Prompt-based Conservation Learning for Multi-hop Question Answering

Zhenyun Deng, Yonghua Zhu, Yang Chen, Qianqian Qi
Michael Witbrock, Patricia Riddle
School of Computer Science, University of Auckland, New Zealand
{zden658, yzhu970, beryl413}@aucklanduni.ac.nz
{yang.chen, m.witbrock, p.riddle}@auckland.ac.nz

Abstract

Multi-hop question answering (QA) requires reasoning over multiple documents to answer a complex question and provide interpretable supporting evidence. However, providing supporting evidence is not enough to demonstrate that a model has performed the desired reasoning to reach the correct answer. Most existing multi-hop QA methods fail to answer a large fraction of sub-questions, even if their parent questions are answered correctly. In this paper, we propose the Prompt-based Conservation Learning (PCL) framework for multi-hop QA, which acquires new knowledge from multi-hop QA tasks while conserving old knowledge learned on single-hop QA tasks, mitigating forgetting. Specifically, we first train a model on existing single-hop QA tasks, and then freeze this model and expand it by allocating additional sub-networks for the multi-hop QA task. Moreover, to condition pre-trained language models to stimulate the kind of reasoning required for specific multi-hop questions, we learn soft prompts for the novel sub-networks to perform type-specific reasoning. Experimental results on the HotpotQA benchmark show that PCL is competitive for multi-hop QA and retains good performance on the corresponding single-hop sub-questions, demonstrating the efficacy of PCL in mitigating knowledge loss by forgetting.

1 Introduction

Multi-hop QA is a challenging task with the goals of reasoning over multiple scattered documents to predict an answer, and providing explanatory supporting evidence (Yang et al., 2018). By fine-tuning pre-trained language models (PLMs) with task-specific data, most existing multi-hop QA models have achieved good performance in both goals (Tu et al., 2020; Fang et al., 2020).

Despite the success of fine-tuned PLMs on the multi-hop QA task, providing supporting evidence is not enough to demonstrate that a multi-hop QA model has performed the desired multi-hop reasoning to reach the correct answer; it may instead have utilized reasoning shortcuts, having neglected to acquire and retain the single-hop reasoning knowledge essential to reliable interpretability (Jiang and Bansal, 2019). Previous work (Tang et al., 2021) has demonstrated that most existing multi-hop QA models with good performance fail to answer a large fraction of the sub-questions whose parent multi-hop questions can be answered correctly. Thus, it is necessary to understand the behaviour on each hop of the reasoning process and mitigate forgetting of the knowledge required for each hop in interpretable multi-hop QA. Doing so should enable humans to better trust the QA mechanism.

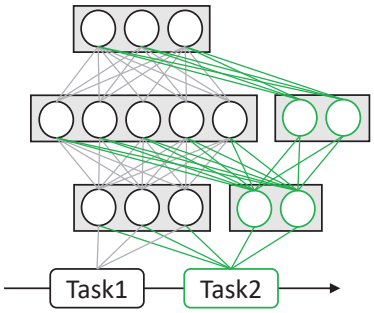

Figure 1: An example of conservation learning based on a continual learning mechanism. The neurons on the left are devoted to Task1 (single-hop QA), and on the right (green) are a novel sub-network created for Task2 (multi-hop QA) that laterally connects to the trained Task1. By adding the sub-network, the model acquires new knowledge of Task2 while retaining knowledge learned in Task1, mitigating forgetting.

In addition, existing QA models integrate all the knowledge by thoroughly pre-training the PLMs on all available data (Schwartz et al., 2020), which integrates the various forms of knowledge from multiple types of questions. However, a downstream QA task may only require knowledge of a specific type. For example, in the multi-hop QA task (Yang et al., 2018), questions can be roughly divided into two different types: bridging and comparison, each of which requires a specific reasoning strategy to answer. To achieve multi-hop reasoning efficiently,

Proceedings of the 29th International Conference on Computational Linguistics, pages 1791–1800
October 12–17, 2022.

it may be useful for PLMs to disentangle knowledge from other question types and stimulate the appropriate reasoning types required for particular multi-hop questions.

To address these issues, we propose Prompt-based Conservation Learning (PCL) for multi-hop QA. Specifically: i) to train a multi-hop QA model without forgetting, we apply conservation learning based on a continual learning mechanism to acquire new knowledge from multi-hop QA tasks while retaining that previously learned on single-hop QA tasks. As shown in Figure 1, we first train a model on the single-hop QA task; when incorporating the new multi-hop QA task, we freeze the model trained on the single-hop task and expand it by allocating novel sub-networks for new multi-hop knowledge; ii) to take full advantage of diverse knowledge in the PLM, we first identify the reasoning type of the multi-hop question as a soft prompt via a transformer-based question classifier, and then transform it into a sub-network that connects laterally with the previously trained QA model, to condition the PLM to perform type-specific reasoning. Since PCL trains the QA model incrementally based on the conserved previously learned parameters, it should be able to perform well on multi-hop QA because it thus retains the previously learned knowledge (Parisi et al., 2019; Sun et al., 2020).

Our contributions are summarized as follows:

- We propose conservation learning for multi-hop QA, which acquires knowledge from the multi-hop QA task while retaining knowledge learned on single-hop QA tasks, which may enable humans to understand the behaviour of each hop in the reasoning process better.
- We propose using a soft prompt based on the reasoning type to condition the PLM, stimulating use of the required knowledge for particular types of multi-hop reasoning.
- Our proposed PCL achieves better performance on the HotpotQA leaderboard, while also retaining good performance on the corresponding single-hop sub-questions.

2 Related Work

Prompt Tuning for PLMs. Prompt tuning is an effective mechanism for learning prompts to condition PLMs to stimulate and apply the appropriate knowledge for a specific downstream task (Liu et al., 2021). Gu et al. (2021) propose to initialize soft prompts by adding them into the pre-training stage of few-shot learning. Li and Liang (2021) prepend a series of learnable continuous embeddings as soft prompts into the input, achieving better performance in text generation tasks. Motivated by these methods, we use the reasoning types of multi-hop questions as soft prompts to condition PLMs to stimulate the knowledge required to answer multi-hop questions.

Continual Learning for PLMs. Continual learning aims to allow systems to repeatedly acquire new knowledge while retaining previously learned experience, mitigating catastrophic forgetting (Parisi et al., 2019). Conceptually, continual learning can be divided into three categories of technique: i) retrain the whole model while imposing additional constraints to retain the important learned model parameters from previous tasks (Li et al., 2021a); ii) perform memory replay to distill the knowledge from previous model backups (Sun et al., 2019; Rolnick et al., 2019); iii) freeze the model trained on previous tasks and retrain the model by allocating new neurons or network layers for new tasks (Qin et al., 2022). In this paper, we propose a learning mechanism based on continual learning, by freezing the model trained on the single-hop QA, and retraining the model for the multi-hop QA using our soft-prompt technique, enables the QA model to achieve single-hop reasoning and multi-hop reasoning simultaneously. Since we only have two tasks, we call this conservation learning. It aims to conserve previously learned knowledge while performing well on a second task; it does not continue for a large number of tasks as in continual learning.

End-to-end Multi-hop QA. Existing end-to-end multi-hop QA systems predict the answer and corresponding supporting facts based on the given question and retrieved relevant paragraphs. Qiu et al. (2019), Fang et al. (2020), and Tu et al. (2020) extract information at different levels of granularity as nodes in a graph, and then apply GNN-based methods to answer the question and provide supporting sentences. Shao et al. (2020a), Beltagy et al. (2020) and Wu et al. (2021) argue that graph structures may not be necessary for multi-hop QA, and propose graph-free reasoning models. Unlike these methods, where there is no training requirement for the models to follow the desired reasoning steps to predict the answer, we propose a multi-hop

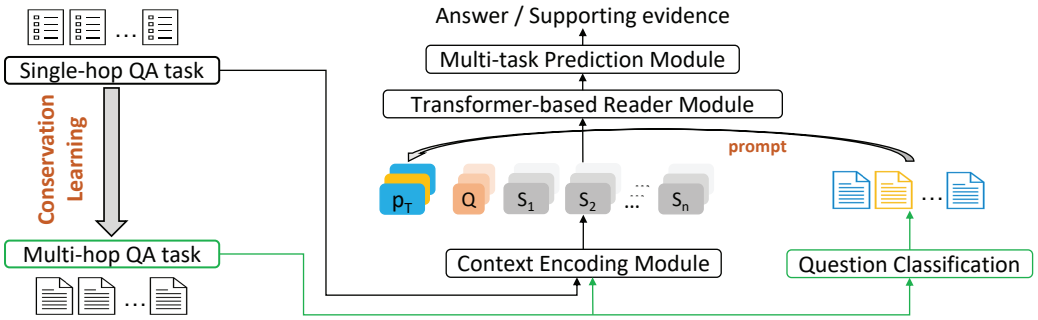

Figure 2: An overview of our proposed PCL framework for multi-hop QA. Specifically, it involves three key steps, (a) train a QA model to acquire knowledge from single-hop QA tasks; (b) identify reasoning types of multi-hop questions as soft prompts, and transform soft prompts into a sequence of continuous type-specific vectors; (c) retrain the QA model to acquire new knowledge from multi-hop QA tasks by freezing the trained network in single-hop QA task and prepending soft prompt vectors to the input.

QA framework with separated learning of the intended behaviour of QA models on each hop of the reasoning process and in the final answer.

3 Methodology

3.1 Overview

This section, we describe prompt-based conservation learning for multi-hop QA. As illustrated in Figure 2, our PCL consists of three components: i) we first acquire single-hop QA knowledge by explicitly training on these tasks; ii) we then acquire knowledge for the new multi-hop QA task while retaining the learned knowledge using conservation learning; iii) we perform type-specific reasoning, identifying the reasoning type of the question via the soft prompt to stimulate application of the appropriate knowledge.

3.2 Single-hop QA

To understand the behaviour of existing QA models on each hop of the reasoning process, we train a QA model based on the PLM, ELECTRA (Clark et al., 2020) on a single-hop QA task, SQuAD (Rajpurkar et al., 2016). This QA model contains two modules: context encoding and a transformer-based reader.

Context Encoding. Given a question Q and n relevant sentences, we concatenate the question and sentences into an input sequence for the pre-trained ELECTRA encoder to obtain a context representation. Specifically, we formulate the input sequence as "[CLS] Q [SEP] yes no [SEP] [SE] s_1 [SEP] [SE] s_2 [SEP] ... [SE] s_i [SEP]... [SE] s_n [SEP]", where [SE] is a special token delineating supporting evidence, and yes no indicates a yes/no answer, which are prepended to the context, subsequently encoded by ELECTRA into the context representation. Consequently, each context sentence s_i in the input sequence can interact with other sentences across the concatenated sequence by using a self-attention mechanism; such interactions are crucial for multi-hop QA (Zhu et al., 2021).

Transformer-based Reader. After context encoding, the context representations are passed through a bi-attention layer to enhance interactions between the question and the context (Qiu et al., 2019). On top of the updated context representation, we have followed (Fang et al., 2020) to design a multi-task prediction module to jointly perform answer and supporting evidence prediction. For answer span prediction, we use two linear layers applied to the context representation to predict the start and end position of the answer. For supporting evidence prediction, we use a binary linear layer to predict a binary relevance label at each sentence start [SE]. The final objective is defined as:

$$\mathcal{L}_{Joint} = \mathcal{L}_{start} + \mathcal{L}_{end} + \lambda_1 \mathcal{L}_{SE}$$

where λ_1 is a hyper-parameter and each loss function $\mathcal{L}$ is the cross-entropy loss between the prediction and ground truth.

3.3 Multi-hop QA with Conservation Learning

Origins in Continual Learning. In one form of Continual Learning, given N existing tasks $\mathcal{T}_{seq} = \{\mathcal{T}_0, \mathcal{T}_1, ..., \mathcal{T}_N\}$, when a new task $\mathcal{T}_{N+1}$ comes, an additional network is created and the lateral connections with the trained model are learned. To avoid knowledge forgetting, the parameters θ^N learned by the existing tasks $\mathcal{T}_{seq}$ remain unchanged while the new parameter set θ^{N+1} is learned for the additional network in Task $\mathcal{T}_{N+1}$ (Parisi et al., 2019).

Conservation Learning for Multi-hop QA. To enable a trained single-hop QA model to learn the

new knowledge required for a subsequent multi-hop QA task without forgetting previously learned knowledge, we propose a truncated-continual-learning-like method that freezes the learned model and allocates additional sub-networks for the new multi-hop QA tasks. In principle this process could be iterated in continual learning, but here we apply one such step, and coin the term "conservation learning" to describe it. PCL's multi-hop QA after conservation learning consists of three components: i) question classification: identifying the reasoning type of the multi-hop question; ii) paragraph selection: retrieving paragraphs related to the multi-hop question; iii) pre-trained soft prompt: conditioning a PLM to perform the type-specific reasoning required for a multi-hop question.

Question Classification. Instead of training a separate QA model for each reasoning type, our design uses a single PLM to integrate the knowledge from all reasoning types. To inform this use, we first need to identify the reasoning type of the multi-hop question. Thus, we train a question classifier, also based on ELECTRA, followed by a binary classification layer, to predict the reasoning type for each multi-hop question. The question classifier only takes the question as its input and outputs a relevance score for different reasoning types. The reasoning type with the highest score is selected as the type of multi-hop question.

Iterative Paragraph Selection. Since not every given paragraph contains relevant information, multi-hop QA models must filter out irrelevant paragraphs. In addition, multi-hop questions also often permit reasoning shortcuts through which QA models can directly locate the final answer by word-matching the question to a single sentence in the paragraph (Qi et al., 2019, 2020). To discourage this kind of direct but unjustified leap to the answer, we propose to retrieve paragraphs related to the question in an iterative fashion, which encodes the question and previously retrieved paragraphs as a new question vector to retrieve the next relevant paragraph. For simplicity, we use the same model encoder as the question classifier to select relevant paragraphs, except that we take the question q and the paragraph p as the input and output a relevance score for each paragraph. We calculate the score for each paragraph at each retrieval step as follows:

$$\mathcal{P}(P_{seq}|q) = \prod_{t=1}^{n} \mathcal{P}(p_t|q, p^1, p^2, ..., p^{t-1})$$

where for $t = 1$ (*i.e.,* the first hop), we only use the original question q for paragraph retrieval. At each subsequent retrieval step, we encode the question q and the most relevant paragraph p^{t-1} in the previous step t as a new question vector to predict the next relevant paragraph. In this way, each subsequent retrieved paragraph is not only related to the question, but also related to the previous retrieved paragraphs, which discourages producing an answer using "reasoning" shortcuts and provides a solid basis for multi-hop reasoning in the next step.

Pre-training Soft Prompt. To enable the PLM to integrate knowledge from multiple reasoning types, we introduce a soft prompt based on the reasoning type to condition the PLM to perform type-specific reasoning, which is connected laterally to the trained QA model during training. Specifically, we first formulate the input sequence as "[CLS] Q [SEP] yes no [SEP] [SE] s_1^1 [SEP] [SE] s_1^2 [SEP] ... [SE] $\mathrm{s}_\mathrm{i}^\mathrm{j}$ [SEP]... [SE] $\mathrm{s}_\mathrm{n}^\mathrm{m}$ [SEP]", where $\mathrm{s}_\mathrm{i}^\mathrm{j}$ indicates the j-th sentence in the relevant paragraph i; we then utilize the previously trained model to initialize the input sequence to obtain the context representation $\mathbf{C} = \{\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_{n-1}\} \in \mathbb{R}^{n \times d}$, where n, d are the length and the dimension of the context, respectively; we finally transform the reasoning type obtained in the question classification into a continuous trainable vector $\mathbf{p} \in \mathbb{R}^{m \times d}$ and prepend it onto $\mathbf{C}$, resulting in the new input $\mathbf{C}' = \{\mathbf{p}_i; \mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_{n-1}\}$, where m is the length of the soft prompt and $\mathbf{p}_i$ is the soft prompt vector of reasoning type i.

Once the new context representation is obtained, it is then processed by the transformer-based reader module. Notably, we optimize $\mathbf{p}_i$ along with other parameters of the PLM during pre-training. During fine-tuning, we prepend the trained soft prompt vector into the input sequence, guiding the model to perform type-specific reasoning. In this way, we condition the PLM to stimulate the proper knowledge required for multi-hop reasoning.

4 Experiments

4.1 Dataset and Metrics

We evaluate our model primarily on three datasets: HotpotQA (Yang et al., 2018), adversarial HotpotQA (Jiang and Bansal, 2019) and a manually verified sub-question QA dataset generated from HotpotQA (Tang et al., 2021). To verify whether our PCL can be generalized to other multi-hop QA

Model	Ans		Sup		Joint	
	EM	F1	EM	F1	EM	F1
Baseline Model (Yang et al., 2018)	45.60	59.02	20.32	64.49	10.83	40.16
DecompRC (Min et al., 2019)	55.20	69.63	-	-	-	-
OUNS (Perez et al., 2020)	66.33	79.34	-	-	-	-
QFE (Nishida et al., 2019)	53.86	68.06	57.75	84.49	34.63	59.61
DFGN (Qiu et al., 2019)	56.31	69.69	51.50	81.62	33.62	59.82
SAE-large (Tu et al., 2020)	66.92	79.62	61.53	86.86	45.36	71.45
C2F Reader (Shao et al., 2020b)	67.98	81.24	60.81	87.63	44.67	72.73
Longformer (Beltagy et al., 2020)	68.00	81.25	63.09	88.34	45.91	73.16
HGN-large (Fang et al., 2020)	69.22	82.19	62.76	88.47	47.11	74.21
AMGN (Li et al., 2021b)	70.53	83.37	63.57	88.83	47.77	75.24
S2G (Wu et al., 2021)	70.72	83.53	64.30	88.72	48.60	75.45
PCL (Ours)	**71.76**	**84.39**	**64.61**	**89.20**	**49.27**	**76.56**

Table 1: Results on the blind test set of HotpotQA in the distractor setting. Our PCL achieves the best performance on the HotpotQA leaderboard. "-" denotes the case where no results are available. Leaderboard: https://hotpotqa.github.io/.

datasets, we also conduct experiments on two similar datasets: 2WikiMultihopQA (Ho et al., 2020) and MuSiQue (Trivedi et al., 2021). Unlike other knowledge-based multi-hop QA datasets (Welbl et al., 2018; Talmor and Berant, 2018; Saxena et al., 2020) that restrict the final answer to the content of explicit knowledge bases, all QA pairs in the HotpotQA are collected from Wikipedia.

HotpotQA. Each multi-hop question is provided with ground truth answers and supporting sentences, which enables us to evaluate the performance and interpretability of multi-hop reasoning. There are two reasoning types of questions: bridging and comparison, each of which requires a specific reasoning strategy to answer.

Sub-question QA dataset. To analyze whether the multi-hop QA models really perform each hop of the reasoning process, Tang et al. (2021) generate a single-hop sub-question dataset with 1000 manually verified samples for the dev set of HotpotQA for evaluation.

Adversarial HotpotQA. Multi-hop questions in the HotpotQA often contain reasoning shortcuts through which models can directly find the answer by word-matching the question to a sentence. To avoid this, Jiang and Bansal (2019) construct adversarial samples by creating contradicting answers to reasoning shortcuts without affecting the validity of the original answers.

Multi-hop QA Dataset. Unlike HotpotQA, 2WikiMultihopQA evaluates the interpretability of the multi-hop QA model not only with supporting evidence, but also with entity-relation tuples. However, for a fair comparison, we do not use the entity-relation tuples in our training. MuSiQue has richer multi-hop questions with 2-4 hops.

Metrics. We use Exact Match (EM) and Partial Match (F1) to evaluate the model performance on answer and supporting facts prediction, and a joint EM and F1 score to evaluate the final performance.

4.2 Implementation Details

We adopt ELECTRA-large (Clark et al., 2020) as the skeleton for each module. Our released implementation is based on Huggingface (Wolf et al., 2020). For question classification and paragraph selection, we train the models for 5 epochs using Adam optimizer, with a batch size of 12, a learning rate of 2×10^{-5}, a warm-up rate of 0.1 and ℓ_2 weight decay of 0.01. For question answering, we use the same setting as stated above, except for a learning rate of 3×10^{-5} and an additional prompt length of 2 tokens. The hyper-parameter of λ_1 is set to 2. Only the context encoding module is frozen during Conservation Learning and additional weights are added to connect the soft prompts.

4.3 Main Results

We compare our PCL model with other published baselines on the test set of HotpotQA in the distractor setting. As shown in Table 1, we observe that our PCL QA-system outperforms all comparison baselines on every metric and achieves the best performance on the HotpoQA dataset, demonstrating the progress made by PCL in multi-hop QA. Specifically, under the same setting, using a transformer-based ELECTRA model, PCL achieves a 1.12/0.91 improvement on the Joint EM/F1 score, compared with the best graph-free model S2G. This indicates that the effectiveness of the proposed conservation learning and soft prompts. For the best graph-based

q	q_{sub1}	q_{sub2}	DFGN	DecompRC	HGN	PCL
c	c	c	23	31.3	39.5	43.6
c	c	w	9.7	7.2	5.1	6.8
c	w	c	17.9	19.1	19.6	21.3
c	w	w	7.5	5.5	3.8	2.1
w	c	c	4.9	3	2.8	1.7
w	c	w	17	18.6	16.7	16.3
w	w	c	3.5	3.4	2.6	1.1
w	w	w	16.5	11.9	9.9	7.1

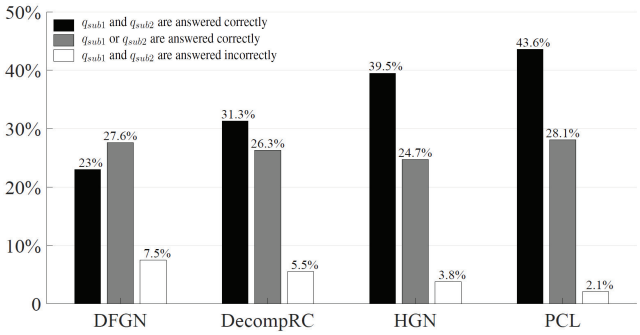

Table 2: (Left) Categorical EM statistics (%) of sub-question evaluation for four multi-hop QA models. c/w denotes that the question is answered correctly/wrongly. For example, the first four rows show the percentage of multi-hop questions that can be correctly answered. (Right) The success rate of four multi-hop QA models.

Model	Ans F1	Sup F1	Joint F1
ELECTRA	81.05	89.97	73.89
- Prompt	82.06	90.36	75.02
- CL	82.99	90.97	76.39
PCL	**84.42**	**91.15**	**77.76**

Table 3: Ablation Study of PCL on the dev set of HotpotQA. Prompt denotes that a soft prompt is used to condition PLM ELECTRA to stimulate the reasoning required for the multi-hop question. CL denotes that conservation learning is used to perform multi-hop reasoning. PCL used both soft prompts and conservation learning.

Model	Accuracy
DecompRC	70.40
QC(ELECTRA-large)	**98.97**

Table 4: The performance of question classification by different models. QC(ELECTRA-large) is a question classifier based on ELECTRA-large.

model AMGN, PCL improves the Joint EM/F1 score by 1.5/1.32, which shows that good performance can be achieved without constructing a graph. In the next section, we provide a detailed analysis to evaluate the performance of conservation learning and soft prompts in our PCL model.

4.4 Ablation Studies

To verify the effect of the components in our PCL model, we perform the following ablation studies on the dev set of HotpotQA.

Effect of Conservation Learning (CL). To verify the effect of conservation learning on multi-hop QA, we compare performance with the PLM ELECTRA with and without conservation learning. For conservation learning, we first trained an ELECTRA-based QA model on the single-hop QA dataset SQuAD (Rajpurkar et al., 2016), and then retrained it on the HotpotQA dataset with conservation learning. As shown in Table 3, we observe that the overall performance (F1 score) increased from 73.89 to 76.39 after using conservation learning, which shows that our model performs well on multi-hop reasoning when the previously learned knowledge is retained. In the following Section 4.6, we provide an in-depth analysis on the performance of our model on the sub-questions, to compare the ability of models to mitigate forgetting.

Effect of Soft Prompts. To verify the effect of the soft prompt and perform type-specific reasoning, we first identified the reasoning type of the multi-hop question using a classifier based on ELECTRA. In Table 4, our classifier $\text{QC}_{\text{ELECTRA}}$ achieves good accuracy compared to DecompRC, providing a solid basis for type-specific multi-hop reasoning. Then, we transform the identified reasoning type into a soft prompt to stimulate the PLM to perform the corresponding type of multi-hop reasoning. In Table 3, we implant the soft prompt both into the baseline ELECTRA and the ELECTRA based on conservation learning (PCL), the Joint F1 score improved by 1.23 and 1.37, respectively. This suggests that the soft prompt based on the reasoning type can stimulate the question-type-specific reasoning knowledge required for multi-hop QA.

Effect of Pre-trained Language Model. To verify the effects of PLMs, we compare PCL with HGN based on the same data and backbone. As shown in Table 5, PCL outperforms HGN on all metrics. This indicates the effectiveness and robustness of PCL across PLMs.

4.5 Evaluation across Reasoning Types

We evaluate the performance of PCL for multi-hop questions with multiple reasoning types. Specifically, we follow HGN in splitting the multi-hop questions into three categories: bridge, comparison-yes/no and comparison-span. "Bridge" questions require identifying a bridge entity to infer the answer, "comparison-yes/no" and "comparison-span"

Model	Ans F1	Sup F1	Joint F1
HGN(RoBERTa)	82.22	88.58	74.37
HGN(ELECTRA)	82.24	88.63	74.51
HGN(ALBERT)	83.46	89.2	75.79
PCL(RoBERTa)	84.33	90.75	77.12
PCL(ELECTRA)	84.42	91.15	77.76
PCL(ALBERT)	85.47	91.28	78.76

Table 5: Results with different PLMs on the dev set of HotpotQA. RoBERTa, ELECTRA and ALBERT denote that we use RoBERTa-large, ELECTRA-large and ALBERT-xxlarge-v2 as the PLM respectively.

Model	Question	Ans F1	Sup F1	Joint F1
HGN	bridge	81.90	87.60	73.31
	comp-yn	93.45	94.22	88.5
	comp-span	79.06	91.72	74.17
PCL	bridge	85.36	90.77	78.17
	comp-yn	93.67	94.73	88.93
	comp-span	82.42	92.65	77.57

Table 6: Results with different reasoning types on the dev set of HotpotQA. PCL outperforms HGN in all reasoning types.

require comparing two entities to infer the answer that could be yes/no or a span of text. As shown in Table 6, our PCL performs better than HGN for all reasoning types, indicating that the performance of the model can be effectively improved by using soft prompts for type-specific reasoning.

4.6 Evaluation of Robustness

In this section, we evaluate the robustness and generalization of PCL on three different datasets.

Evaluation on Sub-question Dataset. To analyze whether existing multi-hop QA models could at least in principle perform the multi-hop reasoning process by composing an answer out of solved sub-questions, we perform an evaluation on 1000 human-verified examples (Tang et al., 2021). These data consist of 1000 multi-hop questions q, and the corresponding 1000 sub-questions q_{sub1}, q_{sub2}. EM and F1 are used in each case to evaluate performance on answer prediction. As shown in Table 7, PCL achieves the best performance on the 1000 human-verified examples. Compared to DFGN and DecompRC, whose performance significantly drops on sub-questions, especially on the second sub-questions. PCL dropped by only 2.4 on average, which demonstrates that PCL can in principle support the expected behaviour on each hop of the reasoning process better than other multi-hop QA models by mitigating knowledge forgetting.

To further analyze whether models effectively mitigate knowledge forgetting, we collect the correctness statistics on each example in the sub-question dataset. As shown in Table 2 (Left), PCL has a 96.25% chance of getting the parent multi-hop question q right when both sub-questions q_{sub1} and q_{sub2} are answered correctly, which indicates that PCL can better retain the learned knowledge, through its use of conservation learning, compared with other multi-hop QA models. However, we observe that PCL still has a high probability of answering the parent multi-hop question correctly when only one of the sub-question is answered correctly. We summarize the sub-question dependent success rate of multi-hop QA models in Table 2 (Right). We observe that these models can answer parent multi-hop questions with a high probability (exceeding 20%) when only one sub-question is answered correctly, which indicates that using potentially unsound reasoning shortcuts to predict answers is a common and difficult to avoid phenomenon in multi-hop QA.

Model	q		q_{sub1}		q_{sub2}	
	EM	F1	EM	F1	EM	F1
DFGN	58.1	71.96	54.6	68.54	49.3	60.83
DecRC	63.1	77.61	61.0	75.21	56.8	70.77
HGN	71.0	84.25	66.1	81.72	66.7	78.24
PCL	**73.8**	**87.15**	**68.4**	**83.62**	**68.5**	**81.07**

Table 7: Results on the sub-question dataset with different multi-hop QA models. q denotes the multi-hop question, q_{sub1} and q_{sub2} denote the corresponding sub-questions of q.

Train	Reg		Reg	
Eval	Reg		Adv	
Model	EM	F1	EM	F1
HGN	47.31	74.37	41.56	69.81
PCL	49.59	77.76	47.87	74.24

Table 8: EM and F1 scores after evaluating on the adversarial dataset designed to probe for the use of unsound reasoning shortcuts. Reg or Adv denotes training or evaluating the model on the standard or adversarial HotpotQA dataset.

Evaluation on Adversarial Dataset. To compare the extent to which models are currently able to avoid the unsound-reasoning-shortcut problem, we conducted an adversarial evaluation on the dev set of HotpotQA, reported in Table 8. In the adversarial examples, the fake answers are sampled from the original HotpotQA dataset, but do not affect the validity of the original answers. As shown in Table 8, we trained PCL and HGN on the standard training data and evaluated them on both the standard and adversarial dev data. The result shows that PCL achieves better performance than HGN, indicating that PCL is more robust than HGN against the use of shortcuts probed by the adversarial dataset.

Evaluation on Other Multi-hop Datasets. To verify whether PCL can generalize to other multi-

Question	Answer	Answer pred by PCL	Answer pred by HGN
(Bridge) Q1: Who directed the film about the living funeral for Morrie Schwartz?	Mick Jackson	Mick Jackson	Mick Albom
$Q1_{sub1}$: Which film is about the living funeral for Morrie Schwartz?	Tuesdays with Morrie	Tuesdays with Morrie	Tuesdays with Morrie
$Q1_{sub2}$: Who directed Tuesdays with Morrie?	Mick Jackson	Mick Jackson	Mick Jackson
(Comp) Q2: Are local H and For Against both from the United States?	Yes	Yes	Illinois
$Q2_{sub1}$: Where does Local H from?	Illinois	Illinois	Illinois
$Q2_{sub2}$: Where does For Against from?	Nebraska	Nebraska	Nebraska

Question: What was the job of the character Jack Nicholson played in a 1992 French-American biographical crime film directed by Danny DeVito?
Answer: Teamsters leader

Supporting fact1: Jack Nicholson plays Hoffa, and DeVito plays Robert Ciaro, an amalgamation of several Hoffa associates over the years.
Supporting fact2: **Hoffa** is a 1992 French-American biographical crime film directed by Danny DeVito and written by David Mamet, based on the life of **Teamsters leader** Jimmy Hoffa.
Adversarial fact: **Sweet Revenge** is a 1992 French-American biographical crime film directed by Danny DeVito and written by David Mamet, based on the life of **Dandy** Jimmy Hoffa.

Answer predicted by PCL: Teamsters leader
Answer predicted by HGN: Dandy

Figure 3: Case studies of the sub-question evaluation and adversarial multi-hop question evaluation. The upper case study indicates that our PCL has stronger composite reasoning ability compared to HGN. The lower case study indicates that the iterative paragraph selection is help to avoid predict the answer by using reasoning shortcuts.

	2WikiMultihopQA		MusiQue	
	EM	F1	EM	F1
HGN	38.74	68.69	39.42	65.12
PCL	46.03	73.42	41.28	67.34

Table 9: Results of PCL and HGN on 2WikiMultihopQA and MusiQue multi-hop QA dataset.

hop QA datasets, we compared PCL against HGN on the 2WikiMultihopQA and MuSiQue dataset. In Table 9 we observe that PCL outperforms HGN on these two datasets, which demonstrates PCL's good potential on generalisation to QA problems with more than 2 hops.

4.7 Case Study

We present two case studies in Figure 3. The upper case illustrates the results of PCL and HGN at each hop of the reasoning process. We observe that PCL correctly answered the bridge question Q1, while HGN did not, when all sub-questions were answered correctly, supporting the claim that PCL learns new QA knowledge while retaining knowledge learned for sub-questions. Similarly, for comparison question Q2, PCL learned the specific reasoning ability based on the reasoning type to which Q2 belongs, indicating soft prompts based on reasoning types can elicit the reasoning knowledge required for multi-hop questions.

The lower case illustrates the results of PCL and HGN on an adversarial multi-hop question. In the example, the question can be directly answered by matching a reasoning shortcut in supporting facts2 *"a 1992 French-American biographical crime film directed by Danny DeVito"*. To avoid it, we follow (Jiang and Bansal, 2019) to construct an adversarial fact from the candidate paragraphs by replacing the subject and the answer, *e.g.,* "*Sweet Revenge*" for "*Hoffa*" and "*Dandy*" for "*Teamsters leader*". We observed that PCL correctly answered the question despite the interference from the adversarial fact, while HGN did not. This supports the claim that the iterative paragraph selection helps establish connections between supporting facts, because PCL selects the next supporting fact2 based on the previous supporting fact1. In this example, the adversarial fact is irrelevant to supporting fact1, so PCL excludes it during paragraph selection.

5 Conclusions and Future Work

In this paper, we introduce a novel prompt-based conservation learning framework for multi-hop QA – a framework that retains knowledge from previous component tasks – able to answer questions in a principled way that matches human expectations by answering sub-questions and integrating the answers. By developing soft prompts related to reasoning types during training, we also show that we can condition PLMs to stimulate and apply the reasoning knowledge required for specific multi-hop questions. Experimental results on multiple multi-hop QA datasets demonstrate the improved performance of PCL over previous multi-hop QA models in multi-hop QA.

Next, we plan to extend PCL on QA problems with arbitrary hop-counts, and to increase generality by extending soft prompts to handle QA with unrestricted numbers of, and implicit, reasoning types, and non-linear reasoning structures.

6 Acknowledgments

This work was supported by a grant from the New Zealand Tertiary Education Commission and by the Strong AI Lab at the University of Auckland.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *EMNLP*, pages 8823–8838.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2021. Ppt: Pre-trained prompt tuning for few-shot learning. *arXiv preprint arXiv:2109.04332*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop qa. *arXiv preprint arXiv:1906.07132*.

Haoran Li, Aditya Krishnan, Jingfeng Wu, Soheil Kolouri, Praveen K Pilly, and Vladimir Braverman. 2021a. Lifelong learning with sketched structural regularization. In *ACML*, pages 985–1000. PMLR.

Ronghan Li, Lifang Wang, Shengli Wang, and Zejun Jiang. 2021b. Asynchronous multi-grained graph network for interpretable multi-hop reading comprehension. In *IJCAI*, pages 3857–3863.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

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

Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Multi-hop reading comprehension through question decomposition and rescoring. *arXiv preprint arXiv:1906.02916*.

Kosuke Nishida, Kyosuke Nishida, Masaaki Nagata, Atsushi Otsuka, Itsumi Saito, Hisako Asano, and Junji Tomita. 2019. Answering while summarizing: Multi-task learning for multi-hop qa with evidence extraction. *arXiv preprint arXiv:1905.08511*.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. 2019. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71.

Ethan Perez, Patrick S. H. Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *EMNLP*, pages 8864–8880.

Peng Qi, Haejun Lee, Oghenetegiri Sido, Christopher D Manning, et al. 2020. Answering open-domain questions of varying reasoning steps from text. *arXiv preprint arXiv:2010.12527*.

Peng Qi, Xiaowen Lin, Leo Mehr, Zijian Wang, and Christopher D Manning. 2019. Answering complex open-domain questions through iterative query generation. *arXiv preprint arXiv:1910.07000*.

Yujia Qin, Jiajie Zhang, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2022. Elle: Efficient lifelong pre-training for emerging data. *arXiv preprint arXiv:2203.06311*.

Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. In *ACL*, pages 6140–6150.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. *Advances in Neural Information Processing Systems*, 32.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *ACL*, pages 4498–4507.

Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. Green ai. *Communications of the ACM*, 63(12):54–63.

Nan Shao, Yiming Cui, Ting Liu, Shijin Wang, and Guoping Hu. 2020a. Is Graph Structure Necessary for Multi-hop Question Answering? In *EMNLP*, pages 7187–7192, Online. Association for Computational Linguistics.

Nan Shao, Yiming Cui, Ting Liu, Shijin Wang, and Guoping Hu. 2020b. Is graph structure necessary for multi-hop question answering? In *EMNLP*, pages 7187–7192.

Fan-Keng Sun, Cheng-Hao Ho, and Hung-Yi Lee. 2019. Lamol: Language modeling for lifelong language learning. *arXiv preprint arXiv:1909.03329*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *AAAI*.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

Yixuan Tang, Hwee Tou Ng, and Anthony Tung. 2021. Do multi-hop question answering systems know how to answer the single-hop sub-questions? In *ACL*, pages 3244–3249, Online. Association for Computational Linguistics.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2021. Musique: Multi-hop questions via single-hop question composition. *arXiv preprint arXiv:2108.00573*.

Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. In *AAAI*, pages 9073–9080.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302.

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 *EMNLP*, pages 38–45.

Bohong Wu, Zhuosheng Zhang, and Hai Zhao. 2021. Graph-free multi-hop reading comprehension: A select-to-guide strategy. *arXiv preprint arXiv:2107.11823*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Yunchang Zhu, Liang Pang, Yanyan Lan, Huawei Shen, and Xueqi Cheng. 2021. Adaptive information seeking for open-domain question answering. *arXiv preprint arXiv:2109.06747*.