Summarizing Chinese Medical Answer with Graph Convolution Networks and Question-focused Dual Attention

Ningyu Zhang[1,2], **Shumin Deng**[1,2], **Juan Li**[1,2], **Xi Chen**[4], **Wei Zhang**[2,3], **Huajun Chen**[1,2] *
[1] Zhejiang University
[2] AZFT Joint Lab for Knowledge Engine
[3] Alibaba Group
[4] Jarvis Lab Tencent
{zhangningyu,231sm,lijuan18,huajunsir}@zju.edu.cn
lantu.zw@alibaba-inc.com,jasonxchen@tencent.com

Abstract

Online search engines are a popular source of medical information for users, where users can enter questions and obtain relevant answers. It is desirable to generate answer summaries for online search engines, particularly summaries that can reveal direct answers to questions. Moreover, answer summaries are expected to reveal the most relevant information in response to questions; hence, the summaries should be generated with a focus on the question, which is a challenging topic-focused summarization task. In this paper, we propose an approach that utilizes graph convolution networks and question-focused dual attention for Chinese medical answer summarization. We first organize the original long answer text into a medical concept graph with graph convolution networks to better understand the internal structure of the text and the correlation between medical concepts. Then, we introduce a question-focused dual attention mechanism to generate summaries relevant to questions. Experimental results demonstrate that the proposed model can generate more coherent and informative summaries compared with baseline models.

1 Introduction

Online search engines (e.g., Google, Bing) have a wealth of fresh health-related information, which is appealing for users with medical questions. Users can enter questions to obtain relevant answers. However, most answers generated by domain experts are incredibly long, and some are even more than 512 words. It is intuitive to generate answer summaries, which will benefit both users and search engines. Such abstract resources are valuable to attract users' attention and encourage clicking and reading. Moreover, answer summaries are expected to reveal the most relevant information in response to questions; hence, the summaries should be focused on the question, which is a challenging topic-focused summarization task, as shown in Table 1.

Question
治疗心脏早搏有什么方法? How to treat the premature heartbeat?
Answer
心脏是人体上中枢环节，也是至关重要的几个部位之一，如果心脏异常跳动出现问题是很危险的。一般来说轻微患者是不需要治疗的，也可以使用安慰剂，严重患者可通过药物或射频消融缓解症状。下面我们来具体说一下治疗心脏早搏有什么方法。第一：... The heart is the central part of the human body, and it is also one of the vital parts. If the heart beats abnormally, it is very dangerous. Generally, mild patients do not require treatment and can get a placebo; serious patients can take medication or radiofrequency ablation to relieve symptoms. Let us talk about the methods available to treat premature heartbeat. First: ...
Summary
轻症患者不需要治疗，严重患者可采取药物治疗或射频消融治疗。 Mild patients do not need treatment; serious patients can take medication or radiofrequency ablation.

Table 1: Example of medical answer summarization task. Because the answer is extremely **long**, only parts of the sentences with **concept** words (blue) are shown.

(Zhou et al., 2006) first introduces answer summarization as an application of extractive summarization. (Deng et al., 2019) designs a question-enhanced pointer-generator network that exploits the correlation information between question-answer pairs to focus on the essential information when generating answer summaries. However, those approaches are trained and tested mainly on generic domain datasets, which are not straightforwardly applicable to the medical scenarios (Zhang et al., 2020). Moreover, there are still several non-trivial challenges for answer summarization in the medical domain as follows:

* Corresponding author: C.Hua(huajunsir@zju.edu.cn)

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 15–24
November 16 - 20, 2020.

- The original answers can be extremely **long**, which makes it intractable for vanilla sequence-to-sequence models.

- The most important parts of the answer not only rely on the keywords of the answer but should also be relative to the **question**. For example, for the question listed in Table 1, note that "治疗" (treat) is more important than "心脏" (heart) although the latter occurs more times in the answer.

- The answer focuses on different **concepts** of the same question, which makes the summaries quite diverse. For instance, a summary can consist of multiple plots, such as "轻微患者" (mild patient) and "严重患者" (serious patient).

Although the answer summarization task is not new, studies and corpus for the Chinese medical domain are *still limited*. To this end, we propose a graph convolution network with question-focused dual attention (**Q-GCN**) model to generate summaries. Our motivation is that *graph-based structure can better represent the correlation between diverse concepts in the answer and capture the plot of the whole text.* Specifically, we decompose the long answer text into several entities/keywords centered clusters of texts and represent the answer with a **medical concept graph**. Each vertex of the graph is formed with concept clusters regarding the entities/keywords. We calculate the edge between vertices via semantic relations between the vertices. Moreover, to enhance the relevance of the summaries regarding questions, we propose a question-focused dual attention mechanism to extract the primary information from the answer. We highlight our contributions as follows:

- We represent the long medical answer with a medical concept graph that explicitly organizes the text into concept-centered vertices.

- We propose a novel graph convolutional network with question-focused dual attention to generate summaries based on the medical concept graph.

- Experimental results on our collected large-scale Chinese question-answer-summary corpus (ChMedQA) and WikiHowQA demonstrate the efficacy of our approach.

2 Related Work

Text Summarization. Text summarization techniques can be classified into two categories: extractive and abstractive summarization. Extractive approaches regard summarization as a sentence classification (Nallapati et al., 2017) or a sequence labeling task (Cheng and Lapata, 2016) to select sentences from the article to form the summary, while abstractive approaches generally employ attention-based encoder-decoder models (Nallapati et al., 2016; See et al., 2017; Ye et al., 2020) to generate abstractive summaries. Our method is an abstract approach that can generate more fluent and coherent summaries. Answer summarization is first introduced by (Zhou et al., 2006) as an application of summarization. Subsequently, studies on answer summarization are still regarded as a separate summarization module in QA pipeline (Song et al., 2017). Moreover, query-based summarization methods (Singh et al., 2018) can also serve as a good solution for this task. (Deng et al., 2019) designs a question-enhanced pointer generator network to generate answer summaries.

There are few previous studies (Kogilavani and Balasubramanie, 2009) on medical answer summarization. As domain knowledge is helpful for generating coherent and informative summaries, previous approaches usually leverage ontologies (Kogilavani and Balasubramanie, 2009), concepts (Morales et al., 2008; Schulze and Neves, 2016) to summarize answers.

Graph Convolution Networks. Recently, graph convolution network (GCN) models have increasingly attracted attention (Zhang et al., 2019), which is beneficial for graph data modeling (Yin et al., 2019). Some existing literature such as SQL-to-Text (Xu et al., 2018), AMR-to-Text (Beck et al., 2018; Song et al., 2018; Zhao et al., 2018) use GCN for generating text. However, these approaches utilize the graph that already exists, and the input text is very short. We are faced with extreme long text. Recently, (Li et al., 2019) proposes to model a news article with a topic graph and utilizes the GCN to generate comments automatically. (Wang et al., 2020) presents a heterogeneous graph-based neural network for extractive summarization. Different from their approaches, we focus on the medical domain, and the generated summaries should be relevant to the input questions. To the best of our knowledge, we are the first to apply GCNs to the medical answer summarization task.

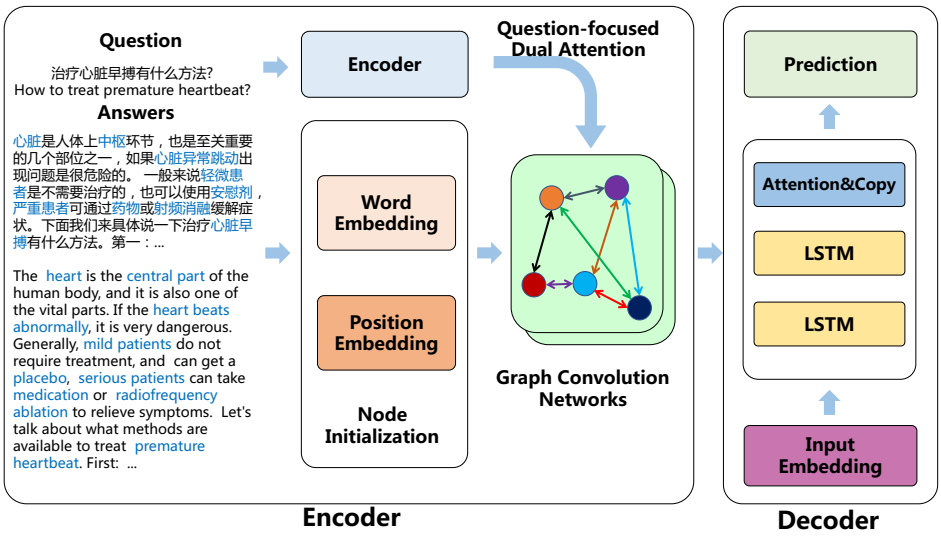

Figure 1: Architecture of our proposed model (**Q-GCN**). Best viewed in color.

3 Methodology

3.1 Problem Definition

Let A denote an answer containing several sentences $[s_1, s_2, s_3, s_4, \cdots, s_m]$, where s_i is the i-th sentence in the answer and Q denotes the input question. Our task is to generate an abstractive summary of A that is most relevant to the input question Q.

3.2 Framework

Our approach is shown in Figure 1 as an encoder-decoder framework. Specifically, our encoder aims to convert the original answer text to a medical concept graph. We propose question-focused dual attention to generate the summary sequence based on the graph and the encoded question.

3.3 Medical Concept Graph Construction

We construct our medical concept graph with the medical answer, as shown in Algorithm 1. For this paper, we define the **medical concepts** as *phrases/words of medical entities or keywords that are vital components of the text*. Note that the answers from online platforms have a considerable amount of noise. Some sentences in the answer are even irrelevant to the main question, for example, "感谢邀请" (Thanks for inviting.). Thus, given an input question Q and an answer A, we first perform word segmentation and then medical named entity recognition (NER) for the text with a pretrained BERT-CRF (Devlin et al., 2018) model. We then apply keyword extraction with TextRank (Mihalcea and Tarau, 2004) to obtain keywords. After

Algorithm 1 Medical Concept Graph Construction

Require: The answer text A, weight calculation function ϕ

```
Segment A into words
Use keyword detection and named entity recognition to generate concepts Ω
for sentence do
    if s_i contains ω ∈ Ω then
        Assign s_i to vectex v_k
    else
        Assign s_i to vertex v_empty
for vertex v_i and v_j do
    Obtain edge weight: w_{i,j} = φ(v_i, v_j)
```

that, we obtain the concepts of the answer, and we associate each sentence in the answer to its corresponding concepts. Specifically, we assign the sentence to the concept ω if ω appears in the sentence. Thus, a single sentence will be connected with more than one concept, which may implicitly indicate the correlation between concepts. We assign sentences that do not contain any of the concepts with an "empty" vertex. The sentences and the concept $\omega \in \Omega$ consist of the vertex v_k in the medical concept graph. We represent each vertex by the concatenation of the concept and sentence words in the answer.

The edges between vertices denoted as ϕ in Algorithm 1 can be constructed via a range of approaches. Whereas, the more sentences mention two concepts together, the closer those two con-

cepts are. To this end, we adopt a structure-based method in this paper. Specifically, if vertices v_i and v_j share at least one sentence, we then add an edge $e_{i,j}$ between them, and its weight is obtained with the number of shared sentences. It is also convenient to utilize content-based approaches, such as TF-IDF, to calculate the similarity.

3.4 Node Initialization

We encode the vertex in the medical concept graph with vector u_i. First, we utilize a multi-head self-attention based vertex encoder. This vertex encoder consists of two modules, namely the embedding module and the self-attention module. We adopt the regular word embedding of both words and concepts via a sharing embedding lookup table to represent **word information**. The regular words refer to words other than concept words. We also add absolute and relative positional embedding $p_i^{absolute}$, $p_i^{relative}$ to represent the **position information**. $p_i^{absolute}$ aims to encoder the absolute locations of the words and concepts in the answer. To better learn relative position embedding, we put the concept ω in front of the word sequence. In this way, the relative position embedding of the concept has the same embedding p_0. We add the word embedding w_i and position embedding $p_i^{absolute}$,$p_i^{relative}$ to get the final embedding u_i, formally:

$$u_i = w_i + p_i^{absolute} + p_i^{relative} \tag{1}$$

After that, we feed u_i into the self-attention module to obtain the hidden representation a_i of each word. The self-attention can explicitly model the interactions among words to capture the context of the vertex. We calculate the hidden representation of self-attention layer using Equation 2 to Equation 4, where Q, K, and V represent the query, key, and value vectors, respectively.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(QK^T\right)V \tag{2}$$

$$MultiHead(Q, K, V) = [head_1; \cdots ; head_h]\, W^o \tag{3}$$

$$head_i = Attention\left(QW_i^{Q_s}, QW_i^{K_s}, QW_i^{V_s}\right) \tag{4}$$

Whereas the concept ω is the vertex's vital information, we adopt the representation of the concept a_0 to represent the entire vertex.

3.5 Graph Convolution Networks

We feed the vertex v_i into a graph encoder after obtaining the hidden vectors, which explicitly models the graph structure of the constructed medical concept graph. We use an implementation of the GCN model following (Kipf and Welling, 2016). To be specific, we denote the adjacency matrix of the interaction graph as $A \in R^{N\times N}$, where $A_{ij} = w_{ij}$ (defined in § 3.3), and D is a diagonal matrix.

$$H^{l+1} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^l W^l\right) \tag{5}$$

$$\tilde{A} = A + I_N \tag{6}$$

where I_N is the identity matrix, $\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}$ is the normalized adjacency matrix, and W^l is a learnable weight matrix. We also add residual connections between layers to avoid over-smoothing.

$$g^{l+1} = H^{l+1} + H^l \tag{7}$$

$$g^{out} = \tanh\left(W_o g^K\right) \tag{8}$$

g^K is the output of the last layer of GCN. We add one feed forward layer to the final output of the GCN.

3.6 Question-focused Dual Attention

Because the question is a crucial signal, we propose a question-focused dual attention mechanism to emphasize those important vertex and de-emphasize irrelevant vertex. We utilize the transformer to generate the hidden output of the question q and calculate the **first attention** weights as:

$$\alpha_j = \frac{\exp\left(\delta\left(q, g_j\right)\right).}{\sum \exp\left(\delta\left(q, g_k\right)\right)} \tag{9}$$

where δ is the attention function, q is the hidden representation of question, and g_i is the final representation of vertex i. We utilize the recurrent neural network with attention. Given the output of the GCN $\langle v_0, v_1, \cdots, v_n\rangle$, and the initial state t_0, the decoder is able to generate a sequence of summery tokens $y_1, y_2, \cdots, y_m$. We calculate the **second attention** weights as:

$$t_i = RNN\left(t_{i-1}, c_{i-1}\right) \tag{10}$$

$$\beta_j = \frac{\exp\left(\delta\left(t_i, g_j\right)\right.}{\sum \exp\left(\delta\left(t_i, g_k\right)\right)} \tag{11}$$

where δ is the attention function, t_i is the hidden representation of state i, and g_i is the final representation of vertex i. We combine α_i and β_i with the following formula to obtain the final attention weight of each state:

$$\begin{aligned}\psi_i &= \text{softmax}\left(\gamma\alpha_i + (1-\gamma)\beta_i\right) \\ &= \frac{\exp\left(\gamma\alpha_i + (1-\gamma)\beta_i\right)}{\sum_{k\in[1,n]}\exp\left(\gamma\alpha_k + (1-\gamma)\beta_k\right)}\end{aligned} \tag{12}$$

Here, ψ_i denotes the final attention weight towards the graph vertex i, and $\gamma \in [0, 1]$ is a soft switch to adjust the importance of two attention weights, α_i and β_i. There are multiple ways to set the parameter γ. The simplest one is to treat γ as a hyper-parameter and manually adjust it to obtain the best performance. Alternatively, γ can also be learned by a neural network automatically. We select the latter approach because it adaptively assigns different values to γ on different scenarios and achieves better experimental results. We calculate γ by using the following formula:

$$\gamma = \sigma\left(w^T[\alpha;\beta] + b\right) \tag{13}$$

where vectors w and scalars b are learnable parameters, and σ is the sigmoid function. Ultimately, the final attention weights are employed to calculate a weighted sum of the state vectors, resulting in a semantic vector that represents the context:

$$c_i = \sum \psi_j v_j \tag{14}$$

Because the concepts v may appear in the summarization, which is vital information for the long answer, we use the copy mechanism following (Gu et al., 2016) by summing the predicted word token probability distribution with the attention distribution. The probability p_{copy} is dynamically calculated using context vector c_i and decoding hidden state t_i.

$$y_i = \text{softmax}\left(W_o\left(\tanh\left(W\left([t_i;c_i]\right)+b\right)\right)\right) \tag{15}$$

$$p_{copy} = \sigma\left(W_{copy}\left[t_i;c_i\right]\right) \tag{16}$$

$$p = \left(1 - p_{\text{copy}}\right) \times y + p_{\text{copy}} \times \psi \tag{17}$$

where W_o, W, W_{copy}, and b are all learnable parameters.

	ChMedQA		WikiHowQA	
	Number	Avg ALen	Number	Avg ALen
Train	80,000	534	142,063	520
Dev	10,000	583	18,909	548
Test	10,000	543	42,624	554

Table 2: Average length of answer (Avg ALen) and number of samples of the datasets (Number).

4 Experiments

We conduct three kinds of experiments: 1) automatic and manual evaluation with ablation study for Chinese medical answer summarization; 2) further experiments on WikiHowQA; 3) model analysis regarding question length, question-focused dual attention, and error analysis.

4.1 Dataset and Settings

We collect question and answer pairs from a popular Chinese search engine and split them into train/dev/test sets with a ratio of 8:1:1. We annotate 70% of the training set by a pretrained sentence ranking model[1] and the rest (train, dev, test) by crowdsourcing. We observe that the medical answer length is excessively long, which is challenging to the sequence-to-sequence model. To further analyze our approach's generalization, we conduct experiments on WikiHowQA[2] dataset that has extreme long answers. WikiHowQA is constructed based on the WikiHow dataset by (Deng et al., 2019) via filtering out those questions without answers or summaries and those answers with punctuation only. We detail the average length concerning the answer and the number of samples in both datasets in Table 2.

We utilize the 100-dimension pre-trained GloVe embeddings. The performance (F1) of medical NER and keyword extraction is **0.91** and **0.89**, respectively. We utilize Stanford CoreNLP[3] and TextRank (Mihalcea and Tarau, 2004) for the WikiHowQA dataset. We only utilize one layer GCN to ease the over-smoothing problem. We use a dropout rate of 0.2. We utilize Adam optimizer to train the parameters with the initial learning rate of 0.0005. We train our approach with four epochs.

[1]The sentence ranking model rank all sentences based on relativity regarding the question.

[2]https://github.com/dengyang17/wikihowQA

[3]https://stanfordnlp.github.io/CoreNLP

4.2 Baselines and Metrics

We compare the proposed method with the following baselines, including four extractive methods (Lead3, TextRank (Mihalcea and Tarau, 2004), NeuralSum (Cheng and Lapata, 2016), and NeuSum (Zhou et al., 2018)); two abstractive methods (Seq2Seq (Nallapati et al., 2016) and PGN (See et al., 2017)); and five query-based methods (BERT (Devlin et al., 2018), XLNet (Yang et al., 2019), PGN (See et al., 2017), SD2 (Nema et al., 2017)), biASBLSTM (Singh et al., 2018), and ASAS (Deng et al., 2019). For BERT/XLNet[4], we utilize the *abstractive summarization* schema as the encoder part is replaced with the BERT/XLNet encoder (question&answer) and the decoder is trained from scratch. We also compare variations of our approach: **w/o position** is the approach without position embedding; **w/o question** is the approach without question-focused dual attention; **w/o GCN** is the approach without GCN. We run each experiment five times and calculate the average performance. We use ROUGE F1 scores to evaluate the summarization methods.

4.3 Main Evaluation Results

Main results. The summarization results are listed in Table 3. We notice that XLNet achieves a higher ROUGE score than BERT, which may because XLNet is an autoregressive approach, while BERT is a denoising autoencoder approach that is not suitable for the generation. PGN outperforms XLNet, which may because there exist severe OOV problems in the medical domain, while PGN can copy words from the source text. We also observe that the question-enhanced approaches outperform all the state-of-the-art methods, which demonstrates the effectiveness of incorporating question information. Besides, by organizing the answer text into the concept graph, our approach further improves the results by a noticeable margin.

Ablation Study Results. We observe that the approach without position embedding has a slight performance decay, which demonstrates that position information is necessary. We also notice a severe performance drop when removing question-focused dual attention, which demonstrates that the question can not be ignored when summarizing answers. Besides, we observe a performance decay without GCN, which illustrates that graph-based structure can better represent the long text.

[4] https://github.com/huggingface/transformers

Models	R-1	R-2	R-L
LEAD3	26.5	7.2	22.3
ETXTRANK	26.6	7.5	23.5
NEURALSUM	27.1	8.1	25.5
NEUSUM	26.4	7.7	25.1
SEQ2SEQ	20.3	5.1	10.2
PGN	22.7	7,5	25.2
SD2	26.6	6.9	24.2
BIASBLSTM	24.7	6.9	22.7
Question-enhanced BERT	25.3	7.0	22.5
Question-enhanced XLNet	27.6	7.1	25.6
Question-enhanced PGN	27.7	7.9	25.8
Q-GCN	**29.0**	**8.2**	**27.0**
w/o position	27.9	7.9	25.9
w/o question	26.8	7.4	24.6
w/o GCN	27.2	7.0	25.1

Table 3: Main and ablation study results.

Models	Info	Conc	Read	Corr
NEURALSUM	3.66	3.12	3.11	3.01
Question-enhanced BERT	2.16	3.12	3.71	3.21
Question-enhanced XLNet	2.26	3.02	4.31	3.35
Question-enhanced PGN	2.71	3.51	4.01	2.95
Q-GCN	**3.70**	**3.99**	3.49	**3.61**

Table 4: Human evaluation results.

Human Evaluation. We conduct human evaluation to evaluate the generated answer summaries in four aspects: (1) **Informativity:** *How well does the summary capture the key information from the original answer?* (2) **Conciseness:** *How concise is the summary?* (3) **Readability:** *How fluent and coherent is the summary?* (4) **Correlatedness:** *How correlated are the summary and the given question?* We randomly sample 50 answers and generate their summaries by using five methods, namely NeuralSum, Question-enhanced BERT, Question-enhanced XLNet, Question-enhanced PGN, and the proposed approach. Three data annotators are requested to score each generated summary on a scale of 1 to 5 (higher the better).

Table 4 lists the human evaluation results, which shows that our approach consistently outperforms the other methods in all aspects. BERT and XLNet achieve relatively low scores in informativity and conciseness, which may be due to the failure of modeling long input text. However, BERT and XLNet generate more fluent summaries with higher readability scores, which may take advantage of the pre-trained language model.

To intuitively observe the advantage of the proposed method, we randomly select one example to show the results of the answer summary generation. As shown in Figure 5, the extractive method

Question
治疗心脏早搏有什么方法? How to treat premature heartbeat?
NeuralSum
一般来说轻微患者是不需要治疗的，也可以使用安慰剂，严重患者可通过药物或射频消融缓解症状。 Generally, mild patients do not require treatment and can a placebo; serious patients can take medication or radiofrequency ablation to relieve symptoms.
Question-enhanced PGN
患者可以采取药物治疗或射频消融治疗。 The patient should take medication or radiofrequency ablation.
Q-GCN
轻症患者不需要治疗，严重患者可采取药物治疗或射频消融治疗。 Mild patients do not require treatment; serious patients should take medication or radiofrequency ablation.

Table 5: Case study.

(e.g., NeuralSum) selects essential sentences from the original answer to form the answer summary, which still contains much insignificant or redundant information. The abstractive method (e.g., PGN) generates the answer summary from the vocabulary and the original answer, which may omit some concepts and essential information. Besides, we observe that some baseline models tend to generate *general* summaries such as "患者可以" (the patient should) when encountering long-tail concepts, which is similar to the dull response problem in dialogue (Du and Black, 2019). It significantly affects the performance scores of conciseness and correlatedness. To address these defects, our approach accounts for the information provided by the question and critical component from the medical concept graph with GCN, which is able to understand the main point of the answer rather than generating high-frequency phrases that are irrelevant or even useless to the given question. Noticeably, our model learns well to generate answer summaries that are highly related to the given questions, so there is a substantial improvement in terms of *informativity*, *conciseness*, and *correlatedness*.

However, we also notice that our approach receives a slightly lower *readability* score. We assume that this is because there exists a similar structure between different models in the decoder. We observe that our model can not distinguish between similar characters and repeatedly generates the same tokens sometimes. These phenomena are common in the natural language generation, which reveals the deficiency of understanding world knowledge. We leave this for future work.

Models	**R-1**	**R-2**	**R-L**
LEAD3	24.7	5.6	22.8
SEQ2SEQ	20.3	5.5	19.8
NEURALSUM	27.8	6.8	25.1
ASAS	27.8	8.2	25.9
Q-GCN	**28.3**	**8.8**	**26.5**

Table 6: Evaluation results on WikiHowQA.

Models	**Info**	**Conc**	**Read**	**Corr**
NEURALSUM	3.60	2.70	3.22	3.24
ASAS	3.67	3.88	3.59	3.71
Q-GCN	3.66	**4.31**	**3.60**	**4.71**

Table 7: Human evaluation results on WikiHowQA.

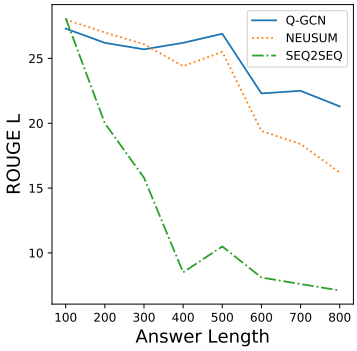

Figure 2: Model performance *#answer length*.

4.4 Evaluation on WikiHowQA

From Table 6, we observe: 1) our approach still performs better than all baselines, which demonstrates that our approach can apply to the general domain; 2) we notice that the performance improvements are relatively smaller. We think this may because in the general domain, in addition to entities and keywords, there also exist some verb phrases which may reveal the critical point in the answers. From the Table 7 we observe: 1) our approach performs better than all baselines in human evaluation except the informativity, which may be caused by the negation of some context in the answers; 2) we notice the significant performance improvement in conciseness and correlatedness, which further proves that the graph-structure can better understand the main point of the answer.

4.5 Analysis

Length of Answer. To validate the effectiveness of the proposed method on long-sentence answer summarization, we sample the test set according to the length of the answer. As shown in Figure 2, we compare our approach with two baseline

methods, SEQ2SEQ, and NEUSUM, by measuring the ROUGE-L. We observe that our approach is more efficient, especially for long answers. For answers that are shorter than 100 words, SEQ2SEQ and NEUSUM are marginally better than our approach, which indicates that the summary may have lost some information for short answers. However, the performance of these two methods deteriorates with an increase in the answer length, whereas our approach maintains excellent stability. In summary, **explicitly organizing the text into a graph-structure can better represent long text.**

Question-Focused Dual Attention. To evaluate whether our question guides the procedure of answer summary generation, we deliberately change the question with the same answer and obtain different summarization results, as shown in Table 8. We observe that our model can control the summarization of answers with different questions, indicating the efficacy of question-focused dual attention. For example, by changing the question from "注意什么" (pay attention) to "吃什么水果" (what fruits to eat), we generate results which directly address the question. However, when changing the original question to a question that cannot be summarized (cannot find an answer regarding the question), our approach fails to generate concise summaries. We also observe that our approach without question-focused dual attention generates *trivial summaries*, which include redundant information and miss the key points relevant to the question. Those observations demonstrate that **question-focused dual attention can enhance generating summaries relevant to questions.**

Error Analysis. We conduct an error analysis of our approach. We first random sample 100 test instances with **wrong entities/keywords**. Surprisingly, we observe that 80% of them generate coherent and informative summaries, which shows that incorrect entities/keywords have little influences on the quality of summarization. We further analyze the wrong instances and divide them into five categories. **First**, our model can generate fluency summaries with significantly long sentences but may fail to generate well with some short answers. **Second**, our model cannot handle time and numbers. For example, when summarizing the answer "正常不外用药物，是三天左右就开始自行消肿。..." (Normally, do not need to take medications and will begin to swell on its own in about three days ...) with the question "被蜜蜂蛰了几天能好" (How many days can I recover if stung by a bee), our model cannot provide reasonable summaries because it does not understand what "几天" (how many days) is. **Third**, our model is vulnerable, to some extent, to adversarial attacking, such as adding a negative modifier "不" (not) in the question; our model fails to understand the true meaning and yields poor results. **Finally**, we find that our model is sensitive to typos and some extreme long-tail terminologies, such as "胃腾" (stoma chache) and "阴超" (vaginal B-ultrasound).

Question1 怀孕三个月吃东西注意什么? What to eat in the third month of pregnancy?
Question2 怀孕三个月不能吃什么水果? What fruits can't I eat in the third month of pregnancy?
Question3 怀孕六个月不能吃什么? What can't I eat in the sixth month of pregnancy?
Q-GCN1 多吃维生素、饱和脂肪酸较多的食物，禁食寒凉水果。 You should eat more foods with vitamins and saturated fatty acids; do not eat cold fruits.
w/o question 怀孕要定期做孕检，注意营养，多吃饱和脂肪酸。 You should take regular pregnancy tests and pay attention to nutrition, eat more foods with saturated fatty acids.
Q-GCN2 水果如龙眼、山楂等。 Fruits such as longan, hawthorn, etc.
w/o question 怀孕要定期做孕检，注意营养，多多多吃包含维生素的食物。 You should take regular pregnancy tests and pay attention to nutrition, eat more more more food with vitamins.
Q-GCN3 注意营养，多吃维生素、饱和脂肪酸较多的食物，不能吃寒凉水果。 Pay attention to nutrition, eat more vitamins, saturated fatty acid foods, do not eat cold fruits.
w/o question 怀孕要定期做孕检，注意营养。 You should take regular pregnancy tests and pay attention to nutrition.

Table 8: Answer summaries of different questions.

5 Conclusion and Future Work

In this paper, we propose an approach of graph convolution network with question-focused dual attention to generate Chinese answer summaries. Experimental results indicate that our model can summarize more coherently and informatively, thereby showing that organizing long text with a graph structure is beneficial and question-focused dual attention further improves the informativeness and correlation. In the future, we plan to 1) exploit knowledge such as commonsense to generate logical summaries; 2) investigate efficient methodologies to model the correlation between concepts;

3) apply our approach to similar applications such multiple document summarization.

Acknowledgments

We want to express gratitude to the anonymous reviewers for their hard work and kind comments, which will further improve our work in the future. This work is funded by NSFC91846204/SQ2018YFC000004. This work is also funded by 2018YFB1402800, Alibaba CangJingGe (Knowledge Engine) Research Plan.

References

Daniel Beck, Gholamreza Haffari, and Trevor Cohn. 2018. Graph-to-sequence learning using gated graph neural networks. *arXiv preprint arXiv:1806.09835*.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252*.

Yang Deng, Wai Lam, Yuexiang Xie, Daoyuan Chen, Yaliang Li, Min Yang, and Ying Shen. 2019. Joint learning of answer selection and answer summary generation in community question answering. *arXiv preprint arXiv:1911.09801*.

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*.

Wenchao Du and Alan W Black. 2019. Boosting dialog response generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 38–43.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

AA Kogilavani and P Balasubramanie. 2009. Ontology enhanced clustering based summarization of medical documents. *International Journal of Recent Trends in Engineering*, 1(1):546.

Wei Li, Jingjing Xu, Yancheng He, Shengli Yan, Yunfang Wu, et al. 2019. Coherent comment generation for chinese articles with a graph-to-sequence model. *arXiv preprint arXiv:1906.01231*.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Laura Plaza Morales, Alberto Díaz Esteban, and Pablo Gervás. 2008. Concept-graph based biomedical automatic summarization using ontologies. In *Proceedings of the 3rd textgraphs workshop on graph-based algorithms for natural language processing*, pages 53–56. Association for Computational Linguistics.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Preksha Nema, Mitesh Khapra, Anirban Laha, and Balaraman Ravindran. 2017. Diversity driven attention model for query-based abstractive summarization. *arXiv preprint arXiv:1704.08300*.

Frederik Schulze and Mariana Neves. 2016. Entity-supported summarization of biomedical abstracts. In *Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016)*, pages 40–49.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Mittul Singh, Arunav Mishra, Youssef Oualil, Klaus Berberich, and Dietrich Klakow. 2018. Long-span language models for query-focused unsupervised extractive text summarization. In *European Conference on Information Retrieval*, pages 657–664. Springer.

Hongya Song, Zhaochun Ren, Shangsong Liang, Piji Li, Jun Ma, and Maarten de Rijke. 2017. Summarizing answers in non-factoid community question-answering. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 405–414. ACM.

Linfeng Song, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2018. A graph-to-sequence model for amr-to-text generation. *arXiv preprint arXiv:1805.02473*.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. Heterogeneous graph neural networks for extractive document summarization. *arXiv preprint arXiv:2004.12393*.

Kun Xu, Lingfei Wu, Zhiguo Wang, Yansong Feng, Michael Witbrock, and Vadim Sheinin. 2018. Graph2seq: Graph to sequence learning with attention-based neural networks. *arXiv preprint arXiv:1804.00823*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Hongbin Ye, Ningyu Zhang, Shumin Deng, Mosha Chen, Chuanqi Tan, Fei Huang, and Huajun Chen. 2020. Contrastive triple extraction with generative transformer. *arXiv preprint arXiv:2009.06207*.

Yongjing Yin, Linfeng Song, Jinsong Su, Jiali Zeng, Chulun Zhou, and Jiebo Luo. 2019. Graph-based neural sentence ordering. *arXiv preprint arXiv:1912.07225*.

Ningyu Zhang, Shumin Deng, Zhanlin Sun, Guanying Wang, Xi Chen, Wei Zhang, and Huajun Chen. 2019. Long-tail relation extraction via knowledge graph embeddings and graph convolution networks. In *Proceedings of NAACL*.

Ningyu Zhang, Qianghuai Jia, Kangping Yin, Liang Dong, Feng Gao, and Nengwei Hua. 2020. Conceptualized representation learning for chinese biomedical text mining. *arXiv preprint arXiv:2008.10813*.

Guoshuai Zhao, Jun Li, Lu Wang, Xueming Qian, and Yun Fu. 2018. Graphseq2seq: Graph-sequence-to-sequence for neural machine translation.

Liang Zhou, Chin-Yew Lin, and Eduard H Hovy. 2006. Summarizing answers for complicated questions. In *LREC*, pages 737–740. Citeseer.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. *arXiv preprint arXiv:1807.02305*.