Evidence-Focused Fact Summarization for Knowledge-Augmented Zero-Shot Question Answering

Sungho Ko[*], **Hyunjin Cho**[*], **Hyungjoo Chae, Jinyoung Yeo, Dongha Lee**[†]
Yonsei University
{k133324,cyberhyunjin,mapoout,jinyeo,donalee}@yonsei.ac.kr

Abstract

Recent studies have investigated utilizing Knowledge Graphs (KGs) to enhance Question Answering (QA) performance of Large Language Models (LLMs), yet structured KG verbalization remains challenging. Existing methods, such as *triple-form* or *free-form* textual conversion of triple-form facts, encounter several issues. These include reduced evidence density due to duplicated entities or relationships, and reduced evidence clarity due to an inability to emphasize crucial evidence. To address these issues, we propose EFSUM, an **E**vidence-focused **F**act **Sum**marization framework for enhanced QA with knowledge-augmented LLMs. We optimize an open-source LLM as a fact summarizer through distillation and preference alignment. Our extensive experiments show that EFSUM improves LLM's zero-shot QA performance, and it is possible to ensure both the helpfulness and faithfulness of the summary.

1 Introduction

Large Language Models (LLMs) have shown remarkable zero-shot abilities but often produce factual errors, known as *hallucinations*, particularly in knowledge-intensive tasks like Question Answering (QA). This happens because the static knowledge within LLM parameters may be incomplete, incorrect, or outdated, failing to keep pace with evolving real-world knowledge. Recent studies remedy this by integrating external knowledge into LLMs (Karpukhin et al., 2020b; Min et al., 2019).

As one form of external knowledge, Knowledge Graphs (KGs) have been considered as the knowledge source to augment LLMs for enhanced performance in knowledge graph QA (KGQA) (Baek et al., 2023a; Wu et al., 2023; Sen et al., 2023). The key challenge of utilizing KGs, which consist of a set of (head entity, relation, tail entity) triples,
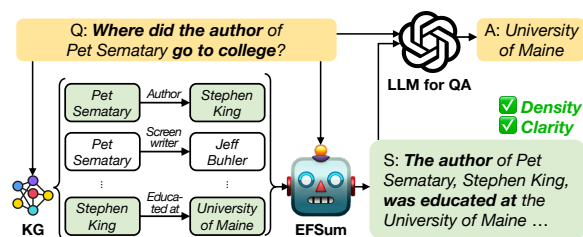

Figure 1: The QA pipeline based on LLM prompting, augmented with relevant facts from KGs. Our fact summarization improves both density and clarity of evidence within contextual knowledge for enhanced QA.

is to bridge the modality gap between graphs and text. Efforts on connecting the gap mostly fall into either training additional layers to blend two representations, or verbalizing graphs into texts. While training fusion layers (Yasunaga et al., 2022a,b) can make models expressive on two different modalities, it takes expensive computaions, and needs to be trained when KGs are updated. On the other hand, recent studies proposed verbalizing KGs into text-form, without training LLMs for QA. For instance, one strategy is simply concatenating the facts in their *triple-form* text (Baek et al., 2023a), and another entails converting them into semantically-coherent textual description (i.e., *free-form* text) through the distillation of LLM's ability to generate text from KGs (Wu et al., 2023).

Despite their remarkable efficacy, existing verbalization strategies for providing contextual knowledge exhibit critical limitations. **(1) Low density of evidence**: Both concatenation and linearization of the facts (Baek et al., 2023a; Wu et al., 2023) are highly likely to include duplicated entities or relations due to limited flexibility; this eventually degrades the density of useful evidence within the contextual knowledge, hindering the ability to answer questions effectively. **(2) Low clarity of evidence**: While contextual knowledge describes factual information, they often fail to highlight the evidence necessary for answering ques-

---

[*]Equal contribution
[†]Corresponding author

10636

tions. This lack of focus can lead to noise from irrelevant facts, which can detrimentally impact the LLM's ability to provide accurate answers.

To address the aforementioned challenges, we introduce a novel **EFSUM** framework, **E**vidence-focused **F**act **Sum**marization for enhanced QA with knowledge-augmented LLM prompting. The key idea is transforming the set of facts into plausible and coherent *summary* while highlighting evidence and filtering out noise given a question; this ensures that the summaries maintain a high density and clarity of evidence, facilitating effective QA (Figure 1). The most straightforward solution to this summarization is prompting LLMs with the detailed instruction. However, LLM-generated summaries often omit crucial evidence (such as answer spans), resulting in *information loss*, or include information that cannot be inferred from retrieved facts, leading to *extrinsic hallucination*.

For enhanced summary quality, we optimize an open-source LLM as a fact summarizer in two steps: LLM distillation and preference alignment (Figure 3). During the first step, we train our fact summarizer by using the reference summaries obtained through LLM prompting. Subsequently, in the second step, we refine our summarizer to better align with the task-specific preference related to QA. To this end, we introduce two preference criteria for the summary candidates: *helpfulness* evaluates LLMs can correctly answer the question based on the summary, and *faithfulness* assesses the factual consistency of the summary in relation to the provided set of facts. By selecting pairs of preferred and dispreferred summaries based on these criteria, we further fine-tune the summarizer through direct preference optimization (DPO) (Rafailov et al., 2023). In the end, EFSUM is capable of generating summaries that are both helpful for QA and faithful to the given facts.

Extensive experiments on two QA benchmark datasets validate the effectiveness of our evidence-focused fact summarization in improving LLM's zero-shot QA performance. EFSUM outperforms other fact verbalization methods in two key settings: (1) when fixing the token length (density, Section 4.2), and (2) when fixing the number of triples (clarity, Section 4.3) within contextual knowledge. Additionally, our approach enhances the helpfulness and faithfulness of the generated fact summaries. For reproducibility, our codes are publicly available at https://github.com/kk13332488/EFSum.

2 Preliminaries

In this section, we introduce a KG-augmented zero-shot QA pipeline, and provide analyses on the verbalized facts obtained by existing methods.

2.1 KG-Augmented LLM Prompting for QA

We focus on a QA approach that leverages LLMs' zero-shot capability for answering the question, enhanced with external knowledge from KGs.

Fact retrieval from knowledge graph. The first stage aims to retrieve question-associated facts from KGs via entity linking, and then to select only top-K ones based on their semantic relevance to the input question. To select only the most relevant facts, recent studies utilize semantic similarities between each fact and the question, employing either a pretrained sentence encoder (Karpukhin et al., 2020a; Xiong et al., 2020) or one fine-tuned specifically for direct fact retrieval (Baek et al., 2023b). In this work, we utilized the former strategy (Song et al., 2020), if not stated.

Fact verbalization into various form text. The second stage is fact verbalization, which refers to the task of transforming symbolic facts into textual strings, for feeding them into the LLM as the contextual knowledge. The linear verbalization simply concatenates the head, relation, and tail texts in the triple while keeping the structured format (i.e., triple-form text) (Baek et al., 2023a), or use manually-designed templates and heuristics for linearization (Oguz et al., 2022; Ma et al., 2022). On the other hand, the graph-to-text verbalization transforms the input facts into the plausible and coherent text by using a fine-tuned model (Ribeiro et al., 2021) or prompting LLMs (Wu et al., 2023).

Fact injection for question answering. The last stage is prompting the LLM to generate the answer with the verbalized facts. This process, also known as knowledge-augmented LLM prompting for zero-shot QA, gathers the output as the predicted answer. To handle with insufficient evidence of knowledge, the detailed instructions are provided to allow the LLM to utilize its internal knowledge if needed. The prompts are in Appendix D.2.

2.2 Analysis on Verbalized Facts

We provide a preliminary analysis of the contextual knowledge obtained by each fact verbalization method, evaluating their (1) density and (2) clarity of evidence. Note that our proposed verbaliza-

The ratio of **duplicated tokens** within the verbalized facts

The length ratio of **original facts** to the verbalized facts

Average position of the gold answer within the verbalized facts

Semantic similarity of the question to the verbalized facts

KAPING KG2Text Rewrite

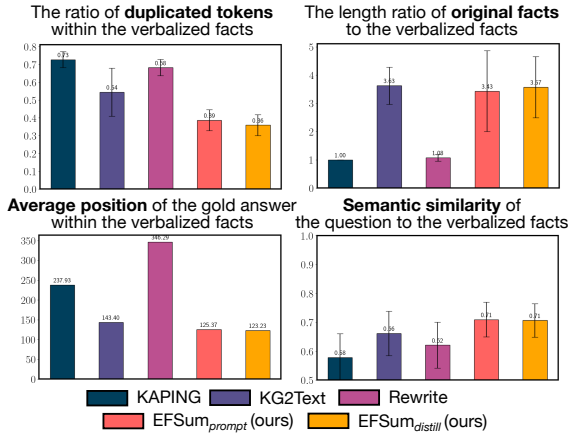EFSum$_{prompt}$ (ours) EFSum$_{distill}$ (ours)

Figure 2: Analysis on each fact verbalization method.

tion method (denoted as EFSUM) summarizes the symbolic facts into free-form texts, prioritizing the evidence relevant to the question.

In Figure 2 Upper, it is evident that the number of duplicated tokens is significantly higher in the linearly verbalized texts (i.e., KAPING (Baek et al., 2023a) and Rewrite (Wu et al., 2023)) compared to the others. This strongly indicates that their outputs contain redundant information stemming from the pre-defined relations within KGs. When considering the ratio of token lengths before and after verbalization (i.e., compression rate), KAPING and Rewrite either maintain or even increase the length, despite maintaining the same amount of information. Consequently, their evidence density remains low or may even decrease during verbalization.

In Figure 2 Lower, the linear verbalization tends to scatter obvious evidence (i.e., answer span) randomly within the contextual knowledge. Their placement seems to rely on the rank obtained from fact retrieval, rather than being positioned at the forefront of the verbalized texts for emphasis. Moreover, the average semantic similarity [1] of their verbalized facts with the question is lower compared to other methods. This shows that their outputs may include noisy or irrelevant information, or they might not clearly highlight semantically relevant evidence. In essence, clarity of evidence is not adequately addressed in this approach.

3 EFSUM: Proposed Method

In this section, we present an **E**vidence-focused **F**act **Sum**marization framework, named **EFSUM**,

which aims to effectively augment the set of relevant facts to LLMs for zero-shot QA tasks.

3.1 LLM Prompting for EFSUM

The most straightforward implementation of evidence-focused fact summarization is prompting LLMs to generate a summary s given a question q and its relevant facts $\mathcal{F} = \{f_k\}_{k=1}^K$, focusing on the evidence to answer the question. Thus, we present **EFSUM**$_{prompt}$ to verbalize the facts with the help of LLM's zero-shot capability on summarization; that is, $s \sim p_{\text{LLM}}(\cdot | t_{sum}, q, \mathcal{F})$, where t_{sum} is the prompt for summarization. Specifically, we instruct LLMs to turn the input facts into the summary for the scenarios where the summary serves as a context to facilitate QA task. We utilized GPT-3.5-turbo for generating summary of EFSUM$_{prompt}$. The detailed prompts are in Appendix D.2.

3.2 LLM Fine-Tuning for EFSUM

To enhance the quality of summaries, we propose **EFSUM**$_{distill}$, a fact verbalization model based on an open-source LLM,[2] specifically fine-tuned for evidence-focused fact summarization. For helpful and faithful summary generation, EFSUM$_{distill}$ is optimized in two steps: (1) LLM distillation and (2) preference alignment. Figure 3 illustrates the overall EFSUM$_{distill}$ framework.

3.2.1 Distillation of Fact Summarization

We first optimize our open-source model to generate diverse evidence-focused summaries given the question and the set of symbolic facts. For optimization, we augment the QA training dataset of question-answer pairs (q, a) into $\mathcal{X} = \{(q, a, \mathcal{F})\}$, where $\mathcal{F} = \{f_k\}_{k=1}^K$ is the top-K retrieved facts relevant to the question q.

Reference summary generation. We utilize a closed-source LLM (i.e., GPT-3.5-turbo) to obtain reference fact summaries used for training our summarizer. For each tuple $(q, a, \mathcal{F})$ of a question q and the relevant facts $\mathcal{F}$, we prompt the LLM to transform the set of facts $\mathcal{F}$ into a concise textual description s that highlights the evidence for the question q, which is same with EFSUM$_{prompt}$. In the end, we construct the training dataset $\mathcal{D} = \{(q, a, \mathcal{F}, s)\}$. The prompt used for summary generation is in Table 5 of Appendix D.2.

---

[1]Average semantic similarity is calculated through the cosine similarity between embeddings encoded via MPNet.

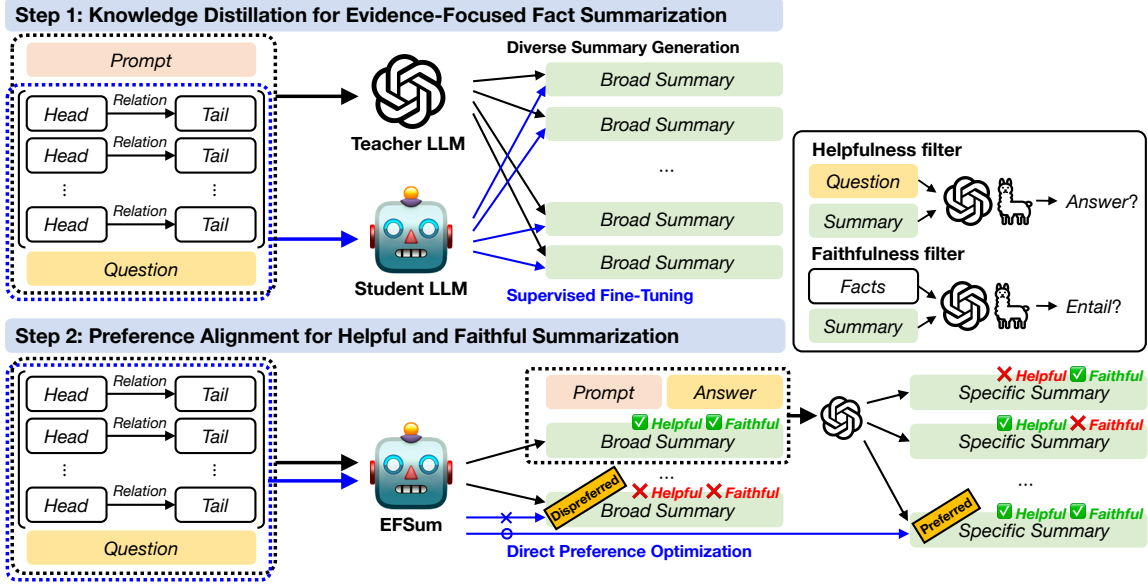[2]In this work, we choose Llama2-7B (Touvron et al., 2023) as the backbone open-source LLM for EFSUM$_{distill}$.

Figure 3: The overall framework of EFSUM$_{distill}$. Our fact summarizer is trained to generate evidence-focused summaries via LLM distillation, and then further optimized to align the QA-specific preference, which enhances the helpfulness and faithfulness of its output summaries.

Supervised fine-tuning. For each quadruplet $(q, \mathcal{F}, s, a)$ from the dataset $\mathcal{D}$, our summarizer θ is optimized to generate s conditioned on q and $\mathcal{F}$, by using the causal language modeling objective:

$$\mathcal{L}_{\text{SFT}} = - \mathop{\mathbb{E}}_{(q,a,\mathcal{F},s)\sim D} \log p_\theta(s|q, \mathcal{F}). \quad (1)$$

3.2.2 Alignment with Summary Preference

Our summarizer θ is now able to generate evidence-focused summaries for a question and the relevant facts, but its output summaries might be unhelpful or unfaithful. Therefore, we additionally adopt preference tuning to enhance EFSUM$_{distill}$ so that its summarization can align with the task-specific preference in the context of knowledge-augmented zero-shot QA; i.e., generating helpful and faithful summaries, while avoiding the counterparts.

For preference tuning, we collect a set of preference pairs $(q, a, \mathcal{F}, s^+, s^-)$, where s^+ and s^- respectively denote the *preferred* and *dispreferred* summaries. To identify the preferred and dispreferred summary for QA task, we first sample M summary candidates $\{s'_m \sim p_\theta(\cdot|q, \mathcal{F})\}_{m=1}^M$ for each tuple $(q, a, \mathcal{F}, s) \in \mathcal{D}$ using the summarizer θ. Then, for constructing the preference pairs, we adopt two summary filters (for checking helpfulness and faithfulness) and additional paraphrase process. We form a preference pair by selecting the answer-aware paraphrased candidate that passes both filter as preferred, and the candidate that cannot pass one of the filters as dispreferred.

Helpfulness filter. The first filter examines the helpfulness of each summary candidate in terms of QA accuracy. That is, it checks whether the summary candidate s' is actually helpful to make the LLM to find the correct answer, by comparing the LLM's generated answer $a' \sim p_{\text{LLM}}(\cdot|q, s')$ with the gold answer a.

Faithfulness filter. The second filter focuses on the faithfulness of each summary candidate. We use the G-Eval approach (Liu et al., 2023) to leverage LLM's ability to evaluate the consistency between the input facts and the given summary in terms of hallucination. Precisely, the LLM is prompted to examine whether or not the summary contains unfaithful information, which cannot be inferred from the given symbolic facts. Please refer to the detailed prompt in Appendix D.2.

Broad-to-specific paraphrasing. In addition, we adopt the paraphrase process guided by the gold answer a. This paraphrasing aims to obtain high-quality summary by refining a broad focus into a specific focus, using the given answer as the main evidence. We prompt the LLM to paraphrase the summary candidate s' into s''; $\{s''_m \sim p_{\text{LLM}}(\cdot|t_{\text{paraphrase}}, s', a)\}_{m=1}^M$, where $t_{\text{paraphrase}}$ is the prompt for paraphrasing. This candidate s'' undergoes another pass through the helpfulness filter and the faithfulness filter. The resulting summary obtained in this manner is much more focused on

the QA task than the initially obtained reference summary, called as the broad summary. The detailed prompt is provided in Appendix D.2

Direct Preference Optimization. Using the preference pairs $\mathcal{P} = \{(q, a, \mathcal{F}, s^+, s^-)\}$, we apply Direct Preference Optimization (DPO) (Rafailov et al., 2023) on our summarizer θ to train a preference-tuned summarizer θ^* that minimizes the following objective:

$$\mathcal{L}_{\text{DPO}}(\theta^*; \theta) = \qquad\qquad\qquad (2)$$
$$- \underset{(q,a,\mathcal{F},s^+,s^-)\sim P}{\mathbb{E}} \log \sigma[r(q, \mathcal{F}, s^+) - r(q, \mathcal{F}, s^-)],$$

where $r(q, \mathcal{F}, s) = \frac{p_{\theta^*}(s|q,\mathcal{F})}{p_\theta(s|q,\mathcal{F})}$. By optimizing the model using preferred-dispreferred summary pairs, our obtained model θ^* is trained to be more biased towards helpful and faithful summary s^+ while avoiding unhelpful or unfaithful summary s^-. Note that θ has been specifically trained for each QA model, as there are different preferences regarding summary helpfulness across various QA models. The following is an example of an output fact summary.

Question: where was george washington carver from?
Answer: Diamond
Facts: (George Washington Carver, occupation, biologist),(George Washington Carver, interested in, botany),(George Washington Carver, occupation, university teacher),(George Washington Carver, place of birth, Diamond),(George Washington Carver, given name, George), (George Washington Carver, field of work, pedagogy),(George Washington Carver, relative, Moses Carver),(George Washington Carver, place of birth, United States of America),(George Washington Carver, residence, Tuskegee),(George Washington Carver, occupation, inventor)
Summary: George Washington Carver, the renowned biologist, was born in Diamond, United States of America. He took great interest in botany and became a university teacher, focusing on pedagogy. Additionally, Carver contributed significantly as an inventor. His relative, Moses Carver, also had a close association with him. Carver later resided in Tuskegee.

---

4 Experiments

In this section, we design our experiments to answer the following research questions:

- **RQ1:** Does a high density of evidence in verbalized facts contribute to QA accuracy?
- **RQ2:** Does a high clarity of evidence in verbalized facts contribute to QA accuracy?
- **RQ3:** Can preference alignment enhance the generation of more helpful and faithful summaries?

4.1 Experimental Settings

Datasets. WebQuestionsSP (WebQSP) (Yih et al., 2016) is a KGQA dataset that filters questions from the WebQuestions (Berant et al., 2013) dataset to include only those answerable via Freebase, and provides SPARQL queries for them. For convenience, we use WebQSP-WD (Sorokin and Gurevych, 2018), in which each question from WebQSP is pre-linked to the Wikidata KG. We use a test set comprising 1,033 examples for evaluation. **Mintaka** (Sen et al., 2022) is a QA dataset that encompasses eight different complexity types. Most question-answer pairs can only be solved by utilizing multi-hop reasoning or the attributes of multiple entities. We use a test set of Mintaka which has 4,000 examples for evaluation.

LLMs for zero-shot QA. To measure the efficacy of EFSUM and other fact verbalization methods, we utilized three different LLMs, **GPT-3.5-turbo**, **Flan-T5-XL** (Chung et al., 2022), and **Llama2-7B-Chat** (Touvron et al., 2023), for our zero-shot QA evaluation. Note that Flan-T5-XL and Llama2-7B-Chat are publicly available as open-source. We provide more details on the models in Appendix A.1.

Baseline methods. As the main baselines for fact verbalization, we consider various approaches.

- **No knowledge** does not pass any knowledge contexts, encouraging the LLMs to use their internal knowledge to answer the question.
- **KAPING** (Baek et al., 2023a) simply linearizes top-K relevant facts as the triple-form text. Triple-form text simply refers to the text that is composed by concatenating triplet strings in the form of (head, relation, tail).
- **Rewrite** (Wu et al., 2023) transforms facts into the free-form text for each relation path with a LLM. We utilize GPT-3.5-turbo to convert triples into free-form text.
- **KG2Text** (Ribeiro et al., 2021) employs an encoder-decoder model fined-tuned for the KG-to-text task by using WebNLG (Gardent et al., 2017) dataset. We utilize a fine-tuned T5-large model[3] as our base KG2Text model.

Evaluation metrics. Our task can be categorized as a generative KGQA. Following previous work (Baek et al., 2023a; Wu et al., 2023), we use accuracy as our evaluation metric. A score of 1

---
[3]https://public.ukp.informatik.tu-darmstadt.de/ribeiro/graph2text/webnlg-t5-large.ckpt

Dataset	Methods	GPT-3.5-turbo		Flan-T5-XL		Llama2-7B-Chat	
		L=200	L=400	L=200	L=400	L=200	L=400
WebQSP	No knowledge	0.506	0.506	0.409	0.409	0.539	0.539
	KAPING (Baek et al., 2023a)	0.507	<u>0.538</u>	0.391	0.439	**0.517**	**0.519**
	KG2Text (Ribeiro et al., 2021)	0.476	0.476	0.316	0.321	0.439	0.481
	Rewrite (Wu et al., 2023)	0.444	0.525	0.350	0.431	0.462	<u>0.511</u>
	EFSUM$_{prompt}$ (Ours)	<u>0.537</u>	<u>0.538</u>	<u>0.447</u>	<u>0.468</u>	0.457	0.491
	EFSUM$_{distill}$ (Ours)	**0.559**	**0.569**	**0.458**	**0.500**	<u>0.489</u>	0.497
Mintaka	No knowledge	0.540	0.540	0.228	0.228	0.440	0.440
	KAPING (Baek et al., 2023a)	**0.539**	**0.539**	0.269	0.279	0.402	<u>0.407</u>
	KG2Text (Ribeiro et al., 2021)	0.492	0.491	0.234	0.234	0.377	0.378
	Rewrite (Wu et al., 2023)	<u>0.515</u>	<u>0.521</u>	0.280	0.288	0.394	0.386
	EFSUM$_{prompt}$ (Ours)	0.496	0.491	<u>0.312</u>	<u>0.321</u>	**0.423**	**0.418**
	EFSUM$_{distill}$ (Ours)	0.474	0.449	**0.326**	**0.338**	<u>0.405</u>	0.406

Table 1: QA accuracy of the LLMs based on various fact verbalization. We limit the maximum token length of contextual knowledge to $L = 200$ and 400. The best and second-best results are in **bold** and <u>underlined</u>, respectively.

Datasets	Methods	GPT-3.5-turbo			Flan-T5-XL			Llama2-7B-Chat		
		Random	Popular	MPNet	Random	Popular	MPNet	Random	Popular	MPNet
WebQSP	No knowledge	0.506	0.506	0.506	0.409	0.409	0.409	0.539	0.539	0.539
	KAPING (Baek et al., 2023a)	0.441	0.437	<u>0.538</u>	0.297	0.329	0.439	<u>0.476</u>	**0.490**	**0.519**
	KG2Text (Ribeiro et al., 2021)	0.469	0.468	0.476	0.317	0.276	0.321	0.465	0.451	0.481
	Rewrite (Wu et al., 2023)	0.473	0.445	0.525	0.323	0.348	0.431	0.458	0.439	<u>0.511</u>
	EFSUM$_{prompt}$ (Ours)	**0.542**	<u>0.534</u>	<u>0.538</u>	<u>0.443</u>	<u>0.442</u>	<u>0.468</u>	**0.477**	0.472	0.491
	EFSUM$_{distill}$ (Ours)	<u>0.475</u>	**0.539**	**0.569**	**0.500**	**0.505**	**0.500**	0.457	<u>0.488</u>	0.497
Mintaka	No knowledge	0.540	0.540	0.540	0.228	0.228	0.228	0.440	0.440	0.440
	KAPING (Baek et al., 2023a)	**0.553**	<u>0.516</u>	**0.539**	0.201	0.198	0.279	<u>0.417</u>	**0.398**	<u>0.407</u>
	KG2Text (Ribeiro et al., 2021)	0.505	0.500	0.492	0.220	<u>0.235</u>	0.234	**0.421**	0.389	0.378
	Rewrite (Wu et al., 2023)	<u>0.527</u>	**0.524**	<u>0.515</u>	<u>0.230</u>	0.224	0.288	0.393	0.374	0.386
	EFSUM$_{prompt}$ (Ours)	0.454	0.492	0.496	0.213	0.215	<u>0.321</u>	0.390	0.392	**0.418**
	EFSUM$_{distill}$ (Ours)	0.427	0.425	0.474	**0.292**	**0.243**	**0.338**	0.397	<u>0.393</u>	0.406

Table 2: QA accuracy of the LLMs based on various fact verbalization, with different fact retrieval strategies (i.e., random facts, popular facts, and question-relevant facts). We limit the maximum token length of contextual knowledge to $L = 400$. The best and second-best results are in **bold** and <u>underlined</u>, respectively.

is assigned if at least one among multiple correct answers is present in the response text of the QA model; otherwise, the score is 0.

Relevant fact retrieval. In retrieving question-related facts, as described in the Section 2.1, we employ MPNet (Song et al., 2020) to retrieve only the top-K triples among given KG, with the highest semantic similarity to the question representation, following Baek et al. (2023a). Since processing entire KGs is impractical, we focus on retrieving information from the n-hop neighbors of question entities within a KG. The value of n is given by the specific KGQA dataset to answer the question. In our experiment, n is set to 1 for WebQSP and 2 for Mintaka. When calculating semantic similarity, we use the linear verbalization approach, which

involves combining the subject, relation, and object texts from the triple.

4.2 Effectiveness of Dense Evidence (RQ1)

To examine the impact of dense evidence within verbalized facts on the final QA performance, we evaluate the LLM's QA accuracy while imposing restrictions on the maximum token lengths L of contextual knowledge. This implies that the number of facts included in the contextual knowledge varies depending on fact verbalization methods.

Effect of knowledge augmentation. First of all, we observe that knowledge augmentation for zero-shot QA does not always produce positive results. Knowledge augmentation cannot be helpful in two scenarios: where the model's ability to ground in-

Dataset	Methods	GPT-3.5-turbo		Flan-T5-XL		Llama2-7B-Chat	
		K=10	K=30	K=10	K=30	K=10	K=30
WebQSP	No knowledge	0.617	0.607	0.498	0.451	0.646	0.628
	KAPING (Baek et al., 2023a)	0.777	<u>0.771</u>	<u>0.643</u>	<u>0.738</u>	**0.668**	**0.699**
	KG2Text (Ribeiro et al., 2021)	0.589	0.608	0.467	0.457	0.409	0.536
	Rewrite (Wu et al., 2023)	0.628	0.728	0.533	0.664	0.594	<u>0.688</u>
	EFSUM$_{prompt}$ (Ours)	**0.788**	0.755	0.629	0.711	0.497	0.571
	EFSUM$_{distill}$ (Ours)	<u>0.786</u>	**0.783**	**0.644**	**0.741**	<u>0.599</u>	0.666
Mintaka	No knowledge	0.810	0.788	0.492	0.444	0.783	0.719
	KAPING (Baek et al., 2023a)	<u>0.912</u>	0.869	0.723	0.673	0.832	<u>0.808</u>
	KG2Text (Ribeiro et al., 2021)	0.879	0.768	0.536	0.491	0.799	0.727
	Rewrite (Wu et al., 2023)	0.901	<u>0.875</u>	0.720	0.691	0.843	0.792
	EFSUM$_{prompt}$ (Ours)	**0.920**	**0.887**	<u>0.742</u>	**0.735**	<u>0.849</u>	0.806
	EFSUM$_{distill}$ (Ours)	0.893	0.824	**0.745**	<u>0.719</u>	**0.852**	**0.826**

Table 3: QA accuracy of the LLMs based on fact verbalization methods. We fix the number of facts to $K = 10$ and 30. The best and second-best results are in **bold** and <u>underlined</u>, respectively.

put knowledge is lacking, or where the retrieved knowledge is noisy while the QA model's internal knowledge is sufficient. In Table 1, when Llama2-7B-Chat was used as the QA model, it demonstrates higher performance under the "No knowledge" condition across both datasets (i.e., WebQSP and Mintaka) compared to other baselines. This is indicative of Llama2-7B-Chat's limited capability in utilizing the provided knowledge.

Comparison with other baselines. We first compare the performance of various fact verbalization methods at $L = 200$ and 400. In Table 1, for most cases, EFSUM shows superior performance over the majority of baseline approaches. Considering the ratio of lengths before/after verbalization (in Figure 2), this clearly indicates that EFSUM can encapsulate more intensive and useful information within shorter summaries. Notably, the effectiveness of our approach is more evident when the length of knowledge decreases from $L = 400$ to 200. This suggests that EFSUM remains highly effective even in contexts when the utilization of extremely concise knowledge is required.

Compatibility with various retrievers. We investigate the QA performance using different types of fact retrieval, to assess the robustness across various knowledge qualities: randomly selected knowledge (**Random**), the knowledge possessing the most frequently occurring relation (**Popular**), and question-relevant knowledge (**MPNet**). In Table 2, EFSUM achieves the highest accuracy across most datasets and QA models, regardless of the retriever used, which implies that our method can extract useful facts from noisy input triples.
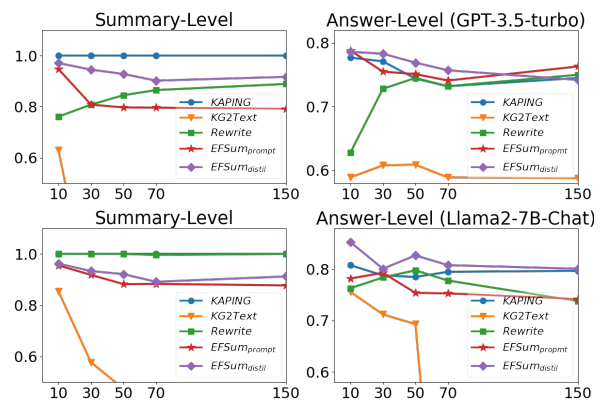

Figure 4: Summary-level and answer-level QA accuracies with respect to the number of relevant facts on WebQSP (Upper) and Mintaka (Lower), respectively.

4.3 Effectiveness of Clear Evidence (RQ2)

To examine the effectiveness of clear evidence within verbalized facts on the final QA performance, we assess the LLM's QA accuracy only on the test tuples $(q, a, \mathcal{F})$ where the facts $\mathcal{F}$ fully contain the ground-truth answer span a. This experimental setup allows us to investigate how effectively each fact verbalization method converts $\mathcal{F}$ into a textual string without overlooking evidence a, thereby enabling correct answers.

Comparison with other baselines. To examine the effectiveness of EFSUM in terms of evidence clarity, we evaluate the LLM's zero-shot QA accuracy when the number of facts is $K = 10$ and 30. In Table 3, EFSUM shows the highest performance across most datasets, showing consistently leading results for both K values. Nevertheless, we notice the presence of uncontrolled model inclination. For
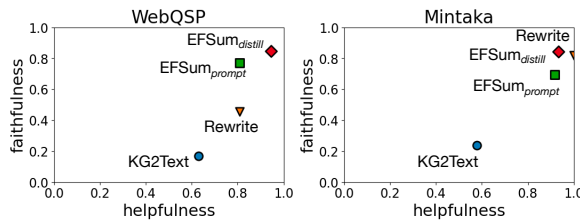
10642

Figure 5: Two quality metrics of verbalized facts.

example, KAPING consistently exhibits the highest performance when Llama2-7B-Chat is utilized as the QA model for the WebQSP dataset. This might be due to the model's inclination towards a specific knowledge format.

Robustness of EFSUM across various K. To assess the robustness of EFSUM, we increase the retrieved number of facts K from 10 to 150 and measure the performance of each baseline at the *answer-level* and *summary-level*. At the *answer-level*, we evaluate the extent to which the answers are contained within the responses generated by the QA model using summaries. In case of the *summary-level*, we evaluate whether the answer is included within the verbalized knowledge produced by the method. In Figure 4, across both datasets and the majority of K values, EFSUM$_{distill}$ consistently outperforms, especially in answer-level accuracy, except for KAPING, which inherently achieves a summary level accuracy of 1 by simple linearization of facts.

4.4 Effect of Preference Alignment (RQ3)

Helpfulness and faithfulness. We examine the *helpfulness* and *faithfulness* of the verbalized facts generated by each verbalization method.[4] Note that helpfulness is calculated through summary-level accuracy, while faithfulness is calculated as 1 - (hallucination occurrence rate). That is, a method with a lower rate of hallucination occurrences possesses higher faithfulness. Figure 5 illustrates that EF-SUM$_{distill}$ achieves superior helpfulness and faithfulness compared to other baselines, including EF-SUM$_{prompt}$. In other words, EFSUM$_{distill}$ is less likely to generate summaries of hallucination and is capable of incorporating more correct answers into the summaries compared to other baselines. Moreover, it validates that using the two filters in the broad-to-specific paraphrasing process definitely aids in improving both faithfulness and helpfulness.

---

[4] We opt not to include KAPING in the plot, as its linear verbalization does not alter the content of the facts.

5 Related Work

5.1 KG-Augmented LLM Prompting

To supplement the incomplete internal knowledge of LLMs, recent research has been exploring knowledge augmentation methods using prompting. Various types of information can be utilized for prompting knowledge, notably KGs. Recently, Baek et al. (2023a); Wu et al. (2023) utilize triple-form facts from KGs for knowledge augmentation for LLM-prompting. However, the knowledge injected into LLMs by these methods is not focused on the question. In other words, the semantics of the question are not considered during the process of deriving the final knowledge from the given triples. Consequently, the final knowledge derived through each method may inevitably have low density and clarity from an evidence perspective.

5.2 Knowledge Graph Question Answering

Studies exploring the integration of additional knowledge extracted from KGs, represented as subgraphs, are divided into two main approaches. The first approach utilizes semantic parsing techniques (Bao et al., 2016; Luo et al., 2018), which enable the extraction of executable queries from the KG by using contextual information as a parsing reference. Alternatively, the information retrieval approach involves encoding assimilated information using techniques like Graph Neural Networks (GNNs). Several recent studies (Yasunaga et al., 2022a,b; Zhang et al., 2022) propose a learning framework that combines GNNs and LMs, allowing concurrent utilization of textual data and KG. Besides, approaches that project text embeddings to graph embeddings (Razzhigaev et al., 2023) struggle to avoid hallucinations by LLMs. In contrast to the conventional KGQA approaches that aim to directly identify the answer entity within KGs, in this work, we mainly focus on how the KG-retrieved facts can be effectively utilized within the zero-shot QA capability of LLMs.

6 Conclusion

In this paper, we explore methods to enhance the zero-shot QA performance of LLMs by augmenting knowledge from KGs. We introduce a novel summarization framework, called EFSUM, which transforms a set of facts into summary with high density and clarity of evidence for answering the question. To achieve this, we optimize an

open-source LLM as a fact summarizer, leveraging a teacher LLM's summarization capabilities and aligning its outputs with QA-specific preferences. Our experiments show that EFSUM significantly improve QA accuracy across various LLMs compared to other fact verbalization approaches. Furthermore, serving as an independent summarization module, it generates helpful and faithful summaries based on relevant facts and target questions.

7 Limitation

Despite our discoveries and improvements, we must acknowledge certain limitations in our work and potential areas for future research.

To begin with, the accuracy, which is the metric used in our experiments has the potential to overestimate the correctness of responses, even if the response does not accurately convey the intended semantic meaning. This discrepancy can occur because the metric simply verifies the existence of the answer entity, regardless of whether it is contextually appropriate. Unlike semantic parsing KGQA, which involves retrieving entities from the KG, or multiple-choice KGQA, where the answer is chosen from several options, evaluating metrics for generative KGQA remains an open field that warrants further investigation.

Secondly, there are cases where the tendency of LLMs to favor a certain fact verbalization method becomes overwhelming and difficult to manage. As demonstrated in Tables 2 and 3, with the example of Llama2-7B-Chat on the WebQSP dataset, it has been observed that certain models may have a inclination for specific knowledge formats in particular datasets. Consequently, while our summarizer generally demonstrates good performance, controlling performance may become challenging when a specific model has a strong inclination towards a particular knowledge format.

Lastly, it is important to note that the performance of our proposed summarizer can be influenced by the performance of the retriever. As can be seen in Table 2, using a better retriever can lead to higher performance. While the off-the-shelf model we used (i.e. MPNet) demonstrates retrieving capabilities based on the semantic similarity between questions and facts, it's difficult to assert that it is a flawless retriever. For example, in a 2-hop dataset(i.e. Mintaka), it tends to retrieve 1-hop neighbors more than 2-hops even if it is irrelevant. This is because an answer entity in 2-hop neighbors is unseen in given question, so that a retrieval model may measure question entities in 1-hop neighbors more similar. Therefore, we are currently conducting further research to propose not only a more powerful summarizer but also a more flawless retrieving method simultaneously.

8 Ethical Consideration

Throughout our research, we thoroughly explore our methodology using an open-source dataset, chosen to ensure transparency and integrity in our work. It is important to acknowledge the inherent potential for biases within our summarizer, which relies on LLMs, and which may inadvertently reflect prevailing social biases. It is crucial to notice that our method is not intended to inflict harm upon any individuals or groups.

9 Acknowledgements

References

Jinheon Baek, Alham Aji, and Amir Saffari. 2023a. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023)*, pages 70–98, Toronto, ON, Canada. Association for Computational Linguistics.

Jinheon Baek, Alham Fikri Aji, Jens Lehmann, and Sung Ju Hwang. 2023b. Direct fact retrieval from knowledge graphs without entity linking. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10038–10055, Toronto, Canada. Association for Computational Linguistics.

Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. 2016. Constraint-based question answering with knowledge graph. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2503–2514, Osaka, Japan. The COLING 2016 Organizing Committee.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Filip Graliński, Anna Wróblewska, Tomasz Stanisławek, Kamil Grabowski, and Tomasz Górecki. 2019. GEval: Tool for debugging NLP datasets and models. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 254–262, Florence, Italy. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020a. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Yu Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. 2020b. Dense passage retrieval for open-domain question answering. *ArXiv*, abs/2004.04906.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2185–2194, Brussels, Belgium. Association for Computational Linguistics.

Kaixin Ma, Hao Cheng, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2022. Open domain question answering with a unified knowledge interface. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1605–1620.

Sewon Min, Danqi Chen, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Knowledge guided text retrieval and reading for open domain question answering. *ArXiv*, abs/1911.03868.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. UniK-QA: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1535–1546, Seattle, United States. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Anton Razzhigaev, Mikhail Salnikov, Valentin Malykh, Pavel Braslavski, and Alexander Panchenko. 2023. A system for answering simple questions in multiple languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 524–537, Toronto, Canada. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2021. Investigating pretrained language models for graph-to-text generation. In *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*, pages 211–227, Online. Association for Computational Linguistics.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1604–1619, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Priyanka Sen, Sandeep Mavadia, and Amir Saffari. 2023. Knowledge graph-augmented language models for complex question answering. In *Proceedings of the 1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pages 1–8, Toronto, Canada. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867.

Daniil Sorokin and Iryna Gurevych. 2018. Modeling semantics with gated graph neural networks for knowledge base question answering. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3306–3317, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti

Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yike Wu, Nan Hu, Guilin Qi, Sheng Bi, Jie Ren, Anhuan Xie, and Wei Song. 2023. Retrieve-rewrite-answer: A kg-to-text enhanced llms framework for knowledge graph question answering. *arXiv preprint arXiv:2309.11206*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D Manning, Percy Liang, and Jure Leskovec. 2022b. Deep bidirectional language-knowledge graph pretraining.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2022a. Qa-gnn: Reasoning with language models and knowledge graphs for question answering.

Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, Berlin, Germany. Association for Computational Linguistics.

Xikun Zhang, Antoine Bosselut, Michihiro Yasunaga, Hongyu Ren, Percy Liang, Christopher D Manning, and Jure Leskovec. 2022. Greaselm: Graph reasoning enhanced language models for question answering. *arXiv preprint arXiv:2201.08860*.

A Implementation Details

A.1 LLMs for Zero-Shot QA

- **Flan-T5** is a variant model grounded in the T5 architecture. T5, an encoder-decoder model, is trained on a text-to-text dataset featuring a diverse array of objectives. Flan-T5 represents an evolution of T5 through the process of instruction tuning, enhancing its performance by aligning it more closely with specific instructional contexts. We employ Flan-T5-XL for our QA model.

- **Llama2-7B-Chat** Llama2 is the developed version of Llama1(Touvron et al., 2023). For the purpose of achieving more accurate measurement of summarizer performance, we employ the Llama2-7B-chat, which is specifically optimized for conversational contexts.

- **GPT-3.5-turbo**, developed by OpenAI, stands as a prominent closed-source model renowned for its high performance, making it particularly well-suited for application in QA models. Due to GPT-3.5-turbo being closed-source, we utilize the API provided by OpenAI[5].

A.2 Faithfulness Evaluation

To achieve a more faithful summary, we undertake distillation using data meticulously purged of summaries affected by hallucinations. For this purpose, we eliminate hallucinations from both broad and specific summaries based on prompts provided by G-eval (Graliński et al., 2019). We revise the G-eval-provided prompts with QA-specific instructions and generated auto Chain of Thought (CoT). This auto-generated CoT are then utilized as evaluation criteria to ascertain the factual consistency between the generated summaries and the facts, marking them as true or false. For a more accurate assessment of faithfulness, we measure hallucination using GPT-4, a more advanced model than GPT-3.5-turbo, which was used for reference summary generation and paraphrasing. The faithfulness of the method is defined as (1 -hallucination occurrence rate).

A.3 Reference Summary Generation

For the purpose of generating reference summaries conducive to distillation, we utilize GPT-3.5-turbo. Given the scarcity of training data available within each dataset, we embark on a strategy of data augmentation to distill a more diverse array of summary cases. Through GPT-3.5-turbo, five summaries are augmented for each sample, and to foster an even greater diversity of summary instances, we adjust the decoding temperature, to 1.1.

A.4 Paraphrased Summary Generation

For generating specific summaries for DPO training, we conduct paraphrasing. Summary from the reference summaries that passed through the helpfulness and faithfulness filters is transformed from broader summary to specific summary through paraphrasing. For this paraphrasing endeavor, GPT-3.5-turbo is once again employed. To select the most effective summary among a variety of paraphrased summaries, multiple paraphrased summaries are generated for each sample. To ensure diversity, the decoding temperature is set to 1.1.

---

[5]https://openai.com/api/

A.5 Distilled Summary Generation

In the generation of the final summary through the EFSUM, the temperature is set to 0.1. While it is appropriate to elevate the temperature to consider a wider range of candidates during the creation of a augmented dataset for distillation, we conclude that for generating summaries for QA inference, it is more important to promote consistency over diversity in the summaries.

B Experimental Settings

B.1 Datasets

- **WebQuestionsSP (WebQSP)** (Berant et al., 2013; Yih et al., 2016) Semantic Parses Dataset, abbreviated to WebQSP, is a KGQA dataset providing SPARQL queries which allows a direct retrieval from Freebase KG. Due to the cessation of updates to Freebase we adopt Wikidata as the foundational KG. As we mentioned in section 4.1, we utilized WebQSP-WD (Sorokin and Gurevych, 2018), a dataset that offers questions from WebQSP pre-linked to the Wikidata KG.

- **Mintaka** (Sen et al., 2022) A KGQA dataset collected from Wikidata, with 8 complexity types, including 'Count', 'Comparative', 'Superlative', 'Ordinal', 'Multi-hop', 'Intersection', 'Difference', 'Yes/No', 'Generic'. Question-Answer pairs are collected from Wikidata entities. This dataset is multilingual, and we use English datasets in this work.

B.2 Various Retrievers

- **Random Knowledge** is a knowledge augmentation method that entails the selection of K arbitrary facts to serve as the final knowledge.

- **Popular Knowledge** employs a method wherein triples are organized based on the releation frequency of their occurrence among one-hop neighbor triples. Then top K triples, sorted by the prevalence of their relations, are utilized as the final knowledge.

- **MPNet** is a retrieval method predicated on the semantic similarity between questions and triples. Each triple in the triple set is compared to the question based on the cosine similarity between their MPNet representations. The K triples exhibiting the highest similarity to the

Dataset	Method	Acc
WebQSP	**EFSUM**$_{distill}$	**0.500**
	w/o Paraphrase	0.477
	w/o Helpfulness	0.453
	w/o All Filters, Parapharase	0.469
Mintaka	**EFSUM**$_{distill}$	**0.338**
	w/o Paraphrase	0.296
	w/o Helpfulness	0.289
	w/o All Filters, Parapharase	0.299

Table 4: Ablation study on different filters and paraphrasing. The best and second-best results are in **bold** and underlined, respectively.

question are then selected as the final knowledge.

C Additional Experiment Results

C.1 Ablation Study

To investigate the effect of different filters and paraphrasing approach on EFSUM, in Table 4, we evaluate the performance of EFSUM on ablation study. For the experimental setting, we used Flan-T5-XL as a QA model on WebQSP and Mintaka with a maximum token length of contextual knowledge as $L = 400$. Each row indicates whether each filter or paraphrasing approach is used or not. 'w/o All Filters, Paraphrase' means the model is trained without using any filtering or paraphrasing technique. Table 4 demonstrates that the performance of the distilled model drops filters are removed or the paraphrasing stage is omitted. This result indicates that EFSUM shows the best performance when all processes are put together. When the helpfulness filter is removed, some summaries that do not aid the QA task get mixed into the chosen set during DPO training. This disrupts the model's optimization process, leading to a performance decline. Consequently, removing the helpfulness filter results in the most critical performance drop. The paraphrasing stage also plays a crucial role. Without paraphrasing, the ability to highlight important information within paragraphs decreases. As a result, key information in the summary becomes distracted, making it difficult for the QA model to digest the knowledge. This inevitably leads to a performance decline.

C.2 Generalization Ability

We examine the generalization capability of EFSUM for unseen datasets. In Table 5, We investigate the cross-dataset experiments on Flan-T5-XL with a maximum token length of contextual knowledge as $L = 400$ (i.e. compatible to Ta-

Method	Trained on	Generate Summaries on	Acc
$\text{EFSUM}_{distill}$	Mintaka	WebQSP	0.455
$\text{EFSUM}_{distill}$	WebQSP	Mintaka	0.281
KAPING	WebQSP	WebQSP	0.439
Rewrite	Mintaka	Mintaka	0.288

Table 5: Generalization capability of $\text{EFSUM}_{distill}$ for unseen datasets.

ble 1). We use $\text{EFSUM}_{distill}$ that generates the summary for Mintaka trained on WebQSP, also vice versa. Referring to Table 1, the most superior baselines scores 0.439 and 0.288 for WebQSP and Mintaka, respectively. And $\text{EFSUM}_{distill}$ that trained on unseen dataset scores 0.455 and 0.281 for WebQSP and Mintaka, respectively. This result shows that our methods outperform the most of baselines and are competitive to the most effective baselines. This indicates that the proposed evidence-focused approach can effectively summarize the evidence even on unseen datasets.

D Qualitative Examples

D.1 Case Study

In Table 6, we present an example of verbalized facts generated by each method. EFSUM emphasizes evidence necessary to answer the question (i.e., *Emilio Estevez's father is Martin Sheen.*) at the beginning, while excluding irrelevant details for brevity, as opposed to other baselines. Duplicated relations (i.e., relations of *siblings*) are aggregated, and it is clear that our method is more compact than KAPING and Rewrite. Comparing to KG2Text, it also sums up the given triples, but reduces a loss of essential information.

D.2 Prompts

We provide various LLM prompts, used for (1) evidence-focused summarization, (2) knowledge-augmented zero-shot QA, (3) paraphrase of summary candidates, and (4) faithfulness evaluation.

Question: Who is Emilio Estevez's father?
Answer: Martin Sheen

KAPING: (Emilio Estevez's, family name, Estévez), (Emilio Estevez's, given name, Emilio), (Emilio Estevez's, father, Martin Sheen), (Emilio Estevez's, occupation, actor), (Joe Estevez, relative, Emilio Estevez), (Ramón Estévez, sibling, Emilio Estevez), (Emilio Estevez's, relative, Joe Estevez), (Ramón Estévez, father, Martin Sheen), (Martin Sheen, child, Emilio Estevez), (Emilio Estevez's, date of birth, +1962-05-12T00:00:00Z), (Joe Estevez, relative, Ramón Estévez), (Emilio Estevez's, birth name, Emilio Estevez Sheen), (Emilio Estevez's, sibling, Ramón Estévez), (Ramón Estévez, relative, Joe Estevez), (Ramón Estévez, family name, Estévez), (Emilio Estevez's, occupation, film producer), (Renée Estévez, sibling, Emilio Estevez), (Charlie Sheen, sibling, Emilio Estevez), (Janet Sheen, child, Emilio Estevez), (Martin Sheen, child, Ramón Estévez), (Joe Estevez, family name, Estevez), (Martin Sheen, family name, Estévez), (Emilio Estevez's, described by source, Obalky knih.cz), (Ramón Estévez, occupation, television actor), (Emilio Estevez's, occupation, film director), (Emilio Estevez's, country of citizenship, United States of America), (Emilio Estevez's, occupation, writer), (Estévez, said to be the same as, Estevez), (Emilio Estevez's, sibling, Renée Estévez), (Emilio Estevez's, place of birth, New York City)

KG2Text the family name of emilio estevez is estévez . he was born in new york city and is from the united states . he is the father of two children , both of whom are children . he is the actor , writer and film director . he is the brother of charlie sheen and his sister is renée .

Rewrite: Emilio Estevez's family name is Estévez. Emilio Estevez's given name is Emilio. Emilio Estevez's father is Martin Sheen. Emilio Estevez is an actor. Joe Estevez is a relative of Emilio Estevez. Ramón Estévez is a sibling of Emilio Estevez. Emilio Estevez is related to Joe Estevez as they are relatives. Ramón Estévez is the father of Martin Sheen. Martin Sheen is the father of Emilio Estevez. Joe Estevez is a relative of Ramón Estévez. Emilio Estevez has a sibling named Ramón Estévez. Ramón Estévez is a relative of Joe Estevez. Ramón Estévez's family name is Estévez. Emilio Estevez is a film producer. Renée Estévez is the sibling of Emilio Estevez. Charlie Sheen is the sibling of Emilio Estevez. Janet Sheen is the mother of Emilio Estevez. Martin Sheen's child is Ramón Estévez. Joe Estevez's family name is Estevez. Martin Sheen's family name is Estévez. Emilio Estevez is described by the source Obalky knih.cz. Ramón Estévez is a television actor. Emilio Estevez is a film director. Emilio Estevez is a citizen of the United States of America. Emilio Estevez is a writer. Estévez is said to be the same as Estevez. Emilio Estevez has a sibling named Renée Estévez. Emilio Estevez was born in New York City. Ramón Estévez is an actor. Charlie Sheen's family name is Estévez.

EFSUM Emilio Estevez's father is Martin Sheen. He was born in New York City in 1962 and is known for his work as an actor, film director, and writer. He has siblings named Ramón Estévez, Joe Estevez, Renée Estévez, and Charlie Sheen. Martin Sheen is the father of both Emilio Estevez and Ramón Estévez.

Table 6: An example of verbalized facts for Mintaka.

Evidence-Focused Summarization Prompt (EFSUM$_{prompt}$)

[Task Description]
You are a knowledge graph summarizer for Question Answering. I will give you "Question", "Fact triples". You should turn triples into *summary*. The *summary* should serve as a context to facilitate QA (Question and Answer) tasks.
Caution1: The *summary* should not explicitly mention what the correct answer is.
Caution2: The *summary* should only contain information of the given triples.
Caution3: Each triplet is seperated with "\n" and head, relation, tail are provided in head | relation | tail format.
Question: {}
Fact triples: {}
Summary:

Table 7: The prompt for evidence-focused summarization.

KG-Augmented Question Answering Prompt (KAPING)
[Task Description]
You are a student who have to solve the question. I'll give you a triples as a context. But if it is not useful, just ignore it and generate your own guess.
Triples: {}
Question: {}
You are aware of the answer. Generate only short answer(You have to guess something):

KG-Augmented Question Answering Prompt (EFSUM)
[Task Description]
You are a student who have to solve the question. I'll give you a summary as a context. But if it is not useful, just ignore it and generate your own guess.
Summary: {}
Question: {}
You are aware of the answer. Generate only short answer(You have to guess something):

Table 8: The prompts for knowledge-augmented zero-shot question answering.

Summary Candidate Paraphrasing Prompt (In case that answer does not exist in the summary candidate)
[Task Description]
You are a knowledge graph summarizer for Question Answering. I will give you "Question", "Knowledge Summary". You should pharaphrase the original "Knowledge Summary". The paraphrased summary should serve as a context to facilitate QA (Question and Answer) tasks. Paraphrase the original "Knowledge Summary" to be more helpful to solve the QA.
Question: {}
Original Summary: {}
Paraphrased Summary:

Summary Candidate Paraphrasing Prompt (In case that answer does exist in the summary candidate)
[Task Description]
You are a knowledge graph summarizer for Question Answering. I will give you "Question", "Answer", "Knowledge Summary". You should pharaphrase the original "Knowledge Summary". The paraphrased summary should serve as a context to facilitate QA (Question and Answer) tasks. Paraphrase the original "Knowledge Summary" to be more helpful to solve the QA.
Question: {}
Answer: {}
Original Summary: {}
Paraphrased Summary:

Table 9: The prompt for summary candidate paraphrasing.

Summary Candidate Faithfulness Evaluation Prompt

[Task Description]
You will be given one summary written to provide useful contexts by given source triples from knowledge graphs. Your task is to check whether the given summary induces factual inconsistency. Please make sure you read and understand these instructions carefully. Please keep this evaluation creteria open while reviewing, and refer to it as needed.

Evaluation Criteria:
Factual Inconsistency (0 or 1): Does the summary untruthful or misleading facts that are not supported by the source triples? If does, mark 1. Otherwise, mark 0.

Evaluation Steps:
1. read and understand the source triples first. note the entities that are in focus and the relations between them.
2. proceed to read through the summary provided.
3. compare the information in the summary with that in the source triples. pay particular attention to the entities, actions, and relations.
4. mark "1" if the summary contains factual inconsistencies, i.e., if it states untruthful or misleading facts that are not supported by the source triples.
5. mark "0" if the summary is consistent with the source triples and does not misrepresent the facts provided by the source triples.

Remember, you are not assessing the quality of the writing, but the factual consistency of the summary compared to the source triples. perfection in grammar or style does not account for factual consistency. conversely, poor grammar or style does not necessarily mean factual inconsistency. the key lies in the alignment of facts between the source triples and the summary.

Source Triples: {}
Summary: {}
Does the summary contain factual inconsistency? Answer:

Table 10: The prompt for faithfulness evaluation.