Learning to Generalize for Cross-domain QA

Yingjie Niu[* 1,2], **Linyi Yang**[* 3,4], **Ruihai Dong**[1,2], **Yue Zhang**[3,4]

[1] School of Computer Science, University College Dublin
[2] SFI Centre for Research Training in Machine Learning
[3] Institute of Advanced Technology, Westlake Institute for Advanced Study
[4] School of Engineering, Westlake University
{yingjie.niu}@ucdconnect.ie, {ruihai.dong}@ucd.ie
{yanglinyi, zhangyue}@westlake.edu.cn

Abstract

There have been growing concerns regarding the out-of-domain generalization ability of natural language processing (NLP) models, particularly in question-answering (QA) tasks. Current synthesized data augmentation methods for QA are hampered by increased training costs. To address this issue, we propose a novel approach that combines prompting methods and linear probing then fine-tuning strategy, which does not entail additional cost. Our method has been theoretically and empirically shown to be effective in enhancing the generalization ability of both generative and discriminative models. Our approach outperforms state-of-the-art baselines, with an average increase in F1 score of 4.5%-7.9%. Furthermore, our method can be easily integrated into any pre-trained models and offers a promising solution to the under-explored cross-domain QA task. We release our source code at Github[*].

1 Introduction

Question answering (QA) models (Oh et al., 2016; Trischler et al., 2017; Lewis et al., 2021; Gu et al., 2021) aim to answer passage-based questions automatically with the help of facts in a given context (sometimes referred to as machine reading comprehension (Dua et al., 2019; Sen and Saffari, 2020)). Over the last few years, pre-trained models have achieved great progress on a variety of large-scale datasets, e.g., SQuAD (Rajpurkar et al., 2016), NewsQA (Trischler et al., 2017), DROP (Dua et al., 2019), CoRA (Asai et al., 2021), and NarrativeQA (Kocisky et al., 2018). However, existing methods can suffer significant performance degradation when the tuned system is directly applied to out-of-domain examples (Gururangan et al., 2018; Wu et al., 2020; Tripuraneni et al., 2020; Kaushik et al., 2020; Malinin et al., 2021; Varshney et al., 2022).
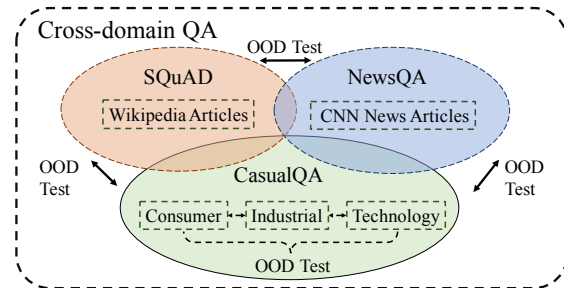

Figure 1: Cross-domain QA task consists of three datasets from different domains.

This paper focuses on a novel cross-domain QA task where we assume models trained on the source domain can be generalized to the target domain, where no labeled or unlabeled data is available. As shown in Figure 1, QA pairs from different domains have intrinsically different feature distributions. For example, in the technology field, the context can frequently contain "e-commerce" and "network". While in the pharmaceutical sector, the context can consist of "COVID-19", "vaccine", and "diagnostic" more frequently. Cross-domain QA poses significant challenges to real-world scenarios, and it is proved that even large-scale pre-trained models (Gu et al., 2021) can still encounter performance degradation under domain generalization.

To address these drawbacks, we introduce a novel cross-domain QA setting, focusing on the methods that consistently improve the domain generalization performance without additional computational costs. Intuitively, cross-domain QA can benefit from prompting in which instances from different domains can share a unified set of label words. Thus, no additional parameters can carry domain-specific information to hinder the OOD generalization for an unseen domain. However, using the prompt method solely could increase the risk of overfitting and bring limited benefits, as prompt templates are fixed, which may be learned as spurious features by models. Thus, we consider

---

[*]These authors contributed equally to this work.
[*]https://github.com/FreddieNIU/Prompt-QA

using the linear-probing and then fine-tuning (LP-FT) strategy to reduce the reliance between prompt patterns with labels by freezing pre-trained parameters. In this way, LP-FT can benefit cross-domain QA by preventing pre-trained features from being distorted when tuning on a specific domain (Kumar et al., 2022). Prompting-based LP-FT method does not introduce new parameters, so the performance decay when training on a source domain and testing on a new target domain can be reduced without entailing additional cost.

Under the LP-FT framework, we introduce four prompt types to extract invariant features in different domains: question type, sentiment, named entity, and key phrase. These prompts aim to increase the question similarity and benefit the model in generalizing to out-of-domain questions. Existing prompting methods have not been applied to natural language processing tasks beyond simple fine-tuning settings. To enable promoting methods to adapt LP-FT, we theoretically prove that LP-FT still holds consistently better robustness for prompting methods (Section 3.3).

We experiment on three different domain datasets (Figure 1). Results show that our prompt-based LP-FT method significantly improves the performance of cross-domain QA models on either the standard hold-out or OOD tests, with an average increase in F1 of 4.5%-7.9% compared to baselines. Also, our method consistently outperforms the standard fine-tuning strategy on both discriminative and generative models. Besides, we provide an in-depth analysis of the ablation study towards the OOD robustness that details the efficacy of LP-FT and prompting methods, respectively. To our knowledge, we are the first to present a new zero-shot cross-domain QA task and propose a novel Prompt-based LP-FT method. All resources are available at https://github.com/FreddieNIU/Prompt-QA.

2 Related Work

Out-of-domain performance degradation has attracted considerable research interest recently. A line of work (Morgan and Winship, 2015; Wang and Culotta, 2021; Kaushik et al., 2021; Yang et al., 2021; Malkiel and Wolf, 2021; Lu et al., 2022) aims to improve models' generalization ability on text classification. Differently, we investigate the OOD generalization problem on the QA task.

Lewis et al. (2021) and Wang et al. (2021) find that 60-70% of test-time answers of popular open-domain QA benchmark datasets exist in the training set, and it is proved that training set memory plays a vital role in testing. Liu et al. (2021a) empirically prove that language models suffer performance degradation when there is no train-test set overlapping. To test the actual generalization ability of QA models, several novel QA datasets have been constructed and released, focusing on evaluating QA models on out-of-domain generalization ability (Gu et al., 2021). Yang et al. (2022) present the first cross-domain QA dataset and observe a performance decay problem regarding the OOD test. Many existing methods intend to improve the OOD performance of QA models through data augmentation. Yue et al. (2022) introduce a synthesizing question-answer pairs method to improve target-domain QA performance. In contrast, we propose a prompt-based method combined with linear probing and then fine-tuning, which is more computationally efficient and does not require target domain annotations.

Prompt-based methods on pre-trained language models have received considerable research interest. The paradigm "*pre-train, prompt, and predict*" replaces the "*pre-train, fine-tune*" procedure for improved few-shot learning (Liu et al., 2021b). Prompt-based methods have been applied not only in sentence-level few-shot learning tasks, such as named entity recognition (Ma et al., 2021) but also in sophisticated learning tasks like natural language understanding (Wang et al., 2022). However, little work applies prompts on the cross-domain QA tasks (Jacovi et al., 2021). We leverage the fixed-format characteristic of the prompt to extract the invariant features in the changing dataset to enhance the OOD generalization of the model.

Instead of fine-tuning, *linear probing* is an alternative to tuning a pre-trained model on a downstream task. Liu et al. (2019a); Tripuraneni et al. (2020) examine the representations produced by several recent pre-trained language models and find that linear models trained on top of frozen contextual representations are competitive with state-of-the-art task-specific fine-tuned models in many cases but fail in tasks where fine-grained language knowledge is required. Kumar et al. (2022) theoretically prove that the linear-probing then fine-tuning (LP-FT) approach can enhance the OOD generalization ability of pre-trained models. In our work, we are the first to provide theoretical evidence that LP-FT still holds consistently better robustness for
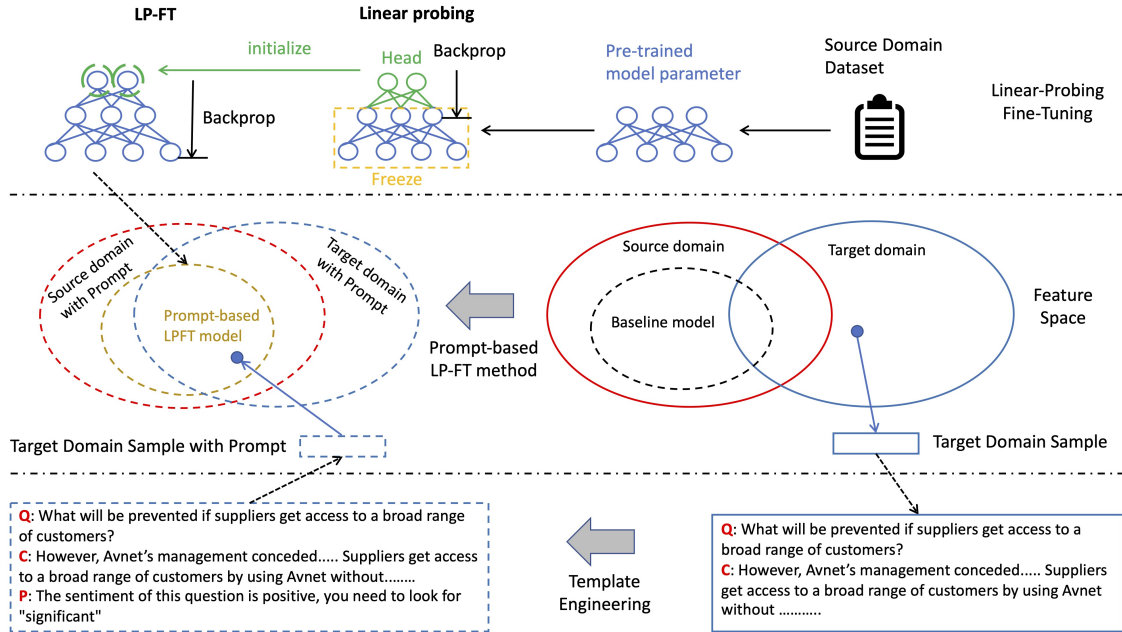
Figure 2: The workflow of the prompt-based linear-probing then fine-tuning strategy. The *bottom* part shows the template engineering process where we add a prompt for each sample. The *middle* part represents the feature space of the Prompt LP-FT model and baseline models. Compared with the baseline model, in the feature space, the prompt-based LPFT model is superior in two respects: the distance between the feature distributions of the two domains is reduced, and the features learned by the model are closer to the intersection of the two domains. The *top* part demonstrates the linear probing and then fine-tuning process. The "Q", "C" and "P" represent "Question", "Context" and "Prompt" in a sample, respectively (*in bottom*)

.

prompting methods in NLP.

3 Method

Figure 2 illustrates the workflow of the Prompt-based LP-FT method. We first generate a prompt for each input sample through template engineering and prompt designing (§3.1). Then, the source domain dataset with prompts is used for linear probing and then fine-tuning (§3.2) a pre-trained model. (*top*). Compared with the baseline model, in the feature space, the prompt-based LPFT model is superior in two respects: the distance between the feature distributions of the two domains is reduced, and the features learned by the model are closer to the intersection of the two domains (*middle*). The feature space demonstrates how the prompt-based method and the LP-FT strategy benefit the cross-domain QA, respectively, and also shows the motivation for using prompt-based LP-FT to benefit the cross-domain QA task.

3.1 Template Designing

We take a *Template Engineering* process to look for a prompt template that results in the most ef-

fective performance on a given task. The template designing rules can be found in *appendix A.2*. The prompt design is inspired by the process of a non-native speaker (or a non-professional reader) reading articles (or professional documents) and answering questions. They may lack some depth of knowledge, such as the meanings of less commonly used words (or domain-specific knowledge). Language models may encounter similar situations in the cross-domain QA task. We design four types of templates. Figure 3 gives an example of a question-type template. Other template designs can be found in *appendix A.3*. Below, we take the question-type template as an example to illustrate the template designing process:

Question-type Templates. Suppose that for a given question, *"Why have we increased our projections for cancer drug Loxo305 and diabetes drug tripeptide is useful?"*, a human tester tries to find the answer from the article. In the question, users might not understand tokens such as *Loxo305, diabetes, tripeptide*, etc. However, if the user is aware that the question might be about "Why something is useful?", then she/he can search
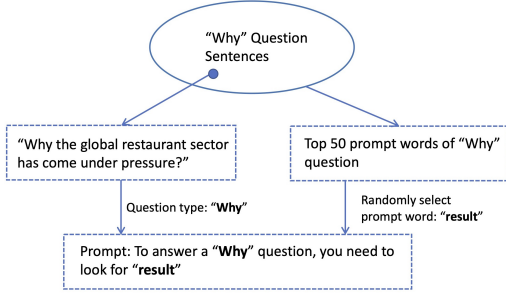
Figure 3: Generating question-type prompts

some keywords such as *because*, *as*, and *since* from the article and the context following these words, which might help her/him to find the correct answer. For each type of question, some specific words might help to locate their answers.

We consider four typical types of questions. For each question type, we first find out the most related words with it, such as *because*, *since* with the question type *why*, by measuring Pointwise Mutual Information (PMI) scores (Bouma, 2009) between candidate words and the question type. Afterward, we select the 50 most related words to generate a prompt for each question.

Loss Functions. For a prompt-based QA task, given a question-context-prompt tuple (Q, C, P), we calculate the probability of each word c_n being the start position or end position of the correct answer for discriminative models as follows:

$$p(c_n|Q,C,P) = Softmax(W_{head}h_{c_n} + b_{head}) \quad (1)$$

where $Q \in \mathbb{R}^{s_q \times d_{word}}$, $C \in \mathbb{R}^{s_c \times d_{word}}$, and $P \in \mathbb{R}^{s_p \times d_{word}}$ denote the question, context and prompt, respectively. $s_q/s_c/s_p$ and d_{word} denote the *question/context/prompt sentence length* and the *word embedding dimension*, respectively. h_{c_n} denotes the feature representation of (Q, C, P) concatenated on the first dimension produced by a pre-trained model, $W_{head} \in \mathbb{R}^{\nu \times d_h}$ and $b_{head} \in \mathbb{R}^{\nu}$. d_h denotes the dimension of h_{c_n} and ν denotes the length of answer sentence. The loss function is the sum of the cross entropy for start and end positions.

$$\mathcal{L}_{dis} = -\sum_{n=1}^{m} \log p(c_n|Q,C,P) \quad (2)$$

where m is the number of words in C.

We regard the QA task as a Seq2Seq generation task for generative models and use the LM loss,

$$\mathcal{L}_{gen} = -\sum_n \log p(c_n|c_{<n}, Q, C, P) \quad (3)$$

where $c_{<n}$ denotes the generated words.

3.2 Linear Probing then Fine-tuning

The OOD generalization problem is defined as follows (Kumar et al., 2022): given a predictor f: to map inputs X to outputs Y. For some loss function $\mathcal{L}$, the predictor in-domain performance L_{id} and out-of-domain performance L_{ood} are:

$$L_{id}(f) = \mathbb{E}_{(X,\mathbf{y}) \sim p_{id}} [\mathcal{L}(f(X)), \mathbf{y}]$$
$$L_{ood}(f) = \mathbb{E}_{(X,\mathbf{y}) \sim p_{ood}} [\mathcal{L}(f(X)), \mathbf{y}] \quad (4)$$

where the predictor is evaluated on test samples $(X, \mathbf{y})$ drawn from in-domain distribution P_{id}, and also evaluated on test samples $(X, \mathbf{y})$ drawn from out-of-domain distribution P_{ood}. To simplify the formula representation, in this paper, X represents the question, and context (Q and C); $\mathbf{y}$ indicates the answer sentence.

The final predictor f is parameterized as a feature extractor and a linear "head". Hence, the training loss is:

$$\hat{\mathcal{L}}(\mathbf{v}, B) = ||XB^\top \mathbf{v} - \mathbf{y}||_2^2 \quad (5)$$

where $\mathbf{v}$ denotes the linear head and B denotes the feature extractor. We assume that the initial feature extractor B_0 is obtained from the pre-trained model, considering two methods to learn a predictor $f_{\mathbf{v}, B}$: 1) linear probing where $B = B_0$ and the linear head is obtained by minimizing some loss on the training data (Liu et al., 2019a), and 2) fine-tuning where both $\mathbf{v}$ and B are updated on the training data with B initialized as B_0 (Kumar et al., 2022).

3.3 Theoretical Proof

We prove that linear probing and then fine-tuning improves the results for prompt tuning by extending the proof for standard fine-tuning (Kumar et al., 2022). In particular, the derivative of Eq 5 with respect to the feature extractor B is:

$$\nabla_B \hat{\mathcal{L}}(\mathbf{v}, B) = 2\mathbf{v}(\mathbf{y} - XB^\top \mathbf{v})^\top X \quad (6)$$

For Eq 6, if U is a sample extracted from a direction orthogonal subspace to the training subspace, $\nabla_B \hat{\mathcal{L}}(\mathbf{v}, B)U = 0$, the training process on X will not decrease the loss on the orthogonal subspace. However, the gradient is not zero for directions in the ID subspace. This explains why fine-tuning can achieve a higher ID performance but a lower OOD performance.

In our proposed prompt-based method, the prompt P is concatenated to the original X (along the sentence length dimension), and the equation can be expressed below:

$$\nabla_B \hat{\mathcal{L}}_p(\mathbf{v}, B) = 2\mathbf{v}(Y - (X + P)B^\top \mathbf{v})^\top (X + P) \tag{7}$$

where assume that P is not orthogonal to the X or its orthogonal subspace. Consequently, we have $\nabla_B \hat{\mathcal{L}}_p(\mathbf{v}, B)(U + P) \neq 0$. In this way, the training process on X with prompt P would modify the loss on the OOD samples with the prompt.

In the linear probing and then fine-tuning method, the OOD error of fine-tuning is

$$\sqrt{L_{ood}(\mathbf{v}_{ft}, B_{ft}(t))} \geq \sigma \frac{min(\varphi, \varphi^2/||w_*||_2)}{(1 + ||w_*||_2)^2} \tag{8}$$

where $\mathbf{v}_{ft}$ and B_{ft} are the linear head and feature extractor after fine-tuning. σ is a fixed parameter (Kumar et al., 2022) related to B_0. $w_* = \mathbf{v}_* B_*$, $\mathbf{v}_*$ and B_* are the optimal parameters. φ is the initial head alignment error $\varphi = |(\mathbf{v}_0^\top \mathbf{v}_*)^2 - (\mathbf{v}_*^\top \mathbf{v}_*)^2|$. In order to decrease the OOD error, the head $\mathbf{v}_0$ has to be as close to the $\mathbf{v}_*$ as possible. It is proved that initializing the head with $\mathbf{v}_{lp}$ (LP-FT) can decrease the OOD error (Kumar et al., 2022) more than random initializing head with $\mathbf{v}_0$ (FT) since $\mathbf{v}_0$ is far away from $\mathbf{v}_*$. Converting input X to $X + P$ does not affect $\frac{min(\varphi, \varphi^2/||w_*||_2)}{(1 + ||w_*||_2)^2}$, implying the LP-FT strategy can be applied after we introduce P. As a result, the Prompt-based LP-FT strategy is used to avoid distorting pre-trained features.

4 Experimental Setup

We introduce experiments' datasets, baseline methods, and evaluation metrics in this section.

4.1 Datasets

We evaluate the proposed method on three datasets: **Causal QA** (Yang et al., 2022), **SQuAD 1.1** (Weissenborn et al., 2017) and **NewsQA** (Trischler et al., 2017). All datasets are in English. Domain-related information is provided in the CausalQA dataset, which is valuable for cross-domain question-answering tasks. For the in-domain test, we experiment on the whole CausalQA dataset before splitting into domains(domain-independent QA) and on each particular domain after splitting into domains. The distribution change and word overlap between datasets can be found in appendix A.4.

For the OOD test, we have two experiment setups: **Setup 1)** we split the CausalQA dataset, based on the domain information, into mutually exclusive training/validation/testing sets in the same ratio of 8:1:1. **Setup 2)** we conduct OOD tests across different datasets from different domains. In cross-domain QA, both the training and validation sets of the source domain are used in the training process for hyperparameter searching. The testing sets of source and target domains are used for in-domain evaluation and OOD tests, respectively.

4.2 Baseline Models

Based on the novel cross-domain QA setting, we establish baselines using generative models – BART (Lewis et al., 2020), T5 (Raffel et al., 2020) – and discriminative models – BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019b), and SpanBERT (Joshi et al., 2020) with the help of Huggingface framework (Wolf et al., 2020). We also implement the commonly used domain adaptation method in previous works (Yue et al., 2022; Cao et al., 2020) to compare with our method. The AdamW optimizer has a default learning rate of 10^{-5}. Other hyper-parameters are tuned by optimizing the performance on the validation set. The standard fine-tuning strategy is considered a baseline when compared to our methods by using four strategies:

1. Baseline: we select the RoBERTa-base model as the baseline of discriminative methods.

2. Baseline + P: we adopt the same baseline models and fine-tuning strategy, only replacing the original dataset with the prompted dataset.

3. Baseline + LP-FT: we first tune the last linear layer (the "Head" for question answering) parameters and replace the head parameters initialized by Huggingface framework models with the head parameters after linear probing. The original dataset is used in this section.

4. Baseline + P + LP-FT: the LP-FT strategy is adopted on the dataset with the prompt.

4.3 Evaluation Metrics

Following previous work (Gu et al., 2021; Yang et al., 2022), The Macro F1-score (F1) and exact match (EM) are used to evaluate the model's performance. If the predicted answer matches the true

BART		Consumer		Industrial		Technology		RoBERTa		Consumer		Industrial		Technology	
Train/Test		F1	EM	F1	EM	F1	EM	**Train/Test**		F1	EM	F1	EM	F1	EM
Con	Ori	**70.29**	24.53	68.44	24.07	68.61	23.61	Con	Ori	**78.20**	51.38	72.49	47.68	74.63	49.07
	Ours	(+0.36)	(+0.47)	(+1.7)	(+2.78)	(+3.39)	(+4.63)		Ours	(+0.19)	(+1.86)	(+2.58)	(+0.93)	(+2.83)	(+0.47)
Ind	Ori	70.11	31.31	**72.53**	32.41	69.63	27.27	Ind	Ori	77.81	49.45	**80.05**	58.46	77.91	48.35
	Ours	(+3.95)	(+4.75)	(+4.05)	(+7.48)	(+3.84)	(+9.93)		Ours	(+2.74)	(+9.57)	(+0.65)	(+2.20)	(+1.09)	(+9.02)
Tech	Ori	69.89	30.30	69.53	27.77	**71.79**	33.83	Tech	Ori	75.54	55.05	73.99	48.98	**76.49**	54.04
	Ours	(+2.78)	(+3.03)	(+2.98)	(+3.45)	(+2.23)	(+0.51)		Ours	(+2.39)	(+0.00)	(+2.71)	(+7.08)	(+0.27)	(-0.50)

Table 1: Out-of-domain test results of the BART-base model (Left) and RoBERTa-base (Right) on CausalQA (Setup 1). The numbers in brackets represent the performance improved by our method. "Ori" denotes the original fine-tuning method, and "Ours" denotes the Prompt-based LP-FT method.

	Dev		Test	
Methods	F1	EM	F1	EM
BART	74.16	36.50	73.26	34.49
BART + LP-FT	74.06	35.03	73.83	34/00
BART + P	75.60	37.47	75.33	37.66
BART + P + LP-FT	**77.60**	**41.22**	**76.90**	**39.44**
RoBERTa	83.97	61.82	83.45	61.28
RoBERTa + LP-FT	84.80	62.15	83.49	61.18
RoBERTa + P	84.55	**62.20**	83.61	61.34
RoBERTa + P + LP-FT	**84.56**	62.15	**83.87**	**61.42**

Table 2: Results of Domain-Independent QA on CausalQA dataset. 'F1' refers to Macro F1, EM refers to exact match. The model name refers to base models, "+P" denotes the base model+prompt method, "+LP-FT" denotes the base model+LP-FT method.

answer for each question-answer pair, EM =1. Otherwise, EM = 0. The Macro F1 score is defined as the mean of token-level F1 scores:

$$Macro\ F1 - score = \frac{1}{N} \sum_{i=0}^{N} F1 - score_i \quad (9)$$

where i is the token index and N is the length of the golden answer.

5 Results and Discussion

Our method is applied to both domain-independent QA tasks (§5.1) and cross-domain QA tasks (§5.2).

5.1 In-domain Performance

For domain-independent QA, the in-domain performance represents the model performance using the traditional hold-out test, where both the training set and test set come from the whole dataset without splitting domains. The domain-independent results are shown in *Table 2*, where the Prompt LP-FT method brings performance gain over both the BART model (in average +3.64% in F1, +4.95% in EM) and the RoBERTa model (in average +0.42% in F1, +0.14% in EM). Taking the BART model as an example, *BART+LP-FT* achieves slightly better performance (+0.57%) compared with *BART*,

which shows the LP-FT method brings limited benefits to the model on the domain-independent QA task. However, *BART+P* (+2.07%) over *BART* outperforms *BART+LP-FT* (+0.57%) over *BART*, which shows that the prompt-based method can benefit the model without splitting domains.

In *Table 1*, the numbers on the diagonal represent the ID performance on each domain, and the values in parentheses below represent the in-domain performance increase brought by our method (in average +2.21% in F1 and +2.82% in EM) (*left*). Though the performance gain on each domain varies , our method consistently improves the performance of in-domain evaluations.

5.2 Out-of-domain performance

Results on CausalQA. The experiment results of cross-domain CausalQA are shown in 3×3 tables *Table 1* where each row represents contrast experiments with the same testing data, and each column represents the model performance on different test sets. The numbers not on the diagonal represent the performance tested on a domain different from the training domain, called OOD test results. Overall, the proposed method benefits the OOD performance by an average of +3.11% in F1 and 4.76% in EM on BART and by an average of +2.39% in F1 and 4.51% in EM on RoBERTa. For example, by comparison in the same scenario, we find that the improvement on *Consumer-Train/Industrial-Test* on BART (+3.95%) is more significant than the improvement on *Consumer-Train/Industrial-Test* based on RoBERTa (+2.74%). Our method brings larger performance gains for generative models (BART) than discriminative models (RoBERTa). These results show that the performance benefits based on discriminative models are less than generative models by using Prompt LP-FT. Intuitively, this can be because the added prompt can be used directly to generate answers as we fine-tune BART

	S – >N	S – >C	N – >S	N – >C	C – >S	C –> N
RoBERTa	37.60	66.58	49.87	44.22	19.44	7.45
RoBERTa+DA (Yue et al., 2022)	38.26	66.14	50.31	43.05	22.74	7.15
RoBERTa+P	38.17*	66.84	50.97*	48.37*	21.41*	**8.64***
RoBERTa+LPFT	37.95*	66.60	50.28*	45.86*	20.92*	7.5
RoBERTa+P+LPFT	**38.76**	**66.86***	**52.41***	**51.64***	**23.02***	7.73*
BART	33.71	46.97	43.49	31.78	26.14	8.69
BART+DA (Yue et al., 2022)	35.09	55.65	44.05	33.47	26.98	9.02
BART+P	**36.81***	**56.22***	43.61*	31.91	25.96	9.26*
BART+LPFT	33.29	53.29*	44.05*	31.95	26.87*	9.49*
BART+P+LPFT	35.23*	55.86*	**44.36***	**33.79***	**27.61***	**9.54***

Table 3: OOD test results on SQuAD (S), CausalQA (C), and NewsQA (N) (Setup 2). [S–>N] represents that the model is trained on SQuAD while tested on NewsQA. "+P" represents the prompting methods. "+DA" represents the Domain Adaptation method (Yue et al., 2022). The proposed method shows statistically significant improvements compared to the baseline model indicated by ∗ using Student T-test (p<0.01, 10-time run).

in a Seq2Seq manner.

In *Table 1*, we compare the performance of BART on *Consumer-Train/Consumer-Test* to *Consumer-Train/Industrial-Test*. Our method improves the performance by +0.36% on the consumer test set and +3.95% when testing on the samples from the industrial domain, indicating that the proposed method is better for cross-domain generalization. Moreover, the benefit on *Consumer-Train/Technology-Test* (+2.78%) is relatively small compared to the improvement on *Consumer-Train/Industrial-Test* (+3.95%). It hints that the same prompt has variant effects on different domains. This can be because different domains have intrinsically different feature distributions.

OOD Tests Between Different Datasets. We show the OOD generalization results between different popular datasets in Table 3. It can be seen that the Prompt LP-FT method improves the OOD test performance of RoBERTa on average by **2.54%** on three data sets and **2.60%** for BART. It is worth noting that our method brings a performance improvement of up to **7.42%** (NewsQA–CausalQA) on RoBERTa, while the maximum performance improvement reaches **8.89%** (SQuAD–CausalQA) on BART. The result is consistent with the finding in Setup 1 that Prompt LP-FT can benefit generative models more than discriminative ones.

It is noteworthy that even though our method assumes that no target domain annotations are available (**zero-shot**), the baseline method using Domain Adaptation (DA) assumes that a small number of target annotations are available (**few-shot**), our method can consistently achieve better performance than the DA method in all six settings. These results based on the OOD generalization

among three datasets suggest that Prompt LP-FT is a highly robust, easy-to-transfer, and convincing method to improve the cross-domain generalization ability of QA models.

5.3 Discussion

We provide discussion to understand better the relative contributions of Prompt LP-FT toward performance improvement, including the universality of our method, the ablation study, and case study.

Universality. The results in *Sec 5.1, 5.2* show that our proposed method improves the OOD generalization performance of various models to varying degrees, with ID performance increasing as well. Experimental results on multiple models demonstrate that our method holds good portability and can benefit variant models, including generative (BART) and discriminative (RoBERTa) models. Results on more backbone models (e.g., T5 and SpanBERT) can be found in appendix A.5.

Ablation Study. Figure 4 shows an ablation study of Prompt-based LP-FT. We find that the combination of prompting methods with LP-FT achieves the best performance in four of six settings, illustrating the advantage of prompt-based LP-FT. In addition, BART+Prompt shows an absolute advantage compared to BART+LP-FT, which can be because prompts benefit the cross-domain QA task by introducing more background knowledge than the adjustment of the tuning strategy. The detailed ablation results are shown in Appendix A.5.

Case Study Table 4 presents a case study of four test samples. For each instance, we show the input context, the prompt sentence, and the output predicted by the baseline method and our method (Prompt LP-FT). It can be seen that the gold an-

Context & Prompt	Question	Gold Answer	Baseline Output	Our Output
Predictive Model: SpanBERT-base				
As Terex has expanded its MP product line, it has captured a larger global market share of the industry, allowing it to gain greater insight into customer demand. This may provide revenue synergies in the future...... Prompt: To answer a "Why useful" question, you need to look for "allowing"	Why Terex has expanded its MP product line is useful?	gain greater insight into customer	it has captured a larger global market share of	it has captured a larger global market share of the industry, allowing it to gain greater insight into customer
However, Avnet's management conceded..... Suppliers get access to a broad range of customers by using Avnet without having to make significant investment in sales and engineering teams. In exchange for these services, Avnet can generate ...... Prompt: The sentiment of this question is positive, you need to look for "significant"	What will be prevented if suppliers get access to a broad range of customers?	investment in sales and engineering	using Avnet	using Avnet without having to make significant investment in sales and engineering
Generative Model: BART-base				
At the end of 2020, the store base had grown about 29% over the prior five-year period, to about 1,920 locations (around 2,100 including Petsense), driving sales and EPS compound annual growth rates over the past three years of 14% and 27%, respectively. We forecast that the firm will grow to around .... Prompt: The entity "EPS" is mentioned in the question. This timing "annual" is mentioned in the question.	Why sales and EPS compound annual growth rates increase?	the store base had grown about 29% over the prior five-year period	14% and 27%	the store base had grown about 29% over the prior five years
Finally, we view the likelihood of sustained economic value creation as quite high for the restaurant brand, which finds itself on the leading edge of most of the changes we expect to impact the restaurant industry over the medium to long term. Though Chipotle saw economic value destruction in 201 ...... Prompt: "restaurant industry" is an important phrase. And also pay attention to these words: "edge", "changes"	What will happen if on the leading edge of most of the changes we expect to impact the restaurant industry?	the likelihood of sustained economic value creation as quite high for the restaurant brand	over the medium to long term	we view the likelihood of sustained economic value creation as quite high for the restaurant brand

Table 4: Case study of "Why" and "What-if" questions answering tasks based on the SpanBERT-base and BART-base models. The Gold Answer is highlighted using the green text, while the Incorrect Answer predicted by the baseline method is highlighted by the red text.
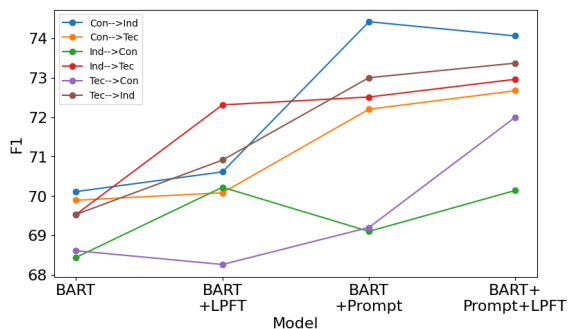

Figure 4: Ablation results based on the BART-base model.

swers are mostly included in the output of Prompt LP-FT, while the output of baseline models is prone to errors. Specifically, baseline models, including SpanBERT-base and BART-base, tend to output the answers closer to the question in the context instead of observing the whole sentence. For example, for the question "*What will be prevented if suppliers ... customers?*", the SpanBERT-base model will output the wrong answer – "*using Avnet*" that is close to the question in the context – while the correct answer – "*investment in sales and engineering*" is ignored. These comparisons provide evidence that our method is beneficial in address-ing the spurious features of sentence order for QA models. This can be because the well-designed prompt combined with LP-FT helps QA models understand the context better.

6 Conclusion

We introduce a zero-shot cross-domain QA task and present a novel Prompt-based LP-FT method by combining prompt with a linear-probing fine-tuning strategy, providing theoretical proof that the proposed method can enhance the model's in-domain performance and out-of-domain generaliz-ability, and empirically showing that the Prompt LP-FT method consistently benefits the QA models. Experimental results show that (1) current meth-ods still have a lag much behind human-level to-wards the cross-domain QA generalization; (2) our method brings larger performance gains for gener-ative models than discriminative models; (3) the use of the prompt-based LP-FT in other NLP tasks is worth trying. Meanwhile, the emergent ability of LLMs will definitely decrease the challenge of the current cross-domain QA setting. Designing challenging datasets of cross-domain QA towards LLMs should be paid more attention in the future.

Limitation

Our method has a few limitations which may inspire future work. First, the prompt templates are manually designed, although we've introduced the rules and intuitions used in our implementation. Second, the proposed method may have low scalability to long text. Because we add the prompt at the end of the context, the prompt would be truncated if the context itself exceeds the maximum acceptable token length of the model.

Ethics Statement

This paper honors the ACL Code of Ethics. Public available datasets are used to establish our results. No private data and crowd-sourcing work are used to produce predictions. The code and data are open-sourced under the CC-BY-NC-SA license.

Acknowledgement

References

Akari Asai, Xinyan Yu, Jungo Kasai, and Hanna Hajishirzi. 2021. One question answering model for many languages with cross-lingual dense passage retrieval. *Advances in Neural Information Processing Systems*, 34:7547–7560.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Yu Cao, Meng Fang, Baosheng Yu, and Joey Tianyi Zhou. 2020. Unsupervised domain adaptation on reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7480–7487.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.

Yu Gu, Sue Kase, Michelle Vanni, Brian Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond iid: three levels of generalization for question answering on knowledge bases. In *Proceedings of the Web Conference 2021*, pages 3477–3488.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.

Alon Jacovi, Swabha Swayamdipta, Shauli Ravfogel, Yanai Elazar, Yejin Choi, and Yoav Goldberg. 2021. Contrastive explanations for model interpretability. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1597–1611.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

Divyansh Kaushik, Amrith Setlur, Eduard Hovy, and Zachary C Lipton. 2021. Explaining the efficacy of counterfactually augmented data. In *International Conference on Learning Representations*.

Tomas Kocisky, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gabor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations (ICLR)*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1000–1008.

Linqing Liu, Patrick Lewis, Sebastian Riedel, and Pontus Stenetorp. 2021a. Challenges in generalization in open domain question answering. *arXiv preprint arXiv:2109.01156*.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019a. Linguistic knowledge and transferability of contextual representations. 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 1073–1094.

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

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

Jinghui Lu, Linyi Yang, Brian Mac Namee, and Yue Zhang. 2022. A rationale-centric framework for human-in-the-loop machine learning. *arXiv preprint arXiv:2203.12918*.

Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Qi Zhang, and Xuanjing Huang. 2021. Template-free prompt tuning for few-shot ner. *arXiv preprint arXiv:2109.13532*.

Andrey Malinin, Neil Band, Yarin Gal, Mark Gales, Alexander Ganshin, German Chesnokov, Alexey Noskov, Andrey Ploskonosov, Liudmila Prokhorenkova, Ivan Provilkov, Vatsal Raina, Vyas Raina, Denis Roginskiy, Mariya Shmatova, Panagiotis Tigas, and Boris Yangel. 2021. Shifts: A dataset of real distributional shift across multiple large-scale tasks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Itzik Malkiel and Lior Wolf. 2021. Maximal multiverse learning for promoting cross-task generalization of fine-tuned language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 187–199.

Stephen L Morgan and Christopher Winship. 2015. *Counterfactuals and causal inference*. Cambridge University Press.

Jong-Hoon Oh, Kentaro Torisawa, Chikara Hashimoto, Ryu Iida, Masahiro Tanaka, and Julien Kloetzer. 2016. A Semi-Supervised Learning Approach to Why-Question Answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

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. *Journal of Machine Learning Research*, 21:1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Priyanka Sen and Amir Saffari. 2020. What do models learn from question answering datasets? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2429–2438, Online. Association for Computational Linguistics.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. 2020. On the theory of transfer learning: The importance of task diversity. *Advances in Neural Information Processing Systems*, 33:7852–7862.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022. Investigating selective prediction approaches across several tasks in iid, ood, and adversarial settings. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1995–2002.

Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can generative pre-trained language models serve as knowledge bases for closed-book qa? 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 3241–3251.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022. Promda: Prompt-based data augmentation for low-resource nlu tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4242–4255.

Zhao Wang and Aron Culotta. 2021. Robustness to spurious correlations in text classification via automatically generated counterfactuals. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14024–14031.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 271–280, Vancouver, Canada. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Sen Wu, Hongyang R Zhang, and Christopher Ré. 2020. Understanding and improving information transfer in multi-task learning. In *International Conference on Learning Representations*.

Linyi Yang, Jiazheng Li, Pádraig Cunningham, Yue Zhang, Barry Smyth, and Ruihai Dong. 2021. Exploring the efficacy of automatically generated counterfactuals for sentiment analysis. 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 306–316.

Linyi Yang, Zhen Wang, Yuxiang Wu, Jie Yang, and Yue Zhang. 2022. Towards fine-grained causal reasoning and qa. *arXiv preprint arXiv:2204.07408*.

Xiang Yue, Ziyu Yao, and Huan Sun. 2022. Synthetic question value estimation for domain adaptation of question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1340–1351.

A Appendix

A.1 Template Comparison

As we see in Table 5, changing "But" in the template to "And" alters the logical relationship between the preceding and following sentences, which had an impact of more than 1% on the performance.

A.2 Template Engineering

The main objective of applying prompt templates is to enhance the model's out-of-domain performance by extracting invariant features between different domain questions. Therefore, the first rule is that a designed template should avoid containing domain-related information. For example, "This [health] company [Hologic] is mentioned in the question." should not be an ideal template because it involves extra domain information that Hologic is a health company.
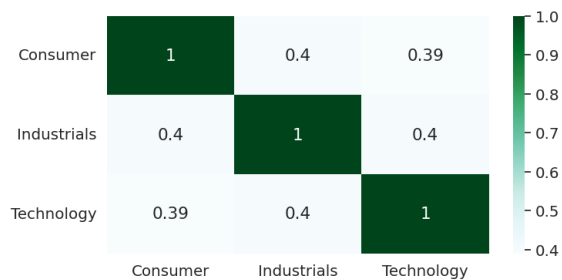

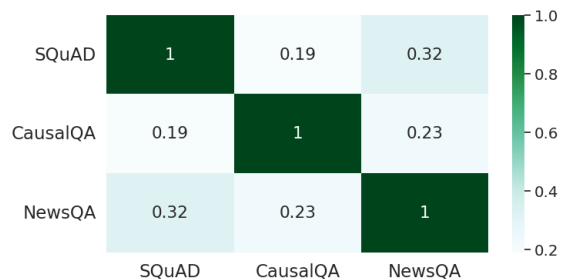Figure 5: The word overlap between different datasets considered by the CasualQA task.

Figure 6: The word overlap between SQuAD, CausalQA and NewsQA datasets.

Second, a template should be a proper English sentence with correct spelling, no grammar mistakes, and proper semantic meaning. Our experiment shows that one wrong word in a template may cause significant performance variation (see Appendix A.1).

Third, since templates are concatenated at the end of the context, templates cannot be too long. If a template has almost the same length as the context or even longer, it will double the amount of input data and thus increase the computational cost of the model; more importantly, it may deprive the leader role of the context, which may make the model too generalized to capture the answers.

Fourth, there are two main varieties of prompt templates: *cloze prompt* and *prefix prompt*.(Liu et al., 2021b) Cloze prompt fill in the blanks of a textual string, and the prefix prompt continue a string prefix. Instead of using only one type, we include both variants of templates in the designed four prompt templates.

According to these rules, we design four types of templates, of which each type has different sentence patterns. Template generalization is modulized as a two-step process: **1)** generating the prompt words, and **2)** filling in the blanks (Liu et al., 2021b).

	Template	F1
Baseline	None	70.29
Experiment1	"There is no important phrase in this query. But also pay attention to these words: __"	69.57
Experiment2	"There is no important phrase in this query. And also pay attention to these words: __"	70.84

Table 5: The effect of using an improper word in a template.

A.3 Template Designing

Sentiment Templates Assume that a person unfamiliar with the restaurant industry tries to answer the question, *"Why the global restaurant sector has come under pressure?"*. This person can easily find that this question concerns the factors that adversely affect restaurants even without industry knowledge. Therefore, looking for negative words from the context, like *destroyed*, *restricted* etc., may help to locate the correct answer. Based on the intuition above, we implement a sentiment analysis framework [*] to give each question and each word in the answer sentence a sentiment score. Afterwards, the highest positive or negative scores are selected to be used as the prompt words. Second, the sentient of the question and the prompt words are filled in the blanks of sentiment templates.

Named Entity Templates Unique entities mentioned in a question could hint at answering the question. Hence, a named entity recognition framework is applied to each question. We intend to recognize five types of entities mentioned in the question: Person, Organization, Location, Country, and Date. Entities not included in the five types are assigned as "Other" entities. Step two fills the recognized entities in the blanks as prompt words.

Phrase Template Phrases are usually the question subject, potentially valuable in locating the correct answer. A simple strategy is designed to find out the phrases composed of an adjective(s) and noun(s). For example, "hybrid environments", "software-as-a-service applications", and "remote access" are phrases in a question. These phrases are selected as prompt words and filled in the blanks in step two.

---
[*]implemented using the NLTK module

Methods	Dev		Test	
	F1	EM	F1	EM
SpanBERT-base	84.77	62.62	84.85	64.04
SpanBERT-large	85.40	61.41	85.53	62.26

Table 6: Domain-independent QA results of SpanBERT-base and SpanBERT-large model.

Domain	Con		Ind		Tech	
	F1	EM	F1	EM	F1	EM
Con	**85.84**	60.64	84.80	56.01	85.54	55.09
Ind	85.76	66.66	**85.84**	67.21	85.34	65.57
Tech	80.24	58.08	81.51	58.58	**81.98**	58.08

Table 7: Out-of-domain test results of SpanBERT-base.

A.4 Word Overlap between Datasets

Fig 5 and 6 show the word overlap percentage between different domains of the CausalQA dataset, and also on datasets from different domains, i.e., between the SQuAD, CausalQA and NewsQA datasets.

A.5 Experiment Results on Other Models

On both domain-independent QA and cross-domain QA tasks, the SpanBERT model achieves state-of-the-art performance. *Table 6* shows the domain-independent QA result of SpanBERT-base and SpanBERT-large model, which also provides evidence that the proposed method works on the large model which can achieve better results than it on the base model. Tab7 shows the result of the Span-BERT OOD test.

Tab 8 shows the cross-domain QA experiment results on T5-base. We show that our method can significantly improve the cross-domain QA performance compared to the standard fine-tuning results based on the CausalQA dataset.

Tab 9, 10 and Fig 7 are the ablation study results on RoBERTa and BART models for cross-domain QA task on the CausalQA dataset.

A.6 Details of experimental results

The experiment is conducted on a GTX 3090 TI with 24GB graphics RAM size. The average training time for each model on the domain-independent QA task is around 2.5 hours, and on the cross-domain QA task is around 30 minutes on CausalQA dataset. On SQuAD and NewsQA dataset, the average training time for each model is around 3 hours. For each experiment setting, we run 10 repeated experiments and report the average results. The model name indicates the base model is no size

Train/Test		Consumer		Industrial		Technology	
		F1	EM	F1	EM	F1	EM
Con	Ori	**59.30**	15.74	56.69	18.60	56.13	17.67
	Ours	(+2.19)	(+3.87)	(+1.79)	(+0.84)	(+2.24)	(+1.77)
Ind	Ori	59.85	24.72	**61.41**	24.03	59.94	25.27
	Ours	(+2.52)	(-0.68)	(+0.06)	(-3.81)	(+1.64)	(-4.51)
Tech	Ori	55.30	17.85	54.41	16.83	**58.76**	20.20
	Ours	(+3.70)	(+5.38)	(+4.83)	(+5.39)	(+0.53)	(+2.02)

Table 8: Out-of-domain test results of the T5-base model. Numbers in brackets represent the performance improved by our method.
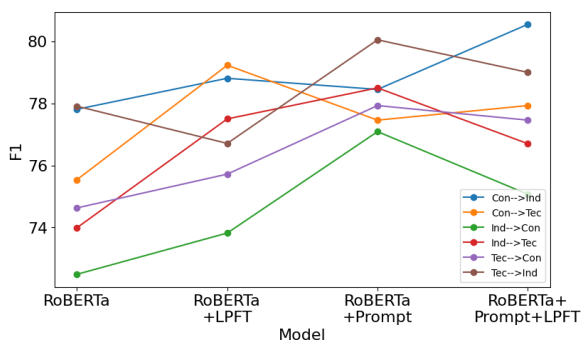

Figure 7: Ablation study results based on the RoBERTa-base model for cross-domain QA.

specification, e.g. "BART+P" indicates the BART-base model plus the prompting method. We also implemented large models to prove the effectiveness of the proposed models.

For the hyperparameter tuning, we split the whole dataset into train/validation/test sets on the domain-independent QA task and use the validation set for hyperparameter tuning. On the cross-domain QA task, we split the dataset of each domain into train/validation/test sets and use the validation set that comes from the same domain with the training set for hyperparameter tuning. The criterion used to select the hyperparameter is the F1 on the validation set. We first select a series of candidate values of a hyperparameter through uniform sampling from a reasonable range, then select the value that achieves the best F1 on the validation set. Three repeated trials decide the value of a hyperparameter. For example, we give the best-performing RoBERTa-base model configuration on *Consumer-Train/Technology-Test* experiment as follows: the learning rate for linear-probing is 10^{-6}, the number of epochs for linear probing is 3, the learning rate for fine-tuning is 10^{-5}, the training batch size is 4, parameters are updated every 8 batches, and the number of epochs for fine-tuning is 14.

Methods		Consumer		Industrial		Technology	
		F1	EM	F1	EM	F1	EM
Baseline: RoBERTa	Con	78.20	51.38	72.49	47.68	74.63	49.07
	Ind	77.81	49.45	80.05	58.46	77.91	48.35
	Tec	75.54	55.05	73.99	48.98	76.49	54.04
RoBERTa + LP-FT	Con	75.09	47.68	73.82	46.75	75.72	50.00
	Ind	78.81	48.90	**81.01***	51.10	76.71	50.00
	Tec	79.23	**56.06***	77.50	55.05	78.22	53.53
RoBERTa + Prompt	Con	**78.97***	53.24	**77.09***	**50.00***	**77.93***	**51.85***
	Ind	78.45	56.28	80.05	**61.74***	**80.05***	**57.92***
	Tec	77.46	54.54	**78.50***	**57.58***	**80.62***	**58.08***
RoBERTa + LP-FT + Prompt	Con	78.39	**53.24***	75.07	48.61	77.46	49.54
	Ind	**80.55***	**59.02***	80.70	60.66	79.00	57.37
	Tec	**77.93***	55.05	76.70	56.06	76.76	53.54

Table 9: Ablation study results based on the RoBERTa-base model. Also, the results are averaged by 10 repeated experiments. The statistically significant performance improvements of our proposed method compared to the baseline model are indicated by $*$ based on the T-test ($P < 0.01$).

Methods		Consumer		Industrial		Technology	
		F1	EM	F1	EM	F1	EM
Baseline: BART	Con	70.29	24.53	68.44	24.07	68.61	23.61
	Ind	70.11	31.31	72.53	32.41	69.53	27.77
	Tec	69.89	30.30	69.53	27.77	71.79	33.83
BART + LP-FT	Con	**70.81***	24.07	**70.22***	24.53	68.26	23.61
	Ind	70.61	30.77	73.39	31.87	70.91	28.57
	Tec	71.08	**35.35**	72.31	30.30	72.37	33.33
BART + Prompt	Con	70.41	**27.31***	69.10	23.15	69.20	23.61
	Ind	**74.42***	**42.07***	76.27	37.16	73.00	34.43
	Tec	72.20	32.32	72.51	30.3	73.32	**35.35***
BART + LP-FT + Prompt	Con	70.65	25.01	70.14	**26.85***	**72.00***	**28.24***
	Ind	74.06	36.06	**76.58***	**39.89***	**73.37***	**37.70***
	Tec	**72.67***	33.33	**72.96***	**31.31***	**74.02***	34.34

Table 10: Ablation study results based on the BART-base model for cross-domain QA. The results are averaged by 10 repeated experiments. The statistically significant performance improvements of our proposed method compared to the baseline model are indicated by $*$ based on the T-test ($P < 0.01$).

A   For every submission:

☑ A1. Did you describe the limitations of your work?
The limitation-related discussion can be found in the section "Limitation".

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Related text can be found in the "Abstract" and "Introduction" sections.

☒ A4. Have you used AI writing assistants when working on this paper?
We have not used AI writing assistants when working on this paper.

B   ☑ Did you use or create scientific artifacts?

We used the Huggingface framework in implementing our model. Related text can be found in the "4.2 Baseline Models" section.

☑ B1. Did you cite the creators of artifacts you used?
Related text can be found in the "4.2 Baseline Models" section.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Related text can be found in the "Ethics Statement" section.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
The usage of the Huggingface framework in this paper was consistent with their intended use. Related text can be found in the "4.2 Baseline Models" section.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
All datasets used in this paper are dominant question-answering datasets created and used by many previous works. Related text can be found in the "4.1 Datasets" section.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Related text can be found in the "4.1 Datasets" section.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Related text can be found in the "4.1 Datasets" section.

C ☑ **Did you run computational experiments?**

Related text can be found in the "A.6 Details of experimental results" section.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Related text can be found in the "4.2 Baseline Models" and "A.6 Details of experimental results" sections.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Related text can be found in the "4.2 Baseline Models" section.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Related text can be found in the "5 Results and Discussion" section.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Related text can be found in the "A.3 Template Designing" section.

D ☒ **Did you use human annotators (e.g., crowdworkers) or research with human participants?**

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.