ExcluIR: Exclusionary Neural Information Retrieval

Anonymous ACL submission

Abstract

Exclusion is an important and universal linguistic skill that humans use to express what they do not want. There is little research on exclusionary retrieval, where users express what they do not want to be part of the results produced for their queries. We investigate the scenario of exclusionary retrieval in document retrieval for the first time. We present ExcluIR, a set of resources for exclusionary retrieval, consisting of an evaluation benchmark and a training set for helping retrieval models to comprehend exclusionary queries. The evaluation benchmark includes 3,452 high-quality exclusionary queries, each of which has been manually annotated. The training set contains 70,293 exclusionary queries, each paired with a positive document and a negative document. We conduct detailed experiments and analyses, obtaining three main observations: (i) existing retrieval models with different architectures struggle to comprehend exclusionary queries effectively; (ii) although integrating our training data can improve the performance of retrieval models on exclusionary retrieval, there still exists a gap compared to human performance; and (iii) generative retrieval models have a natural advantage in handling exclusionary queries.

1 Introduction

Selective attention (Treisman, 1964; LaBerge, 1990; Cherry, 2020), defined as the ability to focus on relevant information while disregarding irrelevant information, plays a crucial role in shaping user's search behaviors. This principle, originating from cognitive psychology, not only shapes human perception of the environment but also extends its influence to interactions with information retrieval systems. When searching for information, users can express a desire to exclude certain information. We refer to this phenomenon as *exclusionary retrieval*, where users explicitly indicate their preference to exclude specific information.

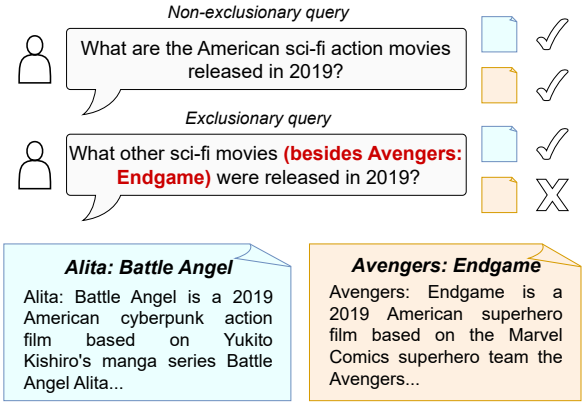

Figure 1: A comparison between non-exclusionary and exclusionary queries. Exclusionary queries often specify content to be excluded (e.g., "Avengers: Endgame") to express the user's requirements for omitting certain information. In this case, if the retrieval system fails to comprehend the exclusionary nature of a query (e.g., one containing the term "besides,") it will produce retrieval results that users do not desire.

Exclusionary retrieval emphasizes a crucial need for precision and relevance in information retrieval. It shows how users use their knowledge and expectations to find information that meets their specific needs. Therefore, the failure to understand exclusionary queries can present a potentially serious problem. For example, as shown in Figure 1, imagine a user searching for movies in the retrieval system. He poses a query like "What other sci-fi movies (besides Avengers: Endgame) were released in 2019?" If the retrieval system cannot correctly address this exclusionary requirement, it may return results containing content irrelevant to the user's interests (e.g., the movie "Avengers: Endgame"), thus reducing user satisfaction.

Research on exclusionary retrieval remains relatively rare. Early studies mainly focus on keyword-based methods (Nakkouzi and Eastman, 1990; McQuire and Eastman, 1998; Harvey et al., 2003). The key idea is to construct boolean queries that include negation terms, which is essentially a

post-processing strategy. However, these methods have limitations due to their reliance on structured queries, making them unsuitable for more diverse and complex natural language queries. Although recent work has explored the impact of negation in modern retrieval models (Rokach et al., 2008; Koopman et al., 2010; Weller et al., 2024), their focus is on comprehending the negation semantics within documents rather than the exclusionary nature of queries.

At present, there is no evaluation dataset to assess the capability of retrieval models in exclusionary retrieval. To address this gap, our first contribution in this paper is the introduction of the resources for exclusionary retrieval, namely ExcluIR. ExcluIR contains an evaluation benchmark to assess the capability of retrieval models in exclusionary retrieval, while also providing a training dataset that includes exclusionary queries. The dataset is built based on HotpotQA (Yang et al., 2018). We first use ChatGPT[1] to generate an exclusionary query for two given relevant documents, requiring that only one document contains the answer while explicitly rejecting information from the other document. Subsequently, we employ 17 workers to check each data instance in the benchmark to ensure data quality. The training set comprises 70,293 exclusionary queries, while the benchmark includes 3,452 human-annotated exclusionary queries. This dataset can evaluate whether retrieval models can correctly retrieve documents when dealing with exclusionary queries, providing a new perspective for evaluating retrieval models.

Our second contribution is to investigate the performance of existing retrieval methods with different architectures on exclusionary retrieval, including sparse retrieval (Robertson and Zaragoza, 2009; Nogueira et al., 2019), dense retrieval (Karpukhin et al., 2020; Ni et al., 2022a), and generative retrieval methods (Bevilacqua et al., 2022; Wang et al., 2022a). We conduct extensive experiments and arrive at three main observations: (i) Existing retrieval models with different architectures cannot fully understand the real intent of exclusionary queries; (ii) Generative retrieval models possess unique advantages in exclusionary retrieval, while late interaction models (Khattab and Zaharia, 2020; Santhanam et al., 2022) like ColBERT have obvious limitations in handling such queries; (iii) Fine-tuning the retrieval models with the training set of ExcluIR can improve the performance on exclusionary retrieval, but the results are still far from satisfactory. We provide in-depth analyses of these observations. These conclusions contribute valuable insights for future research on addressing the challenges of exclusionary retrieval. We share the benchmark and evaluation scripts on https://anonymous.4open.science/r/ExcluIR.

[1] https://platform.openai.com/docs/models/gpt-3-5

2 Dataset Construction

As depicted in Figure 2, the construction of the ExcluIR dataset involves the following steps: (i) we first extract document pairs from HotpotQA (Yang et al., 2018), where each data instance consisting of two interrelated documents; (ii) for each document pair, we employ ChatGPT to generate an exclusionary query. (iii) to enhance the diversity of the synthetic queries, we further use ChatGPT to rephrase them; and (iv) finally, to ensure a high quality of the dataset, we establish annotation guidelines and hire workers for manual correction.

2.1 Collecting documents pairs

We begin the construction process by collecting documents from the HotpotQA (Yang et al., 2018) dataset, which is designed for multi-hop reasoning in question-answering task. Each data instance includes two supporting documents that are related. The model needs to comprehend the association between them and extract information from them to answer the question. We extract two related documents from each data instance to form our document pairs. In total, we collected 74,293 document pairs. After merging and removing duplicates, we obtained a document collection containing 90,406 documents.

2.2 Generating exclusionary queries

To efficiently construct our dataset, we design a prompt carefully to guide ChatGPT in generating exclusionary queries for each pair of documents (see Appendix A). To ensure that the generated queries cover both positive and negative documents, we design a two-step construction strategy. Specifically, we first instruct ChatGPT to generate a query relevant to both documents, and then guide ChatGPT to revise this query by adding a constraint to include the semantics of refusal to information from the negative document.

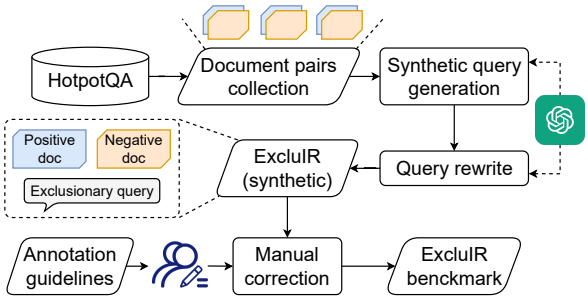

Figure 2: Overview of ExcluIR dataset construction process.

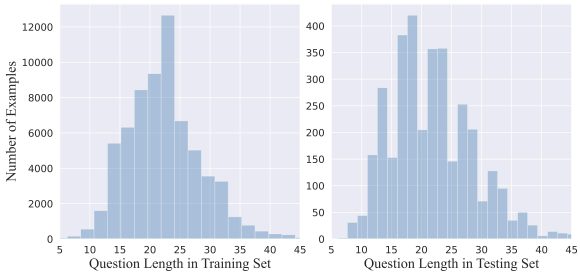

Figure 3: Distribution of the lengths of exclusionary queries in ExcluIR.

2.3 Rewriting synthetic queries

Although the prompt has been carefully adjusted, the generated queries often express the exclusionary phrases in a limited manner, such as "excluding any information about," "except for any information," and "without referencing any information about." These expressions lack naturalness and deviate from real-world queries. To increase the diversity and naturalness of the queries, we further instruct ChatGPT to rephrase them. Then, we partition the ExcluIR dataset obtained in this step into training and test sets. The test set is further manually corrected to construct the benchmark, which we will describe next.

2.4 Manually correcting data

To build a reliable ExcluIR benchmark, we hire 17 workers for manual data correction. We first sample 4,000 instances from the 74,293 exclusionary queries obtained in the previous step. Each instance contains two documents along with a synthetic query generated by ChatGPT. We ask workers to check the synthetic exclusionary query to ensure its naturalness and correctness and they are encouraged to express the exclusionary nature of queries using diverse expressions. The requirements are detailed in Appendix B. To facilitate the correction process, we construct an online correction system. In the system, we define three operations for workers to correct each data instance:

(1) *Criteria Met*. If the synthetic query already meets the criteria, no further modifications are necessary.

(2) *Query Modification*. If the synthetic query fails to meet the criteria, modify or rewrite the query to align with the requirements.

(3) *Discard Data*. If it is difficult to write a query that meets the criteria based on these two documents, the workers can choose to discard the data.

2.5 Quality assurance

We take several measures to ensure data quality: (i) we provide detailed documentation guidelines, including task definition, correction process, and specific criteria for exclusionary queries; (ii) we present multiple examples of exclusionary queries to help workers understand the task and requirements; (iii) we record a video to demonstrate the entire correction process and emphasize the key considerations that need special attention; (iv) we adopt a real-time feedback mechanism to allow workers to share the issues they encounter during the correction process; we discuss these issues and provide solutions accordingly; and (v) we randomly sample 10% of the data of each worker for quality inspection. If there are errors in the sampled data, we will ask the worker to correct the data again.

2.6 Dataset statistics

Following the dataset construction process described above, we obtain 3,452 human-annotated entries for the benchmark and 70,293 exclusionary queries for the training set. The average word counts for exclusionary queries in the training set and benchmark are 22.37 and 21.64, respectively. To further investigate the diversity of data, we visualize the distribution of the lengths of exclusionary queries in Figure 3. We show that the lengths of exclusionary queries are diverse, reflecting varying levels of complexity and details.

3 Experimental Setups

Methods for comparison. In our evaluation of different retrieval motheds for exclusionary retrieval, we select sparse retrieval (BM25, DocT5Query), dense retrieval (DPR, Sentence-T5, GTR, ColBERT), and generative retrieval (GENRE, SEAL, NCI) models. Detailed descriptions of these methods and implementation details are provided in Appendix C.

Evaluation metrics. For the original test

queries, we report the commonly used metrics: Recall at rank N (R@N, $N = 1, 5, 10$) and Mean Reciprocal Rank at rank N (MRR@N, $N = 10$). Recall measures the proportion of relevant documents that are retrieved in the top N results. MRR is the mean of the reciprocal of the rank of the first relevant document.

In ExcluIR, each exclusionary query q has a positive document d^+ and a negative document d^-. Thus, the difference between the rank of d^+ and the rank of d^- can reflect the retrieval model's capability of comprehending the exclusionary query. So we report ΔR@N and ΔMRR@N, which can be formulated as:

$$\begin{aligned} \Delta\text{R@}N &= \text{R@}N(d^+) - \text{R@}N(d^-), \\ \Delta\text{MRR@}N &= \text{MRR@}N(d^+) - \text{MRR@}N(d^-). \end{aligned} \tag{1}$$

In addition, we report Right Rank (RR), which is the proportion of results where d^+ is ranked higher than d^-. The expected value of RR is 50% with random ranking.

4 Results and Analyses

In this section, we present five groups of experimental results and analyses to study: (i) the performance of the existing retrieval models on ExcluIR (Section 4.1), (ii) the strategy to improve the performance on ExcluIR, including expanding the training data domain (Section 4.2), incorporating our dataset into the training data (Section 4.3), and increasing the size of the model (Section 4.4), and (iii) the explanation for the superiority of generative retrieval in ExcluIR (Section 4.5).

4.1 How well do existing methods perform on ExcluIR?

To evaluate the performance of various retrieval models trained on existing datasets in ExcluIR, we conduct our experiments on two well-known standard retrieval datasets: Natural Questions (NQ) (Kwiatkowski et al., 2019) and HotpotQA (Yang et al., 2018). NQ is a large-scale dataset for document retrieval and question answering. The version we use is NQ320k, which consists of 320k query-document pairs. HotpotQA is a question-answering dataset that focuses on multi-hop reasoning. We split the original HotpotQA in the same way as our ExcluIR dataset, resulting in a 70k training set and a 3.5k test set.

The main performance of retrieval methods on the ExcluIR benchmark and other test data are presented in Table 1 and 2. We have the following observations from the results.

First, although these methods achieve good performance on the standard test data including HotpotQA and NQ320k, their performance on the ExcluIR benchmark is unsatisfactory. Nearly all models score less than 10% higher than random ranking on the RR metric. Despite the fact that the Sentence-T5 and GTR models trained on NQ320k achieve the highest ΔR@1/ΔMRR/RR scores, they are far from achieving ideal performance. This is attributed to the fact that negative documents are erroneously retrieved and ranked high, indicating that these models fail to comprehend the exclusionary nature of queries.

Second, sparse retrieval methods demonstrate a significant limitation in comprehending the exclusionary nature of queries, so they have almost no ability to handle ExcluIR. As shown in Table 2, the RR scores of BM25 and DocT5Query are only 53.48% and 53.85%, which are only slightly higher than random. Their ΔR@1 and ΔMRR scores are lower than most neural retrieval models trained on NQ320k. This is because these methods are based on a lexical match between queries and documents. This limitation prevents them from focusing on the exclusionary phrases in the query, instead leading to a high relevance score for negative documents.

Third, the diversity of training data impacts the model's ability to comprehend exclusionary queries. As can be seen from Table 1 and 2, the models trained on NQ320k exhibit better performance on ExcluIR than those trained on HotpotQA. We believe this is because the queries in NQ320k are more diverse and contain more exclusionary queries. Therefore, increasing the domain and diversity of training data can be beneficial for exclusionary retrieval. We will conduct further experimental analysis in Section 4.2.

Furthermore, we also evaluate the performance of additional models trained on different datasets in ExcluIR. Due to space constraints, these results are presented in Appendix D.

4.2 How does expanding the training data affect the performance?

To further understand the impact of training data on performance in exclusionary retrieval, we select representative models from each category for additional experiments. We extend the experiment in Table 1 by adding the NQ320k dataset to the training data. We consider two settings for ex-

Type	Model	HotpotQA				ExcluIR				
		R@2	R@5	R@10	MRR	R@1	MRR	ΔR@1	ΔMRR	RR
Sparse Retrieval	BM25	67.16	76.65	80.98	92.47	49.68	65.17	7.82	4.66	53.48
	DocT5Query	69.19	77.88	81.65	94.10	50.98	67.50	7.85	3.81	53.85
Dense Retrieval	DPR	55.53	67.44	73.49	81.73	49.63	65.79	7.34	5.01	54.02
	Sentence-T5	57.63	68.45	74.29	82.48	51.04	66.27	10.11	7.01	55.41
	GTR	61.82	73.57	79.42	85.50	54.87	70.88	14.40	8.79	57.42
	ColBERT	73.58	83.73	87.95	94.44	54.00	71.24	10.72	6.42	55.57
Generative Retrieval	GENRE	48.87	51.67	53.24	75.25	48.03	63.22	4.35	0.13	52.10
	SEAL	60.78	72.26	78.20	85.76	51.33	67.88	11.64	7.71	55.52
	NCI	47.60	58.14	64.37	74.59	37.22	51.37	1.97	2.29	50.93

Table 1: Performance of models trained on HotpotQA and tested on HotpotQA and ExcluIR. For the evaluation on HotpotQA, we report Recall@2 rather than Recall@1, since each query in HotpotQA has two supporting documents.

Type	Method	NQ320k				ExcluIR				
		R@1	R@5	R@10	MRR	R@1	MRR	ΔR@1	ΔMRR	RR
Sparse Retrieval	BM25	37.96	61.24	68.86	47.86	49.68	65.17	7.82	4.66	53.48
	DocT5Query	42.63	66.18	73.38	52.69	50.98	67.50	7.85	3.81	53.85
Dense Retrieval	DPR	54.81	79.50	85.52	65.39	48.55	60.50	16.45	13.49	58.76
	Sentence-T5	59.63	82.78	87.42	69.57	57.76	66.34	32.90	27.96	67.83
	GTR	62.35	84.67	89.17	71.90	59.79	69.00	34.85	28.12	68.31
	ColBERT	60.08	84.19	89.41	70.50	57.01	70.88	20.02	15.26	59.97
Generative Retrieval	GENRE	56.25	71.21	74.00	62.80	31.63	37.63	11.44	10.15	58.65
	SEAL	55.24	75.13	80.97	63.86	43.54	55.17	16.11	15.27	60.02
	NCI	60.41	76.10	80.19	67.18	31.46	38.95	15.87	16.81	56.84

Table 2: Performance of models trained on NQ320k and tested on NQ320k and ExcluIR.

panding training data: "Mix" means mixing the two datasets for simultaneous training, and "Seq" means training on NQ320k with continual training on HotpotQA. The results in Table 3 show that the impact of expanding the training data domain on ExcluIR varies across models. Specifically, we have the following observations.

For the bi-encoder models, including DPR and Sentence-T5, the "Seq" strategy results in improved performance on ExcluIR. We believe that this is because the initial training on the NQ320k enhances the model's general comprehension capabilities, as evidenced by the improved performance on the HotpotQA test set.

However, expanding the training data does not help ColBERT and SEAL achieve better results on ExcluIR. While ColBERT exhibits competitive performance on two standard datasets, its performance diminishes on ExcluIR. This is because ColBERT calculates the document relevance score based on token-level matching, leading it to overlook exclusionary phrases in queries, which is crucial for exclusionary retrieval. We visualize the relevance calculation of ColBERT to further understand its performance in Appendix H. As for SEAL, the inherent limitation of generative retrieval models in poorly generalizing to new or out-of-distribution documents explains why expanding the training data does not lead to improved performance on ExcluIR (Lee et al., 2023; Mehta et al., 2023).

Overall, expanding training data does not stably enhance the performance of models on ExcluIR. We consider the primary reason to be the lack of exclusionary queries in the training data. Therefore, in the next section, we will investigate the impact of incorporating our training set which consists of exclusionary queries into the training data.

4.3 How does incorporating our dataset into training data affect the performance?

Previous experiments have demonstrated that models trained on HotpotQA and NQ320k perform un-

Model	Training Set	HotpotQA				ExcluIR				
		R@2	R@5	R@10	MRR	R@1	MRR	ΔR@1	ΔMRR	RR
DPR	HotpotQA	55.53	67.44	73.49	81.73	49.63	65.79	7.34	5.01	54.02
	NQ+H(Mix)	53.19	65.05	71.52	79.57	48.93	64.47	6.95	4.59	53.94
	NQ+H(Seq)	**56.91**	**69.02**	**74.59**	**82.74**	**50.87**	**67.12**	**8.66**	**5.99**	**54.66**
Sentence-T5	HotpotQA	57.63	68.45	74.29	82.48	51.04	66.27	10.11	7.01	55.41
	NQ+H(Mix)	54.32	65.67	72.02	79.56	51.45	66.58	11.27	8.71	56.10
	NQ+H(Seq)	**58.40**	**69.05**	**74.72**	**82.66**	**52.49**	**67.82**	**12.92**	**9.44**	**56.82**
ColBERT	HotpotQA	**73.58**	**83.73**	**87.95**	94.44	**53.69**	**70.82**	**10.64**	**6.35**	**55.53**
	NQ+H(Mix)	71.54	82.46	86.40	94.58	52.78	69.91	8.86	5.21	54.49
	NQ+H(Seq)	73.26	83.42	87.69	**94.68**	51.27	69.21	5.82	2.10	52.93
SEAL	HotpotQA	60.78	72.26	78.20	85.76	**51.33**	**67.88**	**11.64**	**7.71**	55.52
	NQ+H(Mix)	**61.65**	**72.80**	**78.61**	**86.39**	51.25	67.68	11.50	7.23	**55.63**
	NQ+H(Seq)	59.86	71.19	76.88	84.30	50.52	66.73	10.77	6.79	55.36

Table 3: Performance of models after expanding the training data domain. NQ+H(Mix) indicates mixing the NQ320k and HotpotQA datasets for simultaneous training. NQ+H(Seq) indicates initial training on the NQ320k dataset followed by continual training on the HotpotQA dataset.

satisfactorily on ExcluIR. We believe that this is partly due to a lack of exclusionary queries in the training data. Therefore, in this section, we incorporate the ExcluIR training set into the training data to assess its impact on performance. From the results in Figure 4, we have three main observations.

First, merging the ExcluIR training set into the training data can significantly enhance the model's ability to comprehend exclusionary queries. For instance, with NQ320k as the original dataset, SEAL achieves 18% improvement (60.02% vs. 78.02%) in RR by integrating the ExcluIR training set, with only a small (1.08%) decrease (63.86% vs. 62.78%) in performance on the original test data. This is because the ExcluIR training set contains a large number of exclusionary queries, which can help the retrieval model to better comprehend the exclusionary nature of queries.

Second, when training data contain exclusionary queries, generative retrieval methods are more adept at learning the exclusionary nature of queries compared to dense retrieval methods. As shown in Figure 4, although dense retrieval models trained on two original datasets perform better on ExcluIR, augmenting the ExcluIR training set leads to a greater improvement in generative retrieval models, ultimately surpassing dense retrieval methods overall. On average, generative retrieval models, including GENRE, SEAL, and NCI, achieve a 17.75% improvement, in contrast to the average 4.77% improvement observed in dense retrieval models. This is because the generative retrieval model is more suitable for capturing the complex relationships between queries and documents in terms of model architecture and training objectives. We present a more detailed analysis in Section 4.5.

Third, consistent with the conclusion in Section 4.2, ColBERT fails to achieve satisfactory performance, even after fine-tuning on ExcluIR. As demonstrated in Figure 4, among the models trained with the ExcluIR training set, ColBERT exhibits the lowest performance, with an RR score of 59.59% on HotpotQA w/ ExcluIR and 59.71% on NQ320k w/ ExcluIR. As mentioned in Section 4.2, the relevance score calculation method used by ColBERT is not conducive to handling exclusionary queries. We provide a more detailed analysis in Appendix H.

4.4 How does model size affect performance?

To analyze the impact of model size on the performance of ExcluIR, we increase model sizes of DPR, sentence-t5, GENRE, and NCI, and train them on different datasets. Specifically, for DPR, we use two variants: bert-base-uncased and bert-large-uncased. For sentence-t5, GENRE and NCI, we adopt t5-base and t5-large.

The results are presented in Table 4. We note that increasing the model size generally improves performance on ExcluIR when the training data includes exclusionary queries. This is consistent

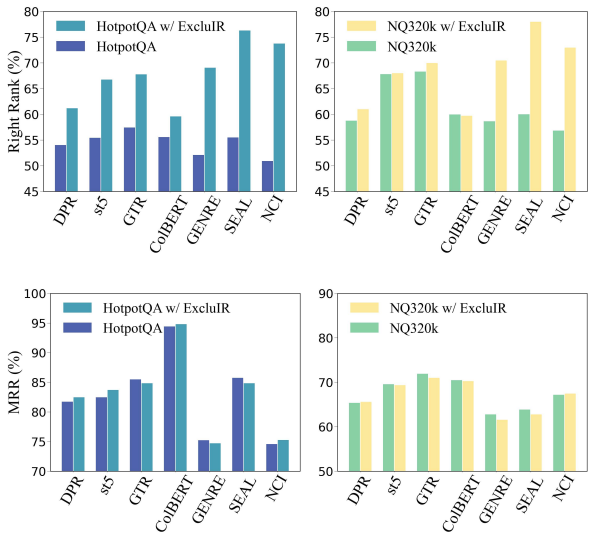

Figure 4: Performance of models under different training data settings. The upper figures show the RR score of various models on the ExcluIR benchmark, and the lower figures show the performance of these models on HotpotQA and NQ320k. The different colors of the bars represent different training data. Full results are presented in Appendix E.

Training set	Model	Base	Large
HotpotQA	DPR	54.02	54.25 ↑
	Sentence-T5	55.41	53.78 ↓
	GENRE	52.10	49.01 ↓
	NCI	50.93	50.64 ↓
HotpotQA w/ ExcluIR	DPR	61.19	62.63 ↑
	Sentence-T5	66.75	69.01 ↑
	GENRE	69.07	70.96 ↑
	NCI	73.75	73.61 ↓
NQ320k	DPR	58.76	61.62 ↑
	Sentence-T5	67.83	69.02 ↑
	GENRE	58.65	55.82 ↓
	NCI	56.84	62.54 ↑
NQ320k w/ ExcluIR	DPR	61.00	63.47 ↑
	Sentence-T5	68.00	69.65 ↑
	GENRE	70.48	72.86 ↑
	NCI	72.97	74.45 ↑

Table 4: RR scores with different model sizes on ExcluIR. For DPR, the base version is bert-base-uncased, and the large version is bert-large-uncased. For sentence-t5, GENRE and NCI, the base version is t5-base, and the large version is t5-large. ↑ indicates that an increase in model size improves performance, while ↓ indicates the opposite. Full results are presented in Appendix F.

with observations by Ravichander et al. (2022), who show that larger models are better at understanding the implications of negated statements in documents.

However, when training on original datasets, increasing the model size does not always lead to improved performance on ExcluIR. The results in Table 6 also support this observation. For example, the performance of stsb-roberta-large decreases significantly compared to stsb-roberta-base. This indicates that simply increasing model size cannot solve the challenges of exclusionary retrieval, we should investigate building more training data and proposing new training strategies.

4.5 Why are generative retrieval models superior in ExcluIR?

Generative retrieval models have inherent advantages in comprehending exclusionary queries. We try to analyze and explain the reason based on the architecture of generative models.

First, as a comparison, we show that bi-encoder models have a representation bottleneck for exclusionary queries. When two documents are similar but have some differences that the user would like to distinguish, it is difficult to ensure that the vector representation of the query remains distant from the negative document while closely aligning with the positive document. This representation bottleneck prevents the model from correctly comprehending the true intent of the query. We present this proof in Appendix G.

Generative retrieval models adopt a sequence-to-sequence framework, such as T5 or BART, which estimates the probability of generating the document IDs given the query using a conditional probability model: $P(d|q)$. When generating document IDs, multiple cross-attention layers in the decoder can capture the token-level semantic information in the query, a phenomenon also explored by Wu et al. (2024). Assuming the decoder consists of L layers, for the l-th layer ($0 \leq l < L$), the cross-attention layer is given by:

$$S^{(l+1)} = \text{softmax}\left(\frac{Q^{(l)}K^{(l)T}}{\sqrt{d_k}}\right)V^{(l)}, \quad (2)$$

where $Q^{(l)} = W_q^{(l)}S^{(l)}$, $K^{(l)} = W_k^{(l)}H_q^{(l)}$, $V^{(l)} = W_v^{(l)}H_q^{(l)}$, and $H_q^{(l)} = [e_{q_1}, \cdots, e_{q_N}]$ are query token vectors generated by encoder, $S^{(l)} = [e_{d_1}, \cdots, e_{d_M}]$ are generated embedding vectors for docid tokens at l-th layer, $W_q^{(l)}$, $W_k^{(l)}$ and $W_v^{(l)}$ are learnable cross-attention weight matrices. We visualize the cross attention in generative models to

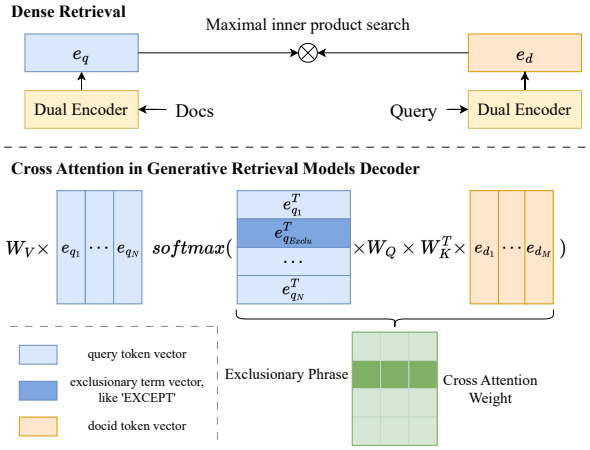

Figure 5: Summary of the analysis that shows the differences between dense retrieval and generative retrieval models in handling ExcluIR.

summarize our analysis. As shown in Figure 5, the multi-level cross-attention mechanism allows the model to strongly focus on key terms in the query, including exclusionary phrases (highlighted in dark green). Thus, even when faced with queries with complex semantics, generative retrieval models are capable of effectively capturing the query intent.

5 Related Work

Early studies in exclusionary retrieval primarily focus on keyword-based methods. These approaches typically treat user queries as logical expressions of boolean operations (Nakkouzi and Eastman, 1990; Strzalkowski, 1995; McQuire and Eastman, 1998; Harvey et al., 2003). However, these methods depend on explicit and deterministic rules, lack the flexibility to handle subtle and conditional exclusions, and are not suitable for more realistic retrieval scenarios.

In addition, there is a task related to exclusionary retrieval, known as argument retrieval (Wachsmuth et al., 2018), which aims to retrieve the best counterargument for a given argument on any controversial topic. While argument retrieval implicitly requires the model to find the counterargument to the query, the intention of exclusion is not explicitly expressed in the query. Wang et al. (2022b) first investigate exclusionary retrieval in Text-to-Video Retrieval (T2VR). They demonstrate that existing video retrieval models performed poorly when dealing with queries like "find shots of kids sitting on the floor and not playing with the dog." To the best of our knowledge, there has been no research on exclusionary retrieval in document retrieval.

(Weller et al., 2024) introduce NevIR, a benchmark designed to assess the ability of neural information retrieval systems to handle negation. NevIR requires retrieval models to rank two documents that differ only in negation, where both documents remain consistent in all other aspects except the key negation. Similarly, Rokach et al. (2008); Koopman et al. (2010) investigate the impact of negation contexts within documents on retrieval performance. For example, a search for "headache" might retrieve patient records containing "the patient has no symptoms of headache." Our work is different as we focus on exclusionary retrieval, studying whether the retrieval model can comprehend the intent of exclusionary queries.

6 Conclusion

In this work, we focus on a common yet understudied retrieval scenario called exclusionary retrieval, where users explicitly express which information they do not want to obtain. We have provided the community with a new benchmark, named ExcluIR, which focuses on exclusionary queries that explicitly express the information users do not want to obtain. We have conducted extensive experiments that demonstrate that existing retrieval methods with different architectures perform poorly on ExcluIR. Notably, ExcluIR cannot be solved by simply adding training data domains or increasing model sizes. Additionally, our analyses indicate that generative retrieval models inherently excel at comprehending exclusionary queries compared with sparse and dense retrieval models. We hope that this work can inspire future research on ExcluIR.

Limitations

This work has the following limitations. First, although the training data we build can significantly improve the performance of various retrieval models on ExcluIR, there is still a considerable gap from human performance (with RR score of 100%). In future work, we plan to investigate how to make use of the advantages of generative retrieval to further improve the ability of retrieval models in exclusionary retrieval. Second, in practical retrieval scenarios, the exclusions in the query can be expressed in different ways. Some are directly stated within a single-round query, while others are implied within the context of multi-round queries. For example, users might prefer that the results of the current query do not include content retrieved

in previous rounds, even though this intent of exclusion is not directly expressed within the query. In this work, we have only considered the former scenario, further research is required to explore a broader range of exclusionary retrieval scenarios.

Ethical Considerations

We realize the potential risks in the research of ExcluIR, thus, it is necessary to pay attention to the ethical issues. All raw data collected in this study are sourced from publicly available datasets, with ethical considerations approved by publishers. In the process of data annotation, all workers are informed of the research objectives in advance. We did not collect any personal or privacy-sensitive information and all data used in our research is obtained following legal and ethical standards.

References

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Scott Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive search engines: Generating substrings as document identifiers. *Advances in Neural Information Processing Systems*, 35:31668–31683.

Kendra Cherry. 2020. How we use selective attention to filter information and focus. Verywell Mind.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. In *ICLR 2021-9th International Conference on Learning Representations*, volume 2021. ICLR.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Valerie J. Harvey, Jeanne M. Baugh, Bruce A. Johnston, Constance M. Ruzich, Arthur J. Grant, et al. 2003. The challenge of negation in searches and queries. *Review of Business Information Systems (RBIS)*, 7(4):63–76.

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.

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.

Bevan Koopman, Peter Bruza, Laurianne Sitbon, and Michael Lawley. 2010. Analysis of the effect of negation on information retrieval of medical data. In *Proceedings of 15th Australasian Document Computing Symposium*, pages 89–92. School of Computer Science and IT, RMIT University.

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.

David L. LaBerge. 1990. Attention. *Psychological Science*, 1(3):156–162.

Hyunji Lee, Jaeyoung Kim, Hoyeon Chang, Hanseok Oh, Sohee Yang, Vladimir Karpukhin, Yi Lu, and Minjoon Seo. 2023. Nonparametric decoding for generative retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12642–12661.

April R. McQuire and Caroline M Eastman. 1998. The ambiguity of negation in natural language queries to information retrieval systems. *Journal of the American Society for Information Science*, 49(8):686–692.

Sanket Mehta, Jai Gupta, Yi Tay, Mostafa Dehghani, Vinh Tran, Jinfeng Rao, Marc Najork, Emma Strubell, and Donald Metzler. 2023. DSI++: Updating transformer memory with new documents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8198–8213.

Ziad S. Nakkouzi and Caroline M. Eastman. 1990. Query formulation for handling negation in information retrieval systems. *Journal of the American Society for Information Science*, 41(3):171–182.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022a. Sentence-T5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, et al. 2022b. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pre-trained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718.

Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to doctttttquery. *Online preprint*, 6:2.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Abhilasha Ravichander, Matt Gardner, and Ana Marasović. 2022. CONDAQA: A contrastive reading comprehension dataset for reasoning about negation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8729–8755.

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.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2825–2835.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Stephen E. Robertson and Steve Walker. 1997. On relevance weights with little relevance information. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 16–24.

Lior Rokach, Roni Romano, and Oded Maimon. 2008. Negation recognition in medical narrative reports. *Information Retrieval*, 11:499–538.

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.

Tomek Strzalkowski. 1995. Natural language information retrieval. *Information Processing & Management*, 31(3):397–417.

Anne M. Treisman. 1964. Selective attention in man. *British Medical Bulletin*, 20(1):12–16.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251.

Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, et al. 2022a. A neural corpus indexer for document retrieval. *Advances in Neural Information Processing Systems*, 35:25600–25614.

Ziyue Wang, Aozhu Chen, Fan Hu, and Xirong Li. 2022b. Learn to understand negation in video retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 434–443.

Orion Weller, Dawn Lawrie, and Benjamin Van Durme. 2024. NevIR: Negation in neural information retrieval. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2274–2287, St. Julian's, Malta. Association for Computational Linguistics.

Shiguang Wu, Wenda Wei, Mengqi Zhang, Zhumin Chen, Jun Ma, Zhaochun Ren, Maarten de Rijke, and Pengjie Ren. 2024. Generative retrieval as multi-vector dense retrieval. *arXiv preprint arXiv:2404.00684*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

A Prompt templates

We present the prompt that was used to guide ChatGPT in generating and rephrasing the exclusionary query for each pair of documents in Table 5.

B Requirements for manual correction

During manual correction, to ensure the quality of data, we provide the following requirements for workers.

(1) The query should be relevant to the positive document.

(2) The query should include an exclusionary constraint to clearly refuse to inquire about the information in the negative document.

(3) The query should contain enough information, not just using a person's name to represent a document.

Task	Prompt template
Generation	You will be provided with two documents, and you need to: 1. generate a query that is relevant to both Document 1 and Document 2; and 2. revise this query to include a constraint or condition that makes it explicitly refuse to inquire about any information in Document 1. Reply Format: Query: Revised Query:
Rephrasing	Rephrase the following query to make it smoother, more reasonable, more natural and more realistic. Do not answer this query but just polish it. You should make this query more like a real human query, but do not change the semantics of this query. Query: *raw query* Rewritten Query:

Table 5: Prompt templates for query generation and rephrasing.

(4) You should use diverse expressions to express the exclusionary constraint, rather than repetitively using the same terms like 'excluding.'

C The detailed experimental setups

C.1 Methods for comparison

To evaluate the performance of various retrieval models on exclusionary retrieval, we select three types of retrieval models with different architectures: sparse retrieval, dense retrieval, and generative retrieval.

Sparse retrieval methods calculate the relevance score of documents using term matching metrics such as TF-IDF (Robertson and Walker, 1997).

- **BM25** (Robertson and Zaragoza, 2009) is a classical probabilistic retrieval method based on the normalization of the frequency of the term and the length of the document.
- **DocT5Query** (Nogueira et al., 2019) expands documents by generating pseudo queries using a fine-tuned T5 model before building the BM25 index (Raffel et al., 2020).

Dense retrieval uses pre-trained language models (PLMs) as the backbones to represent queries and documents as dense vectors for computing relevance scores.

- **DPR** (Karpukhin et al., 2020) is a dense retrieval model based on dual-encoder architecture, which uses the representation of the [CLS] token of BERT (Devlin et al., 2019).
- **Sentence-T5** (Ni et al., 2022a) uses a fine-tuned T5 encoder model to encode queries and documents into dense vectors.
- **GTR** (Ni et al., 2022b) has the same architecture as Sentence-T5 and has been pretrained on two billion question-answer pairs collected from the Web.
- **ColBERT** (Khattab and Zaharia, 2020) is a late interaction model that learns embeddings for each token in queries and documents, and then uses a MaxSim operator to calculate the relevance score.

Generative retrieval is an end-to-end retrieval paradigm.

- **GENRE** (De Cao et al., 2020) retrieves entities by generating their names through a seq-to-seq model, it can be applied to document retrieval by directly generating document titles. The original GENRE is trained based on BART as the backbone, and we reproduce it using T5.
- **SEAL** (Bevilacqua et al., 2022) retrieves documents by generating n-grams within them.
- **NCI** (Wang et al., 2022a) proposes a prefix-aware weight-adaptive decoder architecture, leveraging semantic document identifiers and various data augmentation strategies like query generation.

C.2 Implementation details

In our experiments, we use Elasticsearch to evaluate BM25 on both raw documents and the documents augmented with DocT5Query. We train DPR and ColBERT using the bert-base-uncased architecture, train Sentence-T5, GENRE, and NCI using the t5-base architecture, and train SEAL using the BART-large architecture. We reproduce NCI and SEAL by their official implementations and other methods are reproduced by our own implementations. For query generation, we use a pre-trained model, DocT5Query (Nogueira et al., 2019), to generate pseudo queries for each document. For the training of neural retrieval models, the max input length is set to 256 and the batch size is set to 32.

D The experimental results of additional models on ExcluIR

We present more results in Table 6 showing the performance of models on the ExcluIR dataset. Most of the models are from sentence-transformers (Reimers and Gurevych, 2019), except for RocketQA (Qu et al., 2021; Ren et al., 2021) and monot5 (Nogueira et al., 2020). Since cross-encoder models are used for re-ranking, it is very time-consuming to calculate the relevance score of all documents in the corpus. Therefore, We first retrieve the top 100 documents using BM25, and then re-rank them. We find that the Recall@100 of BM25 for positive and negative documents is 95.77% and 94.74%, so this strategy can ensure fairness.

E The complete results of training with ExcluIR on HotpotQA and NQ320k

Table 7 and 8 show full results of retrieval models performance after augmenting the HotpotQA and NQ320k with the ExcluIR training set, respectively.

F The complete results of different model sizes on ExcluIR

Table 9 shows the complete results of different model sizes on ExcluIR.

G Limitations of bi-encoder models in ExcluIR with similar positive and negative documents.

Bi-encoder models embed queries and documents into a high-dimensional space to compute the relevance score. These methods are effective when the semantics of the query and documents are straightforward and do not overlap. However, in ExcluIR, exclusionary queries contain the semantics of negative documents. We demonstrate that bi-encoder models struggle to distinguish between positive and negative documents when their vector representations are close in embedding space. This limitation leads to a bottleneck for bi-encoder models in ExcluIR. Here is the proof.

Definition 1. Let A, B be queries or documents. We define $q_{A,B} \coloneqq \boldsymbol{f}(A, B)$ where $\boldsymbol{f}(A, B)$ is the query encoding vector of query A and negative query B. $d_A \coloneqq \boldsymbol{g}(A)$ where $\boldsymbol{g}(\cdot)$ is the document encoding vector function. All vectors are normalized to 1.

We also define x, y to be *ε-close* if there exists $\delta \in (0, \frac{1}{9})$ such that $\Pr(\langle x, y\rangle > 1 - \varepsilon) > 1 - \delta$.

Assumption 1. *We make the following assumptions, with A, B as in Definition 1.*

- *d_A and d_B are ε-close, which means both A and B are related documents but have some differences that the user would like to distinguish.*
- *$q_{A,B}$ and d_A are ε-close, which means $q_{A,B}$ have good representation to retrieve d_A. Similar is true for $q_{B,A}$ and d_B.*
- *$\langle q_{B,A}, d_B\rangle - \langle q_{B,A}, d_A\rangle \geq 1 - \varepsilon$ with high probability, which means $q_{B,A}$ prefers d_B rather than d_A.*
- *$\varepsilon < 3 - 2\sqrt{2}$.*

Claim 1. *With A, B as in Definition 1, with high probability we have*

$$\langle q_{A,B}, d_B\rangle - \langle q_{A,B}, d_A\rangle > 0.$$

Proof. We reason as follows:

$$\begin{aligned}
&\langle q_{A,B}, d_B\rangle - \langle q_{A,B}, d_A\rangle && (3)\\
&= \langle q_{A,B} - q_{B,A}, d_B - d_A\rangle && (4)\\
&\quad + \langle q_{B,A}, d_B\rangle - \langle q_{B,A}, d_A\rangle. && (5)
\end{aligned}$$

Specifically,

$$\begin{aligned}
&|\langle q_{A,B} - q_{B,A}, d_B - d_A\rangle| && (6)\\
&\leq \|q_{A,B} - q_{B,A}\|\|d_B - d_A\| && (7)\\
&\leq \sqrt{\|q_{A,B}\|^2 + \|q_{B,A}\|^2}\|d_B - d_A\| && (8)\\
&= \sqrt{2}\sqrt{2 - 2\langle d_B, d_A\rangle} && (9)\\
&\leq 2\sqrt{\varepsilon}. && (10)
\end{aligned}$$

Type	Training Data	Params	Model	ΔR@1	ΔMRR	RR
Bi-Encoders	MSMarco	218M	RocketQA v1	31.62	24.94	65.13
	NQ	218M	RocketQA v1	25.09	21.47	61.31
	NQ	218M	RocketQA v2	17.93	15.74	53.61
	Multi-Datasets	23M	all-MiniLM-L6-v2	26.41	19.42	62.09
	Multi-Datasets	33M	all-MiniLM-L12-v2	27.62	21.05	63.29
	Multi-Datasets	109M	all-mpnet-base-v2	**39.32**	**32.01**	**69.04**
	Multi-Datasets	82M	all-distilroberta-v1	37.56	27.63	67.98
	Multi-Datasets	66M	multi-qa-distilbert-cos-v1	25.90	18.42	61.77
	Multi-Datasets	109M	multi-qa-mpnet-base-dot-v1	37.80	29.04	68.41
	Multi-Datasets	23M	multi-qa-MiniLM-L6-cos-v1	24.11	17.87	60.97
	Multi-Datasets	278M	paraphrase-multilingual-mpnet-base-v2	33.72	27.61	65.85
Cross-Encoders	MSMarco	23M	ms-marco-MiniLM-L-6-v2	27.56	16.61	63.35
	MSMarco	33M	ms-marco-MiniLM-L-12-v2	27.08	16.47	63.12
	SQuAD	109M	qnli-electra-base	23.60	**27.76**	53.87
	STSB	125M	stsb-roberta-base	13.48	15.13	59.38
	STSB	355M	stsb-roberta-large	6.50	8.26	50.27
	MSMarco	223M	monot5-base-msmarco-10k	32.54	18.87	65.85
	MSMarco	738M	monot5-large-msmarco-10k	**42.80**	23.71	**70.91**
	MSMarco	2852M	monot5-3b-msmarco-10k	42.17	23.35	70.74
	MSMarco	109M	RocketQA-v2_marco_ce	37.22	21.11	68.24
	MSMarco	335M	RocketQA-v1_marco_ce	40.39	22.40	70.02
	NQ	335M	RocketQA-v1_nq_ce	41.56	22.98	70.48

Table 6: The performance of various models on ExcluIR. Training Data indicates the source of training data for the model, and Params indicates the number of parameters in the model.

Therefore, with probability $1 - 3\delta$ we have

$$\langle q_{A,B}, d_B \rangle - \langle q_{A,B}, d_A \rangle \quad (11)$$
$$\geq -2\sqrt{\varepsilon} + \langle q_{B,A}, d_B \rangle - \langle q_{B,A}, d_A \rangle \quad (12)$$
$$\geq -2\sqrt{\varepsilon} + 1 - \varepsilon \quad (13)$$
$$> 0. \qquad \square$$

H Why does ColBERT underperform in ExcluIR?

Late interaction models like ColBERT struggle to comprehend the exclusionary nature of queries. From the previous experimental results, we can see that ColBERT performs worse than other neural retrieval models in ExcluIR. As ColBERT uses a late interaction architecture, it calculates document relevance scores based on the matching of token-level vectors between queries and documents. Consequently, exclusionary phrases in queries pose a challenge for matching with document tokens.

As we can see in Figure 6, the token 'exclude' in the query exhibits relatively low relevance with every token in the negative document. This indicates that ColBERT barely comprehends the true intent of the query. We also notice that 'decemberists' appears both in the query and negative document, contributing a very high relevance score, which is disadvantageous for exclusionary retrieval. Although the 'decemberists' band is mentioned in the query, the intent of the query is to avoid retrieving information about this band. Therefore, ColBERT inherently lacks the capability to comprehend the queries with complex intentions, limiting its effectiveness in ExcluIR. We present more cases in Table 7–10.

I Cases from the ExcluIR dataset

Table 10 shows some cases taken from the ExcluIR dataset.

Model	Training Data	HotpotQA				ExcluIR				
		R@2	R@5	R@10	MRR	R@1	MRR	ΔR@1	ΔMRR	RR
DPR	HotpotQA	55.53	67.44	73.49	81.73	49.63	65.79	7.34	5.01	54.02
	H. w/ ExcluIR	**58.26**	**70.48**	**76.81**	**83.60**	**59.30**	**73.20**	**24.45**	**17.88**	**62.63**
Sentence-T5	HotpotQA	57.63	68.45	74.29	82.48	51.04	66.27	10.11	7.01	55.41
	H. w/ ExcluIR	**58.65**	**69.60**	**75.48**	**83.72**	**63.73**	**75.85**	**33.78**	**24.49**	**66.75**
GTR	HotpotQA	61.82	73.57	79.42	**85.50**	54.87	70.88	14.40	8.79	57.42
	H. w/ ExcluIR	**61.99**	**73.83**	**79.45**	84.86	**64.98**	**77.75**	**34.85**	**23.85**	**67.79**
ColBERT	HotpotQA	**73.58**	**83.73**	**87.95**	94.44	54.00	71.24	10.72	6.42	55.57
	H. w/ ExcluIR	72.90	83.26	87.50	**94.80**	**58.14**	**74.95**	**18.80**	**12.74**	**59.59**
GENRE	HotpotQA	**48.87**	**51.67**	**53.24**	**75.25**	48.03	63.22	4.35	0.13	52.10
	H. w/ ExcluIR	48.60	51.26	53.03	74.71	**64.98**	**72.54**	**38.71**	**18.34**	**69.07**
SEAL	HotpotQA	**60.78**	72.26	**78.20**	**85.76**	51.33	67.88	11.64	7.71	55.52
	H. w/ ExcluIR	60.34	**72.39**	77.97	84.85	**69.03**	**78.66**	**48.95**	**39.55**	**76.29**
NCI	HotpotQA	47.60	58.14	64.37	74.59	37.22	51.37	1.97	2.29	50.93
	H. w/ ExcluIR	**47.80**	**59.15**	**64.75**	**75.28**	**59.76**	**68.90**	**42.29**	**38.38**	**73.75**

Table 7: The complete results of the impact of augmenting HotpotQA with ExcluIR training set.

Model	Training Data	NQ320k				ExcluIR				
		R@1	R@5	R@10	MRR	R@1	MRR	ΔR@1	ΔMRR	RR
DPR	NQ320k	54.81	**79.50**	**85.52**	65.39	48.55	60.50	16.45	13.49	58.76
	N. w/ ExcluIR	**55.08**	79.31	85.49	**65.58**	**55.04**	**67.89**	**21.52**	**16.38**	**61.00**
Sentence-T5	NQ320k	59.63	**82.78**	**87.42**	**69.57**	57.76	66.34	32.90	**27.96**	67.83
	N. w/ ExcluIR	**59.80**	81.58	87.13	69.36	**63.09**	**74.57**	**34.47**	26.19	**68.00**
GTR	NQ320k	**62.35**	**84.67**	**89.17**	**71.90**	59.79	69.00	34.85	28.12	68.31
	N. w/ ExcluIR	61.44	83.82	88.34	71.01	**65.64**	**76.98**	**39.05**	**28.46**	**69.98**
ColBERT	NQ320k	60.08	**84.19**	**89.41**	**70.50**	57.01	70.88	**20.02**	**15.26**	**59.97**
	N. w/ ExcluIR	**60.20**	83.59	88.60	70.29	**57.91**	**73.52**	19.30	13.05	59.71
GENRE	NQ320k	**56.25**	**71.21**	**74.00**	**62.80**	31.63	37.63	11.44	10.15	58.65
	N. w/ ExcluIR	55.15	70.00	72.85	61.55	**65.67**	**73.01**	**41.19**	**20.31**	**70.48**
SEAL	NQ320k	**55.24**	**75.13**	**80.97**	**63.86**	43.54	55.17	16.11	15.27	60.02
	N. w/ ExcluIR	53.86	74.84	80.34	62.78	**70.39**	**78.40**	**52.14**	**43.25**	**78.02**
NCI	NQ320k	60.41	76.10	80.19	67.18	31.46	38.95	15.87	16.81	56.84
	N. w/ ExcluIR	**60.61**	**76.53**	**80.55**	**67.46**	**56.92**	**64.67**	**41.13**	**39.92**	**72.97**

Table 8: The complete results of the impact of augmenting NQ320k with ExcluIR training set.

Training set	Model	Base			Large		
		ΔR@1	ΔMRR	RR	ΔR@1	ΔMRR	RR
HotpotQA	DPR	7.34	5.01	54.02	8.00	6.22	54.25 ↑
	Sentence-T5	10.11	7.01	55.41	7.21	5.23	53.78 ↓
	GENRE	4.35	0.13	52.10	−1.71	−3.09	49.01 ↓
	NCI	1.97	2.29	50.93	1.05	1.41	50.64 ↓
HotpotQA w/ ExcluIR	DPR	21.32	14.93	61.19	24.55	17.88	62.63 ↑
	Sentence-T5	33.78	24.49	66.75	37.05	26.50	69.01 ↑
	GENRE	38.71	18.34	69.07	42.15	20.20	70.96 ↑
	NCI	42.29	38.38	73.75	43.74	38.56	73.61 ↑
NQ320k	DPR	16.45	13.49	58.76	20.83	17.16	61.62 ↑
	Sentence-T5	32.90	27.96	67.83	34.36	29.94	69.02 ↑
	GENRE	11.44	10.15	58.65	11.03	8.88	55.82 ↓
	NCI	15.87	16.81	56.84	21.27	22.86	62.54 ↑
NQ320k w/ ExcluIR	DPR	21.52	16.38	61.00	25.52	19.15	63.47 ↑
	Sentence-T5	34.47	26.19	68.00	37.34	28.70	69.65 ↑
	GENRE	41.19	20.31	70.48	46.04	23.24	72.86 ↑
	NCI	41.13	39.92	72.97	43.13	41.86	74.45 ↑

Table 9: Performance with different model sizes on ExcluIR.

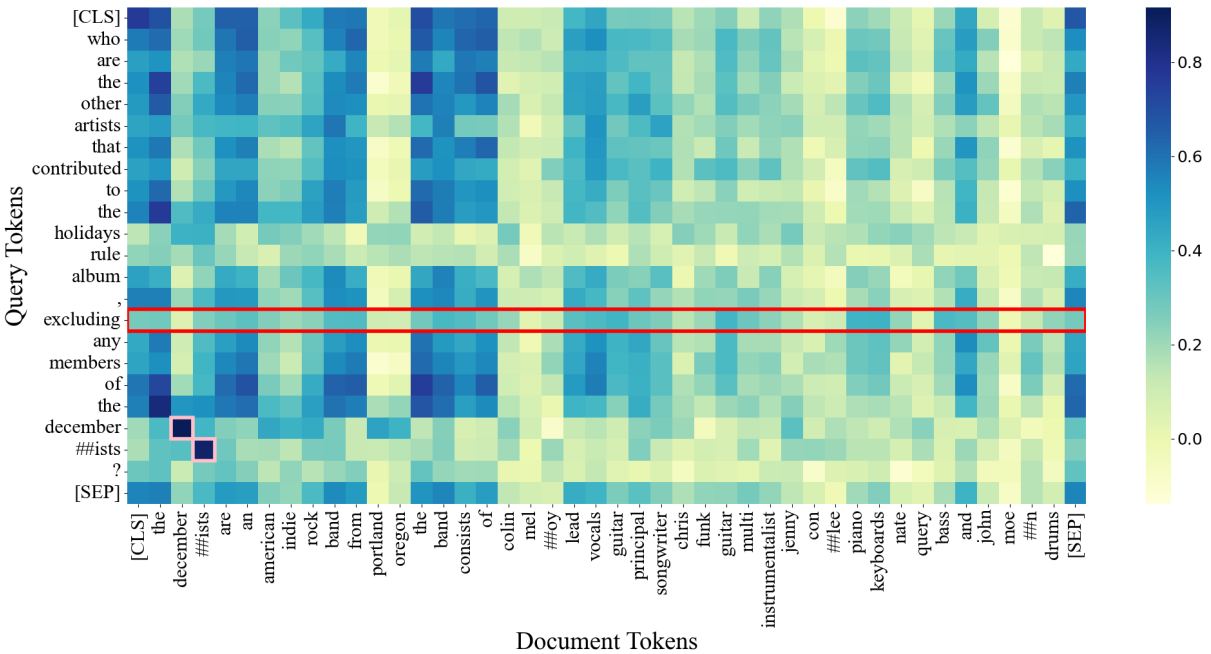

Figure 6: A relevance calculation visualization between query and negative document of ColBERT. Each value in the heatmap represents the result of the dot product between the query token vector and the negative document token vector. The red highlight indicates the relevance of the token 'excluding' in the query to each token in the negative document, and the pink highlights indicate the token with the highest relevance score.

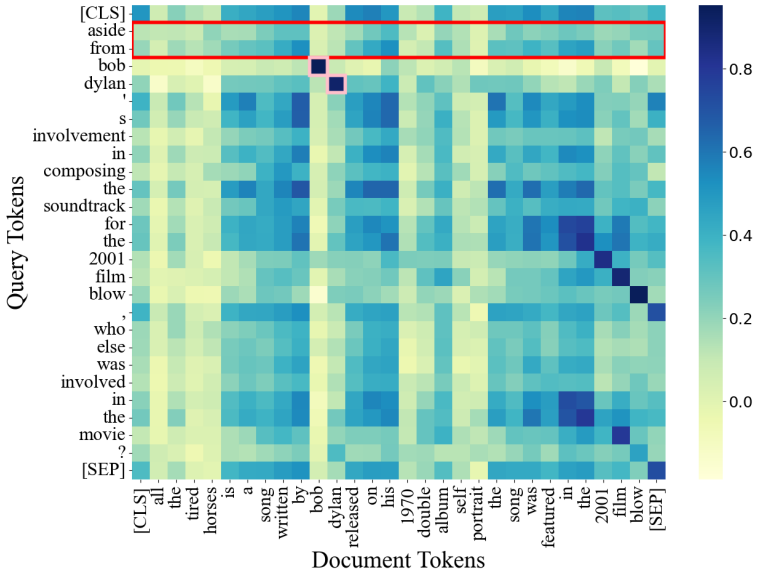

Figure 7: An example of ColBERT on negative document relevance scoring. ColBERT overlooks the semantics of 'aside from' and instead, due to the presence of lexical matches such as 'bob dylan', assigned a high relevance score to this negative document.

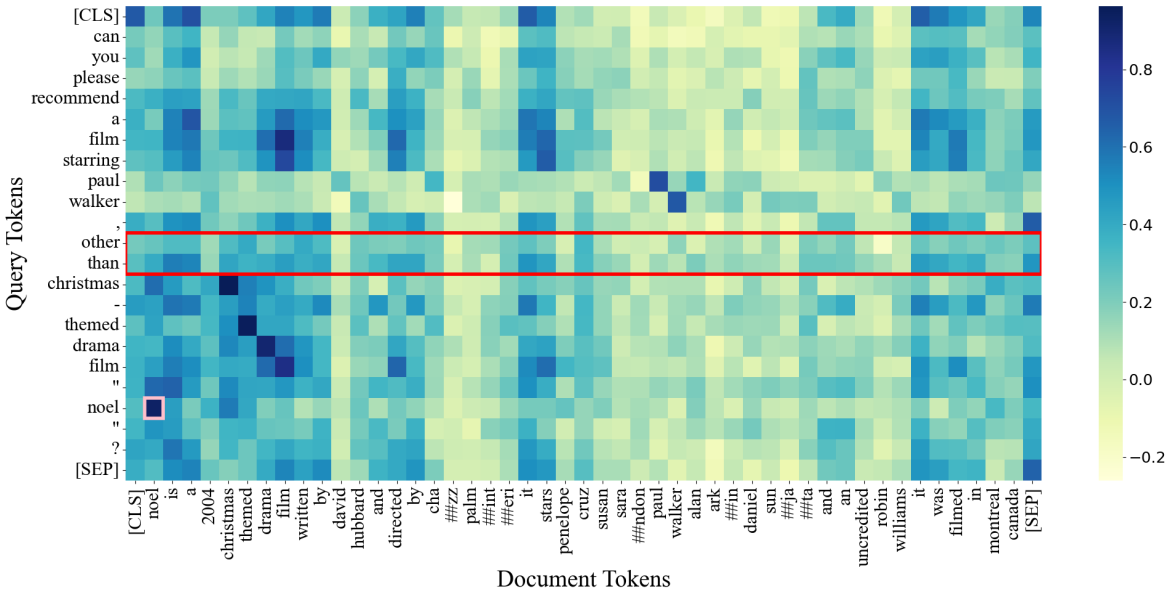

Figure 8: An example of ColBERT on negative document relevance scoring. ColBERT overlooks the semantics of 'other than' and instead, due to the presence of lexical matches such as 'noel', assigned a high relevance score to this negative document.

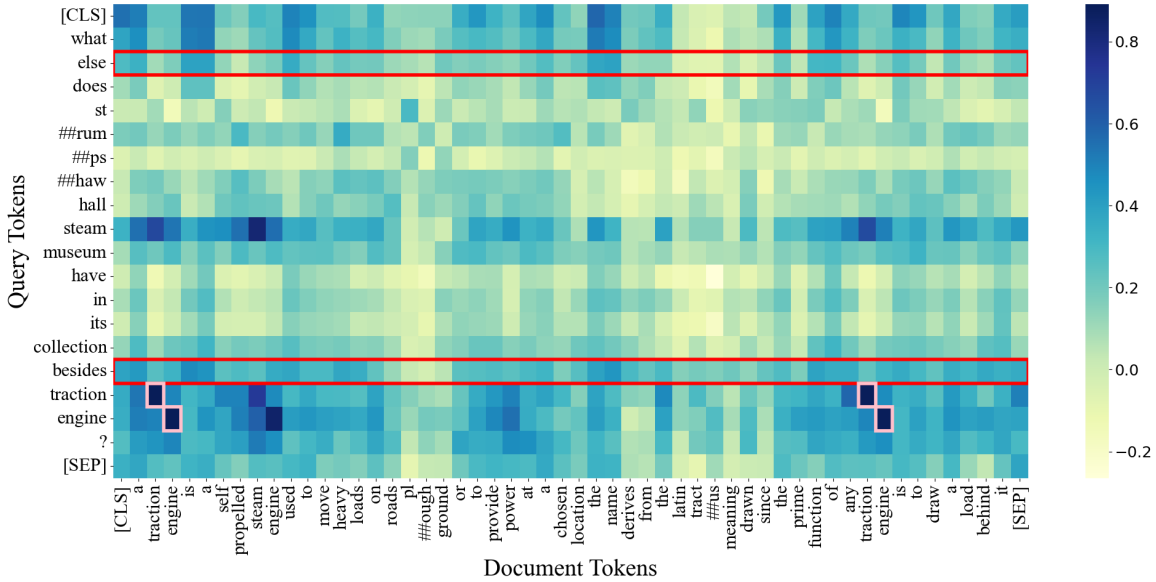

Figure 9: An example of ColBERT on negative document relevance scoring. ColBERT overlooks the semantics of 'else', 'besides' and instead, due to the presence of lexical matches such as 'traction engine', assigned a high relevance score to this negative document.

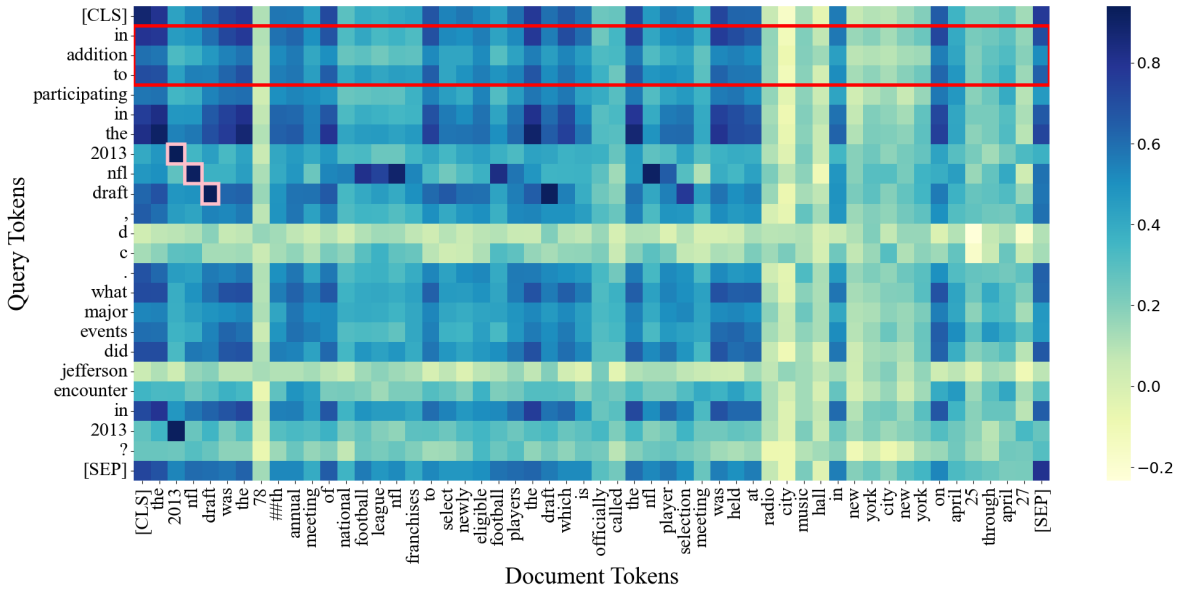

Figure 10: An example of ColBERT on negative document relevance scoring. ColBERT overlooks the semantics of 'in addition to' and instead, due to the presence of lexical matches such as '2013 nfl draft', assigned a high relevance score to this negative document.

Exclusionary query	Aside from Bob Dylan's involvement in composing the soundtrack for the 2001 film Blow, who else was involved in the movie?
Positive document	Blow is a 2001 American biographical crime film about the American cocaine smuggler George Jung, directed by Ted Demme. David McKenna and Nick Cassavetes adapted Bruce Porter's 1993 book "Blow: How a Small Town Boy Made $100 Million with the Medellín Cocaine Cartel and Lost It All" for the screenplay. It is based on the real-life stories of George Jung, Pablo Escobar, Carlos Lehder Rivas (portrayed in the film as Diego Delgado), and the Medellín Cartel. The film's title comes from a slang term for cocaine.
Negative document	"All the Tired Horses" is a song written by Bob Dylan, released on his 1970 double album "Self Portrait". The song was featured in the 2001 film "Blow".
Exclusionary query	Can you please recommend a film starring Paul Walker, other than Christmas-themed drama film "Noel"?
Positive document	Paul William Walker IV (September 12, 1973 – November 30, 2013) was an American actor. Walker began his career guest-starring in several television shows such as "The Young and the Restless" and "Touched by an Angel". Walker gained prominence with breakout roles in coming of age and teen films such as "She's All That" and "Varsity Blues" (1999). In 2001, Walker gained international fame for his portrayal of Brian O'Conner in the street racing action film "The Fast and the Furious" (2001), and would reprise the role in five of the next six installments but died in the middle of the filming of "Furious 7" (2015). He also starred in films such as "Joy Ride" (2001), "Timeline" (2003), "Into the Blue" (2005), "Eight Below", and "Running Scared" (2006).
Negative document	Noel is a 2004 Christmas-themed drama film written by David Hubbard and directed by Chazz Palminteri. It stars Penélope Cruz, Susan Sarandon, Paul Walker, Alan Arkin, Daniel Sunjata and an uncredited Robin Williams. It was filmed in Montreal, Canada.
Exclusionary query	In addition to participating in the 2013 NFL Draft, D C. What major events did Jefferson encounter in 2013?
Positive document	D. C. Jefferson (born May 7, 1989) is an American football tight end who is currently a free agent. He played college football at Rutgers University. He was drafted in the seventh round with the 219th overall pick by the Arizona Cardinals in the 2013 NFL Draft. Jefferson was released on November 4, 2013 after he was arrested on suspicion of driving under the influence.
Negative document	The 2013 NFL draft was the 78th annual meeting of National Football League (NFL) franchises to select newly eligible football players. The draft, which is officially called the "NFL Player Selection Meeting," was held at Radio City Music Hall in New York City, New York, on April 25 through April 27.

Table 10: Cases of ExcluIR dataset.