ReasonChainQA: Text-based Complex Question Answering with Explainable Evidence Chains

Minjun Zhu*†  Yixuan Weng*  Shizhu He*†  Kang Liu*†  Jun Zhao*†
zhaminjun2020@ia.ac.cn  wengyxy@gmail.com  shizhu.he@nlpr.ia.ac.cn  kliu@nlpr.ia.ac.cn  jzhao@nlpr.ia.ac.cn

Abstract—The ability of reasoning over evidence has received increasing attention in question answering (QA). Recently, natural language database (NLDB) conducts complex QA in knowledge base with textual evidences rather than structured representations, this task attracts a lot of attention because of the flexibility and richness of textual evidence. However, existing text-based complex question answering datasets fail to provide explicit reasoning process, while it's important for retrieval effectiveness and reasoning interpretability. Therefore, we present a benchmark ReasonChainQA with explanatory and explicit evidence chains. ReasonChainQA consists of two subtasks: answer generation and evidence chains extraction, it also contains higher diversity for multi-hop questions with varying depths, 12 reasoning types and 78 relations. To obtain high-quality textual evidences for answering complex question. Additional experiment on supervised and unsupervised retrieval fully indicates the significance of ReasonChainQA. Dataset and codes will be made publicly available upon accepted.

Index Terms—Questions and Answers, Reason chain, Explainable

I. INTRODUCTION

Developing systems that can reason over explicit knowledge has attracted substantial attention in current AI research [1]. Complex Question Answering (Complex QA) tasks provide a comprehensive and quantitative way to measure these abilities, with evidence provided by structured knowledge bases (e.g. WikiData) or natural language texts (e.g. Wikipedia). Considering the high cost of constructing structured knowledge bases, this paper focuses on complex QA over textual evidence.

In textual complex QA tasks, prevailing datasets focus on multi-hop reasoning that requires the aggregation of evidence across multiple paragraphs to answer a question [2]. In order to test the ability of reasoning over text in a more fine-grained way, WIKINLDB dataset [3] constructing knowledge triples to sentence level evidence and introducing database queries into complex text QA task. These datasets have sparked significant progress in textual complex QA. However, existing complex text QA datasets lack explicit, high-quality and in-depth reasoning process. Some datasets provide supporting passage sets without ordered evidence [2], [4] or organize evidence chains by overlap spans in passages [5]. This limitation will hinder the further advancement of reasoning over texts.

*National Laboratory of Pattern Recognition, Institute of Automation, CAS
†School of Artificial Intelligence, University of Chinese Academy of Sciences

Fig. 1. An example of the ReasonChainQA. Each sample contains a question, a set of textual evidence, a unique answer, and a chain of textual evidence with a complete path.

In this work, we propose a step in this direction by introducing a new benchmark ReasonChainQA. Given a complex question based on textual evidence databases, models need to conduct two subtasks: (1) extracting evidence chains; (2) generating correct answer questions. In ReasonChainQA dataset, knowledge facts were expressed as natural language sentences. An evidence chain is a logically connected sentence sequence from question to answer. For example (as shown in figure 1), the evidence chain of question “What language is used by the ethnic group of Jesse Eisenberg?” is sentence 4 → 1 → 7. Evidence chains provide systems with strong meaningful and explainable reasoning supervision about how the answer is derived.

In contrast to WIKINLDB dataset where queries are one-hop or two-hops with 4 reasoning types and 25 relations, and HotpotQA [2], where each query is answered over two passages with two reason types, ReasonChainQA not only...
expresses knowledge as NLDB, but also makes progress in questions’ diversity and complexity. Specifically, our explicit reasoning chain depths vary from 1 to 7 (Table I) with 12 inference types (Table V) and much more correlative relations.

In summary, our contributions are as follows:
1. We present ReasonChainQA, the first textual database QA dataset with explicit evidence chains that covers diversified reason depths, relations and inference types. Compared with Open Domain Question Answering and NLDB, ReasonChainQA evaluates multiple combination reasoning capabilities including multi hop, attribute, and set, and its questions are more diverse.
2. We evaluated the performance of existing methods. It can more comprehensively reveal the prediction effect of different methods in the reasoning path, because our data set provides sequential annotation information.
3. We conducted detailed experiments for reasonchain, providing a robust baseline method for subsequent research.

II. RELATED WORK

Complex Textual QA Tasks. There is vast prior research on complex textual QA. Open-domain QA (OpenQA) tasks retrieve passages from large corpora [6]. Complex QA requires to deal well with passage structures, and infer answers from few paragraphs by deep reasoning. For example, DROP needs to perform discrete numerical reasoning. HotpotQA [2] covers multi-hop questions answered by collecting multiple documents. BeerQA integrates the datasets of different reasoning steps. RuleTaker reasons based on the rules. However, these tasks do not have evidence chains, and the impact of evidence chains on QA tasks has not been studied.

Natural language databases is a special textual QA task. It considers the questions similar to database queries, where system needs to reason and filter on a large number of textual evidence sentences. The bAbI [7] needs to find the answer of one evidence from less than 20 evidences. WIKINLDB [3] uses maximum inner-product search (MIPS) retrievers. GRR [9] recursively retrieves correct evidence in each step until the end-of-evidence token is generated. In previous textual QA tasks, retrieve-and-reading QA systems (open domain question answering) with pseudo chain tag indicate that a complete and sequential evidence path is vital in knowledge reasoning [9]–[11]. However, most of these methods lack global evidence chain modeling, and their performance in long-chain is insufficient. Therefore, the introduction of reasoning chain with indefinite length can evaluate the performance of different methods more finely.

III. TASK DESCRIPTION

A. Problem Definition

ReasonChainQA refers to natural language sentences (textual evidence) as database, each sentence containing one or multiple knowledge facts. An example application of text databases is personal assistant. For example, for answering the query “Among the feature films with a publication date after 2003, which one has the smallest duration?”, QA models require set filter(e.g., year, concept, duration), numeric comparison and facts’ bridge over textual database.

For a question $Q_i$ in ReasonChainQA, it could be answered on a large-scale unsequenced factual texts database. To solve complex questions often requires specific logical reasoning path over multiple facts, we provide reasoning chains $E_i$, consisting of correct reasoning evidence $e_j^i$.

$$E_i = \bigcup_{j=1}^{n} e_j^i$$

For example, the evidence set of $E_i$ in Figure 1 is the evidence chain $R_i=[1,4,7]$. We propose a benchmark ReasonChainQA, and set two sub-tasks: (1) Learning to extract reasoning chains. (2) Generating correct answers for complex questions. We believe that a more fine-grained evaluation of multi-hop question answering system can improve QA performance and make the prediction explainable.

B. Challenge

Mainstream systems developed with complex question answering over textual database tasks follow the retriever-and-reader architecture, where the efficiency and effectiveness of evidence chain retrieval still remain to be great challenges.

Retrieval Stopping Strategy. To advance an efficient iterative retriever without dropping accuracy is still a challenge for evidence chain retrievers. Some existing work chooses to determine a fixed number of iterations [5] or a maximum number of retrieved passages [12], which can barely handle queries of varying depths in ReasonChainQA. Therefore, it’s necessary to consider an iterative retrieval stopping strategy.

Evidence Chains Extraction. DPR [8], MDR [5] and SSG [3] use maximum inner-product search (MIPS) retrievers. GRR [9] compares evidence with currently hidden state embedding independently, and fails to consider the context. These methods ignore the modelling of evidence chain in deep reasoning, which badly damages the overall performance of evidence chain retrieval.

IV. EVIDENCE CHAIN GRAPH

We remodel each evidence chain as a directed acyclic graph, and filter the samples containing multiple parent nodes or multiple child nodes, and filter the samples containing multiple parent nodes or multiple child nodes. We redefined
TABLE I  
STATISTICS OF REASONCHAINQA

| Type                        | Train | Dev | Test | Total |
|-----------------------------|-------|-----|------|-------|
| Number                      | 9505  | 1220| 1268 | 11993 |
| Avg. Question Token length  | 11.11 | 11.28| 11.08| 11.12 |
| Avg. Evidence Token length  | 9.88  | 9.88| 9.85 | 9.87  |
| Avg. Depth of Evidence Chain| 2.13  | 2.12| 2.13 | 2.13  |
| Depth of Evidence Chain = 1 | 3231 (33.99%) | 420 (34.43%) | 425 (33.52%) | 4076 (33.99%) |
| Depth of Evidence Chain = 2 | 2376 (25.00%) | 295 (24.18%) | 336 (26.50%) | 3007 (25.07%) |
| Depth of Evidence Chain = 3 | 3352 (35.27%) | 439 (35.98%) | 424 (33.44%) | 4215 (35.15%) |
| Depth of Evidence Chain >= 4| 546 (5.74%) | 66 (5.41%) | 83 (6.54%) | 695 (5.79%) |

![Fig. 2. Percentage of each program in ReasonChainQA](image)

the relationship of each evidence node, including: **Parent node** means that this node needs to be inferred from these nodes. **Child node** means that these nodes can be obtained by reasoning based on this node. **Sibling node** means that these two nodes are deduced by their parent nodes at the same time, which belongs to the parallel evidence relationship. **No-cross node** means that in the process of reasoning, there is no relationship.

V. DATASET COLLECTION

In this section we introduce ReasonChainQA dataset, a diverse and explainable textual evidence-based complex question answering dataset with explicit reasoning chains. We first transform knowledge base triples from Wikidata to natural language sentences, which providing us with abundant texts to get different size of databases. Then, to generate each example, we acquire texts evidence chains from structured triples according to query and program from KQA-Pro dataset [13].

A. Textual Evidence

We employ several methods to generate databases evidence grounded in Wikidata identifiers. For language diversity, we extend triples with same subject ones to generate more natural and informative facts as knowledge triple graphs. Then we train a text generation model by aligned corpus from KELM, which could generate a natural language evidence according to a knowledge triple graphs. For example, for a knowledge triple “{Winna, Lives in, London}”, We find “{Winna, husband, Mike}” that related to Winna from the knowledge base. After combining them as a knowledge triple graph and inputing into the text generation model (based on T5), we can get “Winna’s husband is Mike and she lives in London”. In this way, factual information of database evidence can be enriched. We also repeatedly utilize generation and post-processing methods to ensure faithfulness and diversity of evidence. Moreover, to avoid generation uncertainty, string matching is taken to ensure all triples appear in generated evidence. Data quality will be shown in V.C. Besides, we mainly collect irrelevant evidence from two aspects to enlarge database, facts with same subject and facts similar to queries.

B. QA and Evidence Chains

To get complex natural language questions and explicit reasoning process, we take over the questions and the corresponding answers in KQA Pro, which is a large-scale KBQA dataset with programs and SPARQLs. In order to transform logical queries to evidence chains, firstly, we select programs of KQA-Pro as the processing objective functions of fact reasoning, which contain 12 types. Secondly, we locate corresponding knowledge triples through the programs. Finally, we redefine the relationship of each evidence node by filtering samples containing multiple parent nodes or multiple child nodes. The relationships between its different nodes include parent node, child node, sibling node and no-cross node. The details are shown in IV.

C. Analysis of ReasonChainQA

**Dataset Statistics.** ReasonChainQA is a challenging large-scale OpenQA dataset composed of 11,993 QA pairs. The details are shown in Table I. The reasoning length of the evidence chains is not fixed and the average length is 2.13.
TABLE II: PROCESSING OBJECTIVE FUNCTION DESCRIPTION FOR ENTITY INFERENCE IN REASONCHAINQA (FOLLOWING KQA-PRO)

| Program          | Number | Example               | Example Output |
|------------------|--------|-----------------------|----------------|
| QueryAttr        | 8688   | QueryAttr(height)     | 185cm          |
| FilterConcept    | 8313   | FilterConcept(athlete)| Kobe Bryant    |
| FilterStr        | 5057   | FilterStr(gender, male)| (Entities, Facts) |
| FindAll          | 4837   | FindAll()             | All Entities   |
| Relate           | 3476   | Relate(capital, forward)| Beijing      |
| What             | 2562   | What(KobeBryant)      | Kobe Bryant    |
| VerifyStr        | 2319   | VerifyStr(male)       | True           |
| VerifyYear       | 1641   | VerifyYear(1980, >)   | False          |
| Count            | 743    | Count(Entities)       | 8              |
| VerifyNum        | 479    | VerifyNum(20000dollars, >) | False |
| FilterDate       | 421    | FilterDate(birthday, 1980/06/01, <) | (Entities, Facts) |
| FilterYear       | 90     | FilterYear(birthday, 1980, =) | (Entities, Facts) |

After that, to evaluate the generalization ability of the model on data sets of different scales, we construct three data sets of different sizes by randomly searching unrelated fact triples. Following setting in WIKINLDB, each of them has a database of 25, 250 and 1000 samples, respectively.

Dataset Quality. To measure the quality of generating factual texts from knowledge base triples, we randomly select 100 samples and manually evaluate factual accuracy, text fluency and complexity. We hired two master’s students to analyze 100 samples. 100/100 sentences can accurately be mapped to according to triples, 27/100 sentences are mapped to multiple triples, and some sentences mapped to one triple are Companyed with qualifiers. 88/100 sentences can clearly understand the meaning. 99/100 of the samples can infer the correct answer. We analyze some error examples. The main error comes from the lack of fluency of sentences due to the generation of language models. In future research, we will focus on solving this problem.

VI. EXPERIMENTAL RESULTS

A. Setup

In order to evaluate the performance of different methods in Reasonchain in a more comprehensive way, we divide them into two categories. The first one does not need to use sequence information as an unsupervised experiment, and the second one uses sequence information for supervision, which we regard as a supervised experiment.

In the unsupervised experiments, the model is trained without the evidence chains’ supervision, but evaluated with correct evidence chains. It is compared with some baseline systems without multi-hop retrieval.

B. Evaluation Metrics

ReasonChainQA contains two subtasks, evidence chains extracting and complex question answering. The former subtask is the foundation for the latter, and can also provide interpretability for QA task.

For sequential reasoning tasks, we divide them into unsupervised tasks and supervised tasks to evaluate the system. We use exact match and edit distance to evaluate the effect of a system in multi-hop reasoning. Specifically, for unsupervised tasks, we do not provide relevant evidence chain labels but follow the setting of the original paper. For supervised tasks, we take our evidence chain tag as training, generate an evidence chain according to a model we require, and then calculate the matching degree between this evidence chain and the $R_I$ tag we provide. In addition, in order to more accurately evaluate the performance of the model on larger scale data, we provide additional test sets of $n = 250$ and $n = 1000$. For these, we compute recall (@k), which measures the fraction of times the correct document is found in the top-k predictions.

C. Experimental Details

We train the model using the PyTorch(1.8.2) 1 on the NVIDIA RTX3090 GPU and use the hugging-face 2 framework. For all methods, the bert-base-uncased 3 model are chosen for feature extraction. The pretrained contextual encoders are of base size with 12 layers. We input the output vector of BERT into the average pooling layer to obtain the text vector. We use the AdamW as the optimizer with the warm-up, and finetune the whole model with a learning rate of 1e-5. A bidirectional single-layer LSTM is randomly initialized as the Decoder, hidden state is 384, to obtain a new $V_{H}^{t+1}$ with dimension 768 from input $[V_{Q}, V_{K}, V_{H}]$. We set the maximum length of 40, delete the excess. We use linear decay of learning rate and gradient clipping of 1e-6. The dropout of 0.1 is applied to prevent overfitting. When calculating the prediction time, Random and BM25 are performed on the CPU and the rest of the systems uniformly use the GPU RTX3090 for prediction time statistics.

D. Performance of Benchmarks

Following previous work [3], [9], we report Exact Match (EM), Edit Distance (ED), and F1 in Table III. We can observe that: Firstly, when the evidence chain information of the training set [9] is limited as an unsupervised situation. Almost all methods cannot show excellent performance.

Secondly, for all methods, with the improvement of chain level accuracy, the accuracy of retrieval is also improving.

1https://pytorch.org
2https://github.com/huggingface/transformers
3https://huggingface.co/bert-base-uncased
TABLE III
WE EVALUATED SEVERAL EXISTING RETRIEVAL METHODS IN REASONCHAINQA DATASET. WE USE PATH SEQUENCE AS A SUPERVISED COMPARISON AND NOT USE PATH SEQUENCE AS AN UNSUPERVISED COMPARISON. THE BEST SCORES IN THE TABLE ARE SHOWN IN BOLD, AND THE SCORES IN THE SECOND PLACE ARE UNDERLINED.

| System | 25  | 250 | 1000 |
|--------|-----|-----|------|
| Random | EM↑ | EM↑ | EM↑  |
|        | 1.10% | 0.15% | 0.08% |
| Unsupervised | BM25 | DPR | SSG  |
|        | 29.65% | 40.93% | 37.85% |
|        | 1.38 | 0.59 | 0.93 |
|        | 58.89% | 66.77% | 79.39% |
| Supervised | GRR | MDR | ED↓  |
|        | 50.55% | 54.50% | 85.43% |
|        | 1.03 | 0.45 | 85.43% |
|        | 97.86% | 51.74% | 81.73% |

TABLE IV
EXACT MATCH PERFORMANCE FOR QA TASKS. WE UNIFORMLY USE THE RETRIEVED TOPK EVIDENCE AS INPUT TO THE READER FOR RESULT GENERATION.

| QA Exact Match | 25  | 250 | 1000 |
|----------------|-----|-----|------|
| Random | EM↑ | ED↓ | F1↑ |
|        | 33.44% | 82.58% | 54.35% |
| Unsupervised | BM25 | DPR | SSG  |
|        | 57.10% | 69.72% | 79.42% |
|        | 52.37% | 63.09% | 64.51% |
|        | 50.95% | 54.50% | 34.34% |
| Supervised | GRR | MDR | F1↑  |
|        | 90.38% | 70.27% | 81.37% |
|        | 81.62% | 68.45% | 81.73% |

TABLE V
HYPER-PARAMETER SETTINGS.

| Hyper-parameter | Value |
|-----------------|-------|
| Encoder Hidden Size | 768 |
| Dropout | 0.1 |
| Learning Rate | 1e-5 |
| Batch size | 24 |
| Num Epochs | 10 |
| Beam | 5 |

which proves that improving the reasoning performance of the evidence chain can further assist retrieval.

E. The Effectiveness of QA

We follow the standard Retriever-Reader architecture and evaluate the effectiveness of different searchers in QA tasks through fixed readers. In Table IV, we employ T5 models fine-tuned on golden evidence as a Reader. We uniformly use the same fine-tuned T5 [14] model with a learning rate of 1e-4 for the final question and answer accuracy. It as the reader for fairness consistency and report the EM scores of reader results from different systems.

The retrieval evidence chains or sets from different systems are input to the reader. For the unsupervised system, we uniformly extract the results from Top-k retrieval evidence where k = length of evidence chains. For the supervised system, we take the retrieval model for generation (this means that k is judged by the model), after which the generated results are used as input to the reader model. We train five readers with different random numbers and take the average score as the reported result.

We can find that a more orderly reasoning chain can help readers better understand the structure of evidence and further find the correct answer.

VII. CONCLUSIONS

In this study, we propose a diverse and explainable complex question answering database with explicit reasoning chain. ReasonChainQA provides accurate path retrieval information to help the system learn interpretable evidential reasoning. We evaluate six baseline methods on three benchmark datasets at different scales. In the experiment, we evaluated in detail the modeling ability of different methods of evidence chain sequence, which was lacking in previous experiments. We further showed the importance of the evidence chain to question and answer tasks.

VIII. ACKNOWLEDGEMENT

This work is supported by the National Key Research and Development Program of China (No. 2020AAA0106400), the National Natural Science Foundation of China (No.61922085, No.61976211) and the Key Research Program of the Chinese Academy of Sciences (Grant NO.ZDBS-SSW-JSC006). This research work was supported by the independent research project of National Laboratory of Pattern Recognition, the Youth Innovation Promotion Association CAS and CCF-DiDi GAIA Collaborative Research Funds for Young Scholars.

REFERENCES

[1] P. Clark, O. Tafjord, and K. Richardson, “Transformers as soft reasoners over language,” in Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, C. Bessiere, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2020, pp. 3882–3890, main track.

[2] Z. Yang, P. Qi, S. Zhang, W. Cohen, R. Salakhutdinov, and C. D. Manning, “HotpotQA: A dataset for diverse, explainable multi-hop question answering,” in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 2369–2380. [Online]. Available: https://aclanthology.org/D18-1259

[3] J. Thorne, M. Yazdani, M. Saeidi, F. Silvestri, S. Riedel, and A. Halevy, “Database reasoning over text,” 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), 2021, pp. 3091–3104.
[4] P. Qi, H. Lee, T. Sido, and C. Manning, “Answering open-domain questions of varying reasoning steps from text,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 3599–3614. [Online]. Available: https://aclanthology.org/2021.emnlp-main.292

[5] W. Xiong, X. L. Li, S. Iyer, J. Du, P. Lewis, W. Y. Wang, Y. Mehdad, W.-t. Yih, S. Riedel, D. Kiela, and B. Oğuz, “Answering complex open-domain questions with multi-hop dense retrieval,” International Conference on Learning Representations, 2021.

[6] E. M. Voorhees, “The trec-8 question answering track report,” 2000.

[7] J. Weston, A. Bordes, S. Chopra, A. M. Rush, B. van Merriënboer, A. Joulin, and T. Mikolov, “Towards ai-complete question answering: A set of prerequisite toy tasks,” arXiv: Artificial Intelligence, 2015.

[8] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, “Dense passage retrieval for open-domain question answering,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Nov. 2020, pp. 6769–6781.

[9] A. Asai, K. Hashimoto, H. Hajishirzi, R. Socher, and C. Xiong, “Learning to retrieve reasoning paths over wikipedia graph for question answering,” in International Conference on Learning Representations, 2020.

[10] S. Saha, S. Ghosh, S. Srivastava, and M. Bansal, “PRover: Proof generation for interpretable reasoning over rules,” in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, Nov. 2020, pp. 122–136. [Online]. Available: https://aclanthology.org/2020.emnlp-main.9

[11] S. Saha, P. Yadav, and M. Bansal, “multiPRover: Generating multiple proofs for improved interpretability in rule reasoning,” in Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, Jun. 2021, pp. 3662–3677.

[12] Y. Feldman and R. El-Yaniv, “Multi-hop paragraph retrieval for open-domain question answering,” meeting of the association for computational linguistics, 2019.

[13] J. Shi, S. Cao, L. Pan, Y. Xiang, L. Hou, J. Li, H. Zhang, and B. He, “Kqa pro: A dataset with explicit compositional programs for complex question answering over knowledge base,” 2022.

[14] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” Journal of Machine Learning Research, vol. 21, no. 140, pp. 1–67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html