Reducing Hallucinations in Enterprise Generative AI with Retrieval-Augmented Generation

Sem de Jong
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
s.dejong-7@student.utwente.nl

***Abstract*—Hallucination – the generation of plausible but incorrect information by large language models (LLMs) – poses a serious risk for enterprises that use generative AI assistants on internal documentation [1]. This thesis evaluates the effectiveness of Retrieval-Augmented Generation (RAG) in reducing hallucinations in an enterprise context. A custom demo system was developed using Convex and Next.js. This system combines two embeddings models (Open AI and Google) and two chunking strategies. The system was tested under five different configurations, including a baseline configuration without retrieval. Both the automated and manual evaluations were used to measure hallucination severity and frequency. The results demonstrate that all RAG configurations significantly reduced hallucinations compared to the baseline configuration, with Google embeddings and smaller chunk sizes performing best. The inter-rater agreement between the annotators was high, and the trends in the automated evaluations closely followed the manual evaluation patterns. These findings support RAG as an effective strategy to increase factual accuracy of generative AI system on internal documentation in enterprise environments.**

***Index Terms*—hallucination, Retrieval-Augmented Generation, RAG, LLM, enterprise AI, feedback loops, Canon Medical**

1. Introduction

Though they sometimes hallucinate, generative AI models like large language models (LLMs) have performed exceptionally well in natural language tasks. They sometimes hallucinate by creating factually false or unsupported plausible-sounding claims. In corporate environments, where staff members and consumers expect consistent and accurate information, this tendency is particularly concerning. In fields like healthcare technology, a wrong response produced by an artificial intelligence assistant could undermine confidence or even create safety concerns [2]. Hallucinations have become a well-known problem for LLM deployments; so much so that public worry over untrustworthy AI results caused ”hallucinate” to be highlighted as a word of the year in 2023 [3].

Retrieval-Augmented Generation (RAG) provides a promising way to address this problem [4], [5]. RAG augments an LLM by retrieving relevant documents (for example, from an internal knowledge base or document repository) and providing them as additional context for the model's generation. RAG can increase factual correctness and lower unsupported assertions by grounding the model's replies in actual reference text. Essentially, RAG is an ”open-book” method: the LLM is provided with extra domain-specific information, rather than depending just on its internal training data. In an enterprise generative AI system, this has two main advantages: (1) the model's responses contain verifiable sources, therefore improving transparency and trust; and (2) the model is less likely to generate a plausible but erroneous response (a hallucination), since it can ground its answer on the retrieved facts. Many organizations have adopted RAG to connect LLMs with up-to-date domain-specific information.

1.1. Project Context

Working with Canon Medical Systems Europe, this project will create an RAG-based question-answering system using internal corporate documents, for example, technical manuals, product FAQs, to respond to user prompts. The goal is to assess whether, compared to a baseline where the LLM answers questions without document retrieval, grounding the LLM on Canon Medical Systems Europe's internal documentation lowers the frequency and severity of hallucinations.

1.2. Company Context

Canon Medical Systems Europe, with its headquarter in Amstelveen, the Netherlands, is a significant player in diagnostic medical imaging. The company's portfolio includes

43rd Twente Student Conference on IT, July 4th, 2025, Enschede, The Netherlands.

AI-driven imaging IT solutions, integrated care pathways, and advanced imaging modalities like computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound. This thesis research was conducted as a proof-of-concept within Canon Medical Systems Europe to apply retrieval-augmented generation (RAG) to the company's internal documentation (e.g., service manuals and technical documents). Canon Medical was particularly interested in this initiative as a way to test whether RAG could reduce hallucinations in the responses generated by a large language model when querying its documentation. The system's intended primary users are field engineers performing maintenance tasks, though other employees could also benefit from its reliable access to technical information.

2. Problem Statement

Large language models can produce confident explanations and answers that sound plausible but are factually incorrect or not supported by any source [2]. This hallucination problem undermines the usefulness of generative AI in enterprises dealing with sensitive or specialized information. In an enterprise like Canon Medical Systems Europe, internal documents contain the ground-truth knowledge that employees and systems rely on. If a generative AI assistant fabricates an answer that contradicts these documents or introduces non-existent details, it could mislead users and damage confidence in the system. The core problem addressed by this project is how to reduce and ideally eliminate such hallucinations when using generative AI to answer questions based on enterprise documentation.

To address this problem, we need to explore the mechanisms that ground LLM responses in verified information [5]. Retrieval-Augmented Generation is one such mechanism: by retrieving relevant document snippets and conditioning the LLM on them, the model is expected to generate answers that are factually consistent with the source material. However, many open questions remain about the optimal way to configure and use RAG in practice to minimize hallucinations.

In concrete terms, the project addresses the following problem: How can we reliably measure and reduce the rate of hallucinated (factually inconsistent or unsupported) answers produced by a generative AI system that answers questions using Canon Medical Systems Europe's internal documentation? This breaks down into sub-problems of selecting the right RAG approach and defining metrics for hallucination. Addressing this problem helps to combine the generative strengths of LLMs with the high accuracy standards expected in enterprise knowledge systems.

2.1. Research Question

We aim to answer one main research question and one sub-research question.

- To what extent can Retrieval-Augmented Generation (RAG) reduce hallucinations in generative AI systems for enterprise documentation?
- How do selected RAG configurations (document chunking strategy and embedding models) affect hallucination rates?

The research aims to offer clear knowledge of how much RAG can reduce the hallucination issue for enterprise-grade generative AI solutions. The research will also provide knowledge on best practices, for example optimal chunk size or different embedding models.

3. Related Work

3.1. RAG for Mitigating Hallucinations

Large Language Models (LLMs) are prone to hallucinations, which is the generation of plausible but incorrect information [5], [6]. This is especially a big concern in domains like healthcare, where inaccuracies can have serious consequences [1]. RAG is seen as a promising solution to this issue by grounding LLM responses in external knowledge [6], [3]. In a RAG pipeline a prompt triggers a search in a domain specific knowledge base. This retrieves relevant chunks that are then provided as context to the LLM [7], [8]. By giving the LLM validated information directly into the prompt, RAG can significantly improve factual accuracy and reduce hallucinations [9], [10]. For example, Ozmen and Mathur (2025) mention that RAG "significantly [enhances] the accuracy, relevance, and transparency". Also Béchard and Ayala (2024) report that RAG is a "well-known method that can reduce hallucination and improve output quality", especially when answering prompts that require some form of external knowledge [3].

3.2. Enterprise Applications of RAG

In enterprise settings RAG has become one of the main strategies to ensure factual correctness in LLM based applications like customer support, documentation Q&A, and workflow automation. Packowski et al. (2024) describe RAG as an effective way to use generative AI for answering users questions about product documentation "while avoiding hallucinations and factual inaccuracy" [7]. Their team at IBM built an enterprise scale chatbot that works with company manuals and knowledge base articles [7]. A key takeaway from their work is that small changes in how the knowledge base content is segmented and prepared can have significant impact on on the factual accuracy [7].

3.3. Domain Specific QA

Enterprise settings could also benefit from hybrid approaches that use RAG with other techniques to boost the factual accuracy. Liu et al. (2024) researched closed domain QA on an internal Science IT corpus by comparing fine tuned LLMs against different RAG configurations [11]. Their results showed that retrieval augmented models align better with the ground truth. For example, GPT-4 with RAG produced responses that semantically matched better than a

fine tuned model alone [11]. The most effective implementation was an Aggregated Knowledge Model (AKM) that clustered and "voted" on responses from different RAG and fine tuned models to find the best response possible [11]. This reduced errors by filtering out hallucinated responses, gaining another 8% performance increase over the best performing system (GPT-4 with RAG).

3.4. Impact of Chunking Strategies

How information is retrieved and presented to the LLM has a huge impact on hallucination rates. One of the main factors is the chunking strategy and context length used in a RAG system. If the retrieved chunks are too large or not relevant enough, the LLM can get confused and use irrelevant details, which increases the chance of producing a hallucinated response. Zhang et al. (2025) highlight the "lost-in-the-middle" problem for long contexts, where important facts in the middle of a long input may be ignored [8]. Their BriefContext solution, which splits documents into smaller thematic chunks, helped the LLM to not ignore important information.

In an enterprise setting, chunking and retrieval configuration is also important. Packowski et al. (2024) note that the most popular approach in literature is to segment content into chunks, embed the chunks, and use vector search for retrieval [7]. However, their team found that one size does not fit all situations. For certain knowledge bases traditional vector retrieval wasn't always optimal [7]. They experimented with different search methods and chunk sizes for their documentation.

Overall, the literature suggests that optimal chunking and retrieval are important factors to a RAG implementations success. Too much context can confuse a LLM, while too little context can miss the information needed for an accurate response. If important information is missing, or a lot of irrelevant text is added to the prompt even a state-of-the-art LLM may hallucinate [12].

3.5. Embedding Model Fine-Tuning and Domain Adaptation

A recurring them with in RAG research is the retrieval module's quality. Several researches show that fine-tuning embedding models on domain specific data can yields substantial accuracy gains. A Databricks study by Drozdov et al. showed that finetuning a general text embedding model on enterprise datasets led to major improvements in RAG accuracy [13]. In their experiments, the customized embeddings significantly outperformed both the original pretrained model and OpenAI's high quality embeddings in most cases[13]. The better retrieval results had a direct impact on the answer generation.

The ability to rely on external knowledge means that you can avoid exhaustive model training on every fact within the domain. Fine-tuning the retrieval mechanism (for example the embeddings) is often way cheaper and more adaptable. A recent survey paper identifies retrieval-augmented training as a key trend, allowing LLMs to generalize better to new information by offloading memory to a database [2].

3.6. Evaluation Methods for Hallucination

Evaluating an LLM's hallucination rate is non-trivial and has been addressed in multiple ways in literature. Traditional QA metrics (like exact match, F1 score, BLEU, ROUGE) are useful when a fixed set of correct answers is available, but they can fail to capture subtle factual inaccuracies and are often not meant for real world queries [7]. For example, Packowski et al. found that standard benchmarks did not reflect how well their enterprise RAG bot handled users prompts [7]. Human evaluation is common, especially in specialized domains. Xu et al. (2025) used expert ratings on clinical questions to compare their RAG system with baselines [12]. Their rubric scored factual accuracy of the responses, and therefor directly quantifying hallucination avoidance.

Another popular technique is leveraging LLM's themselves to judge factual accuracy. Some works have used GPT-4 or similar as a critic model that compares the response to the source documents and flag unsupported statements [14], [7].

4. Methodology

To answer our research questions, we create an evaluation system that has different RAG settings, human judgment and automated scoring using an LLM. Here is a summary of how each part of the methodology relates to the research questions:

RQ1 (RAG vs. Hallucinations): We check how accurate the answers are that we get with retrieval support in all settings. This includes both manual hallucination annotation and automated LLM-based consistency scoring to find out how much RAG grounding makes things more accurate. We can tell how well RAG reduces hallucinations overall by comparing these results to what we got from the LLM without any RAG system in place.

RQ2 (Configurations – Chunking & Embeddings): We set up four RAG configurations (a 2×2 mix of chunking strategy and embedding model) and look at how often they hallucinate. We can see how the size and strategy of document chunks and the choice of embedding model affect the chance of hallucinations while controlling all other variables. This comparison shows which configurations give the most accurate answers and why (for example, better retrieval precision or semantic matching).

Human annotators give a scale-based consistency score to the responses generated and an automated LLM evaluator does this on a larger scale. All of the parts work together to give a full picture of RAG's effect on hallucinations **(RQ1)** and the effect of configuration choices **(RQ2)**.

4.1. RAG Configurations (Document Chunking × Embeddings)

We implement four RAG system variants by varying the document chunking strategy and the embedding model for retrieval. Document chunking strategy defines how enterprise documents are split into chunks before embedding. We compare two approaches, a smaller chunks size (512 tokens) and a larger chunk size (1024 tokens). For each chunking method, we apply a different embedding model to vectorize the chunks and user queries (using two general-purpose embedding providers). This yields a 2×2 design of four RAG configurations. Comparing these configurations allows us to examine how chunk size and embedding choice influence hallucination frequency, directly addressing **RQ2**.

4.2. Manual Hallucination Annotation (Human Evaluation)

We use human evaluation to quantify hallucinations in the answers from each configuration, supporting **RQ1** (overall hallucination extent) and **RQ2** (differences between configurations). A set of 25 representative questions about the enterprise documentation was compiled. Each of the four RAG configurations generates an answer for every question. Two human annotators independently review each answer and determines the amount of hallucination on the following scale:

- **0 – Perfect (No Hallucination)**: The answer is entirely accurate, fully supported by the retrieved documentation, and free of any fabricated content. All claims can be directly traced to the source material.
- **1–2 – Minor Hallucination**: The answer is mostly accurate but includes small additions or inferred details not explicitly found in the documentation. These hallucinations are subtle and do not significantly alter the meaning or correctness of the response.
- **3–4 – Moderate Hallucination**: The answer contains noticeable factual inaccuracies, unsupported assertions, or inconsistencies. While parts of the response may still be grounded in the documentation, the hallucinations materially impact the reliability of the answer.
- **5 – Severe Hallucination**: The answer is entirely fabricated, with little to no grounding in the retrieved documentation. It may include major factual errors, logical contradictions, or fabricated content that misleads or confuses the user.

We calculate inter-annotator agreement (e.g. Cohen's kappa) to ensure consistent labeling between the two evaluators. The results of this manual annotation provide a baseline measure of how often the RAG system hallucinates and what type of hallucinations occur. This directly informs **RQ1** by indicating the extent RAG reduces hallucinations (proportion of answers with no hallucination) and also enables comparison of hallucination rates across the four configurations for **RQ2**.

4.3. Automated LLM-Based Hallucination Evaluation

To complement human annotations, we develop an automated evaluation using a Large Language Model (LLM) as a hallucination assessor. The LLM is given access to a known subset of the enterprise documents (ground-truth reference) and is used to systematically generate and evaluate Q&A pairs. The automated LLM-based hallucination evaluation is based on the following steps:

1) **Question Generation:** The evaluator LLM uses the provided documents to generate a question that has a verifiable answer in those documents.
2) **Answer Retrieval:** The question is submitted to each of the four RAG configurations, yielding four answers (one per configuration).
3) **Hallucination Scoring:** The LLM compares each answer against the source documents and assigns a hallucination score on the same 0–5 scale, where 0 means the answer is fully supported (no hallucination) and 5 means the answer is entirely unsupported by the documents (complete hallucination).

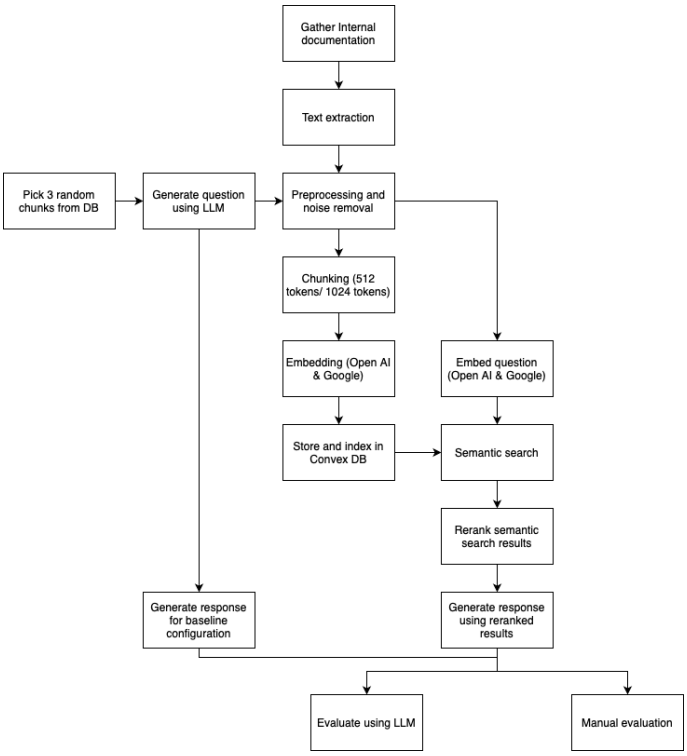

Figure 1. End-to-end Workflow of experimental setup

5. System implementation and Experimental Setup

This section describes the design and technical implementation of the research. An experimental platform was build to evaluate hallucination mitigation in RAG systems for enterprise documentation. A fully custom built Knowledge Management System (KMS) was developed using Convex and Next.js, including an integrated vector database and web-based user interface.

Multiple RAG configurations are supported by the system, using Google's gemini-embedding-exp-0-07 model and OpenAI's text-embedding-3-small model as embedding models. Documents were preprocessed, chunked, embedded, and stored under the same experimental conditions in order to examine the impact of chunking strategies and embedding models on hallucination rates. This infrastructure provided a strong basis for comparative analysis by enabling the automated and manual evaluation of 100 generated question-answer pairs across various configurations.

5.1. System Implementation

5.1.1. Chunking strategies. The system for this research was built using Convex as the backend [15] (including the built in vector database) and a Next.js frontend. Next to that a fully custom document ingestion and question-answering (QA) pipeline was built. On the backend uploaded PDF's are processed by a custom Python service that handles text extraction and cleanup. This python service removes noise (such as boilerplate or formatting) and then splits the documents into manageable chunks. We generate two sets of chunk sizes, one being approximately 512 tokens ("small" chunks), and 1024 tokens ("large" chunks), while respecting sentence boundaries. The two chunk sizes were chosen as a trade-off between semantic resolution and recall. Smaller chunks capture details and are more precise when it comes to context matches, while larger chunks have more overall context from the source documents (reducing the chance of missing relevant information).

5.1.2. Embedding models. For semantic embeddings of the text we integrated two mainstream models: OpenAI's text-embedding-3-small and Google's gemini-embedding-exp-03-07. These models were selected because of their widespread use in the industry and their different vector sizes (OpenAI's model produces 1536 dimensional embeddings, while googles Gemini produces 3072 dimensional embeddings). Each document chunk is embedded into high-dimensional vectors by both embedding models.

All chunk embeddings (along with the chunk text and the metadata) are stored in Convex's vector database. This enables efficient similarity search. This results in four chunk configurations. In addition, we also included a baseline configuration where no retrieval is used. So in total the system can operate in five configurations for answering queries, which were all used in our evaluation:

1) **No RAG (Baseline):** No retrieval, the query is answered by GPT-3.5 Turbo directly without any document context.
2) **Small Chunks + OpenAI Embeddings:** Retrieved from the index of 512-token chunks embedded with OpenAI's model.
3) **Small Chunks + Google Embeddings:** Retrieved from the 512-token chunk index embedded with Google's model.
4) **Large Chunks + OpenAI Embeddings:** Retrieved from the index of 1024-token chunks with OpenAI's embeddings.
5) **Large Chunks + Google Embeddings:** Retrieved from the 1024-token chunk index with Google's embeddings.

Each of the four configurations (2-5) uses the same overall pipeline but with a different configuration of chunk size and embedding model, as outlined above.

5.1.3. Query Processing and Answer Generation Workflow. When a user query is submitted, it undergoes the same pre-processing as the documents and is then converted into an embedding using the model for the chosen configuration. The system performs a similarity search in the Convex vector store to retrieve the most relevant chunks of text. We then apply an LLM-based re-ranking step. The initial set of retrieved chunks is given to a language model which ranks (or filters) them by relevance for the original query from the user. This ensures that the answer in the end is built on the most relevant information. Finally, the top-ranked chunks (as supporting context) are combined with the user's question and passed to OpenAI's GPT-3.5 Turbo to generate the final answer. We used GPT-3.5 Turbo in all configurations to control for model variance and focus only on the effects of the RAG configuration. Newer or other LLMs could differ in hallucination behavior, but using a fixed model allowed for a more controlled comparison. This RAG approach is focusing on grounding the answers in the enterprise internal documentation which reduces hallucination by providing context to the LLM.

5.2. Experimental Setup

To evaluate the system's performance and its tendency to hallucinate, we did experiments with both automated and manual assessment components. First, we constructed a set of 100 queries using a separate LLM. This LLM was given a random subset of the chunks that were available in the knowledge base, and was instructed to generate realistic questions that were answerable using those chunks. By using chunks that were available in the knowledge base as a basis, we ensured that each generated question had a verifiable answer in the knowledge base. We then ran all 100 queries through the QA pipeline under using all five configurations (consisting of four RAG configurations and one no-RAG baseline configuration), producing a total of 500 question-answer pairs for analysis.

5.2.1. Automated evaluation. For the automated evaluation we used an LLM-based scoring system to rate the answers for factual accuracy and hallucination. The evaluation LLM was provided with each answer from the configurations (along with the ground-truth answer) and asked to judge how much of the answer was unsupported or invented. It assigned a hallucination score from 0 to 5 for every answer. Where 0 meant the answer was entirely grounded in the internal documentation (no hallucination at all) and 5 meant the answer was completely fabricated (a complete hallucination). This produced a automatic hallucination score for every output across the five configurations.

5.2.2. Manual evaluation. We also performed a manual evaluation to ensure the robustness of our results. We randomly selected 25 of the queries (out of the 100 available queries) and had two human annotators independently review the answers from each configuration. Using an Excel-based annotation sheet, the annotators scored each answer on the same 0–5 hallucination scale. We used this hybrid evaluation approach to increase confidence in our findings. To ensure the reliability of the manual annotations, we calculated the inter-annotator agreement. Each annotator independently rated 125 answers (25 queries across 5 different configurations.) on a 0-5 scale. Given the ordinal nature of this scale, a weighted Cohens's Kappa was used to account for partial agreement. Although the dataset is relatively small, the resulting agreement score provides an indication of quality of the human evaluation process.

6. Results and Discussion

6.1. Hallucination Scores Across Configurations

When comparing any RAG configuration to the baseline (no retrieval), the results clearly demonstrate a decrease in the severity of hallucinations. The baseline produced responses with an average hallucination score that was noticeably higher in both the automated $(M = 3.33, SD = 1.13)$ and manual results $(M = 4.32, SD = 0.9)$, see Figures: 3 and 2. All RAG configurations performed noticeably better. Google embeddings with small chunks $(M = 1.47, SD = 1.55)$ performed the best in the automated evaluation, while Google embeddings with large chunks $(M = 0.68, SD = 1.16)$ performed best in the manual evaluation. Open AI embeddings with small chunks was the worst performing RAG configuration in both the automated evaluation $(M = 1.67, SD = 1.56)$ and manual evaluation $(M = 1, SD = 1.37)$. Nevertheless, it still significantly outperformed the baseline configuration, roughly halving the hallucination score, see Figure: 8. These scores show a shift from moderate to severe in the baseline configuration to minor levels of hallucination with a RAG configuration. Furthermore, the variability (standard deviation) of the responses of the RAG configurations was higher than the baseline, indicating that while most RAG responses were nearly free of hallucinations, a few outlier cases still showed moderate hallucinations. This spread suggests that the RAG configurations generally perform reliable but can occasionally fail.

6.2. Influence of Retriever Model and Chunk Size

Among the RAG configurations, the choice of the embedding model and chunk size had a measurable, although smaller, impact on the hallucination rates. Configurations using the Google embedding model consistently outperformed the configuration using the Open AI embedding model on the same chunk size. For example, with smaller chunk sizes, the Google-based RAG configurations achieved lower hallucination scores in both the automated $(M = 1.47)$ and manual $(M = 0.78)$ evaluations, compared to the Open AI configurations, which scored $(M = 1.67)$ and $(M = 1)$, respectively. A similar pattern emerges with larger chunk sizes: the Google-based RAG configurations scored an average of $(M = 1.55)$ in the automated evaluation and $(M = 0.68)$ in the manual evaluation, while the Open AI-based configurations scored $(M = 1.57)$ and $(M = 0.82)$, respectively, see Figures: 3 and 2. This suggests that the higher quality retrievals from the Google model led to fewer or less severe hallucinations.

Chunk sizes also influenced the results. For the Open AI embeddings larger chunk sizes performed slightly better in both the automated $(M = 1.57)$ vs $(M = 1.67)$ and the manual $(M = 0.82)$ vs $(M = 1)$ evaluations, see Figures: 9 and 10. For the Google embeddings there was a difference between the better performing chunk size when comparing the automated and the manual evaluations. The automated evaluations showed a better performance for the smaller chunk size $(M = 1.47)$ vs $(M = 1.55)$, whereas the manual evaluations showed a better performance for the larger chunk size $(M = 0.68)$ vs $(M = 0.78)$. So while both the retriever model and chunking strategy affect hallucination rates, their effect is small compared to the overall benefit of retrieval augmentation. In practice, the Google embedding and small chunk size configuration performed best.

6.3. GPT Evaluation vs. Human Annotation

To validate the automated evaluation results, a subset of 25 prompts were manually annotated by two human annotators. The manual ratings closely followed the automated evaluation trends. On the 5 point hallucination severity scale used by both annotators, the baseline configuration responses were judged almost maximally hallucinated **(Annotator 1:** $M = 4.4, SD = 0.91$**; Annotator 2:** $M = 4.24, SD = 0.97$**)**. In contrast, the best RAG configuration (Google embeddings + Large chunks) received nearly flawless scores, with average ratings close to 0 **(Annotator 1:** $M = 0.52, SD = 1.05$**; Annotator 2:** $M = 0.84, SD = 1.34$**)**, see Figures: 4 and 5 . When combining the two annotators results, the baseline dropped from an average of $M = 4.32(SD = 0.9)$ to just $M = 0.68(SD = 1.16)$ using the Google embeddings + Large chunks configuration. This shows an 85% reduction in severity. All the other RAG configurations showed the same magnitude of improvement

over the baseline configuration. The manual annotations confirm that RAG consistently outperforms the no-RAG baseline in reducing hallucinations, supporting the findings from the automated evaluations. Furthermore, both human annotators identified the Google based RAG responses as slightly more accurate on average than the Open AI based ones, confirming the small advantage seen in the automated evaluations.

6.4. Hallucination Severity Distribution

The distribution of hallucination severity ratings further supports the extent of the improvement achieved by RAG. Baseline responses were way more often classified as severe hallucinations. With almost none rated as "perfect" or even "minor" by either evaluation method. For example, the automated evaluation marked only 4 out of the 100 baseline answers as "perfect" (no hallucination at all), while the majority (over two-thirds) were rated as "moderate" or "severe" hallucinations, see Figure: 6. In the manual evaluations none of the 25 baseline answers were completely hallucination free (0 received a perfect rating), and almost all were ranked the highest severity by the annotators, see Figure: 7. In contrast, RAG generated responses shifted towards the lowest hallucination categories. Depending on the configuration around 37-42% of RAG responses were rated "perfect" by the automated evaluations, compared to only 4% for the baseline configuration. Severe hallucinations became quite rare when using RAG based configurations. The worst RAG configuration saw only 32 out of the 100 responses rated as moderate to severe in the automated evaluations, versus 77 out of 100 for the baseline configuration. The manual evaluations showed a similar shift. For the Google embeddings + Large chunks most responses were rated "Perfect" or "Minor", and almost no answers were judged to have severe hallucinations by either annotator. The reduction is not only in the mean score, but also in the severity distributions, see Tables: 1 and 2. RAG shifts the responses from frequently wrong to usually correct (or with minor, subtle mistakes).

6.5. Inter-Rater Agreement and Evaluation Reliability

The alignment between the automated and manual evaluations suggests that the automated evaluation was a reliable method for finding differences in hallucination severity. We found that both methods ranked the configurations from worst (baseline) to best (RAG with Google embeddings). Also both evaluation methods average scores were well aligned. The automated average score for the baseline configuration ($M = 3.3$) fell into the moderate-to-severe range, aligning with the manual evaluation average score which ended up in the high end of the same range. The automated evaluations ranked the Google based RAG responses as having the fewest hallucination, matching the human annotators results. The manual evaluation process proved to be very reliable. The two annotators achieved a Cohen's Kappa of **0.66** (unweighted), showing substantial agreement between both annotators. When looking at the amount of disagreement (using weighted kappa) we see an even stronger agreement **(linear weighted K = 0.84; quadratic weighted K = 0.93)**, see Figure: 11. The high inter-rater agreement and the alignment between automated and manual evaluations provide evidence for the reliability of the results. While the inter-rater agreement was high, the manual sample size (25 queries) is quite small. Future work could benefit from a larger scale manual evaluation. The results highlight that a RAG based generative AI system can reliably reduce factual hallucinations in enterprise documentation Q&A, and that tuning of retrieval configurations (such as embedding model and context chunk size) can result in additional gains in accuracy.

7. Conclusion

This research showed that incorporating Retrieval Augmented Generation (RAG) can significantly reduce hallucination in generative AI question-answering for enterprise documentation. Across all experiments, every RAG configuration outperformed the baseline configuration with retrieval in terms of both frequency and severity. The baseline configuration produced responses that were moderately to severely on average, whereas answers from RAG based configurations were far more grounded in facts. Even the worst performing RAG configuration roughly halved the hallucination severity compared to the baseline, and the best performing configuration achieved an 85% reduction in hallucination severity. Severe hallucination, which were quite common in the baseline configuration responses, became rare under RAG configurations, with a large portion of responses containing no factual errors at all. These improvements were consistent in both evaluation methods. Both the automated and manual evaluations observed the same reduction in hallucination when retrieval was enabled.

7.1. Alignment Between Human and Automated Evaluation Methods

The agreement between the automated and manual evaluations supports the robustness of these results. The automated evaluations aligned with the manual evaluations in ranking the different configurations performances. Both methods ranked the baseline configuration as the worst and the Google embeddings RAG configurations as the best performing configurations. The automated and manual evaluations both found fewer hallucination in the Google based RAG responses that in the Open AI ones.

Also, the manual evaluation was highly reliable. The two human annotators had a high inter-rater agreement, which indicates a consistent way of rating between the two annotators.

The high inter-rater agreement combined with the close alignment between the manual and automated evaluation process, confirms the reliability of the evaluation process,

which confirms the result that RAG mitigates factual hallucinations.

7.2. Practical Implications for Enterprise Deployment

The results show a few practical implications for enterprises looking to implement LLM-based solutions for internal documentation Q&A. The clear takeaway is that augmenting an LLM with a document retrieval component can significantly increase factual accuracy. By grounding the LLM responses with internal documentation, the system produced fewer fabricated or unsupported claims. This could improve users trust and deployment safety in corporate settings. In an enterprise setting (such as the Canon Medical Systems use case for this study), this means that employees get answers that are verifiably correct.

However, this study also shows that to realize RAG's full benefits requires careful tuning of the retrieval process. The choice of embedding model and the strategy on how to chunk the documents had a measurable impact on hallucination rates. In our experiment, configurations using the more advanced Google embedding model outperformed those with the Open AI embedding on the same chunk sizes. And finding the best performing chunk sizes for an embedding model even further improved the results. The best performing configuration was a combination of high-quality embeddings with an appropriate chunks size, which resulted in the lowest hallucination levels.

This suggests that enterprises should invest in retrieval quality. By selecting the best embedding models and fine tuning the how documents are chunked and retrieved, they can achieve a higher accuracy than RAG already provides. RAG is a promising approach for making LLMs more reliable on internal documentation, but the implementation should be optimized to get the most out of it.

7.3. Future Directions and Opportunities

This research concludes by outlining potential directions for further study and development in the field of lowering AI hallucinations. One of the directions is to implement a feedback loop system. For example, in a real deployment domain experts or end-users could correct or flag incorrect or unverified answers. This information can then automatically be used to refine the retrieval process. Such a human-in-the-loop mechanism, like discussed in the initial project proposal, could dramatically improve factual accuracy over time.

In addition to feedback loops, future research can explore improving the retrieval process itself. For example, fine-tuning the embedding model on enterprise domain specific text. This could further improve factual accuracy, especially when internal documentation contains a lot of domain specific jargon. This would be a good measure to catch corner cases where the current RAG configurations failed.

There is also room to explore strategies such as automated fact checking (for example, having multiple LLM's cross verify responses from each other) to combat any remaining hallucinations.

To conclude, this thesis provides evidence that RAG is an effective strategy for improving factual accuracy of LLM responses in enterprise settings. A few moderate hallucinations still remained. In these cases incorrect or loosely related chunks may have been retrieved. Future studies could improve reranking strategies, or incorporate confidence calibration and cross-verification with multiple LLMs to even further reduce hallucinations. By combing retrieval augmentation with ongoing refinement of the retrieval process (through careful tuning and potential feedback mechanisms), enterprises can significantly mitigate hallucination risks, and build more dependable AI systems for their internal documentation and knowledge bases.

8. AI Use Disclosure Statement

Large Language Models (LLMs), including OpenAI's GPT-4, were used to assist in rephrasing text for clarity, and suggesting improvements to the structure and flow of certain sections. All experimental design, implementation, evaluation, data analysis, and final editing were conducted by the author. AI assistance was used solely to support writing quality and did not generate any results or insights independently.

References

[1] Y. Yang, Y. Ma, H. Feng, Y. Cheng, and Z. Han, "Minimizing hallucinations and communication costs: Adversarial debate and voting mechanisms in llm-based multi-agents," *Applied Sciences*, vol. 15, no. 7, p. 3676, 2025. [Online]. Available: https://doi.org/10.3390/app15073676

[2] J. Vrdoljak, Z. Boban, M. Vilović, M. Kumrić, and J. Božić, "A review of large language models in medical education, clinical decision support, and healthcare administration," *Healthcare*, vol. 13, no. 6, p. 18, 2025. [Online]. Available: https://doi.org/10.3390/healthcare13060603

[3] P. Béchard, "Reducing hallucination in structured outputs via retrieval-augmented generation," https://arxiv.org/abs/2404.08189v1, 2024, arXiv preprint arXiv:2404.08189.

[4] C. Yao and S. Fujita, "Adaptive control of retrieval-augmented generation for large language models through reflective tags," *Electronics*, vol. 13, no. 23, p. 4643, 2024. [Online]. Available: https://doi.org/10.3390/electronics13234643

[5] R. Upadhyay and M. Viviani, "Enhancing health information retrieval with rag by prioritizing topical relevance and factual accuracy," *Discover Computing*, vol. 28, no. 27, 2025. [Online]. Available: https://doi.org/10.1007/s10791-025-09505-5

[6] Y. Lee, "Developing a computer-based tutor utilizing generative artificial intelligence (gai) and retrieval-augmented generation (rag)," *Education and Information Technologies*, vol. 30, no. 7841–7862, 2025. [Online]. Available: https://doi.org/10.1007/s10639-024-13129-5

[7] S. Packowski, I. Halilovic, J. Schlotfeldt, and T. Smith, "Optimizing and evaluating enterprise retrieval-augmented generation (rag): A content design perspective," in *Proceedings of the 2024 The 8th International Conference on Advances in Artificial Intelligence (ICAAI 2024)*, ACM. London, United Kingdom: ACM, 2024, p. 6. [Online]. Available: https://doi.org/10.1145/3704137.3704181

[8] G. Zhang, Z. Xu, Q. Jin, F. Chen, Y. Fang, Y. Liu, J. F. Rousseau, Z. Xu, Z. Lu, C. Weng, and Y. Peng, "Leveraging long context in retrieval-augmented language models for medical question answering," *npj Digital Medicine*, vol. 8, p. 239, 2025. [Online]. Available: https://doi.org/10.1038/s41746-025-01651-w

[9] B. B. Ozmen and P. Mathur, "Evidence-based artificial intelligence: Implementing retrieval-augmented generation models to enhance clinical decision support in plastic surgery," *Journal of Plastic, Reconstructive & Aesthetic Surgery*, vol. 104, pp. 414–416, 2025. [Online]. Available: https://doi.org/10.1016/j.bjps.2025.03.053

[10] L.-C. Chen, M. S. Pardeshi, Y.-X. Liao, and K.-C. Pai, "Application of retrieval-augmented generation for interactive industrial knowledge management via a large language model," *Computer Standards & Interfaces*, vol. 89, p. 103995, 2025. [Online]. Available: https://doi.org/10.1016/j.csi.2025.103995

[11] F. Liu, J. Jung, W. Feinstein, J. D'Ambrogia, and G. Jung, "Aggregated knowledge model: Enhancing domain-specific qa with fine-tuned and retrieval-augmented generation models," in *Proceedings of the 4th International Conference on AI-ML Systems (AIMLSystems 2024)*. ACM, 2024. [Online]. Available: https://doi.org/10.1145/3703412.3703434

[12] X. Xu, S. Liu, L. Zhu, Y. Long, Y. Zeng, X. Lu, J. Li, and Y. Dong, "Development and evaluation of a retrieval-augmented large language model framework for enhancing endodontic education," *International Journal of Medical Informatics*, vol. 203, p. 106006, 2025. [Online]. Available: https://doi.org/10.1016/j.ijmedinf.2025.106006

[13] Databricks, "Improving retrieval and rag with embedding model fine-tuning," https://www.databricks.com/blog/improving-retrieval-and-rag-embedding-model-finetuning, 2024, accessed: 2025-06-12.

[14] J. Qu, J. Liu, X. Liu, M. Chen, J. Li, and J. Wang, "Pncd: Mitigating llm hallucinations in noisy environments – a medical case study," *Information Fusion*, vol. 100, p. 103328, 2025. [Online]. Available: https://doi.org/10.1016/j.inffus.2025.103328

[15] Convex, "Convex docs," 2025, accessed: 2025-06-12.

Manual evaluation				
Severity Classification	Perfect	Minor	Moderate	Severe
No RAG	0	0	11	14
Open AI + Small	14	5	6	0
Google + Small	16	5	3	1
Open AI + Large	14	7	4	0
Google + Large	16	5	4	0

TABLE 1. DISTRIBUTION OF HALLUCINATION SEVERITY SCORES ACROSS RAG CONFIGURATIONS FROM THE MANUAL EVALUATION. THE TABLE SHOWS A CLEAR REDUCTION IN MODERATE AND SEVERE HALLUCINATIONS WHEN RAG IS APPLIED.

Automated evaluation				
Severity Classification	Perfect	Minor	Moderate	Severe
No RAG	4	19	69	8
Open AI + Small	38	30	30	2
Google + Small	41	31	26	2
Open AI + Large	42	27	27	4
Google + Large	37	34	27	2

TABLE 2. HALLUCINATION SEVERITY DISTRIBUTION ACROSS CONFIGURATIONS BASED ON AUTOMATED LLM EVALUATION. ALL RAG SETUPS MARKEDLY REDUCED MODERATE AND SEVERE HALLUCINATIONS COMPARED TO THE NO-RAG BASELINE.

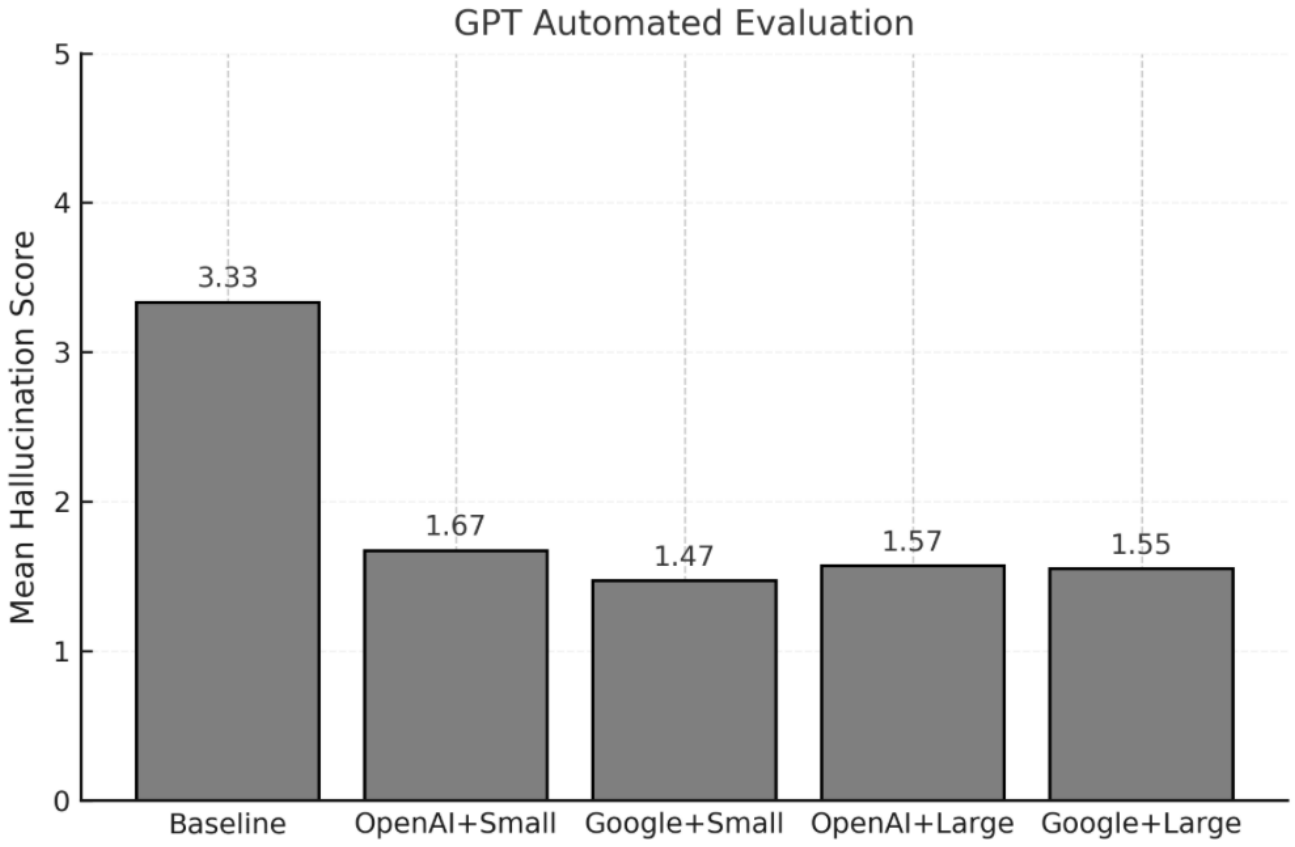

Figure 2. Mean hallucination scores from the automated evaluation across all configurations. The baseline (no RAG) shows the highest average hallucination score (3.33), while all RAG configurations substantially reduce hallucinations.

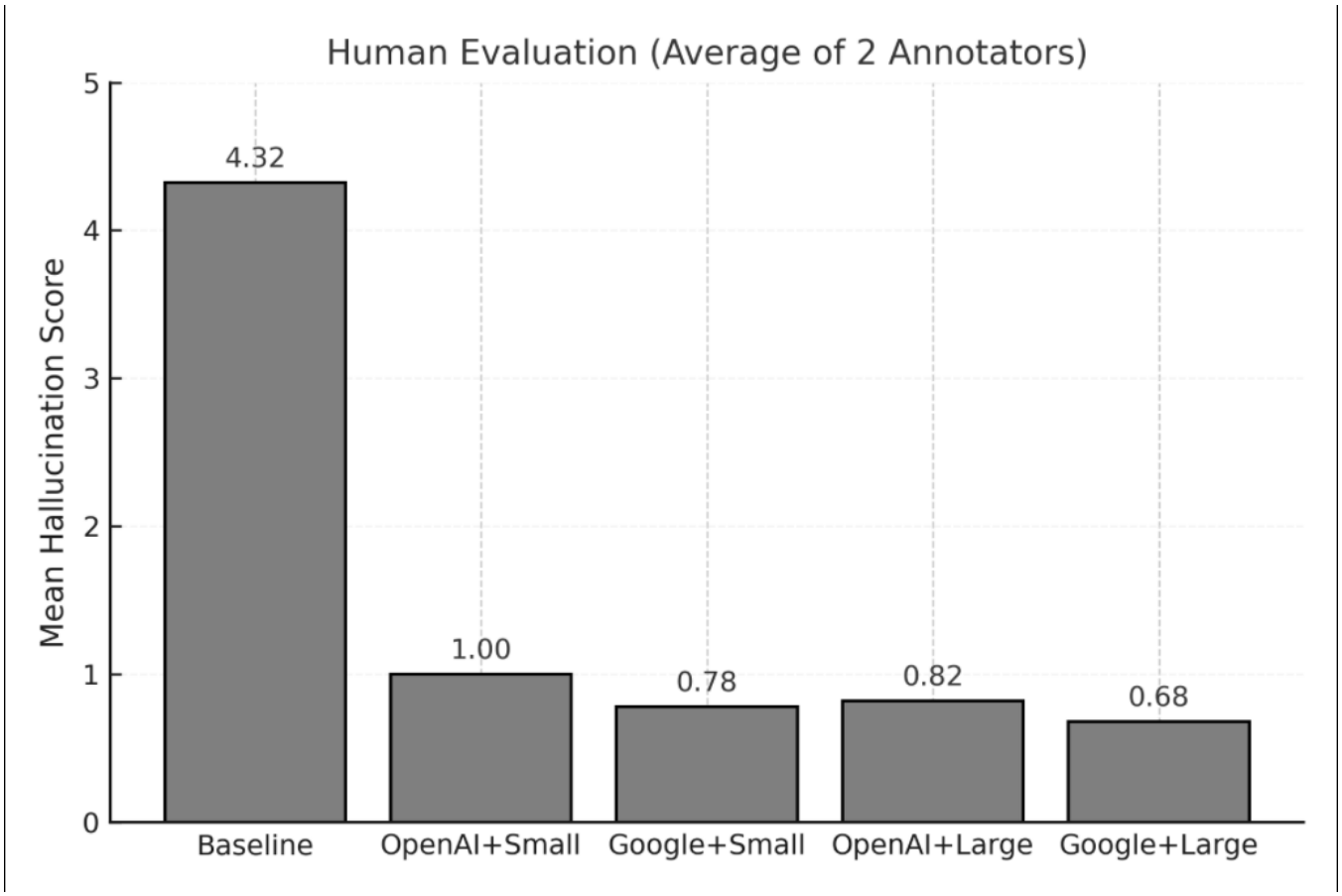

Figure 3. Mean hallucination scores from the manual evaluation (averaged across two annotators). The baseline configuration had the highest hallucination severity (4.32), while all RAG setups substantially lowered scores.

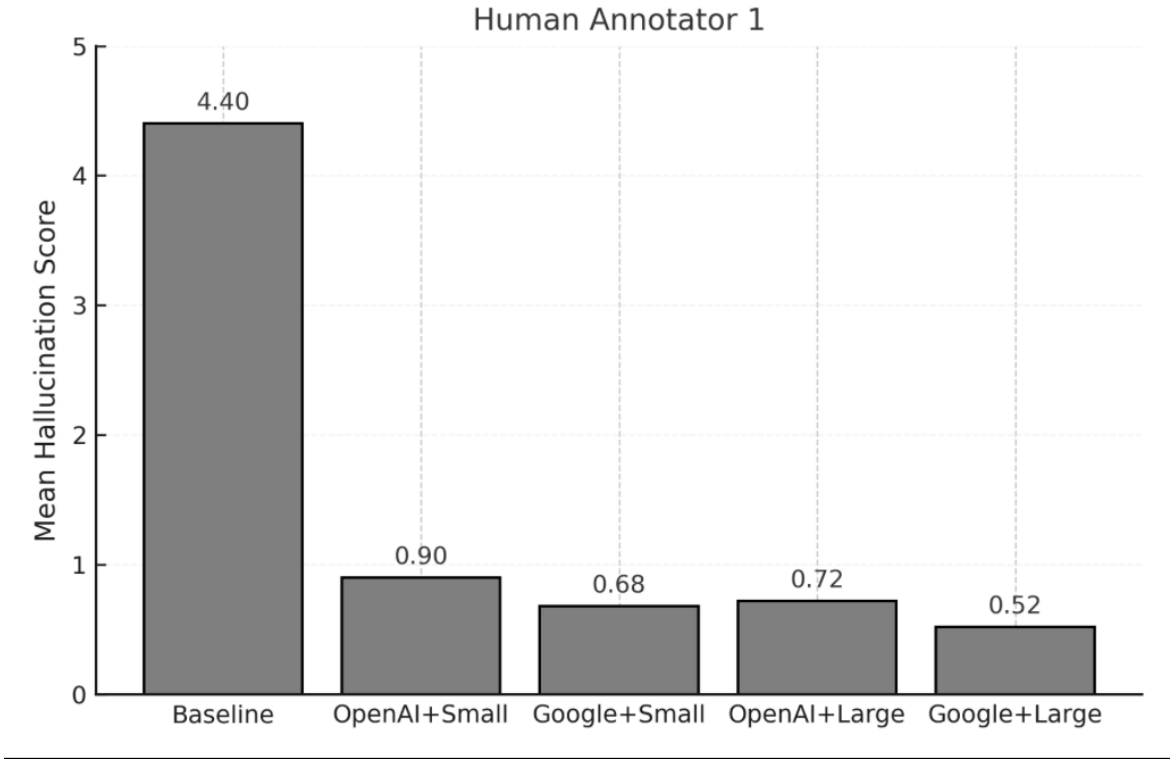

Figure 4. Mean hallucination scores per configuration based on Human Annotator 1. The baseline (no RAG) received the highest hallucination rating (4.40), while all RAG variants performed substantially better.

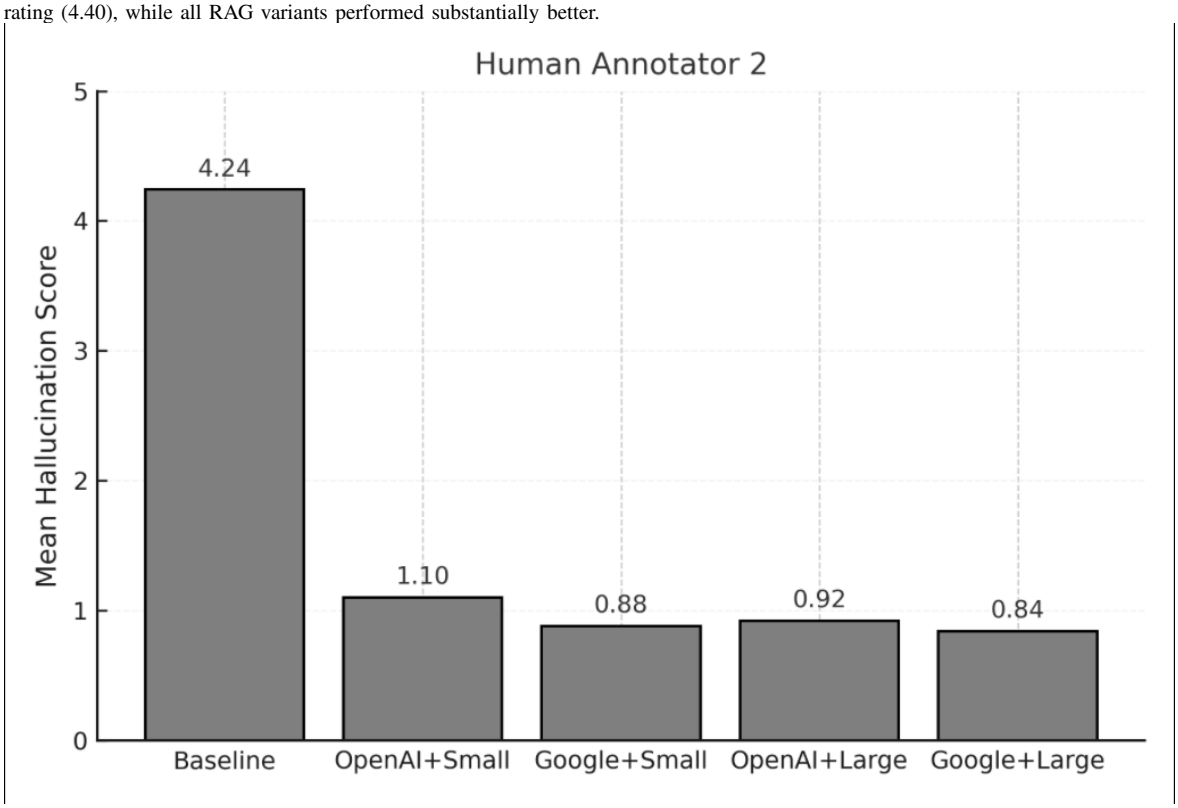

Figure 5. Mean hallucination scores per configuration as rated by Human Annotator 2. The baseline configuration had the highest hallucination severity (4.24), while all RAG setups achieved substantially lower scores.

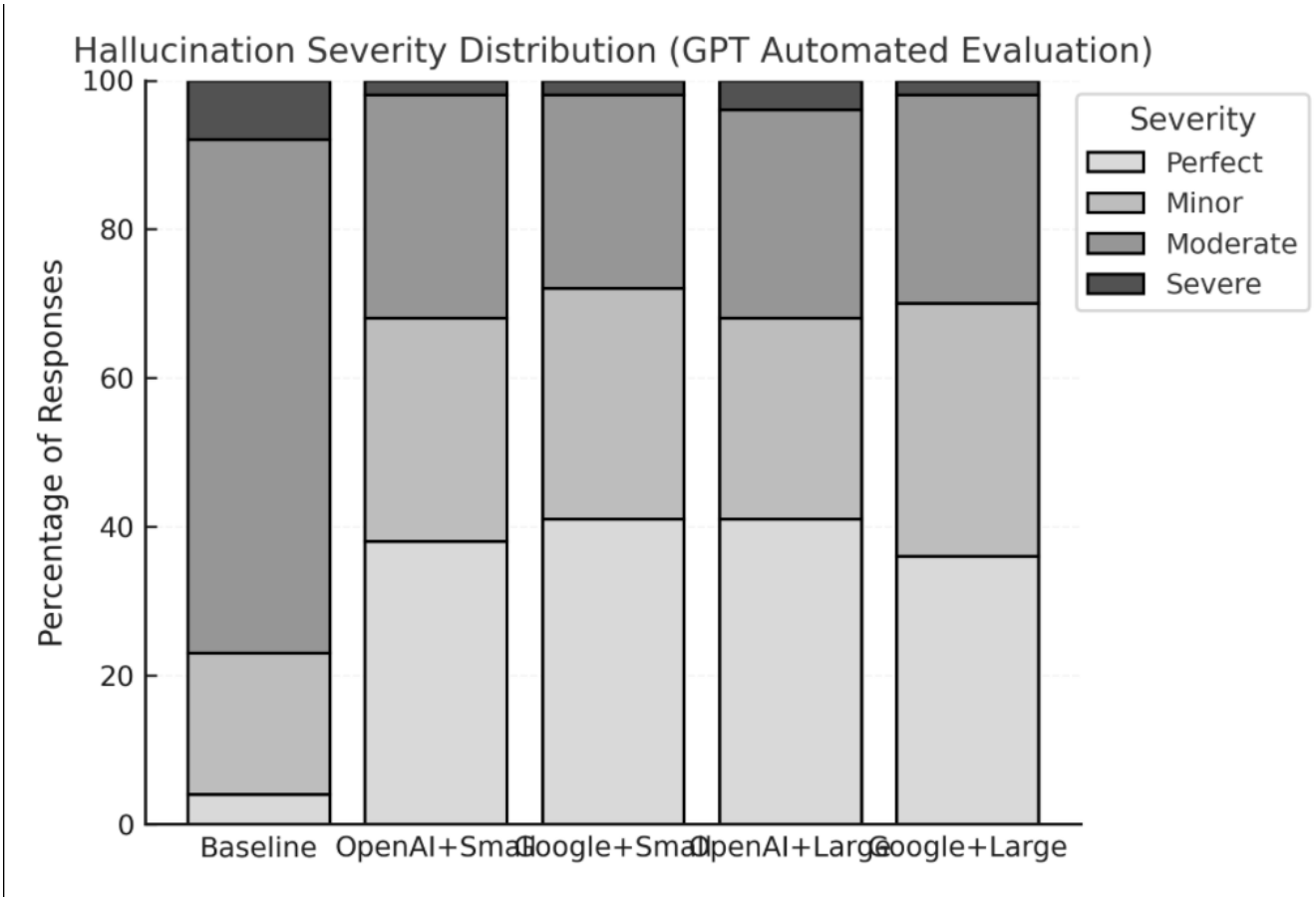

Figure 6. Distribution of hallucination severity levels across configurations from the automated evaluation. The baseline configuration shows a high proportion of moderate and severe hallucinations, while all RAG variants shift the distribution toward "Perfect" and "Minor" categories. This illustrates the effectiveness of RAG in reducing both the frequency and severity of hallucinated responses.

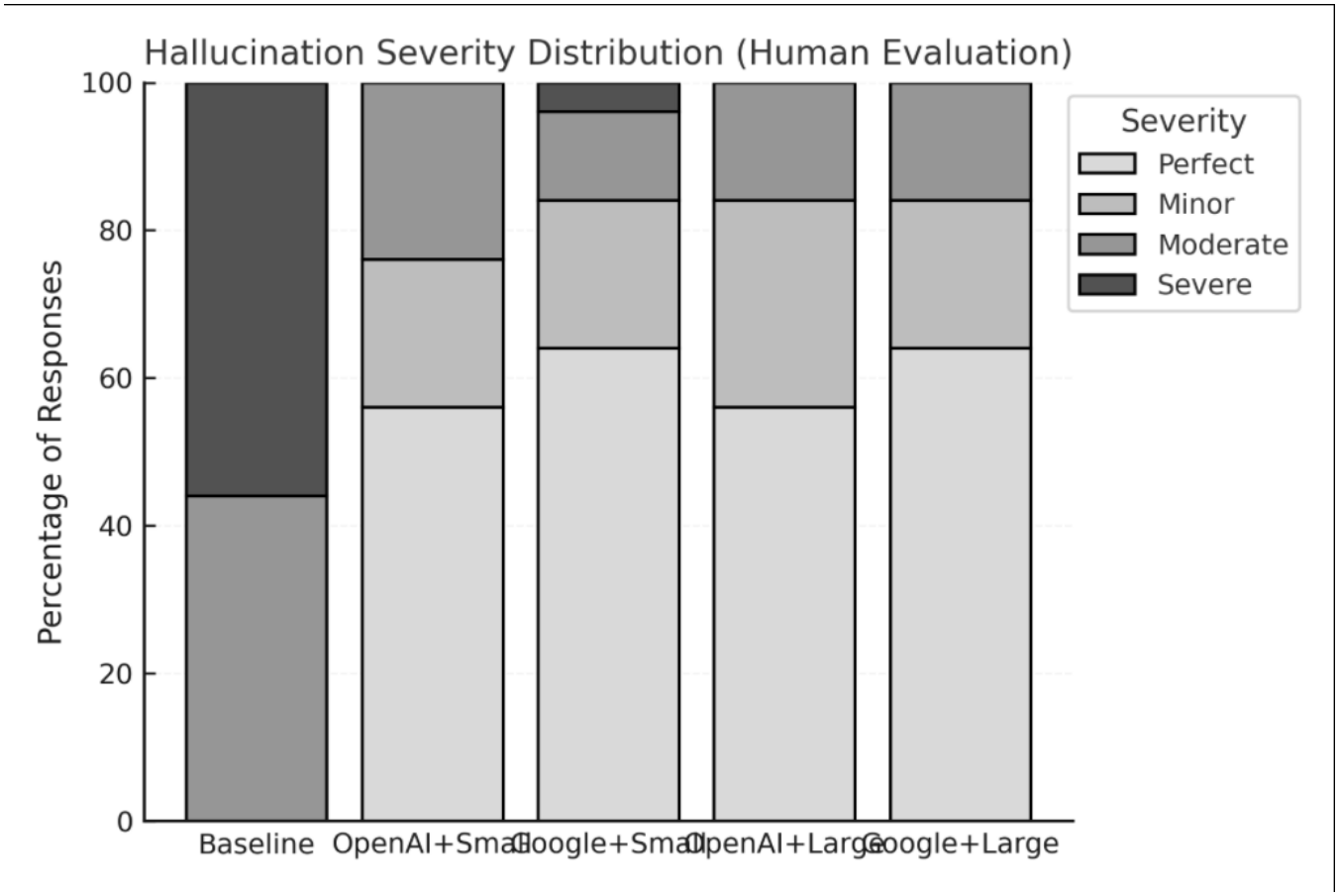

Figure 7. Distribution of hallucination severity levels based on manual evaluation. The baseline condition had the highest share of severe hallucinations, while all RAG configurations shifted responses toward the "Perfect" and "Minor" categories.

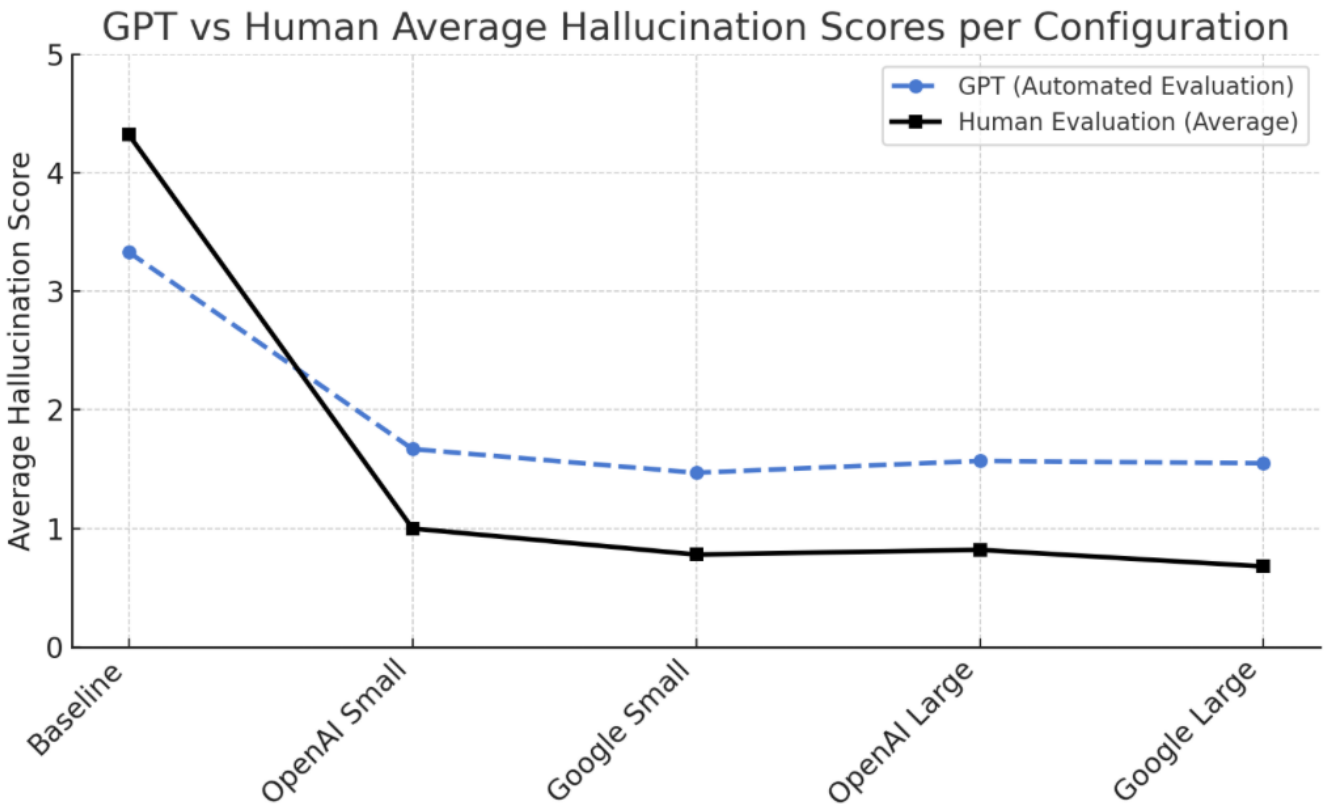

Figure 8. Comparison of average hallucination scores between GPT-based automated evaluation and human annotators across all configurations. Both methods show a consistent trend: RAG configurations significantly reduce hallucination compared to the baseline. The close alignment between the curves supports the reliability of the automated evaluation.

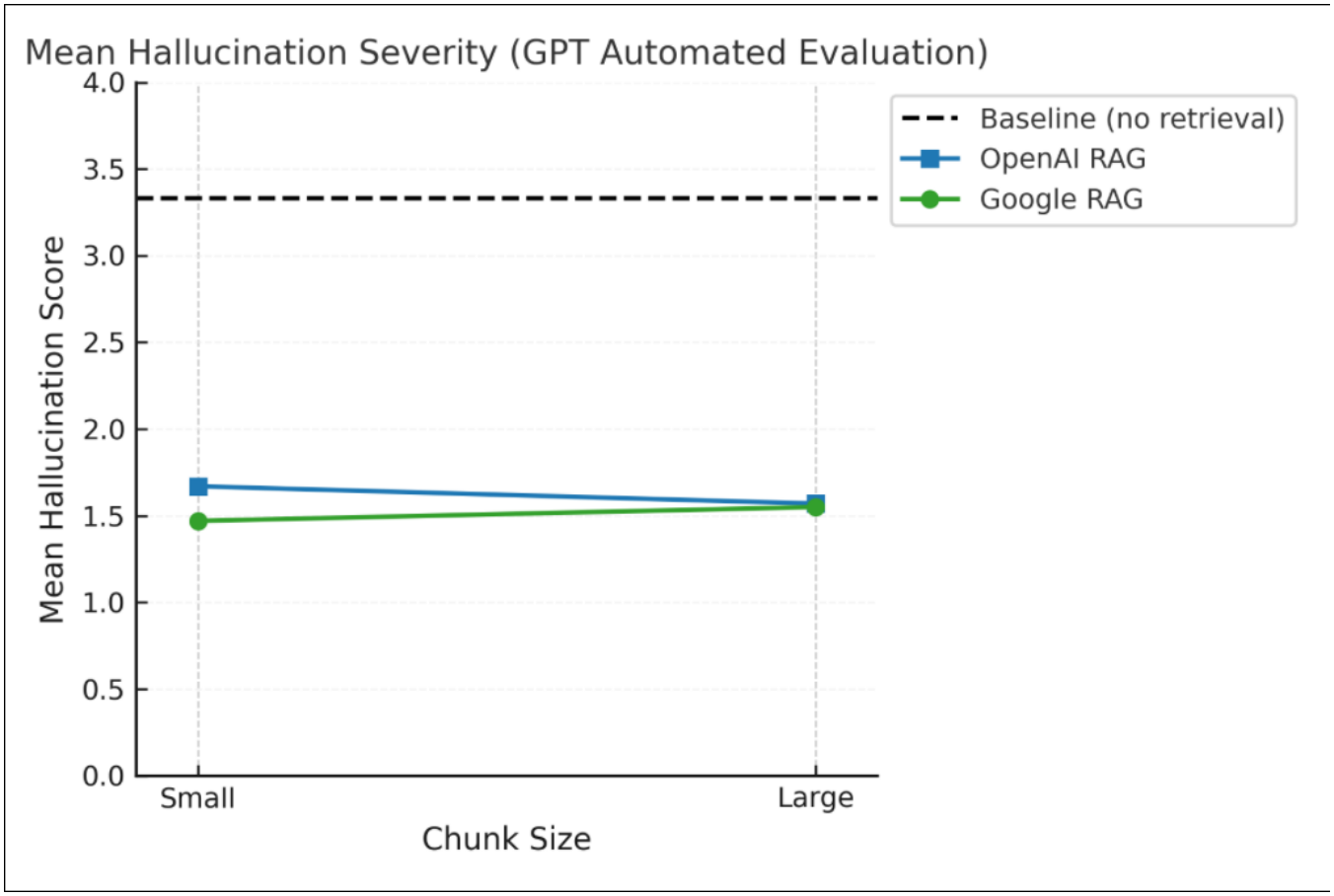

Figure 9. Mean hallucination scores from the automated evaluation across different chunk sizes and embedding models. All RAG configurations outperform the baseline (dashed line), with slight variations based on chunk size. OpenAI shows a small improvement with larger chunks, while Google maintains consistent performance across chunk sizes.

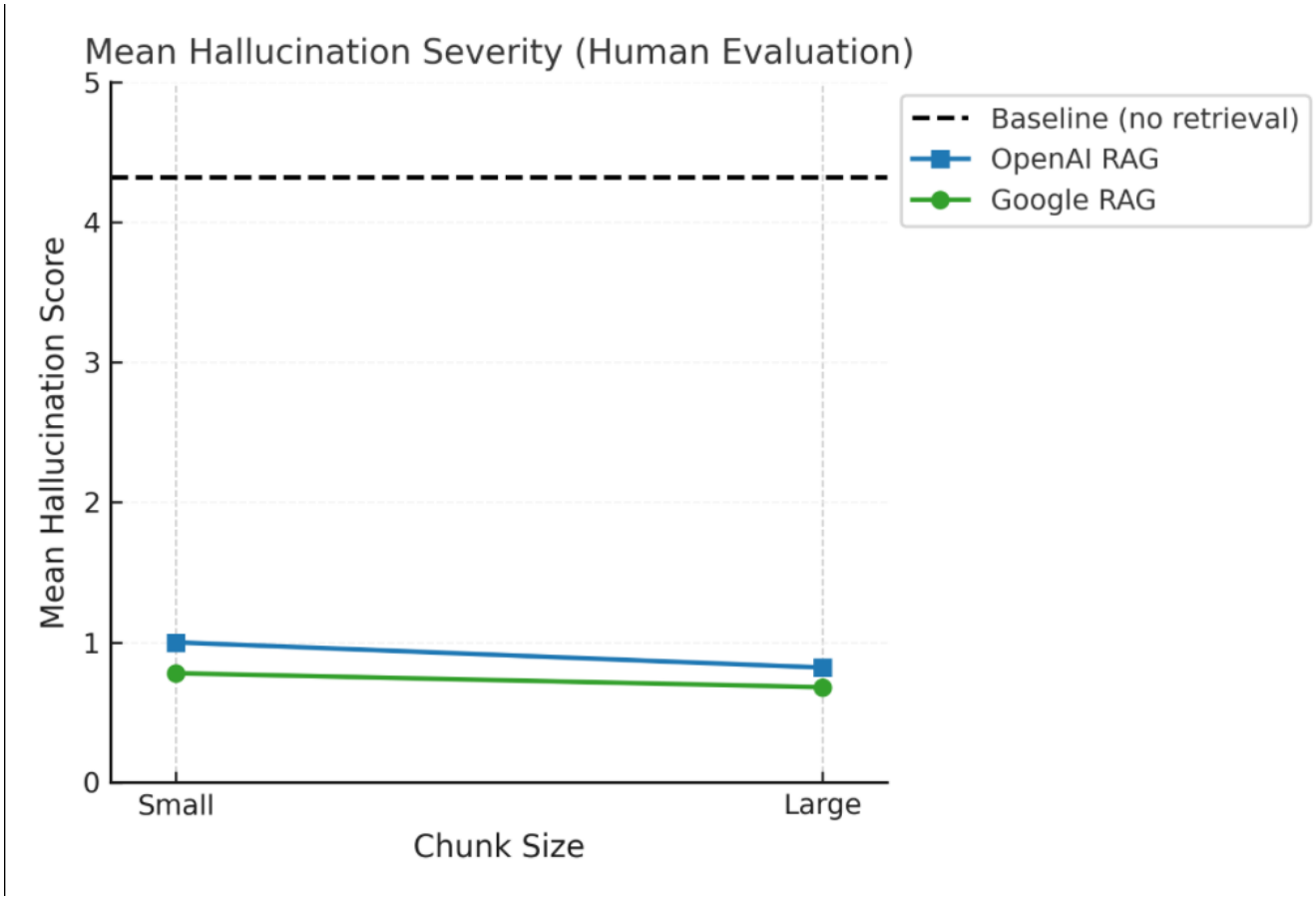

Figure 10. Mean hallucination scores from the manual evaluation across chunk sizes and embedding models. All RAG setups significantly outperform the baseline (dashed line). Google embeddings combined with large chunks achieved the lowest hallucination scores, while both embedding models show a small reduction in hallucination when using larger chunks.

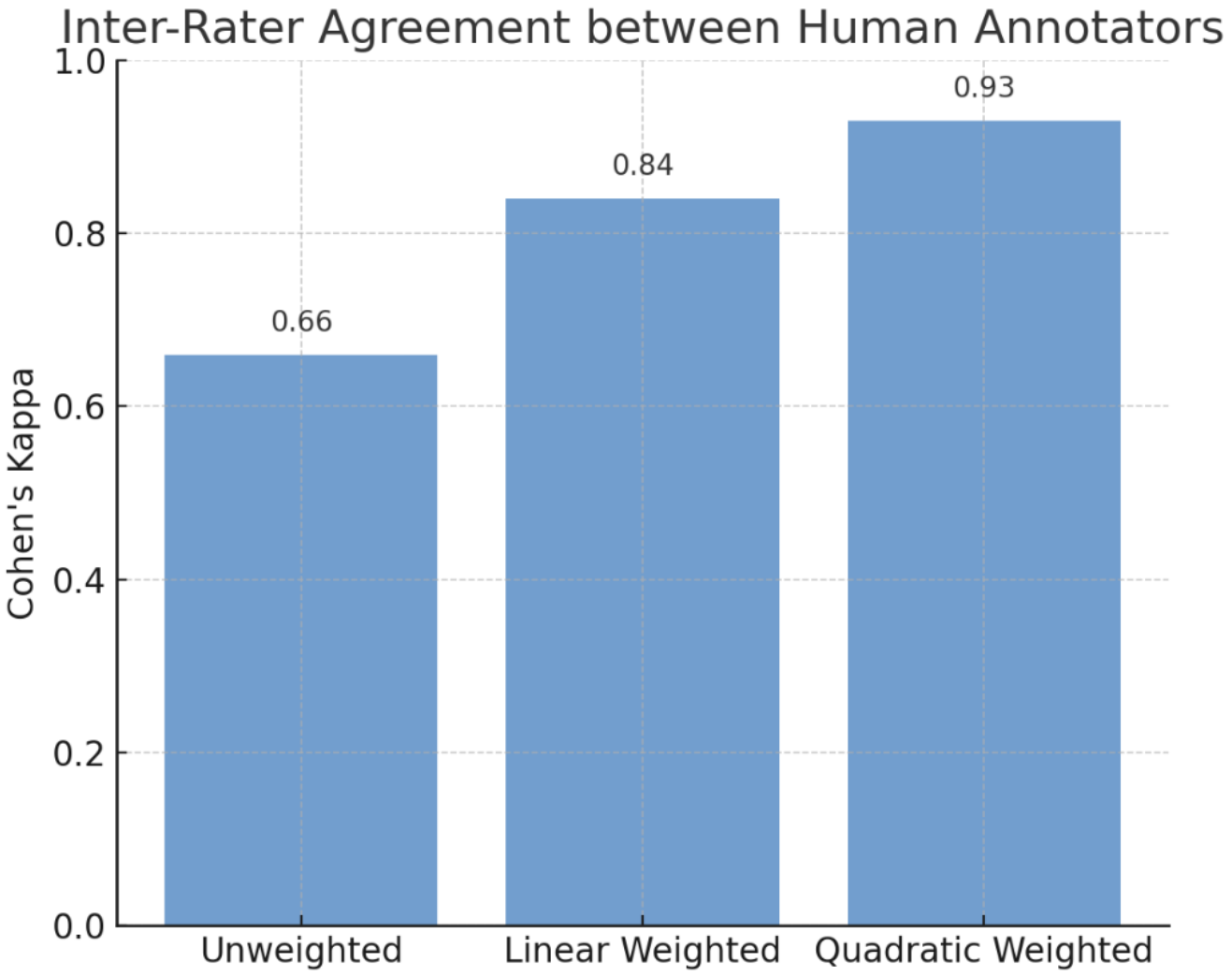

Figure 11. Inter-rater agreement between the two human annotators, measured using Cohen's Kappa. Agreement improves with weighting: unweighted K = 0.66 (substantial), linear weighted K = 0.84, and quadratic weighted K = 0.93 (almost perfect), indicating high reliability of manual hallucination scoring.