Can Retriever-Augmented Language Models Reason? The Blame Game Between the Retriever and the Language Model

Parishad BehnamGhader[‡] **Santiago Miret**[†] **Siva Reddy**[‡]

[‡]McGill University / Mila; [†]Intel Labs

{parishad.behnamghader, siva.reddy}@mila.quebec

santiago.miret@intel.com

Abstract

Augmenting pretrained language models with retrievers to select the supporting documents has shown promise in effectively solving common NLP problems, including language modeling and question answering, in an interpretable way. In this paper, we first study the strengths and weaknesses of different retriever-augmented language models (REALM, kNN-LM, FiD coupled with DPR, and ATLAS and Flan-T5 coupled with Contriever) in reasoning over the retrieved statements in different tasks. We show how the retrieve-then-read models' limitations in reasoning are rooted both in the retriever module as well as the language model. Our experimental results demonstrate that the similarity metric used by the retrievers is generally insufficient for reasoning tasks. Additionally, we show that the language models in retriever-augmented models do not take the complicated relations between the statements into account, which leads to poor reasoning performance even when using the larger models. Moreover, we analyze the reasoning performance of large language models using multihop retrieval but we only observe minor improvements. Overall, this shows great room for further research in this area.[1]

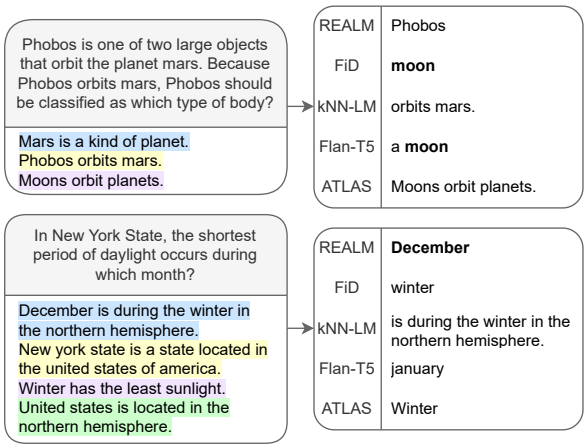

Figure 1: **Example of language model failures in question answering tasks when reasoning is needed.** The correct answer is in **bold** with the majority of retriever-augmented language models failing to answer correctly.

1 Introduction

Large parametric language models, such as decoder-only transformers (e.g. GPT), transformer encoder models (e.g. BERT), and encoder-decoder transformers (e.g. T5), have shown outstanding results on many natural language tasks (Vaswani et al., 2017; Radford et al., 2018; Devlin et al., 2019; Raffel et al., 2020), while their implicit knowledge structure lacks interpretability. Non-parametric models improve pretrained language models by augmenting them with knowledge retrievers (Guu et al., 2020; Izacard and Grave, 2021; Izacard et al., 2022b) or memory components (Zhong et al., 2022; Khandelwal et al., 2020; Verga et al., 2021). The primary goal of using a latent knowledge retriever is to let the model capture information from external knowledge rather than relying entirely on the implicit knowledge hidden in the model's parameters. In other words, non-parametric models perform inference using the additional knowledge from the retrieved documents. This circumvents the limitations of relying solely on memorized knowledge that is often dependent on the size of the model (Izacard et al., 2022b).

While the limitations and capabilities of large parametric language models have been well studied in the literature (Wei et al., 2022; Zelikman et al., 2022), there are no thorough studies of the limitations of non-parametric models. For instance, Mallen et al., 2022 address the strengths and limitations of non-parametric memories when facing less popular factual knowledge. Our work, on the other hand, provides a systematic approach to study the limitations of the retriever-augmented language

[1]The code is publicly available at https://github.com/McGill-NLP/retriever-lm-reasoning.

models in reasoning over the retrieved supporting information. As shown in Figure 1, models sometimes fail in solving the task when some multi-step entailment and sequential logical reasoning (e.g., taxonomic chaining, combining the details, etc.) is required over the given statements to generate the correct answer for the question. For instance, the second example is specifically asking for a *month*, while some models do not reason deeply to reach the answer *December*.

Although retriever-augmented language models have shown promising results in many language modeling or question answering tasks (Guu et al., 2020; Khandelwal et al., 2020; Izacard and Grave, 2021; Chung et al., 2022; Izacard et al., 2022b), our analysis suggests that these models still have some limitations in reasoning. These shortcomings are rooted in two distinct parts of their design: the retriever and the language model. On one hand, the knowledge retriever is not trained to retrieve the required statements for completing the task when reasoning. Instead, it selects the most similar documents based on a similarity metric between each candidate statement and the input query (Guu et al., 2020; Karpukhin et al., 2020; Khandelwal et al., 2020; Izacard et al., 2022a). On the other hand, we observe that current language models show promising performance when a similar form of the query with its answer is provided, but perform worse when some reasoning is required over the statements as presented in Figure 1.

In this study, we demonstrate how retriever-augmented language models fail in entailment and logical reasoning from different perspectives by evaluating them in language modeling (LM) and question answering (QA) tasks using different variations of EntailmentBank (Dalvi et al., 2021) and StrategyQA (Geva et al., 2021) datasets, where we can control for the given supporting statements, as well as the reasoning skills. Unlike most of the existing datasets, in many of the studied samples, the decomposition into the facts and the reasoning path is not evident from the question itself, which makes the retrieval more complicated. In order to perform well in the tasks of these datasets, the models need to 1) retrieve the best statements as supporting evidence leading to a promising result; and 2) aggregate knowledge from retrieved statements using reasoning to get the correct answer. Concretely, we analyze the performance of pretrained REALM, FiD with DPR, ATLAS with Contriever,

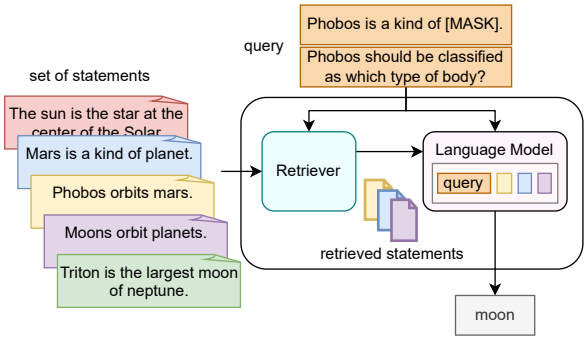

Figure 2: **The architecture of retrieve-then-read retriever-augmented language models.** The language model predicts the answer using the query and retriever's selected statements.

and k-NN-LM as retriever-based language models and the instruction-finetuned Flan-T5 as a recent strong model in reasoning coupled with Contriever. We also perform an analysis on the impact of the models' sizes and the multihop retrieval on larger Flan-T5 language models and GPT-3.5's performance in reasoning datasets.
In this paper, we address the following questions:

Q1 *Can retriever-augmented language models perform multi-step reasoning given supporting statements?* The experimental results show that retriever-augmented language models have some difficulties in performing multi-step entailments or logical reasoning in solving QA and LM examples, especially when facing some distracting statements.

Q2 *What are some shortcomings of retrievers which undermine reasoning performance?* By investigating the shortcomings of retrievers in selecting the required statements for reasoning, we conclude that selecting the statements based on their similarity with the queries is insufficient for the reasoning nature. This approach works promisingly when similar forms of the query containing the answer exist among the supporting statements, while there are sometimes no such statements when reasoning is required to solve a task.

Q3 *What are some shortcomings of language models which undermine reasoning performance?* The results demonstrate that language models cannot consider the relations between the statements, which is necessary for reasoning. Interestingly, we also observe that language models perform much better when a

statement containing information from all relevant ones is provided.

Q4 *What is the impact of model size?* Our experimental results using various sizes of Flan-T5 model demonstrate that larger models lead to better performance on reasoning tasks.

Q5 *What is the impact of using multihop retrieval?* The experimental results with different sizes of Flan-T5 model and GPT-3.5 using a recent multihop approach reflect improvement in some, but not all of the experiments which shows there is still significant room for improvement in this area.

2 Related Work

Augmenting language models with external corpora or memory has been well studied in literature (Guu et al., 2020; Izacard and Grave, 2021; Izacard et al., 2022b; Khandelwal et al., 2020). Additionally, researchers have recently been interested in eliciting the reasoning abilities of parametric language models in tasks where answering the questions involves multiple computational steps (Wei et al., 2022; Chung et al., 2022). Next, we discuss some of the recent studies in these areas.

2.1 Retriever-Augmented Language Models

Retriever-based language modeling or question answering has been well studied in the literature, and different mechanisms have been proposed for integrating language models with retrieved statements. For instance, Retrieval-Augmented Language Model (**REALM**) is a masked language model that is augmented with a latent knowledge retriever (Guu et al., 2020). The knowledge retriever employs a BERT encoder to achieve a dense representation for both the query and the statements. In fact, it selects the most similar statements based on a similarity metric (i.e., dense inner product) between the transformed query's and the statements' representations. In the language modeling task, the retrieved statements are appended to the query and passed to a trained BERT-based language model. In the question answering task, however, a trained BERT-based reader extracts the most promising span from the statements as the answer.

kNN-LM is another proposed model based on a decoder-only Transformer, integrated with a k-nearest neighbor module (Khandelwal et al., 2020). In this model, the most related token sequences from the statements are selected based on an L2 similarity metric between the representation of the query and all token sequences. The distribution over the next token in generation is subsequently computed as the interpolation between the transformer's next token distribution and the distribution of the next tokens in the nearest neighbor module based on the retrieved statements.

Fusion-in-Decoder (**FiD**) is a sequence-to-sequence T5-based neural network (Izacard and Grave, 2021). Given the retrieved statements by a frozen retriever, the encoder encodes the query and each retrieved statement separately. Afterward, the decoder attends to the representations of all the retrieved statements. In this paper, we investigate Dense Passage Retriever (DPR) as the retriever for FiD (Karpukhin et al., 2020). DPR retrieves the most similar documents based on the inner product of the representations of the query and the documents (i.e., embeddings of the `[CLS]` token) from two independently trained BERT encoder models.

Finally, **ATLAS** is a pre-trained retrieval augmented language model with an architecture similar to FiD's, designed for jointly finetuning the retriever and the language model using different pretext tasks with few training examples (Izacard et al., 2022b). In our experiments, a Contriever-based retriever accompanies ATLAS, consisting of a dual BERT-based encoder architecture (Izacard et al., 2022a). The retrieved documents are selected based on the inner product of the representations of the query and the documents (i.e., the average hidden representations of the encoder's last layer).

In addition to the previously mentioned widely used class of *retrieve-then-read* models, multihop retrieving is another method that utilizes retrievers and language models in a recursive framework. For instance, multihop dense retriever (MDR) encodes the question and previously retrieved documents as a new query vector and retrieves the next relevant documents (using DPR) in an iterative manner for a fixed number of iterations (Xiong et al., 2021). Iterative Retriever, Reader, and Reranker (IRRR) also employs a single multi-task transformer model to perform retrieval and predict the answer in an iterative fashion for a variable number of steps (Qi et al., 2021). In IRRR, the new search query tokens are selected from the original question and all of the already retrieved documents. Question Answering via Sentence Composition (QASC) is also a two-step retrieval for questions answering which

retrieves later facts based on the first set of retrieved facts for the query, in order to find information about the new concepts not mentioned in the question (Khot et al., 2020). Furthermore, Demonstrate-Search-Predict (**DSP**) is a framework that enables passing natural language texts in arbitrarily complicated pipelines between the language model and the retriever for various purposes (Khattab et al., 2022). For instance, in multi-hop QA, Khattab et al., 2022 use a language model to generate a query by decomposing a complex question into smaller subproblems, summarize the information from retrieved supporting documents iteratively, and generate the answer in the end.

More recently, with the advent of large language models, other retrieving methods have been introduced. For example, Retrieve and Plug (REPLUG), is a new paradigm where the language model is treated as black box (Shi et al., 2023). REPLUG first retrieves a small set of relevant documents using a possibly tunable retriever. Then the concatenation of each retrieved document with the input context is passed through the language model in parallel, and the answer is generated as the ensemble of predicted probabilities. The relation between the documents, which is quite important in reasoning tasks, is still not considered in this approach.

2.2 Reasoning of Language Models

Eliciting the reasoning ability of parametric large language models has recently attracted the attention of many researchers (Wei et al., 2022; Zelikman et al., 2022; Chung et al., 2022). For instance, Wei et al., 2022 investigate how reasoning abilities emerge in large language models when they are given a few intermediate reasoning steps as exemplars for few-shot prompting (i.e., chain of thoughts). Moreover, **Flan-T5** is an instruction-finetuned T5 model which is shown to have strong reasoning abilities, outperforming the T5 model (Chung et al., 2022; Raffel et al., 2020). Although this finetuned model is not initially constructed for retriever-based language modeling, it can be coupled with Contriever to complete the language modeling and question answering task using the retrieved statements as supporting information.

In this paper, we mainly focus on the currently widely-used retrieve-then-read class of models and study the reasoning ability of REALM, kNN-LM, FiD with DPR, and ATLAS with Contriever as retriever-based language models, and Flan-T5 as a reasoning language model coupled with Contriever. Additionally, we conduct some experiments to measure the effectiveness of larger models, especially with the recent multihop DSP approach. While retrievers generally select statements from a huge common corpus in the literature, as illustrated in Figure 2, we accompany each query with a data-specific collection of statements since we want to have more control over the supporting statements in our experiments.

Language Models		Question Answering Models	
model	# params	model	# params
REALM	∼110M	REALM-QA	∼270M
kNN-LM	∼250M	FiD	∼220M
		ATLAS	∼250M
		Flan-T5-base	∼250M
Model Size and Multihope Retrieval Analysis			
Flan-T5-small	∼80M	Flan-T5-xl	∼3B
Flan-T5-base	∼250M	Flan-T5-xxl	∼11B
Flan-T5-large	∼780M	GPT-3.5 (text-davinci-002)	∼175B

Table 1: **The number of parameters in the studied language models.** We control for model size in the main experiments to circumvent the role of model size in reasoning abilities.

3 Problem Definition

In our retriever-augmented language model reasoning main experimental setting, we provide the model with a complete set of statements $S = \{s_1, s_2, \ldots, s_m\}$ for each sample. In some cases, only some of these statements are necessary to predict the answer (which we call the *gold statements*), while others contain distracting information. For a fixed number of retrieved statements k, the model should retrieve the set of statements $S_r = \{r_1, r_2, \ldots, r_k\} \subseteq S$, which it finds more related and necessary and solve the target task by reasoning over them. A visualization of the general task is illustrated in Figure 2. In this paper, we study REALM, kNN-LM, FiD with DPR, and ATLAS and Flan-T5 with Contriever models in two tasks: Language Modeling (LM) and Question Answering (QA). Based on the implementation details stated in Appendix A, since we control for model size in our main experiments as presented in Table 1, the results are comparable among different models. We also report the size of the other models used in our further model size and multihop retrieval analysis in Table 1.

Language Modeling (LM). In the language modeling setup, we measure the performance of the

Model	Alternative target scores
REALM	$\frac{1}{M}\sum_{i=1}^{M}\log p\left(\texttt{[MASK]}_i = t_i \mid Q, S_r\right)$
kNN-LM	$\frac{1}{N}\log p\left(Q_T \mid S_r\right)$, where Q_T is the query Q with [MASK] tokens substituted with T
FiD Flan-T5 ATLAS	$\frac{1}{M}\sum_{i=1}^{M}\log p\left(t_i \mid Q, S_r\right)$

Table 2: **The alternative target scores in the LM task for each model.** Scores are presented for a target entity mention $T = t_1 t_2 \ldots t_M$ and input query $Q = q_1 q_2 \ldots q_N$, given retrieved statements S_r.

retriever-augmented language models in two tasks: 1) *next token prediction*: predicting the desired token correctly; and 2) *target ranking*: assigning a higher likelihood to the gold sentence than similar but incorrect sentences. We use the latter task as a proxy to compare different autoregressive or masked language models. To this end, we consider at most four alternative sentences by defining alternative targets for the true target entity mention and compare the score of the model corresponding to each sentence by attending to the retrieved statements by the retriever. We present the alternative target scoring functions for each model in Table 2. More detailed information about LM query formats for each model is described in Appendix B.

Question Answering (QA). In the question answering setup, we study the correctness of the generated answers in the reasoning QA datasets.

4 Experimental Results

In this section, we first introduce the datasets and metrics used in the experiments for both LM and QA tasks as explained in Section 2.1. We then demonstrate the limitations of the retriever-augmented language models in detail.

4.1 Datasets

We compare the reasoning ability of the models based on their performance on the following datasets in different formats, with a detailed dataset preparation mechanism explained in Appendix C. Using these datasets, we can evaluate the models' reasoning abilities in both LM and QA, and we control for the input set of supporting statements.

EntailmentBank (**EB**, Dalvi et al., 2021) contains a multi-step entailment tree for a given hypothesis. This dataset consists of three parts, each with a specific characteristic. In EB-1, only the required statements are provided in data samples. In our experiments, we call this dataset EB-Easy. In EB-2, 25 statements have been given for each data sample, including all the required and some irrelevant ones. We call this dataset EB-Hard. The ground-truth entailment trees are available in both EB-Easy and EB-Hard datasets. In EB-3, 25 relevant and irrelevant statements have been given by sampling from a large corpus with no ground-truth entailment tree. Each data sample in these datasets consists of some statements, a question, an answer, and a hypothesis rephrasing the question and answer in a declarative form.

StrategyQA (Geva et al., 2021) contains boolean QA samples, where the required reasoning steps are implicit in the question, as well as supporting evidence from Wikipedia. For evaluating the models in the language modeling setting, we convert each question to a declarative format.

4.2 Evaluation Metrics

We evaluate the performance of different models in the LM task by measuring the target ranking accuracy as well as the accuracy in predicting the first target token in a single run. In the QA task, we take the token overlapping F1 score of the generated answer for EntailmentBank experiments, and the accuracy of ranking the correct *yes* or *no* answer higher than the other in StrategyQA. We also report the token overlapping recall score of the generated answer in Appendix D. When analyzing the retrievers' performance, we take the ground-truth statement retrieval recall score to see how well retrievers find the relevant statements.

4.3 Evaluation and Discussion

In this section, we show the limitations of the retriever-augmented models in different tasks and study the impact of retrievers, language models, and the models' size on the performance in reasoning tasks.

4.3.1 Can retriever-augmented language models perform multi-step reasoning given supporting statements? (Q1)

In this section, we show the limitations of retriever-augmented language models in solving LM and QA tasks by reasoning. We present the overall behavior of the models in target ranking (i.e., ranking the true target higher than at most four other alternative targets), as described in Section 3, on the test sets in

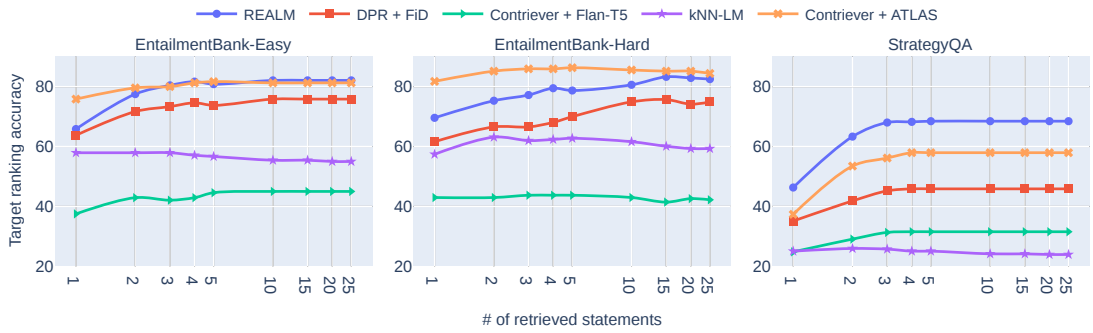

Figure 3: **Target ranking accuracy of the studied models in LM on test sets based on the number of retrieved statements.** It can be observed that Contriever + ATLAS, REALM, and DPR + FiD perform reasonably well in the reasoning datasets.

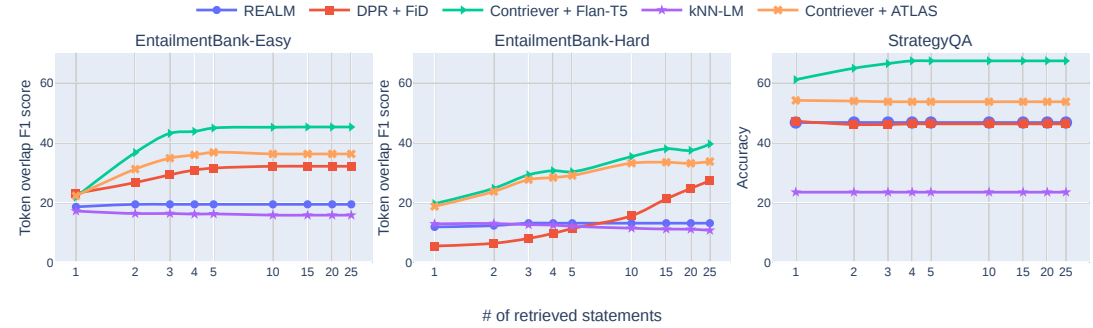

Figure 4: **Performance of the retrieval-augmented models in QA on test sets based on the number of retrieved statements.** The results demonstrate that although Contriever + Flan-T5 and Contriever + ATLAS are superior, the studied models perform poorly at reasoning when answering questions. In StrategyQA experiments, a perfect model should rank the correct answer (*yes* or *no*) higher 100% of the times.

Figure 3. Results show that Contriever + ATLAS, REALM, and DPR + FiD perform better than kNN-LM and Contriever + Flan-T5 in the LM reasoning datasets. Surprisingly, the performance of kNN-LM, a model pretrained for LM, is close to random on StrategyQA. We present the results for EB-3 dataset as well as other experimental results for the next token prediction task in Appendix D.

Figure 4 demonstrates the performance of the models in QA reasoning datasets. Since the QA version of REALM is an extractive model, we append *yes/no* to the statements in StrategyQA in REALM-QA experiments. These results show that Contriever + Flan-T5 performs well in question answering experiments. The superiority of Contriever + Flan-T5 might be because, unlike the others, this model has been finetuned on the chain of thought data, and therefore it can have a more explicit understanding of reasoning. However, all of the models have some difficulty generating the correct answer. Although the token overlap F1 score metric penalizes fundamentally correct answers that do not match the exact wordings of the ground-truth answer, models still have some problems in the easier boolean QA task of StrategyQA. Furthermore, we report the token overlapping recall score on EntailmentBank datasets in Appendix D.

These shortcomings of retriever-augmented language models in reasoning are embedded in different parts of their architectures. In order for the models to perform well, they need to 1) retrieve the best relevant statements leading to promising results; and 2) aggregate knowledge and reason over retrieved statements for correct answers. In the following subsections, we explore the weaknesses of these models caused by the retriever or the accompanying language model.

4.3.2 What are some shortcomings of retrievers which undermine reasoning performance? (Q2)

In this subsection, we study the efficacy of the retrievers in reasoning. Current retrievers select k statements as the *retrieved statements* using a relevance score between the query and each statement

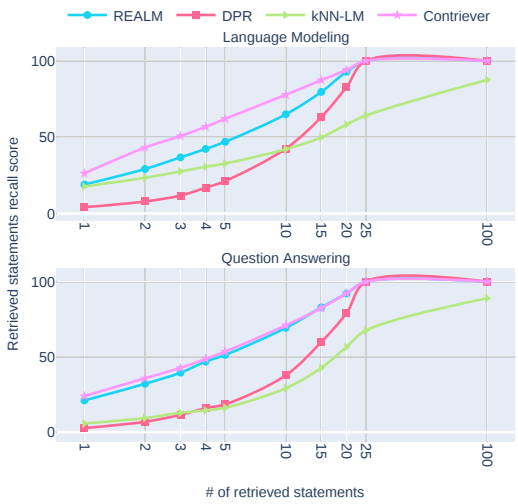

Figure 5: **Retreivers' recall score on EB-Hard test set in LM and QA based on the number of retrieved statements (k).** Results show that some retrievers do not select required statements properly.

as explained in Section 2.1. The commonly used similarity metrics are the inner product and the L2 distance between query's and statements' (or a span in statements) representations derived by one or two different encoders (Guu et al., 2020; Karpukhin et al., 2020; Izacard et al., 2022a; Khandelwal et al., 2020). This naive way of selecting the related statements does not take into account the relation between them. Take the question "What keeps the Moon orbiting Earth?" in Table 3 as an example. Supporting statements such as (1)*"The moon's orbit is elliptical."*, (2)*"Moons orbit planets."*, and (3)*"The moon is earth's moon."* are selected as similar ones by the retriever, none of which include the answer "gravity". Instead, combining information from statements (2), (3), and a missed statement like (4)*"Gravity causes orbits."* would answer the question, although statement (4) itself is not similar to the question according to the retrievers' similarity metrics.

We analyze the retrievers by evaluating the models on the EB-Hard dataset that includes both gold and distracting statements. Figure 5 shows the superiority of Contriever over the other retrievers. This retriever has been jointly pretrained with the ATLAS language model using negative pairs across batches (in contrast to DPR with in-batch negatives (Karpukhin et al., 2020)). Moreover, it has been trained with only unsupervised data (e.g., in masked language modeling task) which is shown to improve robustness in the context of zero-shot transfer (Izacard et al., 2022a). The results also demonstrate that although Contriever is superior among the studied retrievers, all four studied retrievers lack in retrieving the most relevant and necessary statements for reasoning. For instance, the top retrieved statement by DPR is among the gold statements only 15% of the times. Note that in the LM setting, REALM, DPR, and Contriever have access to all the tokens of the query, while kNN-LM is autoregressive and has access only to the tokens before masking tokens. Also note that all the data samples have a set of at most 25 statements, however, kNN-LM's retriever does not retrieve one statement at a time. In this model, each statement $s = w_1 w_2 w_3 \dots w_n$ is stored as $n-1$ key-value pairs in the nearest neighbor module. For instance, $w_1 w_2$ is one of the keys that can be retrieved with value (i.e., next token) w_3. This is why the performance of other retrievers reaches 100 at $k = 25$, but this is not the case for kNN-LM. To this regard, even with letting kNN-LM retrieve 100 sequences, it does not cover all the gold statements and tends to retrieve sequences from the same statement.

Some failures of the retrievers are demonstrated in Table 3. These examples besides the experimental results presented in Figure 5 show that retrieving based on the similarity of the query and the candidate statements is not sufficient in our reasoning experiments. Although the missed statements do not seem similar to the query, they carry important information required for reasoning that can be combined with other statements. Overall, retrievers perform poorly in selecting required statements, and a stronger approach would be necessary to account for the relationships between them than a simple similarity metric.

4.3.3 What are some shortcomings of language models which undermine reasoning performance? (Q3)

In this subsection, we discuss the shortcomings of language models in reasoning. Suppose we have a perfect retriever that can retrieve necessary and sufficient statements for solving the target task (i.e., gold statements). An ideal language model would combine gold statements step-by-step, simulating the entailment reasoning procedure. However, we observe that the language models we study in this paper do not consider the complicated relations between statements and have difficulty in solving the task perfectly regarding reasoning even over the gold statements.

Model	Query	Statements	Prediction
DPR + FiD	In a zoo located in a warm region, what should be included in the polar bear exhibit?	+ If an animal lives a certain environment then that animal usually requires that kind of environment. - Polar bears live in **cold environments**.	warm
Contriever + ATLAS	What keeps the Moon orbiting Earth?	+ Moons orbit planets. - **Gravity** causes orbits.	elliptical
kNN-LM	The robot will weigh less on mars than earth but will have the same [MASK]. Targets: *mass* vs *mars*	+ As the force of gravity decreases, the weight of the object will decrease. - The gravitational force of a planet does not change the **mass** of an object on that planet or celestial body.	mars

Table 3: **Some examples of models' failures rooted in the retriever.** One of the correctly retrieved statements and the one that had to be retrieved in order for the model to solve the task correctly are highlighted in green and red, respectively. The sequence of tokens leading to the true answer is marked in **bold**.

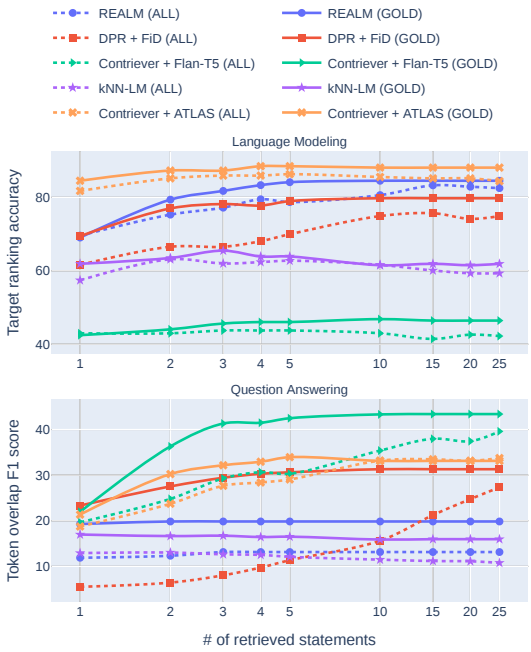

Figure 6: **Performance of language models on EB-Hard test set in both LM and QA.** The solid lines and the dotted lines refer to the experiments using the gold statements and all of the statements (including gold and distracting ones), respectively. It can be observed that even when all gold statements are given, some models still have difficulty in solving the tasks.

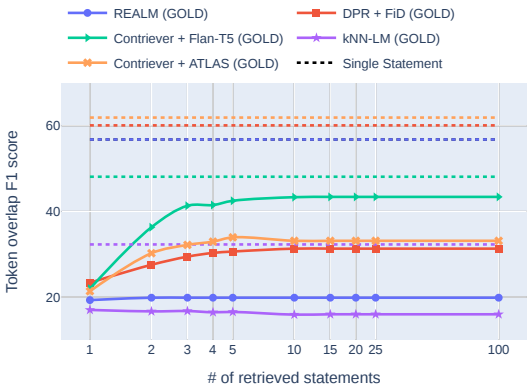

Figure 7: **Token overlap F1 score of language models on EB-Hard test set in QA.** The dotted lines and the solid lines refer to experiments given single statements and gold statements (when reasoning is required), respectively. Results illustrate that the language models perform much better when answering the question requires only one statement.

We evaluate the language models on EB-Hard dataset in a scenario where they have access only to the gold statements to see if they can infer the answer correctly by reasoning over the gold statements. Figure 6 demonstrates the performance of the language models given all or gold statements. According to the solid lines in this figure, we conclude that language models do not perform perfectly, even with all the required gold statements in hand. For instance, the best performance in the QA dataset is around 40 for Flan-T5 which is instruction finetuned on the chain of thought data. Furthermore, in LM experiments, results show that kNN-LM, which is trained for language modeling, is performing substantially worse than REALM, FiD, and ATLAS, and its performance does not differ much even without distracting statements.

Additionally, we conduct an experiment with a scenario where only one statement is given to the models. This given statement is in fact the hypothesis sentence available in EB-Hard data samples which can be inferred from the gold statements using reasoning, and is sufficient to answer the question. Experimental results in Figure 7 show that language models perform better when a similar form of the query with its answer is mentioned in the statements and the question can be answered using only one statement. This phenomenon indicates that the language models do not have a proper ability in reasoning over statements by combining their information. From Figure 7, it can be observed that Flan-T5 is superior in question answering with

Model	Query	Retrieved statements	Prediction
Flan-T5	What allows two students standing ten feet apart to hear each other talk?	+ Talking is when a human produces sound to communicate. + Sound can travel through air by **vibrating air**.	a microphone
REALM	Andy lives in the southern hemisphere. What season does he most likely experience in August?	+ Andy lives in southern hemisphere. + August is during the **winter** in the southern hemisphere.	in southern hemisphere

Table 4: **Some examples of models' failures rooted in the language model.** In each example, two correct retrieved statements are illustrated. The sequence of tokens leading to the true answer is marked in **bold**.

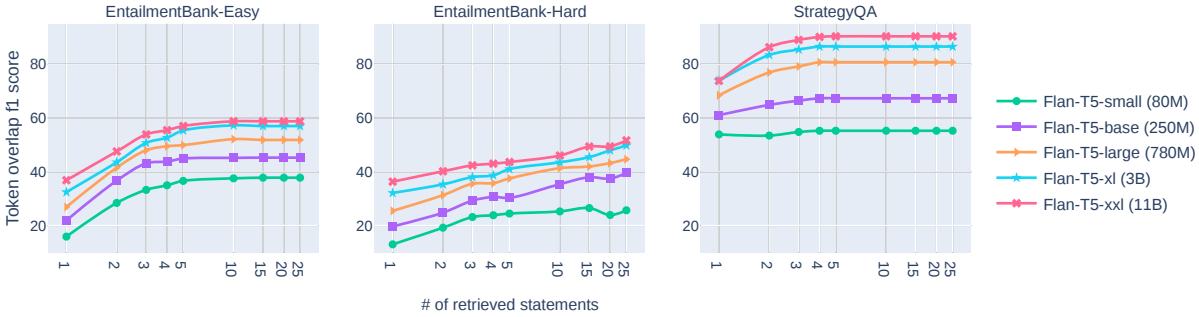

Figure 8: **Token overlap F1 score of various sizes of Flan-T5 in QA on test sets based on the number of retrieved statements.** The results demonstrate that larger models perform better in F1 scores. We use Contriever as the retriever in all experiments.

reasoning over ground-truth statements, while ATLAS and FiD perform better when the hypothesis is given as one statement.

Some failures of the language models are demonstrated in Table 4. Among the QA-finetuned models, the performance of REALM in Figure 7 and the example provided in Table 4 show that the model tends to extract the answer span from the first few retrieved statements, as REALM's performance stays the same after retrieving about 2 statements. Furthermore, we sometimes observe that Flan-T5 answers the queries regardless of the provided retrieved statements, as demonstrated in Table 4.

4.3.4 What is the impact of model size? (Q4)

In this subsection, we study the impact of models' size on the performance of the retriever-augmented language models in the QA datasets. As described in Section 3, although we control for the model size in our main experiments for a fair comparison, we report the token overlapping F1 score of Flan-T5 in different sizes on QA datasets in Figure 8. Experimental results reflect the impressive impact of model size on the performance of the models in the tasks where reasoning is required.

4.3.5 What is the impact of using multihop retrieval? (Q5)

As a possible solution to the poor performance of retrieve-then-read models on reasoning datasets,

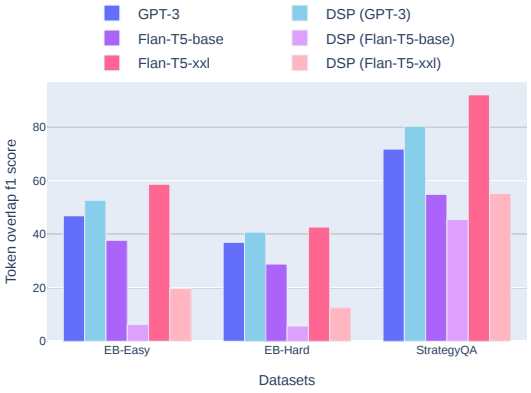

Figure 9: **Token overlap F1 score of GPT-3.5 and Flan-T5 models using multihop DSP program.** All the experiments are done with access to few examples using Contriever as the retriever and 5 retrieved statements in each retrieval step. The experimental results show that while DSP improves GPT-3.5 performance, it does not help Flan-T5 models in F1 score.

in this subsection, we evaluate the Demonstrate-Search-Predict (DSP) approach using Contriever along with different language models as a strong recent multihop retrieval framework (Khattab et al., 2022). DSP uses a language model to first generate subqueries by decomposing a complex question into smaller subproblems, then summarize the information from retrieved supporting documents iteratively, and generate the answer in the end. In our experiments, we retrieve the top 5 statements

in each retrieval, with the same templates as suggested in the original paper. Due to the generally longer context size in multihop retrieval setting and the Flan-T5's context window limitations, multihop retrieval and retrieve-then-read experiments include two and five few-shot demonstrations in the prompt, respectively.

The token overlap F1 scores of the models using the multihop DSP approach are illustrated in Figure 9. We observe that there is a huge difference between F1 score of Flan-T5 models with and without multihop retrieval that is basically because first, Flan-T5-xxl does not generate proper subqueries for the question, and second, Flan-T5-xxl's generated responses usually include all the retrieved information, which leads to higher recall but lower precision score. This phenomenon can also be seen in the qualitative examples and recall scores of the models demonstrated in Appendix F. While Figure 9 demonstrates the superiority of Flan-T5-xxl over GPT-3.5 in a simple retrieve-then-read manner, we do not observe the same pattern with the models' recall scores in EntailmentBank datasets reported in Appendix F, which shows the problem of the common existing metrics with the verbosity of large language models such as GPT-3.5 that tend to elaborate on the generated response. Overall, while DSP enhances GPT-3.5's token overlap F1 score in our experiments, there is still a large room for improvement in our studied reasoning QA datasets.

5 Conclusion

In this paper, we analyzed to what extent the retriever-augmented language models are capable of solving downstream tasks where reasoning is required. To this end, we first evaluate REALM, kNN-LM, FiD coupled with DPR, ATLAS and Flan-T5 coupled with Contriever in language modeling and question answering tasks.

Experimental results demonstrate that retrievers do not retrieve the statements necessary for reasoning. Instead, they select statements based on query similarity which is not a good representative of carrying important information that can be used for reasoning over statements. These incorrect retrieved statements are shown to be harmful to some of the models. Furthermore, we observe that although most of these models perform reasonably well when the answer is mentioned in one of the statements in a similar format, they generally cannot reason over different statements even when all the retrieved statements are among the ground-truth ones. We also observe some deficiencies in the target ranking task where models, especially kNN-LM and Flan-T5, do not rank the true targets higher than the alternative ones. Overall, our qualitative results demonstrate that language models do not take the relations between the statements into account, and therefore, cannot reason over them properly.

These results suggest opportunities for improving the reasoning ability of the retriever-augmented language models by first, improving the retrievers so that they select statements using a more focused approach compared to a simple relevance score, and second, enhancing the language models so that they combine the information from retrieved statements by considering the statements' relations more effectively.

While larger language models seem more powerful in our reasoning tasks, they still have some shortcomings that should be addressed. Additionally, our experiments on multihop retrieval do not show a prominent improvement in all of the studied models' performance. More concretely, DSP shows minor improvements only with very large language models as it relies heavily on the subqueries generated by the model. Our findings lead to an opportunity for future work to improve the performance of both retrieve-then-read and multihop retriever-augmented models in reasoning tasks.

References

Hyung Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa dehghani, Siddhartha Brahma, Albert Webson, Shixiang Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. Explaining answers with entailment trees. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7358–7370, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022a. 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.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022b. Atlas: Few-shot learning with retrieval augmented language models.

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.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. QASC: A dataset for question answering via sentence composition. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8082–8090. AAAI Press.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Peng Qi, Haejun Lee, Tg Sido, and Christopher Manning. 2021. Answering open-domain questions of varying reasoning steps from text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3599–3614, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Pat Verga, Haitian Sun, Livio Baldini Soares, and William Cohen. 2021. Adaptable and interpretable neural MemoryOver symbolic knowledge. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages

3678–3691, Online. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wenhan Xiong, Xiang Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oguz. 2021. Answering complex open-domain questions with multi-hop dense retrieval. In *International Conference on Learning Representations*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. STar: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*.

Zexuan Zhong, Tao Lei, and Danqi Chen. 2022. Training language models with memory augmentation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

A Implementation Details

We present the implementation details of the analyzed models in this section. Most of the experiments are conducted using PyTorch (Paszke et al., 2019) on an RTX8000 GPU with 48GB memory in a single run, each taking a few minutes to run. We also run the experiments with large 30B-parameter models on an A100 GPU with 80GB memory. Note that we have changed the retriever in each model to retrieve statements from a sample-specific set of statements instead of a large common corpus.

In REALM's experiments, we use the Huggingface's transformers implementation for both masked language modeling and question answering (Wolf et al., 2020). We load the `realm-cc-news-pretrained-encoder` checkpoint as a knowledge encoder for masked language modeling and `realm-orqa-nq-openqa` checkpoint for question answering. For kNN-LM experiments, we use the best checkpoint available in the original papers' GitHub repository, and we find $\lambda = 0.65$ the best value as the interpolation hyperparameter based on the experiments on EntailmentBank development sets. In FiD experiments, we use `nq_reader_base` checkpoint available in the papers' GitHub repository with using the `nq.bert-base-encoder`'s checkpoint of the DPR retriever which is available in their GitHub repository. For experimenting ATLAS, we use the trained `atlas_data/models/atlas_nq/base` checkpoint of both the language model and retriever. Also, for the Flan-T5 model, we load the `flan-t5-base` model from Huggingface's transformers to be almost the same size as the other models in the main experiments. In order to analyze the impact of the model size, we experiment with various Flan-T5 models from Huggingface, as well as OpenAI's GPT-3.5 text-davinci-002 model.

B Model and Task-Specific Query Format

This section includes the model-specific query formats in each target task. As stated in Section 3, we aim to study the reasoning abilities of retriever-augmented language models in language modeling and question answering tasks. A sample of what the queries to each model in language modeling look like is presented in Table 5. These examples are all specified for the next token prediction. From these examples, it can be observed that, unlike REALM, FiD, ATLAS, and Flan-T5, kNN-LM does not have access to the tokens appearing after the target tokens in the sentence. We resolve this problem by evaluating the models in the target ranking task as explained in Section 3. The alternative target scoring function in each model is presented in Table 6.

Model	Query
REALM	Surface mining affects the `[MASK]` and biosphere.
kNN-LM	Surface mining affects the
FiD Flan-T5 ATLAS	Surface mining affects the `<extra_id_0>` and biosphere.

Table 5: **A sample of the query formats for each model for the next token prediction in the LM task.** We use `[MASK]` and `<extra_id_0>` as the special tokens in BERT-based and T5-based models, respectively.

Model	Alternative target scores
REALM	$\log p\,(\texttt{[MASK]} = \text{lithosphere}\|Q, S_r)$ $\log p\,(\texttt{[MASK]} = \text{coal}\|Q, S_r)$
kNN-LM	$\frac{1}{8}\log p(\text{Surface mining affects the lithosphere and biosphere.}\|S_r)$ $\frac{1}{8}\log p(\text{Surface mining affects the coal and biosphere.}\|S_r)$
FiD Flan-T5 ATLAS	$\frac{1}{2}\log p(\texttt{<extra\_id\_0>}\ \text{lithosphere}\|Q, S_r)$ $\frac{1}{2}\log p(\texttt{<extra\_id\_0>}\ \text{coal}\|Q, S_r)$

Table 6: **A sample of the ranking strategies for each model for target ranking in the LM task using retrieved statements $\mathbf{S_r}$.** The query (Q) in this example is "Surface mining affects the `[MASK]` and biosphere." with alternative targets "lithosphere" and "coal".

In the question answering setting, on the other hand, we give the whole question to the model and take the generated output as the answer.

C Dataset Preparation Details

In order to prepare the datasets for the language modeling experiments, we first create an LM reasoning dataset for StrategyQA by changing the questions and the *yes/no* answers into declarative-form sentences. We also use hypothesis sentences of the EntailmentBank dataset for LM experiments. Afterward, we keep the data samples that include at least one entity mention and mask out the last entity mention in the sentences of StrategyQA and different EntailmentBank variants using Spacy (Honnibal and Montani, 2017). Also, we randomly pick

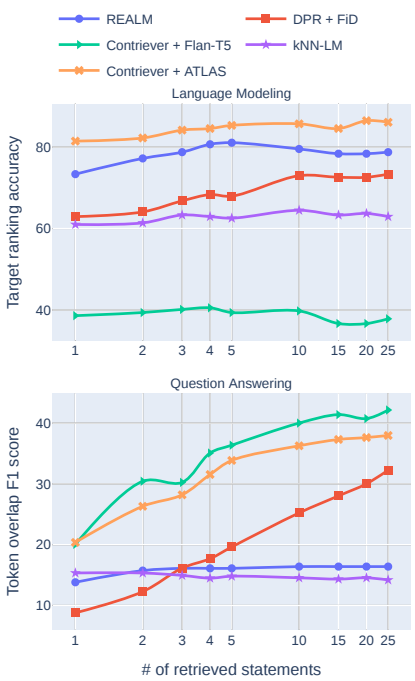

Figure 10: **Performance of the models in EB-3 LM and QA datasets.** Results demonstrate that the models perform similarly to EB-Hard, both datasets consisting of required and distracting statements.

at most four other entities mentioned in the data sample's statements as the alternative targets (as described in Section 3) and compare the model's score for each target. Regarding the question answering experiments, we use datasets' question and answer formats.

For the experiments on the EntailmentBank datasets, we run the experiments on the same development and test sets as the original data. However, in the StrategyQA dataset, since we do not have access to the answers in the test split, we cannot change the samples' formats to declarative form. Therefore, we pick 25% and almost 35% of the train data as the development and test sets, respectively.

D Quantitative Results

This section includes more visualizations and detailed results. We demonstrate the performance of the models on EB-3 test set in both target ranking (LM) and question answering tasks in Figure 10. It can be observed that Contriever + ATLAS is the superior model in LM experiments, and Contriever + Flan-T5 performs the best in QA setting. Also, we observe that the models' performance is similar to EB-Hard dataset with both relevant and distracting statements.

	Token overlap F1 score			Accuracy
	EB-Easy	EB-Hard	EB-3	SQA
REALM	19.43	13.14	16.39	46.75
Contriever + ATLAS	35.95	33.14	37.63	53.91
DPR + FiD	32.14	27.32	32.27	46.09
Contriever + Flan-T5	**45.26**	**39.60**	**42.16**	**67.34**
kNN-LM	17.20	12.64	14.80	23.49

Table 7: **Experimental results of the best retriever-augmented models in QA on test sets.** The two best models are highlighted in green. The results show that Flan-T5 and ATLAS are the superior models in the studied datasets.

The token overlapping recall score of the models on EntailmentBank datasets are also presented in Figure 11. Comparing the visualizations in Figure 4 and Figure 11, results show that kNN-LM performs better than REALM according to the recall score, probably due to longer generated responses.

Figure 12 demonstrates the performance of retriever-augmented language models on the test sets in language modeling based on next token prediction accuracy. As expected, results show that Contriever + ATLAS and REALM are still performing reasonably well in language modeling in predicting the next masked token. According to the results in Figure 12, while kNN-LM, a decoder-only model, does not see the input tokens appearing after the masked tokens in predicting the next target token, it performs better than DPR + FiD and Contriever + Flan-T5 in EntailmentBank datasets and performs almost similar to DPR + FiD in StrategyQA.

We report the performance of the best retriever-augmented models (based on the performance on the development sets) on the test sets in question answering in Table 7. Also, Table 8 demonstrates the performance of the best retriever-augmented models (based on the performance on the dev sets) on the test sets in the language modeling task.

E Qualitative Results

We demonstrate some failure examples in each of the retrievers and language models in Table 9. In this table, a few true retrieved statements and the one that had to be retrieved in order for the model to solve the task correctly are highlighted in green and red, respectively. The true answer (or sequence

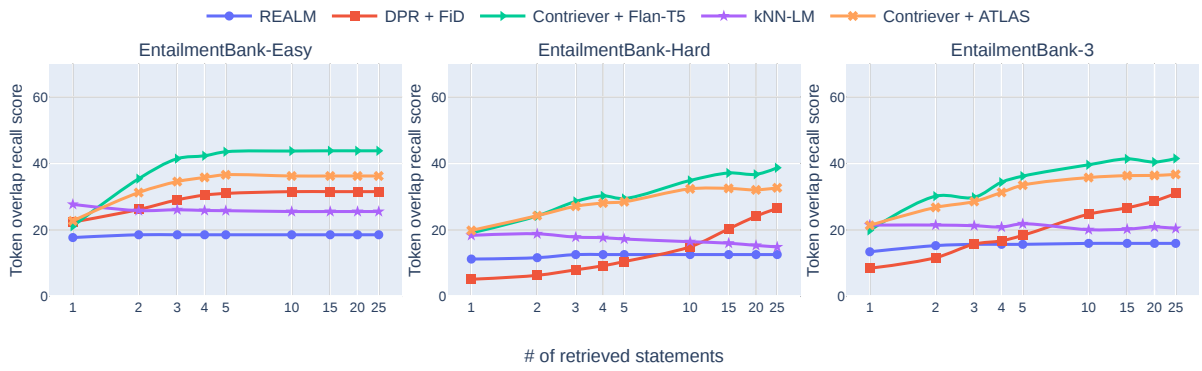

Figure 11: **Token overlapping recall score of the retrieval-augmented models in QA on EntailmentBank based on the number of retrieved statements.** The results demonstrate that Contriever + Flan-T5 and Contriever + ATLAS are superior among the studied models.

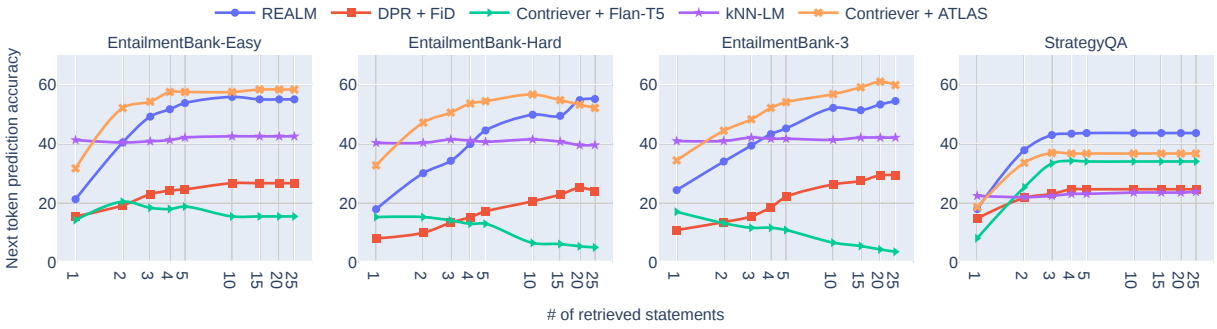

Figure 12: **Next token prediction accuracy of the retrieval-augmented models in LM based on the number of retrieved statements.** The results show that models do not predict the first target token a lot of the times.

	Target ranking accuracy				Next token prediction accuracy			
	EB-Easy	EB-Hard	EB-3	SQA	EB-Easy	EB-Hard	EB-3	SQA
REALM	**82.08**	82.51	81.08	**68.46**	55.00	55.13	45.17	**43.62**
Contriever + ATLAS	81.25	**85.55**	**84.56**	57.94	**57.50**	**56.65**	**59.07**	36.69
DPR + FiD	73.75	74.90	72.59	45.86	24.58	20.53	27.41	24.61
Contriever + Flan-T5	45.00	42.97	39.38	31.54	15.42	15.21	10.81	34.00
*k*NN-LM	57.92	62.74	61.39	25.06	40.42	40.68	40.93	23.04

Table 8: **Experimental results of the best retriever-augmented models in LM on test sets.** The two best models are highlighted in green. The results show that ATLAS and REALM are superior in LM.

of tokens leading to the true answer) for each data sample's statements is marked in bold. These examples explain how not retrieving the necessary statements for reasoning or not reasoning over true statements can lead to incorrect answers.

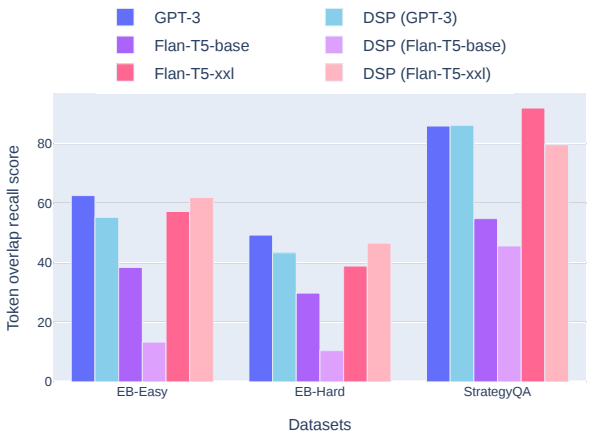

Figure 13: **The token overlap recall score of GPT-3.5 and Flan-T5 models using multihop DSP program.** The results show that DSP is not a perfect multihop program according to the recall score of the models in our reasoning datasets.

F Impact of multihop retrieval

Some of the examples of the multihop retrieval approach of DSP using GPT-3.5 and Flan-T5-xxl are illustrated in Table 10. Even though Flan-T5-xxl includes the correct answer tokens in its generated response, it can be observed that the subqueries generated by this model are sometimes nothing but paraphrasing or repetition of the original questions, while the goal of the multihop DSP program is to break down the problems into smaller subproblems. Moreover, the Flan-T5-xxl's responses usually include all the retrieved information, which is not desired. The recall score of the models using the multihop DSP approach is illustrated in Figure 13 which shows the relatively high recall score of the larger Flan-T5 model due to the problem mentioned above.

	Model	Query	Statements	Answer
Retriever's Failures	DPR + Flan-T5	In a zoo located in a warm region, what should be included in the polar bear exhibit?	+ If an animal lives a certain environment then that animal usually requires that kind of environment. - Polar bears live in **cold environments**.	a polar bear
	DPR + FiD			warm
	Contriever + ATLAS	What keeps the Moon orbiting Earth?	- Moons orbit planets. - **Gravity** causes orbits.	elliptical
	kNN-LM	The robot will weigh less on mars than earth but will have the same [MASK]. Targets: *mass* vs *mars*	+ As the force of gravity decreases, the weight of the object will decrease. - The gravitational force of a planet does not change the **mass** of an object on that planet or celestial body.	mars
	REALM	A complete orbit of mercury around the sun takes [MASK]. Targets: *around 88 earth days* vs *between 1 and 365*	+ A complete revolution / orbit of a planet around its star takes 1 / one planetary year. - One mercury year is **about 88 earth days**.	between 1 and 365
		If a new moon occurred on June 2, when will the next new moon occur?	+ A new moon occurred on **june** 2. + A moon phase occurs 28 days after the last time it occurred. - 2 plus 28 equals **30**.	june 2
Language Model's Failures	DPR + Flan-T5	What allows two students standing ten feet apart to hear each other talk?	+ Talking is when a human produces sound to communicate. + Sound can travel through air by **vibrating air**.	a microphone
	DPR + FiD	Which energy conversion happens when a person shivers and the energy is transferred to make the muscles and joints move?	+ A person is a kind of animal. + When an animal moves, **chemical energy is converted to mechanical energy**. + Shivering is a kind of shaking. + Shaking is a kind of moving.	shaking
	Contriever + ATLAS	Wave energy from the ocean can be harnessed to power generators to make electricity. Energy from ocean tides can also be used to make electricity. How would you categorize these two sources of energy?	+ Tidal energy means energy from ocean tides. + Tidal energy is a **renewable resource**. + Wave energy is a **renewable resource**.	Wave energy
		Which changes will most likely have a negative effect on an ecosystem?	+ Humans changing ecosystems usually has a negative impact on an ecosystem / organisms living in an ecosystem. + **Humans building roads** in an ecosystem causes that ecosystem to change.	Humans changing ecosystems
	kNN-LM	The mass of earth causes the pull of gravity on [MASK]. Targets: *earth* vs *newton*	+ The mass of a planet causes the pull of gravity on **that planet**. + **Earth** is a kind of planet.	newton
	REALM	Andy lives in the southern hemisphere. What season does he most likely experience in August?	+ Andy lives in southern hemisphere. + August is during the **winter** in the southern hemisphere.	in southern hemisphere

Table 9: **A complete set of models' failure examples rooted in the retriever or language model modules.** In the retriever's failures section, a few correctly retrieved statements and the ones that had to be retrieved in order for the model to solve the task correctly are highlighted in green and red, respectively. The sequence of tokens leading to the true answer is marked in **bold**.

Question	Statements	Answer
Which characteristic of a cheetah is more likely to be learned rather than inherited?	+ Skills are learned characteristics. + Inherited characteristics are the opposite of learned characteristics. + Hunting is a kind of skill.	hunting strategies
GPT-3.5		
first subquery	learned characteristics of a cheetah	
first rationale	[1] and [2] suggest that skills are learned characteristics, while [3] suggests that hunting is a kind of skill.	
second subquery	Is hunting a learned or inherited characteristic in cheetahs?	
second rationale	First, we need to understand the definition of each term. [1] and [2] suggest that skills are learned characteristics, while [3] suggests that hunting is a kind of skill. Therefore, the answer must be a learned characteristic of a cheetah that is not hunting.	
response	How to run	
Flan-T5-xxl		
first subquery	What is the most likely characteristic of a cheetah to be learned rather than inherited?	
first rationale	What is the most likely characteristic of a cheetah to be learned rather than inherited?	
second subquery	What is the most likely characteristic of a cheetah to be learned rather than inherited?	
second rationale	What is the most likely characteristic of a cheetah to be learned rather than inherited?	
response	hunting is kind of skill inherited characteristics are opposite of learned characteristics answer hunting	

Question	Statements	Answer
A bee depends on certain flowers for food. The flowers depend on the bee to	+ Pollination is when pollinating animals, wind, or water carry pollen from one flower to another flower. + A bee is a pollinating animal. + Pollination requires pollinating animals. + Plant reproduction requires pollination.	carry pollen for reproduction.
GPT-3.5		
first subquery	What do bees eat?	
first rationale	- A bee is a pollinating animal. - Pollination is when pollinating animals, wind, or water carry pollen from one flower to another flower. - Plant reproduction requires pollination. - Pollination requires pollinating animals.	
second subquery	What is the relationship between bees and flowers?	
second rationale	First, we need to know what pollination is. Pollination is when an animal carries pollen from one flower to another. This is important because flowers need to be pollinated in order to reproduce. Therefore, the bee depends on the flowers for food, and the flowers depend on the bee for pollination.	
response	pollinate	
Flan-T5-xxl		
first subquery	What do flowers depend on bees for?	
first rationale	What do flowers depend on bees for?	
second subquery	Pollination is when pollinating animals, wind, or water carry pollen from one flower to another flower. What do flowers depend on bees for?	
second rationale	What do flowers depend on bees for?	
response	Pollination is when pollinating animals, wind, or water carry pollen from one flower to another flower.	

Question	Statements	Answer
A young boy and his brothers have green eyes. This trait is best described as	+ Eye color is an inherited characteristic. + Green is a kind of color.	inherited
GPT-3.5		
first subquery	What is a trait?	
first rationale	- Eye color is an inherited characteristic. - Green is a kind of color.	
second subquery	What is the definition of "inherited characteristic?"	
second rationale	We are looking for a word that describes how the trait is passed down. This is an inherited characteristic, so it is passed down from parents to children.	
response	inherited	
Flan-T5-xxl		
first subquery	What is the medical term for the color of a person's eyes?	
first rationale	What is the medical term for the color of a person's eyes?	
second subquery	Eye color is an inherited characteristic. What is the medical term for the color of a person's eyes?	
second rationale	What is the medical term for the color of a person's eyes?	
response	Eye color is an inherited characteristic. Eye color is a kind of color. Eye color is inherited. The medical term for the color of a person's eyes is eye color. Eye color is an inherited characteristic. Eye color is a kind of color.	

Table 10: **Some examples of multihop question answering using the DSP approach with Contriever as the retriever.** In each sample, the generated subqueries, rationales, and final response are presented. The correct and incorrect answers included in the generated tokens are highlighted in green and red, respectively.