How to Ask Good Questions? Try to Leverage Paraphrases

Xin Jia, Wenjie Zhou, Xu Sun, Yunfang Wu*
MOE Key Lab of Computational Linguistics, School of EECS, Peking University
{jemmryx, wjzhou013, xusun, wuyf}@pku.edu.cn

Abstract

Given a sentence and its relevant answer, how to ask good questions is a challenging task, which has many real applications. Inspired by human's paraphrasing capability to ask questions of the same meaning but with diverse expressions, we propose to incorporate paraphrase knowledge into question generation(QG) to generate human-like questions. Specifically, we present a two-hand hybrid model leveraging a self-built paraphrase resource, which is automatically conducted by a simple back-translation method. On the one hand, we conduct multi-task learning with sentence-level paraphrase generation (PG) as an auxiliary task to supplement paraphrase knowledge to the task-share encoder. On the other hand, we adopt a new loss function for diversity training to introduce more question patterns to QG. Extensive experimental results show that our proposed model obtains obvious performance gain over several strong baselines, and further human evaluation validates that our model can ask questions of high quality by leveraging paraphrase knowledge.

1 Introduction

Question generation (QG) is an essential task for NLP, which focuses on generating grammatical questions for given paragraphs or sentences. It plays a vital role in various realistic scenarios. For educational purposes, QG can create reading comprehension materials for language learners (Heilman and Smith, 2010). For business use, QG can bring benefits to conversation systems and chat-bots for effective communication with humans (Mostafazadeh et al., 2016). Besides, automatically-generated questions can be conversely used for constructing question answering datasets to enhance reading comprehension systems (Tang et al., 2017; Duan et al., 2017; Xu et al., 2019; Zhang and Bansal, 2019).

Sentence: the next three drives of the game would end in punts. **Answer:** punts **Reference question:** what did the next three drives result in? **Question generated by the baseline model:** the next three drives of the game would end in what?
Sentence: in ring theory, the notion of number is generally replaced with that of ideal. **Answer:** ring theory **Reference question:** in what theory is the idea of a number exchanged with that of an ideal? **Question generated by the baseline model:** in what theory is the notion of number replaced with that of ideal?

Table 1: Real examples of generated questions from SQuAD. We highlight the paraphrase transitions between sentences and questions. Human creates good questions by leveraging paraphrase knowledge, while the automatically generated questions just copy the original sentence, resulting in lower evaluation scores.

Recent neural network-based methods have achieved promising results on QG, most of which are based on the seq2seq attention framework (Du et al., 2017; Zhou et al., 2017; Gao et al., 2018; Kim et al., 2018; Zhou et al., 2019b), enriched with lexical features (Zhou et al., 2017; Sun et al., 2018; Song et al., 2018) or enhanced by copy mechanism (Du and Cardie, 2018; Sun et al., 2018; Zhou et al., 2019a).

Although much progress has been made for QG, existing approaches do not explicitly model the "notorious" lexical and syntactic gaps in the generation process. That is, some parts of two texts (e.g. the input sentence and reference question, the reference question and generated question) may convey the same meaning but use different words, phrases or syntactic patterns. In real communica-

* Corresponding author.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6130–6140

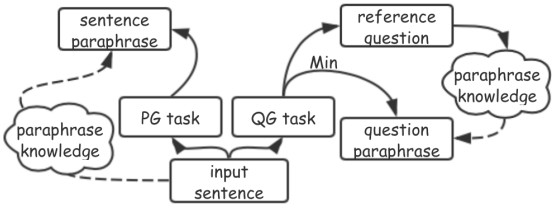

Figure 1: A sketch of our design to leverage paraphrase knowledge in QG.

tion, humans often paraphrase a source sentence to ask questions which are grammatical and coherent. Take SQuAD (Rajpurkar et al., 2016) as an example, which is a popular reading comprehension dataset and has been widely used for QG, there is a large percentage of questions created by paraphrasing (33.3% of the questions contain synonymy variations and 64% of questions contain syntactic variations (Rajpurkar et al., 2016)). Two examples are shown in Table 1. Due to the lack of paraphrase knowledge, the generated questions simply copy certain words from the input sequence, the quality of which is thus not competitive with human-created questions.

To address this issue, we introduce paraphrase knowledge in the QG process to generate human-like questions. The sketch of our design is illustrated in Figure 1. To make our model easy to implement and train the model in an end-to-end fashion, we do not use any extra paraphrase generation (PG) dataset but just use a simple back-translation method to automatically create paraphrases for both the input sentences and reference questions. Based on the high-quality expanded data, we propose a two-hand hybrid model. On the left hand, using the expanded sentence paraphrase as the target of PG, we perform multi-task learning with PG and QG, to optimize the task-share encoder with the paraphrase knowledge. On the right hand, with the gold reference question and question paraphrase as QG's multi-targets, we adopt a new min-loss function, to enable the QG module to learn more diverse question patterns.

We conduct extensive experiments on SQuAD and MARCO (Nguyen et al., 2016). Results show that both separate modules, the PG auxiliary task and the min-loss function, obviously improve the performances of QG task, and combing them achieves further improvements. Furthermore, human evaluation results show that our hybrid model can ask better and more human-like questions by incorporating paraphrase knowledge.

2 Related Work

For current mainstream neural network-based methods on QG, most approaches utilize the Seq2Seq model with attention mechanism (Du et al., 2017; Zhou et al., 2017; Zhao et al., 2018b; Zhou et al., 2019a). To obtain better representations of the input sequence and answer, the answer position and token lexical features are treated as supplements for the neural encoder (Zhou et al., 2017; Song et al., 2018; Kim et al., 2018). Similar to other text generation tasks, many works on QG also employ copy or pointer mechanism to overcome the OOV problem (Du and Cardie, 2018; Sun et al., 2018; Zhang and Bansal, 2019). Recently, Zhou et al. (2019a) employ language modeling (LM) as an auxiliary task to enrich the encoder representations. In this paper, we adopt this work as one of the baseline models, since their universal model is easy to implement and achieves promising results for QG.

In order to make use of the context information of paragraphs, Zhao et al. (2018b) propose a gated self-attention network to encode context passage. Based on this, Zhang and Bansal (2019) apply reinforcement learning to deal with semantic drift in QG; Nema et al. (2019) use a passage-answer fusion mechanism to obtain answer-focused context representations; Li et al. (2019a) utilize gated attention to fuse answer-relevant relation with context sentence. Besides, Chen et al. (2019) design different passage graphs to capture structure information of passage through graph neural networks. Dong et al. (2019) propose a unified language model pre-training method to obtain better context representations for QG. All these works adopt a whole paragraph as input to generate questions. Different from this, our work only takes a sentence as input and leaves paragraph-level QG for future research.

Paraphrase generation is also a challenging task for NLP. Recent works usually obtain paraphrases by reordering or modifying the syntax or lexicon based on some paraphrase databases and rules (Fader et al., 2013; Chen et al., 2016), or by employing some neural generation methods (Prakash et al., 2016; Li et al., 2019b). In this paper, we employ a simple and effective paraphrasing method to expand both input sentences and reference questions. Our method also can be replaced with more sophisticated paraphrasing methods.

Paraphrase knowledge has been used to improve many NLP tasks, such as machine translation, ques-

tion answering, and text simplification. Callison-Burch et al. (2006) use paraphrase techniques to deal with unknown phrases to improve statistical machine translation. Fader et al. (2013) and Dong et al. (2017) employ paraphrase knowledge to enhance question answering models. Kriz et al. (2018) utilize paraphrase and context-based lexical substitution knowledge to improve simplification task. Similarly, Zhao et al. (2018a) combine paraphrase rules of PPDB (Ganitkevitch et al., 2013) with Transformer (Vaswani et al., 2017) to perform sentence simplification task. Guo et al. (2018a) propose a multi-task learning framework with PG and simplification. In addition, Yu et al. (2018) and Xie et al. (2019) use paraphrase as data argumentation for their primary tasks. Different from these works, we leverage paraphrase knowledge for question generation, by automatically constructing a built-in paraphrase corpus without using any external paraphrase knowledge bases.

3 Model Description

In this section, we first describe two baseline models we used: feature-enriched pointer-generator and language modeling enhanced QG. Then we explain how to obtain paraphrase resources and show the quality statistics. Furthermore, we describe in detail two modules of utilizing paraphrase knowledge: the PG auxiliary task and the min loss function, as well as their combination. The overall structure of our hybrid model is shown in Figure 2.

3.1 Baseline Models

3.1.1 Feature-enriched Pointer-generator

Sun et al. (2018) enhance pointer-generator (See et al., 2017) model with rich features proposed by Zhou et al. (2017). They adopt a bidirectional LSTM as the encoder, which takes the feature-enriched embedding e_i as input:

$$e_i = [w_i; a_i; n_i; p_i; u_i] \tag{1}$$

where w_i, a_i, n_i, p_i, u_i respectively represents embeddings of word, answer position, name entity, POS and word case.

Same as the decoder used by See et al. (2017), another unidirectional LSTM with attention mechanism is used to obtain the decoder hidden state s_t and context vector c_t. Based on these, the pointer-generator model will simultaneously calculate the probabilities of generating a word from vocabulary and copying a word from the source text. The final probability distribution is the combination of these two modes with a generation probability p_g:

$$P(w) = p_g P_{vocab} + (1 - p_g) P_{copy} \tag{2}$$

The training objective is to minimize the negative log likelihood of the target sequence $\mathbf{q}$:

$$\mathcal{L}_{qg} = -\frac{1}{T_{qg}} \sum_{t=1}^{T_{qg}} log P(y_t^{qg} = \mathbf{q}_t) \tag{3}$$

3.1.2 Language Modeling Enhanced QG

Zhou et al. (2019a) enhance QG with language modeling under a hierarchical structure of multi-task learning. The language modeling aims at predicting the next and previous words in the input sequence with forward and backward LSTMs, respectively, which serves as a low-level task to provide semantic information for the high-level QG task.

In general, the input sequence will firstly be fed into the language modeling module to get the semantic hidden states, then these states will be concatenated with the input sequence to obtain the input of the feature-rich encoder:

$$e_i = [w_i; a_i; n_i; p_i; u_i; h_i^{lm}] \tag{4}$$

where h_i^{lm} is the semantic hidden state of LM module. The loss function of language modeling is defined as:

$$\begin{aligned} \mathcal{L}_{lm} = &-\frac{1}{T_{lm}-1} \sum_{t=1}^{T_{lm}-1} log(P^{lm}(w_{t+1}|w_{<t+1})) \\ &-\frac{1}{T_{lm}-1} \sum_{t=2}^{T_{lm}} log(P^{lm}(w_{t-1}|w_{>t-1})) \end{aligned} \tag{5}$$

where $P^{lm}(w_{t+1}|w_{<t+1})$ and $P^{lm}(w_{t-1}|w_{>t-1})$ represent the generation probabilities of the next word and the previous word, respectively.

As a result, the total loss of language modeling enhanced QG is formulated as:

$$\mathcal{L}_{lqg} = \mathcal{L}_{qg} + \beta \mathcal{L}_{lm} \tag{6}$$

where β is a hyper-parameter to control the relative importance between language modeling and QG. Follow the work of Zhou et al. (2019a), we set β to 0.6. We re-implement this unified model to base our method on a strong baseline.

3.2 Paraphrase Expansion

The paraphrasing strategy is independent of the neural-based QG model, and we can use any advanced methods to generate paraphrases. In our work, we employ a simple back-translation method to automatically create paraphrases of both sentences and questions. Specially, we use a mature translation tool **Google Translate**, which is a free and accessible online service. We translate an original text into German and then back to English to get its paraphrase. As a result, we obtain $\mathbf{s}'$ which is the paraphrase of the input sentence $\mathbf{s}$, and $\mathbf{q}'$ which is the paraphrase of the golden reference question $\mathbf{q}$. In the following section, we will illustrate the way to use $(\mathbf{s}, \mathbf{s}')$ as a training pair of the auxiliary PG task, and adopt $(\mathbf{q}, \mathbf{q}')$ as multi-references to conduct the diversity training module. The way we expand paraphrases does not need extra PG datasets. Besides, it guarantees the PG and QG tasks share the same input $\mathbf{s}$, so we can optimize their sharing encoder simultaneously and train the model end-to-end.

	Synonym	Syntactic	Fluency
sentence-paraphrase	74%	7%	67%
question-paraphrase	58%	44%	67%

Table 2: Human evaluation of expanded paraphrases.

To assess the quality of expanded paraphrases, we randomly select 100 paraphrases respectively from sentences and questions, and ask two annotators to judge the *Synonym* conversions and *Syntactic* transitions, as well as the paraphrase *Fluency*. As shown in Table 2, 74% sentence paraphrases and 58% question paraphrases have synonym conversions with source sequences, 7% and 44% of them have sentence pattern transitions. Besides, 67% of paraphrases have no grammar errors. Two real expansion examples are shown in Table 3. It indicates that our expansion method introduces rich and high quality paraphrasing knowledge into the original data.

3.3 Multi-task Learning with Paraphrase Generation

3.3.1 Auxiliary PG Task

The multi-task learning mechanism with PG aims at introducing paraphrase knowledge into QG. In general, we employ a parallel architecture to combine PG and QG, where QG is the main task and PG serves as an auxiliary task. To make our model easy to implement and can be trained end-to-end, we conduct the multi-task learning in a simultaneous mode. In detail, feature-riched embeddings will first be encoded by the task-share encoder and then be fed into PG and QG decoders respectively. The PG and QG decoders both have two layers and they are identical in the structure but different in parameters.

Input Sentence:
the current basilica of the sacred heart is located on the spot of fr.
Sentence Paraphrase:
the present basilica of the sacred heart is located in the place of fr.
Input Question:
what structure is found on the location of the original church of father sorin at notre dame?
Question Paraphrase:
what structure can be found at the location of the original church of father sorin at notre dame?

Table 3: Real examples of our paraphrase expansion on the sentences and reference questions respectively. We mark paraphrase transitions with color.

In the auxiliary PG task, the input is the original sentence $\mathbf{s}$, and the training objective is to minimize the cross-entropy loss:

$$\mathcal{L}_{pg} = -\frac{1}{T_{pg}} \sum_{t=1}^{T_{pg}} logP(y_t^{pg} = \mathbf{s}'_t) \quad (7)$$

where y_t^{pg} is the generated word of PG at time step t and $\mathbf{s}'_t$ is the t th word in the expanded sentence paraphrase $\mathbf{s}'$.

3.3.2 Soft Sharing Strategy

To enhance the impact of auxiliary PG task so that the paraphrase knowledge can be absorbed by the question generation process more deeply, we employ a soft sharing strategy between the first layer of PG and QG decoders. The soft sharing strategy loosely couples parameters and encourages them close to each other in representation space. Following the work of Guo et al. (2018b), we minimize the l_2 distance between the shared layer of QG and PG decoders as a regularization. The soft sharing loss is defined as:

$$\mathcal{L}_{sf} = \sum_{\mathbf{d} \in \mathcal{D}} ||\theta_d - \phi_d||_2 \quad (8)$$

where $\mathcal{D}$ is the set of shared decoder parameters, θ and ϕ respectively represent the parameters of the main QG task and the auxiliary PG task.

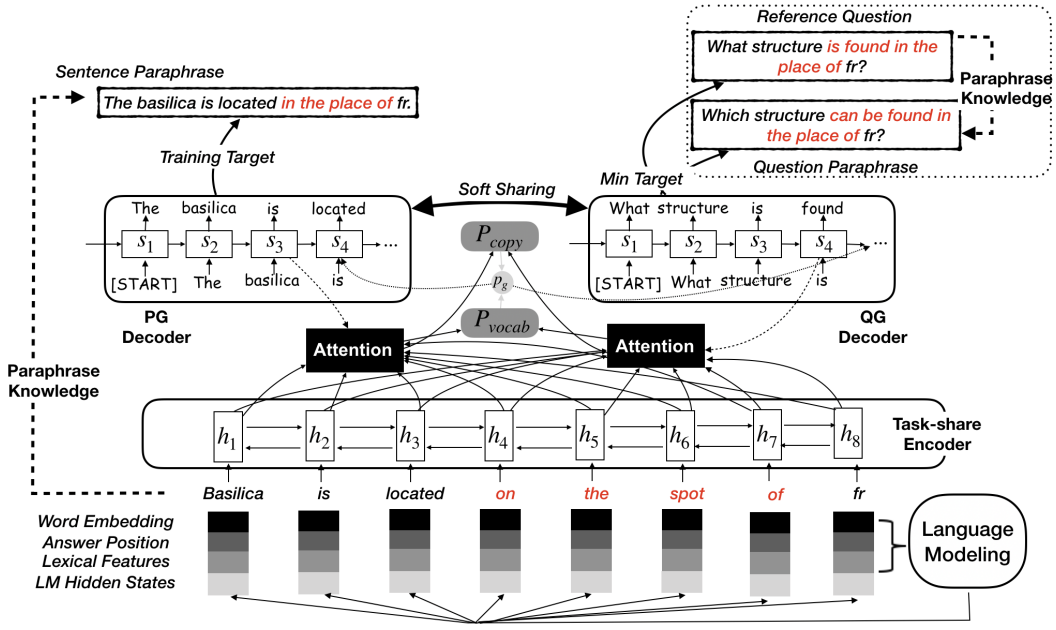

Figure 2: Illustration of our proposed hybrid model.

3.4 Diversity Training with Min-loss Function

For the QG task, a general training goal is to fit the decoded results with the reference questions. To provide more generation patterns, we adjust the training target from one golden reference question to several reference questions by using expanded paraphrase resources. We adopt a min-loss function among several references, and the loss function defined by Equation 3 can be rewritten as:

$$\mathcal{L}_{qg} = \min_{\mathbf{q}\in\mathcal{Q}}(-\frac{1}{T_{qg}}\sum_{t=1}^{T_{qg}} logP(y_t^{qg} = \mathbf{q}_t)) \quad (9)$$

where $\mathcal{Q}$ is the set of gold reference question and expanded question paraphrase $\{q, q'\}$. Each generated question will separately calculate the negative log-likelihood of its multiple references, and the final loss is the minimum of them. Under this training process, our model can learn multiple question expressions which are not in the original training dataset, so that the generation can be more diverse.

Besides, inspired by the work of Kovaleva et al. (2018), we have tried several loss strategies, such as minimum loss, maximum loss, and weighted loss to guide the diversity training. Among them, the minimum is the best performing strategy. By employing minimum strategy, the QG decoder fits the generated question with the most similar sequence among gold reference question and question paraphrase. In this way, more question patterns are introduced into QG process.

3.5 Hybrid Model

Combining the above modules, we get our hybrid model. During training, the feature-enriched inputs are first encoded by the task-share encoder. Then the semantic hidden states are fed into PG decoder and QG decoder, respectively. For PG decoder, it has one fitting target (expanded sentence paraphrase). For QG decoder, it calculates the cross-entropy loss with both the gold reference question and the question paraphrase and regards the minimum loss of them as the QG loss. The auxiliary PG task and diversity training strategy simultaneously optimize the question generation process. The combined training loss function can be defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{lqg} + \alpha\mathcal{L}_{pg} + \lambda\mathcal{L}_{sf} \quad (10)$$

where α and λ are both hyper-parameters. We will describe the chosen of these hyper-parameters later.

4 Experimental Settings

4.1 Datasets

Our experiments are based on two reading comprehension datasets: SQuAD (2016) and MARCO (2016). On SQuAD, since there are two different splits that are most often used, we conduct experiments on both two splits on sentence-level. For

	Zhou Split					Du Split				
Previous Works (conference-year)	**B1**	**B2**	**B3**	**B4**	**MET**	**B1**	**B2**	**B3**	**B4**	**MET**
s2s (ACL-2017)	-	-	-	-	-	43.09	25.96	17.50	12.28	16.62
NQG++ (NLPCC-2017)	-	-	-	13.29	-	-	-	-	-	-
M2S+cp (NAACL-2018)	-	-	-	13.91	-	-	-	-	13.98	18.77
A-P-Hybrid (EMNLP-2018)	43.02	28.14	20.51	15.64	-	-	-	-	-	-
s2sa-at-mp-gsa (EMNLP-2018)	**44.51**	29.07	21.06	15.82	19.67	43.47	28.23	20.40	15.32	19.29
ASs2s (AAAI-2019)	-	-	-	16.17	-	-	-	-	16.20	19.92
LM enhanced QG (EMNLP-2019)	42.80	28.43	21.08	16.23	-	-	-	-	-	-
Q-type (EMNLP-2019)	43.11	29.13	21.29	16.31	-	-	-	-	-	-
Sent-Relation (EMNLP-2019)	44.40	**29.48**	21.54	16.37	**20.68**	**45.66**	**30.21**	21.82	16.27	20.36
Our Models										
baseline-1 +Data augmentation	38.16	24.35	17.60	13.28	17.73	38.91	24.80	17.83	13.36	17.97
baseline-1	41.06	26.63	19.65	14.71	19.12	41.04	27.05	19.92	15.21	19.19
baseline-1 +Min	42.03	27.61	20.27	15.48	19.61	42.97	28.52	21.02	16.06	19.93
baseline-1 + PG	42.76	28.26	20.89	16.09	20.11	43.68	28.99	21.39	16.37	20.23
baseline-1 +Min+PG (hybrid model-1)	43.61	28.67	21.09	16.23	20.29	42.66	28.68	21.39	16.55	20.44
baseline-2	42.39	28.11	20.86	16.13	19.95	42.76	28.80	21.47	16.57	20.38
baseline-2 +Min	43.38	28.92	21.49	16.61	20.40	42.94	29.06	21.73	16.88	20.60
baseline-2 +PG	43.56	28.98	21.57	16.74	20.58	43.73	29.53	22.06	17.08	20.78
baseline-2 +Min+PG (hybrid model-2)	43.63	29.21	**21.79**	**16.93**	20.58	44.32	29.88	**22.28**	**17.21**	**20.96**

Table 4: Experimental results of our models on SQuAD comparing with previous works and different baselines. The results of previous works are copied from their original papers. Baseline-1 and Baseline-2 refer to Feature-enriched Pointer-generator and LM enhanced QG respectively. Bn: BLEU-n, MET: METOER.

Du Split (Du et al., 2017), we use the same settings with Li et al. (2019a) and there are 74689, 10427 and 11609 sentence-question-answer triples for training, validation and test respectively. For Zhou Split (Zhou et al., 2017), we use the data shared by Zhou et al. (2017) and there are 86,635, 8,965 and 8,964 triples correspondingly. On MARCO, there are 74,097, 4,539 and 4,539 sentence-answer-question triples for train, development and test sets, respectively (Sun et al., 2018).

We expand the datasets using the paraphrase expansion approach described in Section 3.2. After that, one sample of the expanded dataset is in the form of ((sentence, sentence paraphrase), (question, question paraphrase), answer).

4.2 Baselines and Metrics

For fair comparison, we report the following recent works on sentence-level Du and Zhou Splits:

s2s (Du et al., 2017): an attention-based seq2seq model.

NQG++ (Zhou et al., 2017): a feature-enriched Seq2Seq model.

M2S+cp (Song et al., 2018): uses different matching strategies to explicitly model the information between answer and context.

A-P-Hybrid (Sun et al., 2018): generates an accurate interrogative word and focuses on important context words.

s2s-a-ct-mp-gsa (Zhao et al., 2018b): employs a gated attention encoder and a maxout pointer decoder to deal with long text inputs.

ASs2s (Kim et al., 2018): proposes an answer-separated Seq2Seq model by replacing the answer in the input sequence with some specific words.

LM enhanced QG (Zhou et al., 2019a): treats language modeling as a low-level task to provide semantic representations for the high-level QG.

Q-type (Zhou et al., 2019b): multi-task learning framework with question word prediction and QG.

Sent-Relation (Li et al., 2019a): extracts answer-relevant relations in sentence and encodes both sentence and relations to capture answer-focused representations.

We evaluate the performance of our models using **BLEU** (Papineni et al., 2002) and **METEOR** (Denkowski and Lavie, 2014), which are widely used in previous works for QG.

4.3 Implementation Details

We set the vocabulary as the most frequent 20,000 words. We use 300-dimensional GloVe word vectors as initialization of the word embeddings. Answer position and token lexical features are randomly initialized to 32-dimensional vectors through truncated normal distribution. The maximum lengths of input sequence and output sequence are 100 and 40, respectively. The hidden

size of the encoder, decoder, and language modeling LSTMs are all 512. We use Adagrad optimization with learning rate 0.15 for training. The batch size is 32 and the beam search decoding size is 12. To alleviate the volatility of the training procedure, we get the average model of the 5 checkpoints closest to the best-trained model on development set.

5 Results and Analysis

5.1 Main Results

The experimental results on two splits of SQuAD are shown in Table 4. In terms of BLEU-4 that is often regarded as the main evaluation metric for text generation, our hybrid model-2 yields the best results on both splits, with 16.93 on Zhou Split and 17.21 on Du Split. We achieve state-of-the-art results on Du Split for sentence-level QG.

Especially for baseline-1, the performance gains of our model are more obvious. Our hybrid model-1 outperforms baseline-1 by 1.52 points on Zhou Split and 1.34 points on Du Split, which are large margins for this challenging task. Even based on this weak baseline, our method also achieves the state-of-the-art, 16.55 BLEU-4 score on Du Split for sentence-level QG.

The previous work of CGC-QG (Liu et al., 2019) obtains a 17.55 BLEU-4 score on Zhou Split. But their model relies on many heuristic rules and ad-hoc strategies. In their full model with clue prediction, they do graph convolutional network (GCN) operations on dependency trees, while our model does not use any hand-crafted rules and is lightweight without graphs and trees.

We also conduct experiments on MARCO, and the results are shown in Table 5. Our hybrid models obtain obvious improvements over two baselines, achieving a state-of-the-art BLEU-4 score of 21.61.

Specifically, SQuAD and MARCO are built in different ways. The questions in SQuAD are generated by crowd-workers, while questions in MARCO are sampled from real user queries. The experimental results on two datasets validate the generalization and robustness of our models.

Effect of Multi-task Learning with PG Task As shown in Table 4, the auxiliary PG task brings consistent improvements over both baseline models. On Zhou Split, it increases baseline-1 by 1.38 points and baseline-2 by 0.61 respectively. On Du Split, it increases baseline-1 by 1.16 points and baseline-2 by 0.51 points respectively. The reason is that the PG task provides abundant paraphrase knowledge into the model and allows the task-share encoder to learn more paraphrasing representations.

Previous Works	**BLEU-4**
s2s(Du et al., 2017)	10.46
s2sa-at-mp-gsa(Zhao et al., 2018b)	16.02
A-P-Hybrid(Sun et al., 2018)	19.45
LM enhanced QG(Zhou et al., 2019a)	20.88
Q-type(Zhou et al., 2019b)	21.59
Our Models	
baseline-1	20.13
hybrid model-1	21.15
baseline-2	20.79
hybrid model-2	**21.61**

Table 5: Main results of our models on MARCO.

Effect of Diversity Training with Min-loss Function From the results in Table 4, we can see the min-loss strategy improves performances over both baseline models. On Zhou Split, we get a 0.77 improvement over baseline-1 and 0.48 improvement over baseline-2, respectively. On Du Split, we get similar improvements.

Effect of Data Augmentation A straightforward way to leverage paraphrase knowledge is data augmentation. To test whether it works by simply adding paraphrase data as external training data, we also conduct an experiment based on the question paraphrase resource. We add the $(\mathbf{s}, \mathbf{q}')$ pairs into the training dataset, where $\mathbf{s}$ represents the input sentence and $\mathbf{q}'$ denotes the paraphrase of the golden reference. Under this setting, we double the training samples. Unfortunately, as shown in Table 4, the baseline-1 model yields much lower BLEU-4 scores on both Zhou Split (13.28) and Du Split (13.36) with such data augmentation. The main reason is that for the same input sentence, there are two different training targets ($\mathbf{q}$ and $\mathbf{q}'$), making the training process cannot easily converge.

5.2 Diversity Test

To investigate whether the paraphrase knowledge introduces more diverse expressions, we conduct evaluations on the **distinct** metric (Li et al., 2016), which is calculated as the number of distinct unigrams (distinct-1) and bigrams (distinct-2) divided by the total number of the generated words. The experimental results are shown in Table 6. It shows that our hybrid models obtain obvious gains over baseline models on both distinct-1 and distinct-2

metrics, validating that our models really generate more diverse questions with the help of paraphrase knowledge.

Models	distinct-1	distinct-2
baseline-1	9.49	39.48
hybrid model-1	9.75	41.97
baseline-2	9.81	41.14
hybrid model-2	9.98	42.43

Table 6: Results of the distinct metric on zhou split.

5.3 Ablation Study of Soft Sharing

We also verify the effectiveness of the soft sharing mechanism by removing it from the full hybrid models. The results are displayed in Table 7. After removing the soft sharing mechanism, both of our models have varying degrees of performance degradation. It demonstrates that the soft sharing strategy enhances the influence of paraphrase knowledge on QG decoder.

Models	BLEU-4	METEOR
hybrid model-1	16.23	20.29
w/o soft sharing	15.87	20.04
hybrid model-2	16.93	20.58
w/o soft sharing	16.32	20.34

Table 7: Ablation studies of soft sharing on Zhou Split.

5.4 Parameters Selection

The soft sharing coefficient hyper-parameter λ is 1×10^{-6}, intuitively chosen by balancing the cross-entropy and regularization losses according to Guo et al. (2018b). The other hyper-parameter α which is to control the balance of QG and PG is tuned by grid search. We set α to different values to explore the best proportion of two tasks. The experimental results of different α are shown in Figure 3. Consequently, we set α to 0.3 for our hybrid model.

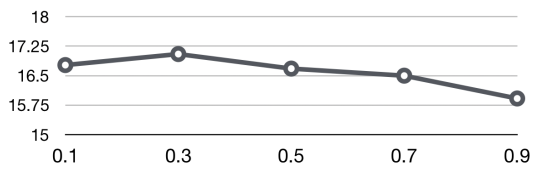

Figure 3: The influence of α on BLEU-4 scores on development set of Zhou Split.

5.5 Human Evaluation

To further assess the quality of generated questions, we perform human evaluation to compare our hybrid model-2 with the strong baseline of language modeling enhanced QG. We randomly select 100 samples from SQuAD (Zhou Split) and ask three annotators to score these generated questions according to three aspects:

Fluency: which measures whether a question is grammatical and fluent;

Relevancy: which measures whether the question is relevant to the input context;

Answerability: which indicates whether the question can be answered by the given answer.

The rating score is set to [0, 2]. The evaluation results are shown in Table 8. The Spearman correlation coefficients between annotators are high, which guarantees the validity of human evaluation. Our hybrid model receives higher scores on all three metrics, indicating that our generated questions have higher quality in different aspects.

Models	Fluency	Relevancy	Answerability
baseline-2	1.785	1.535	1.134
hybrid model-2	**1.874**	**1.682**	**1.333**
Spearman	0.722	0.693	0.861

Table 8: Human evaluation results.

5.6 Case Study

We list two examples of generated questions in Table 9. By introducing paraphrase knowledge into generation, the generated questions well capture the paraphrase transitions between contexts and references. Obviously, the questions generated by our hybrid model are more grammatical and coherent.

5.7 Different Paraphrasing Methods

To further test the generalization of our proposed methods, we use other paraphrasing methods to construct the paraphrase dataset.

PPDB: for each non-stop word and phrase, looking it up in PPDB (2013) and replacing it with its synonyms.

NMT: another back-translation method using a pre-trained Transformer (2017) model.

Mixed: expanding input sentences with Google Trans and expanding reference questions with PPDB.

The results are shown in Table 10. Our hybrid model-2 still achieves excellent performances on both BLEU and METEOR. From the results, we

Sentence:
his lab **was** torn down in 1904, and its contents were sold two years later to satisfy a debt.
Answer:
torn down
Reference Question:
what **happened** to his lab?
Baseline Model-2:
what **was** [UNK] 's lab?
Hybrid Model-2:
what **happened** to his lab in 1904?
Sentence:
newcastle has a horse racing course **at** gosforth park.
Answer:
gosforth park
Reference Question:
where is newcastle 's horse racing course **located**?
Baseline Model-2:
where does newcastle have a horse racing course?
Hybrid Model-2:
where is newcastle 's horse racing course **located**?

Table 9: Examples of generated questions.

Paraphrasing Methods	BLEU-4	METEOR
baseline-2	16.13	19.95
PPDB	16.65	20.57
NMT	16.76	20.44
Google Trans	16.93	20.58
Mixed	**17.05**	**20.75**

Table 10: Hybrid model-2 performances using different paraphrase expansion methods on SQuAD(Zhou Split).

can observe that the Mixed paraphrase method even obtain better results than the mature Google Translate. It proves that our proposed architecture is effective across different paraphrasing methods and has potential for improvement.

6 Conclusion and Future Work

In this paper, we propose a two-hand hybrid model leveraging paraphrase knowledge for QG. The experimental results of independent modules and hybrid models prove that our models are effective and transferable. Besides, human evaluation results demonstrate that the paraphrase knowledge benefits our model to ask more human-like questions of high quality. In the future, we will explore more diverse and advanced paraphrase expanding methods for both sentence and paragraph level QG. Moreover, we will apply our methods to other similar tasks, such as sentence simplification.

Acknowledgments

We thank Weikang Li and Minghua Zhang for their valuable comments and suggestions. This work is supported by the National Natural Science Foundation of China (61773026) and the Key Project of Natural Science Foundation of China (61936012).

References

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 17–24, New York City, USA. Association for Computational Linguistics.

Bo Chen, Le Sun, Xianpei Han, and Bo An. 2016. Sentence rewriting for semantic parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 766–777, Berlin, Germany. Association for Computational Linguistics.

Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2019. Natural question generation with reinforcement learning based graph-to-sequence model. *ArXiv*, abs/1910.08832.

Michael J. Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *WMT@ACL*.

Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886, Copenhagen, Denmark. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *NeurIPS*.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from Wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1907–1917, Melbourne, Australia. 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.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874, Copenhagen, Denmark. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1608–1618, Sofia, Bulgaria. Association for Computational Linguistics.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Yifan Gao, Jianan Wang, Lidong Bing, Irwin King, and Michael R. Lyu. 2018. Difficulty controllable question generation for reading comprehension. *ArXiv*, abs/1807.03586.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018a. Dynamic multi-level multi-task learning for sentence simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 462–476, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018b. Soft layer-specific multi-task summarization with entailment and question generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 687–697, Melbourne, Australia. 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. 2018. Improving neural question generation using answer separation. In *AAAI*.

Olga Kovaleva, Anna Rumshisky, and Alexey Romanov. 2018. Similarity-based reconstruction loss for meaning representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4875–4880, Brussels, Belgium. Association for Computational Linguistics.

Reno Kriz, Eleni Miltsakaki, Marianna Apidianaki, and Chris Callison-Burch. 2018. Simplification using paraphrases and context-based lexical substitution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 207–217, New Orleans, Louisiana. Association for Computational Linguistics.

Jingjing Li, Yifan Gao, Lidong Bing, Irwin King, and Michael R. Lyu. 2019a. Improving question generation with to the point context. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3216–3226, Hong Kong, China. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Qun Liu. 2019b. Decomposable neural paraphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3403–3414, Florence, Italy. Association for Computational Linguistics.

Bang Liu, Mingjun Zhao, Di Niu, Kunfeng Lai, Yancheng He, Haojie Wei, and Yu Xu. 2019. Learning to generate questions by learning what not to generate. *ArXiv*, abs/1902.10418.

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.

Preksha Nema, Akash Kumar Mohankumar, Mitesh M. Khapra, Balaji Vasan Srinivasan, and Balaraman Ravindran. 2019. Let's ask again: Refine network for automatic question generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3314–3323, Hong Kong, China. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *ArXiv*, abs/1611.09268.

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

Pennsylvania, USA. Association for Computational Linguistics.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual LSTM networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2923–2934, Osaka, Japan. The COLING 2016 Organizing Committee.

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.

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.

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.

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.

Duyu Tang, Nan Duan, Tao Qin, and Ming Zhou. 2017. Question answering and question generation as dual tasks. *ArXiv*, abs/1706.02027.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Minh-Thang Luong, and Quoc V. Le. 2019. Unsupervised data augmentation. *ArXiv*, abs/1904.12848.

Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun. 2019. Asking clarification questions in knowledge-based question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1618–1629, Hong Kong, China. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *ArXiv*, abs/1804.09541.

Shiyue Zhang and Mohit Bansal. 2019. Addressing semantic drift in question generation for semi-supervised question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2495–2509, Hong Kong, China. Association for Computational Linguistics.

Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. 2018a. Integrating transformer and paraphrase rules for sentence simplification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3164–3173, Brussels, Belgium. Association for Computational Linguistics.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018b. 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 *NLPCC*.

Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019a. Multi-task learning with language modeling for question generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3394–3399, Hong Kong, China. Association for Computational Linguistics.

Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019b. Question-type driven question generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6032–6037, Hong Kong, China. Association for Computational Linguistics.