From Ambiguity to Accuracy: The Transformative Effect of Coreference Resolution on Retrieval-Augmented Generation systems

Youngjoon Jang[*,1], **Seongtae Hong**[*,1], **Junyoung Son**[1]
Sungjin Park[2], **Chanjun Park**[†,1], **Heuiseok Lim**[†,1]

[1]Korea University
{dew1701, ghdchlwls123, s0ny, bcj1210, limhseok}@korea.ac.kr

[2]Naver Corp
sungjin.park@navercorp.com

Abstract

Retrieval-Augmented Generation (RAG) has emerged as a crucial framework in natural language processing (NLP), improving factual consistency and reducing hallucinations by integrating external document retrieval with large language models (LLMs). However, the effectiveness of RAG is often hindered by coreferential complexity in retrieved documents, introducing ambiguity that disrupts in-context learning. In this study, we systematically investigate how entity coreference affects both document retrieval and generative performance in RAG-based systems, focusing on retrieval relevance, contextual understanding, and overall response quality. We demonstrate that coreference resolution enhances retrieval effectiveness and improves question-answering (QA) performance. Through comparative analysis of different pooling strategies in retrieval tasks, we find that mean pooling demonstrates superior context capturing ability after applying coreference resolution. In QA tasks, we discover that smaller models benefit more from the disambiguation process, likely due to their limited inherent capacity for handling referential ambiguity. With these findings, this study aims to provide a deeper understanding of the challenges posed by coreferential complexity in RAG, providing guidance for improving retrieval and generation in knowledge-intensive AI applications.

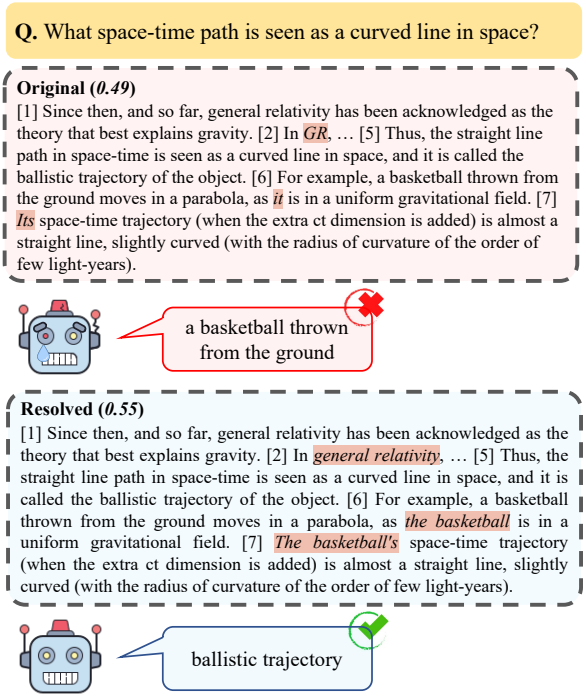

Figure 1: Example of changes in similarity and responses resulting from coreference resolution. Similarity scores are indicated in parentheses using NV-Embed-v2, and responses are generated with the Llama-3.2-1B-Instruct model.

1 Introduction

With the rapid advancement of large language models (LLMs) and information retrieval technologies, Retrieval-Augmented Generation (RAG) has emerged as a fundamental technique widely adopted across various tasks, including knowledge-intensive applications such as question-answering and dialogue systems (Gan et al., 2023; Yang et al., 2023). By integrating retrieval mechanisms with generative language models, RAG enhances factual consistency, improves knowledge recall, and mitigates issues related to hallucination.

Two key challenges in RAG lie in the retrieval of relevant documents from a large corpus and the subsequent in-context learning process, where retrieved documents are leveraged to generate accurate responses. These challenges are particularly pronounced when dealing with documents, as these often contain multiple coreferences to the same entities, making it difficult for language models to resolve coreferential ambiguity effectively (Dasigi et al., 2019). In addition, these hinder the ability

[*]Equal contribution.
[†] Corresponding Author

of LLMs to effectively capture relevant contextual information from the given inputs (Liu et al., 2024).

From this perspective, coreferential complexity can hinder a retrieval model's ability to effectively interpret and represent documents. Specifically, it may prevent the model from accurately capturing the semantic relationships between entities and their references, making it more difficult to align query intentions with the most relevant document. These retrieval errors and drops in relevance propagate throughout the generation process, ultimately reducing the factual accuracy of the responses (Shi et al., 2023). Consequently, such accumulated errors undermine user trust in AI-generated answers, weakening confidence in the system's outputs.

To address these challenges, we aim to systematically investigate the impact of coreferential complexity on each core component of RAG, including document retrieval and in-context learning. Through extensive experiments and analysis, our study reveals two key findings: First, In retrieval tasks, models show performance improvements when coreference resolution is applied, with models utilizing mean pooling demonstrating particularly significant gains. This suggests that resolved coreferences enhance the models' ability to capture document semantics. Second, For QA tasks, we find that smaller language models are likely to benefit more from coreference resolution compared to larger models, indicating that coreferential complexity poses a greater challenge for models with limited capacity. These findings highlight how coreference resolution can enhance different aspects of RAG systems, with specific benefits depending on the model architecture and task type.

2 Coreference Resolution

Coreference resolution is a technique that identifies and links different expressions referring to the same entity in a text by identifying and replacing them with their explicit forms to eliminate ambiguity (Ng, 2010). Figure 1 illustrates how this technique enhances natural language processing tasks through explicit entity references, using an actual example from the SQuAD2.0 dataset. In the document, ambiguous elements such as abbreviations and pronouns ("GR", "it", "Its") are replaced with their explicit forms ("general relativity", "the basketball", "The basketball's"). Comparing the original and resolved documents, the similarity scores computed by the embedding model show an improvement for the resolved version, demonstrating that coreference resolution effectively enhances the precision of similarity computation for retrieval tasks. Beyond retrieval performance, coreference resolution significantly impacts question-answering accuracy by strengthening contextual coherence and logical reasoning. The resolved document provides a more traceable reasoning chain, enabling the model to better understand entity relationships and semantics. As demonstrated in our example, the model provides the correct answer with the resolved document while failing with the original document, showing the benefits of this enhanced clarity. This example clearly illustrates the critical role of coreference resolution in enhancing both document retrieval and question-answering capabilities.

To systematically address coreferential ambiguities, we implement an LLM-powered coreference resolution function f_{coref} that transforms ambiguous coreferences into their explicit antecedents. For each document d_i, this function produces coreferentially explicit document d_i':

$$d_i' = f_{\text{coref}}(d_i)$$

We utilize *gpt-4o-mini* (Hurst et al., 2024) to implement this coreference resolution function. The model takes text containing unresolved coreferences as input and produces an output in which multiple expressions referring to the same entity are explicitly linked, maintaining contextual consistency throughout the text. Through this process, we explore how resolving coreferential ambiguity and providing explicit semantic connections in the document impact retrieval and question answering. The detailed prompt design and implementation specifics are described in Section C.2

3 Experimental Setup

Models We evaluate a variety of publicly accessible embedding models with different architectures and pooling methods to evaluate retrieval performance for both the original document and the coreference-resolved document. For encoder-based embedding models, we use *e5-large-v2* (Wang et al., 2022), *stella_en_400M_v5* (Zhang et al., 2025), *bge-large-en-v1.5* (Xiao et al., 2023), and *gte-modernbert-base* (Zhang et al., 2024). As decoder-based models, we employ *LLM2Vec-Meta-Llama-3-8B-Instruct-mntp-supervised* (BehnamGhader

Architecture	Pool	Models	DocType	BELEBELE			SQuAD2.0			BoolQ			NanoSCIDOCS			AVG			OVR
				@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	
ENCODER	Mean	e5-large-v2	Original	0.922	**0.952**	0.955	0.802	0.881	0.891	**0.839**	**0.905**	**0.913**	0.520	**0.404**	**0.359**	0.809	0.812	**0.810**	0.811
			+ C·R	**0.923**	0.948	**0.956**	**0.816**	**0.893**	**0.902**	0.833	0.902	0.911	**0.520**	0.400	0.352	**0.814**	**0.813**	0.809	**0.812**
		stella_en_400M_v5	Original	0.910	0.946	0.949	0.767	**0.851**	**0.866**	**0.838**	0.907	0.915	0.480	**0.386**	0.345	0.785	0.799	0.803	0.796
			+ C·R	**0.920**	**0.950**	**0.954**	0.767	0.849	0.864	0.837	**0.907**	**0.915**	**0.500**	0.384	**0.349**	**0.790**	**0.799**	**0.804**	**0.798**
	[CLS]	gte-modernbert-base	Original	0.892	0.932	0.937	0.778	0.862	0.876	**0.831**	**0.901**	0.909	0.520	**0.452**	**0.410**	0.793	0.809	**0.811**	0.804
			+ C·R	**0.899**	**0.936**	**0.940**	**0.779**	**0.863**	**0.876**	0.829	0.900	**0.909**	**0.520**	0.448	0.391	**0.794**	**0.809**	0.807	**0.804**
		bge-large-en-v1.5	Original	0.903	0.932	0.939	**0.749**	0.838	**0.854**	0.831	0.899	0.908	0.480	**0.395**	**0.364**	0.776	**0.792**	0.799	0.789
			+ C·R	**0.912**	**0.938**	**0.944**	0.747	**0.838**	0.853	**0.833**	**0.901**	**0.909**	**0.480**	0.382	0.359	**0.777**	0.791	**0.800**	**0.789**
DECODER	Mean	NV-Embed-v2	Original	0.959	0.977	**0.978**	0.865	0.927	0.933	0.874	0.935	0.941	0.460	0.405	**0.356**	0.836	0.842	0.836	0.838
			+ C·R	**0.959**	**0.977**	0.977	**0.873**	**0.933**	**0.938**	**0.874**	**0.935**	**0.941**	**0.480**	**0.414**	0.353	**0.843**	**0.845**	**0.836**	**0.841**
		LLM2Vec	Original	0.938	0.964	0.967	0.835	0.904	0.913	0.854	0.922	0.929	0.440	0.408	0.358	0.814	0.827	0.824	0.822
			+ C·R	**0.941**	**0.965**	**0.968**	**0.839**	**0.907**	**0.916**	**0.854**	**0.922**	**0.929**	**0.500**	**0.424**	**0.372**	**0.826**	**0.831**	**0.827**	**0.828**
	Last	gte-Qwen2-1.5B	Original	0.938	**0.961**	0.963	0.820	0.891	0.901	0.823	0.893	0.904	0.520	0.428	0.387	0.816	0.816	0.812	0.815
			+ C·R	**0.940**	0.959	**0.964**	**0.820**	**0.891**	**0.901**	**0.825**	**0.895**	**0.906**	**0.520**	**0.435**	**0.392**	**0.816**	**0.818**	**0.814**	**0.816**
		Linq-Embed-Mistral	Original	**0.944**	0.967	0.969	**0.800**	**0.885**	**0.895**	0.876	0.937	0.942	0.460	0.407	0.360	0.810	0.828	0.830	0.823
			+ C·R	0.942	**0.967**	**0.969**	0.798	0.882	0.892	**0.877**	**0.937**	**0.942**	**0.500**	**0.423**	**0.373**	**0.815**	**0.830**	**0.832**	**0.826**

Table 1: Performance of retrieval tasks with and without coreference resolution. The @k indicates the top k nDCG results. For each comparison, the higher score is highlighted in **bold**.

et al., 2024) which we refer to as *LLM2Vec*, *NV-Embed-v2* (Lee et al., 2025), *Linq-Embed-Mistral* (Junseong Kim, 2024), and *gte-Qwen2-1.5B-instruct* (Li et al., 2023).

To evaluate how coreference resolution affects LLMs' understanding and answer generation capabilities, we conduct experiments with various instruction-tuned models: *Llama3.2-3B-Instruct*, *Llama3.1-8B-Instruct* (Dubey et al., 2024), *Qwen2.5-3B-Instruct*, *Qwen2.5-7B-Instruct* (Yang et al., 2024), *gemma-2-2b-it*, *gemma-2-9b-it* (Team et al., 2024), *Mistral-7B-Instruct-v0.3* (Jiang et al., 2023).

Datasets To evaluate the effect of coreferential complexity in retrieval performance, we conduct experiments on four datasets: BELEBELE (Bandarkar et al., 2023), which is designed for Machine Reading Comprehension (MRC) tasks, SQuAD2.0 (Rajpurkar et al., 2018), a QA dataset based on Wikipedia, BoolQ (Clark et al., 2019), designed for yes/no questions, and NanoSCIDOCS (Cohan et al., 2020), which is a subset of SCIDOCS dataset, specifically designed for retrieval tasks. For the QA datasets, we adapt the question-document pairs for retrieval evaluation. Details about data preprocessing and extra experiment details can be found in Appendix C.1.

Metrics We use nDCG@k(k=1,3,5) to evaluate retrieval performance. nDCG evaluates retrieval ranking quality by measuring both relevance and position of results with logarithmic position discount. For evaluating QA performance, we calculate the log likelihood on benchmarks such as the BELEBELE and BoolQ datasets for accuracy measurement, and use the F1-score for SQuAD2.0. All experiments are conducted using the library[1] to ensure replicability.

4 Experimental Results and Analysis

4.1 Impact of Coreference Resolution on Retrieval Performance

Table 1 presents a comparison of retrieval performance between original documents and their coreference-resolved versions across different embedding models. Our experiments demonstrate that addressing coreference issues consistently improves retrieval performance across all evaluation metrics, likely due to more explicit and traceable entity references in document representations. The performance improvement is particularly pronounced in decoder-based models, with *LLM2Vec* shows the most significant gains in the average score, improving by 0.012, 0.004, and 0.003 points for nDCG@k (k=1, 3, 5), respectively. These results demonstrate that coreference resolution enhances the overall performance of retrieval tasks, particularly in decoder-based embedding models.

Furthermore, we observe a trend along with the choice of pooling strategies in embedding models. Specifically, models employing mean pooling (e.g., *e5-large-v2*, *stella_en_400M_v5*, *NV-Embed-v2*, and *LLM2Vec*) exhibit a more clear performance gain from coreference resolution compared to models utilizing [CLS] token or last token pooling. This phenomenon can be explained by mean pooling's equal treatment of all tokens. By replacing pronouns with their actual antecedents, more meaningful semantic representations are captured, as each token now carries more explicit semantic

[1] https://github.com/EleutherAI/lm-evaluation-harness

Models	DocType	BoolQ	BELEBELE	SQuAD
Llama3.2-3B -Instruct	Orginal	0.7636	0.8122	0.6437
	+ C·R	**0.7642**	**0.8389**	**0.6888**
Llama-3.1-8B -Instruct	Orginal	0.8202	0.8833	0.5583
	+ C·R	**0.8205**	**0.9133**	**0.7827**
Qwen2.5-3B -Instruct	Orginal	0.7801	0.7800	0.2972
	+ C·R	**0.7804**	**0.8578**	**0.5500**
Qwen2.5-7B -Instruct	Orginal	0.8599	0.8622	0.3980
	+ C·R	0.8599	**0.9022**	**0.7977**
gemma-2 -2b-it	Orginal	0.8006	0.2633	0.5185
	+ C·R	**0.8015**	**0.3067**	**0.6209**
gemma-2 -9b-it	Orginal	0.8645	0.5411	0.7646
	+ C·R	**0.8651**	**0.5467**	**0.8423**
Mistral-7B -Instruct-v0.3	Orginal	0.8321	0.8500	0.4080
	+ C·R	**0.8349**	**0.8511**	**0.4396**

Table 2: Performance of QA tasks on coreference resolution. The higher score is highlighted in bold.

information rather than abstract references. This observation aligns with previous research suggesting that mean pooling is particularly useful for capturing the overall semantics of text data (Zhao et al., 2022). While [CLS] token and last token pooling methods also show improvements with coreference resolution, their reliance on a single-token representation for the entire document embedding leads to relatively smaller gains compared to mean pooling. As shown in Table 9, coreference resolution tends to increase document length by replacing pronouns with their antecedents. This characteristic further amplifies the advantage of mean pooling, which can more effectively integrate information across varying text lengths. These findings highlight the synergistic relationship between mean pooling and coreference resolution in enhancing document representation.

4.2 Impact of Coreference Resolution on Question Answering Performance

Table 2 examines the impact of coreference resolution on QA tasks across different model architectures and sizes. We observe consistent performance improvements across all models and tasks, aligning with previous findings on the benefits of coreference resolution in question answering (Liu et al., 2024).

Notably, smaller models tend to achieve greater performance gains through coreference resolution compared to their larger variants. For instance, in BoolQ, *Qwen2.5-3B-Instruct* shows an improvement of 0.0003 compared to no improvement in the 7B version, and *gemma-2-2b-it* improves by 0.0009 whereas the *9b* model shows an improvement of 0.0006. This pattern becomes more pronounced in the Belebele task, where *Qwen2.5-3B-Instruct* demonstrates an improvement of 0.0778, substantially higher than the 0.0400 gain of its *7B* variant, and *gemma-2-2b-it* achieves a 0.0434 improvement compared to the minimal 0.0056 gain in the *9b* version. As Table 9 shows, applying coreference resolution reduces the number of pronouns, thereby decreasing coreferential complexity. This more explicit representation facilitates easier contextual understanding, particularly benefiting smaller language models.

Interestingly, we find that in SQuAD2.0, some small models with given coreference-resolved document perform comparably to or even surpass larger models using original document. For example, *gemma-2-2b-it* and *Qwen2.5-3B-Instruct* achieve F1-scores of 0.6209 and 0.5500 respectively with coreference-resolved document, which are similar to or higher than the baseline performance of larger models such as *Llama3.1-8B-Instruct*, *Qwen2.5-7B-Instruct*, and *Mistral-7B-Instruct-v0.3* (scoring 0.5583, 0.3980, and 0.4080 respectively). These findings collectively suggest that coreference resolution is impactful for QA tasks, where reducing coreferential complexity directly aids models by facilitating improved contextual understanding.

5 Conclusion

This study investigates the effectiveness of coreference resolution in enhancing natural language understanding across retrieval and question answering tasks. Our comprehensive analysis reveals several key findings. First, dense embedding models show consistent improvements in retrieval performance when coreference resolution is applied, with mean pooling strategies particularly benefiting from more explicit entity representations. Second, the impact of coreference resolution varies across model architectures and sizes: while it enhances performance across all scales, smaller language models show particularly notable improvements, sometimes achieving comparable performance to larger models when given coreference-resolved document. These findings highlight how reducing coreferential complexity can effectively enhance model performance, contributing to our understanding of how to improve contextual comprehension in language models. Our work provides valuable insights for future research in optimizing both retrieval systems and question answering models through better han-

dling of coreferential relationships.

Limitations

Despite the contributions of this study, there are several limitations that should be acknowledged. We identify potential biases arising from the use of GPT-4o-mini for coreference resolution, as the model's interpretations may not always align with human understanding, leading to possible discrepancies. Additionally, despite employing diverse datasets (e.g., BELEBELE, SQuAD2.0, BoolQ, NanoSCIDOCS), our approach may not fully capture the complexities of specialized or highly technical text, indicating the need for broader, domain-specific evaluation. Finally, while providing explicit references can increase clarity by grounding model outputs, this method can sometimes constrain the generative flexibility of language models, thereby limiting their ability to produce a wide range of natural-sounding responses. Balancing clarity with generative versatility thus remains a critical direction for future research.

Ethics Statement

This study acknowledges several ethical considerations. The coreference resolution process may unintentionally perpetuate or amplify existing biases, particularly in sensitive areas such as gender or cultural references, necessitating regular audits of training data. We have documented potential biases and limitations in the use of GPT-4o-mini throughout our research. This paper involved the use of GPT-4o for supporting aspects of the manuscript preparation, such as improving clarity and grammar, while all intellectual contributions, experimental designs, analyses, and core findings remain the responsibility of the authors. Additionally, we acknowledge that the computational cost of coreference resolution raises environmental concerns, and its application in critical decision-making processes requires careful consideration. We maintain transparency in our methodologies to facilitate reproducibility and further research in this area.

Acknowledgments

This work was supported by ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (IITP-2025-RS-2020-II201819) and Institute of Information & communications Technology Planning & Evaluation(IITP) under the Leading Generative AI Human Resources Development(IITP-2025-R2408111) grant funded by the Korea government(MSIT) and Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (RS-2024-00398115, Research on the reliability and coherence of outcomes produced by Generative AI)

References

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2023. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. *arXiv preprint arXiv:2308.16884*.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *Preprint*, arXiv:2404.05961.

Terra Blevins, Hila Gonen, and Luke Zettlemoyer. 2023. Prompting language models for linguistic structure. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6649–6663, Toronto, Canada. Association for Computational Linguistics.

Alfonso Caramazza, Ellen Grober, Catherine Garvey, and Jack Yates. 1977. Comprehension of anaphoric pronouns. *Journal of verbal learning and verbal behavior*, 16(5):601–609.

Haixia Chai, Nafise Sadat Moosavi, Iryna Gurevych, and Michael Strube. 2022. Evaluating coreference resolvers on community-based question answering: From rule-based to state of the art. In *Proceedings of the Fifth Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 61–73, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2024. Dense X retrieval: What retrieval granularity should we use? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15159–15177, Miami, Florida, USA. Association for Computational Linguistics.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. 2020. Specter:

Document-level representation learning using citation-informed transformers. *arXiv preprint arXiv:2004.07180*.

Pradeep Dasigi, Nelson F Liu, Ana Marasović, Noah A Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. *arXiv preprint arXiv:1908.05803*.

Timothy Desmet and Edward Gibson. 2003. Disambiguation preferences and corpus frequencies in noun phrase conjunction. *Journal of Memory and Language*, 49(3):353–374.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Wensheng Gan, Zhenlian Qi, Jiayang Wu, and Jerry Chun-Wei Lin. 2023. Large language models in education: Vision and opportunities. In *2023 IEEE international conference on big data (BigData)*, pages 4776–4785. IEEE.

Yujian Gan, Massimo Poesio, and Juntao Yu. 2024. Assessing the capabilities of large language models in coreference: An evaluation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1645–1665.

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

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Jihoon Kwon Sangmo Gu Yejin Kim Minkyung Cho Jy-yong Sohn Chanyeol Choi Junseong Kim, Seolhwa Lee. 2024. Linq-embed-mistral:elevating text retrieval with improved gpt data through task-specific control and quality refinement. Linq AI Research Blog.

Ben Kantor and Amir Globerson. 2019. Coreference resolution with entity equalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 673–677, Florence, Italy. Association for Computational Linguistics.

Nghia T Le and Alan Ritter. Are language models robust coreference resolvers? In *First Conference on Language Modeling*.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Nv-embed: Improved techniques for training llms as generalist embedding models. *Preprint*, arXiv:2405.17428.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Yanming Liu, Xinyue Peng, Jiannan Cao, Shi Bo, Yanxin Shen, Xuhong Zhang, Sheng Cheng, Xun Wang, Jianwei Yin, and Tianyu Du. 2024. Bridging context gaps: Leveraging coreference resolution for long contextual understanding. *arXiv preprint arXiv:2410.01671*.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Ruslan Mitkov. 1999. *Anaphora resolution: the state of the art*. School of Languages and European Studies, University of Wolverhampton ....

Vincent Ng. 2010. Supervised noun phrase coreference research: The first fifteen years. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1396–1411, Uppsala, Sweden. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

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

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Mingzhu Wu, Nafise Sadat Moosavi, Dan Roth, and Iryna Gurevych. 2021. Coreference reasoning in machine reading comprehension. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5768–5781, Online. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Rui Yang, Ting Fang Tan, Wei Lu, Arun James Thirunavukarasu, Daniel Shu Wei Ting, and Nan Liu. 2023. Large language models in health care: Development, applications, and challenges. *Health Care Science*, 2(4):255–263.

Dun Zhang, Jiacheng Li, Ziyang Zeng, and Fulong Wang. 2025. Jasper and stella: distillation of sota embedding models. *Preprint*, arXiv:2412.19048.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. *Preprint*, arXiv:2407.19669.

Shuai Zhao, Fucheng You, Wen Chang, Tianyu Zhang, and Man Hu. 2022. Augment bert with average pooling layer for chinese summary generation. *Journal of Intelligent & Fuzzy Systems*, 42(3):1859–1868.

A Related Work

A.1 Coreference Resolution

Coreference Resolution plays a crucial role in understanding and representing text. Previous studies have demonstrated that accurately identifying and linking expressions referring to the same entity within a text serves as a fundamental component of natural language understanding (Caramazza et al., 1977; Kantor and Globerson, 2019; Desmet and Gibson, 2003). In particular, coreference resolution is considered one of the complex tasks that requires not only grammatical agreement but also semantic coherence and understanding of discourse structure (Mitkov, 1999).

For Coreference Resolution, Lee et al. (2017) first proposed an end-to-end approach that learns the antecedent distribution of all spans in a document, while Manning et al. (2020) utilized attention mechanisms to analyze how language models perform coreference resolution. Recent research explores the use of prompts with LLMs for coreference resolution, demonstrating that prompt-based methods can effectively leverage the model's inherent linguistic knowledge for this task (Le and Ritter; Blevins et al., 2023; Gan et al., 2024).

A.2 Applications in Downstream Tasks

There have been various attempts to reduce coreferential complexity to downstream tasks. Chen et al. (2024) proposed propositions as self-contained factual units that reduce context dependency caused by coreference in retrieval tasks. Meanwhile, Wu et al. (2021), Chai et al. (2022), Liu et al. (2024) have shown that coreference resolution techniques can improve long context understanding and answering performance in QA tasks.

In our paper, we evaluate the impact of coreference resolution through prompting in LLMs on both retrieval and QA tasks. Our analysis of dense embedding models shows that coreference resolution consistently improves retrieval performance, with models using mean pooling strategies demonstrating particularly notable gains. For QA tasks, experiments across BoolQ, Belebele, and SQuAD2.0 reveal that while coreference resolution generally enhances performance across all model sizes, smaller language models tend to achieve greater relative improvements compared to their larger variants.

B Additional Experiment

Since using GPT-4o-mini is relatively expensive, we perform coreference resolution with a small Language Model, Qwen2.5-7B-Instruct (Yang et al., 2024), and report the retrieval performance of Embedding models and the QA performance of LLMs.

Models	DocType	BoolQ	BELEBELE	SQuAD
Qwen2.5-3B -Instruct	Orginal	0.7801	0.7800	0.2972
	C·R·QWEN	0.7777	0.8489	0.3023
	C·R	**0.7804**	**0.8578**	**0.5500**
gemma-2 -2b-it	Orginal	0.8006	0.2633	0.5185
	C·R·QWEN	0.8003	0.3044	**0.6215**
	C·R	**0.8015**	**0.3067**	0.6209
Mistral-7B -Instruct-v0.3	Orginal	0.8321	0.8500	0.4080
	C·R·QWEN	0.8336	0.8500	0.5742
	C·R	**0.8349**	**0.8511**	**0.7396**

Table 3: Performance of QA tasks on coreference resolution via Qwen2.5-7B-Instruct. The higher score is highlighted in bold.

QA Performance Table 3 shows results for QA tasks on coreference resolution done by Qwen2.5-7B-Instruct. It shows that resolving coreferential complexity by Qwen2.5-7B-Instruct also marginally improves QA performance above all three models.

Retrieval Performance As shown in Table 4, results show that using a lightweight model for coreference resolution also improves retrieval performance. Particularly, models using mean pooling strategy demonstrates superior performance, which aligns the prior results in our paper.

These results show that resolving coreferential complexity with relatively small and cost-effective models can also improve retrieval performance (especially models utilizing mean pooling) and QA performance.

C Detailed Experimental Setup

C.1 Datasets

In processing the data for retrieval tasks, due to the substantial size of SQuAD2.0 and BoolQ datasets, we only use their validation data to construct the retrieval pool, as applying coreference resolution to the entire document set would be computationally intensive. For SQuAD2.0, we exclude all instances where answers are not available.

Among these datasets, BELEBELE, SQuAD2.0, and BoolQ, which contain answer information, are additionally utilized to evaluate the generation capabilities of our model. This allows us to demonstrate comprehensive effectiveness by assessing whether the model can generate improved responses to queries based on the retrieved documents.

C.2 Prompt Templates

This section provides an overview of the prompt templates used in our experiments.

Coreference Resolution Table 5 outlines the prompt applied for coreference resolution. This prompt instructs the model to act as a coreference resolution expert, replacing ambiguous pronouns with their explicit antecedents. The prompt includes examples demonstrating how pronouns should be resolved to their corresponding entities, ensuring consistent and accurate resolution.

QA inference For QA tasks, we utilize different prompts tailored to each dataset's characteristics. Table 7 shows the prompt for BoolQ, which presents the document and question in a straightforward format for yes/no answers. Table 6 presents the prompt for Belebele, structured to handle multiple-choice questions with four options. Table 8 illustrates the prompt for SQuAD2.0, which explicitly instructs the model to provide concise answers to questions based on the given document.

C.3 Hardware

We conducted our experiments using an Intel Xeon Gold 6230R @2.10GHz CPU, 376GB RAM, and an NVIDIA RTX A6000 48GB GPU. The software environment included nvidiadriver, CUDA, and PyTorch, running on Ubuntu 20.04.6 LTS.

Architecture	Pool	Models	DocType	BELEBELE			SQuAD2.0			BoolQ			NanoSCIDOCS			AVG			OVR
				@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	@ 1	@ 3	@ 5	
ENCODER	Mean	stella_en_400M_v5	Original	0.910	0.946	0.949	0.767	0.851	0.866	0.838	0.907	0.915	0.480	0.386	0.345	0.785	0.799	0.803	0.796
			C·R	0.920	**0.950**	**0.954**	0.767	0.849	0.864	0.837	0.907	0.915	0.500	0.384	0.349	0.790	0.799	0.804	0.798
			C·R·Qwen	**0.921**	**0.950**	**0.954**	**0.784**	**0.865**	**0.879**	**0.841**	**0.910**	**0.917**	**0.540**	**0.438**	**0.405**	**0.805**	**0.814**	**0.818**	**0.812**
	[CLS]	bge-large-en-v1.5	Original	0.903	0.932	0.939	**0.749**	**0.838**	**0.854**	0.831	0.899	0.908	**0.480**	**0.395**	**0.364**	0.776	0.792	0.799	**0.789**
			C·R	**0.912**	**0.938**	**0.944**	0.747	**0.838**	0.853	**0.833**	**0.901**	**0.909**	**0.480**	0.382	0.359	**0.777**	0.791	**0.800**	**0.789**
			C·R·Qwen	0.901	0.934	0.940	**0.749**	**0.838**	**0.854**	0.831	0.899	0.906	**0.480**	0.382	0.359	0.775	0.790	0.798	0.788
DECODER	Mean	LLM2Vec	Original	0.938	0.964	0.967	0.835	0.904	0.913	**0.854**	**0.922**	**0.929**	0.440	0.408	0.358	0.814	0.827	0.824	0.822
			C·R	**0.941**	**0.965**	**0.968**	**0.839**	**0.907**	**0.916**	**0.854**	**0.922**	**0.929**	**0.500**	**0.424**	**0.372**	**0.826**	**0.831**	**0.827**	**0.828**
			C·R·Qwen	0.940	0.964	0.967	0.834	0.904	0.912	0.853	0.921	0.928	0.480	0.421	0.366	0.821	0.829	0.825	0.825
	Last	Linq-Embed-Mistral	Original	0.944	0.967	0.969	**0.800**	**0.885**	**0.895**	0.876	**0.937**	**0.942**	0.460	0.407	0.360	0.810	0.828	0.830	0.823
			C·R	0.942	0.967	0.969	0.798	0.882	0.892	**0.877**	**0.937**	**0.942**	**0.500**	**0.423**	**0.373**	0.815	**0.830**	**0.832**	**0.826**
			C·R·Qwen	**0.948**	**0.968**	**0.972**	0.799	**0.885**	**0.895**	0.874	0.936	0.940	**0.500**	**0.423**	**0.373**	**0.817**	**0.830**	**0.832**	**0.826**

Table 4: Performance of retrieval tasks with coreference resolution via Qwen2.5-7B-Instruct. The @k indicates the top k nDCG results. For each comparison, the higher score is highlighted in **bold**.

You are an expert in coreference resolution. Your task is to resolve all ambiguous pronouns and references in the provided document, replacing them with explicit and contextually accurate entities. Do not add any extra text or commentary—output only the fully resolved document.

Below are some examples:

Example 1:
Input:
Document: Alice, who was late, quickly ran to catch the bus because she missed her train.
Output:
Alice, who was late, quickly ran to catch the bus because Alice missed her train.

Example 2:
Input:
Document: Bob said he would finish his work today because he promised his manager.
Output:
Bob said that Bob would finish Bob's work today because Bob promised his manager.

Example 3:
Input:
Document: The committee stated that they would review the proposal after they received feedback.
Output:
The committee stated that the committee would review the proposal after the committee received feedback.

When you receive the input document (which always starts with "Document:"), please output only the resolved document text.

Document: {Document}

Table 5: Prompt template example for CR task.

Please refer to the given passage and choose the correct answer.

P: {Document}
Q: {Question}
A: {mc_answer1}
B: {mc_answer2}
C: {mc_answer3}
D: {mc_answer4}
Answer:

Table 6: Prompt template example for BELEBELE inference.

{Document}

Question: {Question}

Answer:

Table 7: Prompt template example for BoolQ inference.

Instruction
Please answer the question.

Conditions
You must answer the question. with short answer.

Document: {Document}

Question: {Question}

Answer:

Table 8: Prompt template example for SQuAD2.0 inference.

D Coreferential Complexity

Table 9 presents the number of noun and pronoun chunks before and after applying coreference resolution across different datasets. We define referential complexity as the degree of difficulty in understanding a given context, where a higher number of pronouns increases ambiguity in contextual comprehension. The comparison between Table 1 and Table 9 reveals that reduced referential complexity through coreference resolution correlates with improved retrieval performance, particularly in models using mean pooling strategies. When examining Table 2 and Table 9, we observe that this reduction in referential complexity enhances QA performance across all model sizes, with smaller language models showing notable gains. These smaller models particularly benefit from the more explicit representation provided by coreference resolution, as demonstrated by their improved performance in tasks like BoolQ, Belebele, and SQuAD2.0.

	Belebele		Bool Q		SQuAD v2.0		NanoSCIDOCS	
	original	CR	original	CR	original	CR	original	CR
Total words	44,258	46,391	320,991	336,673	176,918	184,348	354,405	362,154
AVG noun chunks	22.05	22.73	26.00	26.70	35.89	36.75	44.83	44.81
AVG pronoun chunks	2.70	1.39	2.36	1.24	2.85	1.86	4.39	2.96

Table 9: Referential complexity computed using noun chunk detection in SpaCy (Honnibal and Montani, 2017). We observe that applying coreference resolution increases the number of noun chunks while reducing the number of pronoun chunks. This implies a reduction in referential ambiguity, thereby simplifying contextual understanding.