Exploring methods to improve effectiveness of ad-hoc retrieval systems for long and complex queries

Dorian Erhan[1]

Supervisor(s): Avishek Anand [1]**, Jurek Leonhardt**[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Dorian Erhan
Final project course: CSE3000 Research Project
Thesis committee: Avishek Anand, Jurek Leonhardt, Alan Hanjalic

Abstract

Ad-hoc retrieval involves ranking a list of documents from a large collection based on their relevance to a given input query. These retrieval systems often show poorer performances when handling longer and more complex queries. This paper aims to explore methods of improving retrieval effectiveness on these types of queries across different information retrieval (IR) tasks, within the context of Fast-Forward indexes. An analysis is conducted to determine the actual impact of query length and complexity. Interestingly, the hypothesis that longer queries are more challenging does not hold true for all cases, and in some datasets the opposite is true. To improve the performance of long and complex queries, two approaches are explored: utilising multiple dense models during the re-ranking stage instead of the traditional single model and reducing the queries via large language models. The use of multiple dense models for re-ranking proves to be effective, with two models providing the best balance between performance and ranking quality. Utilising LLM's for query reduction achieves performance similar to the original queries but fails to improve their ranking scores.

1 Introduction

Ad-hoc retrieval is the task of returning a list of documents from a large collection, such that the documents are ordered by their relevance with respect to an input query. These information retrieval systems are essential for a wide range of technologies such as: web search engines, digital libraries and recommender systems.

One of the major impediments of ad-hoc retrieval systems is that they tend to perform poorly on long and complex queries, even though they are used extensively [1]. Various solutions have been proposed in order to increase the effectiveness for such cases. One approach involves reducing the size of the queries by removing redundant terms that do not influence their overall meaning [2, 3]. Similarly, query reweighting can be applied by identifying the most important keywords in the query and assigning them greater weights before processing them through the retrieval model [4]. The recent employment of deep learning models for retrieval, which better capture the semantic relationships between query terms and documents, also improved the ability to match complex queries with the relevant documents.

Improving the quality of search with long and complex queries is extensively researched in the context of web search [2,5,6]. However, there is a lack of research addressing other information retrieval tasks. Therefore, this study aims to fill this gap by attempting to enhance the effectiveness of long queries across various IR domains. Additionally, it explores a novel method of boosting performance by utilising multiple semantic scoring models for re-ranking an initial set of relevant documents.

This research focuses specifically on Fast-Forward Indexes proposed by Leonhardt et al. [7, 8]. This is an efficient end-to-end framework for ranking long documents without compromising effectiveness. It exploits the capabilities of dual-encoders for the re-ranking phase instead of the retrieval phase and combines the lexical and semantic scores via interpolation.

Thus, this paper aims to answer the following research question: **Can the effectiveness for long and complex queries be improved on Fast-Forward indexes?** To address this question, the following sub questions will be explored:

1. How does query length and complexity affect the re-ranking performance of different encoders on Fast-Forward indexes?
2. What strategies can be employed to improve the effectiveness for long and complex queries on Fast Forward indexes?

Experiments are carried out on four widely-used IR datasets, including TREC-COVID [9], SciFact [10], HotpotQA [11] and Arguana [12]. The results obtained show that query length negatively impacts ranking performance, however, in some cases, shorter queries appear to be as challenging, if not more, than longer ones. Among the strategies tested to enhance query effectiveness, using multiple dense models for re-ranking yields the best results, surpassing the traditional single re-ranker approach. Utilizing large language models (LLMs) without fine-tuning for query reduction delivers performance comparable to the original queries, though it does not exceed their ranking scores.

The paper is structured as follows. In section 2, the theoretical background and the necessary context is presented. Section 3 describes the methodology employed. Section 4 goes over the experimental setup, enumerating the models, datasets and evaluation metrics used. Section 5 presents and analyses the obtained experimental results. Lastly, section 6 discusses the ethical implications of this research.

2 Background

Ad-hoc retrieval has traditionally been dominated by sparse retrieval methods. These methods represent documents as sparse high-dimensional vectors. This can be done using Bag-of-Words models such as BM25 [13], that rely on exact term matching via inverted indexes, ranking documents based on the amount of overlapping terms they have with the query. Although transformer-based approaches are more prevalent in dense retrieval, recent advancements like SPLADE [14] introduced neural-based methods for sparse retrieval as well. These neural models typically infer connections between terms and use that knowledge to enhance the sparse vector embeddings. However, sparse methods tend to have a significant shortcoming: they often fail to capture semantic and contextual information. Consequently, they suffer from the vocabulary mismatch problem, poorly estimating query-document relevance when the terms used in the query do not exactly match the ones used in the relevant documents as seen in Figure 1.

A recent wave of advancements has seen the rise of dense retrieval methods built upon pre-trained large language models [15]. Here the text inputs are embedded into dense vector

Figure 1: Example of the mismatch problem. Document 1 is contextually relevant but has low lexical similarity to the query, whereas document 2 is irrelevant but has high lexical similarity to the query.

representations in a lower dimensional vector space. These dense representations are typically obtained via neural models that are trained to capture semantic information, allowing semantically similar passages to be mapped close to each other in the vector space. This makes retrieval equivalent to performing an approximate nearest neighbor (ANN) search given the vector form of the query. Dense retrieval has recently demonstrated superiority over sparse models in terms of effectiveness [16], however, they are significantly more computationally expensive to use and train, and they are less efficient for large corpora compared to their sparse counterparts.

To address the inefficiency of dense models, several approaches have been developed. Dual-encoder models [15] are a common architecture of dense retrievers. They use language models to encode the queries and documents separately into their own vector representations. A similarity metric (e.g. the dot-product) is used between the query and the document vectors to determine their relevance. By isolating the computations between the queries and documents, this architecture makes it possible to precompute and index all the document vector representations in an offline phase, prior to retrieval.

Another approach is retrieve-and-re-rank. This method uses an efficient sparse retriever to prune out a large set of irrelevant documents from the corpus and obtain a smaller candidate set of relevant documents. Then in a second stage, a more computationally expensive dense neural ranker is employed to re-rank the selected candidates and reorder them, in order to promote the most semantically relevant documents to higher ranks [17].

Fast-Forward index [7, 8] is an end-to-end framework that makes use of dual-encoder models on the re-ranking stage instead of the retrieval stage. An efficient sparse retriever is employed in the first stage, in order to retrieve the top-k documents (k_s). For a query q and a document d, we denote the sparse score of a query-document pair as $\phi_S(q, d)$. In the second stage, a dual-encoder model is employed as a re-ranker. A query encoder ζ and a document encoder η will map the queries and documents to a common vector space. We denote the dense score as $\phi_D(q, d)$, and it is obtained by calculating the the similarity between these vector representations via the dot product:

$$\phi_D(q, d) = \zeta(q) \cdot \eta(d)$$

Fast-Forward indexes make use of dual-encoders to index all document vectors in an offline stage. Thus, computing dense scores consists of merely looking up these pre-computed representations and calculating the similarity with the query.

The final score $\phi_f(q, d)$ is obtained via interpolation-based re-ranking, where the sparse score $\phi_S(q, d)$ and dense score $\phi_D(q, d)$ are assigned different weights, according to an hyper-parameter α:

$$\phi_f(q, d) = \alpha \cdot \phi_S(q, d) + (1 - \alpha) \cdot \phi_D(q, d)$$

Setting $\alpha = 0$ disregards the sparse score from the final ranking score, however research shows including it can actually improve effectiveness [18], thus it was included in all experiments.

3 Methodology

3.1 Effect of query length & complexity on effectiveness

To guide the development of strategies aimed at improving ranking quality, an analysis is conducted to precisely assess the impact of query length and complexity. This analysis consists of two evaluations: one at the dataset level and one at the individual query level.

For the dataset-level evaluation, retrieval is performed on datasets corresponding to different IR tasks. These datasets feature a wide range of query lengths, enabling a comparison of the performance of Fast-Forward indexes across queries of varying levels of complexity. Furthermore, one particular dataset (Arguana) contains queries that exceeds the context length of the models utilised. This provides additional insight into how much this truncation impacts ranking quality.

The element wise evaluation involves measuring the retrieval quality for each individual query of the dataset and plotting it against its respective length. The anticipated outcome is that longer queries will have a significantly lower quality compared to shorter ones.

3.2 Improving effectiveness on long & complex queries

The present study explores two methods aimed at enhancing the performance for long and complex queries.

Multiple semantic scoring models

Traditionally, retrieve-and-re-rank utilises a single model to re-rank the most relevant documents retrieved in the first stage. This method allows for a better, computationally expensive model to process only a small subset of the original corpus.

One of the approaches considered to improve long query effectiveness is the usage of multiple re-rankings. Instead of relying on a single model, two to three different models will be employed to re-rank the documents retrieved in the first stage. The final ranking score is derived by interpolating the dense scores from each model, with different weights assigned to each. Each dense scoring model captures the underlying semantic concepts slightly differently, thus the hypothesis is that leveraging multiple of them might achieve a more robust ranking for these types of queries.

However, the increased computational requirements must be considered. While more models can potentially lead to better results, they also demand more processing time. Therefore we considered employing two to three models to be the best

compromise between performance and improving the effectiveness of long queries.

Query simplification

Transformer based dense models typically have a fixed input sequence length. When the input exceeds this amount of tokens, the overflow is truncated. While the sequence length is usually sufficient for most datasets, very long input queries may suffer from this truncation, as a significant portion of the query might be completely cut off, potentially leading to a big loss of semantic information.

Query reduction operates on the premise that most queries contain redundant terms that can be removed, without losing its overall meaning. Take as example the following query: "I read that ions can't have dipole moments why not". This query can effectively be reduced to: "Why can't ions have net dipole moments". This reduction eliminates three terms, while preserving the semantic content of the original query.

This research will explore the utility of general use LLM's for query reduction. Meta-Llama3-8b-Intruct [1] is going to be used to perform all reductions in the datasets. This model was selected as it is one of the most recently released cutting-edge open-source models, at the time of this research. Additionally, for the purposes of this task, the 8 billion parameter model strikes an optimal balance between performance and quality. The generation process employs hyperparameters set at 0.6 for temperature and 0.8 for top p. All configurations and system prompts used in this research can be found in our repository.

4 Experimental Setup

This section describes the experimental setup, covering the models, datasets and evaluation metrics used. All experiments were run on an Intel Xeon E5-6448Y CPU and NVidia Tesla A100 GPU's. Additionally, the implementation was done with Pyterrier[2] version 0.10.1 and the Fast-Forward Index[3] framework version 0.2.0.

4.1 Models

The following retriever models were used:

1. **Sparse retriever:** BM25 [13] is an Bag-of-Words model that ranks documents based on the amount of overlapping terms they have with the query. It's used in the first stage of retrieval, providing the top-1000 most relevant documents. Additionally, it serves as a baseline when used standalone.
2. **Dense re-rankers:** In the second stage, dense models are employed to re-rank the documents obtained from BM25. Three state-of-the-art dense models are utilized: **snowflake-artic-embed-m** [19], **bge-base-en-v1.5** [20], **gte-base-en-v1.5** [21]. All of them have an embedding dimension of 768 tokens. Large document are split into passages before indexing (maxP).

All models are used in a zero-shot fashion, that is, with no additional fine-tuning on the used datasets.

4.2 Datasets

All experiments will be conducted on datasets sourced from the BEIR benchmark [22]. This benchmark offers a varied collection of information retrieval datasets, covering a wide range of retrieval tasks. Notably, these datasets are tailored for zero-shot retrieval, meaning that retrieval is performed without requiring additional fine-tuning or training on the data. Table 1 provides an overview of the specific subset of BEIR employed in this study, along with the key characteristics of the datasets.

If the dataset contains an official development set (HotpotQA and SciFact), these will be utilised to find the optimal hyper-parameter values for α in the final interpolation scores.

For datasets lacking development sets, one possible solution involves taking a small subset of the original dataset and use it for development. However, this approach has the limitation of making the results obtained not comparable to previous work, because the chosen subset for the hyper-parameter tuning will be subtracted from the test set, possibly causing variations in the results. To counteract this issue, the experiments will instead be conducted with multiple α values to ensure robustness and comparability of results.

4.3 Evaluation metrics

Effectiveness is measured using the established metrics nDCG@10, MRR@10, and MAP@1000, with a primary focus on nDCG@10 for assessing ranking quality.

$CG@k$ is a simple metric which sums up the relevance scores for the top-K items. It's defined as $CG@k = \sum_{i=1}^{k} rel_i$, where rel_i corresponds to the relevance score. For instance, for a particular query, a document might have a relevance score of 1 if it is relevant to the query and 0 otherwise.

A shortcoming of $CG@k$ is that it doesn't consider the position of the items. Optimally, documents with higher relevance scores should appear at higher ranks. Thus, a way to penalise scores the lower they rank is needed. $DCG@k$ introduces a log-based penalty function to reduce the relevance score at each position. It's mathematically defined as $DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i}-1}{log_2(i+i)}$

Finally, $nDCG@k$ normalizes the $DCG@k$ values by using the ideal order of relevant documents. Defined as $nDCG@k = \frac{DCG@k}{IDCG@k}$, with $IDCG@k$ representing the $DCG@k$ score for the ideal order of items. Figure 2 illustrates an example of how $nDCG@k$ is calculated.

5 Experimental Results

This section presents the outcomes of the conducted experiments.

RQ1: How does query length and complexity affect the re-ranking performance of different encoders on Fast-Forward indexes?

Table 2 shows the retrieval results of different datasets using artic-embed-m as the dense model of the Fast Forward in-

[1] Available at: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md

[2] https://pyterrier.readthedocs.io/en/latest/

[3] https://github.com/mrjleo/fast-forward-indexes/

Dataset	Task	Avg. word length		Example query
		Query	Document	
TREC-COVID	Biomedical IR	10.60	160.77	what are the transmission routes of coronavirus
SciFact	Fact checking	12.37	213.63	crosstalk between dendritic cells dcs and innate lymphoid cells ilcs is important in the regulation of intestinal homeostasis
HotpotQA	Question Answering	17.61	46.30	when was the american lawyer lobbyist and political consultant who was a senior member of the presidential campaign of donald trump born
Arguana	Argument retrieval	192.9	166.80	people will die if we dont do animal testing every year 23 new drugs are introduced in the uk alone 13 almost all will be tested on animals a new drug will be used for a long time think of all the people saved by the use of penicillin if drugs cost more to test that means drug companies will develop less this means more people suffering and dying

Table 1: Overview and statistics of the utilised datasets.

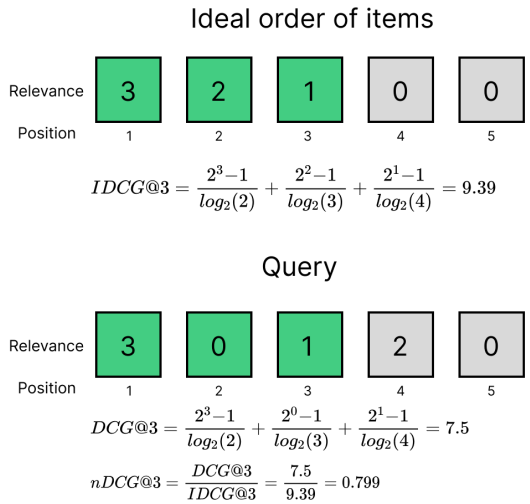

Figure 2: Example of nDCG@3 calculation

dex. Both the sparse and dense scores were normalized prior to interpolation.

As expected, retrieval effectiveness decreases as the average query length of the dataset increases. A downward trend is evident in all three metrics relative to the average query length. Additionally, there is a notable performance drop in the Arguana dataset, indicating that the extreme length of the queries in this dataset, often exceeding the models' context window, adversely affects ranking quality. This is particularly evident as Arguana is the only dataset where the fast-forward index framework shows minimal improvement over the standalone sparse BM25 model.

However, the metrics also contain some outliers, such as in TREC-COVID. The high nDCG@10 and RR@10 suggest that the top 10 results returned by the retrieval system are highly relevant and well-ordered, however the low MAP@1000 (which measures the number of relevant items in the top-1000 results and how well they are ranked) indicates that as you move further down the ranked list, the proportion of relevant documents decreases significantly. Since MAP is also exceedingly low when retrieving with only the sparse model, it shows that a large quantity of relevant results are being pruned out in the first stage, preventing the more powerful semantic models from re-ranking them effectively.

A possible explanation for this might be that the ambiguous nature of short queries makes them difficult to rank, as TREC-COVID is the dataset that on average contains the shortest queries. Not only do they contain less tokens, making the sparse term-matching more susceptible to the vocabulary mismatch problem, but they also lack context or specificity, making it difficult to return relevant results. For example, one of the queries with the lowest precision value is "what is the origin of covid 19". In the experiment, the retrieval system ranks a document titled 'Tracking the origin of early COVID-19 cases in Canada' as the 2nd most relevant and 'Scientists strongly condemn rumors and conspiracy theories about origin of coronavirus outbreak' as the 11th most relevant. Both documents have high term-frequency with the query but aren't contextually relevant, as the truly relevant documents in the dataset relate to the genetic source of the virus and its first transmission into humans. A longer query that explicitly specifies this context would likely yield better precision.

Figures 3-6 show the relation between individual query length and performance. For each individual query nDCG@10 is computed. The queries are grouped by length, so that each box approximately contains the same number of samples. Artic-embed-m is used as the dense model with $k_s = 1000$ and $\alpha = 0.1$.

These results show that the hypothesis that longer queries are consistently more difficult than shorter ones does not hold true across all datasets. In datasets such as HotpotQA and TREC-COVID, shorter queries seem to be harder. Although the median difficulty for each group does not shift signifi-

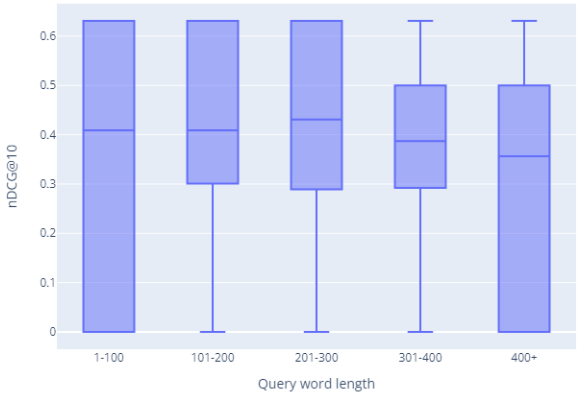

Figure 3: Query word length vs. nDCG@10 on Arguana

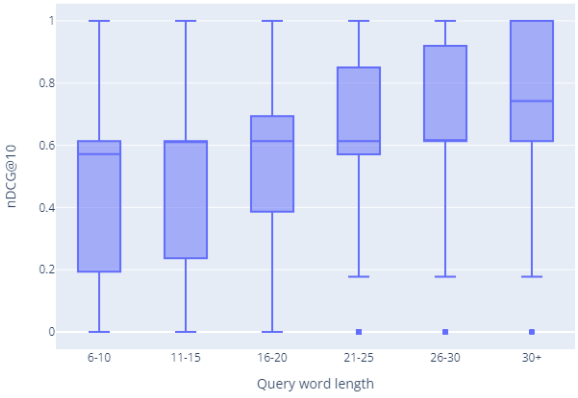

Figure 4: Query word length vs. nDCG@10 on HotpotQA

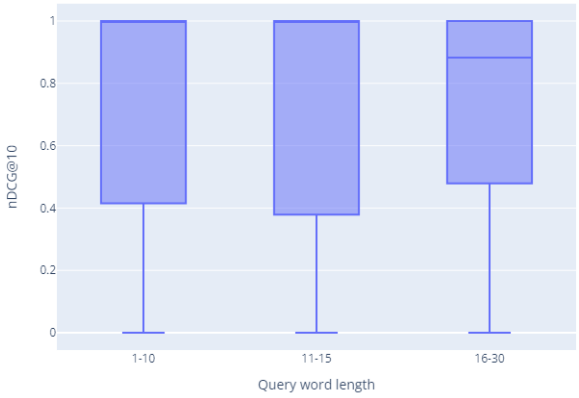

Figure 5: Query word length vs. nDCG@10 on SciFact

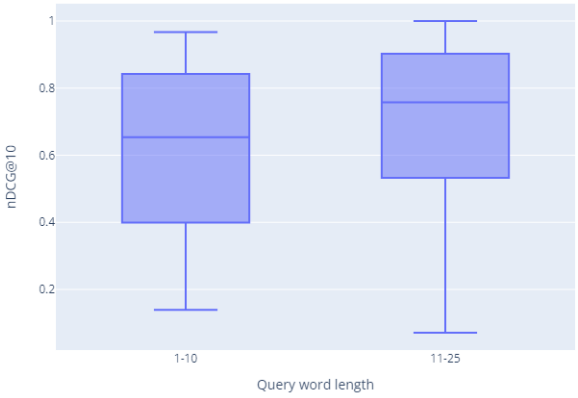

Figure 6: Query word length vs. nDCG@10 on TREC-COVID

cantly, the interquartile range increases gradually with longer queries. For Arguana, the most challenging queries are found at both extremes - the longest and shortest queries - since in the other groups only a small number of outliers shows a nDCG@10 lower than 0.3. This further suggests that the ambiguity inherent in short queries and the complexity of longer ones both add to the challenges of the retrieval process.

RQ2: What strategies can be employed to improve the effectiveness for long and complex queries on Fast Forward Indexes?

Multiple semantic scoring models

In order to reduce the number of hyper-parameters needed when interpolating with multiple semantic scoring models, a single α value is assigned to each model for determining the final ranking score. For instance, in the case of two re-rankers, the final score is computed as: $\phi_f(q,d) = \alpha_S \cdot \phi_S(q,d) + \alpha_{D1} \cdot \phi_{D1}(q,d) + \alpha_{D2} \cdot \phi_{D2}(q,d)$, where $\phi_{D1}(q,d)$ and $\phi_{D2}(q,d)$ represent the dense scores of their respective dense models and $\alpha_S + \alpha_{D1} + \alpha_{D2} = 1$. Additionally, all dense and sparse scores are normalized before interpolating.

Table 3 shows the retrieval results of re-ranking with one, two and three different models in the SciFact and HotpotQA datasets. These datasets were selected because their development sets facilitated the tuning of the multiple hyperparameters needed.

The results demonstrate that using three semantic re-rankers and in some cases two, indeed outperforms the traditional approach of only using one. Although utilising three models constantly outperforms their individual counterparts, they do not provide any significant improvement in effectiveness compared to using two models. Therefore, these findings suggest that using three models is not justified, as it is more costly and yields similar results to using only two.

Hyperparameter tuning revealed that assigning equal weights to each dense model does not yield the best performance. Instead, assigning different weights to each model in the final scoring is optimal. Additionally, optimal results can only be obtained by giving a greater weight to the best

Dataset	Avg. query word length	BM25			Fast Forward: BM25 >> artic-m		
		RR@10	nDCG@10	MAP@1000	RR@10	nDCG@10	MAP@1000
TREC-COVID	10.60	0.8172	0.5761	0.1835	0.9600	0.8093	0.2569
SciFact	12.37	0.6440	0.6839	0.6378	0.7070	0.7427	0.7030
HotpotQA	17.61	0.6624	0.5128	0.4344	0.8693	0.7181	0.6402
Arguana	192.68	0.2408	0.3662	0.2520	0.2511	0.3792	0.2626

Table 2: Comparison of retrieval performances for datasets of different query lengths. BM25 retrieval depth is set at $k_S = 1000$ and $\alpha = 0.1$.

performing models, as seen with BGE and GTE on SciFact and Artic in HotpotQA. This approach likely stems from the fact that different dense models capture the semantic properties of different queries and documents with varying effectiveness for a particular dataset. By giving more weight to the better-performing models, the majority of the final score is influenced by these captured properties. Assigning smaller weights to other models allows harder queries to be slightly adjusted in the rankings, leading to improved overall rankings.

This might also explain why not all combinations of two re-rankers outperform the individual models. In these cases, the less effective re-ranker might introduce misalignments that degrade the performance, preventing the combination from surpassing the best single model.

Finally, it is evident that different datasets benefit differently from the inclusion of the sparse scores. For SciFact, optimal results are achieved with much smaller sparse weights compared to HotpotQA. However, in every case, including the sparse score outperformed not including it and interpolating only the semantic scores.

Query simplification

The datasets where reduction is employed are Arguana and TREC-COVID. Arguana contains queries that exceed 400 words, surpassing the 512-token context window of the model employed. Query reduction is explored as a method to avoid truncation for the longer queries in this dataset.

Each query of the TREC-COVID dataset is composed of three fields [9]. The "query" field provides a condensed version of the question, containing only the main keywords. The "question" field contains a precise natural language question, it is a superset of the information contained in the 'query' field. The "narrative" field offers a longer description that elaborates on the question, however it is not a superset of any other variant, it purely serves to help specify the user's intent. We will refer to the queries provided in the "query" variant as keyword queries moving forward. These keyword queries will be used as a baseline for comparing the reduced version of the original and narrative queries, in order to evaluate the quality of the reductions. Table 5 in the appendix illustrates the main differences between the three variants in this dataset.

The results of reducing the queries using Meta-Llama3-8b-Instruct are presented in table 4. The results on TREC-COVID show that the LLM is successful in obtaining the main semantic information of the query and removing redundant terms, as their ranking score is very comparable to their original versions, while on average removing three terms from the query. Given that the average query length in this dataset is relatively short, not outperforming the unreduced version is expected since no semantic information is added, making it difficult for documents that were previously not ranked correctly to be elevated in rank. As an example of this phenomenon, the query "how has the covid 19 pandemic impacted mental health" was reduced to "impact of covid 19 on mental health".

The narrative queries also displayed poor performance when compared to the standard queries. However, their shortened versions proved to be more effective. The narrative queries are longer and contain an abundant amount of contextually irrelevant terms. Even though no additional semantic information is added, removing these irrelevant terms actually improves retrieval by clarifying the intent of the query. Finally, the poor ranking quality of the keyword queries further demonstrates that the ambiguous nature of shorter queries negatively affects retrieval.

Surprisingly, query reduction did not improve effectiveness on Arguana. This unexpected result may be due to the unique nature of argument retrieval in this dataset, which involves finding the best counterargument to a given argument. This task requires a high level of semantic detail to accurately match the nuanced arguments within the documents. Given the abundance of similar arguments in the corpus, retrieving the best counterargument is particularly challenging. Despite avoiding truncation of the longer queries, the detailed information necessary for this task may be lost during query reduction, as on average the queries were reduced to more than half of their original size by the LLM.

Figure 7 shows the performance comparison between the original and reduced queries on Arguana, demonstrating that the reduction had limited success in improving ranking quality for queries that surpass the dense model's context window. For queries between 300 and 400 words the performance remained nearly identical. For queries exceeding 400 words, while the median performance improved, many high-performing queries experienced a decline, with the upper limit of the interquartile range dropping from approximately 0.6 to 0.5.

6 Responsible Research

Ensuring ethical and reproducible results is one of the highest priorities of this research. This commitment significantly influenced our methodology, dataset selection, and model choices. Thus, this study is compliant with the FAIR principles [23] and Netherlands Code of Conduct for Research

	SciFact					HotpotQA				
	α_S	α_{D1}	α_{D2}	α_{D3}	nDCG@10	α_S	α_{D1}	α_{D2}	α_{D3}	nDCG@10
One re-ranker										
Artic	0.5	0.5	-	-	0.7522	0.3	0.7	-	-	0.7255
BGE	0.3	0.7	-	-	0.7695	0.5	0.5	-	-	0.6957
GTE	0.5	0.5	-	-	0.7694	0.5	0.5	-	-	0.6864
Two re-rankers										
Artic + BGE	0.0025	0.2975	0.7	-	0.7688	0.25	0.5	0.25	-	0.7340
Artic + GTE	0.0025	0.3975	0.6	-	0.7756	0.075	0.725	0.2	-	0.7310
BGE + GTE	0.0025	0.7	0.2975	-	0.7790	0.25	0.5	0.25	-	0.7122
Three re-rankers										
Artic + BGE + GTE	0.0025	0.1975	0.5	0.3	0.7765	0.05	0.55	0.3	0.1	0.7305

Table 3: Performance comparison (nDCG@10) on the SciFact and HotpotQA datasets with varying numbers of re-rankers. BM25 retrieval depth is set at $k_S = 1000$. Order of the models is equivalent to the order of the alpha values. Highlighted results correspond to significant performance improvements.

Dataset	Avg. word length	BM25 >> Artic-m		
		$\alpha = 0$	$\alpha = 0.1$	$\alpha = 0.3$
TREC-COVID				
Original	10.60	0.7984	0.8092	0.8074
Original Reduced	7.48	0.7901	0.8006	0.7985
Keyword	3.48	0.6577	0.6656	0.6688
Narrative	24.96	0.6576	0.6672	0.6686
Narrative Reduced	10.26	0.6910	0.6988	0.6877
Arguana				
Original	192.9	0.3764	0.3792	0.3869
Reduced	65.38	0.3652	0.3691	0.3751

Table 4: Ranking results (nDCG@10) of TREC-COVID and Arguana for different query sets. BM25 retrieval depth is set at $k_S = 1000$.

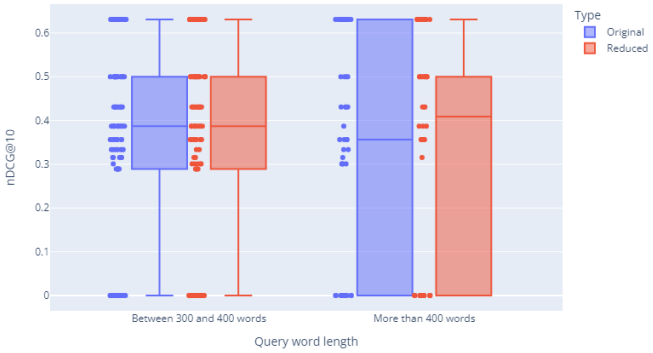

Figure 7: Performance comparison (nDCG@10) between the original and reduced queries in Arguana.

Integrity (2018).

The data used is findable, accessible and interoperable, as we used datasets exclusively from the BEIR collection, a widely recognized benchmark in the information retrieval field. Furthermore, we selected only publicly available datasets[4].

Additionally, both the retrieval and large language models used in our study are open-source and are publicly available on Hugging Face. All the model configurations used are stated in the methodology, experimental setup and appendix. Additionally, implementation was done with open-source libraries, more specifically, Pyterrier[5] version 0.10.1 and the Fast-Forward Index[6] framework version 0.2.0.

To ensure transparency and reproducibility, we have made the source code of our experiments fully open source[7]. Finally, all the results obtained from these experiments, including those that did not successfully improve long query effectiveness, are included in this paper and the repository. As the code is publicly accessible it can be easily reused and it's validity can be verified. All the experiments can be freely reproduced, allowing for any possible errors or inaccuracies to be found.

The dense indexing of the datasets utilised require powerful GPUs, CPUs, and significant amounts of RAM. These are costly resources which may challenge the replicability of the experiments. To address this, we will include all experiment

[4]Available at: https://ir-datasets.com/beir.html

[5]https://pyterrier.readthedocs.io/en/latest/

[6]https://github.com/mrjleo/fast-forward-indexes/

[7]https://github.com/Erhan1706/fast-forward-long-query-effectiveness

results in the repository stored in the TREC format. This will facilitate the replication of the experiments and recalculation of metrics, ensuring the validity of the results.

7 Conclusions and Future Work

Ad-hoc retrieval systems tend to struggle as the length and complexity of the queries utilised increases. This study set out to seek methods that would improve retrieval effectiveness for long and complex queries utilising the Fast-Forward index framework.

Two approaches were explored to improve effectiveness for long and complex queries. The first approach involves using multiple dense models to re-rank an initial set of candidates retrieved by a sparse model. This method proved to be effective in improving ranking quality. Notably, employing two dense models during the re-ranking stage achieved the optimal balance between performance and ranking effectiveness.

The second method explored the utility of LLMs for query reduction. This intended to address the issue of queries being truncated due to their length surpassing the context size of the employed dense models. However, this approach failed to improve performance for the tested datasets. Testing this query reduction technique on additional datasets could be beneficial, leading to more robust results. The effectiveness of LLM generated reductions is influenced by the input and system prompts. While we tested multiple configurations to optimize the results, better configurations may still exist.

Future research could delve deeper into query reductions, potentially applying them selectively based on specific criteria rather than across the entire dataset. Alternatively, given that we show that retrieval systems also struggle with very short queries due to their ambiguity, the opposite path can be taken and query expansion can be researched in order to enhance the performance of these queries. Further studies might also explore modifications to the fast forward framework pipeline, such as employing multi-vector representations for queries and documents instead of single vector approaches.

References

[1] M. Bendersky and W. B. Croft, "Discovering key concepts in verbose queries," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, S. Myaeng, D. W. Oard, F. Sebastiani, T. Chua, and M. Leong, Eds. ACM, 2008, pp. 491–498. [Online]. Available: https://doi.org/10.1145/1390334.1390419

[2] N. Balasubramanian, G. Kumaran, and V. R. Carvalho, "Exploring reductions for long web queries," in *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, F. Crestani, S. Marchand-Maillet, H. Chen, E. N. Efthimiadis, and J. Savoy, Eds. ACM, 2010, pp. 571–578. [Online]. Available: https://doi.org/10.1145/1835449.1835545

[3] P. Yang and H. Fang, "Can short queries be even shorter?" in *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2017, Amsterdam, The Netherlands, October 1-4, 2017*, J. Kamps, E. Kanoulas, M. de Rijke, H. Fang, and E. Yilmaz, Eds. ACM, 2017, pp. 43–50. [Online]. Available: https://doi.org/10.1145/3121050.3121056

[4] P. Karisani, M. Rahgozar, and F. Oroumchian, "A query term re-weighting approach using document similarity," *Inf. Process. Manag.*, vol. 52, no. 3, pp. 478–489, 2016. [Online]. Available: https://doi.org/10.1016/j.ipm.2015.09.002

[5] S. J. Huston and W. B. Croft, "Evaluating verbose query processing techniques," in *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, F. Crestani, S. Marchand-Maillet, H. Chen, E. N. Efthimiadis, and J. Savoy, Eds. ACM, 2010, pp. 291–298. [Online]. Available: https://doi.org/10.1145/1835449.1835499

[6] Y. Chen and Y. Zhang, "A query substitution-search result refinement approach for long query web searches," in *2009 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2009, Milan, Italy, 15-18 September 2009, Main Conference Proceedings*. IEEE Computer Society, 2009, pp. 245–251. [Online]. Available: https://doi.org/10.1109/WI-IAT.2009.42

[7] J. Leonhardt, H. Müller, K. Rudra, M. Khosla, A. Anand, and A. Anand, "Efficient neural ranking using forward indexes and lightweight encoders," *ACM Trans. Inf. Syst.*, vol. 42, no. 5, pp. 117:1–117:34, 2024. [Online]. Available: https://doi.org/10.1145/3631939

[8] J. Leonhardt, K. Rudra, M. Khosla, A. Anand, and A. Anand, "Efficient neural ranking using forward indexes," in *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, F. Laforest, R. Troncy, E. Simperl, D. Agarwal, A. Gionis, I. Herman, and L. Médini, Eds. ACM, 2022, pp. 266–276. [Online]. Available: https://doi.org/10.1145/3485447.3511955

[9] E. M. Voorhees, T. Alam, S. Bedrick, D. Demner-Fushman, W. R. Hersh, K. Lo, K. Roberts, I. Soboroff, and L. L. Wang, "TREC-COVID: constructing a pandemic information retrieval test collection," *SIGIR Forum*, vol. 54, no. 1, pp. 1:1–1:12, 2020. [Online]. Available: https://doi.org/10.1145/3451964.3451965

[10] D. Wadden, S. Lin, K. Lo, L. L. Wang, M. van Zuylen, A. Cohan, and H. Hajishirzi, "Fact or fiction: Verifying scientific claims," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Association for Computational Linguistics, 2020, pp. 7534–7550. [Online]. Available: https://doi.org/10.18653/v1/2020.emnlp-main.609

[11] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds. Association for Computational Linguistics, 2018, pp. 2369–2380. [Online]. Available: https://doi.org/10.18653/v1/d18-1259

[12] H. Wachsmuth, S. Syed, and B. Stein, "Retrieval of the best counterargument without prior topic knowledge," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, I. Gurevych and Y. Miyao, Eds. Association for Computational Linguistics, 2018, pp. 241–251. [Online]. Available: https://aclanthology.org/P18-1023/

[13] S. E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," in *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, ser. NIST Special Publication, D. K. Harman, Ed., vol. 500-225. National Institute of Standards and Technology (NIST), 1994, pp. 109–126. [Online]. Available: http://trec.nist.gov/pubs/trec3/papers/city.ps.gz

[14] T. Formal, B. Piwowarski, and S. Clinchant, "SPLADE: sparse lexical and expansion model for first stage ranking," in *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, and T. Sakai, Eds. ACM, 2021, pp. 2288–2292. [Online]. Available: https://doi.org/10.1145/3404835.3463098

[15] V. Karpukhin, B. Oguz, S. Min, P. S. H. Lewis, L. Wu, S. Edunov, D. Chen, and W. Yih, "Dense passage retrieval for open-domain question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Association for Computational Linguistics, 2020, pp. 6769–6781. [Online]. Available: https://doi.org/10.18653/v1/2020.emnlp-main.550

[16] K. Lee, M. Chang, and K. Toutanova, "Latent retrieval for weakly supervised open domain question answering," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds. Association for Computational Linguistics, 2019, pp. 6086–6096. [Online]. Available: https://doi.org/10.18653/v1/p19-1612

[17] R. F. Nogueira and K. Cho, "Passage re-ranking with BERT," *CoRR*, vol. abs/1901.04085, 2019. [Online]. Available: http://arxiv.org/abs/1901.04085

[18] S. Wang, S. Zhuang, and G. Zuccon, "Bert-based dense retrievers require interpolation with BM25 for effective passage retrieval," in *ICTIR '21: The 2021 ACM SIGIR International Conference on the Theory of Information Retrieval, Virtual Event, Canada, July 11, 2021*, F. Hasibi, Y. Fang, and A. Aizawa, Eds. ACM, 2021, pp. 317–324. [Online]. Available: https://doi.org/10.1145/3471158.3472233

[19] L. Merrick, D. Xu, G. Nuti, and D. Campos, "Arctic-embed: Scalable, efficient, and accurate text embedding models," *CoRR*, vol. abs/2405.05374, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2405.05374

[20] S. Xiao, Z. Liu, P. Zhang, and N. Muennighof, "C-pack: Packaged resources to advance general chinese embedding," *CoRR*, vol. abs/2309.07597, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.07597

[21] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, and M. Zhang, "Towards general text embeddings with multi-stage contrastive learning," *CoRR*, vol. abs/2308.03281, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2308.03281

[22] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, "BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models," *CoRR*, vol. abs/2104.08663, 2021. [Online]. Available: https://arxiv.org/abs/2104.08663

[23] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, "The fair guiding principles for scientific data management and stewardship," *Scientific data*, vol. 3, no. 1, pp. 1–9, 2016.

Appendix

Original queries	Narrative queries	Keyword queries
what is the origin of covid 19	seeking range of information about the sars cov 2 virus s origin including its evolution animal source and first transmission into humans	coronavirus origin
how does the coronavirus respond to changes in the weather	seeking range of information about the sars cov 2 virus viability in different weather climate conditions as well as information related to transmission of the virus in different climate conditions	coronavirus response to weather changes
will sars cov2 infected people develop immunity is cross protection possible	seeking studies of immunity developed due to infection with sars cov2 or cross protection gained due to infection with other coronavirus types	coronavirus immunity
what causes death from covid 19	studies looking at mechanisms of death from covid 19	how do people die from the coronavirus
what drugs have been active against sars cov or sars cov 2 in animal studies	papers that describe the results of testing drugs that bind to spike proteins of the virus or any other drugs in any animal models papers about sars cov 2 infection in cell culture assays are also relevant	animal models of covid 19
what types of rapid testing for covid 19 have been developed	looking for studies identifying ways to diagnose covid 19 more rapidly	coronavirus test rapid testing
are there serological tests that detect antibodies to coronavirus	looking for assays that measure immune response to covid 19 that will help determine past infection and subsequent possible immunity	serological tests for coronavirus
how has lack of testing availability led to underreporting of true incidence of covid 19	looking for studies answering questions of impact of lack of complete testing for covid 19 on incidence and prevalence of covid 19	coronavirus under reporting
how has covid 19 affected canada	seeking data related to infections confirm suspected and projected and health outcomes symptoms hospitalization intensive care mortality	coronavirus in canada
has social distancing had an impact on slowing the spread of covid 19	seeking specific information on studies that have measured covid 19 s transmission in one or more social distancing or non social distancing approaches	coronavirus social distancing impact

Table 5: Comparison between the three different variants of the queries in the TREC-COVID dataset.