R^{2}DQG: A Quality Meets Diversity Framework for Question Generation over Knowledge Bases

Yimeng Ren[1] , **Yanhua Yu**[1*] , **Lizi Liao**[2] , **Yuhu Shang**[1] , **Kangkang Lu**[1] and **Mingliang Yan**[1]
[1]Beijing University of Posts and Telecommunications [2]Singapore Management University
{renyimeng, yuyanhua}@bupt.edu.cn, lzliao@smu.edu.sg, {shangyuhu, lukangkang, yml123}@bupt.edu.cn

Abstract

The task of Knowledge-Based Question Generation (KBQG) involves generating natural language questions from structured knowledge sources, posing unique challenges in balancing linguistic diversity and semantic relevance. Existing models often focus on maximizing surface-level similarity to ground-truth questions, neglecting the need for diverse syntactic forms and leading to semantic drift during generation. To overcome these challenges, we propose Refine-Reinforced Diverse Question Generation (**R^{2}DQG**), a two-phase framework leveraging a generation-then-refinement paradigm. The **Generator** first constructs a diverse set of expressive templates using dependency parse tree similarity, capturing a wide range of syntactic patterns and styles. These templates guide the creation of question drafts, ensuring both diversity and semantic relevance. In the second phase, a **Corrector** module refines the drafts to mitigate semantic drift and enhance overall coherence and quality. Experiments on public datasets show that R^{2}DQG outperforms state-of-the-art models in generating diverse, contextually accurate questions. Moreover, synthetic datasets generated by R^{2}DQG enhance downstream QA performance, underscoring the practical utility of our approach.

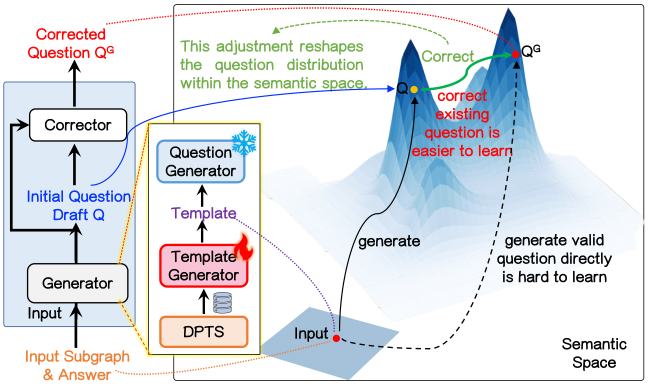


Figure 1: A template-aware *Generator* enhances syntactic and stylistic diversity, while a *Corrector* refines drafts, ensuring semantic alignment and linguistic richness.

1 Introduction

Generating natural language questions based on a set of formatted facts is the core objective of Knowledge-Based Question Generation (KBQG) [Guo *et al.*, 2024a]. Over the last decade, KBQG has garnered substantial research interest across diverse fields, exemplified by its applications in academia and industry. In intelligent tutoring systems, KBQG plays a vital role, as educational questions or quizzes are useful for student assessment and coaching purposes [Agrawal *et al.*, 2024]. Furthermore, in industry, KBQG can potentially be used for generating high-quality QA pairs, thereby enhancing the corpus of training data for Knowledge-Based Question Answering (KBQA) tasks [Li *et al.*, 2024; Lyu *et al.*, 2021].

Over the years, several methods have been developed to address the KBQG task, each focusing on specific aspects of question generation. Early rule-based approaches relied on predefined linguistic templates and graph traversal rules, generating interpretable but limited questions due to their rigidity and lack of scalability [Jia and Liang, 2016; Seyler *et al.*, 2017]. Later, Seq2Seq models introduced encoder-decoder architectures to model relationships between subgraphs and natural language questions, improving flexibility [Chen *et al.*, 2023; Elsahar *et al.*, 2018]. However, these models often struggled to produce diverse syntactic structures. With the advent of pretrained language models (PLMs), such as BERT, T5, and GPT, KBQG has seen significant advancements in generating semantically relevant questions [Kim *et al.*, 2019; Guo *et al.*, 2022]. Nonetheless, these models frequently prioritize surface-level similarity to ground truth at the expense of linguistic diversity. Meanwhile, diversity-driven approaches employ mechanisms like external knowledge inject, content selectors, and diverse decoding strategies to introduce syntactic and lexical variety into the generated questions [Guo *et al.*, 2024c; Bi *et al.*, 2020; Narayan *et al.*, 2022].

Despite these advancements, KBQG systems face several critical challenges. First, many approaches suffer from a lack

[*] Corresponding author.

of semantic guidance, as the templates that guide question structure are often unavailable until the generation process is complete. This makes it difficult to provide token-level guidance during generation, leading to contextually imprecise questions. Second, existing methods follow an inflexible generation process, where suboptimal subgraph-question pairs are discarded instead of refined. Refining "deficient" drafts is often more practical and efficient than generating well-formed questions from scratch, but current models lack the capability to perform such corrections effectively. Finally, KBQG systems face the challenge of conflicting objectives in balancing linguistic diversity and semantic relevance. These goals often conflict, requiring innovative strategies to resolve the trade-off and generate both high-quality and diverse questions simultaneously.

To address these challenges, we propose R^{2}DQG[1]: Refine-Reinforced Diverse Question Generation, a novel two-phase framework that effectively balances linguistic diversity and semantic relevance in Knowledge-Based Question Generation. At its core, R^{2}DQG follows a generation-then-refinement paradigm inspired by human question-drafting processes, ensuring flexibility and robustness. Figure 1 illustrates the R^{2}DQG workflow, comprising two key components: the Generator and the Corrector. The Generator first constructs a diverse set of expressive templates by leveraging dependency parse tree similarity, capturing varied syntactic structures and phrasing styles. These templates serve as guiding scaffolds to generate diverse yet semantically aligned question drafts. The Corrector then refines these drafts, mitigating semantic drift and enhancing coherence, resulting in well-formed, high-quality questions. By decoupling the generation and refinement processes, R^{2}DQG effectively balances diversity and relevance while offering a scalable and practical solution. Our framework overcomes existing limitations with a novel drift compensation mechanism that refines imperfect drafts for enhanced expressiveness and accuracy. Extensive experiments and case studies on two datasets validate the effectiveness of R^{2}DQG through benchmark evaluations and human assessments.

To sum up, the contributions of our work are as follows:

- We propose R^{2}DQG, a novel post-hoc KBQG framework inspired by human drafting and revising processes, to generate diverse and semantically accurate questions using a two-phase generation-then-refinement paradigm.
- We design a template-aware Generator to enhance syntactic and stylistic diversity, coupled with a Corrector module to refine drafts, ensuring semantic alignment and linguistic richness.
- Extensive experiments on public datasets validate that R^{2}DQG achieves superior performance on diversity, while its synthetic datasets significantly boost downstream QA performance.

2 Related Work

Knowledge Base Question Generation. The field of KBQG has witnessed significant advancements. Early rule-based methods relied on predefined linguistic templates and graph traversal rules, generating interpretable but limited questions due to their rigidity and lack of scalability [Jia and Liang, 2016; Seyler *et al.*, 2017]. With the emergence of data-driven learning approaches, Seq2Seq model introduced encoder-decoder architectures to model relationships between subgraphs and natural language questions, improving flexibility [Chen *et al.*, 2023; Elsahar *et al.*, 2018]. With the advent of pretrained language models (PLMs), such as BERT, T5, and GPT, KBQG has seen significant advancements in generating semantically relevant questions [Kim *et al.*, 2019; Guo *et al.*, 2022; Chen *et al.*, 2023; Fei *et al.*, 2022]. LLMs-based methods [Wang *et al.*, 2024], with their powerful generation capabilities, have further boosted KBQG. Techniques such as skeleton-guided prompting [Guo *et al.*, 2024b], in-context demonstrations [Liang *et al.*, 2023] and multi-agent collaborative frameworks [Zhao *et al.*, 2024] inspired by memory mechanisms [Dang *et al.*, 2024a; Dang *et al.*, 2024b] have been used to guide the model generation.

Diversifying Question Generation. Diversity in question generation remains a critical challenge, with research efforts focused on two dimensions: leveraging internal knowledge and exploiting external patterns. The former make use of internal knowledge such as content selection [Wang *et al.*, 2020] and apply various decoding algorithms to promote diversity, such as diverse beam search [Narayan *et al.*, 2022], nucleus sampling [Holtzman *et al.*, 2019], entmax transformation [Martins *et al.*, 2020] and well-designed loss functions [Zhang and Zhu, 2021]. And those that exploit external patterns such as fact-infused [Deschamps *et al.*, 2021], question type ontology [Cao and Wang, 2021], choose reliable pesudo pairs [Guo *et al.*, 2024c].

Most similar to our work is [Guo *et al.*, 2024c]. However, it fails to explicitly guide generation following syntactic structure and focuses on directly generating valid questions. Contrary to this, our focus is to adopt templates contain useful syntactic structures that help organize questions well. Besides, we refine specific expressions that align with quality-meets-diversity control conditions within semantic space, thereby guiding the generated questions toward high-quality outputs on the premise of the diverse expression ways.

3 Methodology

We first formalize the KBQG problem statement and give an overview of our R^{2}DQG. Then we elaborate on the details of two individual components in the following sections.

3.1 Task Definition and Model Architecture

Problem Formulation. The KBQG task aims to generate a natural language question $\hat{q}_i = (w_1, \ldots, w_m)$, which can be expressed as optimizing the model parameter Θ to maximize the conditional likelihood P. This can further be decomposed into a sequential word-by-word generation process:

$$\begin{aligned}\hat{q}_i &= \arg\max_{q_i} P(q_i \mid G_i, a_i; \Theta) \\ &= \arg\max_{w_1,\ldots,w_m} \prod_{j=1}^{m} P(w_j \mid w_1, \ldots, w_{j-1}, G_i, a_i; \Theta) \quad (1)\end{aligned}$$

[1] Codes are here: https://github.com/Elaine-explorer/R2DQG.

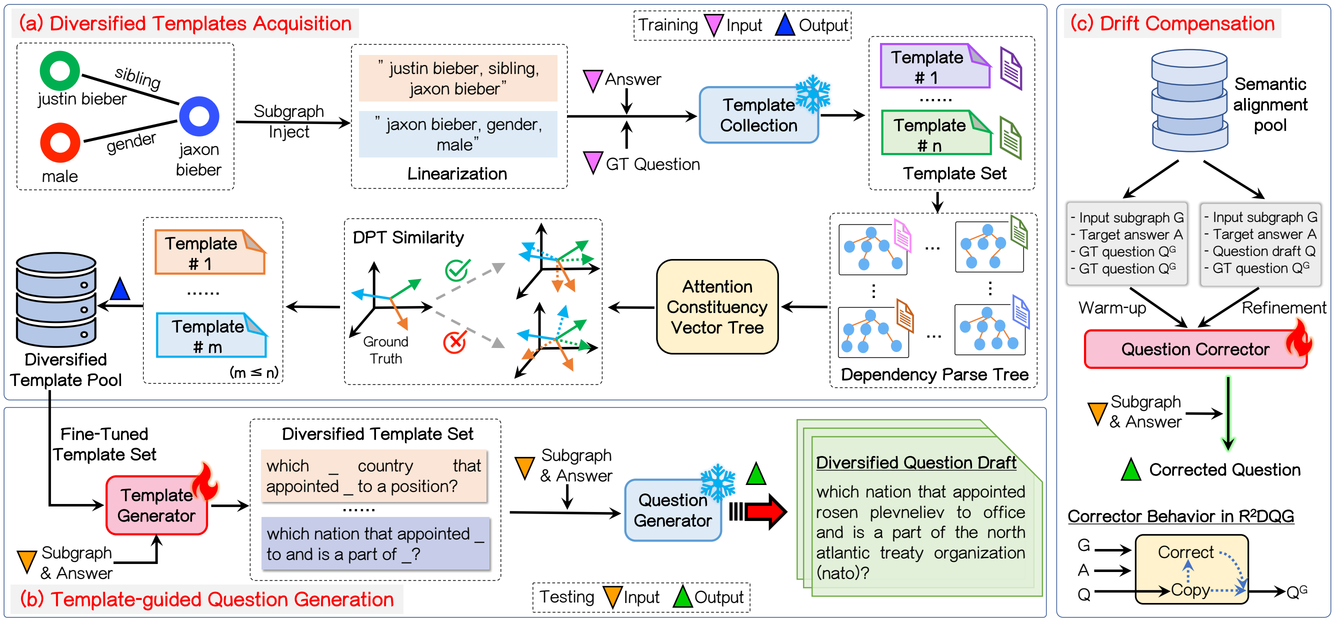


Figure 2: The R^{2}DQG framework consists of two modules: the Generator, which includes *(a) Diversified Templates Acquisition* to enhance diversity and *(b) Template-guided Question Generation* to produce diverse drafts, and the Corrector, which applies *(c) Drift Compensation* to refine drafts into high-quality, semantically accurate questions.

where $G_i = \{k_1^{(i)}, \ldots, k_n^{(i)}\}$ represents the subgraph comprising a set of triples, a_i signifies the target answer, and q_i denotes the generated question.

Model Overview. The overall R^{2}DQG framework, illustrated in Figure 2, comprises two key modules: the Generator and the Corrector. The Generator consists of two components: (a) Diversified Templates Acquisition, which selects expressive templates based on tree kernel similarity to enhance diversity, and (b) Template-guided Question Generation, which leverages these templates to generate diverse question drafts while maintaining syntactic variety. However, these drafts may experience semantic drift, necessitating further refinement. Hence, the Corrector introduces (c) Drift Compensation, a post-editing mechanism that refines initial drafts into high-quality, semantically accurate questions. This two-stage process ensures that the generated questions achieve both linguistic diversity and semantic precision.

3.2 Generator

Now we deal with the first challenge of deriving diversified question drafts with the guidance of syntax templates. To achieve that, we propose a simple but effective template-guided Generator, which collect and filter diversified template by hierarchically expanding constituents in syntax contexts throughout the syntax tree. The resulting templates, containing rich semantic information, can serve as reliable grammatical guidance to assist in the generation of linguistically diverse question drafts.

Diversified Templates Acquisition. The templates play an important part in guiding question generation. As depicted in Figure 2(a), we propose a generate-then-filter strategy to extract diverse template training samples from existing QG datasets. For each input $\mathcal{D} = (G_i, a_i, q_i)$, we first linearize it into a textual prompt, and then feed it into the closed-source LLMs to generate diverse candidate templates. Then, we select the top-K results of the beam search as candidate templates. However, solely relying on LLMs' capability makes it hard to guarantee the effectiveness of the generated templates and may introduce bias. To mitigate errors made by LLMs, we impose constraints grounded in Dependency Parse Tree Similarity (DPTS) to filter templates by jointly computing their semantic and syntactic similarity. We use a neural parser [Qi *et al.*, 2020] to generate parse trees from sentences, which are then transformed into graph representations. These dependency parse trees (DPT) capture syntactic relationships, where nodes represent words and edges denote dependencies. Unlike n-gram-based metrics, DPT-based representations enable the generation of templates that are both diverse and semantically valid. To compute similarity, we adopt the efficient Attention Constituency Vector Tree (ACV-Tree) kernel [Quan *et al.*, 2019], which quantifies similarity based on shared substructures between parse trees. The tree kernel similarity between two trees T_1 and T_2 is calculated as follows:

$$TreeKernel(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \Delta(n_1, n_2), \qquad (2)$$

In this expression, N_{T_1} and N_{T_2} are the sets of nodes in T_1 and T_2, respectively. The function $\Delta(n_1, n_2)$, which is simplified from $\Delta(\cdot)$, measures the similarity between two nodes n_1 and n_2, capturing both semantic and structural features, and is computed as:

$$\Delta(\cdot) = \begin{cases} 0, & n_1 \text{ and/or } n_2 \text{ are non-leaf nodes } \wedge n_1 \neq n_2 \\ S, & n_1 \text{ and } n_2 \text{ are leaf nodes} \\ T, & \text{otherwise} \end{cases}$$

$$S = Att_{w_1} \times Att_{w_2} \times SIM(vec_1, vec_2), \tag{3}$$

$$T = \mu \left(\lambda^2 + \sum_{p=1}^{l_m} \Delta_p(cn_1, cn_2) \right). \tag{4}$$

Let vec_1 and Att_{w_1} (resp., vec_2 and Att_{w_2}) denote the word vector and attention weights of node n_1 (resp., n_2). The cosine similarity between the vectors of the leaf nodes is computed by the function $SIM(\cdot)$. Let c_{n_1} (resp., c_{n_2}) represent the list of child nodes of n_1 (resp., n_2). The term $\Delta_p(.)$ counts the number of common subsequences of length p. The parameters λ and μ are decay factors controlling the influence of the child sequences' lengths and the tree height, respectively.

Finally, we select exemplars of input-template with high DPTS scores to form the training dataset, denoted by $\mathcal{D}_{\text{train}} = \{(G_i, a_i, t_i)\}_{i=1}^{N}$.

Template-guided Question Generation. Afterwards, we focus on generating questions according to the input sub-graph, the expected answer, and the template, in which the input sub-graph offers contextual information to determine *what to ask*, the answer reflects the asking direction to indicate *what is the target*, and the template guides *how to ask* in a reasonable manner. Specifically, we fine-tune the LLaMA-based Template Generator to automatically produce question templates using the constructed dataset $\mathcal{D}_{\text{train}}$. To achieve efficient adaptation, we freeze the parameters Φ of LLaMA and incorporate low-rank adaptation (LoRA) [Hu *et al.*, 2021] adapters. Formally, we generate template t_i conditioned on the given input (G_i, a_i), where the objective $\mathcal{L}'$ is defined as:

$$\mathcal{L}' = \sum_{i=1}^{N} p_\Phi(t_i | G_i, a_i) = -\sum_{i=1}^{N} \sum_{j=1}^{|t|} \log p_\Phi(t_j | t_{<j}, G_i, a_i).$$

Building upon the question templates, we utilize LLMs to seamlessly integrate them into the test input, steering the model to generate the initial question drafts q^d, as shown in Figure 2(b).

The diversified initial question drafts may suffer from "semantic drift", where the generated questions deviate from the provided input subgraph and answer, requiring further adjustments for alignment. Rather than preventing drift, our R^{2}DQG framework introduces a semantic drift compensation method, called Corrector, which learns the corrective residuals between the desired question and the imperfect draft to effectively mitigate the issue of semantic drift.

Similar to how residual blocks enhance an architecture without altering its foundation, the Corrector refines initial question drafts through a flexible copy-and-refine mechanism. To train the Corrector, we first create example pairs that map deficient questions to well-formed ones. As shown in Figure 2(c), the input subgraph G_i, target answer a_i, prior question draft q_i^d, and ground-truth question q_i^{g} are collected to construct the data pool as follows:

$$\mathcal{D}_{\mathcal{C}} = \left\{ G_i, a_i, q_i^d, q_i^{\mathrm{g}} \right\}_{i=1}^{N}, \tag{5}$$

The semantic alignment pool $\mathcal{D}_{\mathcal{C}}$ is applied to fine-tune LLaMA, denoted as $\mu_\phi(q_i^{\mathrm{g}} | G_i, a_i, q_i^d)$, where the parameters ϕ are trained to align the generated questions with the groundtruth q_i^{g}. The probability distribution for generating the refined question can be represented as:

$$\begin{aligned} \pi'(q^{\mathrm{g}} \mid G, a) &= \sum_{q^k} \mu_\phi(q^{\mathrm{g}} \mid q^k, G, a) \pi_\theta(q^k \mid G, a) \\ &\geq \mu_\phi(q^{\mathrm{g}} \mid q^d, G, a) \cdot \pi_\theta(q^d \mid G, a), \end{aligned} \tag{6}$$

where q^k is a possible draft question generated by the template-guided Generator π_θ.

By calculating the empirical loss over the dataset $\mathcal{D}_{\mathcal{C}}$, we get the following constraint:

$$\begin{aligned} -\mathbb{E}_{\mathcal{D}_{\mathcal{C}}} \left[\log \pi'(q^{\mathrm{g}} | G, a) \right] \leq -\mathbb{E}_{\mathcal{D}_{\mathcal{C}}} &\left[\log \mu_\phi(q^{\mathrm{g}} | q^d, G, a) \right] \\ - \mathbb{E}_{\mathcal{D}_{\mathcal{C}}} &\left[\log \pi_\theta(q^d | G, a) \right]. \end{aligned} \tag{7}$$

Since the second term does not involve ϕ, the training objective can be derived as:

$$\min_{\phi} \mathcal{L}_{\text{LLM}}(\phi, \mathcal{D}_{\mathcal{C}}) = -\mathbb{E}_{\mathcal{D}_{\mathcal{C}}} \left[\log \mu_\phi(q^{\mathrm{g}} | q^d, G, a) \right]. \tag{8}$$

Corrector's Training Strategy — Residual Refinement. Now that the generated question drafts q_i^d may vary significantly in quality. Some drafts are close to the target questions q_i^g and do not require further correction. Other drafts may exhibit semantic drift or deviate significantly from the desired target, requiring substantial adjustments. Treating all drafts uniformly during correction not only increases training complexity but also risks over-correcting high-quality drafts. To address this, we first construct a $G - A - Q^G - Q^G$ dataset using a subset of the training data to pre-train the R^{2}DQG model, a process we refer to as warm-up. This phase helps the model establish a foundational ability to handle question drafts with minor semantic drift. Subsequently, we further train the model using the $G - A - Q - Q^G$ dataset to enhance its correction capabilities. This training strategy aligns with techniques applied beyond the question generation domain. In particular, ResNet employs residual connections to mitigate the vanishing gradient problem in deep neural networks, conceptually resembling our drift compensation strategy.

During inference, a question q_j is sampled from the conditional probability distribution μ_ϕ, and the model is then tasked with predicting the corrected question q_j.

$$q_j = \arg\max_{q_j} \mu_\phi(q_j \mid G_j, a_j, \hat{q}_j). \tag{9}$$

The Corrector is a versatile, pluggable component that operates independently of the preceding Generator's parameters, π_θ. Instead of generating new questions directly from the input, it focuses on learning the alignment between the distribution of initial question drafts and the desired questions, effectively mitigating semantic drift.

	Method	**Top-3 Questions**					**Top-5 Questions**					**Top-10 Questions**				
		D@3	D-1	B-1	ME	Avg	D@5	D-1	B-1	ME	Avg	D@10	D-1	B-1	ME	Avg
Graph2Seqs methods	G2S+AE[Chen *et al.*, 2023]	16.08	38.25	41.10	28.25	30.92	18.08	28.29	41.21	28.26	28.96	19.92	18.14	41.01	28.21	26.82
	G2S+AE+RL[Chen *et al.*, 2023]	15.72	37.85	42.72	29.80	31.52	17.29	27.79	42.76	29.89	29.43	19.24	17.82	42.79	29.96	27.45
PLMs methods	MHQG[Kumar *et al.*, 2019]	10.49	35.25	30.23	25.52	25.37	12.31	26.08	30.36	25.63	23.60	14.24	16.72	30.13	25.40	21.62
	MHQG+AE[Kumar *et al.*, 2019]	11.82	35.94	31.84	27.38	26.74	13.55	26.59	31.86	27.53	24.88	15.31	17.05	31.92	27.38	22.92
	JointGT[Ke *et al.*, 2021]	17.65	38.83	55.05	32.68	36.05	19.52	28.51	55.08	32.69	33.95	21.59	18.28	54.98	32.49	31.84
	DSM[Guo *et al.*, 2022]	17.30	38.64	59.65	33.48	37.27	19.37	28.58	59.63	33.42	35.25	20.95	18.21	59.69	33.57	33.10
	DiversifyQG[Guo *et al.*, 2024c]	25.50	42.50	48.70	30.73	36.86	27.22	31.44	48.72	30.82	34.55	28.76	20.03	48.53	30.58	31.98
	T5	16.61	38.22	49.35	29.17	33.34	18.22	28.27	49.47	29.35	31.33	20.01	18.01	49.26	28.99	29.07
	Bart	16.63	38.47	49.87	29.62	33.65	18.37	28.25	49.87	29.42	31.48	20.05	18.04	49.83	29.6	29.38
LLMs methods	GPT-4-Turbo	19.88	41.19	53.35	31.45	36.47	21.56	30.47	53.44	31.48	34.24	23.54	19.41	53.49	31.60	32.01
	GPT-3.5-Turbo	19.92	40.91	51.52	30.61	35.74	21.53	30.04	51.42	30.42	33.35	23.13	19.14	51.60	30.62	31.12
	KQG-CoT[Liang *et al.*, 2023]	17.68	39.09	54.15	31.25	35.54	19.30	28.92	54.07	31.14	33.36	21.18	18.42	54.26	31.40	31.32
	RoleAgentQG[Zhao *et al.*, 2024]	18.55	39.62	54.97	31.91	36.26	20.18	29.09	54.91	31.85	34.01	22.26	18.53	54.86	31.83	31.87
	SGSH(B+S)[Guo *et al.*, 2024b]	19.12	41.05	63.84	35.53	39.89	21.01	30.14	63.91	35.61	37.67	22.95	19.20	63.82	35.46	35.36
	SGSH[Guo *et al.*, 2024b]	20.05	41.29	**65.51**	**36.22**	40.77	21.67	30.32	**65.42**	**36.12**	38.38	23.22	19.32	**65.49**	**36.21**	36.06
Ours	**R^{2}DQG**	**36.93**	**47.32**	58.18	33.29	**43.93**	**38.72**	**36.95**	58.41	33.40	**41.87**	**40.64**	**22.69**	58.40	33.38	**38.78**

Table 1: Overall performance comparison between the three categories KBQG methods and our R^{2}DQG method on WQ datasets. Since few KBQG studies address the problem of how to generate both high-quality and diverse questions simultaneously, we have chosen many quality-centered models as baselines. [Key: **Best Single Metric**; **Best Average Metric**].

	Method	**Top-3 Questions**					**Top-5 Questions**					**Top-10 Questions**				
		D@3	D-1	B-1	ME	Avg	D@5	D-1	B-1	ME	Avg	D@10	D-1	B-1	ME	Avg
Graph2Seqs methods	G2S+AE[Chen *et al.*, 2023]	17.75	33.48	73.70	40.20	41.28	19.90	24.01	73.76	40.34	39.50	21.60	15.45	73.77	40.31	37.78
	G2S+AE+RL[Chen *et al.*, 2023]	16.31	33.39	73.94	40.23	40.97	18.24	24.12	73.81	40.12	39.07	19.98	15.62	73.8	40.11	37.38
PLMs methods	MHQG[Kumar *et al.*, 2019]	11.18	29.17	55.36	31.63	31.83	12.89	21.07	55.46	31.66	30.27	14.93	13.65	55.23	31.48	28.82
	MHQG+AE[Kumar *et al.*, 2019]	12.35	29.43	56.71	32.95	32.86	14.44	21.26	56.81	33.07	31.40	16.15	13.68	56.59	32.85	29.82
	JointGT[Ke *et al.*, 2021]	17.55	34.02	78.72	45.80	44.02	19.45	24.40	78.68	45.71	42.06	21.24	15.70	78.65	45.72	40.33
	DSM[Guo *et al.*, 2022]	17.66	33.98	81.29	46.90	44.96	19.66	24.37	81.22	46.81	43.02	21.43	15.78	81.41	46.92	41.38
	DiversifyQG[Guo *et al.*, 2024c]	28.11	40.60	59.38	33.65	40.44	29.97	29.12	59.45	33.80	38.08	31.71	18.86	59.49	33.71	35.94
	T5	16.97	33.67	74.48	40.74	41.46	19.09	24.32	74.65	40.91	39.74	21.04	15.65	74.39	40.74	37.96
	Bart	17.20	33.73	76.52	41.78	42.31	19.02	24.19	76.47	41.74	40.36	20.78	15.57	76.64	41.91	38.72
LLMs methods	GPT-4-Turbo	20.45	34.2	75.89	40.97	42.88	22.51	24.53	75.86	40.89	40.95	24.31	15.78	75.85	40.84	39.20
	GPT-3.5-Turbo	21.40	34.93	73.14	40.63	42.53	23.19	25.05	73.13	40.59	40.49	24.81	16.22	73.11	40.58	38.68
	KQG-CoT[Liang *et al.*, 2023]	18.25	33.20	77.79	43.47	43.18	20.10	23.98	77.59	43.46	41.28	21.93	15.53	77.70	43.28	39.61
	RoleAgentQG[Zhao *et al.*, 2024]	19.82	33.56	78.90	44.91	44.30	21.86	24.24	78.84	44.75	42.42	23.59	15.70	78.76	44.82	40.72
	SGSH(B+S)[Guo *et al.*, 2024b]	20.71	33.88	81.96	48.11	46.16	22.43	24.48	82.14	48.22	44.32	24.35	15.75	82.13	48.30	42.63
	SGSH[Guo *et al.*, 2024b]	21.41	34.21	**84.74**	**49.23**	47.40	23.33	24.71	**84.67**	**49.22**	45.48	25.13	15.90	**84.73**	**49.14**	43.72
Ours	**R^{2}DQG**	**35.41**	**45.70**	72.08	40.28	**48.37**	**38.72**	**34.61**	72.02	40.21	**46.39**	**41.95**	**21.23**	72.07	40.26	**43.93**

Table 2: Overall performance comparison between the three categories KBQG methods and our R^{2}DQG method on PQ datasets. [Key: **Best Single Metric**; **Best Average Metric**].

4 Experiment

4.1 Experiment Setup

Datasets. Two public benchmark datasets are used to evaluate R^{2}DQG: WebQuestions (WQ) [Kumar *et al.*, 2019] and PathQuestions (PQ) [Zhou *et al.*, 2018]. Specifically, WQ combines instances from WebQuestionsSP and ComplexWebQuestions, both of which are benchmarks for KBQA. Specifically, WQ consists of train/dev/test set with 18,989/2,000/2,000 instances, while PQ contains 9,793/1,000/1,000 instances.

Metrics. We introduce a diversity metric with semantic constraints, termed *Diverse@k* (D@K) [Guo *et al.*, 2024c], to assess the diversity of the top-k generated questions while maintaining their relevance to the ground truth. We also adopt *Distinct-n* (D-n) [Song *et al.*, 2019], which measures the diversity of generated text by calculating the proportion of unique n-grams within the output. We adopt 2 traditional metrics, namely *BLEU 1* (B-1)[Papineni *et al.*, 2002] and *METEOR* (ME) [Banerjee and Lavie, 2005], which compute the ratios of the common n-grams between the generated question and the ground truth. For QA Evaluation, we use *Hits@1* to evaluate whether the top-1 predicted answer is accurate and report the *F1* score. For Human Evaluation, three well-educated annotators are employed to measure the *Fluency*, *Relevance* and *Diversity* of the generated questions.

Baselines. we choose three categories of methods as baselines. ***(i)*** **Graph2Seqs methods**, including G2S+AE and G2S+AE+RL, aim to capture the complex relationship between entities, thereby improving the knowledge consistency of the generated question. ***(ii)*** **PLMs methods**, including MHQG, MHQG+AE, JointGT, DSM, DiversifyQG, T5 and Bart, adopt PLMs to generate question by fine-tuning them on specific tasks or datasets. and ***(iii)*** **LLMs methods**, including GPT-4-Turbo, GPT-3.5-Turbo, KQG-CoT, RoleAgentQG, SGSH and SGSH(B+S), design specific natural lan-

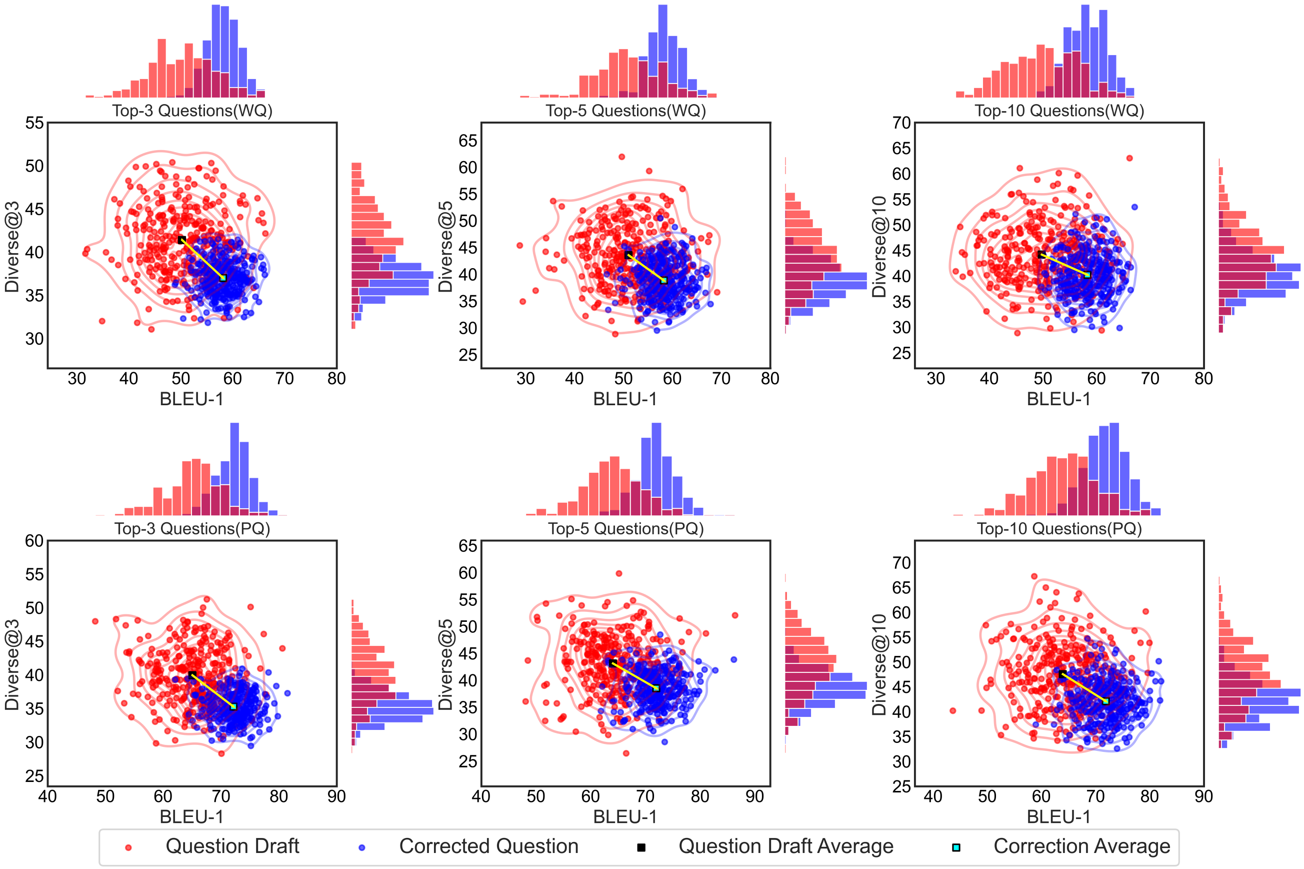


Figure 3: Distribution of BLEU-1 and Diverse on two datasets.

guage prompts to directly influence question generation.

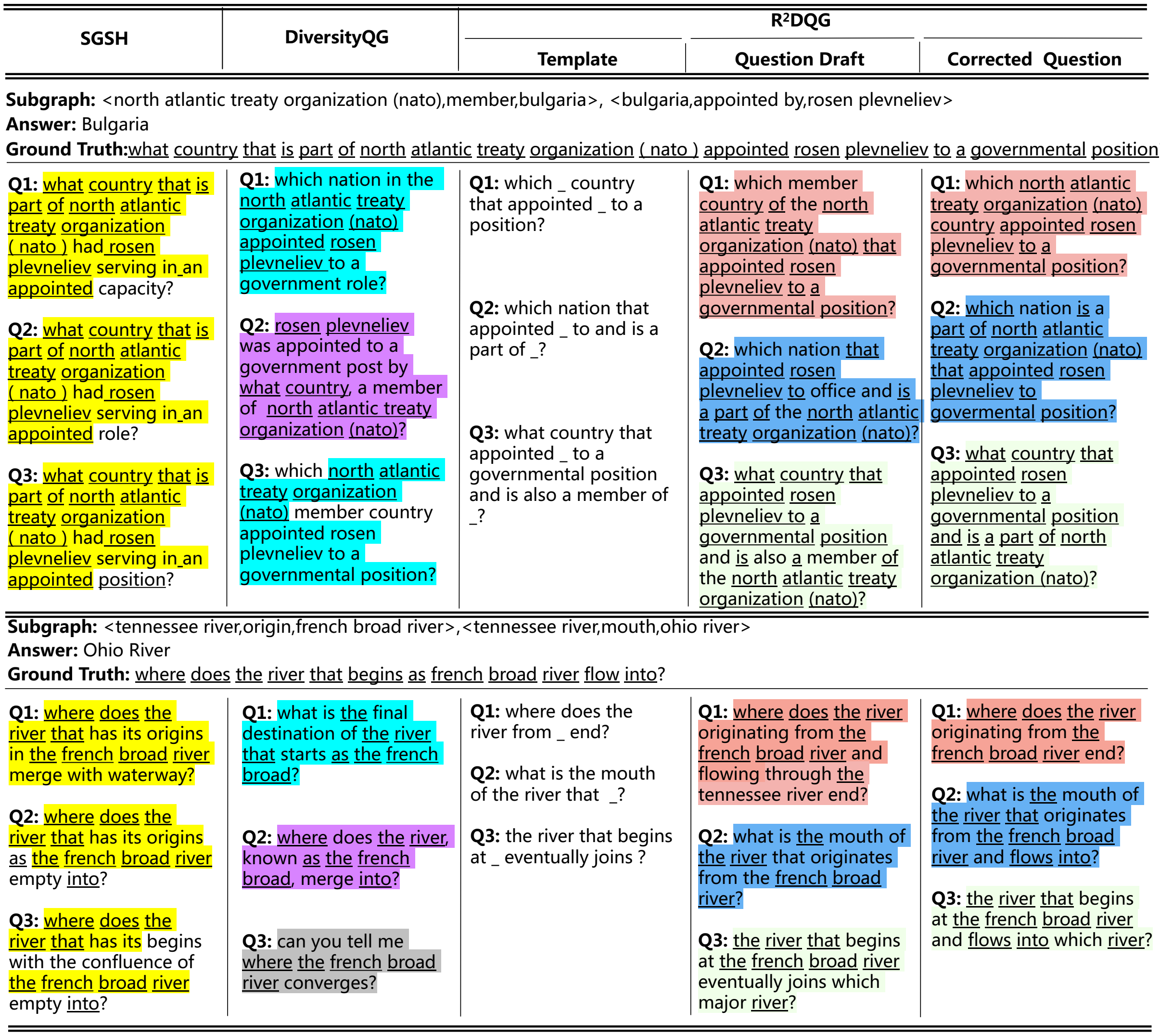
Implementations. All experiments are implemented using PyTorch on a server equipped with 8 NVIDIA GeForce RTX 3090Ti GPU. We select GPT-3.5-turbo-1106 as the default base LLM for the zero-shot setting and GPT-3.5-turbo-16k for the few-shot setting. For the Generator and Corrector that need to be fine-tuned, we employ LLaMA as the backbone and fine-tune them using LoRA. The Generator and Corrector are trained for 3 epochs with a learning rate of 1e-4. Empirically, 20% warm-up samples yield the best performance.

4.2 Performance Comparison

The question generation performances on WQ and PQ are presented in Table 1 and Table 2. The observations from these comparisons are as follows: (1) Regarding the average score, R^{2}DQG achieves the superior results on two datasets, demonstrating the overall high relevance and diversity of the generated questions. (2) Compared with other baselines, our R^{2}DQG method also demonstrates significant improvements regarding the diversity aspect. Notably, considering the top-3 question generation, R^{2}DQG reports up to a 16% increase in Div@3 on the WQ dataset and a 14% increase on the PQ dataset. This can be attributed to the well-designed template generator, which effectively guides the approximate syntactic structure of the generated questions. (3) Notably, the improvements in diversity by R^{2}DQG do not extend equally to quality aspect. This is acceptable, as the R^{2}DQG model often introduces varied forms of expression in pursuit of greater diversity. By design, R^{2}DQG aims to strike an optimal balance between two interrelated yet conflicting objectives—ensuring diverse question expressions while minimizing quality loss. (4) Directly prompting LLM methods such as GPT-4-Turbo and GPT-3.5-Turbo, are proficient at producing a wide range of questions, they face challenges in maintaining high-quality outputs in terms of BLUE-1 and METEOR. (5) Except for DiversifyQG [Guo *et al.*, 2024c], all other baselines exhibit lower diversity, as they tend to generate questions closely resembling the gold standard. This highlights that the absence of effective diversification guidance can significantly degrade performance in diversity metrics. Our results align with those of DiversifyQG.

4.3 Corrector Evaluation

We evaluate the impact of the key Corrector module in our R^{2}DQG model by removing it while keeping all other components unchanged. To assess whether the refined questions achieve the desired relevance and diversity, we analyze the distribution shift before and after the correction process. For this evaluation, we randomly selected 300 samples from different datasets and reported the averaged results in Figure 3.



SGSH	DiversityQG	R^2DQG		
		Template	Question Draft	Corrected Question
Subgraph: <north atlantic treaty organization (nato),member,bulgaria>, <bulgaria,appointed by,rosen plevneliev> **Answer:** Bulgaria **Ground Truth:**what country that is part of north atlantic treaty organization (nato) appointed rosen plevneliev to a governmental position?				
Q1: what country that is part of north atlantic treaty organization (nato) had rosen plevneliev serving in an appointed capacity? **Q2:** what country that is part of north atlantic treaty organization (nato) had rosen plevneliev serving in an appointed role? **Q3:** what country that is part of north atlantic treaty organization (nato) had rosen plevneliev serving in an appointed position?	**Q1:** which nation in the north atlantic treaty organization (nato) appointed rosen plevneliev to a government role? **Q2:** rosen plevneliev was appointed to a government post by what country, a member of north atlantic treaty organization (nato)? **Q3:** which north atlantic treaty organization (nato) member country appointed rosen plevneliev to a governmental position?	**Q1:** which _ country that appointed _ to a position? **Q2:** which nation that appointed _ to and is a part of _? **Q3:** what country that appointed _ to a governmental position and is also a member of _?	**Q1:** which member country of the north atlantic treaty organization (nato) that appointed rosen plevneliev to a governmental position? **Q2:** which nation that appointed rosen plevneliev to office and is a part of the north atlantic treaty organization (nato)? **Q3:** what country that appointed rosen plevneliev to a governmental position and is also a member of the north atlantic treaty organization (nato)?	**Q1:** which north atlantic treaty organization (nato) country appointed rosen plevneliev to a governmental position? **Q2:** which nation is a part of north atlantic treaty organization (nato) that appointed rosen plevneliev to govermental position? **Q3:** what country that appointed rosen plevneliev to a governmental position and is a part of north atlantic treaty organization (nato)?
Subgraph: <tennessee river,origin,french broad river>,<tennessee river,mouth,ohio river> **Answer:** Ohio River **Ground Truth:** where does the river that begins as french broad river flow into?				
Q1: where does the river that has its origins in the french broad river merge with waterway? **Q2:** where does the river that has its origins as the french broad river empty into? **Q3:** where does the river that has its begins with the confluence of the french broad river empty into?	**Q1:** what is the final destination of the river that starts as the french broad? **Q2:** where does the river, known as the french broad, merge into? **Q3:** can you tell me where the french broad river converges?	**Q1:** where does the river from _ end? **Q2:** what is the mouth of the river that _? **Q3:** the river that begins at _ eventually joins ?	**Q1:** where does the river originating from the french broad river and flowing through the tennessee river end? **Q2:** what is the mouth of the river that originates from the french broad river? **Q3:** the river that begins at the french broad river eventually joins which major river?	**Q1:** where does the river originating from the french broad river end? **Q2:** what is the mouth of the river that originates from the french broad river and flows into? **Q3:** the river that begins at the french broad river and flows into which river?

Figure 4: Comparison of top-3 generated questions on WQ. Distinct colors highlight different surface forms for each instance (indicating diversity), and underlined words denote expressions that match the ground truth (indicating quality).

Based on our comprehensive analysis of the experimental results, we have identified several important findings. (1) The correction process leads to higher BLEU-1 scores, demonstrating that the refined questions effectively mitigate the semantic drift problem and align more closely with the target distribution. (2) While the refinement process enhances question quality, it leads to a slight decrease in linguistic diversity, suggesting a necessary trade-off where the Corrector ensures relevance while maintaining creative variation in question generation. (3) Our R^2DQG method exhibits a more concentrated distribution across multiple experiments, reflecting improved stability and consistency in performance. This further validates the effectiveness of R^2DQG in achieving a balanced trade-off between stability and diversity. These analytical findings underscore the importance of the drift compensation strategy in addressing distribution shifts, highlighting its crucial role in optimizing KBQG systems.

4.4 Case Study

To intuitively demonstrate the effectiveness of R^2DQG, Figure 4 presents a comparison of the top-3 generated questions for two examples from the WQ dataset using three methods. Key observations include: (1) We find that questions generated from SGSH are highly consistent in both structure and content, e.g. “what country” and “serving in an appointed” frequently appear across different questions. (2) By contrast, DiversityQG excels in generating express diverse questions, marked with two or three distinct colors. By overly prioritizing linguistic diversity, the model sometimes generates unrelated phrase expressions such as ”can you tell me”, which may detract from the question’s relevance. (3) Notably in R^2DQG, the corrected questions demonstrate superior relevance and diversity compared to its draft and other baseline approach results, highlighting that the drift compensation process not only improves the relevance of the generated question but also maintains diverse expressions. The ability to maintain diversity can be attributed to the strong foundation provided by the initial question drafts. Thus, findings from the case study further reinforce our conclusions.

4.5 Downstream QA Evaluation

To address the issue of insufficient manual annotation corpus during the model training process, KBQG systems are developed to generate synthetic QA pairs and augment QA datasets [Sultan *et al.*, 2020]. Along this line, we evaluate the performance of two typical QA model (i.e., GRAFT-Net [Sun *et al.*, 2018] and NSM [He *et al.*, 2021]) on WebQSP test data with 2,848 (question, answer) training instances. Since 1,409 (question, answer) pairs in the training data of WebQSP overlap with those in WQ, we can just take their corresponding subgraphs from WQ. With these subgraphs, KBQG models can produce their corresponding questions. To evaluate the quality of the augmented questions, we conducted experiments using data augmented by both baselines and R^{2}DQG, as well as the original WebQSP data (referred to as ORI). We denote the new datasets as “ORI + baseline” and “ORI + R^{2}DQG,” respectively. On these augmented datasets, we train GRAFT-Net and NSM separately on “ORI + baseline,” “ORI + R^{2}DQG,” and the original WebQSP data (ORI) to compare their performance. The observations from Table 3 are as follows: (1) The generated additional QA data pairs can be viewed as an approach to data augmentation for KBQA, as both GRAFT-Net and NSM trained on datasets augmented by different KBQG models can enhance their QA performance. (2) We notice that QA models trained on datasets augmented by R^{2}DQG obtain the best results, indicating that the question generation synthetic datasets significantly enhance downstream QA tasks.

Method	GRAFT-Net		NSM	
	F1	Hits@1	F1	Hits@1
ORI	61.80	67.32	67.11	73.52
Augmented by G2S+AE+RL	62.06	67.48	67.14	73.58
Augmented by SGSH	62.58	68.05	67.85	74.26
Augmented by DiversifyQG	63.17	68.55	68.12	74.83
Augmented by RoleAgentQG	62.04	67.62	67.65	73.70
Augmented by R^{2}DQG	**64.22**	**69.75**	**69.26**	**76.12**

Table 3: Downstream QA performance on the augmented QA dataset.

4.6 Human Evaluation

To systemically assess our R^{2}DQG, human evaluation is recruited to investigate the generated question. We followed the existing work to select three evaluation dimensions: Fluency (whether a question has no grammatical errors and is fluent in expression), Relevance (whether a question clearly describes the given subgraphs) and Diversity (whether a question express the same semantics with ground truth, but in different expression forms). We randomly sample 100 instances from the testing set of the WQ dataset, and collected the generated results from R^{2}DQG and several competitive baselines for comparison. Three master’s students were recruited as annotators to rate each question in a blind setup. Each question was rated independently on a Likert scale ranging from 1 to 5, with 5 indicating the highest quality. We calculated Kendall’s tau coefficient to assess how closely the annotators agreed. The resulting value of 0.83 demonstrates a strong level of agreement among them. We average the scores from raters on each question. As shown in Table 4, our R^{2}DQG consistently achieves the best performance and produces questions that are more fluent, diverse, and relevant. Particularly, we observe that R^{2}DQG surpasses baselines on diversity evaluation, which shows that our model equipped with the template generator can produce more diversified questions. This is consistent with the observations from our main results.

Method	Top-3 Questions			Top-5 Questions		
	Fluency	Relevance	Diversity	Fluency	Relevance	Diversity
DiversifyQG	3.98	3.86	4.11	3.93	3.81	4.17
RoleAgentQG	4.01	3.99	4.08	4.01	3.96	4.07
R^{2}DQG	**4.14**	**4.06**	**4.21**	**4.07**	**4.12**	**4.28**

Table 4: Human evaluation results on WQ dataset (Top-3 and Top-5 generated questions).

5 Conclusion

Recent advancements in KBQG have primarily focused on improving question quality, while the challenge of achieving diversity remains largely unexplored. We argue that KBQG inherently follows a one-to-many mapping paradigm, making diverse yet relevant question generation essential. To address this, we introduced R^{2}DQG, a framework that combines template-guided generation with a drift compensation mechanism to produce high-quality and diverse questions. Our approach utilizes diversified templates to guide question structure and employs drift compensation to refine biased drafts into coherent and contextually relevant questions without human intervention. Moreover, R^{2}DQG serves as an effective data augmentation tool, enhancing the robustness of QA models. Extensive experiments on the WQ and PQ datasets demonstrate the effectiveness of our approach in balancing quality and diversity. To address the current limitations in handling large and complex subgraphs in KBQG, we leave this challenge to future work, where we intend to incorporate GNN-based techniques [Lu *et al.*, 2024] for more effective subgraph representation and question generation. In addition, we will focus on extending the framework to more complex domains and exploring adaptive strategies to further enhance question generation.

Acknowledgements

The research was supported in part by the National Natural Science Foundation of China (No.U22B2019) and the China Scholarship Council program (Project ID: 202406470057). We thank the anonymous reviewers for their comments.

References

[Agrawal *et al.*, 2024] Garima Agrawal, Kuntal Pal, Yuli Deng, Huan Liu, and Ying-Chih Chen. Cyberq: Generating questions and answers for cybersecurity education using knowledge graph-augmented llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23164–23172, 2024.

[Banerjee and Lavie, 2005] Satanjeev Banerjee and Alon Lavie. 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, 2005.

[Bi *et al.*, 2020] Sheng Bi, Xiya Cheng, Yuan-Fang Li, Yongzhen Wang, and Guilin Qi. Knowledge-enriched, type-constrained and grammar-guided question generation over knowledge bases. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2776–2786, 2020.

[Cao and Wang, 2021] Shuyang Cao and Lu Wang. 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, 2021.

[Chen *et al.*, 2023] Yu Chen, Lingfei Wu, and Mohammed J Zaki. Toward subgraph-guided knowledge graph question generation with graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[Dang *et al.*, 2024a] Jisheng Dang, Huicheng Zheng, Xiaohao Xu, Longguang Wang, and Yulan Guo. Beyond appearance: Multi-frame spatio-temporal context memory networks for efficient and robust video object segmentation. *IEEE Transactions on Image Processing*, 2024.

[Dang *et al.*, 2024b] Jisheng Dang, Huicheng Zheng, Xiaohao Xu, Longguang Wang, Qingyong Hu, and Yulan Guo. Adaptive sparse memory networks for efficient and robust video object segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

[Deschamps *et al.*, 2021] Arthur Deschamps, Sujatha Das Gollapalli, and See Kiong Ng. On generating fact-infused question variations. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 335–345, 2021.

[Elsahar *et al.*, 2018] Hady Elsahar, Christophe Gravier, and Frederique Laforest. Zero-shot question generation from knowledge graphs for unseen predicates and entity types. In *Proceedings of NAACL-HLT*, pages 218–228, 2018.

[Fei *et al.*, 2022] Zichu Fei, Xin Zhou, Tao Gui, Qi Zhang, and Xuan-Jing Huang. Lfkqg: A controlled generation framework with local fine-tuning for question generation over knowledge bases. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6575–6585, 2022.

[Guo *et al.*, 2022] Shasha Guo, Jing Zhang, Yanling Wang, Qianyi Zhang, Cuiping Li, and Hong Chen. Dsm: Question generation over knowledge base via modeling diverse subgraphs with meta-learner. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4194–4207, 2022.

[Guo *et al.*, 2024a] Shasha Guo, Lizi Liao, Cuiping Li, and Tat-Seng Chua. A survey on neural question generation: Methods, applications, and prospects. *arXiv preprint arXiv:2402.18267*, 2024.

[Guo *et al.*, 2024b] Shasha Guo, Lizi Liao, Jing Zhang, Yanling Wang, Cuiping Li, and Hong Chen. Sgsh: Stimulate large language models with skeleton heuristics for knowledge base question generation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4613–4625, 2024.

[Guo *et al.*, 2024c] Shasha Guo, Jing Zhang, Xirui Ke, Cuiping Li, and Hong Chen. Diversifying question generation over knowledge base via external natural questions. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5096–5108, 2024.

[He *et al.*, 2021] Gaole He, Yunshi Lan, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *Proceedings of the 14th ACM international conference on web search and data mining*, pages 553–561, 2021.

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

[Hu *et al.*, 2021] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021.

[Jia and Liang, 2016] Robin Jia and Percy Liang. Data recombination for neural semantic parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12–22, 2016.

[Ke *et al.*, 2021] Pei Ke, Haozhe Ji, Yu Ran, Xin Cui, Liwei Wang, Linfeng Song, Xiaoyan Zhu, and Minlie Huang. Jointgt: Graph-text joint representation learning for text generation from knowledge graphs. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2526–2538, 2021.

[Kim *et al.*, 2019] Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. Improving neural question generation using answer separation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6602–6609, 2019.

[Kumar *et al.*, 2019] Vishwajeet Kumar, Yuncheng Hua, Ganesh Ramakrishnan, Guilin Qi, Lianli Gao, and Yuan-

Fang Li. Difficulty-controllable multi-hop question generation from knowledge graphs. In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part I 18*, pages 382–398. Springer, 2019.

[Li *et al.*, 2024] Jinhong Li, Xuejie Zhang, Jin Wang, and Xiaobing Zhou. Deep question generation model based on dual attention guidance. *International Journal of Machine Learning and Cybernetics*, pages 1–11, 2024.

[Liang *et al.*, 2023] Yuanyuan Liang, Jianing Wang, Hanlun Zhu, Lei Wang, Weining Qian, and Yunshi Lan. Prompting large language models with chain-of-thought for few-shot knowledge base question generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4329–4343, 2023.

[Lu *et al.*, 2024] Kangkang Lu, Yanhua Yu, Hao Fei, Xuan Li, Zixuan Yang, Zirui Guo, Meiyu Liang, Mengran Yin, and Tat-Seng Chua. Improving expressive power of spectral graph neural networks with eigenvalue correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14158–14166, 2024.

[Lyu *et al.*, 2021] Chenyang Lyu, Lifeng Shang, Yvette Graham, Jennifer Foster, Xin Jiang, and Qun Liu. Improving unsupervised question answering via summarization-informed question generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4134–4148, 2021.

[Martins *et al.*, 2020] Pedro Henrique Martins, Zita Marinho, and André FT Martins. Sparse text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4252–4273, 2020.

[Narayan *et al.*, 2022] Shashi Narayan, Gonçalo Simões, Yao Zhao, Joshua Maynez, Dipanjan Das, Michael Collins, and Mirella Lapata. A well-composed text is half done! composition sampling for diverse conditional generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1319–1339, 2022.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 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, 2002.

[Qi *et al.*, 2020] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, 2020.

[Quan *et al.*, 2019] Zhe Quan, Zhi-Jie Wang, Yuquan Le, Bin Yao, Kenli Li, and Jian Yin. An efficient framework for sentence similarity modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(4):853–865, 2019.

[Seyler *et al.*, 2017] Dominic Seyler, Mohamed Yahya, and Klaus Berberich. Knowledge questions from knowledge graphs. In *Proceedings of the ACM SIGIR international conference on theory of information retrieval*, pages 11–18, 2017.

[Song *et al.*, 2019] Haoyu Song, Weinan Zhang, Yiming Cui, Dong Wang, and Ting Liu. Exploiting persona information for diverse generation of conversational responses. pages 5190–5196, 08 2019.

[Sultan *et al.*, 2020] Md Arafat Sultan, Shubham Chandel, Ramón Fernandez Astudillo, and Vittorio Castelli. On the importance of diversity in question generation for qa. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5651–5656, 2020.

[Sun *et al.*, 2018] Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4231–4242, 2018.

[Wang *et al.*, 2020] Zhen Wang, Siwei Rao, Jie Zhang, Zhen Qin, Guangjian Tian, and Jun Wang. Diversify question generation with continuous content selectors and question type modeling. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2134–2143, 2020.

[Wang *et al.*, 2024] Yuling Wang, Changxin Tian, Binbin Hu, Yanhua Yu, Ziqi Liu, Zhiqiang Zhang, Jun Zhou, Liang Pang, and Xiao Wang. Can small language models be good reasoners for sequential recommendation? In *Proceedings of the ACM Web Conference 2024*, pages 3876–3887, 2024.

[Zhang and Zhu, 2021] Zhiling Zhang and Kenny Zhu. Diverse and specific clarification question generation with keywords. In *Proceedings of the web conference 2021*, pages 3501–3511, 2021.

[Zhao *et al.*, 2024] Runhao Zhao, Jiuyang Tang, Weixin Zeng, Ziyang Chen, and Xiang Zhao. Zero-shot knowledge graph question generation via multi-agent llms and small models synthesis. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 3341–3351, 2024.

[Zhou *et al.*, 2018] Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. An interpretable reasoning network for multi-relation question answering. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2010–2022, 2018.