

Retrieval Augmented Generation with Rich Answer Encoding

Wenyu Huang¹, Mirella Lapata¹, Pavlos Vougiouklis²,
Nikos Papasrantopoulos^{3*}, Jeff Z. Pan^{1,2†}

¹School of Informatics, University of Edinburgh

²Huawei Edinburgh Research Centre, CSI, Edinburgh, UK

³Priceline, Edinburgh, UK

w.huang@ed.ac.uk mlap@inf.ed.ac.uk pavlos.vougiouklis@huawei.com
nikos.papasa@gmail.com <https://knowledge-representation.org/j.z.pan/>

Abstract

Knowledge-intensive generation tasks like generative question answering require models to retrieve appropriate passages from external knowledge sources to support answer generation. The generation quality relies heavily on the retrieved passages, which serve as contextual information. State-of-the-art Retrieval Augmented Generation models with marginalized output dominate this area but focus too much on label-relevant passages, rather than question-relevant passages and answers. This work addresses this issue by incorporating rich answer encoding through Dense Knowledge Similarity (DKS) and Retriever as Answer Classifier (RAC). We demonstrate the advantages of our proposed approach in open domain question answering (MSMARCO) and conversation (Wizard of Wikipedia) datasets, reporting both generation and retrieval metrics. In the MSMARCO development set, our best model achieves 12.1% relative improvement¹ on Recall@1 and 4.5% relative improvement on BLEU-4 compared to the baseline model. In the KILT-WoW leaderboard, our best model achieves 8.9% relative improvement on R-Precision and 13.3% relative improvement on KILT-RL compared to the baseline model. Our codes and models are available at <https://github.com/hwy9855/rag-ae>.

1 Introduction

Knowledge-Intensive Generation (KIG) is a series of tasks that requires external knowledge sources, such as a passage corpus (focus of this paper) or a knowledge graph (Pan et al., 2017) to generate natural language responses to questions. Without external knowledge, even the state-of-the-art Large Language Models (LLMs) like GPT-4 (OpenAI, 2023) still suffer from the hallucination problem

* Work done while at Huawei Edinburgh Research Centre.

† Corresponding author.

¹In this paper, we mainly report relative improvement for better comparison between different methods.

Question: Definition of tactful personality	
Reference Answer (label): <i>Tactful is someone or something that shows a regard for other people's feelings.</i>	
Passage 1 (label-relevant): <i>The definition of tactful is someone or something that shows a regard for other people's feelings. An example of something that would be described as tactful is a suggestion that is worded very carefully so as not to offend.</i>	RAG ✓ ours ✓
Passage 2 (knowledge-relevant): ... tactful(adj) having or showing a sense of what is fitting and considerate in dealing with others. she was tactful enough not to shatter his illusion; ...	RAG ✗ ours ✓

Table 1: Retrieval samples of MSMARCO training set. Both models are trained with the given input (first row). The RAG model only captures the label-relevant passage with high token matching (highlighted with *italics*) in the reference answer, while our approach can also capture the knowledge-relevant passage with knowledge matching (highlighted with **bold**).

(Pan et al., 2023a; Zhang et al., 2023) that produces plausible-looking statements that are factually incorrect. State-of-the-art works focus on the retrieval augmented generation (RAG) systems (Karpukhin et al., 2020; Izacard and Grave, 2021), which fit in a retrieve-generate architecture, where the models first retrieve question-relevant passages from external knowledge sources, then generate responses based on the retrieved passages.

A high-quality retriever is essential for accomplishing KIG tasks. In the era of LLMs, a high-quality off-the-shelf retriever is also important in verifying and correcting the factual errors produced by LLMs (He et al., 2023; Zhao et al., 2023) However, directly training the retriever (Karpukhin et al., 2020; Khattab and Zaharia, 2020) requires a large number of annotations of gold passages according to the given questions. Current state-of-the-art works (Lewis et al., 2020b; Paranjape et al., 2022) provide an end-to-end training framework for this task, where they marginalize the retrieval step and use the natural language labels to guide the

training of both retriever and generator. The gradient of both the retriever and generator is obtained from token matches with reference answers (labels). However, existing works prefer more label-relevant passages (a subset of question-relevant passages), this will result in missing some knowledge-relevant passages which limits the generalisability of the retriever. See the example presented in Table 1. Given the question, the RAG framework can only retrieve the Passage 1, due to label-relevance, but not the Passage 2, which is also highly relevant to the reference answer. Using limited gold label passages when training a retrieve-generate model harms the robustness resulting in overfitting the retrieval performance to the training data (as shown in Table 5).

To mitigate the above issue, in this work, we propose a new framework for extending RAG with rich answer encoding, based on knowledge relevant to answers. More precisely, we introduce the following two objectives: **Retriever as Answer Classifier (RAC)**, and **Dense Knowledge Similarity (DKS)**, for training the retriever to retrieve knowledge-relevant passages. RAC incorporates answer encoding to check whether the retrieved passage contains the knowledge inside the answer, where both passages in Table 1 are positive as they both contain knowledge inside the answer. DKS incorporates both answer encoding and passage encoding to check how close the knowledge inside the answer and that inside the passage are in the knowledge representation² space. The answer encoding and passage encoding in DKS guarantees that knowledge-relevant passages (both passages in Table 1) are much closer to the label (than other passages) in the knowledge representation space.

Both proposed objectives focus on sequence similarity (by answer encoding) instead of the token-level similarity from the backbone architecture (NLL objective), which fundamentally turns the objective from token matching to sequence (knowledge) matching. This allows the proposed loss functions to capture knowledge-level information (instead of token-level information) from answer labels. This knowledge-level information can assist with training the retriever to recall more knowledge-relevant passages (instead of only label-relevant

²In this paper, we use the term "knowledge representation" since the learned representation is for representing the knowledge inside the answer and passage, which has a different meaning from "knowledge representation" in the knowledge graph area.

passages) and improve the performance and generalisation ability of the end-to-end trained retrievers.

Our main contribution is a new training framework for retrieve-generate models that 1) offers two new objectives, RAC and DKS, incorporating rich answer encoding, and 2) paying more attention to knowledge-relevant passages with the two new objectives to train more robust retrieve-generate models that generalize better in retrieval. We evaluate the proposed framework in two KIG tasks, which are the generative QA task on MS-MARCO (Nguyen et al., 2016) and the informative conversation task on Wizard of Wikipedia (Dinan et al., 2019) organized by KILT benchmark (Petroni et al., 2021). Compared with the baseline model RAG (Lewis et al., 2020b), we get substantial improvement on both datasets. For the retrieval quality, our best model achieves 12.1% relative improvement (Recall@1) on MSMARCO and 8.9% relative improvement (R-Prec) on KILT-WoW. While for end-to-end generation quality, our best model achieves 4.5% relative improvement (BLEU-4) on MSMARCO and 13.3% relative improvement (KILT-RL) on KILT-WoW. The reliable improvement in the retrieval quality also indicates the potential contribution of the proposed work in the era of LLM for training a powerful and robust retriever for supporting LLMs.

2 Related Works

Knowledge-intensive Generation Knowledge-intensive generation tasks are a series of Natural Language Generation (NLG) tasks that require access to large, external knowledge sources. Unlike other knowledge-intensive NLP tasks such as fact checking (Thorne et al., 2018) and slot filling (Levy et al., 2017; Elshahar et al., 2018), knowledge-intensive generation tasks further require models to generate natural language response, which are more challenging. Shuster et al. (2021) has shown that in knowledge-intensive generation tasks, pre-trained language models like BART (Lewis et al., 2020a), T5 (Raffel et al., 2020) and GPT series (Brown et al., 2020; OpenAI, 2023) significantly suffer from the hallucination problem (Roller et al., 2021), where they generate plausible looking statements that are factually incorrect.

Retrieval Marginalization In retrieval-based tasks, the false-negative passage problem refers to passage labels not being fully annotated. Ni et al. (2021) has shown that the false-negative pas-

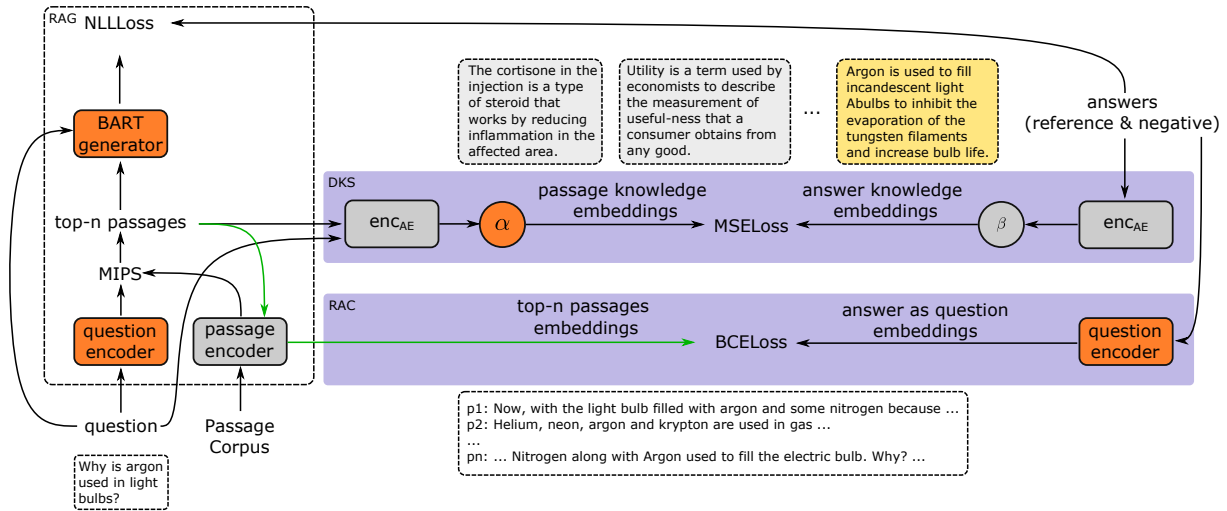


Figure 1: Overall architecture of the proposed model. MIPS indicates Maximum Inner Product Search. Green arrows indicate a different information stream with black arrows. Orange components are trainable while gray components are not trainable. The dashed box indicates the retrieve-generate backbone (Lewis et al., 2020b). Violet boxes indicate the two objectives introduced by this work. The proposed DKS and RAC aim to help the retriever focus more on knowledge-relevant passages instead of only label-relevant passages.

sage problem happens very frequently in multi-document question tasks, with such cases appearing in more than half of the sampled answerable questions of the IIRC dataset (Ferguson et al., 2020). Research has been made to mitigate this problem by marginalizing the retrieval process and directly training the retriever with the final goal, e.g., answer label in multi-document QA (Ni et al., 2021). However, in knowledge-intensive generation tasks, the marginalization methods do not work so well. Lewis et al. (2020b) reported that such marginalization methods get very limited improvement in benefiting the retrieval quality on the generative QA task.

Retrieval Augmented Generation To overcome the hallucination problem, people start to introduce retrieve-generate architectures for building retrieval augmented generation models. RAG (Lewis et al., 2020b) outperforms DPR (Karpukhin et al., 2020) by marginalizing the retrieval step to train the generator and retriever jointly with the supervision of the label answer. FiD (Izacard and Grave, 2021) encodes the concatenation of the passages retrieved by pre-trained DPR and the original question separately, and then fuses them with concatenation to the decoder. KG-FiD (Yu et al., 2022) utilize knowledge graphs to further enhance the retrieval quality by establishing the structural relationship among the retrieved passages. This illustrates the concept of integrating knowledge graphs with re-

trieval augmented generation, but indirectly (i.e., the generator does not benefit from knowledge graphs). KGI (Chowdhury et al., 2022) provides a robust implementation of RAG, where the retriever is trained jointly with both the RAG setting and the vanilla DPR setting. RE2G (Glass et al., 2022) further extends RAG by adding a reranker in the retrieval step to integrate statistical retriever (e.g., TF-IDF and BM25) with DPR trained by RAG to improve the retrieval quality. Hindsight (Paranjape et al., 2022) trains another hindsight retriever that takes inputs of both queries and labels to retrieve label-relevant passages. The original retriever is trained by Evidence Lower Bound (ELBo) which includes both the marginalized loss as in RAG and the KL divergence for fitting the hindsight retriever. The state-of-the-art methods tend to retrieve label-relevant passages, whereas our approach seeks to consider a more diverse set of passages as suitable (knowledge-wise).

Retrieval Augmented Generation with LLMs

The utilization of retrieval augmented generation can be further expanded within the context of Large Language Models (LLMs) (Pan et al., 2023a), to address a critical concern associated with LLMs: the hallucination problem. He et al. (2023) use retrieved passages to help LLMs "rethink" the question to remove factual errors in the decomposed reasoning steps obtained from the chain-of-thought (CoT) prompting (Wei et al., 2022). Zhao et al.

(2023) utilize retrieved passages to post-edit the decomposed reasoning steps which contain factual errors obtained from the CoT prompting, to increase the prediction factuality. Besides retrieving documents (passages), Baek et al. (2023) shows that an off-the-shelf document retriever can also be used to retrieve knowledge graph triples for supporting LLMs. Most of these works utilize off-the-shelf retrievers, which emphasize the importance of powerful and robust retrievers. Their remarkable achievements underscore the central theme of our work, which focuses on training powerful and robust retrievers.

3 Problem Statement

A typical Knowledge-Intensive Generation (KIG) task can be formulated as follows: 1) given a question q and a knowledge source (passage corpus) \mathcal{P} , retrieve question-relevant passages $\mathcal{P}_q \subset \mathcal{P}$; 2) conditioned on the question q and the retrieved question-relevant passages \mathcal{P}_q , generate the natural language response y . Since \mathcal{P}_q is not available, state-of-the-art works marginalize the first retrieval step and train both retriever and generator based on gradient from the negative log-likelihood loss. This results in training a biased retriever which favors label-relevant passages $\mathcal{P}_l \subseteq \mathcal{P}_q$, a biased subset of question-relevant passages. Thus in training, the retriever will overfit to these label-relevant passages.

In this work, we consider another subset of question-relevant passages, which is knowledge-relevant passages $\mathcal{P}_k \subseteq \mathcal{P}_q$. We define \mathcal{P}_k as passages that contain the knowledge to generate label answer y . Therefore it is obvious that $\mathcal{P}_l \subseteq \mathcal{P}_k$. Unlike label-relevant passages that usually have high token overlap with label answer y , knowledge-relevant passages are defined at a higher knowledge level. In examples from Table 1, the Passage 1 is a label-relevant passage (also the gold passage), while the Passage 2 is a knowledge-relevant passage, which has the same meaning as the label answer but with little token overlap with the former. By introducing a training framework that focuses on retrieving knowledge-relevant passages, the retriever can be trained more robustly and provide better generalization ability.

4 Methodology

In order to focus on retrieving knowledge-relevant passages in KIG tasks, we introduce a new training

framework with two new objectives: **Retriever as Answer Classifier (RAC)**, and **Dense Knowledge Similarity (DKS)**. Both objectives aim to mitigate the problem discussed in Section 3, that the retriever solely trained with NLL loss prefers label-relevant passages rather than question-relevant passages. Figure 1 shows the overall architecture of the proposed model.

4.1 Retrieve-Generate Backbone

Transformer-based retrieve-generate systems are widely used in knowledge-intensive generation tasks like generative QA and conversational search (Cooper Stickland et al., 2021; Adiwardana et al., 2020). To apply our knowledge-constrained objectives, we need a retrieve-generate architecture as the backbone. Here we choose to use the architecture from the RAG paper (Lewis et al., 2020b) as the retrieve-generate backbone. Given a question q and passage corpus \mathcal{P} , we first use a Dense Passage Retriever (DPR; Karpukhin et al. (2020)) to compute the similarity of each passage $p_i \in \mathcal{P}$ with q :

$$\text{sim}(p_i, q) = \text{enc}_q(q)^\top \text{enc}_p(p_i), \quad (1)$$

where enc_q is the question encoder and enc_p is the passage encoder. To facilitate easier comparison with prior works, we follow the settings in Lewis et al. (2020b) to fix the parameters of the passage encoder. Based on the similarity, we select top- n passages with Maximum Inner Product Search (MIPS) and calculate the probability

$$p(p_i|q) = \frac{\exp(\text{sim}(p_i, q))}{\sum_{p_j \in \text{Top-}n(\mathcal{P}|q)} \exp(\text{sim}(p_j, q))}, \quad (2)$$

where $\text{Top-}n(\mathcal{P}|q)$ is the selected top- n passages with given q . Then these passages are concatenated with the question and generate the probability distribution of response tokens through a sequence-to-sequence encoder-decoder model:

$$p(y_t|y_{t-1}, q, p_i) = \text{BART}([q; p_i], y_{t-1}) \quad (3)$$

$$p_{p_i}(y_t|q, y_{t-1}) = p(y_t|y_{t-1}, q, p_i)p(p_i|q) \quad (4)$$

$$p_{gen}(y_t|q, y_{t-1}) = \sum_{p_i \in \text{Top-}n(\mathcal{P}|q)} p_{p_i}(y_t|q, y_{t-1}) \quad (5)$$

$$p_{gen}(y|q) = \prod_{t=1}^T p_{gen}(y_t|q, y_{t-1}), \quad (6)$$

where $\text{BART}([q; p_i], y_{t-1})$ is the BART generator (Lewis et al., 2020a) that takes the concatenation

of q and p_i , and y_{t-1} as input. The goal of the backbone is to maximize the probability of label sentence \hat{y} , which is equal to minimizing the negative log-likelihood of \hat{y} :

$$L_{gen} = -\log p_{gen}(\hat{y}|q) \quad (7)$$

4.2 Retriever as Answer Classifier

Since we would like our retriever to retrieve knowledge-relevant passages instead of only retrieving label-relevant passages, we add a new answer classification objective, which makes use of rich answer encoding to capture knowledge inside the answer. The answer classifier outputs whether the retrieved top- n passages contain the knowledge inside the answer. Here we reuse the DPR retriever (sharing parameters) in the retrieve-generate backbone to generate answer encoding and use it for classifying the matched answer.

We apply in-batch negative sampling (Karpukhin et al., 2020) for the RAC objective. Given a question-answer pair (\hat{q}, \hat{y}) , we use m answers in the same batch as negative answers from the rest of the training set:

$$Neg = \{y^{(1)}, y^{(2)}, \dots, y^{(m)}\}, \quad (8)$$

where the batch size is $m + 1$. We then compute the probability of the answer (both positive and negative) y given question \hat{q} as:

$$\text{sim}(y, p_i^{\hat{q}}) = \text{enc}_p(p_i^{\hat{q}})^\top \text{enc}_q(y) \quad (9)$$

$$S(y, \hat{q}) = \sum_{p_i^{\hat{q}} \in \text{Top-}n(\mathcal{P}|\hat{q})} p(p_i^{\hat{q}}|\hat{q}) \text{sim}(y, p_i^{\hat{q}}) \quad (10)$$

$$p_{RAC}(y|\hat{q}) = \sigma(S(y, \hat{q})), \quad (11)$$

where $p(p_i^{\hat{q}}|\hat{q})$ is computed with Eq. (2), enc_p and enc_q are the same as described in Section 4.1, and $\sigma(\cdot)$ is the sigmoid function for generating probability. We then use binary cross-entropy loss to get the training signal:

$$L_{RAC} = \log(p_{RAC}(\hat{y}|\hat{q})) + \sum_{y' \in Neg} \log(1 - p_{RAC}(y'|\hat{q})) \quad (12)$$

4.3 Dense Knowledge Similarity

Another way of focusing on knowledge-relevant passages is to further marginalize the generation process. This objective directly focuses on knowledge instead of the answer label, which changes

the final goal from generating a natural language response to answer the question $p(y|q)$ to generating knowledge k required to answer the question $p(k|q)$:

$$p(k|q) = \sum_y p(k|y)p(y|q) \quad (13)$$

Thus the optimization process becomes maximizing $p(\hat{k}|q)$. To accomplish that, we need to get the gold knowledge \hat{k} for training.

We choose to use the sentence bottleneck auto-encoder³ introduced by Montero et al. (2021) to get answer knowledge representations. The original goal of sentence bottleneck auto-encoder is to reconstruct the input sequence through a bottleneck representation between encoder and decoder (as shown in Figure 2):

$$\text{AE}(y) = \text{dec}_{\text{AE}}(\beta(\text{enc}_{\text{AE}}(y))) = y', \quad (14)$$

where $\beta(\cdot)$ is the knowledge extractor, a multihead attention mechanism that aggregates encoder states \mathbf{H} to a single bottleneck representation:

$$\mathbf{H} = \text{enc}_{\text{AE}}(y) \quad (15)$$

$$\mathbf{z} = \beta(\mathbf{H}) \quad (16)$$

$$\beta(\mathbf{H}) = \text{MultiHead}(\mathbf{q}, \mathbf{K}, \mathbf{V}) \quad (17)$$

where \mathbf{q} is the encoder states of [CLS] token, \mathbf{K} and \mathbf{V} are the encoder states \mathbf{H} . Here we treat the bottleneck representation as the knowledge representation of the input sequence:

$$\mathbf{k} = \mathbf{z} \quad (18)$$

\mathbf{k} is a rich answer encoding containing knowledge needed to rebuild the sentence itself, and thus, implicitly guarantees that the knowledge represented by a wrong answer is different from the one by a right answer. We, subsequently, train another knowledge extractor α to extract knowledge representation from question-passage pairs:

$$\mathbf{k}' = \sum_{p_i \in \text{Top-}n(\mathcal{P}|q)} \alpha(\text{enc}_{\text{AE}}([q; p_i]))p(p_i|q) \quad (19)$$

We expect \mathbf{k}' to be similar to \mathbf{k} in the knowledge representation space, thus we use MSE loss to obtain the training signal:

$$L_{DKS} = \|\hat{\mathbf{k}} - \mathbf{k}'\|_2^2 \quad (20)$$

³Appendix B provides more details about our reasoning behind selecting sentence bottleneck auto-encoder.

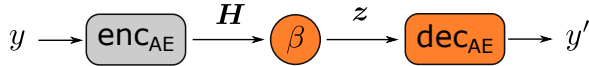


Figure 2: Sentence Bottleneck Auto-encoder. The encoder is initialized from pre-trained language models and is not trainable.

By fitting the dense knowledge representations (rich answer encoding) instead of the original label answers, we change the goal from retrieving label-relevant passages to retrieving knowledge-relevant passages.

4.4 Overall Architecture

Using the retrieve-generate backbone and the two objectives with rich answer encoding, we can train the model with all these objectives in a multi-task setting. In training, we optimize the summation of all the different losses:

$$L = L_{gen} + \mathbf{w}(L_{RAC}, L_{DKS})^T, \quad (21)$$

where L_{gen} , L_{RAC} and L_{DKS} are computed in Eq. (7), (12) and (20) respectively. $\mathbf{w} \in \mathbb{R}^2$ is the weight vector for balancing different losses. We compute \mathbf{w} in each step to make sure that all the losses will have the same value.

5 Experimental Setup

We evaluate the proposed methods on two KIG tasks: generative question answering and informative conversations. The designed experiments aim to answer the following research questions:

- **RQ1** Effectiveness: Can the proposed methods improve the retrieval and end-to-end generation quality?
- **RQ2** Robustness: Do the proposed methods facilitate better generalization at retrieving relevant passages?

5.1 Datasets

For generative question answering, we use the NLGen split of the MSMARCO QA dataset (Nguyen et al., 2016). The answers of the NLGen split are rewritten by crowdworkers into well-formed answers for reducing overlaps between answers and gold passages. For the informative conversation task, we use the Wizard of Wikipedia (WoW) dataset (Dinan et al., 2019) organized by KILT (Petroni et al., 2021).

Since the test split of KILT-WoW is not published, we only report our best result on the test

split as shown on the KILT leaderboard. All other experiments are done on the public development set. For MSMARCO dataset, since the challenge is retired, and we do not have access to the full test split, we further split the original train split to be the train set and validation set in our experiment, and report results on the original development split. All the evaluation metrics reported in this paper are obtained using the official scripts provided by MSMARCO⁴ and KILT.⁵

5.2 Metrics

For the MSMARCO dataset, we report ROUGE-L and BLEU-1/4 scores for evaluating the end-to-end generation quality. We also report the recall rate of retrieved passages to evaluate the retrieval quality. For KILT-WoW dataset, we follow the evaluation setup in Petroni et al. (2021) and report: 1) ROUGE-L score and F1 score for the end-to-end evaluation; 2) R-Precision and Recall@5 for the retrieval evaluation; 3) KILT-RL and KILT-F1 for the combined evaluation. R-Precision metric is identical to Precision@1 in KILT-WoW dataset since the gold passage for every conversation is a single Wikipedia page. KILT-RL and KILT-F1 are two special metrics motivated by FEVER-score (Thorne et al., 2018), which calculate ROUGE-L score and F1 score as 0 when the retriever does not perfectly select the gold passage. Furthermore, for both datasets, we use METEOR (Banerjee and Lavie, 2005), which is a metric with a high correlation with human judgment, to mimic the human preference for the generations.

5.3 Knowledge Sources

For both datasets, we retrieve passages from the knowledge sources which the gold passages are selected from. For MSMARCO, the knowledge source contains 8.8M passages extracted from 3.6M web documents retrieved by Bing, while for KILT-WoW, the knowledge source is extracted from the 2019/08/01 Wikipedia dump, which after our preprocessing contains approximately 24.5M passages.

5.4 Model Configuration

Following Lewis et al. (2020b), we use DPR model fine-tuned with Natural Questions (Kwiatkowski et al., 2019) to initialize the retriever, and

⁴<https://github.com/microsoft/MSMARCO-Question-Answering>

⁵<https://github.com/facebookresearch/KILT>

Model	End-to-end				Retrieval		
	B-1	B-4	R-L	METEOR	R@1	R@5	R@10
RAG (ours)	53.9	33.3	58.6	49.7	10.7	25.3	32.1
DPR (ours)	-	-	-	-	10.5	27.0	34.9
ours (RAC)	54.1	33.8	59.0	49.7	11.2	26.8	34.1
ours (DKS)	54.9	34.1	58.7	50.1	11.5	27.5	35.0
ours (multi-task)	55.1	34.8	59.1	50.5	12.0	28.3	36.0

Table 2: Effectiveness evaluation results in MSMARCO development set. R-L represents ROUGE-L, B-1 and B-4 represent BLEU-1 and BLEU-4. R@k represents for recall rate for top-k retrieved passages. **Bold** highlighted values are the best among all.

Model	Retrieval		End-to-end			Combined	
	R-Prec	Recall@5	RL	F1	METEOR	KILT-RL	KILT-F1
RAG	42.21	61.98	15.68	17.92	12.90	7.74	8.83
DPR	26.92	49.64	-	-	-	-	-
ours (RAC)	44.95	65.39	16.81	18.93	14.47	8.80	9.95
ours (DKS)	47.02	64.87	16.10	18.64	14.07	9.17	10.68
ours (multi-task)	43.94	63.59	16.66	19.04	14.41	8.58	9.84

Table 3: Effectiveness evaluation results in KILT-WoW development set. RL and KILT-RL refer to ROUGE-L and KILT ROUGE-L. R-Prec refers to R-Precision, which is identical to Precision@1 in the KILT-WoW dataset. **Bold** highlighted values are the best among all.

BART_{LARGE} model to initialize the generator. For the KILT-WoW dataset, we initialize all the models using non-finetuned RAG checkpoint provided on HuggingFace⁶. For the MSMARCO dataset, we find that using the same initialization strategy does not work well for baseline and our proposed methods. We hypothesize that this is due to the questions of MSMARCO having a very different structure compared to the Natural Question dataset (where the initialized model is trained). Thus we first pre-train the DPR retriever on the MSMARCO passage ranking task, then use the pre-trained model as the initialization point for all MSMARCO experiments. For the training of sentence bottleneck auto-encoder, we use RoBERTa_{BASE} provided on HuggingFace⁷ as the sentence encoder. At inference time, we use top-5 retrieved passages to support generation in both datasets. Other Detailed settings are shown in Appendix A.

⁶<https://huggingface.co/facebook/rag-token-base>

⁷<https://huggingface.co/facebook/roberta-base>

5.5 Comparison Models

We compare our model with RAG (Lewis et al., 2020b), which also serves as the encoder-decoder backbone of the proposed methods. For fair comparisons, we train our own RAG model with the same configuration as our proposed methods. Additionally, to evaluate how baseline RAG and our proposed model benefit the retriever training, we also compare with the non-finetuned DPR (Karpukhin et al., 2020) model on the retrieval evaluation.

We choose not to compare our model with other state-of-the-art works on the KILT leaderboard since the retriever settings are very different (both the retriever architecture and size) and the comparison is helpless for showing the benefits of our work.

6 Results

In this section, we show our results with respect to RQ1 and RQ2.

6.1 RQ1: Effectiveness

Generative QA Table 2 shows the effectiveness evaluation results on MSMARCO development

Model	Retrieval		End-to-end		Combined	
	R-Prec	Recall@5	RL	F1	KILT-RL	KILT-F1
RAG	50.99	65.98	15.61	17.86	8.98	10.46
ours	55.54	68.63	16.36	18.57	10.17	11.84

Table 4: Test results on KILT leaderboard. We reported our best model under the development set, which is DKS. RL and KILT-RL refer to ROUGE-L and KILT ROUGE-L. R-Prec refers to R-Precision, which is identical to Precision@1 in the KILT-WoW dataset. **Bold** highlighted values are the best.

Model	MSMARCO			KILT-WoW		
	Train	Dev	$\Delta \downarrow$	Train	Dev	$\Delta \downarrow$
RAG (ours)	17.5	10.7	6.8	46.73	42.21	4.52
ours (RAC)	17.8	11.2	6.6	45.32	44.95	0.37
ours (DKS)	16.3	11.5	4.8	47.94	47.02	0.92
ours (multi-task)	16.6	12.0	4.6	46.47	43.94	2.53

Table 5: Robustness evaluation results for the proposed methods. For MSMARCO dataset we report Recall@1 and for KILT-WoW dataset we report R-Prec. $\Delta = \text{Train} - \text{Dev}$ indicates the overfitting level of the given model. **Bold** highlighted values are the best among all.

set. First of all, we find that the proposed methods outperform the baseline in the retrieval step. Both proposed methods achieve consistent improvement compared to the baseline RAG and the non-finetuned DPR. The multi-task setting model works best, which achieves about 14.3% relative improvement (1.5% absolute improvement) to the non-finetuned DPR on Recall@1, showing a large benefit to the training of the retriever. In contrast, the baseline RAG only gets 1.9% relative improvement (0.2% absolute improvement). For the end-to-end generation results, we also see a consistent improvement, especially in BLEU scores. In the multi-task setting, our method got a 4.5% relative improvement on the BLEU-4 score. This indicates that better retrieval quality benefits the end-to-end generation results.

Informative Conversations Table 3 shows the effectiveness evaluation results of proposed methods on KILT-WoW development set. Similarly to the case of the MSMARCO experiments, for all versions of our model, there is a consistent improvement in the retrieval metrics. As a benefit of better retrieval quality, we also get consistent improvement in the end-to-end and combined evaluations. Specifically, the DKS version achieves the best R-Precision across all models, with 11.4% relative improvement compared to the baseline, showing that the continuous sequence-level features are more

suitable for such casual conversational task where the answer label can have less overlap with gold passages.

We submit our best model and the fine-tuned RAG baseline to the KILT-WoW leaderboard.⁸ Table 4 shows the evaluation results. We do not include the official results of the RAG baseline reported by KILT, since the dataset splits are updated due to a mapping issue (Paranjape et al., 2022). The proposed methods get improvements across all metrics compared to the baseline RAG, with 13.3% relative improvement on KILT-RL score and 13.2% relative improvement on KILT-F1 score, proving that the proposed methods benefit on both the retrieval part and generation part.

6.2 RQ2: Robustness

In this section, following Roelofs et al. (2019), we use the differences between train set metrics and development set metrics to measure the overfit degree to evaluate **RQ2** that is relevant to robustness. Table 5 shows the robustness evaluation results of the proposed methods in both datasets. We measure the retrieval performance drop from the training set to the development set as the overfit degree Δ . In the result, we can find that the baseline RAG model generalizes badly, with a significant per-

⁸<https://eval.ai/web/challenges/challenge-page/689/leaderboard/1909/KILT-F1>

Context	A: Do you like Gouda cheese, its a mild, yellow cheese. B: Well I like most cheeses so I'd probably like it, yes. Do you know where it originated from?
Label Answer	It is one of the oldest recorded cheeses still made today, from 1184, people were mentioning Gouda Cheese
RAG	Gouda cheese is a type of cheese made from gouda, which is a type of goat's milk .
ours (RAC)	The first mention of Gouda cheese dates from 1184, making it one of the oldest recorded cheeses in the world still made today.
ours (DKS)	The first mention of Gouda cheese dates from 1184, making it one of the oldest recorded cheeses in the world still made today.
ours (multi-task)	Gouda cheese is a type of cheese made from gouda, which is a type of cow's milk .

Table 6: Sampled Generations from KILT-WoW development set.

formance drop in both datasets. In contrast, the proposed methods have far less overfit degree, especially in the KILT-WoW dataset. We also note that some of the proposed methods have better train set results than the RAG model, indicating that the proposed methods can also help to better fit the training data while achieving better generalization to unseen data.

6.3 Case Studies

Table 6 shows some sampled generations from the KILT-WoW development set. In the example, both RAC and DKS retrieve the same Wikipedia page and give responses with a perfect knowledge match. Though the baseline and the multi-task model both failed in retrieving the gold passage, our model is providing an acceptable answer encompassing the correct knowledge, while the baseline generates hallucination ('goat's milk').

7 Conclusion

In this work, we introduced a new training framework for retrieval augmented generation models, for training a more robust retriever. The new training framework includes two knowledge-constrained objectives, RAC and DKS, to help guide the retriever to retrieve knowledge-relevant passages instead of only label-relevant passages. We evaluate the proposed methods on two well-known KIG datasets, MSMARCO and KILT Wizard of Wikipedia. The results show a consistent improvement of our proposed methods on both retrieval metrics and generation metrics compared to the baseline model. On MSMARCO, our best model achieves 12.1% relative improvement on Recall@1. In the KILT-WoW leaderboard, our best model achieves 8.9% relative improvement on R-Precision, 13.3% relative improvement on KILT-RL, and 13.2% relative improvement on KILT-F1

compared to the baseline model. Furthermore, we show that the focusing on label-relevant passages for the KIG task can result in overfitting, which we manage to mitigate considerably using our training framework. The reliable improvement in the retrieval quality indicates that our work could further contribute to the research community in the era of LLM, where the proposed methods could be used for training retrievers to reduce hallucinations from LLMs.

We believe retrieval augmented methods are crucial for LLMs. As future work, one idea is to combine passages and structured knowledge, such as databases (Vougiouklis et al., 2023) and/or knowledge graphs, possibly including uncertain knowledge graphs (Pan et al., 2005; Stoilos et al., 2006; Qi et al., 2007; Chen et al., 2019), knowledge graph with selected vocabulary (Wang et al., 2014) and temporal knowledge graphs (García-Durán et al., 2018; Bourgaux et al., 2021), for retrieval augmented generations. Furthermore, as suggested in (Pan et al., 2023a), there are a few pressing challenges in this space, such as unifying knowledge editing (Mitchell et al., 2022; Han et al., 2023) and retrieval augmentation, complex reasoning via retrieval augmentation and semi-parametric LLMs (Pan et al., 2023b) in general.

Acknowledgement

This work is supported by Huawei's Dean's Funding (C-00006589) and the Chang Jiang Scholars Program (J2019032).

Limitations

The first limitation is that the parameters of the passage encoder is fixed during the training of the retriever for a fair comparison with the baseline model. This potentially limits the expressiveness of the bi-encoder retriever and harms the overall

performance of the retrieve-generate model. However, unfixing the passage encoder will significantly increase the training cost, which is not feasible for us.

Another potential limitation is that we cannot access the MSMARCO test set since the challenge is retired, which limits the training samples for the experiments.

References

- Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. [Towards a human-like open-domain chatbot](#). *CoRR*, abs/2001.09977.
- Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023. [Knowledge-augmented language model prompting for zero-shot knowledge graph question answering](#). In *Proceedings of the 1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pages 78–106, Toronto, Canada. Association for Computational Linguistics.
- Satanjeev Banerjee and Alon Lavie. 2005. [METEOR: An automatic metric for MT evaluation with improved correlation with human judgments](#). In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.
- Camille Bourgaux, Ana Ozaki, and Jeff Z. Pan. 2021. [Geometric Models for \(Temporally\) Attributed Description Logics](#). In *Proc. of DL2021*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Xuelu Chen, Muhao Chen, Weijia Shi, Yizhou Sun, and Carlo Zaniolo. 2019. [Embedding uncertain knowledge graphs](#). In *AAAI*, pages 3363–3370. AAAI Press.
- Md. Faisal Mahbub Chowdhury, Michael R. Glass, Gaetano Rossiello, Alfio Gliozzo, and Nandana Mihindukulasooriya. 2022. [KGI: an integrated framework for knowledge intensive language tasks](#). *CoRR*, abs/2204.03985.
- Asa Cooper Stickland, Xian Li, and Marjan Ghazvininejad. 2021. [Recipes for adapting pre-trained monolingual and multilingual models to machine translation](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3440–3453, Online. Association for Computational Linguistics.
- Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. [Wizard of wikipedia: Knowledge-powered conversational agents](#). In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.
- Hady Elsahar, Pavlos Vougiouklis, Arslan Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. [T-REx: A large scale alignment of natural language with knowledge base triples](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- William Falcon and The PyTorch Lightning team. 2019. [Pytorch lightning](#).
- James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. 2020. [IIRC: A dataset of incomplete information reading comprehension questions](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1137–1147, Online. Association for Computational Linguistics.
- Alberto García-Durán, Sebastijan Dumancic, and Mathias Niepert. 2018. [Learning sequence encoders for temporal knowledge graph completion](#). In *EMNLP*, pages 4816–4821. Association for Computational Linguistics.
- Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Naik, Pengshan Cai, and Alfio Gliozzo. 2022. [Re2G: Retrieve, rerank, generate](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2701–2715, Seattle, United States. Association for Computational Linguistics.
- Xiaoqi Han, Ru Li, Xiaoli Li, and Jeff Z. Pan. 2023. [A divide and conquer framework for knowledge editing](#). *Knowledge-Based Systems*, 279:110826.
- Hangfeng He, Hongming Zhang, and Dan Roth. 2023. [Rethinking with retrieval: Faithful large language model inference](#). *CoRR*, abs/2301.00303.
- Gautier Izacard and Edouard Grave. 2021. [Leveraging passage retrieval with generative models for open domain question answering](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

- Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. 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.
- Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 39–48. ACM.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.
- Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada. Association for Computational Linguistics.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Yury A. Malkov and Dmitry A. Yashunin. 2020. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(4):824–836.
- Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. 2022. Memory-based model editing at scale. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR.
- Ivan Montero, Nikolaos Pappas, and Noah A. Smith. 2021. Sentence bottleneck autoencoders from transformer language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1822–1831, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I. Jordan, and Ion Stoica. 2018. Ray: A distributed framework for emerging AI applications. In *13th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2018, Carlsbad, CA, USA, October 8-10, 2018*, pages 561–577. USENIX Association.
- Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- Ansong Ni, Matt Gardner, and Pradeep Dasigi. 2021. Mitigating false-negative contexts in multi-document question answering with retrieval marginalization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6149–6161, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.
- Jeff Z. Pan, Simon Razniewski, Jan-Christoph Kalo, Sneha Singhanian, Jiaoyan Chen, Stefan Dietze, Hajira Jabeen, Janna Omeliyanenko, Wen Zhang, Matteo Lissandrini, ussa Biswas, Gerard de Melo, Angela Bonifati, Edlira Vakaj, Mauro Dragoni, and amien Graux. 2023a. Large language models and knowledge graphs: Opportunities and challenges. *Transactions on Graph Data and Knowledge*.
- Jeff Z. Pan, Giorgos Stamou, Vassilis Tzouvaras, and Ian Horrocks. 2005. f-SWRL: A Fuzzy Extension of SWRL. In *Proc. of ICANN 2005, Special section on "Intelligent multimedia and semantics"*.
- Jeff Z. Pan, Guido Vetere, José Manuel Gómez-Pérez, and Honghan Wu, editors. 2017. *Exploiting Linked Data and Knowledge Graphs in Large Organisations*. Springer.
- Xiaoman Pan, Wenlin Yao, Hongming Zhang, Dian Yu, Dong Yu, and Jianshu Chen. 2023b. Knowledge-in-context: Towards knowledgeable semi-parametric language models. In *Proc. of ICLR2023*.

- Ashwin Paranjape, Omar Khattab, Christopher Potts, Matei Zaharia, and Christopher D Manning. 2022. [Hindsight: Posterior-guided training of retrievers for improved open-ended generation](#). In *International Conference on Learning Representations*.
- Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. [KILT: a benchmark for knowledge intensive language tasks](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.
- Guilin Qi, Jeff Z. Pan, and Qiu Ji. 2007. A Possibilistic Extension of Description Logics. In *Proc. of DL2007*.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Rebecca Roelofs, Vaishaal Shankar, Benjamin Recht, Sara Fridovich-Keil, Moritz Hardt, John Miller, and Ludwig Schmidt. 2019. [A meta-analysis of overfitting in machine learning](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 9175–9185.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. [Recipes for building an open-domain chatbot](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325, Online. Association for Computational Linguistics.
- Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. [Retrieval augmentation reduces hallucination in conversation](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Giorgos Stoilos, Giorgos B. Stamou, and Jeff Z. Pan. 2006. [Handling imprecise knowledge with fuzzy description logic](#). In *Proceedings of the 2006 International Workshop on Description Logics (DL2006), Windermere, Lake District, UK, May 30 - June 1, 2006*, volume 189 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. [FEVER: a large-scale dataset for fact extraction and VERification](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.
- Pavlos Vougiouklis, Nikos Papasrantopoulos, Danna Zheng, David Tuckey, Chenxin Diao, Zhili Shen, and Jeff Z. Pan. 2023. [FastRAT: Fast and Efficient Cross-lingual Text-to-SQL Semantic Parsing](#). In *Proc. of IJCNLP-AAACL 2023*.
- Kewen Wang, Zhe Wang, Rodney W. Topor, Jeff Z. Pan, and Grigoris Antoniou. 2014. [Eliminating Concepts and Roles from Ontologies in Expressive Descriptive Logics](#). pages 205–232.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *NeurIPS*.
- Donghan Yu, Chenguang Zhu, Yuwei Fang, Wenhao Yu, Shuohang Wang, Yichong Xu, Xiang Ren, Yiming Yang, and Michael Zeng. 2022. [KG-FiD: Infusing knowledge graph in fusion-in-decoder for open-domain question answering](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4961–4974, Dublin, Ireland. Association for Computational Linguistics.
- Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A. Smith. 2023. [How language model hallucinations can snowball](#). *CoRR*, abs/2305.13534.
- Ruo Chen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023. [Verify-and-edit: A knowledge-enhanced chain-of-thought framework](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5823–5840, Toronto, Canada. Association for Computational Linguistics.

A Detailed Experiment Settings

Training Cost Our experiments are built on top of the RAG research project provided on HuggingFace⁹. We use pytorch-lightning (Falcon and team, 2019) for fine-tuning the model, and use ray (Moritz et al., 2018) and FAISS (Johnson et al., 2019) for distributed retrieval with HNSW index (Malkov and Yashunin, 2020). All the models are trained under FP16 for memory and time efficiency. We use 4 NVIDIA A100 80G GPUs for fine-tuning the retrieve-generate models and 1 NVIDIA A100

⁹https://github.com/huggingface/transformers/tree/main/examples/research_projects/rag

80G GPU for inference. Specially, since the experiment requires fast retrieval using FAISS, it requires approximately 200GB CPU memory for running MSMARCO experiments, and 300GB CPU memory for running KILT-WoW experiments (both training and inference). The memory requirement can be reduced by compressing the HNSW index. Table 10 shows the average runtime for the experiments. Since the fine-tuning runtime varies during training as the retriever training on CPU gets faster and faster, we only report the runtime of the first epoch. Note that the training runtime is not very comparable, as it depends largely on the FAISS retrieval. Table 7 shows the trainable parameters of our models.

Model	Trainable Params
RAG	515M
ours (RAC)	515M
ours (DKS)	518M
ours (multi-task)	518M

Table 7: Trainable parameters of models.

Dataset Details Table 8 shows the statistics of datasets used in our experiments. Both the datasets and knowledge sources are in English. For the MSMARCO dataset, we split the original training set into our training set and validation set, and report all the experimental results on the original development set. This setting is for better comparison to the results reported in Lewis et al. (2020b). For both knowledge sources, we use the script provided by HuggingFace¹⁰ to build FAISS format corpus. Specially, for KILT-WoW knowledge source, we do an extra data cleaning process, including 1) removing paragraphs starting with Section::: and BULLET:::; 2) removing all non-unicode characters.

Dataset	Train	Dev	Test
MSMARCO	143,725	10,000	12,467
KILT-WoW	63,734	3,054	2,944

Table 8: Dataset Statistics. The original development set of MSMARCO dataset is used as the test set in our experiments, which is still referred to as the development set in the result section.

¹⁰https://github.com/huggingface/transformers/blob/main/examples/research_projects/rag/own_knowledge_dataset.py

Hyperparameters Table 9 shows the hyperparameter settings for training and evaluation of the proposed model. If some hyperparameters are not mentioned, then keep them as default in HuggingFace. For Kilt-WoW experiments, we select the best hyperparameter (batch size and passage nums for training) based on the ROUGE-L score. Noticed that it is not possible for our experimental setup to choose batch size of 16 and passage nums of 10, as it explodes the GPU memory. For KILT-WoW experiments, the best combination of (batch size, passage nums) for our DKS model is (8, 10), while for others is (16, 5). For receiving the reference knowledge representations, we train sentence bottleneck auto-encoder (Montero et al., 2021) on each dataset for 100,000 iterations with batch size 64 and learning rate 3e-5.

Hyperparam	MSMARCO	WoW
Random Seed	42	42
Learning Rate	1e-5	1e-5
Batch Size	8	{8, 16}
Largest Epochs	20	20
Early Stop Patience	3	3
Optimizer	Adamax	Adamax
Gradient Clip	0.1	0.1
Passage Nums	10	{5, 10}
Dropout	0.1	0.1
Beam Size	4	4

Table 9: Hyperparameter settings for experiments on two different datasets in this work. All models share the same hyperparameter settings.

B Sentence Bottleneck AutoEncoder

There could be different ways to obtain label knowledge representations. In our preliminary experiments, we tried to use [CLS] token representation of pre-trained BERT model or simply use average or maximum pooling to obtain label knowledge representations. But all these methods don’t work well, as they can not provide enough knowledge about the answer without further fine-tuning. Besides, the sentence bottleneck auto-encoder fits our needs well and also works much better. We initialise the encoder of the sentence bottleneck auto-encoder with bert-base-uncased checkpoint from huggingface, and then finetune it with the MSMARCO and KILT-WoW datasets before training the retrieve-generate model.

C More Generation Samples

Table 11 provides more generation samples from KILT-WoW experiments.

Model	MSMARCO		KILT-WoW	
	Fine-tuning	Inference	Fine-tuning	Inference
RAG	6,918s	0.28s	3,228s	0.37s
ours (RAC)	8,202s	0.28s	6,702s	0.42s
ours (DKS)	9,980s	0.32s	5,134s	0.40s
ours (multi-task)	10,551s	0.32s	5,028s	0.41s

Table 10: Average runtime of experiments. Fine-tuning numbers are average seconds per training first epoch, inference numbers are average seconds per generation sample. For MSMARCO, we report fine-tuning runtime with batch size=16. For KILT-WoW, we report fine-tuning runtime with batch size=8.

Context	A: I have observed that after black hair Brown hair is the second most common human hair color. B: Im not actually sure about my own i think its black but might just be dark brown. A: Really, mine is black, but People with brown hair are often referred to as brunette. B: Yea i think mine might be black but who knows.
Reference response	Maybe, i have observed that brown hair is common in the western world
RAG	Yea brown hair is the second most common hair color after black hair.
ours (RAC)	Brown hair is the second most common human hair color after black hair.
ours (DKS)	Brown hair is common among populations in the Western world, especially among those from Central Europe, Southeastern Europe, Eastern Europe, Southern Europe, Southern Cone, the United States, and also some Asians.
ours (multi-task)	It is characterized by higher levels of the dark pigment eumelanin and lower levels of the pale pigment pheomelanin.
Context	A: I really like cooking pasta, it is some of my favorite Italian food. B: I do too. What type of pastas do you like to make? A: Angel hair is always a great one to use in standard Italian cuisine. B: Yes! I prefer angel hair in spaghetti instead of the traditional spaghetti noodles. Do you make different pasta dishes?
Reference response	One of my favorite pasta dishes is just standard meat sauce and angel hair pasta.
RAG	I make a lot of different pasta dishes. I like to use rice flour instead of wheat flour.
ours (RAC)	I like to make a lot of different pasta dishes. I like to use a lot of different types of pasta.
ours (DKS)	I do, I like to use rice flour instead of wheat flour to get a different taste and texture.
ours (multi-task)	I like to make a lot of different pasta dishes. I like to use eggs instead of water.
Context	A: Hey there!! How are ya? B: Hey there! I'm very good! What do you know about the manta ray? A: Not much really but it was the only thing to choose from. B: Well it seems there are two different main types. One of them is very large and the other decently small. A: Okay so you know some about them. Great!! Have you ever seen one in the ocean before? B: I have seen the larger ones in the ocean and the smaller ones in aquariums. These things are super protected by government agencies in the wild and domestic. A: I didn't know that. Are the endangered then?
Reference response	Not quite, but they will be very soon. They have a very slow reproductive rate so it takes awhile for new ones to be had.
RAG	I'm not sure but I do know that the manta ray is one of the largest ray-finned fish.
ours (RAC)	I'm not sure but I know that they are listed as endangered by the International Union for Conservation of Nature (IUCN).
ours (DKS)	Well they are protected in international waters by the Convention on Migratory Species of Wild Animals, but they are more vulnerable closer to shore.
ours (multi-task)	I'm not sure but I know that they are filter feeders and eat large amounts of zooplankton.

Table 11: More sampled Generations from KILT-WoW development set.