Dense 𝕏 Retrieval: What Retrieval Granularity Should We Use?

Tong Chen[♣*] **Hongwei Wang**[♢] **Sihao Chen**[♡] **Wenhao Yu**[♢]
Kaixin Ma[♢] **Xinran Zhao**[♠] **Hongming Zhang**[♢] **Dong Yu**[♢]

[♣]University of Washington [♢]Tencent AI Lab
[♡]University of Pennsylvania [♠]Carnegie Mellon University

Abstract

Dense retrieval has become a prominent method to obtain relevant context or world knowledge in open-domain NLP tasks. When we use a learned dense retriever on a retrieval corpus at *inference* time, an often-overlooked design choice is the *retrieval unit* in which the corpus is indexed, e.g. document, passage, or sentence. We discover that the retrieval unit choice significantly impacts the performance of both retrieval and downstream tasks. Distinct from the typical approach of using passages or sentences, we introduce a novel retrieval unit, *proposition*, for dense retrieval. Propositions are defined as atomic expressions within text, each encapsulating a distinct factoid and presented in a concise, self-contained natural language format. We conduct an empirical comparison of different retrieval granularity. Our results reveal that proposition-based retrieval significantly outperforms traditional passage or sentence-based methods in dense retrieval. Moreover, retrieval by proposition also enhances the performance of downstream QA tasks, since the retrieved texts are more condensed with question-relevant information, reducing the need for lengthy input tokens and minimizing the inclusion of extraneous, irrelevant information.

Question: What is the angle of the Tower of Pisa?	
Passage Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees. This means the top of the Leaning Tower of Pisa is displaced horizontally 3.9 meters (12 ft 10 in) from the center.
Sentence Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees.
Proposition Retrieval	The Leaning Tower of Pisa now leans at about 3.99 degrees.

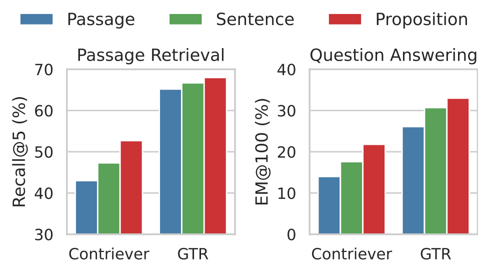


Figure 1: (*Top*) An example of three granularities of retrieval units of Wikipedia text when using dense retrieval. (*Bottom*) We observe that retrieving by propositions yields the best retrieval performance in both passage retrieval task and downstream open-domain QA task, e.g. with Contriever (Izacard et al., 2022) or GTR (Ni et al., 2022) as the backbone retriever. Highlight indicates the part that contains answer to the question.

1 Introduction

Dense retrievers are a popular class of techniques for accessing external information sources for knowledge-intensive tasks (Karpukhin et al., 2020). Before we use a *learned* dense retriever to retrieve from a corpus, an imperative design decision we have to make is the *retrieval unit* – i.e. the granularity at which we segment and index the retrieval corpus for inference. In practice, the choice of retrieval unit, e.g. documents, fixed-length passage chunks or sentences, etc, is usually pre-determined based on how the dense retrieval model is instantiated or trained (Lewis et al., 2020; Lee et al., 2021a; Santhanam et al., 2022; Ni et al., 2022).

In this paper, we investigate an overlooked research question with dense retrieval *inference* – at what retrieval granularity should we segment and index the retrieval corpus? We discover that selecting the *proper* retrieval granularity at inference time can be a simple yet effective strategy for im-

knowledge corpus which is chunked into passages/sentences

[*] Work was done during internship at Tencent AI Lab, Bellevue.

https://github.com/ct123098/factoid-wiki

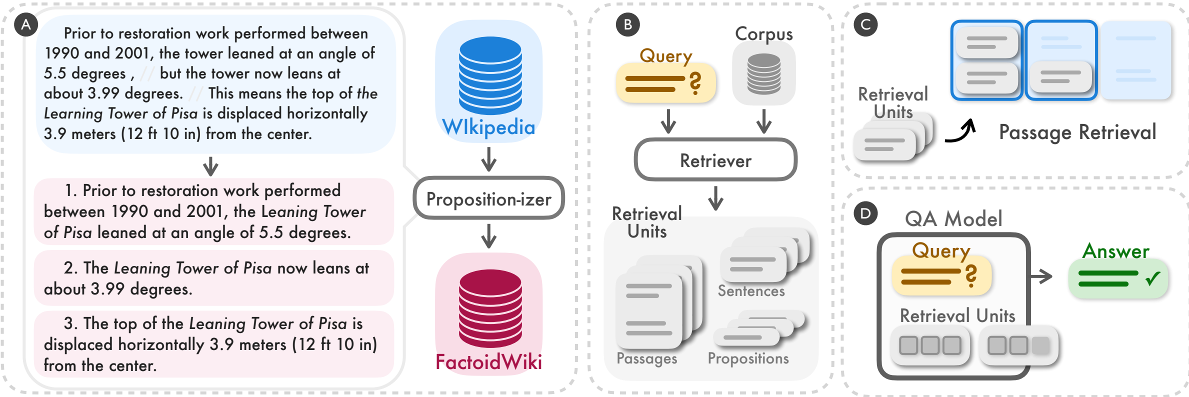


Figure 2: We discover that segmenting and indexing a retrieval corpus on the *proposition* level can be a simple yet effective strategy to increase dense retrievers' generalization performance at inference time *(A, B)*. We empirically compare the retrieval and downstream open-domain QA tasks performance when dense retrievers work with Wikipedia indexed at the level of 100-word passage, sentence or proposition *(C, D)*.

proving dense retrievers' retrieval and downstream task performance. We illustrate our intuition with an example of open-domain question-answering (QA) in Table 1. The example shows retrieved text by the same model at three different granularities. The *passage*, which represents a coarser retrieval unit with a longer context, is theoretically able to provide more relevant information for the question. However, a passage often includes extraneous details (e.g., restoration period and horizontal displacement in the example of Table 1) that could potentially distract both the retriever and the language model in downstream tasks (Shi et al., 2023; Yu et al., 2023b). On the other hand, *sentence*-level indexing provides a finer-grained approach but does not entirely address the issue (Akkalyoncu Yilmaz et al., 2019; Yang et al., 2020). This is because sentences can still be complex and compounded, and they are often not self-contained, lacking necessary contextual information (e.g., in the example of Table 1, "the tower" is coreference of "Pisa Tower") for judging the query-document relevance.

Passage :

- contains additional details which might deviate the retriever or downstream models from extracting the exact response

Sentence :

- contains very precise contents thus might loose the contextual information that can help retriever to retrieve the correct document

To address these shortcomings of typical retrieval units such as passages or sentences, we propose using *proposition* as a novel retrieval unit for dense retrieval. Propositions are defined as atomic expressions within text, each encapsulating a distinct factoid and presented in a concise, self-contained natural language format. We show an example proposition in Table 1. The proposition describes the information regarding the Tower of Pisa's current leaning angle in a self-contained way and precisely responds to what the question is querying. We provide a more detailed definition and description of *proposition* in §2.

To validate the efficacy of using proposition as a retrieval unit for dense retrievers inference, we first process and index an English Wikipedia dump with all documents segmented into propositions, which we refer to as FACTOIDWIKI. Then we conduct experiments on five different open-domain QA datasets and empirically compare the performance of six dual-encoder retrievers when Wikipedia is indexed by passage, sentence, and our proposed proposition. Our evaluation is twofold: we examine both the retrieval performance and the impact on downstream QA tasks. Notably, our findings indicate that proposition-based retrieval outperforms sentence and passage-based methods, especially in terms of generalization, as discussed in §5. This suggests that propositions, being both compact and rich in context, enable dense retrievers to access precise information while maintaining adequate context. The average improvement over passage-based retrieval of Recall@20 is +10.1 on unsupervised dense retrievers and +2.2 on supervised retrievers. Furthermore, we observe a distinct advantage in downstream QA performance when using proposition-based retrieval, as elaborated in §6. Given the often limited input token length in language models, propositions inherently provide a higher density of question-relevant information.

- for eg. if the retrieval context is huge then token length issues can cause loss of certain information

Our main contributions in the paper are:

- We propose using propositions as retrieval units when indexing a retrieval corpus to improve the dense retrieval performance.
- We introduce FACTOIDWIKI, a processed English Wikipedia dump, where each page is segmented into multiple granularities: 100-word passages, sentences and propositions.

- We discover that retrieval by proposition outperforms passage or sentence retrieval in terms of generalization for passage retrieval and accuracy for downstream question-answering given the same input token limit.

2 Proposition as a Retrieval Unit

The goal of our study is to understand how the granularity of a retrieval corpus influences the dense retrieval models' performance empirically. Aside from commonly-used retrieval units such as 100-word passage (Karpukhin et al., 2020) or sentence, we propose using *proposition* as an alternative retrieval unit choice. Here, propositions represent atomic expressions of meanings in text (Min et al., 2023) that are defined by the three principles below.

1. Each proposition should correspond to a distinct piece of meaning in text, where the composition of all propositions would represent the semantics of the entire text.
2. A proposition should be *minimal*, i.e. it cannot be further split into separate propositions.
3. A proposition should be *contextualized and self-contained* (Choi et al., 2021). A proposition should include all the necessary context from the text (e.g. coreference) to interpret its meaning.

The use of proposition as a retrieval unit is inspired by a recent line of work (Min et al., 2023; Kamoi et al., 2023; Chen et al., 2023a,b), which finds success in representing and evaluating text semantics at the level of propositions. We demonstrate the concept of proposition and how a passage can be split into its set of propositions by an example on the left side of Figure 2. The passage contains three propositions, each of which corresponds to a distinct factoid about the *Leaning Tower of Pisa*: the angle before the restoration, the current angle, and the horizontal displacement. Within each proposition, necessary context from the passage is incorporated so that the meaning of the proposition can be interpreted independently of the original text, e.g. the reference of *the tower* is resolved into its full mention, *the Leaning Tower of Pisa*, in the first proposition. We expect each proposition to describe exactly one contextualized atomic fact, and so our intuition is that propositions would suitably work as a retrieval unit for information-seeking questions.

3 FactoidWiki: Proposition-Level Index and Retrieval for Wikipedia

We empirically compare the use of 100-word passages, sentences, and propositions as retrieval units on Wikipedia, a commonly-used retrieval source for knowledge-intensive NLP tasks (Petroni et al., 2021). To allow for a fair comparison across granularities, we process an English Wikipedia dump from 2021-10-13, as used by Bohnet et al. (2022). We segment each document text into three different granularities: 100-word passages, sentences, and propositions. We include the details on passage- and sentence-level segmentation of the corpus in Appendix A.

Parsing Passage to Propositions. To segment the Wikipedia pages into propositions, we finetune a text generation model, which we refer to as the *Propositionizer*. The *Propositionizer* takes a passage as input and generates the list of propositions within the passage. Following Chen et al. (2023b), we train the *Propositionizer* with a two-step distillation process. We first prompt GPT-4 (OpenAI, 2023) with an instruction containing the proposition definition and 1-shot demonstrations. We include the details of the prompt in Figure 8. We start with a set of 42k passages and use GPT-4 to generate the seed set of paragraph-to-propositions pairs. Next, we use the seed set to finetune a Flan-T5-large model (Chung et al., 2022).

We refer to the processed corpus as FactoidWiki. The resulting statistics of FactoidWiki are shown in Table 1.

	# units	Avg. # words
Passage	41,393,528	58.5
Sentence	114,219,127	21.0
Proposition	256,885,003	11.2

Table 1: Statistics of text units in the English Wikipedia dump from 2021-10-13.

4 Experimental Settings

To evaluate the impact of the three retrieval unit choices, we conduct experiments on five different open-domain QA datasets with FactoidWiki. With each dataset, we evaluate both passage retrieval and downstream QA performance when dense retrievers work with Wikipedia indexed in different granularities.

Passage level granularity - use retriever to retrieve the top k matching passages
Sentence level granularity - use retriever to retrieve say top m sentences and then map these m sentences to their respective passages.
Assume we get n unique passages during this mapping then retrieving top k unique passages from those n unique passages
Proposition level granularity - same as above except granularity is proposition

4.1 Open-Domain QA Datasets

We evaluate on five different open-domain QA datasets with Wikipedia as the retrieval source: Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQA) (Joshi et al., 2017), Web Questions (WebQ) (Berant et al., 2013), SQuAD (Rajpurkar et al., 2016), and Entity Questions (EQ) (Sciavolino et al., 2021).

4.2 Dense Retrieval Models

We compare the performance of the six following supervised or unsupervised dense retriever models. Here, *supervised* models refer to ones that have used human-labeled query-passage pairs as supervision during training, and vice versa.

- **SimCSE** (Gao et al., 2021) is a BERT-base (Devlin et al., 2019) encoder trained on unlabeled sentence randomly sampled from Wikipedia.
- **Contriever** (Izacard et al., 2022) is an unsupervised retriever, instantiated with a BERT-base encoder. Contriever is contrastively trained by segment pairs constructed from unlabeled documents from Wikipedia and web crawl data.
- **DPR** (Karpukhin et al., 2020) is a dual-encoder BERT-base model finetuned on five open-domain QA datasets, which includes four of the datasets (NQ, TQA, WebQ and SQuAD) in our evaluation.
- **ANCE** (Xiong et al., 2020) used the same setting from DPR and trained the model with Approximate nearest neighbor Negative Contrastive Estimation (ANCE), a hard negatives-based training approach.
- **TAS-B** (Hofstätter et al., 2021) is a dual-encoder BERT-base model distilled from a teacher model with cross-attention trained on MS MARCO (Nguyen et al., 2016).
- **GTR** (Hofstätter et al., 2021) is a T5-base encoder (Raffel et al., 2020) pretrained on unlabeled pairs of online forum QA data, and fine-tuned on MS MARCO and Natural Question.

4.3 Passage Retrieval Evaluation

We evaluate retrieval performance at the passage level when the corpus is indexed at the passage, sentence, or proposition level respectively. For sentence and proposition level retrieval, we follow the setting introduced in Lee et al. (2021b), where the score of the passage is based on the maximum similarity score between the query and all sentences or propositions in a passage. In practice, we first retrieve a slightly larger number of text units, map each unit to the source passage, and return the top-k unique passages. We use Recall@k as our evaluation metric, which is defined as the percentage of questions for which the correct answer is found within the top-k retrieved passages.

Recall@k - percentage of questions for which correct answer is present in top-k retrieved chunks. it does not consider the rank of the correct answer

4.4 Downstream QA Evaluation

To understand the implications of using different retrieval units on the downstream open-domain QA tasks, we evaluate the use of retrieval models in *retrieve-then-read* setup (Izacard and Grave, 2021). With the *retrieve-then-read* setting, a retrieval model first retrieves k text units given the query. The k retrieved text units are then used as input along with the query to a reader model to derive the final answer. Typically, the choice of k is subject to the reader model's maximum input length constraint, or the limit of compute budget, which scales with the number of input tokens.

- Retrieval Augmented Generation (RAG)

For this reason, we follow an evaluation setup where the maximum number of retrieved words is capped at $l = 100$ or 500, i.e. only the top l words from passage, sentence, or proposition level retrieval are feed into the reader model as input. We evaluate the percentage of questions for which the predicted answer exactly matches (EM) the ground truth. We denote our metric as EM @ l. For our evaluation, we use T5-large size UnifiedQA-v2 as the reader model (Khashabi et al., 2022).

5 Results: Passage Retrieval

In this section, we report and discuss the retrieval tasks performance. Our results show that despite none of the models being trained with proposition-level data, all the retrieval models demonstrated on-par or superior performance when the corpus is indexed at the proposition level.

5.1 Passage Retrieval Performance

We report our evaluation results in Table 2. We observe that retrieval by propositions outperforms retrieval by sentence or passage on most tasks for both unsupervised and supervised retrievers.

With all dense retrievers tested, proposition-level retrieval consistently outperforms sentence and passage-level retrieval on average across the five datasets. With the *unsupervised* retrievers, i.e. SimCSE and Contriever, we see an averaged Recall@5 improvement of +12.0 and +9.3 (35.0%

Retriever	Granularity	NQ R@5	NQ R@20	TQA R@5	TQA R@20	WebQ R@5	WebQ R@20	SQuAD R@5	SQuAD R@20	EQ R@5	EQ R@20	Avg. R@5	Avg. R@20
Unsupervised Dense Retrievers													
SimCSE	Passage	28.8	44.3	44.9	59.4	39.8	56.0	29.5	45.5	28.4	40.3	34.3	49.1
	Sentence	35.5	53.1	50.5	64.3	45.3	64.1	37.1	52.3	36.3	50.1	40.9	56.8
	Proposition	**41.1**	**58.9**	**52.4**	**66.5**	**50.0**	**66.8**	**38.7**	**53.9**	**49.5**	**62.2**	**46.3**	**61.7**
Contriever	Passage	42.5	63.8	58.1	73.7	37.1	60.6	40.8	59.8	36.3	56.3	43.0	62.8
	Sentence	46.4	66.8	60.6	75.7	41.7	63.1	45.1	63.5	42.7	61.3	47.3	66.1
	Proposition	**50.1**	**70.0**	**65.1**	**77.9**	**45.9**	**66.8**	**50.7**	**67.7**	**51.7**	**70.1**	**52.7**	**70.5**
Supervised Dense Retrievers													
DPR	Passage	**66.0**	**78.0**	71.6	80.2	62.9	74.9	38.3	53.9	47.5	60.4	57.3	69.5
	Sentence	**66.0**	**78.0**	**71.8**	**80.5**	**64.1**	74.4	40.3	55.9	53.7	66.0	59.2	71.0
	Proposition	65.4	77.7	70.7	79.6	62.8	**75.1**	**41.4**	**57.2**	**59.4**	**71.3**	**59.9**	**72.2**
ANCE	Passage	**70.7**	81.4	**73.9**	81.4	**65.7**	77.2	43.3	58.6	57.0	69.1	62.1	73.5
	Sentence	70.3	**81.6**	**73.9**	**81.5**	65.2	**77.4**	45.8	60.7	61.4	72.8	63.3	74.8
	Proposition	69.9	81.1	72.8	80.6	65.1	77.1	**46.2**	**61.9**	**66.7**	**76.6**	**64.1**	**75.5**
TAS-B	Passage	**64.2**	77.9	70.4	79.3	**65.1**	77.0	54.3	69.2	72.2	81.3	65.2	76.9
	Sentence	64.0	78.4	**71.4**	**80.2**	63.9	76.7	58.9	72.3	72.7	82.0	66.2	77.9
	Proposition	63.8	**78.6**	**71.4**	80.0	63.8	**76.8**	**59.8**	**73.4**	**75.1**	**83.3**	**66.8**	**78.4**
GTR	Passage	66.3	78.4	70.1	79.4	**63.3**	76.5	54.4	68.1	71.7	80.5	65.2	76.6
	Sentence	66.4	79.4	71.6	**80.9**	62.2	76.8	60.9	73.4	72.5	81.3	66.7	78.4
	Proposition	**66.5**	**79.6**	**72.2**	**80.9**	63.2	**77.4**	**63.3**	**75.0**	**74.9**	**83.0**	**68.0**	**79.2**

dataset not part of training ANCE and DPR

Table 2: Passage retrieval performance (Recall@k = 5, 20) on five different open-domain QA datasets when pre-trained dense retrievers work with the three different granularity from the retrieval corpus. Underline denotes cases where the training split of the target dataset was included in the training data of the dense retriever.

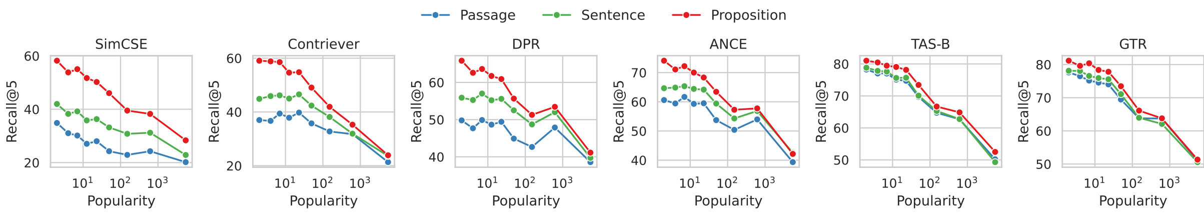


Figure 3: Document retrieval recall vs. the popularity of the target entity in each question from the *EntityQuestions* dataset. The popularity of each entity (i.e. smaller value ⇒ less common entities, and vice versa) is estimated by the occurrence of the entity in its top-1000 passage retrieved by BM25. On queries with less common entities, we observe that retrieving by proposition shows a larger advantage over retrieval by proposition.

- when the popularity of an entity is less, using passage level retrieval granularity finds difficult to retrieve the relevant passages compared to proposition level retrieval. this might be attributed to the more relevant information contained when using a proposition

- when the popularity of an entity increases the recall of both passage and proposition level retrieval drops but still the proposition level retrieval stays ahead of passage level retrieval. the drop might be attributed to presence of the entity more frequently in all the top-1000 chunks

- the recall@5 of DPR and ANCE models are greater compared to SimCSE and Contriever as they have been finetuned on

and 22.5% relative improvement) respectively over five datasets.

With the *supervised* retrievers, proposition-level retrieval still shows an advantage on average, yet the sizes of improvements are smaller. We hypothesize that this is due to these retrievers being trained on query-passage pairs. For instance, with DPR and ANCE, which have been trained on NQ, TQA, WebQ, and SQuAD, we observe that proposition and sentence level retrieval perform slightly worse compared to passage level on three out of the four datasets, with the exception of SQuAD. As shown in Table 2, all supervised retrievers demonstrate comparable performance across three levels of retrieval granularity in NQ, TQA, and WebQ.

However, on datasets that the retriever model has *not* seen during training, we observe that retrieval by proposition demonstrates a clear advantage. For instance, most notably on SQuAD or EntityQuestions, we observe that proposition-based retrieval significantly outperforms the other two granularities. We see 17-25% Recall@5 relative improvement on EntityQuestions with relatively weak retrievers like DPR and ANCE. Furthermore, the Recall@5 of retrieval by proposition on SQuAD improved most on TAS-B and GTR, with 10-16% relative improvements.

5.2 Retrieval by Proposition ⇒ Better Cross-Task Generalization

Our results indicate that the advantage of retrieval by proposition becomes most visible in cross-task generalization settings. We observe that on SQuAD and EntityQuestions, retrieval by proposi-

Retriever	Granularity	NQ EM @100	NQ EM @500	TQA EM @100	TQA EM @500	WebQ EM @100	WebQ EM @500	SQuAD EM @100	SQuAD EM @500	EQ EM @100	EQ EM @500	Avg. EM @100	Avg. EM @500
Unsupervised Dense Retrievers													
SimCSE	Passage	8.1	16.3	22.6	33.7	7.7	14.9	9.8	17.8	10.9	17.5	11.8	20.0
	Sentence	10.1	18.0	27.2	37.2	9.6	15.6	17.3	24.8	13.0	19.8	15.4	23.1
	Proposition	**12.7**	**20.2**	**28.4**	**37.7**	**11.2**	**17.2**	**18.0**	**25.1**	**18.3**	**25.0**	**17.7**	**25.0**
Contriever	Passage	11.1	22.4	25.7	41.4	6.8	14.9	15.6	27.7	10.9	21.5	14.0	25.6
	Sentence	13.8	23.9	30.5	44.2	9.1	17.2	22.6	32.8	12.2	22.2	17.6	28.1
	Proposition	**16.5**	**26.1**	**37.7**	**48.7**	**13.3**	**19.9**	**25.6**	**34.4**	**16.1**	**27.3**	**21.8**	**31.3**
Supervised Dense Retrievers													
DPR	Passage	24.8	**36.1**	40.3	51.0	14.0	22.2	12.4	21.7	18.6	25.9	22.0	31.4
	Sentence	27.6	35.9	44.6	**52.8**	16.3	23.7	18.6	26.1	21.8	28.2	25.8	33.3
	Proposition	**28.3**	34.3	**45.7**	51.9	**19.0**	**23.8**	**19.8**	**26.3**	**26.3**	**31.9**	**27.8**	**33.6**
ANCE	Passage	27.1	**38.3**	43.1	53.1	15.2	23.0	15.3	26.0	23.4	31.1	24.8	34.3
	Sentence	**30.1**	37.3	47.0	**54.7**	16.6	23.8	22.9	**30.5**	25.9	32.0	28.5	**35.7**
	Proposition	29.8	37.0	**47.4**	53.5	**19.3**	**24.1**	**22.9**	30.1	**29.1**	**33.7**	**29.7**	**35.7**
TAS-B	Passage	21.1	33.9	39.3	50.5	13.1	20.7	23.9	34.6	30.9	37.3	25.7	35.4
	Sentence	24.6	33.9	43.6	**52.3**	14.4	21.4	33.8	40.5	31.4	36.1	29.6	36.8
	Proposition	**26.6**	**34.0**	**44.9**	51.8	**18.1**	**23.7**	**34.2**	**38.9**	**34.2**	**37.8**	**31.6**	**37.2**
GTR	Passage	23.4	**34.5**	38.7	49.3	13.1	20.1	23.9	33.8	31.3	36.7	26.1	34.9
	Sentence	26.8	35.1	43.9	52.2	15.9	21.6	35.6	41.3	31.3	35.1	30.7	37.1
	Proposition	**29.5**	34.4	**45.9**	**52.6**	**18.7**	**23.8**	**37.0**	**40.4**	**34.1**	**37.1**	**33.0**	**37.7**

-results show that the average EM@500 score of proposition level retrieval outperforms EM@100 score for the same. this shows that even with proposition level retrieval providing the entire context helps in getting better performance gains

Table 3: Open-domain QA performance (EM = Exact Match) under retrieve-then-read setting where the number of retrieved words to the reader QA model is limited at $l = 100$ or 500. We use UnifedQA V2 (Khashabi et al., 2022) as the reader model. The first l words from the concatenated top retrieved text unit are feed as input to the reader model. Underline denotes cases where the training split of the target dataset was included in the training data of the dense retriever. In most cases, we see better QA performance with smaller retrieval units.

tion brings more performance gain over retrieval by passage.

To better understand where the improvements can be attributed, we conduct an additional analysis on EntityQuestions. As EntityQuestions features questions targeting the properties of longer-tail entities, we study how the retrieval performance under three different granularities is affected by the *popularity* of the target entity in question, i.e. whether the entity appears frequently in Wikipedia or not. We estimate the popularity of each entity with the following method. Given the surface form of an entity, we use BM25 to retrieve the top-1000 relevant passages from Wikipedia. We use the number of occurrences of the entity in its top-1000 passages as an estimate of its popularity. With the 20,000 test queries in EntityQuestion, around 25% of the target entities have a popularity value of less or equal to 3.

Long Tail Entities:

- entities that occur very rarely in the dataset
-frequency of occurence of an entity in the Wikipedia dataset helps determine the popularity of the entity

- given an entity, relevant top-1000 passages are retrieved using BM25(lexical search)

- then the number of occurences of the particular entity in that top-1000 passages gives the popularity of the entity

Figure 3 shows the passage retrieval performance vs. the popularity of the target entity in each question. Across all 6 dense retrievers, we observe that retrieving by proposition shows a much larger advantage over retrieving by passage with questions targeting less common entities. As the popularity of entities increases, the performance gap decreases. Our findings indicate that the performance gain from retrieval by proposition can mostly be attributed to queries for long-tailed information. This echoes our observation that retrieval by proposition improves the cross-task generalization performance of dense retrievers.

Long Tailed Information :

- in search terminology, the questions that get small number of searches.
- in practical, more questions would fall under this category
- eg. what is the correct procedure to remove the angle valve in a heater - this search might occur less than ten 10 times in a month/year in a google search but most of google search questions would fall under this category

https://ahrefs.com/blog/long-tail-keywords/#:~:text=Long%2Dtail%20keywords%20are%20search,gets%20211k%20searches%20per%20month.

6 Results: Open-Domain QA

In this section, we study how the choice of retrieval granularity affects downstream open-domain QA tasks. We show that retrieval by proposition leads to strong downstream QA performance in the retrieve-then-read setting, where the number of retrieved tokens for input to the reader QA model is capped at $l = 100$ or 500 words.

6.1 QA Performance

Table 3 shows the evaluation results. Across different retrievers, we observe higher QA performance in terms of the EM@l metric on average when using propositions as the retrieval unit. The unsupervised retrievers, SimCSE and Contriever, demonstrate improvements of +5.9 and +7.8 in the EM@100 score (50% and 55% relative improvement), respectively. The supervised retrievers, DPR, ANCE, TAS-B, and GTR, improve +5.8,

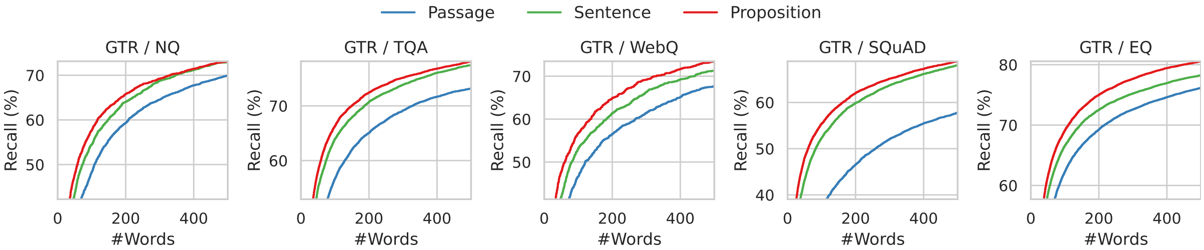


Figure 4: Recall of the gold answer in the retrieved text limited to first k words for the GTR retriever. Finer-grained retrieval has a higher recall across all numbers of words.

+4.9, +5.9, and +6.9 EM@100 (26%, 19%, 22%, 26% relative improvement), respectively. Similar to our observations from passage retrieval evaluations, we find retrieval by proposition becomes more beneficial to downstream QA performance when the retriever has not been trained on the target dataset. In other cases, retrieval by proposition still holds an advantage, but with a smaller margin on average.

6.2 Propositions ⇒ Higher Density of Question-Related Information

Intuitively, compared to sentences or passages as retrieval units, the advantage of propositions is that the retrieved propositions have a higher density of relevant information to the query. With finer-grained retrieval units, the correct answer to the query would more likely appear in the top-l retrieved words by a dense retriever.

We illustrate this phenomenon by an analysis shown in Figure 4. Here, we investigate the position at which the ground truth answer appears in the top-l retrieved words. Specifically, we calculate the recall of the gold answer within the initial l retrieved words with GTR working with Wikipedia indexed in three different granularities.

- proposition level retrieval recall@5 score significantly improves compared to recall@5 score of passage level retrieval especially when considering only 100-200 words as context.

We show the results in Figure 4 and Figure 7 with l ranging from 0 to 500 across all five datasets. For a fixed retrieved word budget, proposition retrieval demonstrates a higher success rate compared to sentence and passage retrieval methods. The most significant improvement of proposition retrieval over passage retrieval occurs within the range of 100-200 words, which corresponds to roughly 10 propositions, 5 sentences, or 2 passages. As the word count further increases, the recall rates of the three granularity converge since all question-relevant information is included in the retrieved text.

- this shows that proposition level retrieval can be a good option of granularity for downstream QA tasks that have compute and budget constraints

- with more than 200 words, the recall score@5 starts to converge as around 400 words, even passages would cover more relevant information. but this might increase the compute and budget when using it in a RAG pipeline

6.3 Error Case Study

To understand the source of errors from each type of retrieval granularity, we present and discuss four typical examples of mistakes in Table 4. With each example, we show the question and its corresponding top-1 retrieved text unit by the GTR retriever across the three granularities.

We observe that with passage-level retrieval, the ambiguity of an entity or its references presents a challenge for dense retrievers, which echoes findings from (Min et al., 2020). For instance, in example Q1, the question asks for "*Super Bowl 50*", but the retrieved passage and sentence refers to "*Super Bowl 5*". In Example Q2, passage retrieval fails to identify the part referring to the correct "*atomic number*". Instead, the top-1 retrieved passage mentions "*atomic number*" in a different and irrelevant context to the question. Retrieval by sentences can also have a similar problem as retrieval by passages like Example Q1. Also, retrieval by sentences faces another challenge of lacking context. In Example Q3, sentence-based retrieval fails as the correct sentence in the retrieved passage uses "*it*" to refer to the pericardial sac.

- retrieval by passage can suffer from ambiguity of an entity or whether the entity is being referred or not due to its lengthy nature of a passage

- retrieval by sentence suffers from lack of context as the sentences are split based on sentence completion and not based on contextual combination

Retrieval by propositions tackles the aforementioned problems by ensuring each retrieval unit contains one piece of fact only and necessary context is incorporated in the propositions. However, proposition-based retrieval faces challenges with questions that involve multi-hop reasoning over long-range textual analysis. In Example Q4, the retrieved passage separately describes the actor's name and the character they portray. There is not a single proposition that entails both the question and the answer.

- retrieval by proposition overcomes ambiguity or reference of an entity by solving the co-reference issues and maintains context

7 Related Work

Recent works on dense retrievers typically adopt a dual-encoder architecture (Yih et al., 2011; Reimers and Gurevych, 2019; Karpukhin et al., 2020; Ni et al., 2022). With dual-encoders,

Passage Retrieval	Sentence Retrieval	Proposition Retrieval
Q1: What was the theme of Super Bowl 50?		
Title: Super Bowl X ✗ The overall theme of the Super Bowl entertainment was to celebrate the United States Bicentennial. Each Cowboys and Steelers player wore a special patch with the Bicentennial logo on their jerseys...	Title: Super Bowl X ✗ The overall theme of the Super Bowl entertainment was to celebrate the United States Bicentennial.	Title: Super Bowl XLV ✓ ... As this was the 50th Super Bowl game, the league [Super Bowl 50] emphasized the "golden anniversary" with various gold-themed initiatives during the 2015 season, as well as...
Q2: The atomic number of indium which belongs to 5th period is?		
Title: Period 5 element ✗ The periodic table is laid out in rows to illustrate recurring (periodic) trends in the chemical behaviour of the elements as their atomic number increases: ...	Title: Period 5 element ✓ Indium is a chemical element with the symbol In and atomic number 49.	Title: Period 5 element ✓ Indium is a chemical element with the symbol In and [Indium has a] atomic number 49. This rare, very soft, malleable ...
Q3: What is the function of the pericardial sac?		
Title: Pericardium ✓ The pericardium, also called pericardial sac ... It separates the heart from interference of other structures, protects it against infection and blunt trauma, and lubricates the heart's movements.	Title: Pericardium ✗ The pericardium, also called pericardial sac, is a double-walled sac containing the heart and the roots of the great vessels.	Title: Cardiac muscle ✓ On the outer aspect of the myocardium is the epicardium which forms part of the pericardial sac that surrounds, protects, and lubricates the heart.
Q4: What is the main character's name in layer cake?		
Title: Layer Cake (film) ✓ ... The film's plot revolves around a London-based criminal, played by Daniel Craig, ... Craig's character is unnamed in the film and is listed in the credits as "XXXX".	Title: Angelic Layer ✗ The primary protagonist is Misaki Suzuhara.	Title: Plot twist ✗ Sometimes the audience may discover that the true identity of a character is , in fact, unknown [in Layer Cake] , as in Layer Cake or the eponymous assassins in V for Vendetta and The Day of the Jackal.

Table 4: Example cases where top-1 retrieved text unit of each retrieval granularity fails to provide the correct answer. The underlined text is the correct answer. The gray text is the context of propositions, but it is for illustration purpose only and not provided to the retrievers and downstream QA models.

each query and document is encoded into a low-dimensional feature vector respectively, and their relevance is measured by a non-parametric similarity function between the embedding vectors (Mussmann and Ermon, 2016). Due to the limited expressivity from the similarity function, dual encoder models often generalize poorly to new tasks with scarce training data (Thakur et al., 2021). To this end, previous studies use techniques such as data augmentation (Wang et al., 2022; Yu et al., 2023a; Izacard et al., 2022; Gao and Callan, 2022; Lin et al., 2023; Dai et al., 2023), continual pre-training (Chang et al., 2020; Sachan et al., 2021; Oguz et al., 2022), task-aware training (Xin et al., 2022; Cheng et al., 2023), hybrid sparse-dense retrieval (Luan et al., 2021; Chen et al., 2022) or mixed strategy retrieval (Ma et al., 2022, 2023) and so on to improve cross-task generalization performance of dense retrievers.

The motivation of our work echoes in part with multi-vector retrieval, e.g. ColBERT (Khattab and Zaharia, 2020), DensePhrase (Lee et al., 2021a,b), ME-BERT (Luan et al., 2021), MVR (Zhang et al., 2022), where the retrieval model learns to encode a candidate retrieval unit into multiple vectors to increase model expressivity and improve retrieval granularity (Seo et al., 2019; Humeau et al., 2019). Our work instead focuses on the setting where we do not update the dense retriever model or its parameters. We show that segmenting the retrieval corpus into finer-grained units of proposition can be a simple and orthogonal strategy for improving the generalization of dual encoders dense retrievers at inference time.

The idea of using propositions as a unit of text representation dates back to the Pyramid method in summarization evaluation (Nenkova and Passonneau, 2004), where model generated summary is evaluated by each proposition. Proposition extraction from text has been a long-standing task in NLP, with earlier formulations focusing on a structured representation of propositions, e.g. Open Information Extraction (Etzioni et al., 2008), Semantic Role Labeling (Gildea and Jurafsky, 2000). More recent studies have found success in extracting propositions in natural language form via few-shot prompting with large language models (Min et al., 2023; Kamoi et al., 2023), or finetuning smaller compact-sized models (Chen et al., 2023b).

Retrieve-then-read, or more broadly – retrieval augmented generation, has recently merged as a popular paradigm for open-domain question answering (Lewis et al., 2021; Jiang et al., 2023; Asai et al., 2023). While earlier works provide up to the top 100 retrieved passages for the downstream reader (Izacard and Grave, 2021; Kedia et al., 2022), the amount of allowed context is significantly reduced when using recent large language models (e.g. top 10) (Touvron et al., 2023; Yu et al., 2023b), due to their limited context window length and inability to reason over long context (Liu et al., 2023). Recent efforts have tried to improve the quality of the reader context by filtering or compressing the retrieved documents (Wang et al., 2023; Xu et al., 2023). Our work offers a new perspective by leveraging a new retrieval unit, the proposition that not only reduces the context length but also offers greater information density, effectively addressing the issue.

8 Conclusion

We propose the use of propositions as retrieval units for indexing corpus to improve dense retrieval performance at inference time. Through our experiments on five open-domain QA datasets with six different dense retrievers, we discovered that retrieval by proposition outperforms passage or sentence in both passage retrieval accuracy and downstream QA performance with a fixed retrieved word budget. We introduce FACTOIDWIKI, an indexed version of the English Wikipedia dump, where text from 6 million pages is segmented into 250 million propositions. We hope that FACTOIDWIKI, along with our findings in the paper, will facilitate future research on information retrieval.

Limitations

The scope of our current study on the granularity of retrieval corpus has the following limitations. (1) *Retrieval Corpus* – Our study only focus on Wikipedia as the retrieval corpus, due to the fact that most open-domain QA datasets adopts Wikipedia as the retrieval corpus. (2) *Types of dense retrievers evaluated* – In the current version of the paper, we only evaluate on 6 types of popular dense retrievers, most of which follow bi- or dual-encoder architecture. In future versions, we will include and discuss results on a broader range of dense retrievers. (3) *Language* – Our current study is limited to English Wikipedia only. We leave the exploration on other languages to future work.

References

Zeynep Akkalyoncu Yilmaz, Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Cross-domain modeling of sentence-level evidence for document retrieval. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3490–3496, Hong Kong, China. Association for Computational Linguistics.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, et al. 2022. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*.

Wei-Cheng Chang, X Yu Felix, Yin-Wen Chang, Yiming Yang, and Sanjiv Kumar. 2020. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations*.

Sihao Chen, Senaka Buthpitiya, Alex Fabrikant, Dan Roth, and Tal Schuster. 2023a. PropSegmEnt: A large-scale corpus for proposition-level segmentation and entailment recognition. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8874–8893, Toronto, Canada. Association for Computational Linguistics.

Sihao Chen, Hongming Zhang, Tong Chen, Ben Zhou, Wenhao Yu, Dian Yu, Baolin Peng, Hongwei Wang, Dan Roth, and Dong Yu. 2023b. Sub-sentence encoder: Contrastive learning of propositional semantic representations. *arXiv preprint arXiv:2311.04335*.

Xilun Chen, Kushal Lakhotia, Barlas Oguz, Anchit Gupta, Patrick Lewis, Stan Peshterliev, Yashar Mehdad, Sonal Gupta, and Wen-tau Yih. 2022. Salient phrase aware dense retrieval: Can a dense retriever imitate a sparse one? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 250–262, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hao Cheng, Hao Fang, Xiaodong Liu, and Jianfeng Gao. 2023. Task-aware specialization for efficient and robust dense retrieval for open-domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1864–1875, Toronto, Canada. Association for Computational Linguistics.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*, 9:447–461.

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

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot dense retrieval from 8 examples. In *The Eleventh International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2843–2853, Dublin, Ireland. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Daniel Gildea and Daniel Jurafsky. 2000. Automatic labeling of semantic roles. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 512–520, Hong Kong. Association for Computational Linguistics.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 113–122.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. *arXiv preprint arXiv:1905.01969*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

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.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. Wice: Real-world entailment for claims in wikipedia. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

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.

Akhil Kedia, Mohd Abbas Zaidi, and Haejun Lee. 2022. FiE: Building a global probability space by leveraging early fusion in encoder for open-domain question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4246–4260, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Daniel Khashabi, Yeganeh Kordi, and Hannaneh Hajishirzi. 2022. Unifiedqa-v2: Stronger generalization via broader cross-format training. *arXiv preprint arXiv:2202.12359*.

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*, pages 39–48.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Jaewoong Lee, Heejoon Lee, Hwanhee Lee, and Kyomin Jung. 2021a. Learning to select question-relevant relations for visual question answering. In *Proceedings of the Third Workshop on Multimodal Artificial Intelligence*, pages 87–96, Mexico City, Mexico. Association for Computational Linguistics.

Jinhyuk Lee, Alexander Wettig, and Danqi Chen. 2021b. Phrase retrieval learns passage retrieval, too. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3661–3672, Online and Punta Cana, Dominican Republic. 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, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

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. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks.

Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen-tau Yih, and Xilun Chen. 2023. How to Train Your DRAGON: Diverse Augmentation Towards Generalizable Dense Retrieval. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Kaixin Ma, Hao Cheng, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2022. Open-domain question answering via chain of reasoning over heterogeneous knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5360–5374, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Kaixin Ma, Hao Cheng, Yu Zhang, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2023. Chain-of-skills: A configurable model for open-domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1599–1618, Toronto, Canada. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. AmbigQA: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online. Association for Computational Linguistics.

Stephen Mussmann and Stefano Ermon. 2016. Learning and inference via maximum inner product search. In *International Conference on Machine Learning*, pages 2587–2596. PMLR.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA. Association for Computational Linguistics.

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. *CoRR*, abs/1611.09268.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Scott Yih, Sonal Gupta, and Yashar Mehdad. 2022. Domain-matched pre-training tasks for dense retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1524–1534, Seattle, United States. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

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.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Devendra Sachan, Mostofa Patwary, Mohammad Shoeybi, Neel Kant, Wei Ping, William L. Hamilton, and Bryan Catanzaro. 2021. End-to-end training of neural retrievers for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6648–6662, Online. Association for Computational Linguistics.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734, Seattle, United States. Association for Computational Linguistics.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. *arXiv preprint arXiv:2109.08535*.

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4430–4441, Florence, Italy. Association for Computational Linguistics.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, Seattle, United States. Association for Computational Linguistics.

Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. 2023. Learning to filter context for retrieval-augmented generation.

Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul Bennett. 2022. Zero-shot dense retrieval with momentum adversarial domain invariant representations. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4008–4020, Dublin, Ireland. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. 2020.

Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94.

Wen-tau Yih, Kristina Toutanova, John C. Platt, and Christopher Meek. 2011. Learning discriminative projections for text similarity measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 247–256, Portland, Oregon, USA. Association for Computational Linguistics.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2023a. Generate rather than retrieve: Large language models are strong context generators. In *The Eleventh International Conference on Learning Representations*.

Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. 2023b. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210*.

Shunyu Zhang, Yaobo Liang, Ming Gong, Daxin Jiang, and Nan Duan. 2022. Multi-view document representation learning for open-domain dense retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5990–6000, Dublin, Ireland. Association for Computational Linguistics.

A Retrieval Corpus Processing

The English Wikipedia dump used in this study, released by Bohnet et al., 2022, was selected because it has been filtered to remove figures, tables, and lists, and is organized into paragraphs. The dump dates back to October 13, 2021. We have segmented Wikipedia into three retrieval units for this study: 100-word passage chunks, sentences, and propositions. Paragraphs are divided into 100-word passage chunks using a greedy method. We divide only at the end of sentences to ensure each passage chunk contains complete sentences. As we process the paragraph, we add sentences one by one. If including the next sentence causes the passage chunk to exceed 100 words, we start a new passage chunk with that sentence. However, if the final passage chunk is shorter than 50 words, we merge it with the previous one to avoid overly small segments. Each passage is further segmented into sentences using the widely used Python SpaCy `en_core_web_lg` model. Additionally, each passage is decomposed into propositions by our *Propositionizer* model. We decomposed 6 million pages into 41 million passages, 114 million sentences, and 257 million propositions. On average, a passage contains 6.3 propositions, and a sentence contains 2.3 propositions.

B Training the Propositionizer

We generated a list of propositions from a given paragraph using GPT-4 with a prompt, as shown in Figure 8. After filtering, 42,857 pairs were used to fine-tune a Flan-T5-Large model. We named the model Propositionizer. The AdamW optimizer was used with a batch size of 64, learning rate of 1e-4, weight decay of 1e-4, and 3 epochs.

To compare the proposition generation performance of different models, we set up a development set and an evaluation metric. The development set contains an additional 1,000 pairs collected by GPT-4 using the same approach as the training set. We evaluated the quality of the predicted propositions by the F1 score of two sets of propositions. Motivated by the F1 score of two sets of tokens in BertScore, we designed the F1 score for two sets of propositions. Let $P = \{p_1, ..., p_n\}$ denote the set of labeled propositions and $\hat{P} = \{\hat{p}_1, ..., \hat{p}m\}$ the set of predicted propositions. We use $\mathrm{sim}(p_i, \hat{p}j)$ to represent the similarity between two propositions. Theoretically, any text similarity metric can be used. We chose BertScore with roberta-large configuration as our sim function since we wanted our metric to reflect the semantic difference between propositions. We define

$$\text{Recall} = \frac{1}{|P|} \sum_{p_i \in P} \max_{\hat{p}_j \in \hat{P}} \text{sim}(p_i, \hat{p}_j)$$

$$\text{Precision} = \frac{1}{|\hat{P}|} \sum_{\hat{p}_j \in \hat{P}} \max_{p_i \in P} \text{sim}(p_i, \hat{p}_j)$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Here is a figurative explanation of the F1 score: Recall represents the percentage of propositions in the labeled set that are similar to those in the generated set, Precision represents the percentage of propositions in the generated set that are similar to the labeled set, and F1 is the harmonic mean of Recall and Precision. F1 is 1 if the two sets are exactly the same, and 0 if any two propositions are semantically different.

We conducted a comparative analysis of base-size and large-size Flan-T5 models, which were trained using varying amounts of data (shown in Figure 5). Our findings suggest that larger models, coupled with extensive training data, yield better results. The *Propositionizer* presented in this paper attained an F1 score of 0.822. Upon manually reviewing the generated propositions, we found them to be satisfactory.

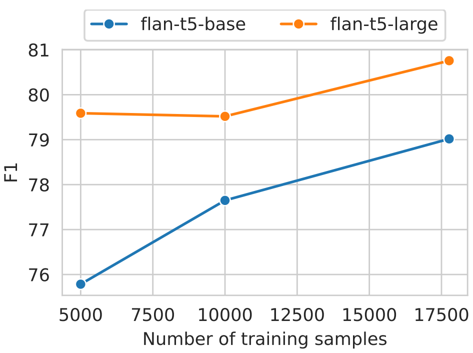


Figure 5: Performance of proposition-level decomposition by models with different sizes and number of training data.

C Offline Indexing

We used the `pyserini` and `faiss` packages to encode retrieval units into embeddings. We exploited multiple GPUs to encode each text unit in groups of 1M units with a batch size of 64. After preprocessing the embeddings, we used an exact search for the inner product (`faiss.IndexFlatIP`) in all experiments. The plain index of FACTOIDWIKIis approximately 768GB in size. To reduce memory pressure, the embeddings are split into 8 shards. An approximate nearest neighbor search is conducted per shard before aggregating all results.

Although the number of propositions is six times that of passages, using efficient indexing techniques can enable sub-linear search times relative to the total count of vectors. Moreover, utilizing GPU parallelism and distributed indexes significantly decreases the online search time. As a result, with proper implementation, we can make proposition retrieval a practically viable and efficient option.

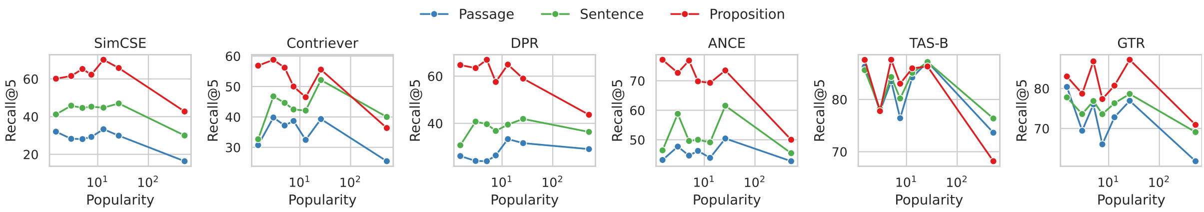
D Retrievers Models

We used `transformers` and `sentence-transformers` packages for the model implementation. We used the following checkpoints released on HuggingFace: SimCSE (`princeton-nlp/unsup-simcse-bert-base-uncased`), Contriever (`facebook/contriever`), DPR (`facebook/dpr-ctx_encoder-multiset-base`, `facebook/dpr-question_encoder-multiset-base`), ANCE (`castorini/ance-dpr-context-multi`, `castorini/ance-dpr-question-multi`,), TAS-B (`sentence-transformers/msmarco-distilbert-base-tas-b`), and GTR (`sentence-transformers/gtr-t5-base`).

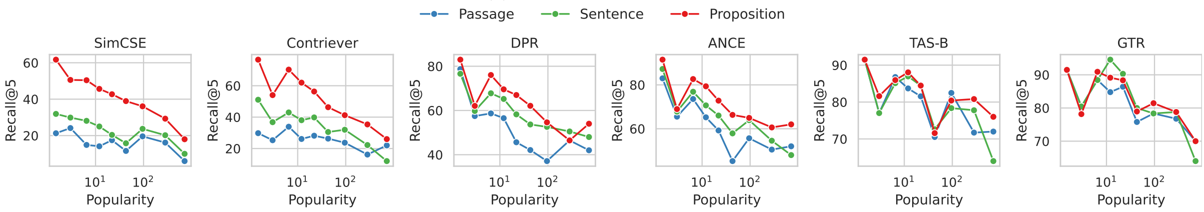
E Additional Results

In Section 5.2, we demonstrated the advantage of retrieval by proposition over retrieval by sentence, particularly as the population of the entity decreases in EQ. We used the occurrence in the top-1000 paragraphs retrieved by BM25 as a proxy for popularity, rather than counting the number of hyperlinks to the entity used in Sciavolino et al., 2021. Therefore, the trend in the performance versus popularity plot shows some differences (Figure 6) between our results and those in Sciavolino et al., 2021. For example, some entities are ambiguous (e.g., *1992*, a TV series). In such cases, the occurrence of the surface form of the entity is large. Simultaneously, questions related to ambiguous entities are challenging to answer, leading to lower recall.

In Section 6.2, we discussed the recall of answers in the retrieved text with respect to the context length. We further illustrate the performance trends of six dense retrievers, as detailed in Figure 7. The results indicate that the recall rate of propositions consistently outperforms that of sentences and passages. Our findings lead to the conclusion that question-related density is greater in proposition units compared to sentences and passages.



(a) Where was [X] born?



(b) Who was [X] created by?

Figure 6: Document retrieval recall vs. the popularity of the target entity in each question from the *EntityQuestions* dataset. We display the performance of two relations.

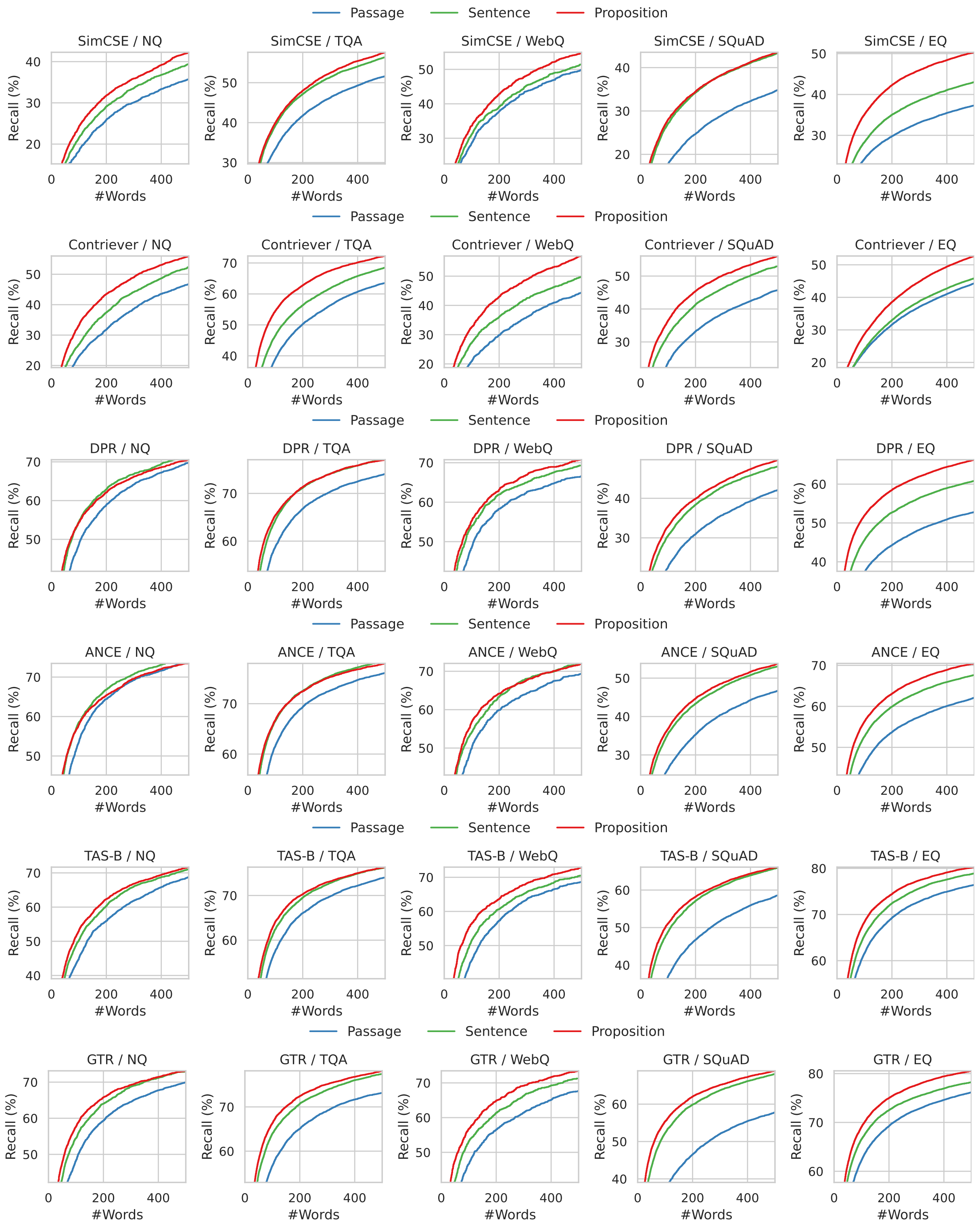


Figure 7: Recall of the gold answer in the retrieved text limited to first k words. Finer-grained retrieval has a higher recall across all numbers of words.

Passage ⇒ Propositions

Decompose the "Content" into clear and simple propositions, ensuring they are interpretable out of context.

1. Split compound sentence into simple sentences. Maintain the original phrasing from the input whenever possible.
2. For any named entity that is accompanied by additional descriptive information, separate this information into its own distinct proposition.
3. Decontextualize the proposition by adding necessary modifier to nouns or entire sentences and replacing pronouns (e.g., "it", "he", "she", "they", "this", "that") with the full name of the entities they refer to.
4. Present the results as a list of strings, formatted in JSON.

Input: Title: Ēostre. Section: Theories and interpretations, Connection to Easter Hares. Content: The earliest evidence for the Easter Hare (Osterhase) was recorded in south-west Germany in 1678 by the professor of medicine Georg Franck von Franckenau, but it remained unknown in other parts of Germany until the 18th century. Scholar Richard Sermon writes that "hares were frequently seen in gardens in spring, and thus may have served as a convenient explanation for the origin of the colored eggs hidden there for children. Alternatively, there is a European tradition that hares laid eggs, since a hare's scratch or form and a lapwing's nest look very similar, and both occur on grassland and are first seen in the spring. In the nineteenth century the influence of Easter cards, toys, and books was to make the Easter Hare/Rabbit popular throughout Europe. German immigrants then exported the custom to Britain and America where it evolved into the Easter Bunny."
Output: ["The earliest evidence for the Easter Hare was recorded in south-west Germany in 1678 by Georg Franck von Franckenau.", "Georg Franck von Franckenau was a professor of medicine.", "The evidence for the Easter Hare remained unknown in other parts of Germany until the 18th century.", "Richard Sermon was a scholar.", "Richard Sermon writes a hypothesis about the possible explanation for the connection between hares and the tradition during Easter", "Hares were frequently seen in gardens in spring.", "Hares may have served as a convenient explanation for the origin of the colored eggs hidden in gardens for children.", "There is a European tradition that hares laid eggs.", "A hare's scratch or form and a lapwing's nest look very similar.", "Both hares and lapwing's nests occur on grassland and are first seen in the spring.", "In the nineteenth century the influence of Easter cards, toys, and books was to make the Easter Hare/Rabbit popular throughout Europe.", "German immigrants exported the custom of the Easter Hare/Rabbit to Britain and America.", "The custom of the Easter Hare/Rabbit evolved into the Easter Bunny in Britain and America."]

Input: <*a new passage*>
Output:

Figure 8: Prompt for generating propositions from a passage using GPT-4.