Improving Question Generation With to the Point Context

Jingjing Li[1]*   **Yifan Gao**[1]*   **Lidong Bing**[2]   **Irwin King**[1]   **Michael R. Lyu**[1]

[1] Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[2] R&D Center Singapore, Machine Intelligence Technology
Alibaba DAMO Academy
[1]{lijj, yfgao, king, lyu}@cse.cuhk.edu.hk [2]l.bing@alibaba-inc.com

Abstract

Question generation (QG) is the task of generating a question from a reference sentence and a specified answer within the sentence. A major challenge in QG is to identify answer-relevant context words to finish the declarative-to-interrogative sentence transformation. Existing sequence-to-sequence neural models achieve this goal by proximity-based answer position encoding under the intuition that neighboring words of answers are of high possibility to be answer-relevant. However, such intuition may not apply to all cases especially for sentences with complex answer-relevant relations. Consequently, the performance of these models drops sharply when the relative distance between the answer fragment and other non-stop sentence words that also appear in the ground truth question increases. To address this issue, we propose a method to jointly model the unstructured sentence and the structured answer-relevant relation (extracted from the sentence in advance) for question generation. Specifically, the structured answer-relevant relation acts as the to the point context and it thus naturally helps keep the generated question to the point, while the unstructured sentence provides the full information. Extensive experiments show that to the point context helps our question generation model achieve significant improvements on several automatic evaluation metrics. Furthermore, our model is capable of generating diverse questions for a sentence which conveys multiple relations of its answer fragment.

1 Introduction

Question Generation (QG) is the task of automatically creating questions from a range of inputs, such as natural language text (Heilman and Smith, 2010), knowledge base (Serban et al., 2016) and

---
* These two authors contributed equally.

Sentence: The daily mean temperature in January, the area's coldest month, is 32.6 °F (0.3 °C); however, temperatures usually drop to 10 °F (-12 °C) several times per winter and reach 50 °F (10 °C) several days each winter month.
Reference Question: What is New York City 's daily January mean temperature in degrees celsius ?
Baseline Prediction: What is the coldest temperature in Celsius ?
Structured Answer-relevant Relation: (The daily mean temperature in January; is; 32.6 °F (0.3 °C))

Figure 1: An example SQuAD question with the baseline's prediction. The answer ("0.3 °C") is highlighted.

image (Mostafazadeh et al., 2016). QG is an increasingly important area in NLP with various application scenarios such as intelligence tutor systems, open-domain chatbots and question answering dataset construction. In this paper, we focus on question generation from reading comprehension materials like SQuAD (Rajpurkar et al., 2016). As shown in Figure 1, given a sentence in the reading comprehension paragraph and the text fragment (i.e., the answer) that we want to ask about, we aim to generate a question that is asked about the specified answer.

Question generation for reading comprehension is firstly formalized as a declarative-to-interrogative sentence transformation problem with predefined rules or templates (Mitkov and Ha, 2003; Heilman and Smith, 2010). With the rise of neural models, Du et al. (2017) propose to model this task under the sequence-to-sequence (Seq2Seq) learning framework (Sutskever et al., 2014) with attention mechanism (Luong et al., 2015). However, question generation is a one-to-many sequence generation problem, i.e., several aspects can be asked given a sentence. Zhou et al. (2017) propose the answer-aware question generation setting which assumes the answer, a contiguous span inside the input sentence, is al-

Distance	B1	B2	B3	B4	MET	R-L
0~10 (72.8% of #)	45.25	30.31	22.06	16.54	21.54	46.26
>10 (27.2% of #)	35.67	21.72	14.82	10.46	16.72	37.63

Table 1: Performance for the average relative distance between the answer fragment and other non-stop sentence words that also appear in the ground truth question. (Bn: BLEU-n, MET: METEOR, R-L: ROUGE-L)

ready known before question generation. To capture answer-relevant words in the sentence, they adopt a BIO tagging scheme to incorporate the answer position embedding in Seq2Seq learning. Furthermore, Sun et al. (2018) propose that tokens close to the answer fragments are more likely to be answer-relevant. Therefore, they explicitly encode the relative distance between sentence words and the answer via position embedding and position-aware attention.

Although existing proximity-based answer-aware approaches achieve reasonable performance, we argue that such intuition may not apply to all cases especially for sentences with complex structure. For example, Figure 1 shows such an example where those approaches fail. This sentence contains a few facts and due to the parenthesis (i.e. "the area's coldest month"), some facts intertwine: "The daily mean temperature in January is 0.3°C" and "January is the area's coldest month". From the question generated by a proximity-based answer-aware baseline, we find that it wrongly uses the word "coldest" but misses the correct word "mean" because "coldest" has a shorter distance to the answer "0.3°C".

In summary, their intuition that "the neighboring words of the answer are more likely to be answer-relevant and have a higher chance to be used in the question" is not reliable. To quantitatively show this drawback of these models, we implement the approach proposed by Sun et al. (2018) and analyze its performance under different relative distances between the answer and other non-stop sentence words that also appear in the ground truth question. The results are shown in Table 1. We find that the performance drops at most 36% when the relative distance increases from "0 ~ 10" to "> 10". In other words, when the useful context is located far away from the answer, current proximity-based answer-aware approaches will become less effective, since they overly emphasize neighboring words of the answer.

To address this issue, we extract the structured answer-relevant relations from sentences and pro-

pose a method to jointly model such structured relation and the unstructured sentence for question generation. The structured answer-relevant relation is likely to be to the point context and thus can help keep the generated question to the point. For example, Figure 1 shows our framework can extract the right answer-relevant relation ("The daily mean temperature in January", "is", "32.6°F (0.3°C)") among multiple facts. With the help of such structured information, our model is less likely to be confused by sentences with a complex structure. Specifically, we firstly extract multiple relations with an off-the-shelf Open Information Extraction (OpenIE) toolbox (Saha and Mausam, 2018), then we select the relation that is most relevant to the answer with carefully designed heuristic rules.

Nevertheless, it is challenging to train a model to effectively utilize both the unstructured sentence and the structured answer-relevant relation because both of them could be noisy: the unstructured sentence may contain multiple facts which are irrelevant to the target question, while the limitation of the OpenIE tool may produce less accurate extracted relations. To explore their advantages simultaneously and avoid the drawbacks, we design a gated attention mechanism and a dual copy mechanism based on the encoder-decoder framework, where the former learns to control the information flow between the unstructured and structured inputs, while the latter learns to copy words from two sources to maintain the informativeness and faithfulness of generated questions.

In the evaluations on the SQuAD dataset, our system achieves significant and consistent improvement as compared to all baseline methods. In particular, we demonstrate that the improvement is more significant with a larger relative distance between the answer and other non-stop sentence words that also appear in the ground truth question. Furthermore, our model is capable of generating diverse questions for a single sentence-answer pair where the sentence conveys multiple relations of its answer fragment.

Sentence: Beyoncé received critical acclaim and commercial success, selling <mark>one million</mark> digital copies worldwide in six days; The New York Times noted the album's unconventional, unexpected release as significant.
N-ary Relations:
0.85 (Beyoncé; received commercial success selling; one million digital copies worldwide; in six days)
0.92 (The New York Times; noted; the album's unconventional, unexpected release as significant)

Sentence: The daily mean temperature in January, the area's coldest month, is 32.6 °F (<mark>0.3 °C</mark>); however, temperatures usually drop to 10 °F (-12 °C) several times per winter and reach 50 °F (10 °C) several days each winter month.
N-ary Relations:
0.95 (The daily mean temperature in January; is; 32.6 °F (0.3 °C))
0.94 (temperatures; drop; to 10 °F (12 °C); several times per winter; usually)
0.90 (temperatures; reach; 50 °F)

Figure 2: Examples for n-ary extractions from sentences using OpenIE. Confidence scores are shown at the beginning of each relation. Answers are highlighted in sentences. Waved relations are selected according to our criteria in Section 2.2.

2 Framework Description

In this section, we first introduce the task definition and our protocol to extract structured answer-relevant relations. Then we formalize the task under the encoder-decoder framework with gated attention and dual copy mechanism.

2.1 Problem Definition

We formalize our task as an answer-aware Question Generation (QG) problem (Zhao et al., 2018), which assumes answer phrases are given before generating questions. Moreover, answer phrases are shown as text fragments in passages. Formally, given the sentence S, the answer A, and the answer-relevant relation M, the task of QG aims to find the best question $\overline{Q}$ such that,

$$\overline{Q} = \arg\max_{Q} \text{Prob}(Q|S, A, M), \qquad (1)$$

where A is a contiguous span inside S.

2.2 Answer-relevant Relation Extraction

We utilize an off-the-shelf toolbox of OpenIE [1] to the derive structured answer-relevant relations from sentences as to the point contexts. Relations extracted by OpenIE can be represented either in a triple format or in an n-ary format with several secondary arguments, and we employ the latter to

---

[1] http://openie.allenai.org/

	Sentence	Answer-relevant Relation
Avg. length	32.46	13.04
# overlapped words	2.87	1.86
Copy ratio	8.85%	14.26%

Table 2: Comparisons between sentences and answer-relevant relations. Overlapped words are those non-stop tokens co-occurring in the source (sentence/relation) and the target question. Copy ratio means the proportion of source tokens that are used in the question.

keep the extractions as informative as possible and avoid extracting too many similar relations in different granularities from one sentence. We join all arguments in the extracted n-ary relation into a sequence as our to the point context. Figure 2 shows n-ary relations extracted from OpenIE. As we can see, OpenIE extracts multiple relations for complex sentences. Here we select the most informative relation according to three criteria in the order of descending importance: (1) having the maximal number of overlapped tokens between the answer and the relation; (2) being assigned the highest confidence score by OpenIE; (3) containing maximum non-stop words. As shown in Figure 2, our criteria can select answer-relevant relations (waved in Figure 2), which is especially useful for sentences with extraneous information. In rare cases, OpenIE cannot extract any relation, we treat the sentence itself as the to the point context.

Table 2 shows some statistics to verify the intuition that the extracted relations can serve as more to the point context. We find that the tokens in relations are 61% more likely to be used in the target question than the tokens in sentences, and thus they are more to the point. On the other hand, on average the sentences contain one more question token than the relations (1.86 v.s. 2.87). Therefore, it is still necessary to take the original sentence into account to generate a more accurate question.

2.3 Our Proposed Model

Overview. As shown in Figure 3, our framework consists of four components (1) Sentence Encoder and Relation Encoder, (2) Decoder, (3) Gated Attention Mechanism and (4) Dual Copy Mechanism. The sentence encoder and relation encoder encode the unstructured sentence and the structured answer-relevant relation, respectively. To select and combine the source information from the two encoders, a gated attention mechanism is
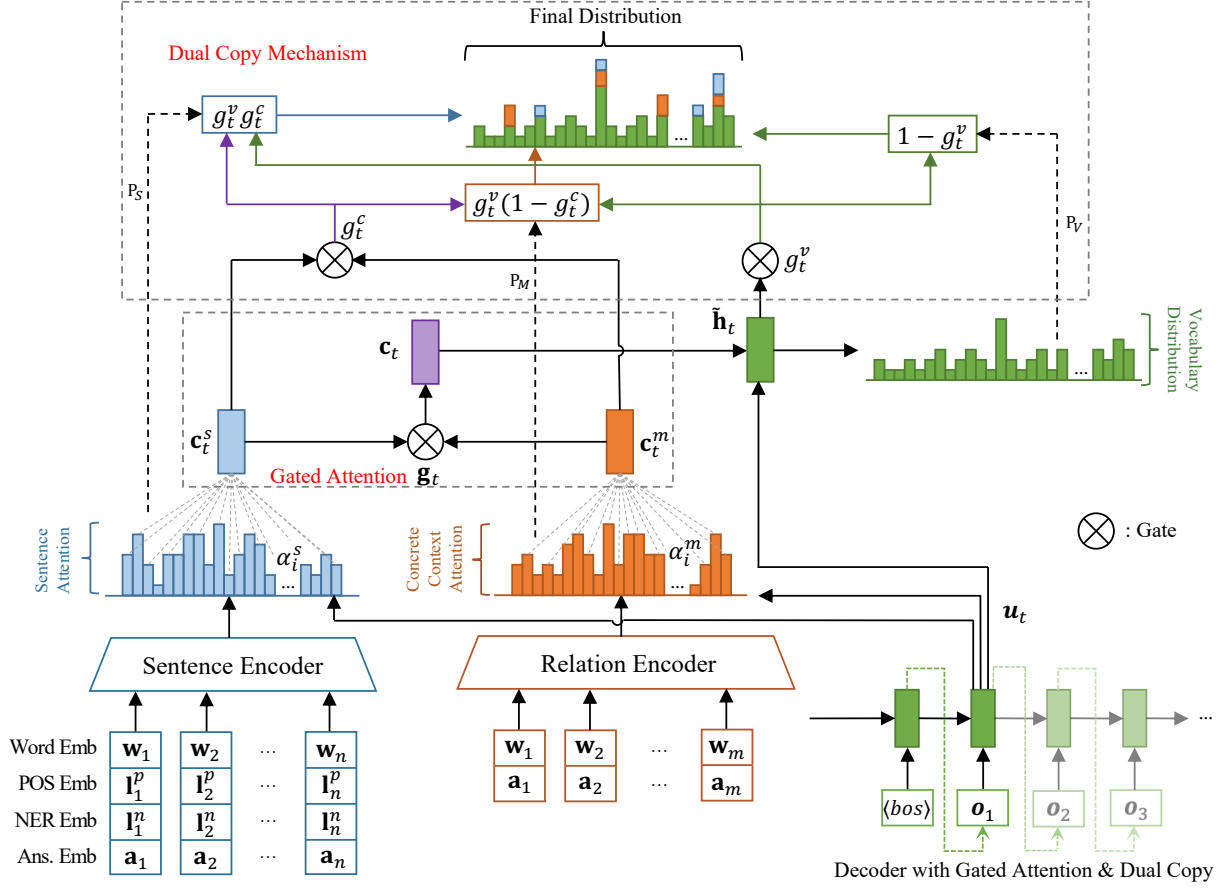
Figure 3: The framework of our proposed model. *(Best viewed in color)*

employed to jointly attend both contextualized information sources, and a dual copy mechanism copies words from either the sentence or the relation.

Answer-aware Encoder. We employ two encoders to integrate information from the unstructured sentence S and the answer-relevant relation M separately. Sentence encoder takes in feature-enriched embeddings including word embeddings $\mathbf{w}$, linguistic embeddings $\mathbf{l}$ and answer position embeddings $\mathbf{a}$. We follow (Zhou et al., 2017) to transform POS and NER tags into continuous representation ($\mathbf{l}^p$ and $\mathbf{l}^n$) and adopt a BIO labelling scheme to derive the answer position embedding (B: the first token of the answer, I: tokens within the answer fragment except the first one, O: tokens outside of the answer fragment). For each word w_i in the sentence S, we simply concatenate all features as input: $\mathbf{x}_i^s = [\mathbf{w}_i; \mathbf{l}_i^p; \mathbf{l}_i^n; \mathbf{a}_i]$. Here $[\mathbf{a}; \mathbf{b}]$ denotes the concatenation of vectors $\mathbf{a}$ and $\mathbf{b}$.

We use bidirectional LSTMs to encode the sentence $(\mathbf{x}_1^s, \mathbf{x}_2^s, ..., \mathbf{x}_n^s)$ to get a contextualized rep-

resentation for each token:

$$\overrightarrow{\mathbf{h}}_i^s = \overrightarrow{\mathrm{LSTM}}(\overrightarrow{\mathbf{h}}_{i-1}^s, \mathbf{x}_i^s), \ \overleftarrow{\mathbf{h}}_i^s = \overleftarrow{\mathrm{LSTM}}(\overleftarrow{\mathbf{h}}_{i+1}^s, \mathbf{x}_i^s),$$

where $\overrightarrow{\mathbf{h}}_i^s$ and $\overleftarrow{\mathbf{h}}_i^s$ are the hidden states at the i-th time step of the forward and the backward LSTMs. The output state of the sentence encoder is the concatenation of forward and backward hidden states: $\mathbf{h}_i^s = [\overrightarrow{\mathbf{h}}_i^s; \overleftarrow{\mathbf{h}}_i^s]$. The contextualized representation of the sentence is $(\mathbf{h}_1^s, \mathbf{h}_2^s, ..., \mathbf{h}_n^s)$.

For the relation encoder, we firstly join all items in the n-ary relation M into a sequence. Then we only take answer position embedding as an extra feature for the sequence: $\mathbf{x}_i^m = [\mathbf{w}_i; \mathbf{a}_i]$. Similarly, we take another bidirectional LSTMs to encode the relation sequence and derive the corresponding contextualized representation $(\mathbf{h}_1^m, \mathbf{h}_2^m, ..., \mathbf{h}_n^m)$.

Decoder. We use an LSTM as the decoder to generate the question. The decoder predicts the word probability distribution at each decoding timestep to generate the question. At the t-th timestep, it reads the word embedding $\mathbf{w}_t$ and the hidden state $\mathbf{u}_{t-1}$ of the previous timestep to gen-

erate the current hidden state:

$$\mathbf{u}_t = \text{LSTM}(\mathbf{u}_{t-1}, \mathbf{w}_t). \tag{2}$$

Gated Attention Mechanism. We design a gated attention mechanism to jointly attend the sentence representation and the relation representation. For sentence representation $(\mathbf{h}_1^s, \mathbf{h}_2^s, ..., \mathbf{h}_n^s)$, we employ the Luong et al. (2015)'s attention mechanism to obtain the sentence context vector $\mathbf{c}_t^s$,

$$a_{t,i}^s = \frac{\exp(\mathbf{u}_t^\top \mathbf{W}_a \mathbf{h}_i^s)}{\sum_j \exp(\mathbf{u}_t^\top \mathbf{W}_a \mathbf{h}_j^s)}, \ \mathbf{c}_t^s = \sum_i a_{t,i}^s \mathbf{h}_i^s,$$

where $\mathbf{W}_a$ is a trainable weight. Similarly, we obtain the vector $\mathbf{c}_t^m$ from the relation representation $(\mathbf{h}_1^m, \mathbf{h}_2^m, ..., \mathbf{h}_n^m)$. To jointly model the sentence and the relation, a gating mechanism is designed to control the information flow from two sources:

$$\mathbf{g}_t = \text{sigmoid}(\mathbf{W}_g[\mathbf{c}_t^s; \mathbf{c}_t^m]), \tag{3}$$
$$\mathbf{c}_t = \mathbf{g}_t \odot \mathbf{c}_t^s + (\mathbf{1} - \mathbf{g}_t) \odot \mathbf{c}_t^m, \tag{4}$$
$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_h[\mathbf{u}_t; \mathbf{c}_t]), \tag{5}$$

where $\odot$ represents element-wise dot production and $\mathbf{W}_g, \mathbf{W}_h$ are trainable weights. Finally, the predicted probability distribution over the vocabulary V is computed as:

$$\text{P}_V = \text{softmax}(\mathbf{W}_V \tilde{\mathbf{h}}_t + \mathbf{b}_V), \tag{6}$$

where $\mathbf{W}_V$ and $\mathbf{b}_V$ are parameters.

Dual Copy Mechanism. To deal with the rare and unknown words, the decoder applies the pointing method (See et al., 2017; Gu et al., 2016; Gulcehre et al., 2016) to allow copying a token from the input sentence at the t-th decoding step. We reuse the attention score α_t^s and α_t^m to derive the copy probability over two source inputs:

$$\text{P}_S(w) = \sum_{i:w_i=w} \alpha_{t,i}^s, \ \text{P}_M(w) = \sum_{i:w_i=w} \alpha_{t,i}^m.$$

Different from the standard pointing method, we design a dual copy mechanism to copy from two sources with two gates. The first gate is designed for determining copy tokens from two sources of inputs or generate next word from P_V, which is computed as $g_t^v = \text{sigmoid}(\mathbf{w}_g^v \tilde{\mathbf{h}}_t + b_g^v)$. The second gate takes charge of selecting the source (sentence or relation) to copy from, which is computed as $g_t^c = \text{sigmoid}(\mathbf{w}_g^c[\mathbf{c}_t^s; \mathbf{c}_t^m] + b_g^c)$. Finally, we

	Du Split	Zhou Split
# pairs (Train)	74689	86635
# pairs (Dev)	10427	8965
# pairs (Test)	11609	8964
Sentence avg. tokens	32.56	32.72
Question avg. tokens	11.42	11.31

Table 3: Dataset statistics on Du Split (Du et al., 2017) and Zhou Split (Zhou et al., 2017).

combine all probabilities P_V, P_S and P_M through two soft gates g_t^v and g_t^c. The probability of predicting w as the t-th token of the question is:

$$\begin{aligned} \text{P}(w) &= (1 - g_t^v)\text{P}_V(w) \\ &+ g_t^v g_t^c \text{P}_S(w) + g_t^v(1 - g_t^c)\text{P}_M(w). \end{aligned} \tag{7}$$

Training and Inference. Given the answer A, sentence S and relation M, the training objective is to minimize the negative log-likelihood with regard to all parameters:

$$\mathcal{L} = - \sum_{Q \in \{Q\}} log\, \text{P}(Q|A, S, M; \theta), \tag{8}$$

where $\{Q\}$ is the set of all training instances, θ denotes model parameters and $\log P(Q|A, S, M; \theta)$ is the conditional log-likelihood of Q.

In testing, our model targets to generate a question Q by maximizing:

$$\overline{Q} = \arg\max_Q log\, \text{P}(Q|A, S, M; \theta). \tag{9}$$

3 Experimental Setting

3.1 Dataset & Metrics

We conduct experiments on the SQuAD dataset (Rajpurkar et al., 2016). It contains 536 Wikipedia articles and 100k crowd-sourced question-answer pairs. The questions are written by crowd-workers and the answers are spans of tokens in the articles. We employ two different data splits by following Zhou et al. (2017) [2] and Du et al. (2017) [3]. In Zhou et al. (2017), the original SQuAD development set is evenly divided into dev and test sets, while Du et al. (2017) treats SQuAD development set as its development set and splits original SQuAD training set into a training set and a test set. We also filter out questions which do not have any overlapped non-stop words with the corresponding sentences and perform some preprocessing steps, such as tokenization and sentence splitting. The data statistics are given in Table 3.

---

[2]https://res.qyzhou.me/redistribute.zip
[3]https://github.com/xinyadu/nqg/tree/master/data/raw

	Du Split (Du et al., 2017)						Zhou Split (Zhou et al., 2017)					
	B1	B2	B3	B4	MET	R-L	B1	B2	B3	B4	MET	R-L
s2s (Du et al., 2017)	43.09	25.96	17.50	12.28	16.62	39.75	-	-	-	-	-	-
NQG++ (Zhou et al., 2017)	-	-	-	-	-	-	-	-	-	13.29	-	-
M2S+cp (Song et al., 2018)	-	-	-	13.98	18.77	42.72	-	-	-	13.91	-	-
s2s+MP+GSA (Zhao et al., 2018)	43.47	28.23	20.40	15.32	19.29	43.91	**44.51**	29.07	21.06	15.82	19.67	44.24
Hybrid model (Sun et al., 2018)	-	-	-	-	-	-	43.02	28.14	20.51	15.64	-	-
ASs2s (Kim et al., 2019)	-	-	-	16.20	19.92	43.96	-	-	-	16.17	-	-
Our model	**45.66**	**30.21**	**21.82**	**16.27**	**20.36**	**44.35**	44.40	**29.48**	**21.54**	**16.37**	**20.68**	**44.73**

Table 4: The main experimental results for our model and several baselines. '-' means no results reported in their papers. (Bn: BLEU-n, MET: METEOR, R-L: ROUGE-L)

We evaluate with all commonly-used metrics in question generation (Du et al., 2017): BLEU-1 (B1), BLEU-2 (B2), BLEU-3 (B3), BLEU-4 (B4) (Papineni et al., 2002), METEOR (MET) (Denkowski and Lavie, 2014) and ROUGE-L (R-L) (Lin, 2004). We use the evaluation script released by Chen et al. (2015).

3.2 Baseline Models

We compare with the following models.

- s2s (Du et al., 2017) proposes an attention-based sequence-to-sequence neural network for question generation.
- NQG++ (Zhou et al., 2017) takes the answer position feature and linguistic features into consideration and equips the Seq2Seq model with copy mechanism.
- M2S+cp (Song et al., 2018) conducts multi-perspective matching between the answer and the sentence to derive an answer-aware sentence representation for question generation.
- s2s+MP+GSA (Zhao et al., 2018) introduces a gated self-attention into the encoder and a max-out pointer mechanism into the decoder. We report their sentence-level results for a fair comparison.
- Hybrid (Sun et al., 2018) is a hybrid model which considers the answer embedding for the question word generation and the position of context words for modeling the relative distance between the context words and the answer.
- ASs2s (Kim et al., 2019) replaces the answer in the sentence with a special token to avoid its appearance in the generated questions.

3.3 Implementation Details

We take the most frequent 20k words as our vocabulary and use the GloVe word embeddings (Pennington et al., 2014) for initialization. The embed-

ding dimensions for POS, NER, answer position are set to 20. We use two-layer LSTMs in both encoder and decoder, and the LSTMs hidden unit size is set to 600.

We use dropout (Srivastava et al., 2014) with the probability $p = 0.3$. All trainable parameters, except word embeddings, are randomly initialized with the Xavier uniform in $(-0.1, 0.1)$ (Glorot and Bengio, 2010). For optimization in the training, we use SGD as the optimizer with a minibatch size of 64 and an initial learning rate of 1.0. We train the model for 15 epochs and start halving the learning rate after the 8th epoch. We set the gradient norm upper bound to 3 during the training.

We adopt the teacher-forcing for the training. In the testing, we select the model with the lowest perplexity and beam search with size 3 is employed for generating questions. All hyper-parameters and models are selected on the validation dataset.

4 Results and Analysis

4.1 Main Results

Table 4 shows automatic evaluation results for our model and baselines (copied from their papers). Our proposed model which combines structured answer-relevant relations and unstructured sentences achieves significant improvements over proximity-based answer-aware models (Zhou et al., 2017; Sun et al., 2018) on both dataset splits. Presumably, our structured answer-relevant relation is a generalization of the context explored by the proximity-based methods because they can only capture short dependencies around answer fragments while our extractions can capture both short and long dependencies given the answer fragments. Moreover, our proposed framework is a general one to jointly leverage structured

(a) Evaluation results on all sentences.

	Hybrid			Our Model		
	BLEU	MET	R-L	BLEU	MET	R-L
0~10 (72.8% of #)	28.54	21.54	46.26	29.73 (4.17%)	22.03 (2.27%)	46.85 (1.28%)
>10 (27.2% of #)	20.67	16.72	37.63	22.12 (7.01%)	17.46 (4.43%)	38.47 (2.23%)

(b) Evaluation results on sentences with more than 20 words.

	Hybrid			Our Model		
	BLEU	MET	R-L	BLEU	MET	R-L
0~10 (58.3% of #)	28.00	21.03	45.37	29.11 (3.96%)	21.50 (2.21%)	45.97 (1.31%)
>10 (26.6% of #)	20.58	16.66	37.53	22.04 (7.09%)	17.38 (4.30%)	38.37 (2.24%)

Table 5: Performance for the average relative distance between the answer fragment and other non-stop sentence words that also appear in the ground truth question (BLEU is the average over BLEU-1 to BLEU-4). Values in parenthesis are the improvement percentage of Our Model over Hybrid. (a) is based on all sentences while (b) only considers long sentences with more than 20 words.

relations and unstructured sentences. All compared baseline models which only consider unstructured sentences can be further enhanced under our framework.

Recall that existing proximity-based answer-aware models perform poorly when the distance between the answer fragment and other non-stop sentence words that also appear in the ground truth question is large (Table 1). Here we investigate whether our proposed model using the structured answer-relevant relations can alleviate this issue or not, by conducting experiments for our model under the same setting as in Table 1. The broken-down performances by different relative distances are shown in Table 5a. We find that our proposed model outperforms Hybrid (our re-implemented version for this experiment) on all ranges of relative distances, which shows that the structured answer-relevant relations can capture both short and long term answer-relevant dependencies of the answer in sentences. Furthermore, comparing the performance difference between Hybrid and our model, we find the improvements become more significant when the distance increases from "0 ~ 10" to "> 10". One reason is that our model can extract relations with distant dependencies to the answer, which greatly helps our model ignore the extraneous information. Proximity-based answer-aware models may overly emphasize the neighboring words of answers and become less effective as the useful context becomes further away from the answer in the complex sentences. In fact, the breakdown intervals in Table 5a naturally bound its sentence length, say for "> 10", the sentences in this group must be longer than 10. Thus, the length variances in these two intervals could be

Sentence: Beyoncé received critical acclaim and commercial success, selling one million digital copies worldwide in six days; The New York Times noted the album 's unconventional, unexpected release as significant.
Reference Question: How many digital copies of her fifth album did Beyonc sell in six days?
Baseline Prediction: How many digital copies did the New York Times sell in six days ?
Structured Answer-relevant Relation: (Beyoncé; received commercial success selling; one million digital copies worldwide; in six days)
Our Model Prediction: How many digital copies did Beyoncé sell in six days ?

Sentence: The daily mean temperature in January, the area's coldest month, is 32.6 °F (0.3 °C); however, temperatures usually drop to 10 °F (-12 °C) several times per winter and reach 50 °F (10 °C) several days each winter month.
Reference Question: What is New York City 's daily January mean temperature in degrees celsius ?
Baseline Prediction: What is the coldest temperature in Celsius ?
Structured Answer-relevant Relation: (The daily mean temperature in January; is; 32.6 °F (0.3 °C))
Our Model Prediction: In degrees Celsius , what is the average temperature in January ?

Figure 4: Example questions (with answers highlighted) generated by crowd-workers (ground truth questions), the baseline model and our model.

significant. To further validate whether our model can extract long term dependency words. We rerun the analysis of Table 5b only for long sentences (length > 20) of each interval. The improvement percentages over Hybrid are shown in Table 5b, which become more significant when the distance increases from "0 ~ 10" to "> 10".

4.2 Case Study

Figure 4 provides example questions generated by crowd-workers (ground truth questions), the base-

line Hybrid (Sun et al., 2018), and our model. In the first case, there are two subsequences in the input and the answer has no relation with the second subsequence[4]. However, we see that the baseline model prediction copies irrelevant words "The New York Times" while our model can avoid using the extraneous subsequence "The New York Times noted ..." with the help of the structured answer-relevant relation. Compared with the ground truth question, our model cannot capture the cross-sentence information like "her fifth album", where the techniques in paragraph-level QG models (Zhao et al., 2018) may help. In the second case, as discussed in Section 1, this sentence contains a few facts and some facts intertwine. We find that our model can capture distant answer-relevant dependencies such as "mean temperature" while the proximity-based baseline model wrongly takes neighboring words of the answer like "coldest" in the generated question.

4.3 Diverse Question Generation

Another interesting observation is that for the same answer-sentence pair, our model can generate diverse questions by taking different answer-relevant relations as input. Such capability improves the interpretability of our model because the model is given not only what to be asked (i.e., the answer) but also the related fact (i.e., the answer-relevant relation) to be covered in the question. In contrast, proximity-based answer-aware models can only generate one question given the sentence-answer pair regardless of how many answer-relevant relations in the sentence. We think such capability can also validate our motivation: questions should be generated according to the answer-aware relations instead of neighboring words of answer fragments. Figure 5 show two examples of diverse question generation. In the first case, the answer fragment 'Hugh L. Dryden' is the appositive to 'NASA Deputy Administrator' but the subject to the following tokens 'announced the Apollo program ...'. Our framework can extract these two answer-relevant relations, and by feeding them to our model separately, we can receive two questions asking different relations with regard to the answer.

---

[4]One might think that the two subsequences should be regarded as individual sentences, however, several off-the-shelf tools do recognize them as one sentence.

Sentence: In July 1960, NASA Deputy Administrator Hugh L. Dryden announced the Apollo program to industry representatives at a series of Space Task Group conferences.
Relation 1: (Hugh L. Dryden; [is] Deputy Administrator [of]; NASA)
Question 1: Who was the NASA Deputy Administrator in 1960 ?
Relation 2: (NASA Deputy Administrator Hugh L. Dryden; announced; the Apollo program to industry representatives at a series of Space Task Group conferences; In July 1960)
Question 2: Who announced the Apollo program to industry representatives ?
Sentence: One of the network's strike-replacement programs during that time was the game show Duel, which premiered in December 2007.
Relation 1: (the game show Duel; premiered; in December 2007)
Question 1: What game premiered in December 2007 ?
Relation 2: (One of the network's strike-replacement programs during that time; was; the game show Duel)
Question 2: What was the name of an network 's strike - replacement programs ?

Figure 5: Example diverse questions (with answers highlighted) generated by our model with different answer-relevant relations.

5 Related Work

The topic of question generation, initially motivated for educational purposes, is tackled by designing many complex rules for specific question types (Mitkov and Ha, 2003; Rus et al., 2010). Heilman and Smith (2010) improve rule-based question generation by introducing a statistical ranking model. First, they remove extraneous information in the sentence to transform it into a simpler one, which can be transformed easily into a succinct question with predefined sets of general rules. Then they adopt an overgenerate-and-rank approach to select the best candidate considering several features.

With the rise of dominant neural sequence-to-sequence learning models (Sutskever et al., 2014), Du et al. (2017) frame question generation as a sequence-to-sequence learning problem. Compared with rule-based approaches, neural models (Yuan et al., 2017) can generate more fluent and grammatical questions. However, question generation is a one-to-many sequence generation problem, i.e., several aspects can be asked given a sentence, which confuses the model during train and prevents concrete automatic evaluation. To tackle this issue, Zhou et al. (2017) propose the answer-aware question generation setting which assumes the answer is already known

and acts as a contiguous span inside the input sentence. They adopt a BIO tagging scheme to incorporate the answer position information as learned embedding features in Seq2Seq learning. Song et al. (2018) explicitly model the information between answer and sentence with a multi-perspective matching model. Kim et al. (2019) also focus on the answer information and proposed an answer-separated Seq2Seq model by masking the answer with special tokens. All answer-aware neural models treat question generation as a one-to-one mapping problem, but existing models perform poorly for sentences with a complex structure (as shown in Table 1).

Our work is inspired by the process of extraneous information removing in (Heilman and Smith, 2010; Cao et al., 2018). Different from Heilman and Smith (2010) which directly use the simplified sentence for generation and Cao et al. (2018) which only consider aggregate two sources of information via gated attention in summarization, we propose to combine the structured answer-relevant relation and the original sentence. Factoid question generation from structured text is initially investigated by Serban et al. (2016), but our focus here is leveraging structured inputs to help question generation over unstructured sentences. Our proposed model can take advantage of unstructured sentences and structured answer-relevant relations to maintain informativeness and faithfulness of generated questions. The proposed model can also be generalized in other conditional sequence generation tasks which require multiple sources of inputs, e.g., distractor generation for multiple choice questions (Gao et al., 2019b).

6 Conclusions and Future Work

In this paper, we propose a question generation system which combines unstructured sentences and structured answer-relevant relations for generation. The unstructured sentences maintain the informativeness of generated questions while structured answer-relevant relations keep the faithfulness of questions. Extensive experiments demonstrate that our proposed model achieves state-of-the-art performance across several metrics. Furthermore, our model can generate diverse questions with different structured answer-relevant relations. For future work, there are some interesting dimensions to explore, such as difficulty levels (Gao et al., 2019a), paragraph-level information (Zhao et al., 2018) and conversational question generation (Gao et al., 2019c).

References

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *CoRR*, abs/1504.00325.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Yifan Gao, Lidong Bing, Wang Chen, Michael R. Lyu, and Irwin King. 2019a. Difficulty controllable generation of reading comprehension questions. In *Proceedings of the Twenty-Eightth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization.

Yifan Gao, Lidong Bing, Piji Li, Irwin King, and Michael R. Lyu. 2019b. Generating distractors for reading comprehension questions from real examinations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6423–6430.

Yifan Gao, Piji Li, Irwin King, and Michael R. Lyu. 2019c. Interconnected question generation with

coreference alignment and conversation flow modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4853–4862, Florence, Italy. Association for Computational Linguistics.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy. PMLR.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany. Association for Computational Linguistics.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 140–149, Berlin, Germany. Association for Computational Linguistics.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617, Los Angeles, California. Association for Computational Linguistics.

Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *AAAI Conference on Artificial Intelligence*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Ruslan Mitkov and Le An Ha. 2003. Computer-aided generation of multiple-choice tests. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing*, pages 17–22.

Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016. Generating natural questions about an image. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1802–1813, Berlin, Germany. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

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, Austin, Texas. Association for Computational Linguistics.

Vasile Rus, Brendan Wyse, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Christian Moldovan. 2010. The first question generation shared task evaluation challenge. In *Proceedings of the 6th International Natural Language Generation Conference*.

Swarnadeep Saha and Mausam. 2018. Open information extraction from conjunctive sentences. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2288–2299, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Iulian Vlad Serban, Alberto García-Durán, Caglar Gulcehre, Sungjin Ahn, Sarath Chandar, Aaron Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 588–598, Berlin, Germany. Association for Computational Linguistics.

Linfeng Song, Zhiguo Wang, Wael Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 569–574, New Orleans,

Louisiana. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3930–3939, Brussels, Belgium. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine comprehension by text-to-text neural question generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25, Vancouver, Canada. Association for Computational Linguistics.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910, Brussels, Belgium. Association for Computational Linguistics.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In *Proceedings of the 6th CCF International Conference on Natural Language Processing and Chinese Computing (NLPCC)*, pages 662–671, Dalian, China.