# Explainable Question Answering based on Semantic Graph by Global Differentiable Learning and Dynamic Adaptive Reasoning

**Jianguo Mao**[1,2‡] **Wenbin Jiang**[3*] **Xiangdong Wang**[1] **Hong Liu**[1]
**Yu Xia**[3], **Yajuan Lyu**[3], **Qiaoqiao She**[3]
[1]Beijing Key Laboratory of Mobile Computing and Pervasive Device,
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Baidu Inc., Beijing, China
{maojianguo20s,xdwang,hliu}@ict.ac.cn
{jiangwenbin,lvyajuan,sheqiaoqiao}@baidu.com

## Abstract

Multi-hop Question Answering is an agent task for testing the reasoning ability. With the development of pre-trained models, the implicit reasoning ability has been surprisingly improved and can even surpass human performance. However, the nature of the black box hinders the construction of explainable intelligent systems. Several researchers have explored explainable neural-symbolic reasoning methods based on question decomposition techniques. The undifferentiable symbolic operations and the error propagation in the reasoning process lead to poor performance. To alleviate it, we propose a simple yet effective Global Differentiable Learning strategy to explore optimal reasoning paths from the latent probability space so that the model learns to solve intermediate reasoning processes without expert annotations. We further design a Dynamic Adaptive Reasoner to enhance the generalization of unseen questions. Our method achieves 17% improvements in F1-score against BreakRC and shows better interpretability. We take a step forward in building interpretable reasoning methods.

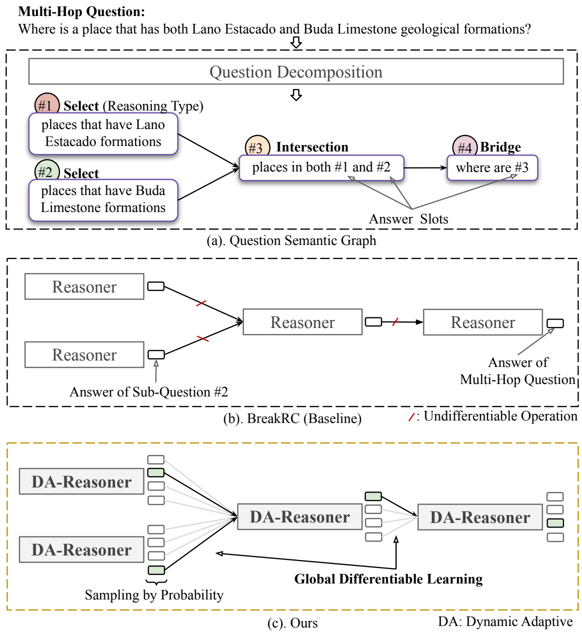


Figure 1: **Overall architecture of the proposed method**. (a) gives an instance of the Question Semantic Graph. As (c) shows, we propose two simple yet effective improvements for the explainable reasoning method illustrated in (b), including Global Differentiable Learning and Dynamic Adaptive Reasoner (DA-Reasoner).

## 1 Introduction

Multi-hop Question Answering involves retrieving supporting facts from multiple documents along with the explicit reasoning path and reasoning out the answer (Yang et al., 2018). As pre-trained language models evolved, the performance on this task improved spectacularly (Kenton and Toutanova, 2019; Beltagy et al., 2020; Zaheer et al., 2020; Joshi et al., 2020; Zhu et al., 2021a; Li et al., 2022). Despite the success, the black-box nature of pure neural networks has raised concerns among researchers that the unexplainable reasoning process is unacceptable for building trustworthy and robust intelligent systems (Min et al., 2019; Ding et al., 2019; Perez et al., 2020; Wolfson et al., 2020; Tang et al., 2021).

A feasible way to realize an explainable reasoning mechanism is by modeling the reasoning path explicitly. Some researchers have successfully explored the idea of breaking up a multi-hop question into sub-questions and solving them step by step according to the logical relationships to arrive at the final answer. Due to the complexity and expense of constructing question decomposition datasets, early work explored unsupervised (Perez et al., 2020) or weakly supervised (Min et al., 2019) question decomposition methods. However, the sub-questions lack reasoning over logical relationships, thus they are only valuable for retrieving supporting facts. As Figure 1 (a) shows, the Allen institute (Wolfson

*Corresponding author.
‡This work was done during internship at Baidu Inc.



*Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5318–5325
December 7-11, 2022 ©2022 Association for Computational Linguistics

et al., 2020) proposed the first large-scale question decomposition dataset, where each instance contains a multi-hop question and a question semantic graph consisting of sub-questions annotated by human experts according to a reasoning path. Based on this, they further explored BreakRC, a neural-symbolic reasoning method, and achieved good interpretability. However, the undifferentiable symbolic operations make the neural network reasoner untrainable. Thus, the semantic space of the reasoner does not match the target sub-questions. Furthermore, the error propagation in the reasoning path exacerbates this effect leading to the performance lagging behind the mainstream implicit reasoning models.

We propose a simple yet effective Global Differentiable Learning strategy to alleviate the problem, as is shown in Figure 2. It learns reasoning capability by exploring the optimal reasoning path in the latent reasoning space. The reasoner will predict a set of candidate answers for each sub-question one by one. Then the answer will be sampled by probability and passed to the answer slots in the next logically adjacent sub-question. During training, for the same instance, the model explores a variety of reasoning paths in the potential space by probability. We let the gradients backpropagated under symbolic operations by using the Straight-Through Estimator (Jang et al., 2017). The trick allows the reasoner to become trainable to adapt to sub-questions without ground-truth answers. We further design an Dynamic Adaptive Reasoner to improve generalization to unseen sub-questions.

## 2 Method

This section first introduces the backbone network, including question decomposition and neural-symbolic reasoning mechanism. Then we introduce the proposed two improvements.

**Backbone** Figure 1 illustrates the architecture of BreakRC. Given a multi-hop question $Q$, it first uses a decomposition model (Wolfson et al., 2020) to break out a multi-hop question into a set of sub-questions $\mathbf{sq} = \langle \{sq^1\}, ..., \{sq^n;\text{\#n-1;\#n-2}\} \rangle$. Each sub-question contains zero or several answer slots. The slot number corresponds to the sub-question number, meaning that slot #n should be filled with the answer to the nth sub-question $sq^n$. A directed acyclic question semantic graph can be constructed based on the slot relationship. Reasoner $\mathbf{R}$ is an off-the-shelf single-hop reading com-

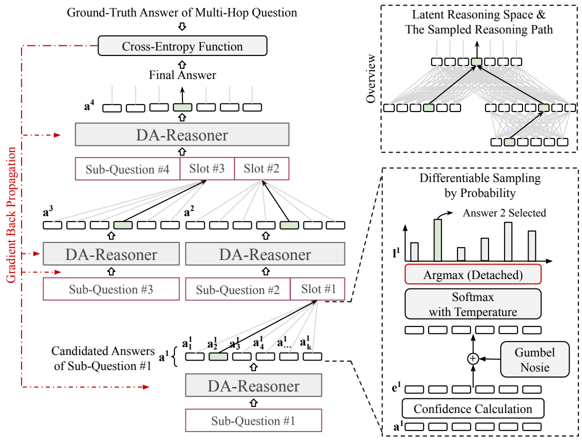


Figure 2: **Global Differentiable Learning**. The box in the top right corner shows one of the reasoning paths (bold black line) sampled from the latent reasoning space by probability.

prehension model. It predicts answers $\{a^1, ..., a^n\}$ to the sub-questions $\{s^1, ..., s^n\}$ based on context $C$ one by one and fills the corresponding answer slots. The answer to the last sub-question $sq^n$ is used as the reasoning result of the original multi-hop question. $\theta$ is set of the model parameters.

$$a^n = \mathbf{R}(sq^n, a^{n-1}, a^{n-2}, C, \theta) \tag{1}$$

**Global Differentiable Learning** Figure 2 illustrates the learning process. It learns to solve intermediate reasoning process by exploring various reasoning paths from the latent reasoning space through a differentiable sampling strategy, alleviating the problem of semantic space mismatch and error propagation.

Due to the lack of ground-truth answers for the sub-questions, we assume that all the candidate answers are possible correct options. Different reasoning paths are generated when different sub-answers are passed over the question semantic graph. All possible paths form a large latent reasoning space. Specifically, the reasoner $\mathbf{R}$ takes the sub-question $sq^n$ for which the slot has been filled and the context $C$ as input, and predicts $k$ candidate answers $\mathbf{a^n} = \{a_1^n, ..., a_k^n\}$. Each answer is a successive token span in the context. The confidence $\mathbf{e^n} = \{e_1^n, ..., e_k^n\}$ of each answer is the sum of the probabilities of the first and the last token being the span's start and end points, respectively. We reparameterize the confidence $\mathbf{e^n} \in \mathcal{R}^{k\times 1}$ by adding Gumbel noise $G^n \in \mathcal{R}^{k\times 1}$ to it:

$$\mathbf{e}^n = \mathbf{e}^n + \mathbf{G}^n \tag{2}$$

$$\mathbf{G}^n = -\log(-\log(\mathbf{U})) \tag{3}$$

$$\mathbf{U} \sim \text{Uniform}(0, 1) \tag{4}$$

We then apply $\mathrm{softmax}$ with temperature $\tau$ to calculate the logits $\mathbf{l^n} = \{l_1^n, ..., l_k^n\}$ for the reparameterized confidence $\mathbf{e^n}$. Finally, we sample answer by applying $\mathrm{Argmax}$ function. The above process achieves probability-based sampling. The sampled answer is passed to the slot in the corresponding sub-question. Repeat the above process until the last sub-question.

$$l_i^n = log\frac{\exp(e_i^n/\tau)}{\sum_{k=1}^{K}\exp(e_k^n/\tau)} \tag{5}$$

where the temperature $\tau$ is a hyper parameter that controls the degree of smoothness of the probability distribution. The higher the temperature, the smoother the probability distribution, tending to explore diverse reasoning paths. As training progresses, the temperature is adjusted from high to low, limiting the available sampling space to approximate the actual distribution.

The final reasoning result is the predicted answer to the last sub-question. First, we use the cross-entropy function to measure the difference between it and the ground-truth answer to the original multi-hop question. Then, we use a Straight-Through Estimator (Jang et al., 2017) to detach the undifferentiable discrete operation $\mathrm{Argmax}$ from the computational graph. It makes it possible to back-propagate the gradient along the reasoning path. The reasoner learns to solve the intermediate reasoning process by performing gradient updating.

**Dynamic Adaptive Reasoner** The Dynamic Adaptive Reasoner is a parameter-sparsified version of the classic reading comprehension model consisting of a transformer encoder and a classification head. It enhances the generalization of unseen sub-questions by leveraging the semantics of sub-question and reasoning types to route encoding blocks.

The encoder consists of a static and a dynamic adaptive part. In the dynamic adaptive part, each layer contains $\mathrm{M}$ transformer blocks $\{\mathbf{TRM}_1^j, ..., \mathbf{TRM}_M^j\}$ with the same structure and initial parameters. $\mathbf{TRM}_m^j$ is the $\mathrm{mth}$ transformer block of the $\mathrm{jth}$ layer. Each block is also assigned a handle features for routing computations

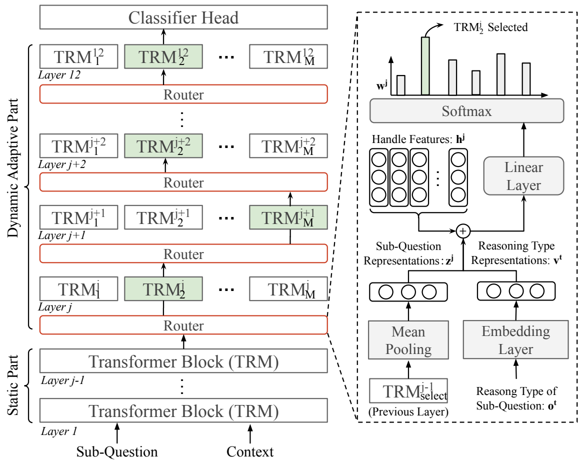


Figure 3: **Dynamic Adaptive Reasoner**. In the dynamic adaptive part, each layer's router will dynamic determine the block for encoding according to the semantic and reasoning type of sub-questions.

$\mathbf{h}^j = \{h_1^j, ..., h_M^j\}$. The router dynamic selects one block for encoding based on the semantics and reasoning type of the sub-question and the handle features. Specifically, when conducting routing for layer j, the semantic representation $z_j$ is the average feature of all tokens of the sub-question encoded by the selected block in layer $j-1$:

$$z_j = \text{MeanPooling}(\mathbf{TRM}_{selected}^{j-1}(sq^n)) \tag{6}$$

Each sub-question $sq^n$ belongs to a specific reasoning type $o^t$ determined during the question decomposition phase. There are a total of 13 types, such as Select, Filter, Project, etc. We embed them in a vector space with each reasoning type corresponding to a learnable vector $v^t$. We project the sum of them into a low-dimensional space and apply softmax function to calculate the probability of the distribution $\mathbf{w}_j \in R^{M\times 1}$. Finally, the router selects the block $\mathbf{TRM}_{select}^j$ with the highest probability.

$$v^t = \text{Embedding}(o^t) \tag{7}$$

$$\mathbf{w}_j = \text{softmax}(\text{Linear}(z_j + v^t + \mathbf{h}^j)) \tag{8}$$

$$\mathbf{TRM}_{select}^j = \text{argmax}(\mathbf{w}_j) \tag{9}$$

Sub-questions with similar semantics and reasoning types will be encoded by the same blocks, achieving approximate clustering to improve generalization to unseen sub-questions.

## 3 Related Work

Many early works focused on improving information retrieving and implicit reasoning mechanism (Nishida et al., 2019; Qiu et al., 2019; Asai et al., 2019; Beltagy et al., 2020; Zaheer et al., 2020; Joshi et al., 2020; Perez et al., 2020; Xiong et al., 2020; Fang et al., 2020; Groeneveld et al., 2020; Li et al., 2021; Zhang et al., 2021; Zhu et al., 2021b; Wu et al., 2021; Qi et al., 2021). Despite the success, they are unexplainable. Various interpretable methods have been proposed for HotpotQA. **DecompRC** (Min et al., 2019) explored a weakly supervised question decomposition method and ensembles the results of the question decomposition-based and implicit reasoning methods. **CogQA** (Ding et al., 2019) built a cognitive graph by coordinating an implicit extraction module and an explicit reasoning module to provide explainable reasoning paths. **SNMN** (Jiang and Bansal, 2019) leveraged the Neural Module Network to construc explainable system. **ModularQA** (Khot et al., 2021) learns to ask sub-questions to existing simple QA models without annotated decompositions. **BreakRC** (Wolfson et al., 2020) constructed the first large-scale question decomposition dataset and proposed a novel neural-symbolic reasoning method that shows good interpretability.

## 4 Experiments

**Datasets** We evaluate our method on both the distractor and fullwiki settings of HotpotQA (Yang et al., 2018). The dataset contains 105,257 multi-hop questions derived from Wikipedia paragraphs, where the correct answer is a span in these paragraphs. We present the EM (Exact Match) and F1 scores.

**Implementation Details** For Global Differentiable Learning, we set the temperature $\tau$ to 10 and halve it after each epoch. For Dynamic Adaptive Reasoner, we choose the last four layers of the encoder as the dynamic adaptive part, with the number of blocks per layer set to 3. We follow the same approach of BreakRC to use the BERT-based RC model from (Min et al., 2019) as the basic reasoner, trained solely on SQuAD (Rajpurkar et al., 2016) (a single-hop question answering dataset). For optimization, we use Adam and set the learning rate to 2e-5. The dimension of handle features is set to 768. The neuron number of reasoning type embedding layer is set to 768. The maximum number of epochs is set to 5. We conduct our experiments on NVIDIA V100 GPU with 32GB memory.

| Model | Distractor | | Fullwiki | |
|---|---|---|---|---|
| | EM | $F_1$ | EM | $F_1$ |
| CogQA | - | - | 37.6 | 49.4 |
| DecompRC | - | 61.7 | - | 39.1 |
| ModularQA | - | 61.8 | - | - |
| SNMN | - | 63.1 | - | - |
| BreakRC$^P$ | 37.6 | 49.4 | 28.8 | 43.3 |
| BreakRC$^G$ | 39.2 | 51.4 | 34.6 | 44.6 |
| Ours$^P$ | **53.1**$^{\uparrow 15.5}$ | **67.3**$^{\uparrow 17.9}$ | **43.7**$^{\uparrow 14.9}$ | **60.2**$^{\uparrow 16.9}$ |
| w/o GDL | 40.3 | 51.9 | 31.2 | 45.6 |
| w/o DAR | 49.7 | 65.8 | 40.1 | 57.6 |
| Ours$^G$ | **55.4**$^{\uparrow 16.2}$ | **69.1**$^{\uparrow 17.7}$ | **50.3**$^{\uparrow 15.7}$ | **61.7**$^{\uparrow 17.1}$ |
| w/o GDL | 42.7 | 53.6 | 37.1 | 47.3 |
| w/o DAR | 52.2 | 66.2 | 45.9 | 59.2 |

Table 1: **Results on HotpotQA.** The - means that the work did not report the result. Global Differentiable Learning (GDL). Dynamic Adaptive Reasoner (DAR)

**Baseline Models** We compare our method with some explainable models used for HotpotQA, including **BreakRC**,**DecompRC**,**CogQA**,**SNMN** and **ModularQA**. For a fair comparison, we use the DecompRC 1hop train version, which excludes an additional scorer module.

**Results** Table 1 shows the results. We report results for Ours$^P$, which uses the predicted question semantic graph, and Ours$^G$, which uses gold question semantic graph. Our method significantly improves the performance against our baseline BreakRC and other explainable models. Furthermore, the ablation study further demonstrates the effectiveness of the two improvements.

**Case Study** Figure 4 shows two cases of explainable reasoning process. Our method learns to solve the intermediate sub-questions and shows better interpretability. For more cases and analysis, please refer to Appendix A.

## 5 Conclusion

We take a step forward in constructing the explainable method for Multi-hop Question Answering by proposing two effective improvements. The Global Differentiable Learning strategy learns optimal reasoning paths by exploring latent probability space to alleviate the problem of semantic space mismatch and error propagation. The Dynamic Adaptive Reasoner improves generalization to unseen sub-questions.

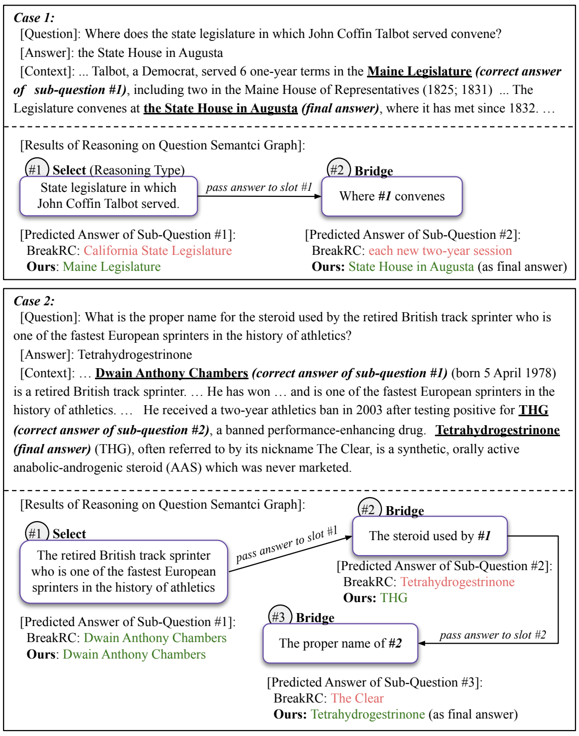


Figure 4: **Case Study**. The green font represents the correct predicted answer, and the red font represents the incorrect. Our method successfully learns the intermediate reasoning process and shows better interpretability.

## Acknowledgements

Our work is supported by the National Natural Science Foundation of China (62276250). This work is also supported by Baidu and ICTCAS Joint Project. We would like to thank the anonymous reviewers for their valuable feedback.

## 6 Limitations

- Question decomposition is the pre-stage of building interpretable models. To the best of our knowledge, there is only one large-scale question decomposition dataset (Wolfson et al., 2020), and the performance of existing automatic decomposition models is far below human performance. Inaccurate question decomposition leads to errors in reasoning. Therefore, exploring better question decomposition techniques is a challenging and rewarding direction.

- Existing interpretable models (Min et al., 2019; Jiang and Bansal, 2019; Ding et al., 2019; Khot et al., 2021; Wolfson et al., 2020), including our approach, focus on solving complex questions, ignoring a simple question with a complex context that requires a deep understanding of the context to reason out the answer.

- The Dynamic Adaptive Reasoner introduces a small number of additional parameters in the router, which can increase the computational cost. A more efficient parameter-free routing approach can be explored in the future.

## References

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

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

Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. 2019. Cognitive graph for multi-hop reading comprehension at scale. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2694–2703.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8823–8838.

Dirk Groeneveld, Tushar Khot, Ashish Sabharwal, et al. 2020. A simple yet strong pipeline for hotpotqa. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8839–8845.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparametrization with gumble-softmax. In *International Conference on Learning Representations (ICLR 2017)*. OpenReview. net.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4474–4484.

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

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Tushar Khot, Daniel Khashabi, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2021. Text modular networks: Learning to decompose tasks in the language of existing models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1264–1279.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Xin Jiang, Qun Liu, Chengjie Sun, Zhenzhou Ji, and Bingquan Liu. 2021. Hopretriever: Retrieve hops over wikipedia to answer complex questions.

Xin-Yi Li, Wei-Jun Lei, and Yu-Bin Yang. 2022. From easy to hard: Two-stage selector and reader for multi-hop question answering. *arXiv preprint arXiv:2205.11729*.

Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Multi-hop reading comprehension through question decomposition and rescoring. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6097–6109.

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. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2335–2345.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8864–8880.

Peng Qi, Haejun Lee, Tg Sido, and Christopher D Manning. 2021. Answering open-domain questions of varying reasoning steps from text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3599–3614.

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 *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150.

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

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 *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3244–3249.

Tomer Wolfson, Mor Geva, Ankit Gupta, Matt Gardner, Yoav Goldberg, Daniel Deutch, and Jonathan Berant. 2020. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*, 8:183–198.

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

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, et al. 2020. Answering complex open-domain questions with multi-hop dense retrieval. *arXiv preprint arXiv:2009.12756*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297.

Yuyu Zhang, Ping Nie, Arun Ramamurthy, and Le Song. 2021. Answering any-hop open-domain questions with iterative document reranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 481–490.

Yunchang Zhu, Liang Pang, Yanyan Lan, Huawei Shen, and Xueqi Cheng. 2021a. Adaptive information seeking for open-domain question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3615–3626.

Yunchang Zhu, Liang Pang, Yanyan Lan, Huawei Shen, and Xueqi Cheng. 2021b. Adaptive information seeking for open-domain question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3615–3626.

## A More Cases

As Figure 5 shows, we present four extra cases to illustrate the effectiveness and interpretability of our method. We present all the intermediate reasoning results predicted by our method and baseline model (BreakRC). The green font represents the correct predicted answer, and the red font represents the incorrect.

- **Case 3**: This is an example of error propagation. For the first sub-question, the answer predicted by BreakRC is wrong, affecting the subsequent reasoning process, thus outputting the wrong final answer. Our method leverages the proposed Global Differentiable Learning Strategy to learn the optimal reasoning path by exploring the latent reasoning space. Thus it successfully learns to solve the intermediate reasoning process.

- **Case 4**: This is an example of semantic space mismatch. The reasoner in BreakRC is untrainable. Even if it correctly answers the first sub-question, it is also prone to errors in the subsequent reasoning process.

- **Case 5**: The reasoning type of sub-question 3 is comparison. It needs to select the entities that meet the requirements according to the results of the first and second sub-questions. The answer to the second sub-question predicted by BreakRC is wrong and coincidentally the same as the answer to the first sub-question, so the program randomly selects one as the final answer. Therefore, its interpretability is greatly affected.

- **Case 6**: This is an example of interpretability. Our method correctly completes all intermediate reasoning processes, showing good interpretability. In contrast, BreakRC correctly answers the second sub-question based on the wrong answer to the first sub-question. It may indicate that it does not learn to reason but instead predicts the answer based on biased information.

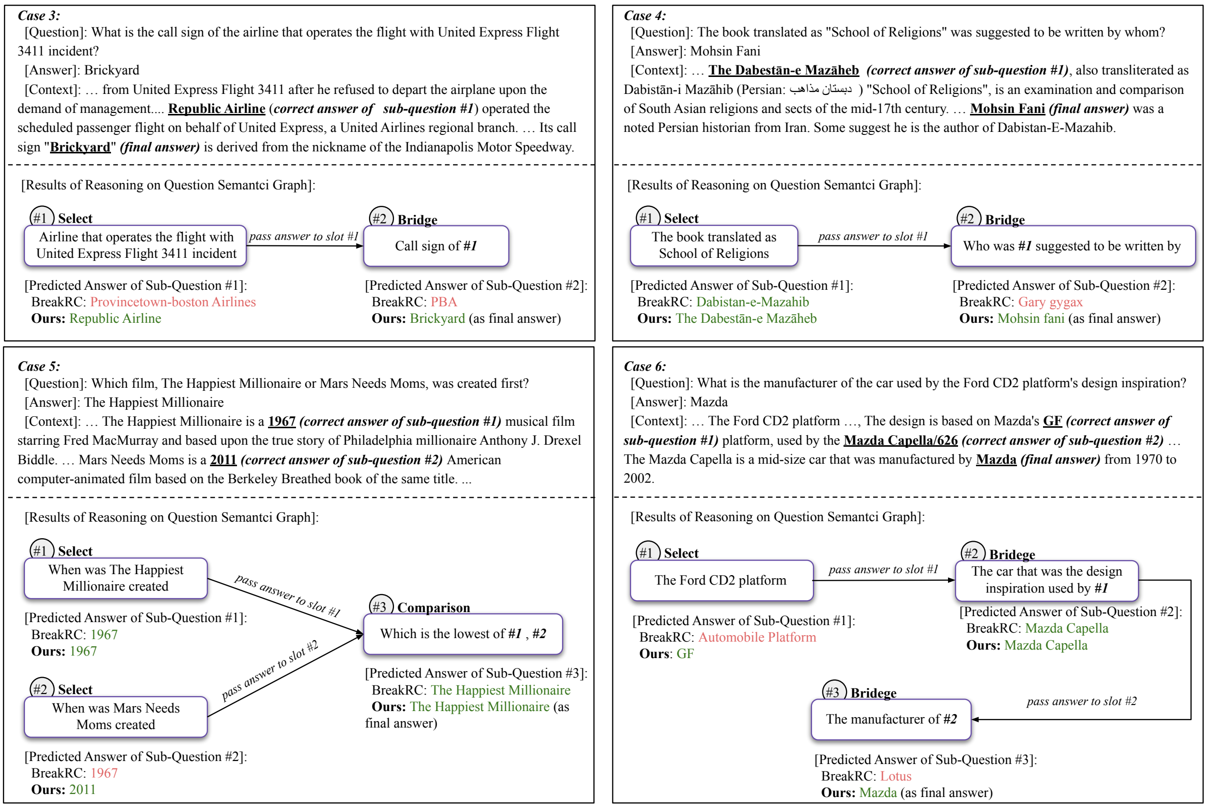


Figure 5: **Case Study**. The green font represents the correct predicted answer, and the red font represents the incorrect. Our method successfully learns the intermediate reasoning process and shows better interpretability.