Cutting Redundant Knowledge for Effective RAG

Mingyu Lee, Swislar Tan, Ervin Teo

Abstract

Retrieval systems often return multiple near-duplicate documents reducing retrieval diversity and efficiency. This project addresses that challenge by clustering near-duplicates and selecting representative documents to keep Retrieval-Augmented Generation (RAG) both efficient and relevant. We evaluate two deduplication methods on their impact on RAG performance. Overall, moderate deduplication effectively reduces redundancy without harming performance, suggesting that RAG systems can safely benefit from cleaner, more diverse corpora.

Introduction

Near-duplicate detection has been extensively explored in the IR community, from plagiarism detection to search result diversification. However, its implications for Retrieval-Augmented Generation (RAG) remain underexplored. In this project, we adapt and apply established near-duplicate clustering methods to RAG pipelines, examining their effectiveness and limitations. Focusing on commonsense reasoning tasks (e.g., RACo), our analysis highlights how classical IR techniques can improve retrieval efficiency and context relevance in RAG, while also revealing trade-offs between diversity and representativeness.

Approach

We replicated the core RAG pipeline from Yu et al.'s RACo framework (EMNLP 2022), which comprises two stages: (1) a Dense Passage Retriever (DPR) for retrieving relevant commonsense knowledge and (2) a Fusion-in-Decoder (FiD) model for answer generation.

While we successfully reproduced the retrieval component - including passage encoding and top-k retrieval - we diverged from the original work in the generation stage. Instead of using FiD, we integrated modern instruction-tuned LLMs such as Llama3.2 1B Instruct. This modification allows us to evaluate retrieval effectiveness within contemporary language model architectures while maintaining fidelity to the retrieval process defined in RACo.

Similarity Analysis of Corpus

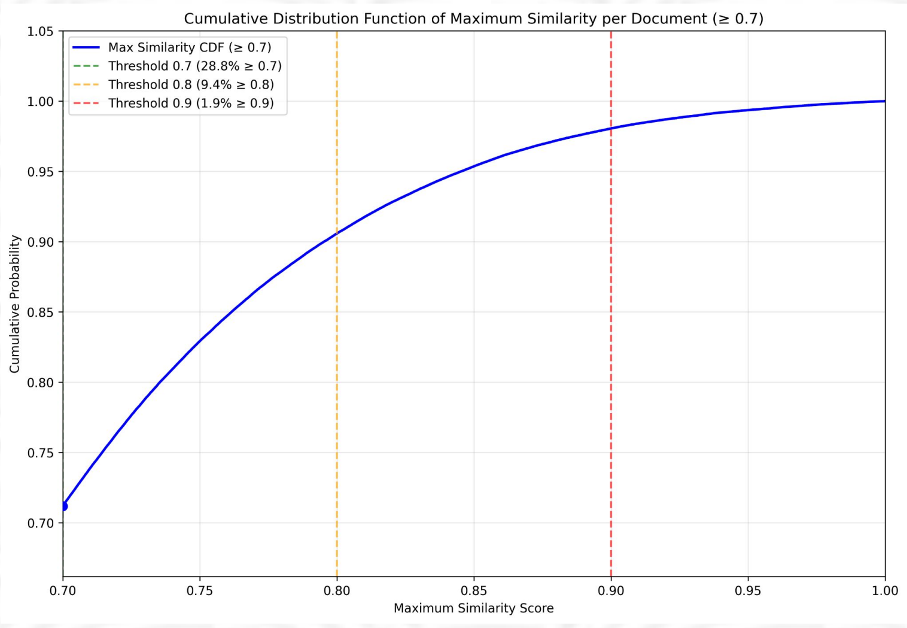

Fig 1. Cumulative distribution of maximum similarity scores across 500k documents using Sentence-BERT

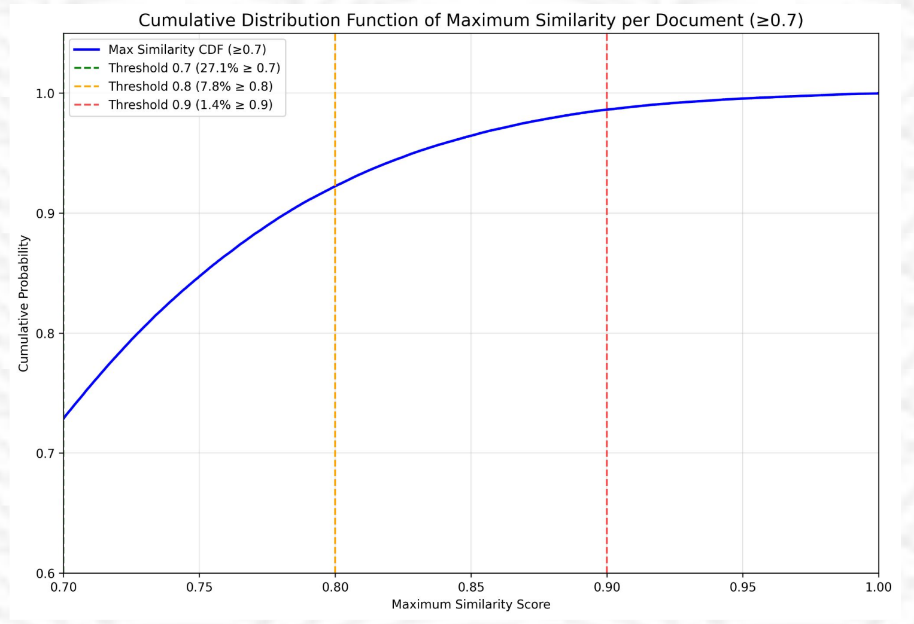

Fig 2. Cumulative distribution of maximum similarity scores across 500k documents using SimCSE

Sentence-BERT

Similarity	Document 1	Document 2
0.97	A dog running in the grass	a dog is running in the grass
0.92	Car is a vehicle	An automobile is a car
0.87	Farm is a place with field where animals and agriculture is done	farm is related to agriculture.

Table 1: Observed results for Sentence-BERT similarity analysis

MinHash

Similarity	Document 1	Document 2
0.90	You are likely to find a dog around in a dog house	You are likely to find a small dog around in a house
0.79	randomised is defined as simple past tense and past participle of randomise	degenderised is defined as simple past tense and past participle of degenderise
0.77	concrews is defined as Third-person singular simple present indicative form of concrew	miscarries is defined as Third-person singular simple present indicative form of miscarry

Table 2: Observed results for MinHash similarity analysis

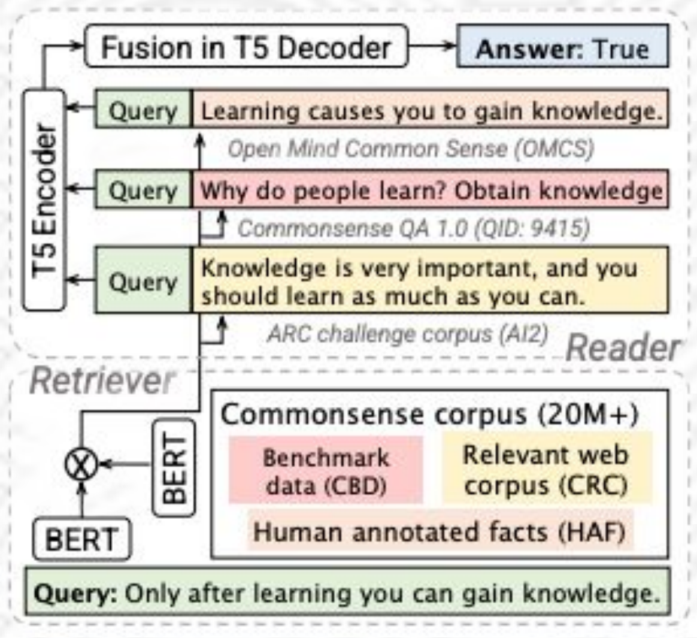
Fusion in T5 Decoder
Answer: True
T5 Encoder
Query
Learning causes you to gain knowledge.
Open Mind Common Sense (OMCS)
Query
Why do people learn? Obtain knowledge
Commonsense QA 1.0 (QID: 9415)
Query
Knowledge is very important, and you should learn as much as you can.
ARC challenge corpus (AI2)
Reader
Retriever
BERT
BERT
Commonsense corpus (20M+)
Benchmark data (CBD)
Relevant web corpus (CRC)
Human annotated facts (HAF)
Query: Only after learning you can gain knowledge.

Figure 3: Architecture of DPR

References

1. [KOM20] Dense Passage Retrieval for Open-Domain Question Answering, EMNLP 2020
2. [YZZ22] Retrieval Augmentation for Commonsense Reasoning: A Unified Approach, EMNLP 2022.

Extension

Building on the replicated retrieval pipeline, we extended the RACo framework by investigating the impact of corpus deduplication on retrieval quality and downstream reasoning. Redundancy in large-scale knowledge corpora may lead to inefficient retrieval and reduced contextual diversity.

We applied three establish near-duplicate detection techniques to a 1M passage subset from the Commonsense Corpus:

- **SimHash**: A locality-sensitive hashing (LSH) method that generates compact binary fingerprints for each passage. Near-duplicates are identified by computing the Hamming distances between fingerprints.
- **MinHash**: Estimates Jaccard similarity between text shingles via probabilistic hashing. MinHash signatures was used with LSH bands to efficiently cluster highly overlapping passages.
- **Sentence-BERT**: A semantic similarity approach to generate dense embeddings for each passage. Using cosine similarity, semantically equivalent passages could be identified despite lexical differences.

Results

We evaluated deduplicate effectiveness on the 1M passage subset using top-k = 4 retrieval, measuring answer accuracy.

1. **SimHash** (64-bit, TF-IDF features):
   - Hamming distance k = 3 removed 29,157 near-duplicates (2.92% of corpus)
   - Hamming distance k = 5 removed 37,384 near-duplicates (3.74% of corpus)

Method/OBQA	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	22.65	**68.54**	**31.66**
RAG (No deduplication)	25.85	67.13	30.06
RAG (SimHash k=3)	**26.45**	67.54	29.46
RAG (SimHash k=5)	26.25	67.54	30.26

Table 3: OBQA Accuracy (%)

Method/CSQA1.0	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	19.90	61.34	33.01
RAG (No deduplication)	20.72	62.82	35.30
RAG (SimHash k=3)	**20.97**	**63.06**	35.79
RAG (SimHash k=5)	20.88	62.74	**36.95**

Table 4: CSQA1.0 Accuracy (%)

2. **MinHash** (128 permutations, word-based tokenization):
   - Similarity threshold 0.8 removed 19,849 near-duplicates (2.06% of corpus)

Method/OBQA	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	22.65	**68.54**	31.66
RAG (No deduplication)	**26.85**	65.93	**34.27**
RAG (MinHash)	**26.85**	65.93	**34.27**

Table 5: OBQA Accuracy (%)

Method/CSQA	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	19.90	61.34	33.01
RAG (No deduplication)	**20.97**	**63.31**	**36.36**
RAG (MinHash)	**20.97**	**63.31**	**36.36**

Table 6: CSQA1.0 Accuracy (%)

3. **Sentence-BERT**:
   - Similarity threshold 0.9 removed 23,755 near-duplicates (2.38% of corpus)

Method/OBQA	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	22.65	68.54	31.66
RAG (No deduplication)	**26.85**	67.13	**30.06**
RAG (Sentence-BERT)	26.45	**67.33**	29.86

Table 7: OBQA Accuracy (%)

Method/CSQA	Llama 3.2 1B Instruct	Qwen2.5 1.5B Instruct	Gemma3 1B Instruct
Baseline	19.90	61.34	33.01
RAG (No deduplication)	20.72	62.34	35.30
RAG (Sentence-BERT)	**20.88**	**62.90**	**35.46**

Table 8: CSQA1.0 Accuracy (%)

Conclusion

Deduplicating the retrieval corpus using SimHash, MinHash or Sentence-BERT removes between 2–4% of near-duplicate documents but results in minimal or slightly positive effects on RAG accuracy.

Overall, these findings suggest that corpus-level deduplication is safe and sometimes beneficial for RAG systems but may offer diminishing returns beyond moderate thresholds. Future work should examine semantic-level redundancy removal and retrieval diversity metrics to better understand the relationship between corpus uniqueness and RAG performance.