Reasoning over Hybrid Chain for Table-and-Text Open Domain Question Answering

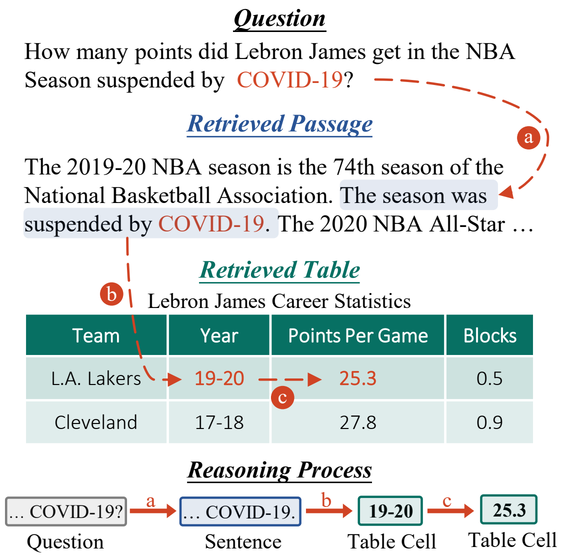
Wanjun Zhong[1†*] , **Junjie Huang**[3*†] , **Qian Liu**[3†] , **Ming Zhou**[4]
, **Jiahai Wang**[1] , **Jian Yin**[1] and **Nan Duan**[2]
[1] The School of Computer Science and Engineering, Sun Yat-sen University
[2] Microsoft Research Asia
[3] Beihang University
[4] Langboat Technology
{zhongwj25@mail2, wangjiah@mail, issjyin@mail}.sysu.edu.cn
{huangjunjie, qian.liu}@buaa.edu.cn
nanduan@microsoft.com, zhouming@chuangxin.com

Abstract

Tabular and textual question answering requires systems to perform reasoning over heterogeneous information, considering table structure, and the connections among table and text. In this paper, we propose a ChAin-centric Reasoning and Pre-training framework (CARP). CARP utilizes hybrid chain to model the explicit intermediate reasoning process across table and text for question answering. We also propose a novel chain-centric pre-training method, to enhance the pre-trained model in identifying the cross-modality reasoning process and alleviating the data sparsity problem. This method constructs the large-scale reasoning corpus by synthesizing pseudo heterogeneous reasoning paths from Wikipedia and generating corresponding questions. We evaluate our system on OTT-QA, a large-scale table-and-text open-domain question answering benchmark, and our system achieves the state-of-the-art performance. Further analyses illustrate that the explicit hybrid chain offers substantial performance improvement and interpretablity of the intermediate reasoning process, and the chain-centric pre-training boosts the performance on the chain extraction. [1]

1 Introduction

Open domain question answering [Joshi *et al.*, 2017; Dunn *et al.*, 2017; Lee *et al.*, 2019; Gao *et al.*, 2021] requires systems to retrieve and perform reasoning over supported knowledge, and finally derive an answer. Generally, the real-world knowledge resource is heterogeneous, which involve both semi-structured web tables and unstructured text like Wikipedia passages. Therefore, question answering over hybrid tabular and textual knowledge is essential and attracts wide attentions [Chen *et al.*, 2020a], and is more challenging as systems need to aggregate information in both table and text considering their connections and the table structure.


Question

How many points did Lebron James get in the NBA Season suspended by COVID-19?

Retrieved Passage

The 2019-20 NBA season is the 74th season of the National Basketball Association. The season was suspended by COVID-19. The 2020 NBA All-Star …

Retrieved Table

Lebron James Career Statistics

Team	Year	Points Per Game	Blocks
L.A. Lakers	19-20	25.3	0.5
Cleveland	17-18	27.8	0.9

Reasoning Process

Figure 1: An example of the table-and-text QA with intermediate reasoning process. The answer is 25.3.

As the example shown in Fig. 1, the complete reasoning process for answering the question involves hybrid information pieces in both the table ("*Year*" and "*Points*" columns in the first row) and the passage ("*COVID-19*"). Therefore, modeling the structural connections inside heterogeneous knowledge is critical for the reasoning process. Many recent works on table-and-text open domain QA simply take the supported flattened table and passages [Chen *et al.*, 2020a; Li *et al.*, 2021] as a whole for question answering, which neglects the structural information and connections among table and text, and leads to more noise as full tables always contain redundant information. Secondly, these methods tackle the

[*] Indicates equal contribution

[†] Work is done during internship at Microsoft Research Asia.

[1] Code is available at https://github.com/zhongwanjun/CARP

whole reasoning process as a black box, and lack the interpretability of the intermediate reasoning process. Moreover, the data sparsity problem is also severe, as the high-quality annotated reasoning process is hard to be obtained.

To tackle these challenges, we propose a ChAin-centric Reasoning and Pre-training framework (CARP), which models the intermediate reasoning process across table and text with a hybrid chain for question answering. CARP first formulates a heterogeneous graph, whose nodes are information pieces in the relevant table and passages, to represent the interaction residing in hybrid knowledge. Then, it identifies the most plausible reasoning path leading to the answer with a Transformer-based extraction model. Moreover, to augment the pre-trained model with ability to identify the reasoning process, we propose a novel chain-centric pre-training method, which takes the advantage of the table structure and table-passage connections to construct large-scale pseudo reasoning paths, and reversely generate questions. CARP framework has following advantages. Firstly, the hybrid chain models the interaction between table and text, and reduces the redundant information. Secondly, it provides a guidance for QA, and better interpretability of the reasoning process. Lastly, both the training of the extraction model and the pre-training corpus construction require no human annotation, which alleviates the data sparsity problem and broadens the potential applications of the framework.

Experiments show that our system achieves the state-of-the-art result on a large-scale table-and-text open-domain question answering benchmark OTT-QA. Notably, the effectiveness of the chain-centric pre-training method is proved by the significant performance boost of the chain extraction model. Results show that incorporating the hybrid chain enhances the QA model, especially for the questions requiring more complicated reasoning process. In summary, our contributions are: 1) We propose to model the intermediate reasoning process for question answering over table and text, with a fine-grained hybrid chain. 2) We propose a novel pre-training method, which captures the reasoning process by pre-training on a synthesized reasoning corpus consisting of large-scale cross-modality reasoning paths and corresponding questions. 3) Experiments show that our system achieves the state-of-the-art result and further analysis proves the effectiveness of utilizing the hybrid chain and the pre-training method.

2 Task Definition

In this paper, we study the task of question answering over table and text in a challenging open-domain setting, because the supported knowledge is not always provided in a realistic application. Taking a question as input, the task [Chen *et al.*, 2020a] requires the system to first retrieve supported tables and passages and then make inference over the retrieved knowledge to derive a free-formed answer as output. The answer is a span from either table cells or passages. One of the core challenges of this task is that problem solving always requires complex reasoning process across table and text, considering the cross-modality interaction and table structure.

3 Framework: CARP

Fig. 2 shows the pipeline of our CARP framework, which has three parts: (1) a **retriever** that retrieves tabular and textual knowledge with the given question; (2) a **chain extractor** that extracts hybrid chain from the retrieved knowledge. (3) a **reader** that answers questions with retrieved knowledge and the extracted hybrid chains. Below we detailedly illustrate the hybrid chain (i.e., definition, extraction, pre-training, and application in QA), and briefly introduce the retriever.

3.1 Hybrid Chain Notation

Hybrid chain logically reveals the fine-grained reasoning process from question to the answer across table and text. We define the **hybrid chain** as a sequence of nodes extracted from a fine-grained heterogeneous graph $\mathcal{G}$, whose nodes V contain the question, cells in the table and sentences in the related passages. One example of the hybrid chain is shown in Fig. 1. Two nodes in the graph are connected by edges E defined by two types of connections: *structural connections* and *contextual connections*. The former indicates that pairs of cells within a same row (e.g., edge c in Fig. 1), or a cell to the a sentence in its linked passage (e.g., edge b), are structurally connected. The latter indicates that pairs of nodes with relevant context (i.e., entity/ keyword co-occurrence) are contextually connected (e.g., edge a indicates co-occurred keyword "COVID-19"). Specifically, we use off-the-shelf named entity recognition model [Peters *et al.*, 2017] to extract entities, and extract noun phrase and numerical items as keywords from the node context. Moreover, a table cell and a passage is linked by the entity linker as described in § 3.5.

3.2 Hybrid Chain Extraction

Here we introduce how to extract hybrid chains, including the model architecture, training and inference process.

Model Architecture. We tackle the chain extraction as a semantic matching problem, which selects the best chain from several candidate chains. Taking a question and a candidate hybrid chain as the inputs, the model calculates the confidence score of the hybrid chain for answering the question. Each candidate hybrid chain is represented as a flattened sequence of its nodes context. We utilize rich contextual representations embodied in pre-trained models like RoBERTa [Liu *et al.*, 2019] to measure the relevance of a question to every chain candidates. Let's take RoBERTa as an example. The input of the hybrid chain extractor is $input = (\texttt{[CLS]}; q; \texttt{[SEP]}; c_i)$ where q and c_i indicate tokenized word-pieces of the question and the flattened i^{th} chain candidate. The `[SEP]` and `[CLS]` are speicial symbols. The representation $\boldsymbol{h}_{c_i} \in \mathbf{R}^d$ is obtained via extracting the hidden vector of the `[CLS]` token. The score $s^+_{c_i}$ for raking the candidates is calculated by:

$$(s^-_{c_i}, s^+_{c_i}) = \mathrm{softmax}(\boldsymbol{W}\boldsymbol{h}_{c_i} + \boldsymbol{b}) \tag{1}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are the learnable parameters. The model is trained with the cross-entropy loss.

Model Training. As mentioned above, the key challenge is constructing the training instances (i.e., ground-truth chains

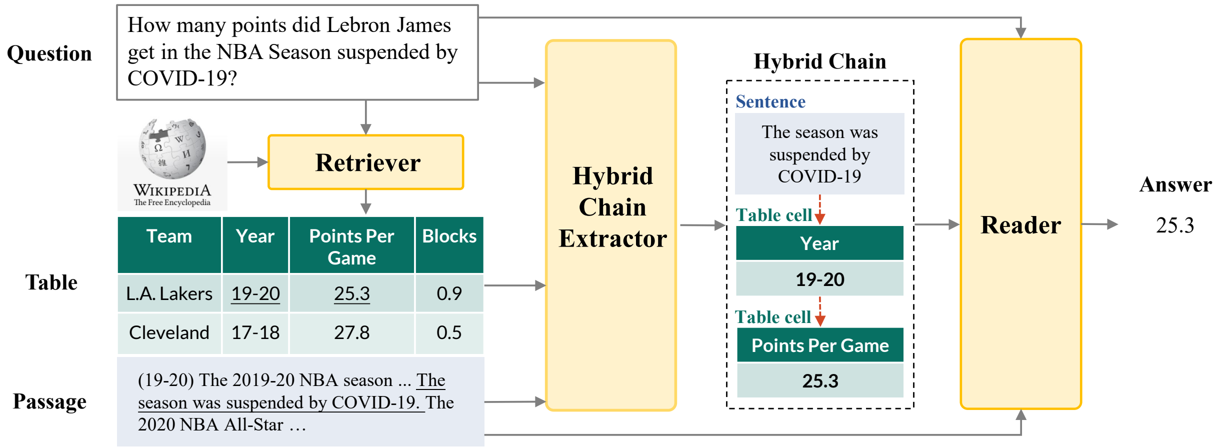


Figure 2: Overview of our system. Retriever (§ 3.5) first retrieves knowledge from the corpus for the question. Secondly, hybrid chain extractor (§ 3.2) extracts hybrid chains from the knowledge, which is improved by pre-training (§ 3.3). Finally, reader (§ 3.4) answers the questions with retrieved evidence and extracted hybrid chain.

and negative chains), as there is no gold-annotated reasoning process given as a prior. We first introduce how to build ground-truth hybrid chains from the heterogeneous graph $\mathcal{G}$. Partly inspired by Chen *et al.* [2019a], we use a heuristic algorithm to derive pseudo ground-truth hybrid chains. Starting from the question, we do the exhaustive search to find all the shortest paths to the nodes containing the answer as the candidate chains. Then, we select the best chain from all candidate chains that have maximum textual similarity with the question as the final ground-truth hybrid chain, and take it as the positive instance. To build the hard negative instances, we find the shortest paths from the question node to the non-answer nodes and select the one with maximum textual similarity with the question.

Model Inference. We first build a set of candidate hybrid chains from graph $\mathcal{G}$, and adopt the extraction model to rank all chains, and finally select the best chain with highest confidence score. Specifically, the set of candidate hybrid chains contains the shortest paths from the question node to all other nodes in the graph. Suppose the number of nodes is n in the graph, the number of candidate chains is $\sum_{i=0}^{n-1} SP(i)$, where SP is the number of shortest paths to node i.

3.3 Chain-centric Pre-training

Pre-training for reasoning is always challenging because high-quality reasoning data is hard to be obtained. To better help the pre-trained model in capturing the complicated reasoning process across table and text and alleviate the data sparsity problem, we propose a chain-centric pre-training method. The method augments the chain extraction model by pre-training on a synthesized reasoning corpus in larger scale and of higher reasoning complexity. The overall process of adopting pre-training strategy is illustrated in Fig. 3: (1) synthesizing heterogeneous chains from the Wikipedia corpus and reversely generating corresponding questions by a trained generator; (2) pre-training a generic extraction model with the synthesized corpus; (3) fine-tuning a specific extraction model with the downstream data. We introduce the pre-training task and the corpus construction.

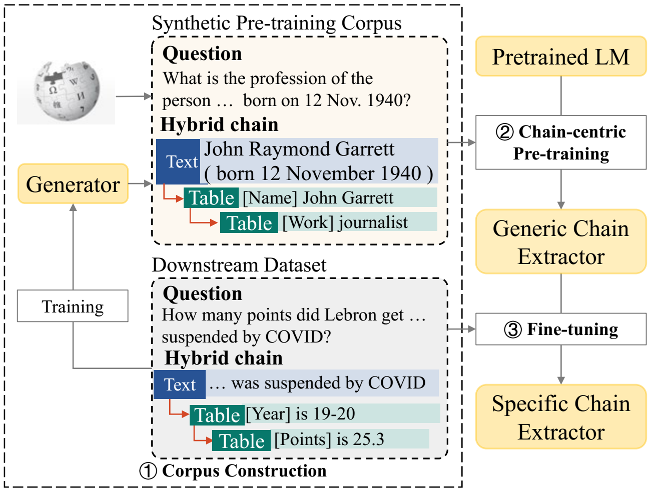


Figure 3: An overview of our pre-training approach. A generic chain extractor is first learned by pre-training on the synthetic corpus. Then we fine-tune the extractor on the downstream dataset.

Task Formulation. The pre-training task can be viewed as a similar semantic matching task that maps hybrid chains to the corresponding pseudo questions. The pre-training objective is in the same spirit of the chain extraction model as described in § 3.2. If the model can better distinguish the relevant hybrid chain for answering the given question, then it has deeper understanding of the reasoning process.

Corpus Construction. To construct the large-scale reasoning corpus, we adopt a novel way of first synthesizing heterogeneous reasoning paths, and then reversely generating corresponding questions. Tables in Wikipedia often contain hyperlinks to their related passages. The clear table structure and the explicit table-text links provide natural benefits for automatically synthesizing logically reasonable reasoning paths.

Therefore, we select semi-structured tables on Wikipedia as the table source, and take the passages hyper-linked to the table cells as the source of passages. The parsed Wikipedia corpus consists of over 200K tables and 3 millions of hyperlinked passages. Then, we synthesize pseudo chains with different reasoning depths. For example, to synthesize a 4-hop reasoning path, we randomly select two cells (c_0, c_1) within the same row and their related passages (p_0, p_1) to form a chain (p_0, c_0, c_1, p_1). Similarly, (p_0, c_0) or (c_0, c_1, p_1) can be selected as a 2-hop or a 3-hop chain, respectively. Finally, taking a synthesized flattened chain as the input, we adopt a generation model built based on BART [Lewis *et al.*, 2019] to reversely generate a pseudo question to construct a pair of *(question, chain)* as a positive instance. It is worth noting that the generation model is trained by the ground-truth *(question, chain)* pairs as described in § 3.2. To encourage the model to better discriminate relevant chains, we select other chains sampled from the same table with top-n similarity with the question as the hard negative instances.

3.4 Hybrid Chain for QA

Having extracted the hybrid chains for each table segment and its related passages, we need to build a reader model to extract the answer a with the inputs. We build a reader model based on a sparse-attention based Transformer architecture *Longformer* [Beltagy *et al.*, 2020] to process long sequence efficiently. With longer limited length up to 4096 tokens, the reader can read top-k retrieved evidences jointly for question answering. The input sequence x is the concatenation of the *question* and top-k pairs of (*table segment, passages*, *hybrid chain*). The Longformer encodes the input x of length T into a sequence of hidden vectors as $\boldsymbol{h}(x) = [\boldsymbol{h}(x)_1, \boldsymbol{h}(x)_2, \cdots, \boldsymbol{h}(x)_T]$. The probabilities $p_{start}(i)$ and $p_{end}(i)$ of the start and end token of a are calculated by:

$$
\begin{aligned}
p_{start}(i) &= \frac{exp(\boldsymbol{W}_s\boldsymbol{h}(x)_i + \boldsymbol{b}_s)}{\sum_j exp(\boldsymbol{W}_s\boldsymbol{h}(x)_j + \boldsymbol{b}_s)} \\
p_{end}(i) &= \frac{exp(\boldsymbol{W}_e\boldsymbol{h}(x)_i + \boldsymbol{b}_e)}{\sum_j exp(\boldsymbol{W}_e\boldsymbol{h}(x)_j + \boldsymbol{b}_e)}
\end{aligned}
\tag{2}
$$

where $\boldsymbol{W}_s$, $\boldsymbol{W}_e$, $\boldsymbol{b}_s$, $\boldsymbol{b}_e$ are learnable weights and bias parameters of the answer extraction layer. Specifically, to alleviate the bias that the model only looks at the extracted chain, we only set the chain as a guidance of the intermediate reasoning process and force the model to select answer from the tokens of the table and passages.

3.5 Knowledge Retrieval

Instead of independently retrieving tables and passages, we follow Chen *et al.* [2020a] and use an "early-fusion" mechanism, which groups highly-relevant table cells in a row and their related passages as a self-contained group (**fused block**). This strategy integrates richer information from two modalities and benefits following retrieval process. We adopt BLINK [Ledell *et al.*, 2020] as the entity linker to link a table cell to its related passage. Specifically, taking the cell to be linked and the table metadata as the inputs, BLINK automatically finds the relevant passages for each cell. After the linking procedure, we represent each fused block as a row in the table and linked related passages. We tackle the fused block as a basic unit for retrieval.

Finally, a Transformer-based retriever retrieves top-k fused blocks as the knowledge. We apply a shared RoBERTa-encoder $RoBERTa(\cdot)$ [Liu *et al.*, 2019] to separately encode questions and fused blocks. The relevance of the question and a fused block is measured by the dot-product over their representations of the `[CLS]` token. We train the retriever as in Karpukhin *et al.* [2020], where each question is paired with a positive fused block and m negative blocks to approximate the softmax over all blocks. Negative blocks are a combination of in-batch negatives which are fused blocks of the other instances in the mini-batch, and hard negative blocks which are sampled from the other rows in the same table. During inference, we apply the trained encoder to all fused blocks and index them with FAISS [Johnson *et al.*, 2021] offline.

4 Experiments

We conduct experiments to explore the effectiveness of our method from the following aspects: (1) the performance of our overall system on QA; (2) the performance of the hybrid chain extraction model; (3) the ablation study about the pretraining strategy; (4) the comprehensive qualitative analysis.

4.1 Dataset and Evaluation

In the real-world scenario, solving many questions requires retrieving supporting heterogeneous knowledge and making reasoning over it. Therefore, we evaluate the performance of our approach on the OTT-QA [Chen *et al.*, 2020a] dataset. OTT-QA is a large-scale table-and-text open-domain question answering benchmark for evaluating open-domain question answering over both tabular and textual knowledge. OTT-QA has over 40K instances and it also provides a corpus collected from Wikipedia with over 400K tables and 6 million passages. Furthermore, the problem solving in OTT-QA requires complex reasoning steps. The reasoning types can be divided into several categories: single hop questions (13%), two hop questions (57%), and multi-hop questions (30%). We adopt the exact match (EM) and F1 scores [Yu *et al.*, 2018] to evaluate the overall QA performance.

4.2 Baselines

We compare our system to the following methods: 1) **HYBRIDER** [Chen *et al.*, 2020b] is a model that uses BM25 to retrieve relevant tables and passages, and adopts a two stage model to cope with heterogeneous information. 2) **Iterative Retriever and Block Reader** The model family is proposed by Chen *et al.* [2020a], which couples Iterative Retriever (IR) / Fusion Retriever (FR) with Single Block Reader (SBR) / Cross Block Reader (CBR). IR and FR indicate retrieving supported knowledge by standard iterative retrieval or using "early fusion" strategy to group tables and passages as fused blocks before retrieval, respectively. SBR indicates the standard way of retrieving top-k blocks and then feeding them independently to the reader and selecting the answer with the highest confidence score. CBR means concatenating the top-k blocks together to the reader, with the goal of utilizing the cross-attention mechanism to model their dependency.

	Dev		Test	
Models	EM	F1	EM	F1
HYBRIDER	10.3	13.0	9.7	12.8
IR + SBR	7.9	11.1	9.6	13.1
FR + SBR	13.8	17.2	13.4	16.9
IR + CBR	14.4	18.5	16.9	20.9
FR + CBR	28.1	32.5	27.2	31.5
DUREPA	15.8	–	–	–
CARP	**33.2**	**38.6**	**32.5**	**38.5**
CARP w/o hybrid chain	29.4	34.2	–	–

Table 1: Performance of different methods on the dev. set and the blind test set on OTT-QA. The performance of CARP without hybrid chain is also reported.

3) **DUREPA** [Li *et al.*, 2021] is a recent method that jointly reads tables and passages and selectively decides to generate an answer or an SQL query to derive the output.

4.3 Model Comparison

Table 1 reports the performance of our model and baselines on the dev. set and blind test set on OTT-QA. In terms of both EM and F1, our model significantly outperforms previous systems with 32.5% EM and 38.5% F1 on the blind test set, and achieves the state-of-the-art performance on the OTT-QA dataset. Our approach, which exploits explicit hybrid chain, helps the model to capture the reasoning process and boost the performance of the QA model.

4.4 Evaluation of Chain-centric Reasoning

To verify the effectiveness of our proposed hybrid chain, we firstly eliminate hybrid chain from the QA model inputs, and report the result of "*CARP w/o hybrid chain*" on the development set in Table 1. Incorporating hybrid chain into the QA model improves the performance significantly.

Then, we explore various variants in hybrid chain extraction, whose backbone is the pre-trained model RoBERTa [Liu *et al.*, 2019]. The variants consider three aspects: (1) **Encoding**: Dual Ranking vs Cross Matching. Dual-tower ranking model [Karpukhin *et al.*, 2020] encodes the question and the hybrid chain separately, and uses the cosine distance to measure their relevance for ranking. Cross matching means that we use a semantic matching model described in § 3.2. (2) **Heterogeneous Graph Construction**: Simple (S) vs Weighted (W). Simple indicates the edges in the graph are unweighted. Weighted graph means that the edges connecting highly-related (higher ratio of overlapped keywords) nodes have lower weight, and thus the paths with higher overall relatedness (shorter length) are ranked higher in the ground-truth chain construction (§ 3.2). (3) **Negative Sampling**: BMNeg vs InnerNeg. BMNeg means that the most similar chain from other positive instances with BM25 are selected as the negative instance. InnerNeg indicates that we select negative instances from other chains constructed from the same fused block, as described in § 3.2.

Table 2 reports the performance of the hybrid chain extraction model (without pre-training) with different components. We note that a selected chain is correct when it contains an answer node. We take Recall@n as the evaluation metric. Based on the table, we have following findings. Firstly, semantic matching model with cross-attention mechanisms performs better than standard dual-tower ranking model, which verifies that cross-attention mechanism is beneficial for modeling the connections among heterogeneous information. Secondly, finding the shortest path in the weighted graph is better than in the simple graph, which shows that modeling the relatedness of nodes is essential in finding a more reasonable hybrid chain. Finally, negative sampling strategy is extremely essential for hybrid chain selection. The goal of inference is to select the most plausible chain from several candidate chains sampled from the same fused block. Therefore, sampling hard negative instance from the same fused block is much better than sampling from other training instances. We take the setting of "*Cross Matching (W + InnerNeg)*" as the final setting of the extraction model.

Methods	Rec@1	Rec@2
Dual Ranking (W + InnerNeg)	61.61	73.15
Cross Matching (W + BMNeg)	44.21	61.14
Cross Matching (S + InnerNeg)	68.32	79.87
Cross Matching (W + InnerNeg)	70.75	80.19

Table 2: Performance of the hybrid chain extraction model with different variances.

Methods	Rec@1	Rec@2
Extractor	70.75	80.19
Extractor + Pre-training (Shortest)	73.40	82.87
Extractor + Pre-training (All)	74.01	83.46

Table 3: Performance of the chain extraction with chain-centric pre-training under different settings.

4.5 Evaluation of Chain-centric Pre-training

In this part, we evaluate the effectiveness of the chain-centric pre-training strategy under different settings. The table cells are aligned to the passages according to their hyperlinks in the Wikipedia website. For pre-training, we explore the different way of constructing instances for training the BART-based generator. **All** means that we take all the paths from the question node to the answer node as positive chains. **Shortest** indicates that we only select the shortest paths.

As shown in Table 3, the pre-training strategy improves the performance of the hybrid chain extraction model by a large margin, showing the effectiveness of chain-centric pre-training in helping the model to capture the intermediate reasoning process with given questions. We believe that the improvement is lead by following reasons. Automatically synthesizing pre-training data is an effective data augmentation scheme because it can generate data in larger scale and of higher reasoning complexity. Besides, selecting all paths leading to answer as positive chains is better than selecting the shortest paths, which is reasonable because pre-training can encourage the model to learn a general reasoning ability.

4.6 Qualitative Analysis

We randomly select 100 instances from the dev. set and annotate the hybrid chains and conduct qualitative analyses.

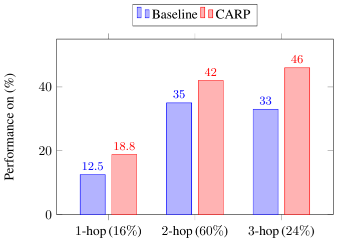


Figure 4: The performance of baseline and our CARP on the randomly selected 100 instances across different hops. The performance on 1-hop questions is lower mainly because these questions are much less frequent in the dataset [Chen *et al.*, 2020a], and always require more complex numerical table understanding.

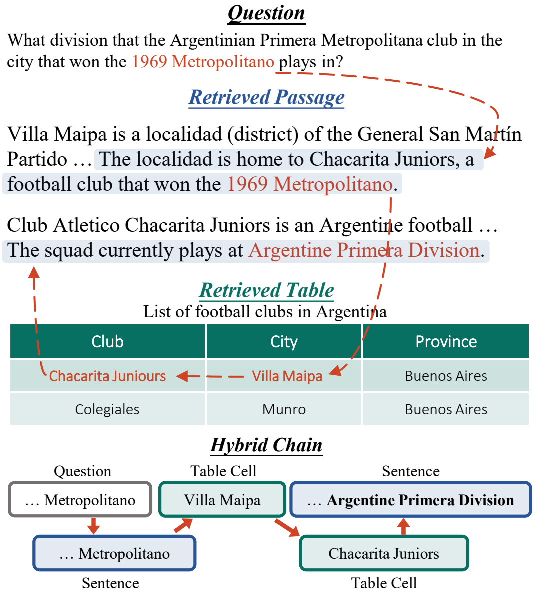
Performance on M-hop Questions. As shown in Fig. 4, we report the performance of the baseline (CARP without hybrid chain) and CARP on the selected questions with different reasoning steps. It can be observed that as the number of reasoning steps increases, the improvement brought by our method to the baseline becomes more significant. This observation verifies that, the hybrid chain is essential in helping the model to identify the intermediate reasoning steps towards the answer especially when the reasoning is more complex. Synthesized pre-training corpus includes higher ratio of 3-hop questions, which enhance the multi-hop reasoning ability.

Case Study. We conduct a case study by giving an example shown in Fig. 5. From the example, our chain extraction model selects a semantic-consistent hybrid chain from the fused block and the QA model correctly predicts the answer with the help of the hybrid chain. This observation reflects that our model has the ability to extract intermediate reasoning process from the given inputs and utilize these information to facilitate the question answering process. Hybrid chain also makes the predictions become more interpretable.

5 Related Work

Web table is an essential knowledge source that storing significant amount of real-world knowledge. There has been a growing interest in QA with both tabular and textual knowledge. HybridQA [Chen *et al.*, 2020b] is a close-domain table-and-text question answering dataset with ground-truth knowledge provided. There are other table-based datasets, like WikiTableQuestions [Pasupat and Liang, 2015], WikiSQL [Zhong *et al.*, 2017], SPIDER [Yu *et al.*, 2018], and TABFACT [Chen *et al.*, 2019b], etc. These datasets mainly focus on table and may discard some important information stored in textual corpus. We study OTT-QA [Chen *et al.*, 2020a], which is a large open-domain table-and-text QA dataset requiring aggregating information from hybrid knowledge.

There exist text-based question answering datasets designed in open-domain [Joshi *et al.*, 2017; Dunn *et al.*, 2017; Lee *et al.*, 2019] or multi-hop [Yang *et al.*, 2018; Welbl *et al.*, 2018] settings. Graph-based models [Fang *et al.*, 2019; Ding *et al.*, 2019] utilize graph structure and graph neural network to model the connections among sentences or entities for multi-hop QA. There are works adopting chain-like reasoning to solve multi-hop textual QA [Chen *et al.*, 2019a; Asai *et al.*, 2019; Feng *et al.*, 2020]. Our approach differs from previous methods mainly in two aspects: (1) using hybrid chain to model the reasoning process across table and text; (2) the chain-centric pre-training method.



Question

What division that the Argentinian Primera Metropolitana club in the city that won the 1969 Metropolitano plays in?

Retrieved Passage

Villa Maipa is a localidad (district) of the General San Martín Partido … The localidad is home to Chacarita Juniors, a football club that won the 1969 Metropolitano.

Club Atletico Chacarita Juniors is an Argentine football … The squad currently plays at Argentine Primera Division.

Retrieved Table

List of football clubs in Argentina

Club	City	Province
Chacarita Juniours	Villa Maipa	Buenos Aires
Colegiales	Munro	Buenos Aires

Hybrid Chain

Figure 5: A case study of CARP. The answer is *Argentine Primera Division*. We omit some unimportant sentences for simplification.

6 Conclusion

In this paper, we present a chain-centric reasoning and pre-training (CARP) framework for table-and-text question answering. When answering the questions given retrieved table and passages, CARP first extracts explicit hybrid chain to reveal the intermediate reasoning process leading to the answer across table and text. The hybrid chain provides a guidance for QA, and explanation of the intermediate reasoning process. To enhance the extraction model with better reasoning ability and alleviate data sparsity problem, we design a novel chain-centric pre-training method. This method synthesizes the reasoning corpus in a larger scale and of higher reasoning complexity, which is achieved by automatically synthesizing heterogeneous reasoning paths from tables and passages in Wikipedia and reversely generating multi-hop questions. The pre-training task boosts performance on the hybrid chain extraction model, especially for questions requiring more complex reasoning, which leads to significant improvement on the performance of the QA model. The hybrid chain also provides better interpretability of the reasoning process.

Acknowledgments

Jian Yin is the corresponding author. This work is supported by the National Natural Science Foundation of China (U1811264,U1811262,U1811261,U1911203,U2001211), Guangdong Basic and Applied Basic Research Foundation (2019B1515130001), Key-Area Research and Development Program of Guangdong Province (2018B010107005, 2020B0101100001)

References

[Asai *et al.*, 2019] Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. Learning to retrieve reasoning paths over wikipedia graph for question answering. *arXiv preprint arXiv:1911.10470*, 2019.

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

[Chen *et al.*, 2019a] Jifan Chen, Shih-ting Lin, and Greg Durrett. Multi-hop question answering via reasoning chains. *arXiv preprint arXiv:1910.02610*, 2019.

[Chen *et al.*, 2019b] Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. Tabfact: A large-scale dataset for table-based fact verification. *arXiv preprint arXiv:1909.02164*, 2019.

[Chen *et al.*, 2020a] Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Wang, and William W Cohen. Open question answering over tables and text. *arXiv preprint arXiv:2010.10439*, 2020.

[Chen *et al.*, 2020b] Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *arXiv preprint arXiv:2004.07347*, 2020.

[Ding *et al.*, 2019] Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. Cognitive graph for multi-hop reading comprehension at scale. *arXiv preprint arXiv:1905.05460*, 2019.

[Dunn *et al.*, 2017] Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*, 2017.

[Fang *et al.*, 2019] Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. Hierarchical graph network for multi-hop question answering. *arXiv preprint arXiv:1911.03631*, 2019.

[Feng *et al.*, 2020] Yufei Feng, Mo Yu, Wenhan Xiong, Xiaoxiao Guo, Junjie Huang, Shiyu Chang, Murray Campbell, M. Greenspan, and Xiao-Dan Zhu. Learning to recover reasoning chains for multi-hop question answering via cooperative games. *ArXiv*, abs/2004.02393, 2020.

[Gao *et al.*, 2021] Yifan Gao, Jingjing Li, Michael R Lyu, and Irwin King. Open-retrieval conversational machine reading. *arXiv preprint arXiv:2102.08633*, 2021.

[Johnson *et al.*, 2021] Jeff Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7:535–547, 2021.

[Joshi *et al.*, 2017] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

[Karpukhin *et al.*, 2020] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020.

[Ledell *et al.*, 2020] Wu Ledell, Petroni Fabio, Josifoski Martin, Riedel Sebastian, and Zettlemoyer Luke. Zero-shot entity linking with dense entity retrieval. In *EMNLP*, 2020.

[Lee *et al.*, 2019] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300*, 2019.

[Lewis *et al.*, 2019] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[Li *et al.*, 2021] Alexander Hanbo Li, Patrick Ng, Peng Xu, Henghui Zhu, Zhiguo Wang, and Bing Xiang. Dual reader-parser on hybrid textual and tabular evidence for open domain question answering. *arXiv preprint arXiv:2108.02866*, 2021.

[Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

[Pasupat and Liang, 2015] Panupong Pasupat and Percy Liang. Compositional semantic parsing on semi-structured tables. *arXiv preprint arXiv:1508.00305*, 2015.

[Peters *et al.*, 2017] Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and R. Power. Semi-supervised sequence tagging with bidirectional language models. In *ACL*, 2017.

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

[Yang *et al.*, 2018] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*, 2018.

[Yu *et al.*, 2018] Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. *arXiv preprint arXiv:1809.08887*, 2018.

[Zhong *et al.*, 2017] Victor Zhong, Caiming Xiong, and Richard Socher. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*, 2017.