is plausible.” forming the manual template “Overall, [MASK] is plausible.” for facilitating model inference, and each of another two [MASK] merges with the discrete token “, it is” to constitute “, it is [MASK]” append after each sentence sequence. Furthermore, the discrete tokens “choice1:” and “choice2:” are placed before sequences S_i^1 and S_i^2 respectively for splitting two sequences. These three [MASK] tokens are used for achieving the purpose of model self-consistency by ensuring three model outputs consistently. By considering the mentioned sentence sequences and the example in Figure 1, humans are able to recognize that H_i^2 is more plausible, resulting from the sentence sequence S_i^1 is not plausible or less plausible than S_i^2. In this case, the pre-trained language model guided to predict “not plausible” for S_i^1, “plausible” for S_i^2, and “choice2” in the third mask in the learning process. Throughout this learning process, the model will learn the output consistency ability. Therefore, we introduce three masks in our model to predict the plausibility of S_i^1 and S_i^2, and the last mask for final determined labels (i.e., “choice1” or “choice2” representing the H_i^1 and H_i^2).

Self-Consistent Verbalizer A typical verbalizer usually maps a label y to a single answer token z or a series of spans $z^1, z^2, \cdots$ greedily (Schick and Schütze, 2021; Liu et al., 2021a). We extend it by mapping two class labels (i.e., H_i^1 and H_i^2) to three tokens, i.e. $\{\mathcal{H}^j\} \rightarrow \mathcal{Z} \times \mathcal{Z} \times \mathcal{Z}$, where $\mathcal{Z}$ is the vocabulary and three [MASK] tokens denoted as z^1, z^2, and z^3. Using the tailored prompt template featuring three [MASK]s and the verbalizer, the probability distribution over $\{\mathcal{H}^j\}$ can be formalized as the joint probabilities of z^1, z^2, and z^3, i.e. $\Pr(\mathcal{H}^j \mid \tilde{x}_i) = \Pr(\mathcal{V}(\mathcal{H}^j) \mid \tilde{x}_i) = \Pr(z_i^1 = h_3^j, z_i^2 = h_1^j, z_i^3 = h_2^j \mid \tilde{x}_i)$, where a hypothesis $\mathcal{H}^j$ consists of h_1^j (the plausibility of S_i^1), h_2^j (the plausibility of S_i^2), and h_3^j (the probability of H_i^1 and H_i^2). Table 1 summarizes the label words. Given that T5 is able to predict masked tokens synchronously, the joint probability can be written as

$$\Pr(\mathcal{H}^j \mid \tilde{x}_i) = \prod_{k=1}^{3} \Pr(z_i^k = v^k(\mathcal{H}^j) \mid \tilde{x}_i), \tag{2}$$

where $v^k(\cdot) : \{\mathcal{H}^j\} \rightarrow \mathcal{Z}$ is the submap of $\mathcal{V}(\cdot)$ for the k-th [MASK]. And then the final learning objective of α-PACE is to maximize

$$\mathcal{J} = \frac{1}{|\mathcal{D}|} \sum_{(x_i,y_i)\in\mathcal{D}} \log \sum_{k=1}^{3} \Pr(z_i^k = v^k(\mathcal{H}^j) \mid \tilde{x}_i). \tag{3}$$

The final prediction of $\mathcal{H}_i^*$ by choosing the maximum joint probability (i.e., self-consistency score) as Eq. (2).

3.4 General Self-Consistent Narrative framework

The self-consistent framework introduced by previous studies may not be the optimal approach for the αNLI task due to its lack of consideration for various narrative sequences (Wang et al., 2022). Furthermore, Bhagavatula et al. (2020) and their follow-up works (Du et al., 2021; Paul et al., 2020) only form probabilistic models focusing on the **Linear Chain Model** ($Pr(O^2|H^j)P(H^j|O^1)$, where H^j can be H^1 or H^2) and **Fully Connected Model** ($Pr(O^2|H^j, O^1)P(H^j|O^1)$). This means their framework primarily considers the given fixed time sequence, i.e., O^1, H^j, and O^2, and may not align with the representation of the pre-trained language model. Therefore, we permute $\{O^1, O^2, (H^1, H^2)\}$ and design six narrative sequence patterns for this task according to the orders of observations and the pair of hypotheses. The six patterns are illustrated in the overall framework in Figure 2. For example, the O2HO1 sequence pattern means that we put H^j in the middle of O^2 and O^1 and try to utilize the possible inter-sentential coherence information among them in this order. After receiving the joint generation probabilities from each pattern, we normalize the probability distribution between "H^1" and "H^2 " to make it more contrastive. Then, we perform the majority voting over six narrative sequence pattern distributions and map token predictions to the label prediction.

Train	Dev	Test	Leaderboard
169,654	1,532	3,059	3,040

Table 2: Statistics of ART and αNLI leaderboard data.

4 Experimental Setup

4.1 αNLI Dataset

The experiments are conducted on the ART dataset, aimed at assessing the performance of our model on the αNLI task (Bhagavatula et al., 2020). The observations of ART data were collected from a story corpus known as ROCstory (Mostafazadeh et al., 2016), while the corresponding hypotheses were generated by crowdsourcing. Moreover, αNLI has a dedicated leaderboard with 3,040 test instances to measure the generalizability of the models. The detailed data statistics can be found in Table 2. By following previous work (Bhagavatula et al., 2020; Du et al., 2021; Zhu et al., 2020), we employ accuracy as an evaluation metric to evaluate the empirical performance of our method in experiments and ablation studies.

4.2 Baselines

The implementation detail of α-PACE is displayed in Appendix A.2, and we compare α-PACE with two categories of competitive baselines. The first category is the previous state-of-the-art (SOTA) baselines, such as *BERT* (Devlin et al., 2019), *RoBERTa* (Liu et al., 2019), *DeBERTa* (He et al., 2021), *McQueen* (Mitra et al., 2019), *MHKA* (Paul et al., 2020), *ege-RoBERTa-large* (Du et al., 2021), *L2R*^{2} (Zhu et al., 2020), *IMSL* (Li et al., 2021a), *UNIMO* (Li et al., 2021b), and *UNICORN (T5)* (Lourie et al., 2021). Another category of baselines is T5-based models, and we include the fine-tuned T5 model to illustrate the performance gain of our model. Furthermore, the prompt-based

Model	Dev (%)	Test (%)
Random	-	50.40
BERT (Devlin et al., 2019)	69.10	68.90
RoBERTa (Liu et al., 2019)	85.76	84.48
McQueen (Mitra et al., 2019)	86.68	-
MHKA (Paul et al., 2020)	87.44	87.12
ege-RoBERTa (Du et al., 2021)	-	87.50
$L2R^2$ (Zhu et al., 2020)	88.44	86.11
RoBERTa-L+IMSL (Li et al., 2021a)	89.20	-
Prefix-Tuning (T5) (Lester et al., 2021)	84.20	83.88
Prompt-Tuning (T5) (Li and Liang, 2021)	86.23	85.98
Fine-Tuning (T5) (Raffel et al., 2020)	87.07	87.68
Fine-Tuning$_{\text{General Consistency}}$(T5)	87.54	88.05
α-PACE$_{\text{O1HO2 \& w/o Consistency}}$(T5)	90.60	89.93
α-PACE$_{\text{HO1O2 \& Task Consistency}}$(T5)	92.43	91.83
α-PACE$_{\text{General \& Task Consistency}}$(T5)	**93.15**	**92.54**
Human Performance	-	91.40

Table 3: The accuracy (%) is evaluated on the αNLI task. The approximation of learnable parameters for all models is displayed in Table 10 in Appendix A.5. Fine-Tuning$_{\text{General Consistency}}$(T5) means the fine-tuned T5 model with our general narrative self-consistent framework, which considers six narrative sequences. α-PACE$_{\text{O1HO2 \& w/o Consistency}}$ means this model with a single mask to predict the "choice1" or "choice2" and the O1HO2 pattern is the best model among all patterns.

methods such as *Prefix-Tuning (T5)* (Li and Liang, 2021) and *Prompt-Tuning (T5)* (Lester et al., 2021) are included as baselines for exhibiting the exact contribution of our proposed method. The implementation details of the T5 fine-tuning model are described in Appendix A.3, while Prefix-Tuning and Prompt-Tuning methods are appended in Appendix A.4. More details of the baselines are in Appendix A.1.

5 Experimental Results

5.1 Main Results

Table 3 and Table 4 report the main experimental results on the αNLI task, from which we derive the following conclusions. **First**, our model significantly outperforms all competitive baselines on the αNLI task. Specifically, our method (i.e., α-PACE$_{\text{HO1O2}}$) achieved a considerable improvement of 5.36% on the development set, 4.15% on the test set over the fine-tuning of the T5 model in the αNLI task. It demonstrates that our model effectively utilizes a task-specific self-consistent method to validate the model's output and finalize a consistent answer. **Second**, α-PACE$_{\text{HO1O2}}$ excels the prompt-based baselines (e.g., Prefix-Tuning and Prompt-Tuning) with at least 5.85% test accuracy. This result exhibits the exact contribution of the task-specific self-consistent tailored prompt tuning-based model. **Third**, by adopting the general self-consistent narrative prompts, α-PACE$_{\text{General \& Task Consistency}}$ obtains 92.54% test accuracy and 92.01% accuracy on the leaderboard test set. This result demonstrates that eliciting the inter-sentential coherence from the pre-trained language model and utilizing a general self-consistent framework considering six narrative sequences can partially solve this abductive reasoning task.

Model	Leaderboard (%)
Random	50.41
BERT (Devlin et al., 2019)	66.75
RoBERTa (Liu et al., 2019)	83.91
McQueen (Mitra et al., 2019)	84.18
ege-RoBERTa (Du et al., 2021)	85.95
$L2R^2$ (Zhu et al., 2020)	86.81
UNICORN (T5)(Lourie et al., 2021)	87.34
RoBERTa-L+IMSL (Li et al., 2021a)	87.83
DeBERTa (He et al., 2021)	89.70
DeBERTa(Ensemble) (He et al., 2021)	90.00
UNIMO (Li et al., 2021b)	91.18
α-PACE$_{\text{O1HO2 \& w/o Consistency}}$ (T5)	89.51
α-PACE$_{\text{HO1O2 \& Task Consistency}}$(T5)	91.61
α-PACE$_{\text{General \& Task Consistency}}$(T5)	**92.01**
Human Performance	92.90

Table 4: The accuracy (%) is evaluated on the test dataset from the αNLI task leaderboard. The approximation of learnable parameters for all models is displayed in Table 10 in Appendix A.5.

5.2 Ablation Study

To better study the factors of the α-PACE model, we have devised numerous ablations on the joint probability for self-consistency, general narrative self-consistency, continuous prompt length, prompt engineering, and model size.

Joint Probability for Task-Specific Self-Consistency In our method, by estimating the likelihood of three masks to achieve self-consistency purposes, the dependencies of these three masks are exploited to enhance the ability of the pre-trained language model on this αNLI task. According to the experimental results in Table 5, we can conclude that 1) The performance of our task-specific self-consistent prompt model incorporating the signals from all three masks surpasses other models (e.g., α-PACE$_{\text{First \& Second}}$), emphasizing the significance of dependencies and effectiveness of self-consistency; 2) The model with a single mask (e.g., α-PACE$_{\text{First}}$), without integrating information from the other two masks, exhibits the worst performance; 3) The model with the third mask (e.g., α-PACE$_{\text{Second \& Third}}$), which selects the best hypothesis, performs better than other models that lack the third mask. This finding highlights the importance and necessity of the third mask, summarizing the overall plausibility of two narrative sequences.

Model	Dev (%)	Test (%)
α-PACE$_{\text{First}}$	87.41	86.94
α-PACE$_{\text{Second}}$	88.98	88.09
α-PACE$_{\text{Third}}$	90.60	89.93
α-PACE$_{\text{First \& Second}}$	88.78	89.03
α-PACE$_{\text{First \& Third}}$	90.24	89.48
α-PACE$_{\text{Second \& Third}}$	91.05	90.38
α-PACE$_{\text{SM}}$	91.20	90.13
α-PACE$_{\text{General Consistency}}$	91.78	90.64
α-PACE$_{\text{SM \& Task Consistency}}$	92.43	92.06
α-PACE$_{\text{General \& Task Consistency}}$	**93.15**	**92.54**

Table 5: Ablation study in the joint probability for task-specific self-consistency and general narrative self-consistency on our model with HO1O2 patterns. α-PACE$_{\text{SM}}$ means adopting the sample-and-marginalize method proposed by Wang et al. (2022) on our prompt model without the task-specific discrete prompt tokens.

Model	Dev (%)	Test (%)
BERT-large	49.96$\pm$0.73	50.79$\pm$0.60
RoBERTa-large	58.20$\pm$2.78	58.68$\pm$2.80
McQueen	60.38$\pm$3.23	58.71$\pm$2.25
ege-RoBERTa-large	65.80$\pm$4.30	65.18$\pm$3.27
$L2R^2$	64.81$\pm$2.40	65.68$\pm$3.33
Fine-Tuning(T5)	69.57$\pm$2.01	71.18$\pm$2.06
Prefix-Tuning(T5)	73.96$\pm$5.36	72.29$\pm$5.10
Prompt-Tuning(T5)	76.34$\pm$1.70	75.38$\pm$1.83
α-PACE$_{\text{O1HO2}}$(T5)	82.25$\pm$1.09	81.90$\pm$1.22
α-PACE$_{\text{General Consistency}}$(T5)	**83.48$\pm$0.93**	**83.15$\pm$0.93**

Table 6: Model accuracy (%) using 100 training instances compared with prompt-based models. We report the mean and standard deviation of five runs with different random seeds.

Model	Test (%)
Random	50.40
ChatGPT$_{\text{Prompt}}$	71.07
ChatGPT$_{\text{Task Consistency}}$	72.57
ChatGPT$_{\text{Sample-and-Marginalize}}$	73.17
ChatGPT$_{\text{General \& Task Consistency}}$	74.42

Table 7: The performance of ChatGPT performs on the test set of αNLI task. ChatGPT$_{TaskConsistency}$ means utilizing the concatenate the O^1,O^2, and H^j as the HO1O2 narrative patterns, instead of ChatGPT$_{Prompt}$ treat it as multi-choice questions. ChatGPT$_{\text{Sample-and-Marginalize}}$ means a prompt template sampling six times while the ChatGPT$_{\text{General \& Task Consistency}}$ means sampling with six narrative patterns.

General Narrative Self-Consistency The prior research on the self-consistent prompt-based method (i.e., sample-and-marginalize method) relied on an individual prompt to sample various outputs and perform majority voting to resolve the inconsistency issue that language models suffered (Wang et al., 2022). Therefore, we conducted experiments to compare the performance of both this method and our proposed general self-consistent method on our model, and the result is displayed in Table 5. By combining the likelihood of various outputs, the performance of sample-and-marginalize is slightly improving over the original single pattern-based model. Simultaneously, our general self-consistent approach surpasses this sample-and-marginalize method in two settings, with or without considering task consistency. Therefore, the results evidence the significance of our linguistic phenomenon-based self-consistent prompt, which considers various narrative sequences.

Prompt Length & Prompt Engineering & Model Size Furthermore, we conduct the ablation study on the continuous prompt length, prompt engineering, and the model size of our model in both few-shot and full training configurations. The details we described are in Appendix B.1. The vital information worth mentioning is that without inserting prefix prompt and cloze prompt into our prompt template, the performance will significantly drop, and it illustrates the necessity of these two parts of learnable prompt tokens in our model.

5.3 Few-Shot Setting

Few-Shot Setting Comparing with Prompt-based methods With the sampled 100 training instances, we summarize our experimental results for the αNLI task in Table 6. We report the mean accuracy and standard deviation for five random seeds. As shown in Table 6, our proposed model consistently outperforms other prompt-based models and appears more beneficial in this few-shot setting. Both our general consistency model and single narrative pattern model provide a significant gain over all stated baselines. With training on 100 instances, we observe that our proposed model with a single narrative sequence pattern significantly exceeds the Fine-Tuning (T5) in accuracy on the dev and test datasets by 12.68% and 10.72%, respectively. Furthermore, compared with the prompt-based models, our model still surpasses at least 6.52% test accuracy. The large gap between our model and other T5-based models emphasizes the significance of the task-specific self-consistent method by considering the inter-sentential coherence information, proving that our model can effectively elicit and utilize temporal and causal information between observations and hypotheses. Moreover, after considering six narrative patterns, our α-PACE$_{\text{General Consistency}}$ outperforms all our single pattern models by at least 1.23% and 2.06% in validation and test accuracy,

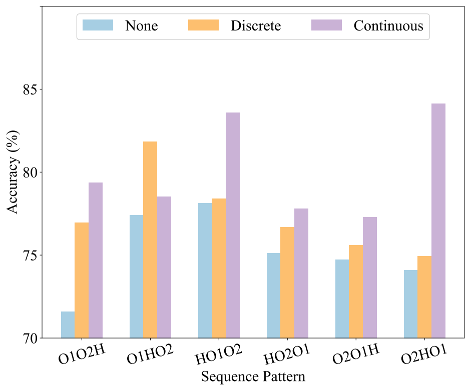

Figure 3: Performance comparison by adopting discourse connective in different settings. All models are training with 100 training instances.

Patterns	Discourse Connectives
O1O2H	meanwhile, in fact, because
O1HO2	in fact, as, as a result
HO1O2	because, meantime, as a result
HO2O1	if, as a result, after
O2O1H	meanwhile, if, as
O2HO1	in fact, as long as, because

Table 8: Top selected connectives for different patterns based on the model performance in the few-shot learning setting.

Pattern	Example
O1O2H	***Meanwhile***, Carl went to the store desperately searching for flour tortillas for a recipe. ***In fact***, Carl left the store very frustrated ***because*** the store had corn tortillas, but not flour ones.
O1HO2	***In fact***, Carl went to the store desperately searching for flour tortillas for a recipe. ***As*** the store had corn tortillas, but not flour ones. ***As a result***, Carl left the store very frustrated.
HO1O2	***Because*** the store had corn tortillas, but not flour ones. ***Meantime***, Carl went to the store desperately searching for flour tortillas for a recipe. ***As a result***, Carl left the store very frustrated.
HO2O1	***If*** the store had corn tortillas, but not flour ones. ***As a result***, Carl left the store very frustrated ***after*** Carl went to the store desperately searching for flour tortillas for a recipe.
O2O1H	***Meanwhile***, Carl left the store very frustrated ***if*** Carl went to the store desperately searching for flour tortillas for a recipe ***as*** the store had corn tortillas, but not flour ones.
O2HO1	***In fact***, Carl left the store very frustrated ***as long as*** the store had corn tortillas, but not flour ones. ***Because*** Carl went to the store desperately searching for flour tortillas for a recipe.

Table 9: Case study for discourse connectives of different model patterns using the same case. The learned connectives are indicated in **boldface**.

respectively. We also study the influence of various training examples. We randomly subsample the entire dataset to obtain smaller datasets of size {1, 5, 10, 20, 50}. More details for the performance are shown in Figure 9 and Figure 10 in Appendix B.2.

5.4 Prompt Adaptation For ChatGPT

With the powerful ability of LLMs exhibited on numerous tasks, we are curious about the capability of ChatGPT on zero-shot abductive commonsense reasoning tasks. Therefore, we test the ability of ChatGPT with four designed templates. The performance is shown in Table 7. All the baselines can outperform random prediction. ChatGPT$_{TaskConsistency}$ improves the performance by 1.5% over ChatGPT$_{Prompt}$ by utilizing the prompt template in the task-specific consistency method. We also find that the general self-consistent prompting (ChatGPT$_{\text{General \& Task Consistency}}$) demonstrates the additional performance boost over other baselines. Compared with ChatGPT$_{\text{Sample-and-Marginal}}$, instead of an individual prompt template sampling six times, our narrative framework, which considers six sequences, performs better on the αNLI task.

5.5 Interpretability

An ideal interpretable prompt should be composed of natural language that makes it obvious why this prompt elicited such behavior from the model (Lester et al., 2021). Since the prompt tuning process only updates the prompt parameters and freezes the pre-trained language model, the learned prompt is expected to encode the inter-sentential coherence information (e.g., temporal and causal information) in our method. Therefore, the nearest neighbors discourse connectives of our learned cloze prompt (used to represent the discourse connectives in each pattern) should reasonably and appropriately describe the relationship between each sentence in various sentence sequences. The motivation of the interpretability section is to provide a view of the perspective of the significance of discourse connective and explore the possibility of different narrative sequences on the αNLI task.

To obtain the nearest neighbors discourse connectives of these continuous cloze prompts in our method, we compute the cosine similarity between the averaged representation of learned cloze prompt tokens and the embedding vector of discourse connectives. The top selected connectives for each sequence pattern are shown in Table 8, and more details of discourse connectives can be found in Appendix B.3. We use the data example utilized to illustrate the full-connect model in Bhagavatula et al. (2020) and insert the top selected connectives in between the sentences to form a narrative text, as shown in Table 9. We observe that the learned discourse connectives can describe the same collection of sentences in various sentence sequences in a rational and acceptable way. More case studies are shown in Table 13 in Appendix B.3.

We further test the performance on three input settings: (1) without continuous prompts inserted, (2) with inserting the top selected connectives as the discrete prompts, and (3) with the cloze continuous prompts. The results are shown in Figure 3, and we see that the discrete connective is substantially superior to the without one. This finding underscores the plausibility and effectiveness of adopting the discourse marker to elicit coherent information from PLM. Moreover, the overall performance of our method with the continuous prompts outperforms the other two settings except for the O1HO2 pattern, where the discrete prompts are slightly better than the continuous prompts. It emphasizes the significance of utilizing continuous prompts to represent the connectives.

6 Conclusion

We developed a model that considers intersentential coherence and self-consistency through prompt tuning for improving the narrative understanding on the αNLI task. Moreover, we propose a general self-consistent framework based on linguistic phenomena. The extensive experiments evidence the effectiveness of our proposed method.

Limitations

Since all utilized information is only elicited from pre-trained language models (PLMs), our method relies on information or knowledge implicitly stored in the PLMs and the task dataset. This limitation restricts the capability owing to the reporting bias (Gordon and Durme, 2013) in the pretrained language models (PLMs). Moreover, our method is limited to the information type that can be elicited from PLMs. The future work for the constraint is to incorporate more abundant and sufficient knowledge to equip the model with more vital abilities. A possible method is adopting the grounding method (Lin et al., 2019) or retrieving the relevant nodes in the knowledge graph for each data instance, providing more contextual information and enhancing the capability of the model on this task.

Ethics Statement

In this work, we conformed to accepted privacy practices and strictly followed the data usage policy. This paper presents a framework for guiding the PLM to understand the narrative context of input from the abductive natural language inference task. The ART dataset from the αNLI task (Bhagavatula et al., 2020) we used to train and evaluate the abductive inference ability of our model is publicly available, and this work is in the intended use. This dataset is collected from the manually curated story corpus ROCstory and should not contain any information that names or uniquely identifies individual people. Since we do not introduce social and ethical bias into the model or amplify any bias from the data, we can foresee no direct social consequences or ethical issues.

Acknowledgements

The authors of this paper were supported by the NSFC Fund (U20B2053) from the NSFC of China, the RIF (R6020-19 and R6021-20) and the GRF (16211520 and 16205322) from RGC of Hong Kong. We also thank the support from NVIDIA AI Technology Center (NVAITC) and the UGC Research Matching Grants (RMGS20EG01-D, RMGS20CR11, RMGS20CR12, RMGS20EG19, RMGS20EG21, RMGS23CR05, RMGS23EG08).

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot. *CoRR*, abs/2001.09977.

Henning Andersen. 1973. Abductive and deductive change. *Language*, pages 765–793.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *CoRR*, abs/2302.04023.

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 *ICLR*.

Antoine Bosselut, Asli Celikyilmaz, Xiaodong He, Jianfeng Gao, Po-Sen Huang, and Yejin Choi. 2018. Discourse-aware neural rewards for coherent text generation. In *NAACL-HLT*, pages 173–184.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: commonsense transformers for automatic knowledge graph construction. In *ACL*, pages 4762–4779.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott M. Lundberg, Harsha Nori, Hamid Palangi, Marco Túlio Ribeiro, and Yi Zhang. 2023. Sparks of artificial general intelligence: Early experiments with GPT-4. *CoRR*, abs/2303.12712.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2020. Make up your mind! adversarial generation of inconsistent natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4157–4165. Association for Computational Linguistics.

Chunkit Chan and Tszho Chan. 2023. Discourse-aware prompt for argument impact classification. In *Proceedings of the 2023 15th International Conference on Machine Learning and Computing*, ICMLC '23, page 165–171, New York, NY, USA. Association for Computing Machinery.

Chunkit Chan, Jiayang Cheng, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2023a. Chatgpt evaluation on sentence level relations: A focus on temporal, causal, and discourse relations. *CoRR*, abs/2304.14827.

Chunkit Chan, Xin Liu, Jiayang Cheng, Zihan Li, Yangqiu Song, Ginny Y. Wong, and Simon See. 2023b. Discoprompt: Path prediction prompt tuning for implicit discourse relation recognition. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 35–57. Association for Computational Linguistics.

Seymour Benjamin Chatman. 1980. *Story and discourse: Narrative structure in fiction and film*. Cornell University Press.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Du, Xiao Ding, Ting Liu, and Bing Qin. 2021. Learning event graph knowledge for abductive reasoning. In *ACL/IJCNLP*, pages 5181–5190.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard H. Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Trans. Assoc. Comput. Linguistics*, 9:1012–1031.

Jonathan Gordon and Benjamin Van Durme. 2013. Reporting bias and knowledge acquisition. In *AKBC*, pages 25–30.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4921–4933, Online. Association for Computational Linguistics.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. PTR: prompt tuning with rules for text classification. *CoRR*, abs/2105.11259.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Jerry R. Hobbs, Mark E. Stickel, Douglas E. Appelt, and Paul A. Martin. 1993. Interpretation as abduction. *Artificial intelligence*, 63(1-2):69–142.

Peter Jansen, Mihai Surdeanu, and Peter Clark. 2014. Discourse complements lexical semantics for non-factoid answer reranking. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 977–986. The Association for Computer Linguistics.

Yuxin Jiang, Chunkit Chan, Mingyang Chen, and Wei Wang. 2023. Lion: Adversarial distillation of closed-source large language model. *CoRR*, abs/2305.12870.

Jan Kocon, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydlo, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, Anna Kocon, Bartlomiej Koptyra, Wiktoria Mieleszczenko-Kowszewicz, Piotr Milkowski, Marcin Oleksy, Maciej Piasecki, Lukasz Radlinski, Konrad Wojtasik, Stanislaw Wozniak, and Przemyslaw Kazienko. 2023. Chatgpt: Jack of all trades, master of none. *CoRR*, abs/2302.10724.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. 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 3045–3059. Association for Computational Linguistics.

Hector J. Levesque. 1989. A knowledge-level account of abduction. In *IJCAI*, pages 1061–1067.

Haoran Li, Dadi Guo, Wei Fan, Mingshi Xu, and Yangqiu Song. 2023. Multi-step jailbreaking privacy attacks on chatgpt. *CoRR*, abs/2304.05197.

Linhao Li, Ming Xu, Yongfeng Dong, Xin Li, Ao Wang, and Qinghua Hu. 2021a. Interactive model with structural loss for language-based abductive reasoning. *CoRR*, abs/2112.00284.

Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2021b. UNIMO: towards unified-modal understanding and generation via cross-modal contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2592–2607. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4582–4597. Association for Computational Linguistics.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *EMNLP-IJCNLP*, pages 2829–2839.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CoRR*, abs/2107.13586.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021b. GPT understands, too. *CoRR*, abs/2103.10385.

Xin Liu, Jiefu Ou, Yangqiu Song, and Xin Jiang. 2021c. Exploring discourse structures for argument impact classification. In *ACL/IJCNLP*, pages 3958–3969.

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.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. UNICORN on RAINBOW: A universal commonsense reasoning model on a new multitask benchmark. 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 13480–13488. AAAI Press.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella Béguelin. 2023. Analyzing leakage of personally identifiable information in language models. *CoRR*, abs/2302.00539.

Arindam Mitra, Pratyay Banerjee, Kuntal Kumar Pal, Swaroop Mishra, and Chitta Baral. 2019. How additional knowledge can improve natural language commonsense question answering? *CoRR*, abs/1909.08855.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Hwee Tou Ng and Raymond J Mooney. 1990. The role of coherence in constructing and evaluating abductive explanations. In *Working Notes of the 1990 Spring Symposium on Automated Abduction, volume TR*, pages 90–32.

Susana Onega and José Angel García Landa. 2014. *Narratology: an introduction*. Routledge.

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

TB OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI*.

Debjit Paul, Anette Frank, and Yang Liu. 2020. Social commonsense reasoning with multi-head knowledge attention. In *Findings of EMNLP*, pages 2969–2980.

Gabriele Paul. 1993. Approaches to abductive reasoning: an overview. *Artif. Intell. Rev.*, 7(2):109–152.

Charles Sanders Peirce. 1974. *Collected Papers of Charles Sanders Peirce*, volume 5. Harvard University Press.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. In *NAACL-HLT*, pages 5203–5212.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.

Marie-Laure Ryan et al. 2007. Toward a definition of narrative. *The Cambridge Companion to Narrative*, 22.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *EACL*, pages 255–269.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4603–4611. PMLR.

Teo Susnjak. 2022. Chatgpt: The end of online exam integrity? *CoRR*, abs/2212.09292.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Slávka Tomaščíková. 2009. Narrative theories and narrative discourse. *Bulletin of the Transilvania University of Braşov• Vol*, 2:51.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *CoRR*, abs/2203.11171.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: learning vs. learning to recall. In *NAACL-HLT*, pages 5017–5033.

Yunchang Zhu, Liang Pang, Yanyan Lan, and Xueqi Cheng. 2020. L2r^2: Leveraging ranking for abductive reasoning. In *SIGIR*, pages 1961–1964.

A Appendix for Experimental Settings

A.1 Baselines

We compare α-PACE with the following competitive baselines :

(a) *BERT* (Devlin et al., 2019) is a language model trained with masked-language modeling and the next sentence prediction objective.

(b) *RoBERTa* (Liu et al., 2019) is a powerful encoder that has the same architecture as BERT with robust optimization and more pre-training data.

(c) *McQueen* (Mitra et al., 2019) is a method to integrate external knowledge (e.g., commonsense knowledge) into a pre-trained language model (i.e., RoBERTa) to address the αNLI task.

(d) *MHKA* (Paul et al., 2020) enhances RoBERTa by incorporating the social commonsense knowledge for the αNLI task.

(e) *ege-RoBERTa-large* (Du et al., 2021) is a variational autoencoder-based model that learns commonsense knowledge by utilizing a latent variable for guiding the abductive reasoning task.

(f) $L2R^2$ (Zhu et al., 2020) reformulates the αNLI task as a ranking problem using the learning-to-ranking framework to rank candidate hypotheses.

(g) *IMSL* (Li et al., 2021a) is an interactive language model that groups the correct/wrong hypotheses instead of ranking the hypotheses and adopts joint softmax focal loss for this αNLI task.

(h) *Fine-tuning (T5)* (Raffel et al., 2020) is an encoder-decoder model pre-trained on a multi-task mixture, where each task is converted into a text-to-text format. T5 performs well out of the box on many tasks by prepending a different prefix to the inputs.

(i) *Prefix-Tuning (T5)* (Li and Liang, 2021): a method concatenates the tunable prefix tokens before the discrete input text, keeps language model parameters frozen, and optimizes these continuous task-specific prefix tokens. The implementation details of the Prefix-Tuning methods are appended in Appendix A.4.

(j) *Prompt-Tuning (T5)* (Lester et al., 2021): a vanilla Prompt Tuning-based model conditioning on a frozen model, releasing the constraints of the prompt templates from discrete to learnable prompts. The implementation details of the prompt tuning methods are appended in Appendix A.4.

(k) *UNICORN (T5)* (Lourie et al., 2021) is a universal commonsense reasoning model with multi-task pre-training based on T5-11b.

(l) *DeBERTa* (He et al., 2021) improves RoBERTa with disentangled attention and enhanced mask decoder training. It is only trained with half of the data used in RoBERTa.

A.2 α-PACE Implementation Details

Our method is built upon the FLAN-T5 (Chung et al., 2022) model was an enhanced version of the T5 model (Raffel et al., 2020) that has been finetuned in a mixture of tasks. We primarily use the 11B version but also experiment with various sizes (Small, Base, Large, and 3B versions) for the ablations. All the T5-based baselines are built upon the same FLAN-T5 model size (e.g., Fine-Tuning, Prompt-Tuning, and Prefix-Tuning). The general configuration follows the setting in Lester et al. (2021). For the full-data training setting, the batch size and maximum sequence length are 1 and 350. We set the prefix length p^0, p^4 as 30, and all remaining cloze prompt lengths as 3. We adopt an Adafactor optimizer by selecting a learning rate in {8e-4, 8e-5, 6e-5, 5e-5, 3e-5}, which yields the best performance on the dev set. The training is performed using cross-entropy loss, and the training steps are 30,000.

For the few-shot learning, we follow the full dataset setting except for the batch size and training steps being 3 and 5,000. Furthermore, we primarily use training set size K = 100 but explore K = {1, 5, 10, 20, 50} in the ablations. We sample the K examples from the full training data with five fixed seeds {55, 58, 68, 72, 1,000}. In this setting, we report the performance by averaging results along with the variance obtained for five different seeds. Prompt tuning is conducted on two NVIDIA RTX A6000 GPUs, and it takes around 52 hours for full-data training and 3 hours for few-shot training.

A.3 Implementation Details of T5 Model Fine-Tuning

All the fine-tuning experiments are run on a server with 4 V100-32GB GPUs. When fine-tuning the 11b version, we use DeepSpeed (Rajbhandari et al., 2020) with ZeRO stage 3 to offload parameters to memory.

Model Input and Output The T5-based model serves as a competitive baseline in the main experiment by adopting the same model and model size. We use the template “`Observation 1: {}\nHypothesis 1: {}\nHypothesis 2: {}\nObservation 2: {}`” to transform a dataset instance into an input string. The model is asked to generate either `Hypothesis 1` or `Hypothesis 2` as

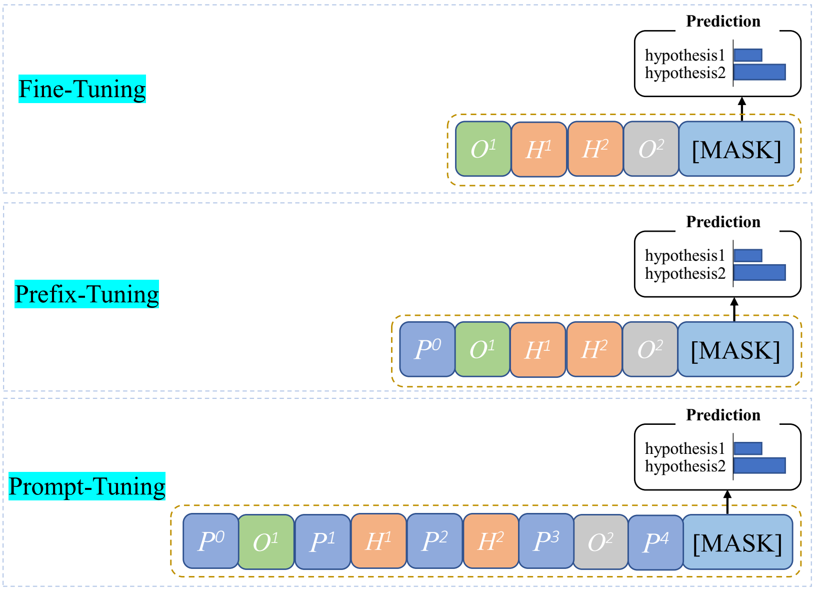

Figure 4: Fine-Tuning, Prefix-Tuning, and Prompt Tuning Templates. Two prompt tuning-based templates perform best among all designed templates in the template searching process for these baselines. The order of observations and hypotheses following the fully connected model proposed by Bhagavatula et al. (2020).

the predicted label. The order of two observations and two hypotheses following the fully connected model proposed by Bhagavatula et al. (2020) and shown in Figure 4 in Appendix.

Hyperparameter Search We first conduct a preliminary experiment to determine the range of hyper-parameters. For base and large model sizes, we set the per-device train and validation batch size as 16 and 64, respectively. For the 11b version, they are set as 8 and 32. Then, we search for the optimal learning rate within {3e-5, 1e-4, 3e-4}. The test performance of the model with the best validation accuracy is reported.

A.4 Implementation Details of the Prefix-Tuning and Prompt Tuning

The prefix tuning (Li and Liang, 2021) and prompt tuning (Lester et al., 2021) methods have been implemented as the baseline in full data learning and few-shot setting for comparison with our model. For a fair comparison, we count all the discrete textual tokens (non-tunable tokens) and the tunable tokens in our prompt template. There are 55 tokens, including 46 tunable tokens and nine textual tokens. In these two baselines, we will insert 55 tunable tokens into the respective prompt template. Moreover, we also adopt the same scale of T5 for these two baselines.

Prefix-Tuning The overall configuration of this model follows the settings of prefix tuning (Li and Liang, 2021). The batch size and maximum sequence length of this model are 8 and 350. The training is performed using cross-entropy loss with an Adafactor optimizer (Shazeer and Stern, 2018). A learning rate selecting in {3e-1, 5e-1, 8e-1} yields the best performance on the validation set, and the training steps are 30,000. We insert 55 prefix tunable tokens into the prefix part of the input template. Since Bhagavatula et al. (2020) stated that the given fixed time sequence (i.e., O_1, H_i, O_2) perform best among all the sequence, the order of two observations and two hypotheses is shown in Figure 4.

Prompt-Tuning The overall configuration of this model follows the settings of prompt tuning (Lester et al., 2021). The batch size and maximum sequence length of this model are 8 and 350. The training is performed using cross-entropy loss, an Adafactor optimizer (Shazeer and Stern, 2018), and a learning rate selecting in {3e-1, 5e-1, 8e-1} yields the best performance on the validation set, with 30,000 training steps. We insert 55 tunable tokens evenly into inter-sentences or between sentences and mask tokens in the input template of this model. The order of two observations and two hypotheses is the same as the above method shown in Figure 4.

A.5 The Approximation of Learnable Parameters

To demonstrate the efficiency of our method, we attach the approximation of the learnable parameters for all models, including our model and the

Model	Parameters
BERT-large (Devlin et al., 2019)	340M
RoBERTa-large (Liu et al., 2019)	355M
McQueen (Mitra et al., 2019)	355M
MHKA (Paul et al., 2020)	355M
ege-RoBERTa-large (Du et al., 2021)	355M
$L2R^2$ (Zhu et al., 2020)	355M
IMSL (Li et al., 2021a)	355M
DeBERTa-large (He et al., 2021)	304M
UNICORN (T5) (Lourie et al., 2021)	11,000M
Prompt Tuning (Lester et al., 2021)	1M
Prefix Tuning (Li and Liang, 2021)	1M
Fine-Tuning (Raffel et al., 2020)	11,000 M
α-PACE$_{\text{HO2O1}}$	34 M

Table 10: The approximation of tunable parameters for models. Most baselines use RoBERTa-large as the backbone model, and their tunable parameters are approximated to be similar.

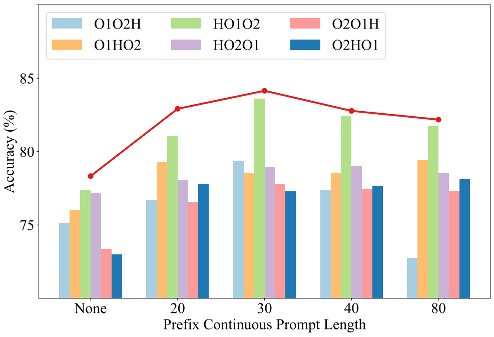

Figure 5: Performance of our method with different numbers of prefix continuous prompt tokens (p^0, p^4) on the test dataset using 100 training instances. The red line indicates the performance of α-PACE$_{\text{General \& Task Consistency}}$.

baselines. The approximation of the learnable parameters is displayed in Table 10 in the Appendix.

B Appendix for Evaluation Result and Analysis

B.1 Ablation study on the α-PACE

Prompt Length Within our designed prompt template, two parts of continuous prompts are concatenated with the input sentences. The first part is two prefix prompts with p^0 and p^4 tokens inserted before template tokens “choice1” and “choice2”. The other part is the cloze prompts inserted to two positions: p^1 (or p^5) tokens between “choice1” (or “choice2”) and the first input sentence, and p^2, p^3 (or p^6, p^7) tokens between input sentences (observations or hypotheses).

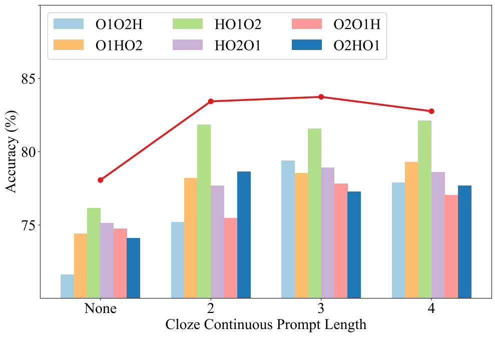

Figure 6: Performance of our method with different numbers of cloze continuous prompt tokens ($p^1, p^2, p^3, p^5, p^6, p^7$) on the test dataset using 100 training instances. The red line indicates the performance of α-PACE$_{\text{General \& Task Consistency}}$.

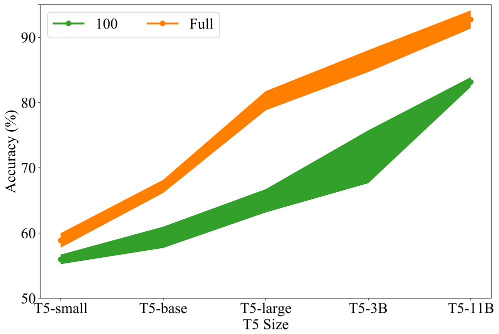

Figure 7: Performance (test accuracy %) comparison on various T5 model sizes in the few-shot and full training settings on the test set.

For the prefix prompt, we train prompts for our model on 100 training instances by varying the prefix prompt length in {None, 20, 30, 40, 80} while keeping the other setting unchanged. Figure 5 shows that the performance of most models with continuous prefix prompts exceeds the “None” one. Inserting the prefix prompt is critical to achieving good performance. After increasing beyond 30 prefix prompt tokens, the performance for different patterns becomes unstable, and some patterns yield low performance, which hurts the performance of the voting method.

For the cloze prompt, we tune our model on 100 training instances by varying the cloze prompt length in {None, 2, 3, 4} while fixing other settings. The result is given in Figure 6, and the overall performance of our model with the cloze prompt is better than the “None” one. Hence, inserting the cloze prompt is another essential factor in obtaining good performance. With the cloze prompt excess of three prompt tokens, the performance of each pattern does not improve significantly, and the performance of the voting method falls.

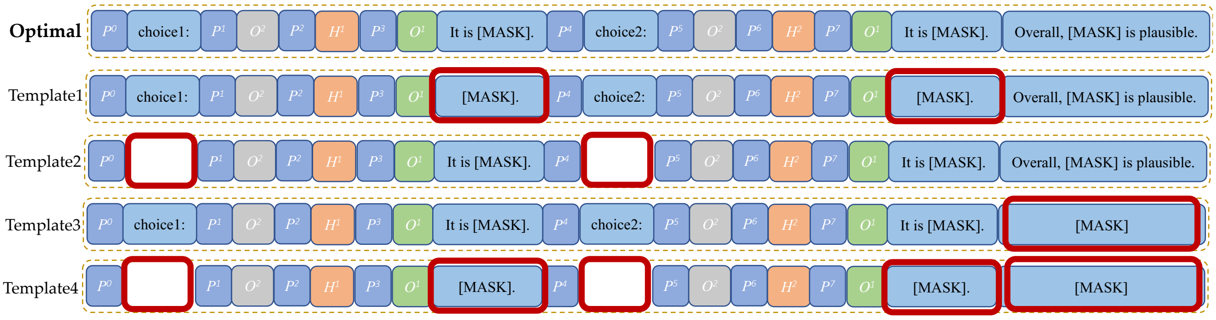

Figure 8: α-PACE prompt template searching. The "Optimal Templates" is the finalized optimal template for implementing experiments to compare with extensive baselines. The red rectangle highlights the modified parts.

Models	Dev	Test
Optimal Template	**82.25**$_{\pm 1.09}$	**81.90**$_{\pm 1.22}$
Template 1	81.81$_{\pm 0.59}$	80.69$_{\pm 1.19}$
Template 2	81.78$_{\pm 0.34}$	81.59$_{\pm 1.90}$
Template 3	81.98$_{\pm 1.09}$	81.26$_{\pm 1.22}$
Template 4	80.61$_{\pm 0.91}$	80.48$_{\pm 0.95}$

Table 11: Ablation study on the discrete textual prompt tokens of α-PACE on αNLI task in few-shot (100 instances) setting.

Prompt Engineering Apart from the continuous prompt in our designed prompt template, there are some tokens in natural textual form and discrete non-tunable tokens. As shown in Figure 8, we gradually remove different portions of these discrete textual tokens and evaluate their importance. The performance shown in Table 11 demonstrates that all discrete textual prompts are essential for achieving satisfactory performance compared with those without manual tips (i.e., Template 4). Among all discrete prompt tokens, the portion "It is <mask>" significantly affects the performance of our model as this discrete part facilitates eliciting the evaluation of PLMs on the plausibility of two hypotheses.

Model Size We compare the performance of various T5 model sizes in both few-shot and full training configurations. As demonstrated in Figure 7, increasing the model size from T5-Large to T5-11B results in an average accuracy increase of around 10% for the few-shot setting and 6% for the full data learning setting. The results imply that a larger model encodes richer narrative knowledge and is advantageous for effectively eliciting more narrative knowledge, which is also our motivation for experimenting with pre-trained models as large as feasible.

B.2 Training Instances

To further study the influence of various training examples. We randomly subsample the entire dataset to obtain smaller datasets of size {1, 5, 10, 20, 50}. More training examples means more narrative context information our model can learn. Figure 9 shows that the average performance of six patterns of α-PACE increases as the number of training instances increases and consistently keeps a large gap against other baselines (e.g., RoBERTa). Interestingly, with a single training instance, the performance of the HO1O2 pattern (from Figure 10) achieves 57.37% test accuracy, much greater than the other five patterns in our model. Therefore, employing an instance-specific narrative sequence pattern may give the pre-trained model a better outcome on limited training instances. This again verifies the motivation for why we involve six narrative sequence patterns in our method. Furthermore, all prompt-based methods received excellent performance in this few-shot setting, consistent with previous works (Lester et al., 2021; Li and Liang, 2021). Furthermore, when baselines train with ten times more data than our method, our single model still outperforms most of these baselines. For example, in Figure 9, the performance of α-**PACE**$_{\text{O1HO2}}$ training with five instances significantly exceeds almost all baseline training with 50 instances.

B.3 Discourse Connectives in Interpretability Section

The Penn Discourse Treebank 2.0 (PDTB 2.0) is a commonly used dataset in discourse parsing tasks and is a large-scale corpus containing 2,312 Wall Street Journal (WSJ) articles annotated by experts (Prasad et al., 2008). There are many discourse connectives in PDTB 2.0 that belong to various discourse relations. Discourse relations are of utmost importance for achieving textual coherence and are deemed an essential step for a multitude of downstream tasks that involve more context, including but not limited to question answering (Jansen et al., 2014), text generation (Bosselut et al., 2018),

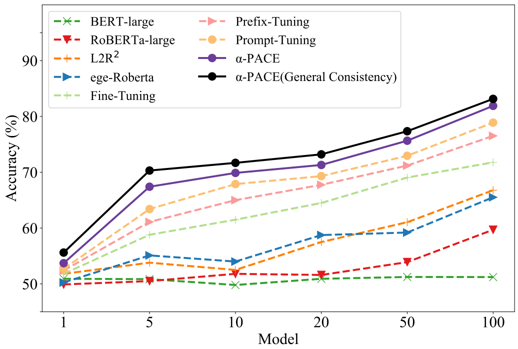

Figure 9: Model performance comparison by using various numbers of the training instances on the test set.

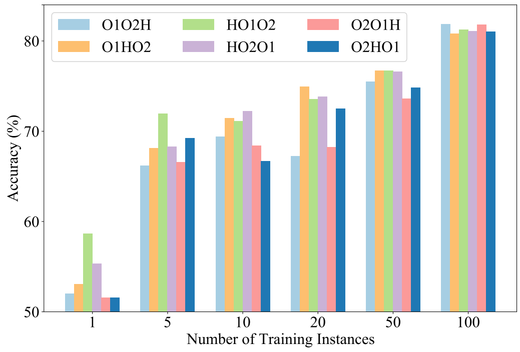

Figure 10: Model performance comparison by using various numbers of the training instances on the test set.

and argument mining (Liu et al., 2021c; Chan and Chan, 2023).

To obtain the nearest neighbors discourse connectives of these continuous cloze prompts in our method, we compute the cosine similarity between the averaged representation of learned cloze prompt tokens and the embedding vector of discourse connectives. We acquired these discourse connectives from the Penn Discourse Treebank 2.0 (Prasad et al., 2008), a commonly used dataset in discourse analysis. These connectives are composed of two main categories of discourse relations: ***Contingency*** and ***Temporal***. There are 23 connectives in these two categories after removing duplicates. The details of connectives can be found in Table 12 and Figure 11 in Appendix. The top selected connectives for each sequence pattern are shown in Table 8 in the few-shot setting. We use the data example utilized to illustrate the full-connect model in Bhagavatula et al. (2020) and insert the top selected connectives in between the sentences to form a narrative text, as shown in Table 9. We observe that the learned discourse connectives can describe the same collection of sentences in various sentence sequences in a rational and acceptable way. More case studies are shown in Table 13 in Appendix B.3.

B.4 ChatGPT Capability on Abductive Commonsense Reasoning

The impressive ability of instruction-following large language models (e.g., ChatGPT (OpenAI, 2022) and GPT-4 (OpenAI, 2023)) has been exhibited by many studies (Bubeck et al., 2023; Bang et al., 2023; Kocon et al., 2023; Chan et al., 2023a; Taori et al., 2023; Chiang et al., 2023; Jiang et al., 2023). There are some challenges remain unresolved such as the associated ethical implications and privacy concerns (Li et al., 2023; Susnjak, 2022; Lukas et al., 2023). In this work, we are curious about the capability of ChatGPT on zero-shot abductive commonsense reasoning tasks. Hence, we test the ability of ChatGPT [2] with four designed templates on the test set of αNLI task. ChatGPT$_{prompt}$ reformulate the task as the multi-choice questions to predict the class label by following Bang et al. (2023). ChatGPT$_{TaskConsistency}$ concatenates the O^1,O^2, and H^j as two narrative sentence sequences, which is the same as the prompt template shown in Figure 2. ChatGPT$_{\text{Sample-and-Marginalize}}$ is to sample six times with an individual prompt template. ChatGPT$_{\text{General \& Task Consistency}}$ utilizes six narrative patterns as the input template and each time only feeds only one narrative pattern. Furthermore, it is imperative to note that the input template, which incorporates in-context learning, is heavily dependent on the chosen training examples that form the prefix demonstration of the prompt template. The performance of in-context learning is subject to high variance based on the specific examples chosen, the quantity of examples, as well as the order in which they are presented. Consequently, this particular template has been excluded from consideration in this particular section. The performance of the random guess model is derived via the averaging of the results obtained from five distinct iterations.

[2] The evaluation is performed in February 2023 by calling ChatGPT API.

Relation	**Connectives**
Contingency	when (4), furthermore (1), indeed (1)
Contingency.Cause	as (1)
Contingency.Cause.Reason	because (2098), as (502), since (248)
Contingency.Cause.Result	so (843), as a result (271), thus (220)
Contingency.Condition.Factual past	if (5)
Contingency.Condition.Factual present	if (48), if then (7), when (7)
Contingency.Condition.General	if (145), when (124), as long as (6)
Contingency.Condition.Hypothetical	if (563), when (22), if then (20)
Contingency.Condition.Unreal past	if (47), if then (1)
Contingency.Condition.Unreal present	if (87)
Contingency.Pragmatic cause.Justification	because (30), as (13), in fact (7)
Contingency.Pragmatic condition.Implicit assertion	if (23), when (11), because (2)
Contingency.Pragmatic condition.Relevance	if (12), when (1), so (1)
Temporal	when (6), meanwhile (1), while (1)
Temporal.Asynchronous	before and after (1), meantime (1), in turn (1)
Temporal.Asynchronous.Precedence	then (556), before (240), until (100)
Temporal.Asynchronous.Succession	after (452), when (181), previously (124)
Temporal.Synchrony	when (475), as (427), while (236)

Table 12: Penn Discourse Treebank 2.0 contingency and temporal connectives, where frequencies are reported in brackets.

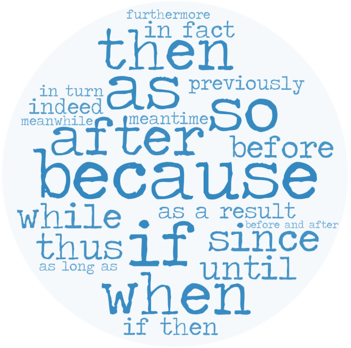

Figure 11: Penn Discourse Treebank 2.0 top frequency discourse markers.

Pattern	**Example**
O1O2H	**Meanwhile**, Jimmy had to get a root canal. **In fact**, He did not feel a thing and the procedure went smoothly **because** Jimmy got plenty of novocaine for the procedure.
O1HO2	**In fact**, Jimmy had to get a root canal.**As** Jimmy got plenty of novocaine for the procedure. **As a result**, He did not feel a thing and the procedure went smoothly.
HO1O2	**Because** Jimmy got plenty of novocaine for the procedure. **Meantime**, Jimmy had to get a root canal. **As a result**, he did not feel a thing and the procedure went smoothly.
HO2O1	**If** Jimmy got plenty of novocaine for the procedure. **As a result**, he did not feel a thing and the procedure went smoothly **after** Jimmy had to get a root canal.
O2O1H	**Meanwhile**, he did not feel a thing and the procedure went smoothly **if** Jimmy had to get a root canal **as** Jimmy got plenty of novocaine for the procedure.
O2HO1	**In fact**, he did not feel a thing and the procedure went smoothly **as long as** Jimmy got plenty of novocaine for the procedure. **Because** Jimmy had to get a root canal.
O1O2H	**Meanwhile**, Jane was a professor teaching piano to students. **In fact**, Jane spent the morning sipping coffee and reading a book **because** none of Jane's students had a lesson that day.
O1HO2	**In fact**, Jane was a professor teaching piano to students. **As** none of Jane's students had a lesson that day. **As a result**, Jane spent the morning sipping coffee and reading a book.
HO1O2	**Because** none of Jane's students had a lesson that day. **Meantime**, Jane was a professor teaching piano to students. **As a result**, Jane spent the morning sipping coffee and reading a book.
HO2O1	**If** none of Jane's students had a lesson that day. **As a result**, Jane spent the morning sipping coffee and reading a book **after** Jane was a professor teaching piano to students.
O2O1H	**Meanwhile**, Jane spent the morning sipping coffee and reading a book **if** Jane was a professor teaching piano to students **as** none of Jane's students had a lesson that day.
O2HO1	**In fact**, Jane spent the morning sipping coffee and reading a book **as long as** none of Jane's students had a lesson that day **because** Jane was a professor teaching piano to students.

Table 13: Case study for the discourse connectives of different model patterns using the same data example. The learned connectives are highlighted in **bold**.