Multi-Granularity Guided Fusion-in-Decoder

Eunseong Choi, Hyeri Lee, Jongwuk Lee*
Sungkyunkwan University, Republic of Korea
{eunseong, bluepig94, jongwuklee}@skku.edu

Abstract

In Open-domain Question Answering (ODQA), it is essential to discern relevant contexts as evidence and avoid spurious ones among retrieved results. The model architecture that uses concatenated multiple contexts in the decoding phase, *i.e.*, Fusion-in-Decoder, demonstrates promising performance but generates incorrect outputs from seemingly plausible contexts. To address this problem, we propose the ***Multi-Granularity guided Fusion-in-Decoder (MG-FiD)***, discerning evidence across multiple levels of granularity. Based on multi-task learning, MGFiD harmonizes passage re-ranking with sentence classification. It aggregates evident sentences into an *anchor vector* that instructs the decoder. Additionally, it improves decoding efficiency by reusing the results of passage re-ranking for *passage pruning*. Through our experiments, MGFiD outperforms existing models on the Natural Questions (NQ) and TriviaQA (TQA) datasets, highlighting the benefits of its multi-granularity solution.

1 Introduction

Open-domain question answering (ODQA) (Chen et al., 2017) is a challenging task that requires deriving factual responses from a vast knowledge corpus without relying on explicit evidence, *i.e.*, the evidence context is not given. Recently, retrieval-augmented generation (RAG) (Lewis et al., 2020) has emerged to combine the retrieval of relevant information with response generation.

Exemplified by the *retriever-reader* architecture (Chen et al., 2017; Lee et al., 2019; Guu et al., 2020), RAG effectively addresses ODQA. The retriever first pinpoints the most relevant passages using the question as a query. Subsequently, the reader extracts or generates a response using the question and the relevant passages. It generally allows us to perform a decoupled optimization for the retriever or the reader. In this paper, we mainly focus on optimizing the reader.

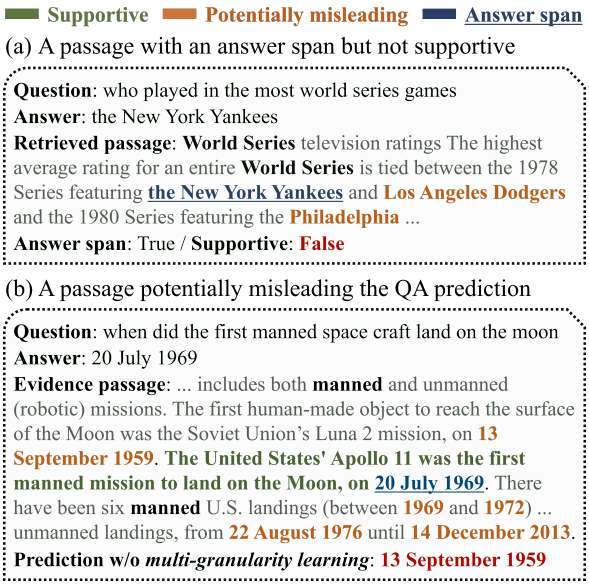


Figure 1: Examples that may harm the QA systems. Black **Bold** terms in the passages are overlapped with the question. (a) The passage is not supportive while containing a correct answer span. (b) Confusing sentences within the passage mislead model prediction.

To improve the reader, existing studies (Izacard and Grave, 2021b; Asai et al., 2022; Wang et al., 2023) focus on addressing two questions: (i) how to effectively use the evidence in multiple passages, and (ii) how to improve the discrimination in dealing with spurious passages.

Multi-passage reader. As the representative model, Fusion-in-Decoder (FiD) (Izacard and Grave, 2021b) using a generative text-to-text model (Raffel et al., 2020) is an effective multi-passage reader to aggregate evidence across multiple passages. It first encodes multiple pairs of a question and a relevant passage at the encoder. Then, it generates an answer using a cross-attention mechanism over concatenated embeddings at the decoder. One limitation of the FiD architecture is inefficiency due to the intensive cross-attention op-

*Corresponding author

Findings of the Association for Computational Linguistics: NAACL 2024, pages 2201–2212

erations performed on the concatenated matrix. To mitigate this, some studies have proposed either shortening the input length (Hofstätter et al., 2023; Yu et al., 2022) or omitting some layers in the decoder (de Jong et al., 2023). More importantly, the standard FiD model often struggles with handling spurious passages that degrade the accuracy of generating the answer (Asai et al., 2022).

Multi-task reader. Several studies (Yu et al., 2022; Ju et al., 2022; Lakhotia et al., 2021; Hofstätter et al., 2023; Wang et al., 2023) have attempted to address handling spurious passages by employing multi-task learning. It aims to improve the reader's discernment regarding the evidentiality of the retrieved passages, thereby achieving robustness against spurious ones. Yu et al. (2022) and Ju et al. (2022) proposed to incorporate information from factual triplets contained in the knowledge graph. Another solution is to employ passage labels to discern spurious passages in the FiD architecture. Lakhotia et al. (2021); Hofstätter et al. (2023); Wang et al. (2023) determined the rationality of passages based on whether they contain an answer span. Although learning signals from answer span inclusions has been proven effective, it may lead to false positive passages, producing sub-optimal results. Furthermore, existing multi-task readers face challenges in identifying the key sentences within the passage.

We argue that relying solely on answer spans or identifying evident passages is insufficient to determine the evidence. Figure 1 illustrates two plausible scenarios, highlighting the limitations of existing methods using either the answer spans or passage-level evidentiality. In Figure 1(a), the mere presence of the answer span in the passage does not guarantee relevance for the question. More importantly, Figure 1(b) shows that a model trained primarily on aggregating evidence across passages generates an incorrect answer, and there is a need to distinguish complex and confusing sentences.

This paper aims to discern evidence in coarse- and fine-grained textual information, *i.e.*, passages and sentences, and utilize the byproduct from multi-task learning to enhance the model's performance. To this end, we propose a novel model called ***M**ulti-**G**ranularity guided **F**usion-**i**n-**D**ecoder (**MGFiD**)*. Specifically, the key idea behind MGFiD is two-fold. (i) We train the FiD to distinguish evidentiality using multi-task learning to minimize the influence of false contexts during answer generation. In this process, we employ both passage- and sentence-level contexts to account for evidentiality in multi-granularity contexts. Since it is expensive to label gold passages, we use the ranking abilities of language models (Sun et al., 2023) to filter out irrelevant contexts for the question. (ii) We reuse auxiliary information from multi-task learning to improve accuracy and efficiency. We generate an *anchor vector* derived from sentence-level classification and infuse it into the [BOS] token used in the decoder. Since the anchor vector indicates a significant feature for relevant sentences, it helps the decoder generate the correct answer. Furthermore, we employ passage-level re-ranking results to prune less supportive passages, improving the efficiency in the decoding phase.

To summarize, the key contributions of this paper are as follows. (i) We introduce the evidentiality of the FiD using multi-granularity contexts. (ii) We utilize LLMs to generate pseudo-labels for supportive passages in ODQA task. (iii) We reuse multi-granularity contexts to improve accuracy and efficiency further using an anchor vector in the decoder and thresholding-based passage pruning. (iv) Through our experiments on two benchmark datasets, we show that MGFiD improves the original FiD by more than 3.5% and 1.0% in Exact Match on Natural Questions, and TriviaQA, outperforming the other baselines.

2 Related Work

We briefly review existing studies for improving the FiD (Izacard and Grave, 2021b) architecture in two key aspects: *accuracy* and *efficiency*.

2.1 Encoding Evident Passages

Several works (Lakhotia et al., 2021; Hofstätter et al., 2023; Wang et al., 2023) introduce multi-task learning to endow the model with discriminative ability, *i.e.*, the capacity to identify spurious passages. Ju et al. (2022) incorporates informative contexts in the knowledge graph with the reader. It extracts entity embeddings from the intermediate layer and combines them with graph knowledge fused through GNN. While relational information from the knowledge graph is helpful, it requires external sources. Another direction is to use heuristic rationale in multi-task learning. Lakhotia et al. (2021) proposed a special sentence marker token to enable the decoder to generate a marker corresponding to the grounds along with the answer. Wang et al. (2023) introduced a binary classifier

to determine whether each passage is supportive between the encoder and decoder. Defining rational passages is based on the answer span. As it does not guarantee the evidentiality of the passage, Asai et al. (2022) pointed out this limitation and suggested a classifier for mining pseudo-evidentiality labels. However, it still requires expensive annotations to train the classifier, and labeling with a partially trained model can be affected by the model's memorization.

2.2 Decoding Efficiency

The decoding step, mainly due to the large key-value matrix, is the most time-consuming phase in the FiD architecture during inference. Simply reducing the number of FiD inputs is not as optimal as reducing the decoder input alone (Hofstätter et al., 2023). Previous work has reduced the burden on the decoder by giving only necessary information. Hofstätter et al. (2023) reduce the length of each encoded query-passage pair to the first few vectors. Compressing the amount of information fed to the decoder can significantly improve inference efficiency while slightly reducing effectiveness. de Jong et al. (2023) removes most cross-attention layers and employs multi-query attention to reduce the cost of the decoder. Yu et al. (2022) takes intermediate layer representation for passage re-ranking and improves efficiency by passing only the high-ranked passages to the decoder. However, using a fixed number of passages is problematic as it assumes that the number of supporting documents is constant, whereas they vary.

3 Proposed Method

In this section, we first outline our method for multi-task learning, which integrates generating answers and determining their evidence at different levels of granularity, *i.e.*, passages and sentences (Section 3.1). Second, leveraging sentence-level predictions, we introduce an anchor vector to provide a rationale signal to the decoder (Section 3.2). We then present threshold-based pruning using passage-level scores to enable efficient decoding (Section 3.3). Lastly, we describe the process of generating pseudo-labels for supportive passages (section 3.4).

3.1 Learning Multi-granularity Contexts

Answer generation. We adopt the standard FiD (Izacard and Grave, 2021b) architecture as our base model. The FiD encoder takes as input the top-K retrieved passages $\mathrm{P_q} = [\mathrm{p}_1, \mathrm{p}_2, ..., \mathrm{p}_K]$ for the question q. Each p_i is prepended with q, and the FiD encoder outputs the token embeddings $\mathbf{H_i}$, which are then concatenated to obtain $\mathbf{V}$.

$$\begin{aligned} \mathbf{H}_i &= \mathrm{FiD}_{\text{encoder}}\left(\mathrm{q} + \mathrm{p}_i\right) \in \mathbb{R}^{L \times d}, \\ \mathbf{V} &= [\mathbf{H}_1; \mathbf{H}_2; \ldots; \mathbf{H}_K] \in \mathbb{R}^{(K \times L) \times d}. \end{aligned} \tag{1}$$

Here, L denotes the maximum sequence length, and d denotes the hidden dimension. The FiD decoder utilizes $\mathbf{V}$ as the key-value matrix to generate the answer auto-regressively. When T is the target sequence, the loss function is as follows:

$$\mathcal{L}_{\text{gen}} = -\sum_{t=1}^{T} \log p(\hat{y}_t \mid y_{<t}, \mathbf{V}). \tag{2}$$

Passage re-ranking. When the original FiD is solely trained on answer generation, it tends to predict incorrect answers from plausible passages, *e.g.*, passages with word overlap to the question but not supportive. To mitigate this, we account for re-ranking the evidentiality of passages. Inspired by (Nogueira and Cho, 2019), we obtain the evidence embedding $\mathbf{e}_i \in \mathbb{R}^{1 \times d}$ by passing the first token embedding of each pair to the projection layer. Specifically, let $\mathbf{h}_i^j \in \mathbb{R}^{1 \times d}$ be j-th token embedding of the $\mathbf{H}_i$, which denotes token embeddings of the question and i-th passage; we pass $\mathbf{h}_i^0 \in \mathbb{R}^{1 \times d}$ through the projection layer $\mathbf{W}_p \in \mathbb{R}^{d \times d}$. Then, a single-layer neural network $\mathbf{W}_r \in \mathbb{R}^{1 \times d}$ takes $\mathbf{e}_i$ to predict the logit for each passage. A softmax function is applied to K logits to get a probability p_i for the question and i-th passage pair.

$$p_i = \mathrm{softmax}(\mathbf{e}_i \mathbf{W}_r^\top), \text{where } \mathbf{e}_i = \mathbf{h}_i^0 \mathbf{W}_p. \tag{3}$$

Using the probability p_i, the loss function with negative log likelihood for passage re-ranking $\mathcal{L}_{\text{passage}}$ is defined by:

$$\mathcal{L}_{\text{passage}} = -\frac{1}{|\mathcal{P}|} \sum_{pos \in \mathcal{P}} \log(p_{pos}). \tag{4}$$

$\mathcal{P}$ denotes a set of indices for positive passages corresponding to the question. Here, $\mathcal{L}_{\text{passage}}$ highlights passages containing evidence and guides the decoder to focus on considering more relevant passages in generating the answer.

We adopt a listwise loss function rather than a pointwise because it makes sense to focus on relative evidentiality between K passages. Furthermore, p_i, which represents the relative importance

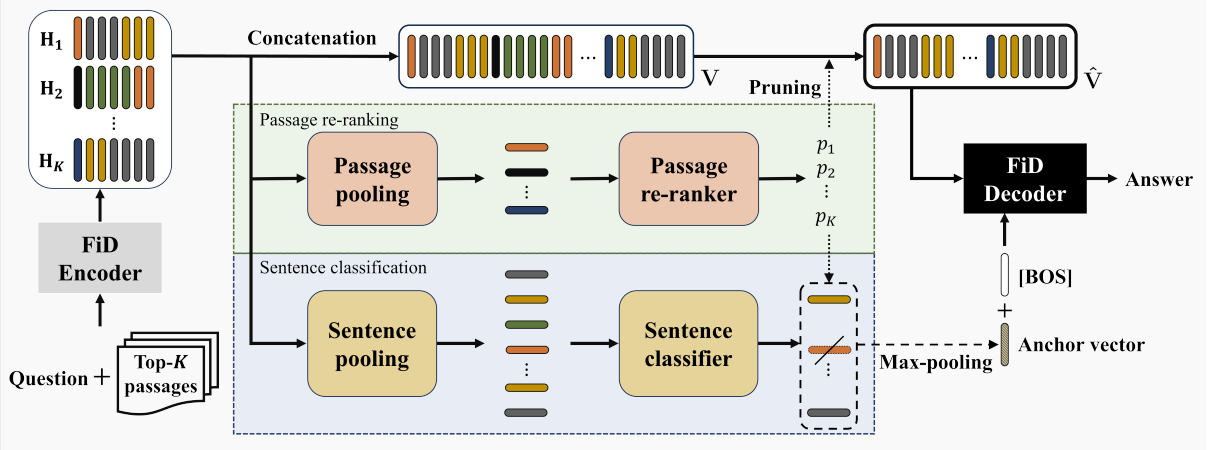


Figure 2: The MGFiD framework incorporates multi-task learning for answer generation, leveraging passage re-ranking to identify coarse-grained evidence and sentence classification for fine-grained evidence. It utilizes the outcomes of these tasks—threshold-based masking from passage re-ranking and anchor embedding from sentence classification—to enhance both efficiency and effectiveness in the answer generation process.

of each passage, is subsequently used for threshold-based masking for efficient decoding (Section 3.3).

Sentence classification. To leverage evidentiality in nuanced text, we deal with a fine-grained sentence-level task. Previous work (Liu et al., 2023) has combined different granularities to enrich the global semantics, suggesting that the information that can be captured at different levels of granularity is different. This implies that the coarse-grained semantics alone is insufficient to determine which sentences are support sentences. Therefore, multi-granularity evidentiality helps improve discrimination.

We enhance the model by learning local evidence from fine-grained sentences. Specifically, the sentence classifier takes a sentence embedding as input to predict whether the answer span is in the sentence or not. Since we need to distinguish between sentences, not their relative importance, it is designed as a simple classification task rather than a ranking task. The n-th sentence embedding of the i-th passage $\mathbf{s}_i^n \in \mathbb{R}^{1\times d}$ is expressed as the average of token embeddings projected by $\mathbf{W}_p \in \mathbb{R}^{d\times d}$. The loss function is defined after the sentence classification layer $\mathbf{W}_s \in \mathbb{R}^{d\times 2}$.

$$\mathbf{s}_i^n = \text{mean-pooling}\left(\left\{\mathbf{h}_i^j \mathbf{W}_p\right\}_{j=a_n}^{b_n}\right), \quad (5)$$
$$\mathcal{L}_{\text{sentence}}^{n,i} = \text{Focal}\left(y_i^n, \mathbf{s}_i^n \mathbf{W}_s\right).$$

Let a_n and b_n be the start and end token indices for the n-th sentence. Focal$(\cdot,\cdot)$ is the focal loss function (Lin et al., 2017) that addresses class imbalance, with y_i^n as the label indicating the presence of the answer span in the n-th sentence of the i-th passage. $\mathcal{L}_{\text{sentence}}$ is calculated as the average of all sentences in the batch.

Since the passage has been validated by LLM, using the answer span information within the passage gets more accurate. We label all sentences as negative if the passage was deemed unsupportive, regardless of the answer span. Finally, the multi-task learning loss to train MG-FiD is computed as a linear combination of the three loss functions.

$$\mathcal{L} = \mathcal{L}_{\text{gen}} + \lambda_1 \cdot \mathcal{L}_{\text{passage}} + \lambda_2 \cdot \mathcal{L}_{\text{sentence}}, \quad (6)$$

where λ_1 and λ_2 are hyperparameters that adjust the influence of passage re-ranking and sentence classification, respectively.

Figure 3 illustrates the result when only sentence classification multi-task learning is performed. It fails to learn the relationship between the question and the coarse-grained passage, and may generate the answer from the plausible passages. That is, focusing solely on sentences can limit broader semantic understanding. In detail, since the passage is prepended with the question, the first token embedding $\mathbf{h}_i^0$ used for the passage embedding is always the *"question"* token embedding. On the other hand, a sentence uses the start and end token indices of each sentence in the passage, so there is no overlap between the passage and sentence embeddings.

3.2 Incorporating an Anchor Vector

While our model is trained to discern at multiple levels of granularity, thereby highlighting evidential passages and sentences, how the decoder lever-

Potentially misleading

Question: when does the black panther movie soundtrack come out
Answer: February 9, 2018
Plausible passage: Title: Avengers: Infinity War. ... Göransson's theme from "Black Panther" is also used in the film. **Hollywood Records and Marvel Music released the soundtrack album digitally on April 27, 2018**, with a release on physical formats ...
MGFiD w/o *passage re-ranking*: **April 27, 2018 (X)**
MGFiD: **February 9, 2018 (O)**

Figure 3: Learning solely from sentences may lead to a lack of understanding of the broader context.

Document # Passage prefix
title: **{title}**
context: **{context}**

Question # Question prefix
{question}

Is this document sufficient to derive any answers among **{answers}** for the question? # A list of answers
Answer in 'Yes' or 'No'. No explanations needed.

Figure 4: A prompting example used for LLMs to filter out contexts that have an answer span but are not evident to the question.

ages this highlighted information remains unexplored. To deal with it, we align the multi-task of identifying supportive contexts with the answer generation. Specifically, we initiate the decoder with an *anchor vector*, aiming for a more focused and effective processing of relevant contexts. We leverage a set of sentences positively predicted by the sentence classifier and deal with them as an extractive summarization across multiple passages. The anchor vector, denoted as $\mathbf{e}_{\text{anchor}} \in \mathbb{R}^{1\times d}$, is obtained by performing the max-pooling operation on these positively predicted sentence embeddings, which then employed to couple the generation with the multi-task learning.

We first obtain a set of sentence embeddings S that are predicted as positives by the sentence classifier across the K passages as follows:

$$\mathcal{S} = \bigcup_{i=1}^{K} \Big\{ \mathbf{s}_i^n \mid \text{argmax}(\mathbf{s}_i^n \mathbf{W}_s) = 1, \quad \forall n \in \{1, \ldots, \mathrm{N}_i\} \Big\}, \tag{7}$$

Here, argmax$(\cdot)$ is applied to a two-dimensional vector for each of N_i sentences in p_i, returning zero for negative and one for positive. As we collect the sentence embeddings that are predicted to be positive, we then apply max-pooling over $\mathcal{S}$ to obtain the anchor vector to capture the most salient evidence.

$$\mathbf{e}_{\text{anchor}} = \text{max-pooling}\left(\mathcal{S}\right). \tag{8}$$

Lastly, we add the anchor vector to the existing [BOS] token embedding, allowing the decoder to use the evident information in the cross-attention mechanism. Our approach differs from the existing learnable guided embedding proposed by Wang et al. (2023). That is, we directly incorporate the fine-grained supportive embedding into cross-attention by adding it to the query token, as distinct from expecting guided embeddings to be reflected within the decoder layer.

3.3 Pruning Passages via a Threshold

To improve the cross-attention cost bottleneck in the decoder, we employ a threshold-based pruning method. Specifically, we reuse the probabilities for the K passages computed in the passage re-ranking task, discarding passages below a threshold τ. The resulting pruned key-value matrix $\hat{\mathbf{V}}$ based on the probability p_i in Equation (3) is formed as follows:

$$\hat{\mathbf{V}} = \bigoplus_{i=1}^{n} \mathbf{H}_i \quad \text{if } p_i > \tau. \tag{9}$$

Let $\hat{K}$ be the number of passages that exceed the threshold τ; we obtain the pruned key-value matrix $\hat{\mathbf{V}} \in \mathbb{R}^{(\hat{K}\times L)\times d}$. Adjusting the threshold from 0.0 to 0.1, we found it efficient yet effective at $\tau = 0.05$. As a result, MGFiD dynamically uses only the necessary evidence for each question, instead of a fixed number of passages as in the previous methods (Lee et al., 2022; Yu et al., 2022), thereby improving efficiency more effectively.

3.4 Evidence Labeling

A critical part of the passage re-ranking is the quality of the labels. However, gold context labels are often provided in a limited way in ODQA. While prior work (Wang et al., 2023) has shown improvement using the signal from the answer span, we propose to leverage the ranking capabilities of Large Language Models (LLMs) (Sun et al., 2023). Specifically, we use large language models to generate pseudo-labels according to the evidentiality of passages. The prompt shown in Figure 4 instructs the LLMs to identify a passage if it is sufficient to answer the question. While it would be a burden to generate pseudo-labels for every triplet candidate, *i.e.*, a question, answers, and a passage, we

Dataset	train	dev	test	R@20	# pos/q
NQ	79,168	8,757	3,610	0.87	4.5
TQA	78,785	8,837	11,313	0.86	8.9

Table 1: Data statistics. # pos/q indicates the average number of passages that have the answer span per the question. R@K is one if there exists a positive among the K passages and zero. R@20 and # pos/q are for the training dataset using the retriever (Karpukhin et al., 2020) trained by Izacard and Grave (2021a).

reduce the cost by focusing only on those that contain the answer span. To assess the effectiveness of LLMs in the labeling task, we validate them indirectly by observing the filtering rate based on the retrieval (Izacard and Grave, 2021a) rank. (Refer to the details in Appendix A.1.)

4 Experiments Setup

4.1 Datasets

We conduct our experiments on two benchmark datasets: Natural Questions (NQ) (Kwiatkowski et al., 2019) and TriviaQA (TQA) (Joshi et al., 2017). NQ comprises actual Google search queries, while TQA comprises question-answer pairs sourced from trivia and quiz-league websites. Table 1 presents the statistical details of datasets.

4.2 Metrics

We use three metrics in our experiments. Exact Match (EM) evaluates the accuracy of the QA task by examining whether normalized predictions exactly match ground-truth answers. Recall@K (R@K) assesses passage ranking, with a scoring one if the passage containing the answer span is among the top-K passages. For sentence classification, we use the area under the ROC curve (AUC) (Bradley, 1997), to account for class imbalance, *i.e.*, most are negative.

4.3 Baselines

We compare MGFiD with several baselines. FiD (Izacard and Grave, 2021b) is the first work that utilizes concatenated passage embedding at the decoder. FiD-KD (Izacard and Grave, 2021a) improves the performance of the retriever with the aggregation capability of FiD. GRAPE (Ju et al., 2022) exploits the relationships of triplets in a knowledge graph. RFiD (Wang et al., 2023) performs multi-task learning by using the answer span and proposes learnable embedding to guide the decoder. To be concise, the difference between anchor vector and guide embedding is twofold: 1) Anchor vector expects a combination of significant evidence, unlike the predicted binary label for guide embedding. 2) Guide embedding is used in the same way as other token embeddings, while an anchor vector is explicitly added in a query token for the decoder. EvidentialityQA (Asai et al., 2022) adopts an additional decoder for evidence classification and proposes a classifier for evidence labeling to perform multi-task learning.

4.4 Implementation Details

As a backbone model, we initialize the model t5-base (Raffel et al., 2020). Due to the computing cost, we mainly use the top-20, *i.e.*, $K = 20$, retrieved results provided by FiD-KD [1]. We use Adam (Kingma and Ba, 2015) as the optimizer, with a learning rate of 1e-4. We set a batch size of 2 and an accumulation step of 16 to imitate a large batch. We set λ_1 for passage ranking loss to 0.5 and λ_2 for sentence classification to 1. The α for focal loss (Lin et al., 2017), which we omit for readability in the equation 5, is set to 0.95, and the τ for threshold-based pruning is set to 0.05. The total number of steps is set to 160k, and for every 8k, we perform an evaluation with the validation set and select the checkpoint with the highest validation score. The maximum input sequence length is set to 192 for NQ and 250 for TQA. We use NLTK library (Wagner, 2010) to tokenize sentences in the passages. For evidence labeling, we use ChatGPT [2] and MythoMax [3]. While ChatGPT is a powerful large language model accessible via API, MythoMax can easily be used in the local GPUs. We fix the temperature to 0 and do not use sampling to ensure reproducibility. We use all the answer candidates in NQ; however, in the TQA dataset, which has many more answer candidates than NQ, we only collected answers in the top 20 passages for efficient prompting. The experiments in Table 2, Table 5 and Figure 6 represent the averages of five seeds, while the other experiments use a single, fixed seed. We use two NVIDIA A100 GPUs for training and inference.

For a fair comparison, we attempt to reproduce several baselines. We use the publicly available official implementations of each methodology and

[1] https://github.com/facebookresearch/FiD
[2] https://chat.openai.com/chat
[3] https://huggingface.co/TheBloke/MythoMax-L2-13B-GPTQ

Model	Multi-task learning	Retriever	Avg. # psgs in Decoder	NQ (EM) Dev	NQ (EM) Test	TQA (EM) Dev	TQA (EM) Test
FiD (2021b)	-	DPR	100	46.5	48.2	64.7	65.0
GRAPE (2022)	O	DPR	100	-	48.7	-	66.2
FiD-KD (2021a)	-	FiD-KD	100	49.2	50.1	68.7	69.3
RFiD (2023)	O	FiD-KD	100	50.0	50.7	69.6	69.6
FiD (2021b)	-	DPR	25	45.3	-	63.2	-
KG-FiD (2022)	O	DPR + GNN	20	-	49.6	-	66.7
EvidentialityQA (2022)	O	FiD-KD	20	47.8	49.8	67.7	67.8
Our implementations							
FiD (2021b)	-	DPR	20	45.3 $\pm$ 0.31	46.3 $\pm$ 0.10	61.5 $\pm$ 0.12	62.1 $\pm$ 0.34
FiD-KD (2021a)	-		100	49.1	50.1	-	-
FiD-KD (2021a)	-		20	47.8 $\pm$ 0.16	48.4 $\pm$ 0.31	67.4 $\pm$ 0.12	67.6 $\pm$ 0.25
EvidentialityQA (2022)	O	FiD-KD	20	48.0 $\pm$ 0.20	49.0 $\pm$ 0.39	n/a	n/a
RFiD (2023)	O		100	49.2	50.4	-	-
RFiD (2023)	O		20	48.6 $\pm$ 0.29	49.4 $\pm$ 0.53	67.8 $\pm$ 0.12	68.1 $\pm$ 0.20
MGFiD	O	FiD-KD	20	**49.0** $\pm$ 0.21	**50.1** $\pm$ 0.33	**68.0** $\pm$ 0.09	**68.3** $\pm$ 0.23
Pruned MGFiD (τ=0.05)	O		4.8 / 7.7	48.8 $\pm$ 0.20	49.7 $\pm$ 0.52	67.8 $\pm$ 0.07	**68.3** $\pm$ 0.16

Table 2: Performance comparison between MGFiD and baseline models. Avg. # psgs in Decoder for Pruned MGFiD is the average number of passages passed to the decoder in NQ / TQA, respectively. $\pm$ indicates the standard deviation of 5 runs. The best result among the models using $K = 20$, which is the number of retrieved passages used in the encoder, is marked **bold**, and the second best is underlined.

report the average of five runs with the same seed set with MGFiD. For EvidentialityQA [4] (2022), we observed a technical issue with the TQA dataset in the official repository, where all evidence labels were incorrectly marked as 0. On the NQ test set, we got results that were lower than the original paper, while we got slightly better results on dev. Considering the standard deviation, we consider this to be a valid reproduction. FiD, FiD-KD [1] (2021b; 2021a), and RFiD [5] (2023) originally used K as 100, but for a fair comparison, we trained them using 20 after validating reproducibility.

5 Results and Analysis

5.1 Main Results

Table 2 shows the effectiveness of our model with the baseline models on the NQ and TQA datasets. We report the results of our model and four replications averaged over five seeds, along with their standard deviations. All models in this experiment are initialized with T5-base (Raffel et al., 2020). Note that MGFiD incorporates components to discriminate evidence, consisting merely of only a few MLP layers, which marginally increases the number of parameters by less than 1% from the backbone model. Avg. # psgs in Decoder, which is the number of passages passed to the decoder, is identical with the number of retrieved passages using in the encoder, *i.e.*, top-K, except for $K = 20$ for Pruned MGFiD and $K = 100$ for KG-FiD (Yu et al., 2022).

First, MGFiD significantly improves over the baseline models using the same retriever and the same number of passages. Compared to the original model, FiD-KD (Izacard and Grave, 2021a), which only performs answer generation task, MGFiD improves the EM score on the test set by 3.5% on the NQ and 1.0% on the TQA, and is comparable to FiD-KD using 100 passages on the NQ dataset. This implies that MGFiD identifying evidence in the multi-granularity approach effectively guides the model into supportive passages to the question.

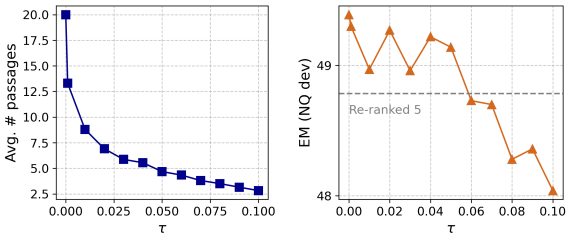
Second, EvidentialityQA (Asai et al., 2022) and RFiD (Wang et al., 2023) show improved performance compared to models without multi-task learning. This implies that determining evidentiality among passages enhances the quality of answer generation. Additionally, MGFiD further improves this process by integrating fine-grained, sentence-level evidence, demonstrating an improvement of 2.2% and 1.4% on the NQ test set over EvidentialityQA and RFiD, respectively.

Third, MGFiD using passage pruning significantly reduces the number of passages used by 76% on the NQ and 61.5% on the TQA, lowering the number of passages to 4.8 and 7.7 passages. The key-value matrix in the decoder, as noted in FiD-

[4] https://github.com/AkariAsai/evidentiality_qa
[5] https://github.com/wangcunxiang/RFiD

Model	R@1	AUC
DPR	49.9	-
FiD-KD (cross-attention)	58.6	-
MGFiD (Passage ranker)	**62.2**	**0.82**
* w/ Cross-entropy $\mathcal{L}_{\text{sentence}}$	60.9	0.70
* w/o $\mathcal{L}_{\text{sentence}}$	61.7	-

Table 3: Effectiveness of the proposed method for ranking and classification tasks on the NQ dev dataset. We report MGFiD trained with labels generated by MythoMax. The AUC metric is only reported for MGFiD variants that include the sentence classifier.



(a) (b)

Figure 5: (a) The average number of passages provided to the decoder as a function of τ. (b) The effectiveness of varying τ. We utilized the NQ dev dataset and the best checkpoint of MGFiD. When $\tau = 0.05$, MGFiD significantly outperforms using a constant number of 5 re-ranked passages with fewer passages.

Light (Hofstätter et al., 2023), is the most resource-intensive part of FiD. Despite passage pruning in MGFiD, there is only a decrease of less than 1% in performance, indicating effective pruning of irrelevant passages. Furthermore, it maintains or even improves performance compared to other baseline models on both the NQ and TQA datasets.

5.2 In-depth Analysis

Ranking & classification performance. In Table 3, we measured the outputs of the evidence ranker and sentence classifier as Recall and AUC score, to evaluate our model's ability to identify evidence paragraphs and supporting sentences. (i) The passage ranking score can be implicitly measured by the decoder's cross-attention score. The cross-attention score of each document is calculated by summing the cross-attention scores of the tokens. In this way, the Recall@1 score improved by 17.2% compared to the DPR retriever. (ii) The improvement is even higher for the passage ranker with explicit ranking capability. When trained with MythoMax labels, it shows an outstanding 5.1% improvement over the re-ranking result using the cross-attention score in FiD. This suggests that it is more effective to add a module that specializes in determining evidence rather than relying on the cross-attention of the decoder. (iii) When sentence classification is applied, Recall@1 improves even more. Using the focal loss for better classification of imbalanced sentence labels, the AUC score improved to 0.82, and the Recall@1 score reached 62.2. This suggests that emphasizing the embedding of important sentences also helps to distinguish supportive passages.

$\mathcal{L}_{\text{ranking}}$	$\mathcal{L}_{\text{sent}}$	e_{anchor}	τ	NQ	TQA
✓	✓	✓	0.05	49.1	67.7
✓	✓	✓	*top*-5	48.8	-
listwise	✓	✓	×	**49.4**	67.9
listwise	✓	×	×	48.9	67.9
×	✓	×	×	48.1	67.9
listwise	×	×	×	48.8	67.8
pointwise	×	×	×	48.3	67.6
×	×	×	×	47.8	67.5

Table 4: Ablation study on the impact of multi-task learning and Threshold-based masking. Note that we are reporting for seed 0 in this result.

Efficiency via passage pruning. Figure 5 shows the number of passages used by the decoder and the effectiveness depending on the pruning threshold τ. It takes all 20 passages when no pruning is applied, *i.e.*, $\tau = 0$. Increasing τ to 0.05 results in a small performance drop even if the number of passages drops drastically below 5. It is worth noting that the performance is much higher than simply using the top-5 passages among the re-ranked passages. Since the concatenation of all encoded tokens causes high computational cost (Hofstätter et al., 2023), it helps to avoid a significant performance drop while reducing the decoding overhead.

Ablation study. Table 4 shows an ablation study on our different methods. (i) Listwise loss for multi-task learning achieves 0.5%p higher accuracy than the point-wise loss on the NQ dataset. This implies that listwise is a more reasonable approach due to the structure of FiD, which concatenates multiple passage embeddings and utilizes them at once. (ii) The anchor vector provides core sentence-level information by adding up the anchor vector to the [BOS] token. With this additional information, the accuracy improved by 0.2%p on the NQ dataset. (iii) When $\mathcal{L}_{\text{passage}}$ and $\mathcal{L}_{\text{sentence}}$ are used, the accuracy reaches a peak of 49.4 and 67.9 for the NQ and TQA, respectively. This suggests that multi-granularity can help performance by obtain-

Evidence label	NQ dev (EM)	# pos.	TQA dev (EM)	# pos.
-	47.8 ± 0.16	-	67.4 ± 0.12	-
Ans. span	48.5 ± 0.21	4.5	67.7 ± 0.16	8.9
ChatGPT	**48.9** ± 0.13	4.0	67.7 ± 0.20	8.3
MythoMax	48.8 ± 0.23	2.8	**67.8** ± 0.18	6.5

Table 5: Model performance with different evidence labels. # pos. denotes the average number of positive labels in top-20 passages.

ing more evidentiality. (iv) When τ is 0.05, the average number of passages used in the decoder is 4.8 in the NQ dataset. Pruned MGFiD gets 0.3%p higher than using the fixed top-5 re-ranked passages, suggesting that it is more effective to utilize only the supportive passages for each question.

5.3 Effectiveness of Evidence Labels

Table 5 shows the experiment results of FiD with only passage-level evidence learning, *i.e.*, $\mathcal{L}_{\text{passage}}$. We use three different labels for passage re-ranking: Ans. span, which checks if the answer span is included, and labels filtered by ChatGPT, and MythoMax. (i) FiD without multi-task learning significantly underperforms on both datasets compared to the others, trained with the additional passage re-ranking. These results suggest that it is insufficient to implicitly let the reader determine evidence without additional ranking information. (ii) The models trained with LLM-generated labels for passage re-ranking outperform those trained with answer span presence as a label, improving by up to 0.4 on the NQ dataset. This suggests that mislabeled spurious passages act as noisy data when the answer spans are used as a determinant of labeling, thereby leading to sub-optimal results. (iii) We note that the performance using the MythoMax label is not significantly different from the performance using the ChatGPT label. This suggests that our framework can effectively determine evidence regardless of the size of the LLMs.

5.4 Effectiveness by the Number of Passages

Figure 6 illustrates the performance of MGFiD and two baseline models on the NQ and TQA test sets with varying numbers of passages used by the encoder, *i.e.*, K. We trained each model using the top-K passages. Our findings reveal that performance is enhanced with more passages for all models, aligning with the aggregating capability of the FiD architecture noted by Izacard and Grave (2021b). Second, MGFiD consistently outperforms

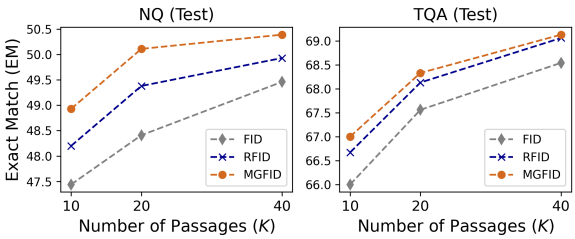


Figure 6: Effectiveness of FiD-KD (Izacard and Grave, 2021a), RFiD (Wang et al., 2023), and MGFiD varying the number of passages used in the encoder, *i.e.*, K.

the baselines across different numbers of passages (10, 20, and 40), highlighting the significance of the capability to discern supportive passages. Lastly, the efficacy of evidence-based multi-task learning, as utilized by MGFiD and RFiD (Wang et al., 2023), is most significant with fewer documents, *i.e.*, $K = 10$. This observation is counterintuitive to the expectation that filtering spurious passages becomes more critical as the number of passages increases. We interpret this to suggest that increasing the number of passages may have a similar effect as increasing the batch size (Qu et al., 2021), whereas the multi-task learning can efficiently achieve high performance even with smaller batch sizes. We leave a more detailed analysis to future work.

6 Conclusion

This paper presents the Multi-Granularity Guided Fusion-in-Decoder (MGFiD), a novel reader for managing evidence across multiple granularities. Addressing the prevalent challenges of misleading passages and sentences, MGFiD synergies coarse-level passage re-ranking with fine-level sentence classification. We also incorporate LLMs to enhance the quality of heuristic labels. Moreover, MGFiD capitalizes on its multi-granularity evidence by constructing an anchor vector that guides the decoder toward significant evidence and employs passage pruning to enhance decoding efficiency. Our empirical results demonstrate that MGFiD using multi-granularity contexts achieves significant advancements over baseline models.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00421, 2022-0-00680-003, 2022-0-01045, and RS-2023-00219919).

Limitations

We briefly describe the limitations of our method. (i) LLM filtering methods are limited to extractive QA for the current setting. (ii) There needs to be validation on more passages. (iii) Marginal improvement on TQA dataset.

Limitation of LLM labels. Because our label filtering method is based on answer span, it is still quite limited to the extractive task. However, the criterion for silver labels is not necessarily answer span, and we have shown in the paper that the filtering task does not necessarily require expensive models. This means that for relatively low K, it is available to perform on all the retrieved results. The fact that harsh filtering by MythoMax worked even with fewer labels means that the multi-task does not necessarily require many labels.

A large number of passages. We do not report results using a large number of passages, *e.g.*, 100, and a bigger backbone model, *i.e.*, T5-large, due to the computational cost. Previous research has shown that using more passages increases the probability that the passage set contains evidence and thus improves performance. We also found in our experiments that the standard deviation of the NQ dataset is large, depending on the seed. This was true for all of the baseline models we reproduced. Although we compared our model and the baseline with five seeds, it would be desirable to validate additional seeds to further examine generalizability.

Marginal improvement on TQA. The performance improvement on TQA is marginal compared to that of NQ. We can assume that the multi-task learning to identify supportive context is less effective for TQA because it has numerous answer candidates and passages regarding evidence are present. It is thus relatively easy to get an EM score. However, we still need to analyze this further.

Ethics Statement

The ethical guidelines of ACL are fully respected in this work. We have utilized scientific resources available for research under liberal licenses. Our use of these tools is consistent with their intended applications.

References

Akari Asai, Matt Gardner, and Hannaneh Hajishirzi. 2022. Evidentiality-guided generation for knowledge-intensive NLP tasks. In *NAACL*, pages 2226–2243.

Andrew P. Bradley. 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.*, 30:1145–1159.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *ACL*, pages 1870–1879.

Michiel de Jong, Yury Zemlyanskiy, Joshua Ainslie, Nicholas FitzGerald, Sumit Sanghai, Fei Sha, and William W. Cohen. 2023. Fido: Fusion-in-decoder optimized for stronger performance and faster inference. In *Findings of the ACL*, pages 11534–11547.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *ICML*, pages 3929–3938.

Sebastian Hofstätter, Jiecao Chen, Karthik Raman, and Hamed Zamani. 2023. Fid-light: Efficient and effective retrieval-augmented text generation. In *SIGIR*, pages 1437–1447.

Gautier Izacard and Edouard Grave. 2021a. Distilling knowledge from reader to retriever for question answering. In *ICLR*.

Gautier Izacard and Edouard Grave. 2021b. Leveraging passage retrieval with generative models for open domain question answering. In *EACL*, pages 874–880.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, pages 1601–1611.

Mingxuan Ju, Wenhao Yu, Tong Zhao, Chuxu Zhang, and Yanfang Ye. 2022. Grape: Knowledge graph enhanced passage reader for open-domain question answering. In *Findings of EMNLP*, pages 169–181.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP*, pages 6769–6781.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. 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. *Trans. Assoc. Comput. Linguistics*, pages 452–466.

Kushal Lakhotia, Bhargavi Paranjape, Asish Ghoshal, Scott Yih, Yashar Mehdad, and Srini Iyer. 2021. Fid-ex: Improving sequence-to-sequence models for extractive rationale generation. In *EMNLP*, pages 3712–3727.

Haejun Lee, Akhil Kedia, Jongwon Lee, Ashwin Paranjape, Christopher D. Manning, and Kyoung-Gu Woo. 2022. You only need one model for open-domain question answering. In *EMNLP*, pages 3047–3060.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *ACL*, pages 6086–6096.

Patrick S. H. 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. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *NeurIPS*.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *ICCV*, pages 2999–3007.

Meizhen Liu, Jiakai He, Xu Guo, Jianye Chen, Siu Cheung Hui, and Fengyu Zhou. 2023. Grancats: Cross-lingual enhancement through granularity-specific contrastive adapters. In *CIKM*, pages 1461–1471.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *CoRR*.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In *NAACL-HLT*, pages 5835–5847.

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. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agents. In *EMNLP*, pages 14918–14937.

Wiebke Wagner. 2010. Steven bird, ewan klein and edward loper: Natural language processing with python, analyzing text with the natural language toolkit - o'reilly media, beijing, 2009, ISBN 978-0-596-51649-9. *Lang. Resour. Evaluation*, 44:421–424.

Cunxiang Wang, Haofei Yu, and Yue Zhang. 2023. Rfid: Towards rational fusion-in-decoder for open-domain question answering. In *Findings of ACL*, pages 2473–2481.

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 *ACL*, pages 4961–4974.

A Appendix

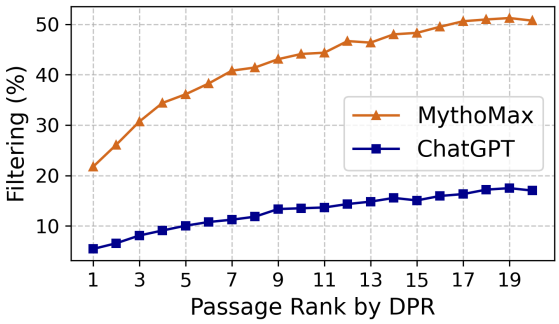


Figure 7: Filtering percentage by rank. Both MythoMax and ChatGPT show more than 10% and 40% filtering ratios at top-10 ranking results. This suggests that both systems are doing the task reasonably, as the rankings in DPR are likely related to how well the content semantically matches.

A.1 Evaluation on label filtering

Assuming that the rank provided by the retriever (Izacard and Grave, 2021a) represents the contextual relevance of a query to a paragraph, it is reasonable to expect the distribution of desirable supporting passages in the top 20 documents to be asymmetric, with dense at the high ranks and sparse at the low ranks. Figure 7 shows the percentage of passages filtered out (labeled as irrelevant) when passages corresponding to each DPR rank are given to Mythomax and ChatGPT along with a question. As we expected, both models are more likely to label rank20 passages as irrelevant than rank1 passages. ChatGPT labels very few passages as irrelevant at rank 1, but this increases to almost 20% as the rank decreases. Mythomax labels over 50% of passages as irrelevant at low rank. This empirically verifies that LLM's label filtering tendency is consistent with the contextual relevance across ranks.

A.2 Importance of Sentence-level Evidence

Existing works only identify which passages are supporting and focus on aggregating evidence across multiple passages. Still, there is a lot of information in the passages that can mislead the model. Figure 8 shows that a model trained only on identifying supporting passages, *i.e.*, w/o multi-granularity learning, generated incorrect answers. On the other hand, MGFiD, which learned sentence-level evidence, avoided plausible incorrect answers and generated correct answers.

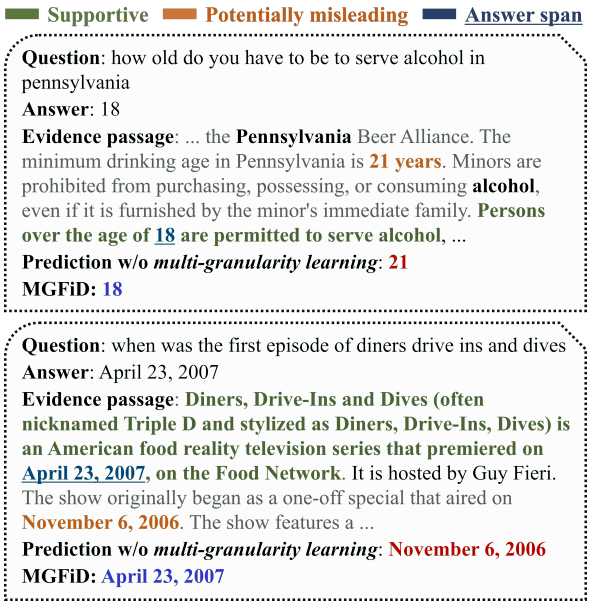


Figure 8: More examples that can harm QA systems similar to Figure 1. Two examples show the need to identify which sentence is supportive and which is not. Black **bold** terms in the passages are overlapped with the question.