SkillQG: Learning to Generate Question for Reading Comprehension Assessment

Xiaoqiang Wang[1], **Bang Liu**[2†‡], **Siliang Tang**[1‡] and **Lingfei Wu**[3‡]
[1]Zhejiang University, [2]Université de Montréal & Mila, [3]Pinterest
{xq.wang, siliang}@zju.edu.cn
bang.liu@umontreal.ca, teddy.lfwu@gmail.com

Abstract

We present **SkillQG**: a question generation framework with controllable comprehension types for assessing and improving machine reading comprehension models. Existing question generation systems widely differentiate questions by *literal* information such as question words and answer types to generate semantically relevant questions for a given context. However, they rarely consider the *comprehension* nature of questions, *i.e.* the different comprehension capabilities embodied by different questions. In comparison, our SkillQG is able to tailor a fine-grained assessment and improvement to the capabilities of question answering models built on it. Specifically, we first frame the comprehension type of questions based on a hierarchical skill-based schema, then formulate SkillQG as a skill-conditioned question generator. Furthermore, to improve the controllability of generation, we augment the input text with question focus and skill-specific knowledge, which are constructed by iteratively prompting the pre-trained language models. Empirical results demonstrate that SkillQG outperforms baselines in terms of quality, relevance, and skill-controllability while showing a promising performance boost in downstream question answering task.

Context: *The princess climbed out the window of the high tower and climbed down the south wall when her mother was sleeping. She wandered out a good way. Finally, she went into the forest where there are no electric poles.*
$\boldsymbol{Q_1}$: *Who climbed out of the castle?* **A**: *Princess.*
$\boldsymbol{Q_2}$: *Why did the princess climb out when her mother was sleeping?* **A**: *In case of being caught.*
$\boldsymbol{Q_3}$: *What would happen if her mother was not sleeping?* **A**: *The princess would be caught soon.*

Figure 1: Example questions that require different comprehension capabilities to answer.

1 Introduction

Question generation (QG) systems aim to generate natural language questions conditioned on a text passage. As a dual task of question answering (QA), QG is widely applied to create question-answer pairs as data augmentation for QA training (Zhang and Bansal, 2019; Liu et al., 2020; Chen et al., 2019a), help chatbots continue a conversation with human users (Mostafazadeh et al., 2016; Shum et al., 2018), and facilitate reading assessment (Heilman and Smith, 2010; Jia et al., 2021).

Most prior QG research has typically focused on generating factoid-based questions that are relevant to a piece of the fact of a single sentence (Zhou et al., 2017; Liu et al., 2019; Zhao et al., 2022). Recently, motivated by building the read comprehension (RC) systems that are competent in understanding and reasoning (Kaushik and Lipton, 2018; Sinha et al., 2019; Chen et al., 2019b), there is an increasing interest in developing systems that are capable of generating deep questions (Chen et al., 2020; Pan et al., 2020; Fei et al., 2022). However, these works generate diverse questions by relying on different surface-level mentioned information (Cheng et al., 2021; Kai et al., 2021) and consider primarily simple connections between two facts in the context (*e.g.* bridge and intersection). Less explored have been more facts and the deeper comprehension types between them (Desai et al., 2018), such as analysis of discourse relations (Johnstone, 2017), a thorough evaluation of stated arguments, and deduction of the high-level semantics (Gao et al., 2022). As shown in Figure 1, Q_1 asks for the mentioned facts in stories (*e.g.* "*The princess climbed out the window of the high tower*"), whereas Q_2 and Q_3 ask for a deep connection about the events (causal relation in Q_2 and future prediction in Q_3).

[†] Canada CIFAR AI Chair.
[‡] Corresponding authors.

Findings of the Association for Computational Linguistics: ACL 2023, pages 13833–13850
July 9-14, 2023 ©2023 Association for Computational Linguistics

We argue that generating questions with deeper comprehension brings two major benefits: (i) compared with factoid-based QG models, it reflects higher cognitive skills and requires an in-depth understanding of the input text and reasoning over relevant contexts, better imitating how human intelligence embodies the application and integration of skills; (ii) compared with existing deep QG models, it can help build more controllable questions with different comprehension types rather than literal information such as answer types. Based on such questions, we can better identify the downstream performance of QA systems in specific comprehension types, and assess their corresponding intrinsic ability, further allowing us to provide tailored guidance to them and improve training efficiency.

In this paper, we propose `SkillQG`: a question generation framework with controllable comprehension types. Specifically, we define the comprehension types as five skill dimensions ordered by cognitive complexity: REMEMBER, UNDERSTAND, ANALYZE, CREATE, and EVALUATE, which are inspired by Bloom's Taxonomy (Krathwohl, 2002), an educational schema by which teachers structure a curriculum to ensure that learners possess the necessary abilities before progressing to more complex tasks. Based on the definition, we can better differentiate questions from cognitive demands than previous surface-level information and formulate `SkillQG` as question generation conditioned on the given comprehension skill.

Furthermore, to improve the specificity of generating questions with a certain comprehension skill, we devise a set of prompts based on the indicative words and question templates of Bloom's Taxonomy. Using these prompts to iteratively elicit chain-of-thought reasoning of pre-trained language model (PLM), we explicitly generate question focuses (what to ask about) and skill-specific knowledge (how to ask it) to augment the input context.

Finally, to evaluate the `SkillQG` framework, we introduce evaluation protocols covering question content quality, skill controllability, and downstream QA performance improvement when incorporating the generated questions as additional training data. Our experimental results show that `SkillQG` can produce more relevant and skill-controllable questions compared to baseline QG models, and boost the QA performance significantly.

2 Methodology

In this section, we elaborate our `SkillQG` for generating skill-infused questions. Specifically, we first define the comprehension types of questions as a 5-dimensional skill schema, which is drawn upon Bloom's Taxonomy (Krathwohl, 2002) of research in cognitive science and describes the cognitive load of different levels of topics or samples. Based on this schema, we categorize the questions into different comprehension skills, regarding `SkillQG` as a conditional generator given a skill. Furthermore, to improve the controllability of the skill-infused questions, we adapt the indicative words and templates of Bloom's Taxonomy as a set of prompts to discover question focuses and skill-specific knowledge by prompting PLM iteratively. Finally, these question focuses and knowledge text act as auxiliary inputs to steer the question generator.

2.1 Formulation of Comprehension Types

Question generation has long served as an essential component for knowledge learning (Tobin, 1990; Lai et al., 2017) and assessing learning progress (Holme, 2003; Yudkowsky et al., 2019), especially asking questions about texts at various comprehension levels deepens the understanding of the text and aids in the learner's understanding and growing from what they have read (Holme, 2003). Among relevant research in cognitive science and pedagogy, Bloom's Taxonomy (Krathwohl, 2002) is one of the most basic and influential theories. Bloom's Taxonomy is a cognition model used for the classification of educational learning objectives into levels of complexity and specificity, including knowledge, comprehension, application, analysis, synthesis, and evaluation. Inspired by the hierarchical cognitive objectives of Bloom's Taxonomy, we define the comprehension types of questions as a 5-dimensional skill-based schema in Table 1. We sketch out the meaning of each comprehension skill with some examples as follows.

REMEMBER. The objective of this skill is to promote retention of the presented material in the same form as it exists. Therefore, it requires retrieving relevant content from what a model has read, *e.g. recall the dates of some events in the input passage*. Empirically, Sugawara et al. (2018) has shown that some questions can be answered correctly by just string-based matching with the given passage. In this study, the factoid-based questions (Zhou et al., 2017) involving a single fact with

Skill	Description	Example
REMEMBER	Retrieve relevant facts from input passage.	Factoid: what is X?, when did X happen? Definition: what does X mean?
UNDERSTAND	Construct meanings from recalled facts.	Interpreting: how would you rephrase X? Classifying: what is an example of X? Summarizing: what is the main idea of X? Comparing: how would you compare X and Y?
ANALYZE	Break facts into its constituent parts and determine how the parts are related to one another.	Explanation: what caused X? Consequence: what will X cause?
CREATE	Re-organize elements into a new pattern or structure.	Predicting: would it arrive on time?
EVALUATE	Make judgments based on established criteria.	Judgment: what do you think of X? Justification: why is X the case?

Table 1: Formulation of hierarchical comprehension skills. Skills are sorted by levels of cognition (lower to higher). See Section 2.1 for details.

explicit mentions and definition questions are categorized into this kind of comprehension skill.

UNDERSTAND. To build a holistically semantic representation of text from recalled facts in the passage, the easiest way is to build connections between the "new" knowledge to be gained and their prior knowledge. We exemplify four kinds of questions to represent this skill, consisting of interpreting (*e.g. paraphrase important speeches and documents*.), classifying (*e.g. classify observed or described cases of mental disorders*.), summarizing (*e.g. write a brief summary of the events portrayed on a videotape*.), and comparing (*e.g. compare historical events to contemporary situations*.).

ANALYZE. To step towards a higher comprehension skill, break-down-then-combination is required. This skill aims to break facts into their constituent parts and determine how the parts are related to one another. It usually involves the relationships between two events that are causally related where the prior events causally lead to the latter event in question. Similarly, Ko et al. (2020) reveals that cause-effect analysis is more challenging in understanding tasks than bridging or comparing the known facts, particularly for the cases where the passage contains no explicit causal conjunctions and corresponding background knowledge is required. Therefore, we include explanation (*e.g. why are the stock prices retreating*?) and consequence questions (*e.g. what happened to Timmy after he got in the hamper*?) in this skill.

CREATE. One of the highest cognitive levels is to put elements together to form a coherent whole. Although it seems impossible to empower a data-driven model with creative thinking, this skill asks for the possible outcome of a current event, which is predictable based on the existing information in the text. Inspired by the existing datasets that find textual clues and use them to guess what would happen next (Gao et al., 2022), we instantiate this comprehension skill as predicting questions (*e.g. How will the other animals treat the duckling*?).

EVALUATE. The other of the highest cognitive levels is making judgments based on criteria and standards. Because the criteria are constructed based on either elaborated details in the passage or external commonsense knowledge, this skill reflects the application of something known into a new scenario. Besides, this skill helps find out internal inconsistencies and also benefits the development of CREATE skill. We classify the judgment (*e.g. what do you think of the scientist's conclusions*?) and justification questions into this comprehension skill.

2.2 SkillQG

Based on the formulation of comprehension types, we follow the common question generation setup (Zhou et al., 2017; Liu et al., 2020) and frame SkillQG by a sequence-to-sequence question generator. Formally, given a context c, answer a and comprehension skill s, we aim to generate a question q that reflects the corresponding skill by modeling the conditional probability $p_\theta(q \mid c, s, a)$:

$$p_\theta(q \mid c, s, a) = \prod_{t=1}^{T} p_\theta(q_t \mid q_{<t}, c, s, a) \qquad (1)$$

where T is the length of generated question comprised of a sequence of tokens $q = \langle q_1, \cdots, q_t, \cdots, q_T \rangle$, and the generator is parameterized by θ. To improve the controllability of generation, we further guide the generator with question-worthy concepts and skill-specific knowledge. Precisely, we leverage chain-of-thought prompting (Wei et al., 2022; Madaan et al., 2022) of PLM, a prompting paradigm of successively eliciting relevant knowledge from PLM, to steer the

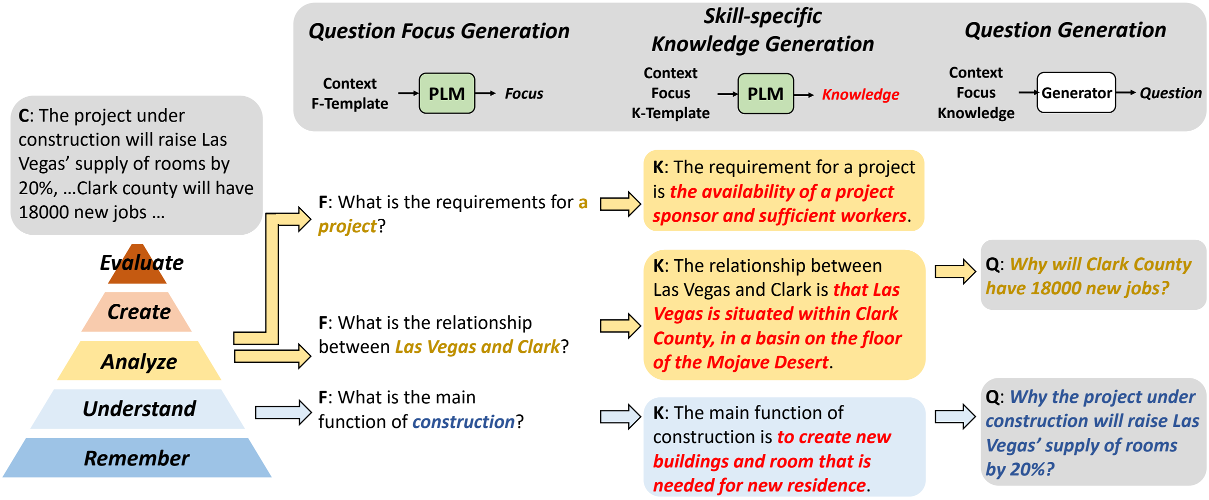

Question Focus Generation
Context
F-Template
PLM
Focus
Skill-specific Knowledge Generation
Context
Focus
K-Template
PLM
Knowledge
Question Generation
Context
Focus
Knowledge
Generator
Question
C: The project under construction will raise Las Vegas' supply of rooms by 20%, ...Clark county will have 18000 new jobs ...
Evaluate
Create
Analyze
Understand
Remember
F: What is the requirements for a project?
F: What is the relationship between Las Vegas and Clark?
F: What is the main function of construction?
K: The requirement for a project is the availability of a project sponsor and sufficient workers.
K: The relationship between Las Vegas and Clark is that Las Vegas is situated within Clark County, in a basin on the floor of the Mojave Desert.
K: The main function of construction is to create new buildings and room that is needed for new residence.
Q: Why will Clark County have 18000 new jobs?
Q: Why the project under construction will raise Las Vegas' supply of rooms by 20%?

Figure 2: Illustration of SkillQG pipeline. A skill-infused question (**Q**) is generated from the following steps: question focus generation, skill-specific knowledge generation, and question generation conditioned on the corresponding context (**C**), question focus (**F**), and elicited knowledge (**K**). The **PLM** represents an off-the-shelf GPT2 model, while the **generator** is initialized from a pre-trained BART model and fine-tuned on the training set.

generation of skill-infused questions. Based on it, we can first capture the question focuses and then externalize the implicit knowledge required for mastering the given comprehension skill.

As illustrated in Figure 2, we design several pairs of templates for each comprehension level, *i.e.* *F-template* and *K-template*, denoted as $\mathcal{T}_F$ and $\mathcal{T}_K$ respectively. These template pairs are with a form of information-seeking questions (Bruner, 1961), such as "*What is the definition of* _" and "*The definition of* _ *is* _", which can help PLM talk with itself to explicitly discover what it cares about when given a comprehension skill. More specifically, the $\mathcal{T}_F$ together with the input context is used to construct the prompt input for discovering possible question focuses by template-infilling, while the $\mathcal{T}_K$ can generate skill-related knowledge based on the context and question focuses. Finally, we take the generated knowledge as an auxiliary context and expect that it can contribute to improving the generation quality. Denoting the question focus and knowledge text as f and k, respectively, the above procedure can be formulated as:

$$f = \mathcal{M}(\mathcal{P}_F(c)) \tag{2}$$

$$k = \mathcal{M}(\mathcal{P}_K(c, f)) \tag{3}$$

$$c = \mathrm{Aug}(c, f, k) \tag{4}$$

where $\mathcal{M}$ denotes the employed PLM, *i.e.* GPT2, $\mathcal{P}_F$ and $\mathcal{P}_K$ represents the prompt input constructed by $\mathcal{T}_F$ and $\mathcal{T}_K$, respectively. $\mathrm{Aug}(c, f, k)$ means augmenting the original context with elicited question focus and knowledge text.

Question focus generation. To improve the controllability of generated questions, we take inspiration from the chain-of-thought prompting to capture question focuses and skill-related knowledge. Precisely, considering the close association between the comprehension skill and its involved narrative elements and questioning styles, we devise several pairs of *F-template* $\mathcal{T}_F$ and *K-template* $\mathcal{T}_K$ for each skill. An example is shown in Figure 2 and all of the templates are summarized in Appendix A. They are adapted from the indicative words and question templates of Bloom's Taxonomy. After that, question focus is generated by feeding the context and $\mathcal{T}_F$ into PLM. Following the prompt format of causal language models such as GPT2 (Radford et al., 2019), the prompt input $\mathcal{P}_F$ in Eq. 2 of question focus generation is built as:

$$\mathcal{P}_F(c) = \texttt{c From the context:}\ \mathcal{T}_F$$

Implicit knowledge generation. We further utilize *K-template* $\mathcal{T}_K$ to inquire PLM for generating skill-related knowledge. This kind of knowledge-externalization method has shown substantial improvements in zero-shot commonsense reasoning (Shwartz et al., 2020). Differing from Shwartz et al. (2020) heuristic designs for sample patterns of different datasets, our $\mathcal{T}_K$ is based on our hierarchical comprehension skills and collaborates with the question focus to develop a complete chain of thought of PLM. To be specific, the prompt input $\mathcal{P}_K(c, f)$ in Eq. 3 of skill-specific knowledge generation is represented as:

$$\mathcal{P}_K(c, f) = \boxed{\text{c}}\ \texttt{From the context:}\ \boxed{\mathcal{T}_F(f)}\ \boxed{\mathcal{T}_K}$$

where $\mathcal{T}_F(f)$ means infilling the $\mathcal{T}_F$ with corresponding generated question focus f.

Model training. To augment the original input context, we first fill the *F-template* and *K-template* with the generated question focus and knowledge text. After that, we append them to the original context to obtain the augmented input:

$$\text{Aug}(c, f, k) = \boxed{\text{c}}\ \boxed{\mathcal{T}_F(f)}\ \boxed{\mathcal{T}_K(k)}$$

Furthermore, to help our `SkillQG` learn the relationship between multiple pieces of input text and capture their functions, we utilize natural language prompts as well as special tokens as the delimiter to combine the multiple inputs into a single sequence, *i.e.* including the knowledge-augmented context c, answer text a, and skill s. This kind of method has been proven to help better learn the relationship between multiple pieces of input text and capture their functions, improving performance on various tasks (Schick and Schütze, 2021; Zhou et al., 2022). Formally, the input sequence fed into our question generator is as follows:

$$\texttt{[CXT]}\ \boxed{\text{c}}\ \texttt{[ANS]}\ \boxed{\text{a}}\ \texttt{[SKL]}\ \boxed{\text{s}}\ \texttt{Ask a question:}$$

where `[CXT]`, `[ANS]` and `[SKL]` are special tokens to mark the boundary between multiple input sequences (Radford et al., 2019). $\boxed{\text{c}}$, $\boxed{\text{a}}$ and $\boxed{\text{s}}$ are formulated as the corresponding context text, answer text, and skill name, respectively. After that, the sequence is fed into a BART-base (Lewis et al., 2020) question generator which models the probabilities $p_\theta\left(q \mid c, s, a\right)$ in Eq. 1 by minimizing the conditional negative log-likelihood (NLL) loss:

$$\mathcal{L}_{QG} = -\sum_{t=1}^{T} \log \hat{p_\theta}\left(q_t \mid q_{<t}, c, s, a\right) \qquad (5)$$

where $\hat{p_\theta}\left(q_t \mid q_{<t}, c, s, a\right)$ denotes the predicted probability for the token in the reference question.

3 Experiments

Datasets. We employ the official train and dev splits of FairytaleQA dataset (Xu et al., 2022) to train our `SkillQG`. This dataset, focusing on narrative comprehension of English text for both machines and young children, is annotated with seven fine-grained skills comprised of Character, Setting, Action, Feeling, Causal relationship, Outcome resolution, and Prediction. Its annotation process is supervised by three experts in literacy education and its categorization of questions is based on prior educational research (Paris and Paris, 2003) so that we can easily match the samples of the FairytaleQA dataset with our defined skill schema. Table 2 presents this mapping relationship and corresponding breakdown statistics of the dataset.

Annotation	Count	Percentage (%)	Skill
Character	1172	11.08	REMEMBER
Setting	630	5.95	REMEMBER
Action	3342	31.59	UNDERSTAND
Feeling	1024	9.68	EVALUATE
Causal rel.	2940	27.79	ANALYZE
Outcome res.	986	9.42	ANALYZE
Prediction	486	4.59	CREATE

Table 2: Breakdown statistics of the FairytaleQA dataset and its mapping to our proposed skill-based schema.

Baselines. We compare SkillQG to the following two types of QG baselines. The first type is typically trained without the knowledge input, including NQG++ (Zhou et al., 2017), and QAG (Yao et al., 2022). The other is knowledge-augmented generators consisting of CsQG (Xin et al., 2021) and CQG (Fei et al., 2022), which retrieve external knowledge from knowledge bases or generate knowledge with another model and regard the knowledge as extra context to generate questions.

3.1 Evaluation Protocol

Automatic evaluation metrics. We use standard question generation metrics to evaluate the question quality from the following three aspects. The **syntactic similarity** between generated questions and reference is measured by BLEU-4 (Papineni et al., 2002) and ROUGE-L (Lin, 2004). The **answerability** and structural integrity of generated questions is gauged by Q-BLEU-4 (Nema and Khapra, 2018). The **relvance** of generated questions to the reference is evaluated by BERTScore (Zhang et al., 2019), while that to the given context is evaluated by the factuality dimension of CTC (Deng et al., 2021) and BARTScore (Yuan et al., 2021).

Human evaluation. We conduct a voluntary human evaluation to analyze `SkillQG` by asking five annotators to rate the quality of candidates generated by different models when using 300 $\langle passage, skill, answer, question\rangle$ samples in the unseen test split as the input. For **question**

Method	Q-B4	R-L	B4	BE.S	CTC	BA.S
NQG++	0.503	0.421	0.141	0.342	0.328	0.266
QAG	0.552	0.427	0.146	0.424	0.408	0.333
QTD	0.576	0.431	0.150	0.478	0.456	0.372
CsQG	0.592	0.431	0.151	0.506	0.485	0.393
CQG	0.609	0.433	0.153	0.532	0.510	0.415
SkillQG	**0.656**	**0.440**	**0.159**	**0.620**	**0.596**	**0.485**

Table 3: Quantitative results in terms of answerability, syntactic similarity, and relevance evaluation metrics on the FairytaleQA dataset. Please refer to Section 3.1 for the full name of employed metrics. The best result is marked as **bold**.

Method	Grammaticality		Answerability		Relevance	
	wins%	ties%	wins%	ties%	wins%	ties%
QAG	46.3	8.7	47.0	9.0	41.7	20.5
QTD	48.7	9.3	48.3	2.7	48.2	6.3
CsQG	49.0	7.3	49.2	5.3	49.4	6.0
CQG	50.3	4.0	51.3	5.0	52.0	7.0
SkillQG	**53.0**	10.0	**53.6**	5.6	**54.0**	3.7

Table 4: Human evaluation results on question content quality. We show the percentage of times annotators prefer each variant to NQG++ and ties.

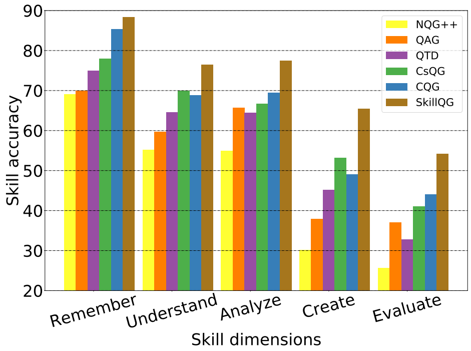


Figure 3: Human evaluation results on skill controllability, which is computed by comparing the given with the annotated skill. We depict the accuracy for each skill alongside the horizontal axis.

content quality, following the human criteria of QG elaborated by Rus et al. (2010) and Nema and Khapra (2018), we conduct pairwise comparison where we present a context and two questions made by two different models and ask the annotators to choose the better of the two or "tie" in terms of grammaticality, answerability, and relevance. We report the percentage of times annotators prefer each model to NQG++ and ties, *i.e.* wins/ties ratio. For **skill controllability**, we ask the annotator to read the context, the generated questions, and the corresponding answer, choose the evidence sentences in context, and then respectively annotate the required comprehension skill from our defined 5-dimensional skill schema. *Please refer to Appendix C for more details about the annotation.*

3.2 Main Results

Table 3 summarizes the quantitative results on the FairytaleQA dataset. On the one hand, compared with the baselines without extra knowledge (*i.e.* NQG++, QAG, QTD), `SkillQG` achieves obviously higher metrics scores in terms of answerability, and relevance, demonstrating the significant contribution of incorporating extra knowledge and question focuses to generate the questions. The comparable results on syntactic similarity metrics may be attributed to the wrong penalization of these metrics to the novel generation of our `SkillQG`. On the other hand, `SkillQG` consistently outperforms all the knowledge-augmented baselines (*i.e.* CsQG and CQG) by a considerable margin (*i.e.* gain ratio $> 5\%$), which indicates the effectiveness of externalized knowledge by our devised prompts.

Inter-annotator agreement. For the examined two aspects of human evaluation, *i.e.* question content quality and skill-controllability, the inter-annotator Krippendorff's α for them are 87.20 and 90.73, respectively, which demonstrates an acceptable level of agreement (> 80%) between annotators (Krippendorff, 2004). Then, we invite the annotators to discuss the few annotation conflicts before the final annotations are determined. Specifically, since the skill-controllability is calculated as the accuracy between the given skill and the annotated one, the annotators are asked to discuss the discrepancies of annotated skills and reach as unique skill annotation as possible for every sample, which then is used as the final annotation.

Question content quality. As shown in Table 4, the pairwise comparisons show that `SkillQG` produces more grammatical and relevant questions and questions that are mostly answerable ($> 50\%$), compared to all baseline models. Besides, knowledge-augmented baselines (lower part in Table 4) consistently receive more preference from annotators than others (upper part in Table 4). It demonstrates that the generated skill-specific knowledge indeed enhances the question content and relevance.

Skill controllability. Figure 3 reports the consistency between the given skill name that `SkillQG` generates questions conditioned on and the one chosen by the annotators, *i.e.* skill accuracy. We can see our `SkillQG` surpasses other baselines by a significant margin, and this becomes more obvious to the skills that have a relatively smaller number

Baseline	Q-B4	R-L	B4	BE.S	CTC	BA.S
concat-only (M_1)	0.626	0.436	0.155	0.569	0.550	0.445
symbol-only (M_2)	0.639	0.437	0.156	0.581	0.562	0.457
prompt-only (M_3)	0.641	0.438	0.158	0.598	0.572	0.466
generator (M_4)	0.620	0.434	0.155	0.558	0.536	0.435
conceptnet (M_5)	0.636	0.436	0.156	0.582	0.559	0.455
SkillQG	**0.656**	**0.440**	**0.159**	**0.620**	**0.596**	**0.485**

Table 5: Quantitative results of ablation experiments. The best result is marked as **bold**.

of samples in the dataset, *i.e.* around 30% gain in CREATE and EVALUATE dimension. It justifies that `SkillQG` can not only successfully control the comprehension skill of generated questions, but also be able to learn the underrepresented skills in the dataset, owing to the built prompts containing indicative words of different comprehension skills and the rich skill-specific knowledge of language models. *Please refer to Section D for more results.*

3.3 Ablation Analysis

We conduct ablation experiments and summarize the results in Table 5 from the following aspects.

First, *How do the special symbols and prompts of input representation contribute to the generation quality?* The first three baselines combine multiple input sequences (*i.e.* context, answer, and skill) with the concatenation operation, special symbols or natural language prompts, denoted as "concat-only (M_1)", "symbol-only (M_2)" and "prompt-only (M_3)", respectively. As shown in Table 5, we can observe that M_1 achieves worse performance than M_2 and M_3, demonstrating that simple concatenation operation cannot encode the input sequences well. Besides, both M_2 and M_3 degrade the performance w.r.t. `SkillQG`, showing the integration of special symbols and natural language prompts can help the generator better understand the relationship between multiple input sequences and improve the final quality.

Second, *What is the impact of question focus and skill-specific knowledge?* The baseline "generator (M_4)" does not utilize skill-specific knowledge to augment the context and trains the question generator directly, *i.e.* a BART model for question generation, while the baseline "conceptnet (M_5)" is trained in the similar setting to `SkillQG` but its extra knowledge is attained by retrieving the ConceptNet rather than inquiring PLM. We perform alignment between the context and ConceptNet following the embedding-based matching as Zhou et al. (2022). In Table 5, we can find that the contribution of extra knowledge from PLM (`SkillQG` v.s. M_4) is more significant than that from the ConceptNet (M_5 v.s. M_4). A possible reason is that chain-of-thought prompting of PLM can reflect better relevance and specificity of knowledge to the given context and the required comprehension skill compared to matching with the limited number of triplets in a knowledge base. This result also agrees with the recent study on evaluating PLM as a knowledge base (Heinzerling and Inui, 2021).

3.4 Boosting QA Performance using Unlabeled Corpus

We further evaluate whether the skill-controllable questions can improve QA performance through data augmentation and help us better understand the QA models' intrinsic ability. Specifically, we first devise an information extractor to obtain $\langle passage, skill, answer \rangle$ combinations on an unlabeled corpus, *i.e.* the passage without annotations of the question, answer, and skill. After that, we feed the extracted combinations of ⟨ *passage*, *answer*, *skill* ⟩ into `SkillQG` to generate skill-infused questions. Finally, we put the generated questions into the FairytaleQA training set and train a QA model with such an augmented dataset to further evaluate the effectiveness of our `SkillQG`.

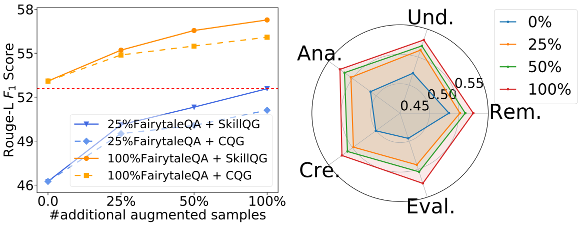
Information extraction. Since the answer and required skill are dependent on each other, we cannot sample the combinations of ⟨ *passage*, *answer*, *skill* ⟩ randomly. Following the widely adopted solutions (Liu et al., 2020; Ghanem et al., 2022), we decompose the process into two steps to sequentially sample the required skills, and corresponding answers to select reasonable combinations. Formally, the sampling procedure can be written as:

$$p\,(s, a \mid c) = p\,(s \mid c)\, p\,(a \mid c, s) \qquad (6)$$

where $p\,(s \mid c)$ and $p\,(a \mid c, s)$ are devised as a model-based and rule-based extractor, respectively.

On the one hand, $p\,(s \mid c)$ is formulated as a multi-label classification task because a passage may involve more than one skill. We first fine-tune a DistilBERT model (Sanh et al., 2019) on the FairytaleQA dataset to learn skill-related patterns in the context. After that, we use it to predict the candidate skills when given an unlabeled passage.

On the other hand, we borrow the statistical analysis on the FairytaleQA dataset from Yao et al. (2022) and implement $p\,(a \mid c, s)$ using heuristic rules. Specifically, REMEMBER and EVALUATE skills, *i.e.* the narrative elements consisting of character, setting, and feelings, are usually based on



(a) Comparison of different question generators.

(b) Breakdown analysis of the overall QA performance.

Figure 4: Overall and decomposed performance of the state-of-the-art QA model on the FairytaleQA dataset, augmented with data generated by question generators.

the named entities, such as a mentioned name and a particular place. Therefore, we resort to the Spacy tool (Honnibal and Montani, 2017) to extract named entities as the candidate answers. Other skills, *i.e.* the narrative elements consisting of action, causal relationship, outcome resolution, and prediction are mainly made up of the action events. Thus, we first leverage Propbank's semantic role labeler (Johansson and Nugues, 2008) to extract the trigger verb as well as the involved subject and object and then concatenate them into a complete sentence as the candidate answers.

We conduct the sampling procedure on the passages of FairytaleQA training set and discard all their annotations, then feed the extracted ⟨ *passage*, *answer*, *skill* ⟩ into SkillQG by keeping all beam search (size=8) outputs for each sample. Consequently, we can generate diverse questions for the existing paragraphs in the FairytaleQA training set. Finally, we randomly select 80,000 candidate questions and augment the FairytaleQA training set with them. As a comparison, following the same setting as above, we design a baseline by utilizing CQG as the question generator, which is one of the most competitive metrics in Table 3.

We train a state-of-the-art QA baseline (Xu et al., 2022) on the augmented dataset to further evaluate the quality of generated questions. Following Xu et al. (2022), we report the QA performance in Rouge-L F_1 Score which is a commonly used metric for generative question answering. The results in a high-resource setting (with the whole FairytaleQA training set) and a low-resource setting (with only 25% of data sampled from the original FairytaleQA training set) are illustrated in Figure 4a. We can observe that the questions generated by SkillQG can improve the QA performance to a greater extent than CQG under both settings. In particular, the QA model under the low-resource setting achieves a comparable performance to the high-resource setting when leveraging the 100% additional samples generated by our SkillQG.

Furthermore, Figure 4b illustrates the decomposed performance of SkillQG under the low-resource setting (*i.e.* "25%FairytaleQA + SkillQG" setting shown in Figure 4a) alongside the defined skill dimension. This result shows that the questions generated by SkillQG can significantly boost all of the comprehension capabilities for the QA model. Among them, the cognitively challenging ones that the QA model struggles in, such as EVALUATION and CREATE, even achieve the largest improvement. It demonstrates that the skill-controllable questions that generated by the SkillQG can compensate for the limited number of training samples in the FariytaleQA dataset and are favorable for the fine-grained assessment of comprehension capability of QA models.

4 Related Work

Deep question generation. Previous QG systems mainly generated factoid-based questions by a sequence-to-sequence model (Zhou et al., 2017; Liu et al., 2019), a PLM (Liu et al., 2020), or a graph-based architecture (Talmor and Berant, 2018; Kumar et al., 2019). Recent-emerged QG models aimed at generating questions that require deep reasoning. On the one hand, Cheng et al. (2021) proposed to generate difficulty-controllability questions through step-by-step rewriting, while Bi et al. (2021) decoded multi-hop questions by a soft template. On the other hand, Yao et al. (2022) and Zhao et al. (2022) devised educational question generators to facilitate the assessment of children's literacy. Our SkillQG is inspired by their fine-grained analysis but driven by the motivation that generating questions with deep comprehension is beneficial to QA training. More recently, Cao and Wang (2021) charted a new question ontology, but they focused on constructing diversified open-ended questions from the specified question types.

Knowledge-augmented generation. Although explicit knowledge generation has been explored in natural language understanding (Liu et al., 2022; Wei et al., 2022), similar research on natural language generation (Zhou et al., 2022), especially for QG is relatively rare (Rajani et al., 2019). Xin et al. (2021) retrieved knowledge triplets from ConceptNet (Speer et al., 2017) to enhance the QG models, while Fei et al. (2022) adopted a graph attention networks (GAT) (Veličković et al., 2018) to capture focuses for question generators. We considered the lessons of these works and extend the knowledge source with pre-trained language models.

5 Conclusion

Existing QG systems focus on the literal nature of questions and rarely consider the comprehension types of the generated questions. To better assess and improve machine reading comprehension models, we propose SkillQG to generate questions with controllable comprehension types. Besides, we engage the question focus and specific knowledge to improve the controllability of generation. Empirical results show that SkillQG outperforms baselines while achieving a significant performance boost in downstream QA training.

6 Limitations

Our work proposes a new QG framework, namely SkillQG, to frame the comprehension skill required by a question and generate the corresponding comprehension-oriented questions. The limitations are three-fold:

Firstly, we propose a new skill-based schema for the comprehension nature of questions and map the existing annotations on narrative elements of the FairytaleQA dataset to it and conduct our experiments. This kind of mapping might not reflect the required skills accurately since a narrative element can cover more than one comprehension types. Furthermore, although our proposed skill-based schema is drawn upon general text comprehension, SkillQG is only verified on the FairytaleQA dataset and lacks the analysis on generalizability. However, identifying skills and correlations with comprehension skills on new datasets can be challenging because SkillQG may struggle with the input passage with a relatively simple discourse structure, which usually does not contain complicated relations. One remedy to this issue could be collecting a new QA dataset with the annotations following our proposed schema. We regard it as our future work and deem designing a new annotation specification a promising direction.

Besides, although we boost the downstream QA performance in Section 3.4 by augmenting the original training set with generated questions, the final performance (56.9%) is also far behind the human performance (64.4%) reported by Xu et al. (2022). However, the breakdown analysis of QA performance demonstrates that SkillQG can strengthen all of the comprehension capabilities, especially the challenging ones. As a result, generating questions that are matched with the current comprehension capabilities of the QA model and co-evolving the QA system and corresponding QG system, could be two interesting research topics.

Last but not least, our SkillQG is built on the PLMs of general domains, ignoring the domain-specific and multilingual application. The backbone PLMs are also shown a biased representation, such as race and gender (Gonen and Goldberg, 2019). Therefore, additional evaluation protocols are left for our future work.

Acknowledgements

We would like to thank anonymous reviewers for their valuable comments and suggestions. This work has been supported in part by the NSFC (No. 62272411), the Zhejiang NSF (LR21F020004), Ant Group and Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, the Canada CIFAR AI Chair Program, and the Canada NSERC Discovery Grant (RGPIN-2021-03115).

References

Jacopo Amidei, Paul Piwek, and Alistair Willis. 2019. The use of rating and Likert scales in natural language generation human evaluation tasks: A review and some recommendations. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 397–402, Tokyo, Japan. Association for Computational Linguistics.

Sheng Bi, Xiya Cheng, Yuan-Fang Li, Lizhen Qu, Shirong Shen, Guilin Qi, Lu Pan, and Yinlin Jiang. 2021. Simple or complex? complexity-controllable question generation with soft templates and deep mixture of experts model. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4645–4654, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jerome S Bruner. 1961. The act of discovery. *Harvard educational review*, 31:21–32.

Shuyang Cao and Lu Wang. 2021. Controllable open-ended question generation with a new question type ontology. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6424–6439, Online. Association for Computational Linguistics.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.

Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2019a. Bidirectional attentive memory networks for question answering over knowledge bases. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2913–2923, Minneapolis, Minnesota. Association for Computational Linguistics.

Yu Chen, Lingfei Wu, and Mohammed J Zaki. 2019b. Reinforcement learning based graph-to-sequence model for natural question generation. In *The Eighth International Conference on Learning Representations (ICLR 2020)*.

Yu Chen, Lingfei Wu, and Mohammed J Zaki. 2020. Toward subgraph guided knowledge graph question generation with graph neural networks. *arXiv preprint arXiv:2004.06015*.

Yi Cheng, Siyao Li, Bang Liu, Ruihui Zhao, Sujian Li, Chenghua Lin, and Yefeng Zheng. 2021. Guiding the growth: Difficulty-controllable question generation through step-by-step rewriting. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5968–5978, Online. Association for Computational Linguistics.

Mingkai Deng, Bowen Tan, Zhengzhong Liu, Eric Xing, and Zhiting Hu. 2021. Compression, transduction, and creation: A unified framework for evaluating natural language generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7580–7605, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Takshak Desai, Parag Dakle, and Dan Moldovan. 2018. Generating questions for reading comprehension using coherence relations. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 1–10, Melbourne, Australia. Association for Computational Linguistics.

Zichu Fei, Qi Zhang, Tao Gui, Di Liang, Sirui Wang, Wei Wu, and Xuanjing Huang. 2022. CQG: A simple and effective controlled generation framework for multi-hop question generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6896–6906, Dublin, Ireland. Association for Computational Linguistics.

Lingyu Gao, Debanjan Ghosh, and Kevin Gimpel. 2022. "what makes a question inquisitive?" a study on type-controlled inquisitive question generation. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 240–257, Seattle, Washington. Association for Computational Linguistics.

Bilal Ghanem, Lauren Lutz Coleman, Julia Rivard Dexter, Spencer von der Ohe, and Alona Fyshe. 2022. Question generation for reading comprehension assessment by modeling how and what to ask. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2131–2146, Dublin, Ireland. Association for Computational Linguistics.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 60–63, Florence, Italy. 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.

Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1772–1791, Online. Association for Computational Linguistics.

Thomas Holme. 2003. Assessment and quality control in chemistry education. *Journal of Chemical Education*, 80(6):594.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

M Honnibal and I Montani. 2017. Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *https://spacy. io*.

Xin Jia, Wenjie Zhou, Xu Sun, and Yunfang Wu. 2021. Eqg-race: examination-type question generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13143–13151.

Richard Johansson and Pierre Nugues. 2008. Dependency-based semantic role labeling of PropBank. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 69–78, Honolulu, Hawaii. Association for Computational Linguistics.

Barbara Johnstone. 2017. *Discourse analysis*. John Wiley & Sons.

Shen Kai, Lingfei Wu, Siliang Tang, Yueting Zhuang, Zhuoye Ding, Yun Xiao, Bo Long, et al. 2021. Learning to generate visual questions with noisy supervision. *Advances in Neural Information Processing Systems*, 34:11604–11617.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, Brussels, Belgium. Association for Computational Linguistics.

Wei-Jen Ko, Te-yuan Chen, Yiyan Huang, Greg Durrett, and Junyi Jessy Li. 2020. Inquisitive question generation for high level text comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6544–6555, Online. Association for Computational Linguistics.

David R Krathwohl. 2002. A revision of bloom's taxonomy: An overview. *Theory into practice*, 41(4):212–218.

Klaus Krippendorff. 2004. Reliability in content analysis: Some common misconceptions and recommendations. *Human communication research*, 30(3):411–433.

Vishwajeet Kumar, Yuncheng Hua, Ganesh Ramakrishnan, Guilin Qi, Lianli Gao, and Yuan-Fang Li. 2019. Difficulty-controllable multi-hop question generation from knowledge graphs. In *International Semantic Web Conference*, pages 382–398. Springer.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Bang Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. 2020. Asking questions the human way: Scalable question-answer generation from text corpus. In *Proceedings of The Web Conference 2020*, pages 2032–2043.

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. In *The World Wide Web Conference*, pages 1106–1118.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022. Memory-assisted prompt editing to improve gpt-3 after deployment. *arXiv preprint arXiv:2201.06009*.

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 and Mitesh M. Khapra. 2018. Towards a better metric for evaluating question generation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3950–3959, Brussels, Belgium. Association for Computational Linguistics.

Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic graphs for generating deep questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1463–1475, Online. 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 the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Alison H Paris and Scott G Paris. 2003. Assessing narrative comprehension in young children. *Reading Research Quarterly*, 38(1):36–76.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. 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*. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Heung-Yeung Shum, Xiao-dong He, and Di Li. 2018. From eliza to xiaoice: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering*, 19(1):10–26.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4615–4629, Online. Association for Computational Linguistics.

Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L. Hamilton. 2019. CLUTRR: A diagnostic benchmark for inductive reasoning from text. 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 4506–4515, Hong Kong, China. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Saku Sugawara, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. 2018. What makes reading comprehension questions easier? In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4208–4219, Brussels, Belgium. Association for Computational Linguistics.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. 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 641–651, New Orleans, Louisiana. Association for Computational Linguistics.

Kenneth Tobin. 1990. Research on science laboratory activities: In pursuit of better questions and answers to improve learning. *School science and Mathematics*, 90(5):403–418.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625, Seattle, United States. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jia Xin, Wang Hao, Yin Dawei, and Wu Yunfang. 2021. Enhancing question generation with commonsense knowledge. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 976–987, Huhhot, China. Chinese Information Processing Society of China.

Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Li, Nora Bradford, Branda Sun, Tran Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu, and Mark Warschauer. 2022. Fantastic questions and where to find them: FairytaleQA – an authentic dataset for narrative comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 447–460, Dublin, Ireland. Association for Computational Linguistics.

Bingsheng Yao, Dakuo Wang, Tongshuang Wu, Zheng Zhang, Toby Li, Mo Yu, and Ying Xu. 2022. It is AI's turn to ask humans a question: Question-answer pair generation for children's story books. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 731–744, Dublin, Ireland. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34.

Rachel Yudkowsky, Yoon Soo Park, and Steven M Downing. 2019. *Assessment in health professions education*. Routledge.

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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Zhenjie Zhao, Yufang Hou, Dakuo Wang, Mo Yu, Chengzhong Liu, and Xiaojuan Ma. 2022. Educational question generation of children storybooks via question type distribution learning and event-centric summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5073–5085, Dublin, Ireland. Association for Computational Linguistics.

Pei Zhou, Karthik Gopalakrishnan, Behnam Hedayatnia, Seokhwan Kim, Jay Pujara, Xiang Ren, Yang Liu, and Dilek Hakkani-Tur. 2022. Think before you speak: Explicitly generating implicit commonsense knowledge for response generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1237–1252, Dublin, Ireland. 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 *National CCF Conference on Natural Language Processing and Chinese Computing*, pages 662–671. Springer.

Skill	*F-template* $\mathcal{T}_F$	*K-template* $\mathcal{T}_K$
REMEMBER	What is the definition of <blank> What are the properties of <blank> How would you describe <blank>	The definition of <focus> is <blank> The properties of <focus> are <blank> <focus> is a <blank>
UNDERSTAND	What is the purpose of <blank> What is the main function of <blank> How would you classify the type of <blank> What is the difference between <blank> How would you rephrase the meaning of <blank> How would you summarize <blank>	The purpose of <focus> is to <blank> The main function of <focus> is <blank> The type of <focus> is <blank> The difference between <focus> is <blank> The meaning of <focus> is <blank> The summarization of <focus> is <blank>
ANALYZE	How would <blank> feel afterwards? What happened as a result of <blank> What might have caused <blank> Why did <blank> do this?	<focus> felt <blank> As a result of <focus>, <blank> The cause of <focus> was <blank> <focus> did this because they wanted <blank>
CREATE	What will <blank> want to do next? What will happen to <blank> next? What would happen if <blank> What will be the outcome if <blank>	<focus> want <blank> <focus> will <blank> If <focus>, <blank> If <focus>, the output will be <blank>
EVALUATE	Why do you recommend <blank> Why is it better that <blank>	You recommend <focus> because <blank> It is better that <focus> because <blank>

Table 6: *F-template* and *K-template* used for each defined comprehension skills.

A Focus and Knowledge Templates

We manually design a few generic templates to conduct chain-of-thought prompting to the PLM, which are with a form of information-seeking questioning pairs. Table 6 summarizes the employed focus and knowledge templates, *i.e.* *F-template* and *K-template*, where "<blank>" means the placeholder to be filled with the generated question focus and knowledge text, while "<focus>" represents the question focus text. In addition to the prefix-style templates, *i.e.* "<blank>" is located in the trailing part of the input prompt, we also resort to cloze-style templates, *i.e.* "<blank>" is in the middle part of the input prompt, such as "*How would <blank> feel afterwards*?". Our employed PLM, *i.e.* GPT2, is based on causal language modeling, and does not well in finishing such cloze-style template. Therefore, we leverage Spacy tool (Honnibal and Montani, 2017) to extract named entities as the generated question focus for these prompts and follow the same pipeline elaborated in Section 2.2.

As shown in the table, each comprehension skill is equipped with at least 2 *F-template*s. We generate 5 question focuses for each *F-template* using Nucleus sampling (Holtzman et al., 2019) with well-adopted p = 0.2, *i.e.* sampling from the top 20% tokens (Holtzman et al., 2019) and obtain the full question focus when the eos token generates. In addition, each *F-template* is paired with a corresponding *K-template*. We use Nucleus sampling with p = 0.5 to generate 10 pieces of knowledge text for each *K-template*.

B Implementations Details

The question generator of our SkillQG is built on the basis of a BART-base model (Lewis et al., 2020), while the skill-specific knowledge is generated by a GPT2 model (Radford et al., 2019). The number of their parameters is around 140M and 117M, respectively. Both of them are first initialized by the pre-trained parameters of the HuggingFace Transformers package (Wolf et al., 2020). After that, the parameters of the GPT2 model will be frozen and ones of the BART-base model will be fine-tuned on the training set. Our information extractor is initialized by a DistilBERT model with about 66M parameters from the HuggingFace package. AdamW (Loshchilov and Hutter, 2018) optimizer with weight decay 5e-4 and epsilon 8 is used to fine-tune the model with a maximum sequence length of 384. During training, we extract mentioned sections of a whole passage as the input context, which is annotated in the $corr\_sec$ field of the FairytaleQA dataset. The learning rate warms up over the first 10% steps and then decays linearly to 0 for all experiments with training batch size 16 and maximum iteration 40,000. The whole training takes 25 hours on 4 NVIDIA GTX 2080Ti GPUs. We use the official train split to fine-tune the question generator of our SkillQG and employ grid search to determine the hyper-parameters based on the val split. We report the average results of ten

	Candidate question	Instruction	Description
Grammaticality	**A**. How many solo tackles did Von Miller make at Super Bowl? **B**. What site is locate in the San Franc?	A wins B.	B is not grammatically correct.
Answerability	**A**. How many Grammys has Lady Gaga won? **B**. How many ~~professors~~ does the Warsaw University of Technology employ?	A wins B.	B misses some important information, such as named entities, relation words, and question words.
Relevance	**A**. What is the axis of Warsaw which divides it into two parts? **Context of A**. [. . .] the Vistula River is the specific axis of Warsaw, which divides the city into two parts [. . .] **B**. How big is the greater metropolitan area? **Context of B**. [. . .] within a greater metropolitan area of 2.666 million residents [. . .]	A wins B.	B is partially relevant but unable to be grounded by the context.

Table 7: Scoring examples for the human evaluation on the question content quality. The problematic words in corresponding candidate questions are marked in red.

runs for automatic evaluation and conduct the human evaluation on the candidates generated by a single run.

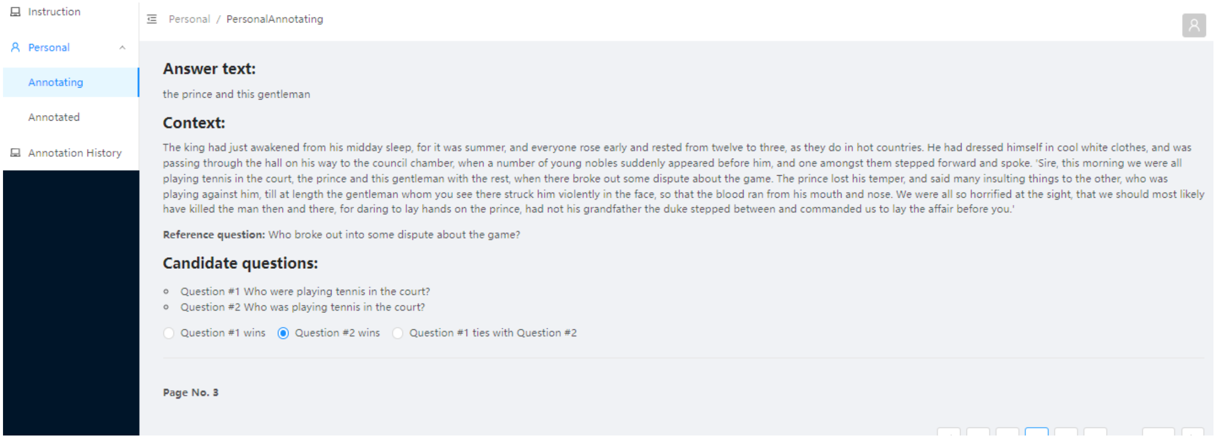
C Annotation Details

Our human evaluation is conducted by a total of five annotators. All of the annotators are from China, between 25 and 30 years old, competent in English, and studying as Computer Science graduates. They are informed of the necessary background knowledge on QG and evaluation for QG, as well as detailed annotation instructions along with examples when participating in our study. In addition, they gladly volunteered to provide their assistance without being compensated in any form. The candidate questions are anonymized and evaluated in the following aspects:

- **Question content quality.** Following the human criteria elaborated in QG-STEC Task B (Rus et al., 2010), we check whether a question is well-formed, answerable, and relevant to the context. Besides, previous works have shown that pairwise comparison produces a more reliable evaluation than directly asking humans to score the candidate (Amidei et al., 2019; Celikyilmaz et al., 2020). Therefore, we present a context and two questions made by two different models and ask the annotators to choose the better of the two or "tie". Specifically, we first show the annotators a candidate question generated by NQG++ and another one generated by others as well as the corresponding input context and answer text. After that, we ask the annotators to compare the two questions in terms of grammaticality, answerability, and relevance. To better guide the annotators to distinguish between high-quality candidate questions and low-quality ones, we also show the annotators clearing examples as presented in Table 7.

- **Skill-controllability.** It checks the consistency between the given skill that the question generator is conditioned on and the one chosen by the annotators, *i.e.* skill accuracy. This kind of fine-grained annotation is inspired by the recent study on the educational question generation (Ghanem et al., 2022) and is used to evaluate the controllability of generation. Before the annotation, we show the template samples for each comprehension skill as summarized in Table 1 to the annotators. During annotation, they are informed of the annotation instruction in the three steps. (1) Make a statement using the reference question and gold standard answer. (2) Extract sentences from the context required to support the statement. (3) Re-read our defined skill-based schema in Table 1 and choose only one required skill to understand an entailment from extracted context to the statement.

- **Knowledge quality.** Since evaluating the overall quality of knowledge is challenging (Heinzerling and Inui, 2021; West et al., 2022), this aspect checks the groundedness and relevance of our generated knowledge text to the given context. Specifically, we first show the annotators the input context, candidate question, answer text, and corresponding generated knowledge text. After that, we ask the annotator to answer two questions ("*does the generated knowledge make sense*" and "*is the generated knowledge relevant to the input context*"). Only our `SkillQG` and knowledge-augmented baselines are involved with this aspect of evaluation, and the annotation option is either *yes* or *no*.



Answer text:

the prince and this gentleman

Context:

The king had just awakened from his midday sleep, for it was summer, and everyone rose early and rested from twelve to three, as they do in hot countries. He had dressed himself in cool white clothes, and was passing through the hall on his way to the council chamber, when a number of young nobles suddenly appeared before him, and one amongst them stepped forward and spoke. 'Sire, this morning we were all playing tennis in the court, the prince and this gentleman with the rest, when there broke out some dispute about the game. The prince lost his temper, and said many insulting things to the other, who was playing against him, till at length the gentleman whom you see there struck him violently in the face, so that the blood ran from his mouth and nose. We were all so horrified at the sight, that we should most likely have killed the man then and there, for daring to lay hands on the prince, had not his grandfather the duke stepped between and commanded us to lay the affair before you.'

Reference question: Who broke out into some dispute about the game?

Candidate questions:

- Question #1 Who were playing tennis in the court?
- Question #2 Who was playing tennis in the court?

○ Question #1 wins ● Question #2 wins ○ Question #1 ties with Question #2

Page No. 3

Figure 5: A screenshot of our human annotation process.

Method	Makes Sense	Relevant
CsQG	85.60%	85.00%
CQG	77.70%	78.90%
SkillQG	85.30%	90.40%

Table 8: Human evaluation results on knowledge quality. We report the percentage of *yes* answers for the two involved questions described in Section 3.1 .

As shown in Figure 5, we develop a web application to present and collect the human evaluation results automatically. This software can send the candidate samples to the annotators, guide them to evaluate samples from the aforementioned three dimensions and finally post the annotation results to our server. These results are based on the original collection of the dataset and will not violate the rights of individuals and groups. Based on the results, we report the human evaluation results in Section 3.2 and Section D.

D More Experimental Results

We also analyze the quality of generated knowledge and better understand its contribution to the final performance. The human evaluation results on the knowledge quality are summarized in Table 8 and the inter-annotator Krippendorff's α is 88.42, indicating an acceptable level of consistency ($> 80\%$) between annotators (Krippendorff, 2004). The few annotation conflicts are addressed after a discussion among the annotators. The table shows that SkillQG can generate implicit knowledge that makes sense and is pertinent to the context for around 85% of the time as evaluated by human annotators. Compared with other knowledge-augmented baselines that retrieve knowledge from ConceptNet, SkillQG generates knowledge that is similar in terms of common sense and has better relevance to the input context. The possible reason behind it is that SkillQG generates knowledge by asking and answering information-seeking questions based on the given context, benefiting the specialization of general knowledge of language models to each sample.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
6

☑ A2. Did you discuss any potential risks of your work?
6

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

3

☑ B1. Did you cite the creators of artifacts you used?
3

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Our employed scientific artifact is a manually annotated dataset that is publicly released by the authors of the corresponding paper. They do not provide a formal distribution license and terms for use. We use this dataset for research purposes only.

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
3

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Our employed scientific artifact is publicly released by the authors of the corresponding paper. It is constructed using the narrative texts and supervised by three experts in literacy education. All of the involved information in the scientific artifact are fictional and do not contain offensive content.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
3

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

3

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
B

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
B

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
B

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
B

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

3

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
C

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
C

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
C

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
C