A Three-step Method for Multi-Hop Inference Explanation Regeneration

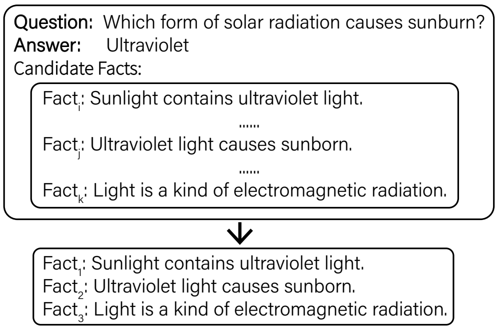
Yuejia Xiang[1], **Yunyan Zhang**[1], **Xiaoming Shi**[1], **Liu Bo**[2], **Wandi Xu**[3], **Chen Xi**[1]*
[1] Tencent Jarvis Lab
[2] SCMSIT, CETC SCRXX
[3] Northeastern University
{yuejiaxiang,yunyanzhang,xiaomingshi,jasonxchen}@tencent.com
liubo69@cetc.com.cn xuwandi@stumail.neu.edu.cn

Abstract

Multi-hop inference for explanation generation is to combine two or more facts to make an inference. The task focuses on generating explanations for elementary science questions. In the task, the relevance between the explanations and the QA pairs is of vital importance. To address the task, a three-step framework is proposed. Firstly, vector distance between two texts is utilized to recall the top-K relevant explanations for each question, reducing the calculation consumption. Then, a selection module is employed to choose those most relative facts in an autoregressive manner, giving a preliminary order for the retrieved facts. Thirdly, we adopt a re-ranking module to re-rank the retrieved candidate explanations with relevance between each fact and the QA pairs. Experimental results illustrate the effectiveness of the proposed framework with an improvement of 39.78% in NDCG over the official baseline.[1]

1 Introduction

Multi-hop inference for explanation generation (Jansen and Ustalov, 2020), aiming to combing two or more facts to make an inference and providing users with human-readable explanations, has shown significant potential and alluring technological value to improve medical or judicial systems. A typical application in natural language processing is **q**uestion **a**nswering tasks (QA). Multi-hop explanation generation for QA aims to retrieve multiple textual facts from pre-defined candidates (typically retrieved from different books, web pages, or other documents) for a given question-answer pair. Figure 1 shows an example. The input is a QA sample and candidate facts, and the task is designed to retrieve facts f_1, f_2, f_3, which contribute greatly to inferring the answer.



Question: Which form of solar radiation causes sunburn?
Answer: Ultraviolet
Candidate Facts:
Fact$_i$: Sunlight contains ultraviolet light.
......
Fact$_j$: Ultraviolet light causes sunborn.
......
Fact$_k$: Light is a kind of electromagnetic radiation.

Fact$_1$: Sunlight contains ultraviolet light.
Fact$_2$: Ultraviolet light causes sunborn.
Fact$_3$: Light is a kind of electromagnetic radiation.

Figure 1: An example of multi-hop inference for explanation generation.

Multi-hop explanation generation for QA suffers from a key issue: computationally prohibitory, which causes by unaffordable amount of fact combinations, especially when the number of facts required to perform an inference increases. Empirically speaking, the issue causes large drops in performance (Fried et al., 2015; Jansen et al., 2017) and limits the inference capacity (Khashabi et al., 2019). To solve the issue, previous works compute scores for facts in isolation, or by severely limiting the number of combinations of facts (Das et al., 2019; Banerjee, 2019; Chia et al., 2019). Cartuyvels et al. (2020) proposed a two-step inference algorithm for multi-hop explanation regeneration with a relevant fact recall step and an autoregressive fact selection step. In this way, the two-step algorithm prompts efficiency and accuracy.

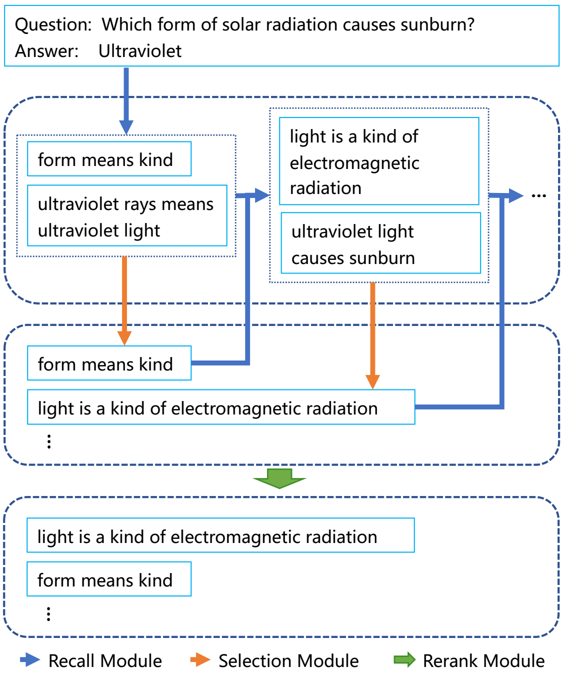
In the TextGraphs 2021 Shared Task, the relevance between the explanations and the QA pairs is of vital importance. However, the autogression selection process may hinder model's ability to recognize the relevance between each fact and QA. The main reason is that the autogression selection proceess emphasizes the relevance between QA and the retrieved facts, paying more attention on retrieved facts when there are many retrieved facts. As the example in Figure 2, the two-step algorithm fails to recognize the order of the retrieved two facts `form means kind` and `ultraviolet rays means ultraviolet light`. To ad-

[*] Corresponding author
[1] https://github.com/apricotxingya/tg2021task

Proceedings of the Fifteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-15), pages 171–175
June 11, 2021. ©2021 Association for Computational Linguistics

dress the problem, we propose a reranking module to fine-rank the results of the two-step method with the relevance between each fact and the QA pair. Then, we propose a three-step framework to solve the task: recall, selection and reranking, aiming to iteratively recall facts, select core facts, and then rerank retrieved core facts, respectively.

Experiments on the 2021 version of the task demonstrate the effectiveness of the proposed method, which achieves improvements of 39.78% in NDCG, in comparison with the official baseline.



Question: Which form of solar radiation causes sunburn?
Answer: Ultraviolet
form means kind
ultraviolet rays means ultraviolet light
light is a kind of electromagnetic radiation
ultraviolet light causes sunburn
...
form means kind
light is a kind of electromagnetic radiation
⋮
light is a kind of electromagnetic radiation
form means kind
⋮
Recall Module
Selection Module
Rerank Module

Figure 2: An overview of our method.

2 Method

The proposed framework is designed to predict a ranked list of facts inferring a QA sample, including three modules: a recall, selection and reranking module, as illustrated in Figure 2.

2.1 Recall Module

We stitch the text of Question and Answer together as q_a. We extract the roots of words in q_a and all e to reduce the number of different textual expressions caused by singular and plural tenses. For example, `cats` and `made` are modified to `cat` and `make`, respectively.

The recall module aims to iteratively recall facts with high relevance from the candidates. Formally, the recall module can be defined as a function: $f(q, a, f_1, \cdots, f_i, \cdots) : \mathbb{T}^{\mathbb{L}} \mapsto \mathbb{R}^{|C|}$, where q denotes the question token sequence, a denotes the answer token sequence, f_i denotes the recalled facts, $\mathbb{T}$ denotes the token set, $\mathbb{L}$ denotes the length of the sequence $[q, a, f_1, \cdots, f_i, \cdots]$, and C denotes the candidate set.

Specifically, we use the distances between *tf-idf* vectors to compute the distances between two texts. Let $s_i = [q_a, f_1^*, ..., f_i^*]$, where f_i^* is the ith best candidate selected from C_i by the selection module (refer to subsection 'Selection Module'). For the convenience of expression, we will write q_a as s_0.

First, we compute the Top$k$ of f_i with the smallest distance from q_a, forming C_1. Then we compute the top K f_i with the smallest distance from s_1 to form C_2. And so on.

2.2 Selection Module

We first normalize the score of each candidate fact to between 0 and 1. Since the score of s_i is 0 to 6, we divide the score by 6 to complete the normalization. Then we use Bert's own binary classification model to calculate the probability size $P(f_i|s_{i-1}) = BERT(f_i, s_{i-1})$ of each candidate f_i under s_{i-1}. Eventually, we will select a preferred choice with the highest probability as f_i^*.

In the prediction process, we keep Top$B$ candidates for each f_i for iteration and treat the currently used fact in Top$B$ as f_i^* in the iteration process. That is, for a q_a our method will generate $B^{(m-1)} * K$ fact links of length m. The probability of each fact link is obtained by chain decomposition to $P(q_a, f_1, ..., f_m) = P(f_1|s_0)P(f_2|s_1).......$ Our algorithm computes only sequences of length $m < M$. We finally sort the output sequences $(f_1^{r_1}, f_2^{r_1}, ..., f_m^{r_1}, f_1^{r_2}, f_2^{r_2}, ..., f_m^{r_2}, ...)$. where $f_i^{r_j}$ denotes the f_i of the fact link of sort jth. Then the output sequence is de-weighted by removing the non-first occurrence of the fact, to obtain the sequence O.

2.3 Rerank Module

The selection module hypothesizes that the predicted facts are always true and predicts the next fact given the previous facts. Such a process tends to suffer from error propagation since errors in the early modules cannot be corrected in later modules. Furthermore, one QA pair may have 20-30 relevant facts in average. The selection module may pay attention to QA at the beginning, but retrieved facts when there are many retrieved facts.

To relieve this problem, we introduce a rerank

Parameters	Value
Learning rate	2e-5
L2 weight decay	0.01
K	50
B	5
M	4
Epochs	4

Table 1: Hyperparameters

module, which computes the relevance between the q-a pair and each fact. Unlike the selection module, rerank module does not consider the correlations between pieces of facts, which is complementary to the selection module. Inspired by Natural Language Inference(NLI) task (Williams et al., 2017; Bowman et al., 2015), we cast q-a pairs as premises and candidate facts as hypotheses and identify whether a candidate fact is related to a q-a pair. Following the standard practice for sentence-pair tasks as in BERT (Devlin et al., 2018), we concatenate the q-a pair and the candidate fact with [SEP], prepend the sequence with [CLS], and feed the input to BERT. The representation for [CLS] is fed into a sigmoid layer for a binary classifier.

We select the top N candidate facts from the predicted results of the selection module and assign a score for every candidate fact according to its order. In the inference process, we calculate the probability for each candidate fact. If the probability is above a threshold, the original score of the specific candidate fact is added by a constant. After that, we rerank these candidate facts according to the updated scores. In this way, the model can obtain complementary results from both the selection module and rerank module.

3 Experiment

3.1 Data and Setting

In the 2021 version of the task, some facts are marked as deleted, duplicated, or low quality. We removed these facts, leaving 8983 facts in the end. The training dataset has 2206 data, the development dataset has 496 data, and the test dataset has 1664 data. This year, the sponsors include a very large dataset of approximately 250,000 expert-annotated relevancy ratings for facts ranked highly by baseline language models from previous years (e.g. BERT, RoBERTa).

We ran experiments on one 16GB Nvidia Tesla P100 GPU. The details of the experimental setup are shown in the table 1. The parameters not mentioned in the table use the default parameter settings of the Bert model.

method	NDCG
Baseline	50.10%
Recall+Selection	67.89%
Recall+Selection+Rerank	70.03%

Table 2: Main Results

3.2 Evaluation

The evaluation uses NDCG and the organizer provides a very large dataset of approximately 250,000 expert-annotated relevancy ratings for facts ranked highly by baseline language models from previous years (e.g. BERT, RoBERTa).

3.3 Baseline

The shared task data distribution includes a baseline that uses a term frequency model (tf.idf) to rank how likely table row sentences are to be a part of a given explanation. The performance of this baseline on the development partition is 0.513 NDCG.[2]

3.4 Main Results

It can be seen from the experimental results that our method is significantly better than the baseline model. At the same time, the Rerank module brings an improvement of 2.14%. The experimental results prove that our strategy of recall module and selection module is effective, which is 17.79% higher than the baseline. The rerank module also brings performance improvements as we expected, thus the rerank module make the results more focused on question is reasonable.

3.5 Case Study

We show three cases in Table 3. For each case, we show the top10 facts before the rerank module and after the rerank module. We can see from these cases that after applying the recall module and the section module, most of the top10 facts are related to the question and the answer. But there will be some irrelevant facts or less relevant facts that are ranked higher. And, after applying the rerank module, The ranking of facts with high references has generally been improved.

[2] https://github.com/cognitiveailab/tg2021task

Recall+Selection		Recall+Selection+Rerank	
Fact (Top10)	Ref.	Fact (Top10)	Ref.
the amount of daylight is greatest in the summer	6	the amount of daylight is greatest in the summer	6
summer is a kind of season	4	summer is a kind of season	4
daylight hours means time during daylight	0	summer has the most sunlight	6
the amount of daylight is least in the winter	2	increase means more	0
winter is a kind of season	2	daylight means sunlight	0
increase means more	0	summer is hemisphere tilted towards the sun	5
daylight means sunlight	0	high is similar to increase	0
summer is hemisphere tilted towards the sun	5	greatest means largest; highest	1
summer has the most sunlight	6	receiving sunlight synonymous absorbing sunlight	0
high is similar to increase	0	amount of daylight means length of daylight	0

(a) **Question:** *About how long does it take Earth to make one revolution around the Sun?* **Answer:** *summer*.

Recall+Selection		Recall+Selection+Rerank	
Fact (Top10)	Ref.	Fact (Top10)	Ref.
seals return the same beaches to give birth	4	if humans disturb animals; move to different location	6
a seal is a kind of animal	4	a seal is a kind of sea mammal	4
if humans disturb animals; move to different location	6	a seal is a kind of animal	4
a seal is a kind of sea mammal	4	seals return the same beaches to give birth	4
mammals give birth to live young	0	a mammal is a kind of animal	2
a mammal is a kind of animal	2	mammals give birth to live young	0
a beach is a kind of habitat; environment	4	a beach is a kind of location	4
a beach is a kind of location	4	a human is a kind of mammal	2
if something moves; something in different location	0	an environment is a kind of place	2
a human is a kind of mammal	2	an animal is a kind of living thing	2

(b) **Question:** *Female seals usually return to the same beaches year after year to give birth. If they are repeatedly disturbed by humans at those beaches, how will the seals most likely respond?* **Answer:** *They will give birth at different beaches.*

Recall+Selection		Recall+Selection+Rerank	
Fact (Top10)	Ref.	Fact (Top10)	Ref.
plucking; strumming a string cause that string to vibrate	6	matter; molecules vibrating can cause sound	5
a violin is a kind of musical instrument	4	plucking; strumming a string cause that string to vibrate	6
to cause means to be responsible for	0	a violin is a kind of musical instrument	4
musical instruments make sound when they are played	4	musical instruments make sound when they are played	4
matter; molecules vibrating can cause sound	5	a string is a kind of object	3
a string is a part of a guitar for producing sound	1	to cause means to be responsible for	0
a string is a kind of object	3	a string is a part of a guitar for producing sound	1
a guitar is a kind of musical instrument	0	a musical instrument is a kind of object	3
a musical instrument is a kind of object	3	make means produce	0
make means produce	0	vibrating matter can produce sound	5

(c) **Question:** *Bruce plays his violin every Friday night for the symphony. Before he plays, he plucks each string to see if his violin is in tune. Which is most responsible for the generation of sound waves from his violin?* **Answer:** *vibrations of the string.*

Table 3: Some cases in evaluation dataset.

3.6 Parameters in Rerank Module

Different number of parameter N in the rerank module can affect the performance to some extent, thus we report the performances using different parameter N. As shown in Table 4, the model achieves best performance with 70.03% NDCG score when N is 50. The NDCG score decreases when N is too low since the rerank module does not play its due role. Further more, a larger N is not necessary.

K	NDCG
5	67.64%
20	69.84%
30	70.03%
50	69.20%
100	68.13%

Table 4: Experiments on parameter of K

4 conclusion

We proposed our approach to the shared task on "Multi-hop Inference Explanation Regeneration". Our framework consists of three modules: a recall module, a selection module and a reranking module. The recall module retrieves top-K relevant facts using the distances between *tf-idf* vectors. Then an antoregressive fact selection module is applied to predict the next fact considering the retrived facts. Finally a rerank module is applied to correct the order. The proposed framework achieved an improvement of 39.78% over the official baseline.

References

Pratyay Banerjee. 2019. Asu at textgraphs 2019 shared task: Explanation regeneration using language models and iterative re-ranking. *ACL Workshop*.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Ruben Cartuyvels, Graham Spinks, and Marie-Francine Moens. 2020. Autoregressive reasoning over chains of facts with transformers. *ACL Workshop*.

Yew Ken Chia, Sam Witteveen, and Martin Andrews. 2019. Red dragon ai at textgraphs 2019 shared task: Language model assisted explanation generation. *ACL Workshop*.

Rajarshi Das, Ameya Godbole, Manzil Zaheer, Shehzaad Dhuliawala, and Andrew McCallum. 2019. Chains-of-reasoning at textgraphs 2019 shared task: Reasoning over chains of facts for explainable multi-hop inference. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 101–117.

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

Daniel Fried, Peter Jansen, Gustave Hahn-Powell, Mihai Surdeanu, and Peter Clark. 2015. Higher-order lexical semantic models for non-factoid answer reranking. *Transactions of the Association for Computational Linguistics*, 3:197–210.

Peter Jansen, Rebecca Sharp, Mihai Surdeanu, and Peter Clark. 2017. Framing qa as building and ranking intersentence answer justifications. *Computational Linguistics*, 43(2):407–449.

Peter Jansen and Dmitry Ustalov. 2020. TextGraphs 2020 Shared Task on Multi-Hop Inference for Explanation Regeneration. In *Proceedings of the Graph-based Methods for Natural Language Processing (TextGraphs)*, pages 85–97, Barcelona, Spain (Online). Association for Computational Linguistics.

Daniel Khashabi, Erfan Sadeqi Azer, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2019. On the capabilities and limitations of reasoning for natural language understanding. *arXiv preprint arXiv:1901.02522*.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.