Generating Deep Questions with Commonsense Reasoning Ability from the Text by Disentangled Adversarial Inference

Jianxing Yu, Shiqi Wang, Libin Zheng, Qinliang Su, Wei Liu, Baoquan Zhao, Jian Yin*
School of Artificial Intelligence, Sun Yat-sen University
Guangdong Key Laboratory of Big Data Analysis and Processing, China
Pazhou Lab, Guangzhou, 510330, China
Key Laboratory of Sustainable Tourism Smart Assessment Technology, Ministry of Culture and Tourism
{yujx26,wangshq25,zhenglb6,suqliang,liuw259,zhaobaoquan,issjyin}@mail.sysu.edu.cn

Abstract

This paper proposes a new task of commonsense question generation, which aims to yield deep-level and to-the-point questions from the text. Their answers need to reason over disjoint relevant contexts and external commonsense knowledge, such as encyclopedic facts and causality. The knowledge may not be explicitly mentioned in the text but is used by most humans for problem-shooting. Such complex reasoning with hidden contexts involves deep semantic understanding. Thus, this task has great application value, such as making high-quality quizzes in advanced exams. Due to the lack of modeling complexity, existing methods may produce shallow questions that can be answered by simple word matching. To address these challenges, we propose a new QG model by simultaneously considering asking contents, expressive ways, and answering complexity. We first retrieve text-related commonsense context. Then we disentangle the key factors that control questions in terms of reasoning content and verbalized way. Independence priors and constraints are imposed to facilitate disentanglement. We further develop a discriminator to promote the deep results by considering their answering complexity. Through adversarial inference, we learn the latent factors from data. By sampling the expressive factor from the data distributions, diverse questions can be yielded. Evaluations of two typical data sets show the effectiveness of our approach.

1 Introduction

Text-oriented question generation (QG) aims to endow machines with the ability to ask relevant and thought-provoking questions about the given text. This task can support a wide range of real-world applications, such as yielding quizzes from course materials for education (Qu et al., 2021), and generating questions as synthetic data to train a QA system (Wang et al., 2019). According to Bloom's taxonomy (Zhang et al., 2022), questions can be classified into different levels of cognitive ability. The simple ones involve only the shallow meaning of the text. For example, the question "*What is the longest river in the world?*" about the given text "*The Nile is the longest river in the world*" can be answered directly by matching. However, matching is far from a real understanding of the semantics (Ko et al., 2020). For example, in the field of education, simple questions are hard to fully evaluate students' learning effects, especially in advanced exams. Thus, the deep questions that require semantic understanding and reasoning have attracted extensive attention. As shown in Fig.(1), the question asks about some kind of clothing. The answer needs to be deduced from multiple relevant but disjoint clues in the contexts, i.e., "*traditional local costumes*," "*mountain*," "*Mount Fuji*," as well as implicit commonsense knowledge, such as *Mount Fuji is a famous mountain in Japan*, *Kimono is the traditional local costume of Japanese,* and *Japanese kimono is a kind of robe clothing.* Here, commonsense refers to the self-evident and unwritten knowledge shared by most humans, such as encyclopedism and causality. Although it does not appear in the text, it is hard to find the correct answer without it due to the incomplete context. Asking this kind of question requires a full understanding of commonsense and the ability to make

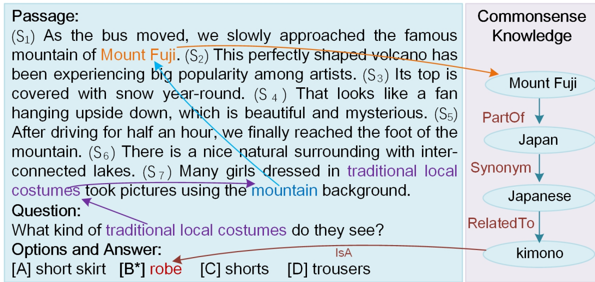


Figure 1: Sample deep question whose answer needs to be derived by complex commonsense reasoning skills.

*Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2023, pages 470–486
July 9-14, 2023 ©2023 Association for Computational Linguistics

inferences. That is a key ingredient for general intelligence. Some works have studied how to answer such questions, represented by commonsense QA and multi-hop QA (Rajani et al., 2019), but less effort explores how to generate them. We thus propose a new QG task to fill this research gap.

Raising deep questions involves three fundamental processes: *what to ask*, *how to ask*, and *how to answer*. *What to ask* is to identify the answer and its relevant reasoning contents. Learning *how to ask* focuses on the language qualities, such as grammatical correctness and expressive diversity, since the question could be asked in various ways and each way needs to be fluent. Respectively, *how to answer* reflects the question's complexity, shallow questions only need to match the text while deep reasoning ones require understanding the semantics in contexts with long-range dependencies and hidden commonsense knowledge. For these processes, traditional QG models have considerable defects. The rule-based method relies on handcrafted rules or transformation templates with a limited scale. That would restrict the coverage of results. Due to the neglect of indispensable answering feedback, the results are not guaranteed to be inferable and deep. On the other hand, the neural model mainly follows the *sequence-to-sequence* framework which is data-driven and labor-saving, but this monotonous mapping is hard to learn the one-to-many diversified generation. Besides, this method cannot cover the nuances of data by using a single vector to encode complex input features, especially when the training data is insufficient or has a long tail distribution. Spurious correlations and unexpected variances would easily mislead the single-factor model and deteriorate its robustness.

Motivated by the above observations, we propose a practical model for the new commonsense reasoning QG task. Concretely, we first leverage a knowledge-enhanced model to represent the text contexts, as well as relevant commonsense concepts and relations. We then learn the key factors related to the necessary ask contents and expressive ways. The first factor refers to the reasoning clues involved in asking deep questions, including entities and relations in the commonsense deductive context. Another encompasses other variations not covered by the content factor, like the verbalized styles and patterns. These factors can be sampled from the data manifold and used as conditions to generate results. This sampling-then-generate way alleviates the difficulty of collecting real data at the lower ends of a distribution tail in order to learn diversified generation. All these unknown factors may be mutually interrelated. Simply assuming that they are independent would oversimplify the latent manifold, leading to unsatisfied results due to the incorrect preservation of the redundant noises. We thus propose to disentangle such factors explicitly to ensure their independence and prevent information leakage between them. To achieve this goal, we introduce two kinds of latent variables to characterize the factors and impose constraints to learn their disentangled representations. These variables are forced to obey two prior non-overlapping distributions, including an isotropic *Gaussian* for the expressive way and another conditional *Gaussian* mixture for the reasoning content. Each component can be viewed as a cluster of neural templates or prototypes, which can be used as a guide to control the detailed nuances of a generation process. To encourage the deep and inferable questions, we impose regularization on the distributions by considering the answering complexity, including whether the answer matches the question and involves multi-hop reasoning with implicit commonsense knowledge. Moreover, we design an adversarial inference mechanism to derive optimal distributions for the disentangled factors. To facilitate deployment, we further employ the prefix-tuning technique (Li and Liang, 2021) that can support inference with limited labeled data. Our model enables one-to-many generation by randomly sampling the expressive factor from the distributions to yield new reasoning questions. Experimental results on two popular data sets show the effectiveness of our approach.

The main contributions of this paper include,

- We are the first to study the task of commonsense reasoning question generation from text.
- We propose a new model for the commonsense reasoning QG task. By a latent space with disentangled priors, our model can grasp the key factors that control the reasoning content and expressive way. Based on the factors as generative conditions, we can yield new diverse results by sampling data distributions.
- We design a discriminator and learn it by adversarial inference. It can provide complexity feedback as a guide to regularize the generator. Extensive experiments are conducted to evaluate our model quantitatively and qualitatively.

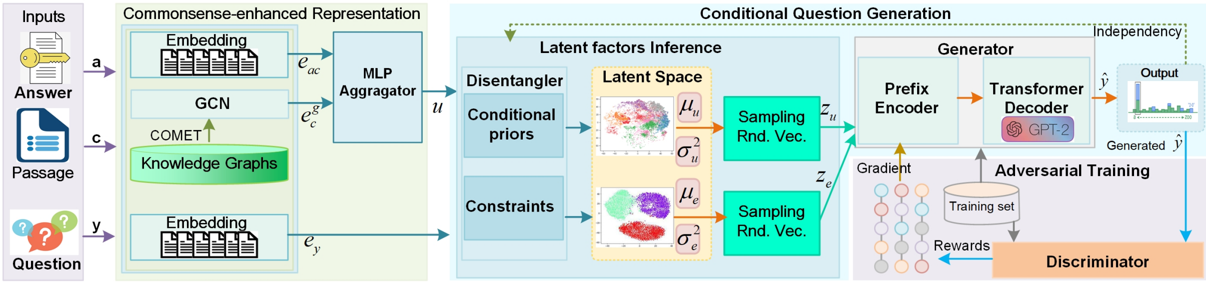


Figure 2: Overview of our approach for the task of generating deep question with commonsense reasoning ability.

The rest of this paper is organized as follows. Section 2 elaborates on the proposed method for the new commonsense QG task. Afterward, Section 3 presents experimental results. Section 4 reviews related works and Section 5 concludes the paper.

2 Approach

As shown in Fig.(2), we propose a new framework for this task. We first encode the text context and relevant commonsense knowledge. We then disentangle the key ask-related factors on reasoning content and expressive way. The answering feedback is also considered. By sampling the learned data manifold, we can derive factors as conditions to yield new questions. Next, we define some notations and then show the details of each component.

2.1 Notations and Problem Formulation

Given a passage c, the QG task aims to generate a valid question y corresponding to c and the answer a. The answering process involves deducing over a subset of disjoint supporting clues $P_y = \{p_1, \cdots, p_k\}$ from c, that is, $\{y \to p_1 \to \cdots \to p_k \to a\}$, where $\to$ represents entailment, p_j is a necessary problem-solving clue which can be a sentence or entity in c, k is the number of clues. When $k = 1$, we call y a traditional shallow question whose answer can be found by one-step matching of the given text. Respectively, when $k > 1$, y is a deep question with k reasoning steps. In complex reasoning, some clues are not in c, but from the background knowledge outside c. That is, a cannot be derived by using only c, and we have to answer y by introducing commonsense knowledge to supplement the missing contexts. Our task aims to yield this kind of question with a commonsense multi-step reasoning requirement. Compared to existing QG tasks, our question needs a deeper understanding of the semantics in c. Moreover, it is necessary to simultaneously figure out the ask-related contents, verbalized ways, and answering complexity. This task can be applied to many commercial scenarios like making quizzes for advanced exams. Since a question can be asked in many acceptable ways, where each y should be answered by a, we input a to indicate the asking direction.

2.2 Commonsense-enhanced Representation

Since asking deep questions involves understanding and reasoning the input text content, we need to derive a good semantic representation of the text. In particular, we first embed the context features in the input sample by looking up the pre-trained vectors in RoBERTa (Liu et al., 2019). The given text c and answer a are embedded as $\mathbf{e}_{ac} = ROBERTA([CLS], c, [SEP], a, [SEP])$, where $[CLS]$ and $[SEP]$ are special separator tokens. Similarly, question y is represented as $\mathbf{e}_y = ROBERTA(y)$. Afterward, we retrieve the commonsense features related to the given text. We resort to the knowledge graphs (*KG*) which contain plentiful human-shared knowledge. The first *KG* we consider is *ConceptNet* (Li et al., 2016). It contains millions of factual knowledge like encyclopedic concepts and parent-child relations. Another is *ATOMIC* (Sap et al., 2019) with plentiful procedural knowledge like *if-then* causal events. Such *KG*s can help to fill the implicit commonsense gap in the context. Since the *KG*s have different structures, we adopt the work of Ma et al. (2019b) to elicit the relevant *KG* contents. In particular, we identify *ConceptNet* entities appearing in the text by phrase-based matching, and then collect the relevant ν-hop triples. Accordingly, we utilize a transformer called *COMET* (Bosselut et al., 2019) which is pretrained on *ATOMIC* to generate the event triples based on the text and pre-defined relation types. Nine reasoning types in *COMET* are employed. Based on the extracted and generated contents, we can obtain a commonsense augmented graph. We then employ graph convolutional networks (*GCN*) (Kipf and Welling, 2017) to encode the graph as $\mathbf{e}_c^{kg} = GCN(V, E)$, where V, E denotes the set of nodes and edges, respectively. The nodes are the concepts, entities in the text and *KG*s, and the edges represent their relations. To integrate context and commonsense features, we apply an *MLP* network with *ReLU* activation to fuse the vectors as $\mathbf{u} = MLP([\mathbf{e}_{ac}; \mathbf{e}_c^{kg}])$, where $[\cdot; \cdot]$ is the concatenation operator.

2.3 Commonsense Reasoning QG Model

Traditional QG methods often learn an encoding vector of the input to decode the result. This single vector is insufficient to grasp the subtle structure of reasoning questions, and the one-to-one mapping is hard to capture diverse expressive ways. It is also difficult to find a suitable mapping for the rare cases at the distribution tail. We thus design a conditional generation framework that can disentangle multiple factors to finely model the reasoning contents and expressive patterns. The results can be easily inferred from a continuous data manifold, which has better generalization ability than learning the mapping of scattered points. That provides great flexibility to yield diverse results by adjusting the expression factors sampled from data distributions.

Conditional Generation: Our QG model yields the question based on the input of two latent variables. One is to characterize the reasoning contents related to *what to ask*, namely $\mathbf{z}_u$. Another is used to quantify the verbalized expressions of *how to ask*, i.e., $\mathbf{z}_e$. These variables can be learned from data by conducting approximate inference. Since the latent space allows invariance of distracting transformations, it is easier to discover elements of variations governing the data distribution. That helps to reason the data at an abstract level and find the key question-controlled factors. Our task can be formalized as an iterative word generative process based on a marginal distribution $p_\theta(\hat{y}|\mathbf{z}_e, \mathbf{z}_u)$, where θ is the model parameters. $\mathbf{z}_e$ can be sampled from a verbalized prior distribution, which helps to form the results expressed in various ways. To reduce the labeled data demand for training θ, we further employ the prefix-tuning technique that can freeze pre-trained vectors and learn only a few prompt parameters. The continuous prompt is designed as $\mathbf{M}_\theta[i,:] = MLP_\theta([\mathbf{M}'_\theta[i,:]; \mathbf{z}_e; \mathbf{z}_u])$, where $\mathbf{M}'_\theta$ is a learnable matrix, $MLP(\cdot)$ is a multilayer network. Based on this prompt, we can produce the question word-by-word by Eq.(1), where $\hat{y}_{<t}$ represents the outputted 1^{th} to $(t-1)^{th}$ words.

$$p_\theta(\hat{y}|\mathbf{z}_e, \mathbf{z}_u) = \prod_{t=1}^{J} p_\theta(\hat{y}_t|\hat{y}_{<t}, \mathbf{M}_\theta[i,:]) \quad (1)$$

To well capture abundant expressive patterns in the questions, we let $\mathbf{z}_e$ obey the prior distribution p_ψ of factorized *Gaussian* $\mathcal{N}(\mathbf{z}_e; \boldsymbol{\mu}_e^y, \lambda_e \mathbf{I})$, where $\boldsymbol{\mu}_e^y$ is the mean, and λ_e is the variance. Different from a standard normal distribution $\mathcal{N}(0, \mathbf{I})$, this allows us to associate its mean with the linguistic features $\Phi(y)$ from the question y by $\boldsymbol{\mu}_e^y = \mathbf{W}_y \Phi(y)$, where $\mathbf{W}_y$ is a project matrix and $\Phi(y)$ is the mean of question encodings. Considering the given text may contain multiple inquiry topics, the content latent $\mathbf{z}_u$ is expected to be composed of K-independent components. Thus, we make $\mathbf{z}_u$ follow *Gaussian* mixture distributions, i.e., $\sum_{k=1}^{K} p_\psi(M_k|\mathbf{u}) \mathcal{N}(\mathbf{z}_u; \boldsymbol{\mu}_{u_k}^y, \lambda_u \mathbf{I})$, where M_k is a random variable to indicate the k^{th} component.

Disentangled Inference: To better learn the latent representation z, we introduce a series of constraints. First, the latent vector should be able to fully characterize the corresponding content. That can be quantified by maximizing *mutual information* (*MI*) (Cheng et al., 2020) of $MI(\mathbf{z}_e, y)$ and $MI(\mathbf{z}_u, \mathbf{u})$, where $\mathbf{u}$ is the commonsense-enhanced representation of the inputs c and a. To improve the model's robustness, we impose disentangled constraints. The content vector is encouraged to encode disjoint information with the expression vector and vice versa. That can reduce redundancy and provide refined control over results. We seek to explicitly minimize the shared information of vectors by adding a divergence-based regularization of *Maximum Mean Discrepancy* (*MMD*) (Gretton et al., 2012), as $MMD(p(\mathbf{z}_e|y), p(\mathbf{z}_u|\mathbf{u}))$. By aggregating the constraints, our generator's objective of Eq.(1) can be reformulated as Eq.(2).

$$\begin{aligned} &\max \int p_\theta(\hat{y}|\mathbf{z}_e, \mathbf{z}_u) p_\psi(\mathbf{z}_e|y) p_\psi(\mathbf{z}_u|a, c) d\mathbf{z}_e d\mathbf{z}_u \\ &= \max \sum_{i=1}^{n} [\log p(\hat{y}|y_i, a_i, c_i) + MI(\mathbf{z}_{y_i}, y_i) \\ &+ MI(\mathbf{z}_{u_i}, \mathbf{u}_i) - MMD(p(\mathbf{z}_{y_i}|y_i), p(\mathbf{z}_{u_i}|\mathbf{u}_i))] \end{aligned} \quad (2)$$

We then utilize the variational inference technique to solve it since direct optimization is intractable. A variational posterior $q_\phi(\cdot)$ is introduced to approximate the prior distribution $p_\psi(\cdot)$. By maximizing the *evidence lower bound* (*ELBO*) of Eq.(2), we can derive an equivalent objective as Eq.(3).

$$\begin{aligned} \max \mathbb{E}_{q_\phi(\mathbf{z}_e, \mathbf{z}_u|y, \mathbf{u})} &[\log p_\psi(\hat{y}, \mathbf{z}_e, \mathbf{z}_u|y, \mathbf{u}) \\ &- \log q_\phi(\mathbf{z}_e, \mathbf{z}_u|y, \mathbf{u})] \end{aligned} \quad (3)$$

This *ELBO* can be decomposed into Eq.(4) by minimizing the reconstruction loss $\mathcal{L}_r$ of y given the inputs c and a (encoded as $\mathbf{u}$), and regularizing the approximate posterior $q_\phi(\cdot)$ to be close to the prior $p_\psi(\cdot)$ by *KL divergence*, where $\mathcal{L}_e$ and $\mathcal{L}_u$ are the divergence losses for latent $\mathbf{z}_e$ and $\mathbf{z}_u$, respectively.

$$
\begin{aligned}
&\mathcal{L}_{generator}(\psi, \phi, y, c, a) = \mathcal{L}_r + \mathcal{L}_e + \mathcal{L}_u \\
&\mathcal{L}_r = \mathbb{E}_{q_\phi}(\mathbf{z}_e, \mathbf{z}_u | y, \mathbf{u})[\log p_\psi(\hat{y}|\mathbf{z}_e, \mathbf{z}_u)] \\
&\mathcal{L}_e = \mathbb{D}_{KL}(q_\phi(\mathbf{z}_e|\hat{y}, y)||p_\psi(\mathbf{z}_e|y)) \\
&\mathcal{L}_u = \mathbb{D}_{KL}(q_\phi(\mathbf{z}_u|\hat{y}, u)||p_\psi(\mathbf{z}_u|\mathbf{u}))
\end{aligned}
\tag{4}
$$

$\mathcal{L}_e$ is the loss related to the expression factor. Similar to the prior $p_\psi(\cdot)$, the posterior $q_\phi(\cdot)$ is followed the factorized *Gaussian*, as $\mathcal{N}(\mathbf{z}_e; \boldsymbol{\mu}_e^y, diag(\boldsymbol{\sigma}_{ye}^2))$. By applying the reparameterization trick (Kingma and Welling, 2014), we can calculate the latent $\mathbf{z}_e$ as $\mu_e + \boldsymbol{\sigma}_e \odot \boldsymbol{\epsilon}_e$, where $\boldsymbol{\epsilon}_e$ is the *Gaussian* factor drawn from $\mathcal{N}(0, \mathbf{I})$, $\odot$ is the element-wise product. Based on $\mathbf{z}_e$, $\mathcal{L}_e$ can be calculated as Eq.(5).

$$
\mathcal{L}_e = -\frac{1}{\lambda_e}||\mathbf{z}_e - \boldsymbol{\mu}_e^y||^2 + \log \boldsymbol{\sigma}_{ye}^2 \tag{5}
$$

Another loss $\mathcal{L}_u$ is relevant to the reasoning contents in passage c and answer a. Considering the contents may contain multiple inquiry topics, we characterize the posterior q_ϕ by *Gaussian* mixture distributions, and introduce K latent topic prototypes $\{\mathbf{t}_k\}_{k=1}^K$. Each *Gaussian* component is promoted to be close to the prototype variational distribution. That can be achieved by making the component be $\mathcal{N}(\mathbf{z}_u; \boldsymbol{\mu}_{u_k}^y, diag(\boldsymbol{\sigma}_u^2))$. The K is preset, when the value is small, the content modeling is simple and coarse-grained. The reasoning aspects involved in the generated results will be less. When the K value is large, the convergence speed becomes slower. By tuning, we set K to 10 in the experiment. To encourage its mean corresponding to one kind of topic, we compute $\boldsymbol{\mu}_{u_k}^y$ as $\mathbf{W}_t\mathbf{t}_k$, where $\mathbf{t}_k$ is the centroid of a cluster k. Each cluster can be computed by the k-means method. The probability of the input content belonging to the k prototype is parameterized as $q_\phi(M_k|\mathbf{u}) = \frac{\exp(-dist(\mathbf{z}_u, \boldsymbol{\mu}_{u_k}^y)/\tau)}{\sum_{k'}\exp(-dist(\mathbf{z}_u, \boldsymbol{\mu}_{u_{k'}}^y)/\tau)}$, where τ is a temperature set to 1 normally, $dist(\cdot)$ is a *Euclidean* distance between the mean and the latent $\mathbf{z}_u$. In this way, we compute the loss $\mathcal{L}_u$ as Eq.(6)

$$
\mathcal{L}_u = \sum_{k=1}^{K} q_\phi(M_k|\mathbf{u})[-\frac{1}{2\lambda_u}||\mathbf{z}_u - \boldsymbol{\mu}_{u_k}^p||^2] + \log \boldsymbol{\sigma}_{\boldsymbol{u}}^2 \tag{6}
$$

Adversarial Training: Unlike shallow question, complex one has an inherent reasoning structure. Based on traditional supervised training, the model is only required to have maximum likelihood with the ground truth, but neglects to grasp this crucial structure. It may learn some trivial tricks to simply copy similar terms, leading to shallow results. Thus, it is necessary to inject the answering feedback into the generator for judging the rationality of results. Instead of using a discrete judged metric, we design a differentiable discriminator that can guide the generator optimization via policy gradient. It is trained to distinguish between real data examples and synthetic ones produced by the generator. The generator is then optimized for fooling the discriminator. By their adversarial game, the distribution of the generated examples moves towards the distribution of real data. That directs the generator to learn complex distributions and produce reasonable realistic questions. In particular, we use a QA model called *UNICORN* (Lourie et al., 2021) to capture the answerable feedback. It obtains state-of-the-art performance on solving commonsense reasoning questions. For each sample $(c, a, \hat{y})$, we compute $d_{ans} = \sigma_1(\mathbf{W}_1[\mathbf{e}_{\hat{a}}; \mathbf{e}_a])$, where $\mathbf{W}$ is the weight, $\sigma(\cdot)$ is the logistic function, $\mathbf{e}_{\hat{a}}$ is the answer predicted by $UNICORN(c, \hat{y})$, $\mathbf{e}_a$ is an answer encoding. To ensure that the question is inferable, we thus leverage a typical matching-based QA model called gated-attention reader(GA) (Dhingra et al., 2017). We then compare its answer against the reasoning model *UNICORN*. When these two answers match, there is no need for reasoning. It is highly likely to be a simple but not deep question. We introduce a metric $d_{cpx} = \sigma_2(\mathbf{W}_2[\mathbf{e}_{\hat{a}_1}; \mathbf{e}_{\hat{a}_2}])$, where $\mathbf{e}_{\hat{a}_1}$ and $\mathbf{e}_{\hat{a}_2}$ are the answers predicted by $UNICORN(c, \hat{y})$ and $GA(c, \hat{y})$, respectively.

The discriminator is developed by integrating these aspects. For each sample $x = (c, a, y)$, we can predict a reward as $d_\delta(x) = \gamma d_{ans}(x) + (1 - \gamma)d_{cpx}(x)$, where λ is a trade-off factor. This reward can be used as guidance to co-train the generator by reinforcement learning. The discriminator can be trained based on the supervised loss of human-written data. Considering such labeled data may not be sufficient, we use the model-generated samples as extra data to augment the training.

In the prediction phase, the input is a passage and an answer. Each test case can generate multiple questions with three steps. We first encode the input passage and answer, and then derive a latent content factor $\mathbf{z}_u$ based on $p_\psi(\mathbf{z}_u|\mathbf{u})$. Accordingly, we sample another verbalized factor $\mathbf{z}_e$ from the prior p_ψ. Afterward, we feed them into the prefix encoder and decode question $\hat{y}$ by $p_\theta(\cdot)$ in Eq.(1).

3 Evaluations

We extensively evaluated the effectiveness of our method with quantitative and qualitative analysis.

3.1 Data and Experimental Settings

Since QG is a complementary task of QA, we conducted experiments on two typical QA data sets that involved commonsense reasoning, including *Cosmos QA* (Huang et al., 2019) and *MCScript 2.0* (Ostermann et al., 2018). These data sets were split as train/dev/test sets with the size of 25.6k/3k/7k and 14.2k/2.0k/3.6k samples, respectively. The samples mostly required context understanding and commonsense reasoning. They were more suitable than other data sets like *CommonsenseQA* (Talmor et al., 2019) which provided no text context, *SQuAD* (Rajpurkar et al., 2016) did not need multi-hop deduction, and *LogiQA* (Liu et al., 2020) with the general questions such as "*Which one is true?*" that can be yielded by rules. For each test case, our inputs included a passage and an answer to guide the asking direction. We employed three standard metrics in the field of text generation to evaluate the generative quality based on n-gram overlap with the ground truth, including *BLEU-4* (Papineni et al., 2002), *METEOR* (Banerjee and Lavie, 2005), and *ROUGE-L* (Lin, 2004). In addition, we observed that the question involves fine-grained reasoning logic on the answering process. Even if a similar word is substituted, the questions may mismatch the answers, or become too shallow to be inferable. Thus, we utilized two distribution overlap metrics, i.e., QA-based Evaluation (*QAE*) (Zhang and Bansal, 2019b), and Reverse QAE (*R-QAE*) (Lee et al., 2020a) to measure diversity instead of using traditional similarity-based metrics. To compute *QAE*, we first trained a QA model on the generated data and then tested it on ground-truth data. The score is high when these two distributions match, which indicates the generated quality reaches human annotations. *R-QAE* was calculated by swapping the train and test data. Its value is lower when the generated data is more diverse than the ground truth. That is more suitable to evaluate our task by considering the answering process. Besides, the commonsense reasoning ability was evaluated by human evaluation. To avoid biases, we randomly sampled 500 test cases and rated the predictions by a crowdsourcing platform *Figure-Eight* [1] with five participants. It was a rating in terms of three metrics, including valid *syntax*, *relevance* to input text, and commonsense *deductibility* of the answer. We averaged the cumulative scores of judgments as performance. The scores are between $1 \sim 10$, where 1 is the worst, 10 is the best. For the methods with multiple diverse results, we computed metrics for each prediction and reported the average scores.

Our model was implemented based on the PyTorch (Paszke et al., 2019) and ran on the 24 GB Nvidia RTX 3090 GPU for 18 hours. We leveraged the RoBERTa-large (355M parameters) model provided by HuggingFace library to initialize the word embeddings. We employed the transformer-based *GPT-2 medium* as the decoder. In the diversity evaluation, the metrics (i.e., *QAE* and R-QAE) were computed based on the *UNICORN* QA model. We trained for a maximum of $10,000$ steps and validated every 200 steps, with early stopping after one round of no improvement in validation loss. *AdamW* (Loshchilov and Hutter, 2019) was used as the optimizer, with a linear learning rate scheduler taking 5,000 warm-up steps. Gradients were clipped if their norm exceeds 1.0, and weight decay on all non-bias parameters was set to 0.01. In the prediction phase, the outputted candidate size was set to 3. The trade-off factor γ was tuned to 0.3.

3.2 Comparisons against State of the Arts

To evaluate the model persuasively, we utilized six baselines that performed well in the QG task, including (a) *NQG++* (Zhou et al., 2017), a basic *sequence-to-sequence* model; (b) *UniLM* (Dong et al., 2019), a pre-trained language model that can fine-tune on *KG*s to incorporate commonsense context; (c) *SGGDQ* (Pan et al., 2020), a graph-based model which can produce results with multi-hop deduction ability by capturing the context dependency of the text; (d) *HCVAE* (Lee et al., 2020b), a VAE-based model that can yield results in several ways for one test case. (e) *DAANet* (Xiao et al., 2018), dual learning of QG and QA that mutually provided feedback to enhance each other simultaneously; (e) *SemQG* (Zhang and Bansal, 2019b), which trained QG by reinforcement learning with a QA-based reward. These baselines were open-source and we reimplemented them with the original settings.

Fig.(3) showed the comparison results in terms of three n-gram overlap metrics. Our model held the best performance against other baselines. As illustrated in Tab.(1), our model obtained high *QAE* but low *R-QAE*. That reflected the synthetic data

[1] https://appen.com/figure-eight-is-now-appen/

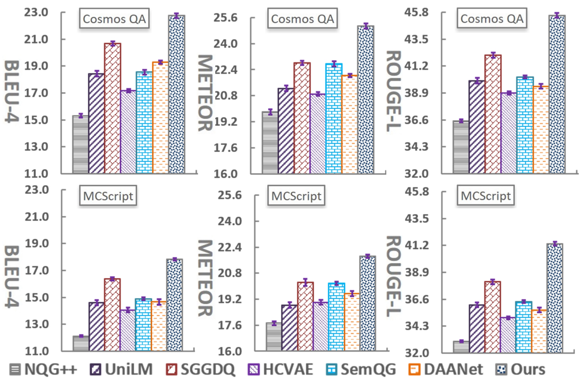


Figure 3: Comparisons of methods in terms of n-gram overlap metrics with corresponding variances.

Table 1: Comparisons of evaluated methods in terms of distribution overlap metrics with related variances.

Datasets	Cosmos QA		MCScript	
Method	QAE(↑)	R-QAE(↓)	QAE(↑)	R-QAE(↓)
NQG++	78.5 ± 0.2%	88.1 ± 0.2%	77.4 ± 0.3%	89.2 ± 0.4%
UniLM	80.2 ± 0.3%	85.3 ± 0.1%	79.2 ± 0.3%	86.3 ± 0.6%
SGGDQ	81.3 ± 0.2%	84.2 ± 0.5%	80.2 ± 0.4%	83.2 ± 0.4%
HCVAE	83.6 ± 0.3%	82.6 ± 0.6%	81.8 ± 0.5%	81.7 ± 0.3%
SemQG	82.4 ± 0.3%	80.3 ± 0.7%	81.2 ± 0.3%	79.6 ± 0.4%
DAANet	84.1 ± 0.2%	81.4 ± 0.3%	81.5 ± 0.4%	80.1 ± 0.2%
Ours	88.9 ± 0.2%	77.3 ± 0.4%	83.4 ± 0.3%	75.6 ± 0.3%

were closer to human annotations. As shown in Lee et al. (2020a), lower *R-QAE* means resultant data covers larger distributions. Although trivially invalid questions may also cause low *R-QAE*, a combination of high *QAE* and low *R-QAE* can indicate the diversity of our results. By a single encoded vector, *NQG++* was difficult to cover the nuances of data. *UniLM* could encode commonsense but its reasoning ability was insufficient. The graph model *SGGDQ* was good at multi-hop samples, but its monotonous mapping framework is difficult to support one-to-many generation. Due to the lack of disentanglement, VAE-based model *HCVAE* would be affected by unexpected irreverent noises which will harm performance. All baselines neglected to consider the feedback of answering complexity. Without this crucial guidance, the performance would be deteriorated. *DAANet* and *SemQG* used the QA feedback, but the dual soft constraint of *DAANet* and the high variance of the reinforced *SemQG* were hard to ensure results' consistency.

Moreover, we evaluated our model's applicability in low-resource scenarios. We started to train it with the full training data and gradually halved the size. The results on 1/2 and 1/8 data size were presented in Tab.(2) and Tab.(3), respectively. We found that our performance decline was smallest when training sets shrunk. That reflected our model had a good generalization ability to achieve greater outperformance by disentangling key question-controlled factors.

Table 2: Performance change ratios on 1/2 data size.

CosmosQA	BLUE4	METEOR	ROUGE	QAE	R-QAE
NQG++	↓ 16.0%	↓ 16.7%	↓ 17.9%	↓ 9.7%	↑ 9.4%
UniLM	↓ 15.0%	↓ 15.3%	↓ 14.4%	↓ 9.2%	↑ 9.0%
SGGDQ	↓ 18.8%	↓ 17.6%	↓ 16.7%	↓ 8.0%	↑ 7.8%
HCVAE	↓ 12.0%	↓ 13.4%	↓ 12.4%	↓ 9.0%	↑ 8.5%
SemQG	↓ 12.5%	↓ 14.8%	↓ 15.5%	↓ 8.6%	↑ 8.0%
DAANet	↓ 13.6%	↓ 14.1%	↓ 13.9%	↓ 7.8%	↑ 7.6%
Ours	↓ 9.0%	↓ 8.7%	↓ 7.1%	↓ 3.3%	↑ 3.1%
MCScript	BLUE4	METEOR	ROUGE	QAE	R-QAE
NQG++	↓ 21.8%	↓ 22.0%	↓ 23.2%	↓ 10.2%	↑ 9.9%
UniLM	↓ 18.0%	↓ 19.3%	↓ 18.4%	↓ 9.3%	↑ 9.0%
SGGDQ	↓ 22.4%	↓ 20.9%	↓ 21.8%	↓ 8.4%	↑ 7.7%
HCVAE	↓ 14.8%	↓ 16.5%	↓ 17.2%	↓ 9.2%	↑ 8.9%
SemQG	↓ 15.0%	↓ 16.3%	↓ 17.3%	↓ 8.8%	↑ 8.1%
DAANet	↓ 17.2%	↓ 19.7%	↓ 17.6%	↓ 8.0%	↑ 7.2%
Ours	↓ 7.9%	↓ 7.5%	↓ 8.1%	↓ 3.9%	↑ 3.4%

Table 3: Performance change ratios on 1/8 data size.

CosmosQA	BLUE4	METEOR	ROUGE	QAE	R-QAE
NQG++	↓ 50.5%	↓ 51.1%	↓ 52.1%	↓ 18.3%	↑ 13.8%
UniLM	↓ 45.2%	↓ 45.8%	↓ 46.3%	↓ 14.2%	↑ 11.4%
SGGDQ	↓ 44.3%	↓ 44.8%	↓ 45.7%	↓ 13.8%	↑ 11.0%
HCVAE	↓ 41.7%	↓ 40.2%	↓ 42.8%	↓ 12.1%	↑ 10.7%
SemQG	↓ 46.2%	↓ 45.3%	↓ 47.2%	↓ 12.6%	↑ 10.3%
DAANet	↓ 43.6%	↓ 43.7%	↓ 45.6%	↓ 11.5%	↑ 9.4%
Ours	↓ 30.2%	↓ 29.5%	↓ 31.4%	↓ 7.8%	↑ 6.8%
MCScript	BLUE4	METEOR	ROUGE	QAE	R-QAE
NQG++	↓ 57.1%	↓ 58.4%	↓ 60.1%	↓ 17.8%	↑ 13.0%
UniLM	↓ 55.2%	↓ 56.2%	↓ 57.2%	↓ 15.7%	↑ 12.2%
SGGDQ	↓ 56.3%	↓ 55.5%	↓ 55.8%	↓ 14.3%	↑ 11.6%
HCVAE	↓ 49.6%	↓ 50.2%	↓ 52.3%	↓ 14.0%	↑ 11.5%
SemQG	↓ 51.2%	↓ 50.8%	↓ 51.7%	↓ 13.5%	↑ 10.8%
DAANet	↓ 53.6%	↓ 54.0%	↓ 53.2%	↓ 14.8%	↑ 11.2%
Ours	↓ 38.4%	↓ 37.2%	↓ 39.6%	↓ 8.1%	↑ 7.5%

3.3 Ablation Studies

To better gain insight into the relative contributions of our QG's components, we performed ablation studies on four parts, including (1) *Ours-LM* which replaced the commonsense-enhanced model with the raw *PLM*; (2) *Ours-Disentangler* that discarded the independence constraints with disentangled priors; (3) *Ours-Prefix* threw away the prefix tuning then trained the model on the full parameters; (4) *Ours-Discriminator* that abandoned the discriminator and learned with typical supervised loss.

As shown in Tab.(4), the ablation of all evaluated parts led to a performance drop, where some drops were more than 10%. We could infer that commonsense knowledge can help to supplement missing contexts implied in the text. Without this guidance, the results' rationality will be harmed. When the prefix tuning module was discarded, the training

Table 4: Ablation studies, performance change ratios.

CosmosQA	BLUE4	METEOR	ROUGE	QAE	R-QAE
-LM	↓ 4.6%	↓ 5.2%	↓ 5.1%	↓ 4.4%	↑ 3.6%
-Disentanger	↓ 14.0%	↓ 15.2%	↓ 16.3%	↓ 5.8%	↑ 4.5%
-Prefix	↓ 6.1%	↓ 6.4%	↓ 6.6%	↓ 4.8%	↑ 3.7%
-Discriminator	↓ 11.3%	↓ 9.5%	↓ 10.7%	↓ 5.6%	↑ 4.0%
MCScript	BLUE4	METEOR	ROUGE	QAE	R-QAE
-LM	↓ 5.8%	↓ 5.3%	↓ 5.7%	↓ 4.8%	↑ 4.0%
-Disentanger	↓ 18.2%	↓ 16.1%	↓ 17.5%	↓ 5.9%	↑ 4.8%
-Prefix	↓ 7.4%	↓ 8.1%	↓ 8.6%	↓ 5.0%	↑ 4.1%
-Discriminator	↓ 12.5%	↓ 13.0%	↓ 13.8%	↓ 5.7%	↑ 4.5%

adequacy would be reduced with limited labeled data. Deleting a disentangled module would reduce the model's robustness and controllability. Without the discriminator, there was inadequate to indicate that the results were deep and logically consistent.

3.4 Human Evaluations and Analysis

Furthermore, we conducted human evaluations to judge whether the results were deep and had high-level answering skills like commonsense reasoning. We employed Randolph's kappa for inter-rater reliability measurement. The kappa κ scores were 0.77, 0.65, and 0.75 for *syntax*, *relevance*, and *deductibility*, respectively, which indicated a good agreement. As presented in Fig.(4), our model significantly outperformed the baselines in terms of three metrics. That was consistent with the quantitative results in the previous section. The improvement in the *deductibility* metric was the largest. That indicated our results were to-the-point and valid, especially inferable, due to the simultaneous consideration of *what to ask*, *how to ask*, and *how to answer*.

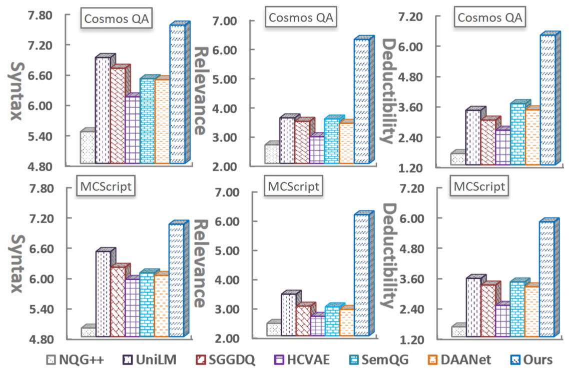


Figure 4: Human Analysis. κ agreement > 0.65

3.5 Evaluations on the Trade-off Parameter

To examine the trade-off parameter (i.e., γ) in the discriminator d_δ, we tuned it from $[0, 1]$ with 0.1 as an interval. The performance change curve was plotted in Fig.(5). The best results were obtained at around 0.3. The performance dropped dramatically when any parameter was close to 0 or 1. We could infer that all loss metrics were helpful, thereby training our model efficiently.

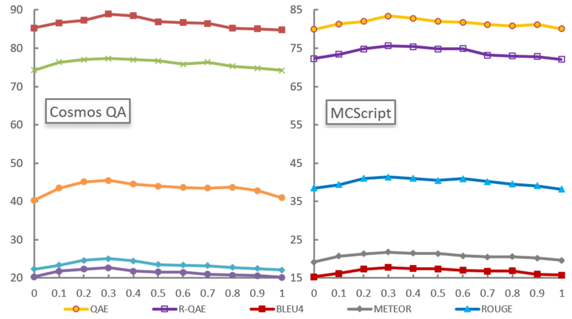


Figure 5: Evaluations on the trade-off parameters.

3.6 Case Studies and Discussions

We next conducted case studies to analyze the results of each method qualitatively. As exhibited in Fig.(6), our model could produce multiple commonsense questions. Contrastively, the sequential *NQG++* yielded a shallow question that can be answered by directly matching the input text. The pre-trained *UniLM* showed a bit of fluency and graph-based *SGGDQ* reflected a certain amount of reasoning. Their results were monotonous and cannot yield results in other acceptable expressive ways. *HCVAE* could produce diverse results which could not match the answers. The reinforced *SemQG* and dual model *DAANet* were answer-related, but their results' deductibility was weak. These results further validated the effectiveness of our model. By analyzing our bad cases, the mistakes mainly came from temporal errors, e.g. "*do*" should be "*did*" at "*Which country do Bob visit yesterday?*" and special symbols errors, e.g. missing "*'s*." These challenges would be studied in future work.

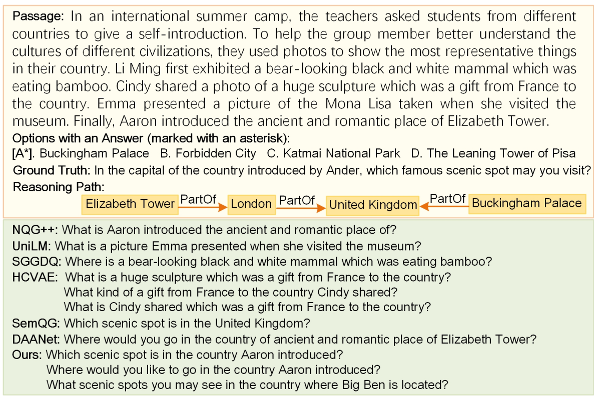


Figure 6: Case study on our commonsense QG model.

4 Related Works

Question Generation (QG) is a hot research topic that can support many valuable applications, including synthesizing training data for the question-answering (QA) task (Duan et al., 2017), producing exercises on the textbook (Chen et al., 2018), and clarifying users' needs for a dialog agent (Aliannejadi et al., 2019). Previous studies mainly focus on shallow questions (Wang et al., 2020a). They can be tackled by matching the text without demanding a real understanding of semantics (Yu et al., 2023). The researchers gradually pay attention to deep questions (Hua et al., 2020), such as multi-hop QG (Yu et al., 2020). However, these questions only involve the context that appears in the text without the need of understanding the commonsense knowledge. Asking questions with this background knowledge is indispensable for machine intelligence, but has been less explored. Thus, we propose a new QG task to fill this research gap.

Most of the earlier methods in the QG task were rule-based (Dhole and Manning, 2020). The handcrafted rules were labor-intensive with poor scalability (Zhang et al., 2022). To reduce labor costs, recent attempts turned to a data-driven neural model with better language flexibility (Dou and Peng, 2022). They learned direct mappings from input texts to questions by an encoder-decoder framework (Du et al., 2017). Considering the question would be asked in diverse ways (Shu et al., 2020), it was hard to support one-to-many generation based on a fixed encoded vector (Lachaux et al., 2020). Some studies proposed to enhance the generalization ability (Wang et al., 2021) by variational autoencoder (*VAE*) (Li et al., 2022a). It can learn an ask-related vector (Li et al., 2022b) which can be resampled to produce multiple questions (Wang et al., 2022) based on data distribution. However, one single vector was not sufficient to capture the complex and entangled asking features (Wang et al., 2020b). In contrast, we consider multiple factors and disentangle them to control the generation finely.

Deep questions require reasoning the knowledge both inside and outside the text (Zhang et al., 2021), including hidden commonsense context (Lv et al., 2020). To capture this context, we can resort to the knowledge graphs (*KG*) (Zhao et al., 2020) or pre-training models (Chen et al., 2020), such as BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), and GPT-3 (Brown et al., 2020). The *KG* knowledge can be collected by matching (Ye et al., 2022), and the pre-training one is often obtained by prompt learning (Gao et al., 2021). In addition, the depth of questions is mainly reflected in *how to answer* (Hu et al., 2017). There are often two ways to incorporate the answering feedback (Liu et al., 2022). One is reinforcement learning which views the answer as a reward (Bao et al., 2018). Since there is no prior guidance, the robustness of this method is weak (Bao et al., 2018). Another way is to use generative adversarial learning (*GAN*) to jointly train the QA and QG tasks (Sun et al., 2020). This method only judges the final answer but neglects to grasp the answering process, leading to the results' lack of commonsense reasoning ability (Wu et al., 2022). Also, this discrete judge is non-differentiable (Jin et al., 2020), causing unstable training (Ma et al., 2019a). In contrast, our discriminator simultaneously consider the matched answer and its reasoning complexity, which can facilitate the training of deep question generator.

5 Conclusions

We have proposed a new commonsense reasoning QG task which aimed to generate valid and inferable questions about the given text. Unlike traditional QG tasks, our questions needed to deduce multiple clues in disjoint contexts, where not all clues were provided in the given text, and some required to resort to commonsense knowledge outside the text. Since understanding semantics is the prerequisite to asking high-quality questions, our complex QG task requires a higher level of machine intelligence. Due to the lack of modeling complexity, traditional methods often yield shallow results. To address the problem, we proposed a practical framework that can flexibly incorporate the asking contents, expressive ways, and answering complexity to yield deep results by disentangling adversarial inference. We first retrieved the commonsense knowledge related to the given text. We then disentangled the key question-controlled factors in terms of reasoning content and verbalized way based on the independency priors and constraints. To promote deep results, we further designed a discriminator to regularize the generator by providing the answering feedback. By adversarial inference, we can derive the factors and use them as conditions to decode questions. By sampling the expressive factor from the data distribution, diverse results can be produced. Experimental results on two typical data sets showed the effectiveness of our approach.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62276279,U1911203, 62102463, 62276280, U2001211, U22B2060), Guangdong Basic and Applied Basic Research Foundation (2019B1515130001), Tencent WeChat Rhino-Bird Focused Research Program (WXG-FR-2023-06), Natural Science Foundation of Guangdong Province of China (2022A1515011135), Key-Area Research and Development Program of Guangdong Province (2020B0101100001), Zhuhai Industry-University-Research Cooperation Project (2220004002549, ZH22017001210010PWC).

Limitations

Deep questions not only require an in-depth understanding of the semantics in the text, but also involve the formulation of questions with correct grammar, such as tense transformation, and special symbols adjustment. For this task, our model simultaneously capture the key factors on the reasoning content, expressive way, and answering complexity, aiming to make results valid, relevant and inferable. However, as mentioned in the case study section, our model has some bad cases with grammatical flaws. For example, "do" needs to be transformed to "did" when the given text is in the past tense. This requires linguistic knowledge on top of words. Learning to ask with the guidance of this abstract knowledge is not covered in this paper. One way to tackle this problem is to resort to post-processing with a grammar error corrector. In addition, the interpretability of latent variables and the robustness of the model are not explored in this paper. We will investigate them in future works.

Ethics Statement

The technology proposed in this paper can be used in many applications, such as in the fields of education, Q&A, and dialogue systems. For example, it can yield quizzes for exams, or provide reasonable clarification question to warm up the conversation. Unlike shallow matching-based questions, our deep questions require fully understanding the semantics inside and outside the text. That involves many high-level cognitive skills, including reasoning the incomplete contexts with hidden commonsense knowledge. That can better support the real applications such as advanced exams in TOEFL and SAT, since there are few or even no simple questions. When excluding the misusage scenarios, there are usually no ethical issues with this technology. However, the questions can be generated as long as we input the text. It is possible to input some inappropriate content related to the topics of racial discrimination, war, and so on, resulting in some offensive questions. This problem can be addressed by limiting the topics of input contents.

References

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM Conference on Research and Development in Information Retrieval, SIGIR*, pages 475–484, Paris, France.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan.

Junwei Bao, Yeyun Gong, Nan Duan, Ming Zhou, and Tiejun Zhao. 2018. Question generation with doubly adversarial nets. *IEEE ACM Transactions on Audio, Speech and Language Processing, TASLP*, 26(11):2230–2239.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS*.

Guanliang Chen, Jie Yang, Claudia Hauff, and Geert-Jan Houben. 2018. Learningq: A large-scale dataset for educational question generation. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM*, pages 481–490, Stanford, California, USA.

Zhiyu Chen, Harini Eavani, Wenhu Chen, Yinyin Liu, and William Yang Wang. 2020. Few-shot NLG with pre-trained language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 183–190.

Pengyu Cheng, Martin Renqiang Min, Dinghan Shen, Christopher Malon, Yizhe Zhang, Yitong Li, and Lawrence Carin. 2020. Improving disentangled text representation learning with information-theoretic guidance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 7530–7541.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1832–1846.

Kaustubh D. Dhole and Christopher D. Manning. 2020. Syn-qg: Syntactic and shallow semantic rules for question generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 752–765.

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 *Proceedings of the Advances in Neural Information Processing Systems 32, NeurIPS*, pages 13042–13054, Vancouver, BC, Canada.

Zi-Yi Dou and Nanyun Peng. 2022. Zero-shot commonsense question answering with cloze translation and consistency optimization. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI*, pages 10572–10580.

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, ACL*, pages 1342–1352, Vancouver, Canada.

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, EMNLP*, pages 866–874, Copenhagen, Denmark.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP*, pages 3816–3830.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. 2012. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017. Toward controlled generation of text. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, pages 1587–1596, Sydney, NSW, Australia.

Yuncheng Hua, Yuan-Fang Li, Gholamreza Haffari, Guilin Qi, and Tongtong Wu. 2020. Few-shot complex knowledge base question answering via meta reinforcement learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 5827–5837.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: machine reading comprehension with contextual commonsense reasoning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP-IJCNLP*, pages 2391–2401, Hong Kong, China.

Shuning Jin, Sam Wiseman, Karl Stratos, and Karen Livescu. 2020. Discrete latent variable representations for low-resource text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 4831–4842.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR*, Banff, AB, Canada.

T.N. Kipf and M. Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of International Conference on Learning Representations, (ICLR)*, pages 243–253.

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 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 6544–6555, Florence, Italy.

Marie-Anne Lachaux, Armand Joulin, and Guillaume Lample. 2020. Target conditioning for one-to-many generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2853–2862.

Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020a. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional VAEs. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics*, pages 208–224, Online.

Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020b. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional vaes. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 208–224.

Jin Li, Peng Qi, and Hong Luo. 2022a. Generating consistent and diverse QA pairs from contexts with BN conditional VAE. In *25th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD*, pages 944–949.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL*, Berlin, Germany.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP*, pages 4582–4597.

Zhuang Li, Lizhen Qu, Qiongkai Xu, Tongtong Wu, Tianyang Zhan, and Gholamreza Haffari. 2022b. Variational autoencoder with disentanglement priors for low-resource task-specific natural language generation. *arXiv*, abs/2202.13363.

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

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, ACL*, pages 3154–3169, Dublin, Ireland.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI*, pages 3622–3628.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations, ICLR*, New Orleans, LA, USA.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. UNICORN on RAINBOW: A universal commonsense reasoning model on a new multitask benchmark. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI*, pages 13480–13488.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI*, pages 8449–8456, New York, NY, USA.

Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019a. Learning disentangled representations for recommendation. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS*, pages 5712–5723, Vancouver, BC, Canada.

Kaixin Ma, Jonathan Francis, Quanyang Lu, Eric Nyberg, and Alessandro Oltramari. 2019b. Towards generalizable neuro-symbolic systems for commonsense question answering. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pages 22–32, Hong Kong, China. Association for Computational Linguistics.

Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. 2018. Mcscript: A novel dataset for assessing machine comprehension using script knowledge. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC*, Miyazaki, Japan.

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, ACL*, pages 1463–1475.

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.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS*, pages 8024–8035, Vancouver, BC, Canada.

Fanyi Qu, Xin Jia, and Yunfang Wu. 2021. Asking questions like educational experts: Automatically generating question-answer pairs on real-world examination data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 2583–2593, Punta Cana, Dominican Republic.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 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.

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, EMNLP*, pages 2383–2392, Austin, Texas, USA.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. ATOMIC: an atlas of machine commonsense for if-then reasoning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, pages 3027–3035.

Lei Shu, Alexandros Papangelis, Yi-Chia Wang, Gökhan Tür, Hu Xu, Zhaleh Feizollahi, Bing Liu, and Piero Molino. 2020. Controllable text generation with focused variation. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 3805–3817.

Yibo Sun, Duyu Tang, Nan Duan, Tao Qin, Shujie Liu, Zhao Yan, Ming Zhou, Yuanhua Lv, Wenpeng Yin, Xiaocheng Feng, Bing Qin, and Ting Liu. 2020. Joint learning of question answering and question generation. *IEEE Transactions on Knowledge and Data Engineering, TKDE*, 32(5):971–982.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*, pages 4149–4158, Minneapolis, MN, USA.

Huazheng Wang, Zhe Gan, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, and Hongning Wang. 2019. Adversarial domain adaptation for machine reading comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, EMNLP-IJCNLP*, pages 2510–2520, Hong Kong, China.

Liuyin Wang, Zihan Xu, Zibo Lin, Haitao Zheng, and Ying Shen. 2020a. Answer-driven deep question generation based on reinforcement learning. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING*, pages 5159–5170, Barcelona, Spain.

Peifeng Wang, Filip Ilievski, Muhao Chen, and Xiang Ren. 2021. Do language models perform generalizable commonsense inference? In *Findings of the Association for Computational Linguistics: ACL/IJCNLP*, pages 3681–3688.

Ye Wang, Jingbo Liao, Hong Yu, and Jiaxu Leng. 2022. Semantic-aware conditional variational autoencoder for one-to-many dialogue generation. 34(2):13683–13695.

Zhen Wang, Siwei Rao, Jie Zhang, Zhen Qin, Guangjian Tian, and Jun Wang. 2020b. Diversify question generation with continuous content selectors and question type modeling. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2134–2143.

Linjuan Wu, Shaojuan Wu, Xiaowang Zhang, Deyi Xiong, Shizhan Chen, Zhiqiang Zhuang, and Zhiyong Feng. 2022. Learning disentangled semantic representations for zero-shot cross-lingual transfer in multilingual machine reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL*, pages 991–1000, Dublin, Ireland.

Hang Xiao, Feng Wang, Jianfeng Yan, and Jingyao Zheng. 2018. Dual ask-answer network for machine reading comprehension. *CoRR*, abs/1809.01997.

Hongbin Ye, Ningyu Zhang, Shumin Deng, Xiang Chen, Hui Chen, Feiyu Xiong, Xi Chen, and Huajun Chen. 2022. Ontology-enhanced prompt-tuning for few-shot learning. In *The ACM Web Conference, WWW '22*, pages 778–787, Lyon, France.

Jianxing Yu, Xiaojun Quan, Qinliang Su, and Jian Yin. 2020. Generating multi-hop reasoning questions to improve machine reading comprehension. In *Proceedings of the 2020 World Wide Web Conference*, pages 550–561, Taipei, Taiwan.

Jianxing Yu, Qinliang Su, Xiaojun Quan, and Jian Yin. 2023. Multi-hop reasoning question generation and its application. *IEEE Transactions on Knowledge and Data Engineering, TKDE*, 35(1):725–740.

Jiawen Zhang, Jiaqi Zhu, Yi Yang, Wandong Shi, Congcong Zhang, and Hongan Wang. 2021. Knowledge-enhanced domain adaptation in few-shot relation classification. In *The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '21*, pages 2183–2191, Singapore.

Ruqing Zhang, Jiafeng Guo, Lu Chen, Yixing Fan, and Xueqi Cheng. 2022. A review on question generation from natural language text. *ACM Transactions on Information Systems*, 40(1):14:1–14:43.

Shiyue Zhang and Mohit Bansal. 2019b. Addressing semantic drift in question generation for semi-supervised question answering. In *Proceedings of the 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.

Xueliang Zhao, Wei Wu, Chongyang Tao, Can Xu, Dongyan Zhao, and Rui Yan. 2020. Low-resource knowledge-grounded dialogue generation. In *8th International Conference on Learning Representations, ICLR*, Addis Ababa, Ethiopia.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In *Natural Language Processing and Chinese Computing - 6th CCF International Conference, NLPCC*, volume 10619, pages 662–671, Dalian, China. Springer.

A Settings of All Evaluated Baselines

The pre-trained language model *RoBERTa* was used to initialize the word embeddings. The distribution overlap metrics (i.e., *QAE* and R-QAE) were computed by the *UNICORN* QA model.

Settings of NQG++: The hidden state size of the GRU was set to 512. The lexical and answer position features were embedded to 32-dimensional vectors. The dropout was used with a probability $p = 0.5$. *Stanford CoreNLP* v3.7.0 was utilized to annotate *POS* and *NER* tags in the sentences. During training, the model was initialized randomly by a *Gaussian* distribution with the *Xavier* scheme. A combination of *Adam* and simple *SGD* was used as the optimizer. For the *Adam* optimizer, the learning rate was set to 0.001 with two momentum parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ respectively. ϵ was set to 10^{-8}. The *SGD* optimizer was initialized with a learning rate of 0.5 and halved if the *BLEU* score on the development set drops for twelve consecutive tests. Gradient clipping with range $[-5, 5]$ was utilized for both *Adam* and *SGD* phases. To speed up convergence, grid search was employed with the mini-batch size of 64. In the test phase, a beam search was used with a size of 12.

Settings of UniLM: The batch size was 32. The masking probability, learning rate and label smoothing rate were 0.7, $2e^{-5}$ and 0.1, respectively.

Settings of SGGDQ: It adopted a 1-layer GRU with hidden units of 512 dimensions. For the graph encoder, the node embedding size was set to 256, plus the *POS* and answer tag embeddings with 32 dimensions for each. The number of layers was set to 3 and the hidden state size was 256. *Adam* was employed with a mini-batch size 32. The learning rate was initially set to 0.001, and adaptive learning rate decay was applied. Early stopping was utilized with a dropout rate of 0.3 for both the encoder and decoder and 0.1 for all attention mechanisms.

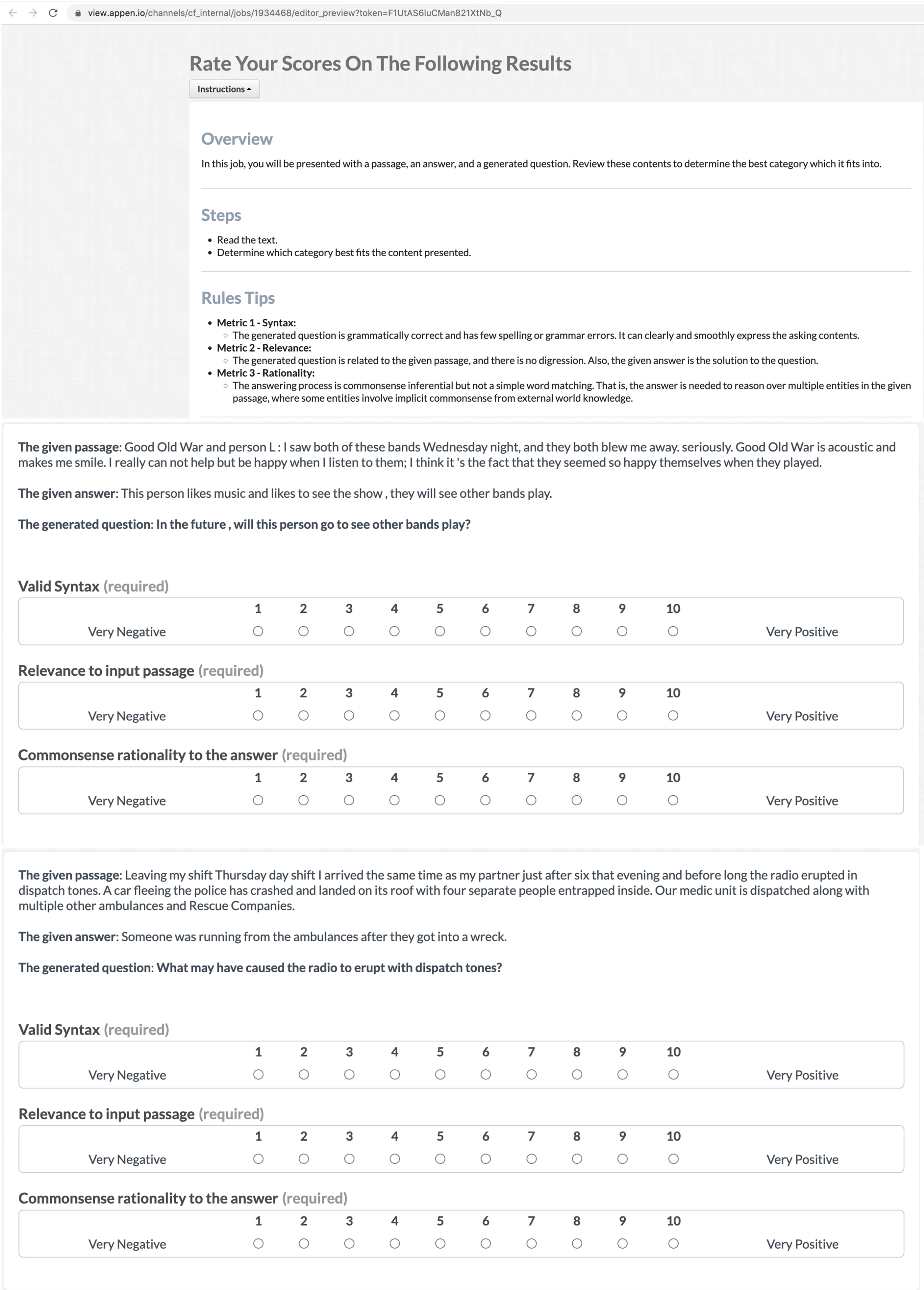
Settings of HCVAE: The hidden dimension of the Bi-LSTM was set to 300 for posterior and prior generation networks. The dimension of the encoder and the decoder was set to 450 and 900, respectively. The dimension of latent variable z_x was set as 50, and z_y was defined to be a 10-way categorical variable. The QA model was fine-tuned for 2 epochs. *Adam* optimizer was used with a batch size of 32 and the initial learning rate of $5 \cdot 10^{-5}$ and 10^{-3} respectively. To prevent posterior collapse, the model multiplied 0.1 to the KL divergence terms of question and answer.

Settings of DAANet: The parameters were randomly initialized by the *fan-avg* strategy. Dropout was mainly applied to the encoding layer with a keep rate of 0.9. The coverage loss weight κ was 1.0. The gradient was clipped by restricting its $\ell_2 - norm$ less than or equal to 5.0. *Adam* optimizer was adopted with a batch size of 16. The learning rate was increased from zero to 0.001 with an inverse exponential function and then fixed for the remainder of the training. During testing, autoregressive decoding was conducted separately for QA and QG. Decoding is terminated when the model encountered the first *<END>* or when the sequence contained more than 100 words.

Settings of SemQG: The *WordPiece* tokenizer was used to tokenize each word and extend the *POS / NER* tags to each word piece. A 2-layer LSTM-RNNs was employed for both the encoder and decoder with a hidden size of 600. Dropout with a probability of 0.3 was applied to the input of each LSTM-RNN layer. *Adam* was utilized as the optimizer with a learning rate of 0.001 for teacher forcing and 0.00001 for reinforcement learning. The batch size was set to 32. For stability, It was first pre-trained with teacher forcing until convergence, then fine-tuned with the mixed loss. Hyper-parameters were tuned on the development set with $\gamma^{qpp} = 0.99$, $\gamma^{qap} = 0.97$, and $n : m = 3 : 1$. The beam search was employed with the size of 10 for decoding. The bigram and trigram repetition penalty was applied.

B Human Evaluation Settings

The rated guideline was shown in Fig.(7).



view.appen.io/channels/cf_internal/jobs/1934468/editor_preview?token=F1UtAS6luCMan821XtNb_Q

Rate Your Scores On The Following Results

Instructions

Overview

In this job, you will be presented with a passage, an answer, and a generated question. Review these contents to determine the best category which it fits into.

Steps

- Read the text.
- Determine which category best fits the content presented.

Rules Tips

- **Metric 1 - Syntax:**
  - The generated question is grammatically correct and has few spelling or grammar errors. It can clearly and smoothly express the asking contents.
- **Metric 2 - Relevance:**
  - The generated question is related to the given passage, and there is no digression. Also, the given answer is the solution to the question.
- **Metric 3 - Rationality:**
  - The answering process is commonsense inferential but not a simple word matching. That is, the answer is needed to reason over multiple entities in the given passage, where some entities involve implicit commonsense from external world knowledge.

The given passage: Good Old War and person L : I saw both of these bands Wednesday night, and they both blew me away. seriously. Good Old War is acoustic and makes me smile. I really can not help but be happy when I listen to them; I think it 's the fact that they seemed so happy themselves when they played.

The given answer: This person likes music and likes to see the show , they will see other bands play.

The generated question: In the future , will this person go to see other bands play?

Valid Syntax (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

Relevance to input passage (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

Commonsense rationality to the answer (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

The given passage: Leaving my shift Thursday day shift I arrived the same time as my partner just after six that evening and before long the radio erupted in dispatch tones. A car fleeing the police has crashed and landed on its roof with four separate people entrapped inside. Our medic unit is dispatched along with multiple other ambulances and Rescue Companies.

The given answer: Someone was running from the ambulances after they got into a wreck.

The generated question: What may have caused the radio to erupt with dispatch tones?

Valid Syntax (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

Relevance to input passage (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

Commonsense rationality to the answer (required)

	1	2	3	4	5	6	7	8	9	10	
Very Negative	○	○	○	○	○	○	○	○	○	○	Very Positive

Figure 7: Human evaluation guideline and an evaluated example.

ACL 2023 Responsible NLP Checklist

A For every submission:

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

Limitations

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

Ethics Statement

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

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

B ☒ Did you use or create scientific artifacts?

Left blank.

☒ B1. Did you cite the creators of artifacts you used?
Left blank.

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☒ 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)?
Left blank.

☒ 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?
Left blank.

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

☑ 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.
Left blank.

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?**

3.1 Data and Experimental Settings

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

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3.1 Data and Experimental Settings

☑ 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?
3.2 Comparisons against State of the Arts

☑ 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.)?
Appendix A Settings of All Evaluated Baselines

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

3.4 Human Evaluations and Analysis

☑ 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.?
Appendix B Human Evaluation Settings

☑ 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)?
Appendix B Human Evaluation Settings

☐ 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?
Not applicable. Left blank.

☐ 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?
Not applicable. Left blank.