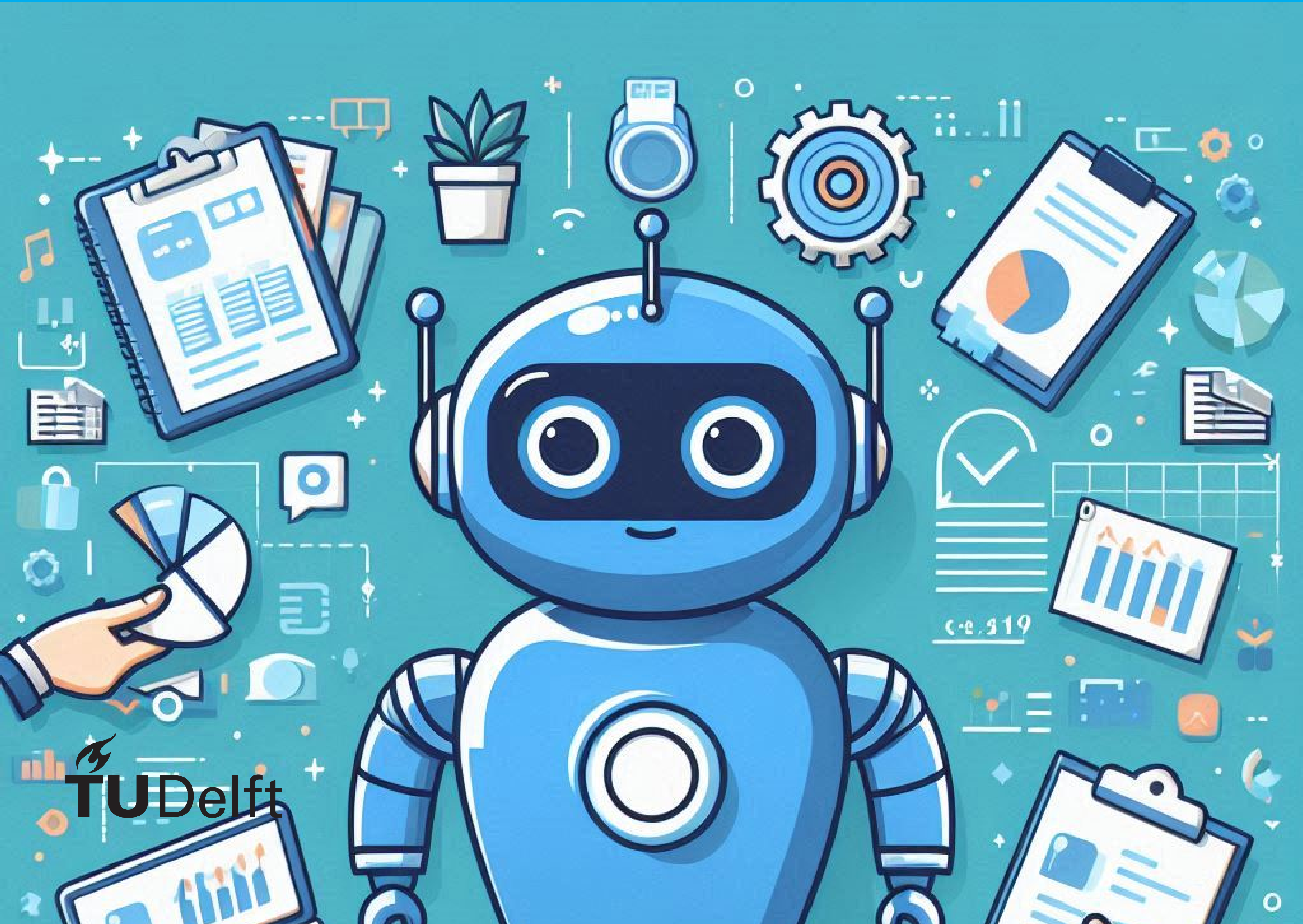
Evaluating the Efficacy and User Reliance on RAG Model Outputs

A comparative study with human experts

R. R. Sobha



TUDelft

Evaluating the Efficacy and User Reliance on RAG Model Outputs

A comparative study with human experts

by

R. R. Sobha

to obtain the degree of Master of Science
at Delft University of Technology,
to be defended publicly on Thursday August 29, 2024 at 12:30 PM.

Student number:	4565487	
Project duration:	November 13, 2023 – July 31st, 2024	
Thesis committee:	Asst. Prof. U.K. Gadiraju	TU Delft; thesis supervisor / chair
	Asst. Prof. B. P. Ahrens	TU Delft; co-chair
	G. He, MSc.	TU Delft; daily supervisor
Internship supervisor:	J. Slaats, MSc.	Ministry of Finance; Government of the Netherlands

TU Delft

Abstract

The emergence of conversational AI systems like ChatGPT and Microsoft Copilot has impacted how users engage in information retrieval. Retrieval Augmented Generation (RAG) harnesses the potential of Large Language Models (LLMs) with unstructured data, creating opportunities in science and business. RAG-based models have gained popularity, but their effectiveness and user reliance in organizational settings call for exploration. This thesis involves a user study with policy experts in the financial domain. They were tasked with text aggregation using a basic RAG model. The study delves into the model's performance and the temporal development of user reliance among the experts over four weeks. Our key findings reveal that outputs assisted by RAG do not match the quality produced by human experts. The RAG model, however, excels in specific aspects such as structure, spelling, and grammar. Additionally, the experts express satisfaction with the efficiency of RAG. Our findings suggest that user reliance on RAG increases with experience. This underscores the need for interventions and policies to support responsible human-AI collaboration. This work represents an effort to measure the temporal aspects of user reliance within an RAG system. Simultaneously, it assesses the system's efficacy in a field study with policy experts in the financial domain.

Preface

It all started in September 2016, when I set off to study Computer Science Engineering at TU Delft for my bachelor's degree. Eight years later, having collected 300 study credits accompanied by many projects, exams, and challenges. For example, writing my bachelor's thesis in 2020 during COVID-19, defending my research poster through Discord, and picking up my diploma on a random afternoon from the education office; is a memory forever engraved in my mind. Despite TU Delft having been one of the greatest challenges I have had to face in life, being able to close this chapter feels rewarding, and yet it feels nostalgic.

In the last nine months, I have been writing this thesis, and, during this time, I have learned to do literature research, set up user experiments, perform stakeholder management, maintain flexibility, and navigate experimental challenges. Most importantly, I found my passion in conversational AI and plan to turn it into my career for the foreseeable future. I am glad to have made the right call by writing my thesis in conjunction with an internship which allowed me to have multiple people checking in.

I would like to express my deepest gratitude to my main supervisor, Ujwal Gadiraju, for always being patient and encouraging me when I felt insecure about my progress. Simultaneously, I would like to thank my daily supervisor, Gaole He, who even agreed to meet with me online from *down under* (i.e., Australia) for an hour almost weekly for three months despite the large time difference. I would like to believe his guidance and critical view made me a better researcher. Without him, the quality of my work would not have been salvageable. Next, I would like to thank Benedikt Ahrens for taking the time to assess the quality of my thesis work and join the thesis defense. Of all the people who supported me during my thesis, last, but not the least important is my internship supervisor, Jop Slaats, for giving me unending support, allowing me to be flexible with my thesis work and my internship work, and keeping me accountable through multiple progress meetings. Not to forget, I express my gratitude to the Ministry of Finance for taking me on as an intern and to my co-workers for fostering a welcoming and progressive environment. Outside of working and writing my thesis, I would like to thank my relatives, friends, and partner for always supporting me emotionally. In particular, there are two people close to my heart whom I could not have done this without: my father and mother. Thank you for bearing with me and your unending love.

Finally, I am proud of myself for mustering the persistence and courage necessary to achieve my grand goals. Although this thesis report marks the end of my journey studying for a master's in Computer Science at TU Delft, a new journey commences as dawn approaches, the sun rises, and it shines a bright shimmering ray of light on a new graduate's future. Congratulations!

Rohan Ray Sobha
Rotterdam, the Netherlands
August 2, 2024

Contents

List of Figures

List of Tables

Glossary of Terms and Definitions

Technical terms:

- **GenAI**: Generative AI
- **MVP**: Minimally Viable Product
- **RAG**: Retrieval Augmented Generation
- **SLM/LLM**: Small Language Model/Large Language Model
- **DoRA**: Document Retrieval and Analysis (in Dutch: Documenten Raadplegen en Analyseren); the naive-RAG system used in this thesis.
- **QoW**: Quality of Work
- **IV/DV**: Independent/Dependent Variable
- **CoSim/COSIM**: Cosine Similarity
- **MMR**: Maximal Marginal Relevance
- **NLP/NLG**: Natural Language Processing/Natural Language Generation
- **SSO**: Single-Sign-On
- **PII**: Personally Identifiable Information
- **BPE**: Byte-Pair Encoding
- **DQL**: Database Query Language
- **QMDS**: Query-focused Multi-Document Summarization
- **RL/UL**: Reinforcement/Unsupervised Learning
- **DKE**: Dunning-Kruger Effect
- **GAISS**: General Attitudes towards Artificial Intelligence Scale
- **HCTS**: Human-Computer Trust Scale
- **TIA**: Trust in Automation
- **GDPR**: General Data Protection Regulation

Definitions:

- *Text aggregation tasks*: tasks that require analyzing one or multiple source texts and generating a new piece of text that is semantically relatable to its sources. For example, a generative summarization task and a task juxtaposing two source texts belong to this category.

Related to the Ministry of Finance:

- **MoF/MinFin**: Ministry of Finance (belonging to the Government of the Netherlands)
- **ADR/IT**: Auditdienst Rijk (Governmental Audit Services) - Sector IT
- **CdIO/Beleid**: Concerndirectie Informatievoorziening en Openbaarmaking - Beleid (Executive Board on Information Provision and Publication - Policy)
- **DGRB/BBH**: DG Rijksbegroting - Begrotingsbeheer (General Directorate of State Budget - Budget Management)

1 Introduction

The advent of OpenAI's ChatGPT in 2022 marked a significant milestone in the use of conversational AI. By March 2023, nearly 1.5 million individuals aged 13 years or older in the Netherlands [106] had used ChatGPT, many of whom are students in higher education. As students frequently express a high level of reliance on the source of information, they experience difficulty distinguishing hallucinated outputs from outputs referencing a credible source [44]. The increased accessibility and availability of AI dialog systems will exacerbate this behavior. For example, Zhai, Wibowo, and Li [2024] found that over-reliance on AI dialog systems, such as ChatGPT, negatively affects students' cognitive abilities such as decision-making [2], critical thinking, and analytical reasoning. As such, the detection of over-reliance will play a more prominent role in the upcoming years.

Former literature [145, 68, 52] investigates user reliance, but limits itself to instantaneous expressions of user reliance during one session. As temporal specificity [80, 13] is a function of changes in trust over time, this thesis addresses the temporal aspect of user reliance that takes shape between two sessions.

LLMs have demonstrated impressive abilities in various downstream tasks [111, 113, 25, 136, 162]. However, they often generate answers containing factual inaccuracies and fabricated content [161, 120, 148, 54, 94, 24], commonly referred to as ”hallucinations”, which causes users to have less trust in these answers. Generation paradigms [81, 82], such as Retrieval Augmented Generation (RAG), combine retrieval of vectors from a database with the power of a Large Language Model to generate verifiable answers. However, as the quality of RAG outputs has not been evaluated yet across multiple domains, this thesis is the first step towards evaluating the efficacy of RAG outputs in an organizational domain.

1.1. Research Questions

With the motivational context from the previous section in the back of our minds, we, firstly, would like to find out where an RAG model is positioned to another domain expert in terms of quality and how user reliance develops among the experts' interaction with the model.

When adopting a new technology, such as RAG, into large organizations, concerns regarding how this will fit in the existing (business) processes will be raised. There are three degrees of automation when embedded into an existing process. The first is full automation without supervision which requires the highest level of performance. The second is the human-in-the-loop [155, 1] which is a hybrid process with a human overseeing the input and output of the process and requires a medium level of performance as mistakes will be caught in the output of the process. The last is a system that supports workers in the existing process which is a form of Human-AI collaboration and requires the lowest level of performance as mistakes are spotted directly through constant interaction with the system. To find out the degree of automation, together with an AI risk assessment and other organizational factors, it is necessary to investigate the extent of the RAG model's output quality compared to its human counterpart. As participants within the research context deal with summarization tasks daily and summarization in RAG has been researched before [158, 150, 38], we consider document summarization as a user task. In addition, we consider juxtaposition, as many of these participants need to summarize multiple documents and highlight their differences in daily workflows. Hence, this leads to our first research question:

> *RQ1: To what extent does the output generated by a RAG-based LLM system achieve similar quality as human experts on text aggregation tasks such as summarization and juxtaposition?*

In the prior section, we mentioned that existing literature only focuses on instantaneous expressions of user reliance and we aim to investigate the temporal aspects of user reliance in this thesis. Cabiddu et al. [2022] investigate the effect of experience on building trust. They state that the following proposition:

> ”the greater the familiarity a user demonstrates with algorithms, the higher the probability of building trust in AI algorithms over time.”

This thesis posits that this proposition holds for shaping user reliance because the user's level of competence affects their expectation about the utility of decision aids [52, 19]. Therefore, we pose a second research question:

> *RQ2: How does experience with a RAG-based LLM system shape user reliance over time?*

1.2. Contributions

To answer the defined research questions in section 1.1, we propose a conversational RAG interface and three assessments where the interface serves to answer the research questions. More specifically, we enumerate the contributions of this thesis below:

- We propose DoRA, Document Retrieval and Analysis (in Dutch: Documenten Raadplegen en Analyseren), a RAG-driven chatbot that can perform query-focused document summarization.
- We evaluate the quality of the produced text outputs based on several dimensions of text output. They are clarity, completeness, relevance, accuracy, consistency, structure, conciseness, grammar and spelling, and coherency. We measure these using subjective 7-point Likert-scale questions asking to rate these dimensions. Our findings show that DoRA performed better than humans in two of these dimensions: structure, and spelling and grammar, but performed worse in the others. These dimensions are essentially black-box metrics because they solely rely on the source documents and the final summary. Therefore, our evaluation provides insight into the efficacy of the RAG system independent of any NLP metrics allowing for comparison between a RAG-generated and a human-generated summary.

  Additionally, to investigate the background of the ratings for each dimension, we perform a thematic analysis based on the freeform responses from the experts who used DoRA. Based on the participant's feedback, we derive four themes: Prompt Composition, Exact Compliance, Efficiency over Quality, and Prioritization of Content. Each theme suggests that domain context and user preference are crucial for the perceived quality of DoRA and the configuration of DoRA concerning prompt engineering, setting up the retrieval engine, and injecting the retrieved context into the prompt. As such, our analysis reinforces the importance of domain-driven design for implementing textual generative AI systems such as retrieval augmented generation.
- We provide a measurement of the temporal difference between two sessions in user reliance using both the metadata from the user session and the edit distance between the user's preferred summary from DoRA and their self-reported modification of that summary. We capture the edit distance using a variety of syntactic NLP metrics (e.g. BLEU and ROUGE-L) and semantic ones (e.g. Cosine Similarity [COSIM] and BERTScore). Our findings demonstrate a small increase in user reliance as participants spent less time overall in the second session and performed zero edits that changed the semantic meaning. Our work is a first step towards considering the temporal aspect of user reliance.

1.3. Thesis Outline

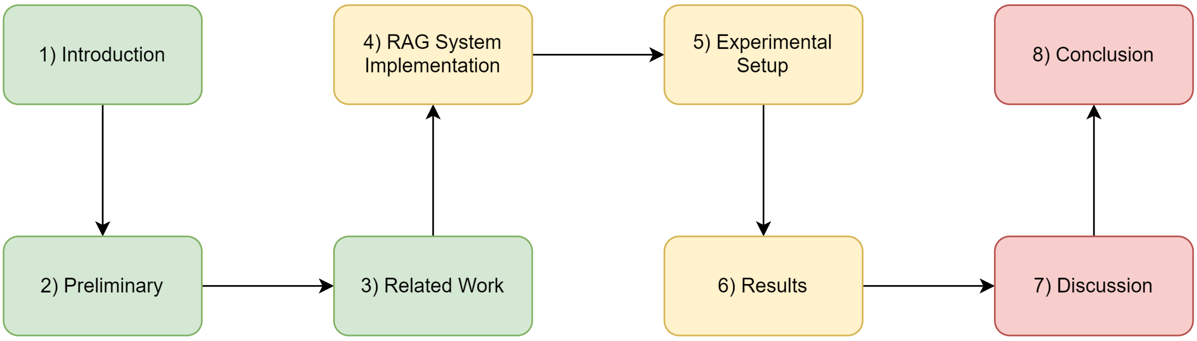


Figure 1.1: Illustration of the thesis outline: green indicates the chapters that discuss the build-up to the study, yellow indicates the chapters discussing the study itself, and red indicates the chapters discussing the results and implications.

This thesis encompasses the subsequent chapters, depicted in Figure 1.1. In Chapter 2, we explain preliminary knowledge of Large Language Models (LLMs) and Retrieval Augmented Generation (RAG), we define the user tasks related to text aggregation, and we contextualize the environment in which the experiment was conducted. Subsequently, we describe related work about user trust and reliance as well as summary evaluation and retrieval augmented generation in Chapter 3. Followed by a description of the proposed RAG system in Chapter 4 where both the front-end and back-end are discussed. The next chapter, Chapter 5, describes the hypotheses, variables, metrics, and the procedure for the user experiment. After that, we describe the results from the experiment in Chapter 6 and discuss these in Chapter 7 together with any limitations and future work. Lastly, we end this work with a conclusion in Chapter 8.

We supplement this thesis with some appendices related to the user experiment. Appendix A contains the survey that the users were requested to fill in. Appendix B outlines the answer options according to the 7-point Likert scale across several units of measurement. Appendix C gives an example of the task manual distributed to participants. We include these appendices to endeavor transparency, auditability, and reproducibility.

2 Preliminary

This chapter describes preliminary knowledge on Large Language Models (LLMs) and Retrieval Augmented Generation (RAG), defines the user tasks of summarization and juxtaposition, and provides the context of the environment in which the user experiment was conducted.

2.1. Large Language Models

State-of-the-Art Overview Large Language Models (LLMs) are essential for allowing a computer to understand humans through natural language. They are deep-learning, transformer-based models that can predict the next token based on a set of earlier tokens.

Tokens represent concepts in the natural language and can vary in length between single alphanumeric characters to words. Before the LLM can understand written text, it first needs to be tokenized. There are many ways of tokenizing depending on the granular level desired and the method has to suit the choice of large language model. Our tokenization method will be described in section 4.1.3.

Transformer-based models, c.q. LLMs, can be categorized into three categories: encoder-based, decoder-based, and encoder-decoder-based [103]. Many BERT-based models [27, 72, 33, 87, 62, 64] are examples of encoder-based models. Decoder-based models consist of the GPT family [130, 17, 111], LLaMA 2 [136] and Mistral-7B [60]. The last category of encoder-decoder-based models consists of models such as T5-based models [118] and MT-NLG [43].

However, LLMs have two shortcomings: the knowledge contained in custom documents – not present in the training set – is not known to them and their usefulness is dependent on leveraging this knowledge [29]. Additionally, as re-training a model is very expensive [123], retrieval-based augmented generation techniques help to save economic and environmental resources by decreasing power consumption and consuming less time training models. This technique has been discussed in section 2.2.

Open-source Dutch LLMs As the Dutch government conducts its business in the Dutch language, it is highly recommended to use a model that has been trained on datasets in this language [142]. Currently, the landscape of Open Dutch LLMs [142] consists of all fine-tuned or instruction-tuned models founded on either Mistral-7B [60] or LLaMA 2 [136]. Some of the tuned models that Vanroy [2023] examines are Zephyr [137], Orca-2 [101], Neural Chat [58] and GEITje [121].

Moreover, the paper proposes [142] "two fine-tuned variants of the Llama 2 ... 13B model." This includes a text generation LLM, `llama2-13b-ft-mc4_nl_cleaned_tiny` [144], and a chat model, `Llama-2-13b-chat-dutch` [143]. Vanroy has set up a leaderboard[1] to allow for an easy comparison between models.

Furthermore, there are plans to release GPT-NL [135], a virtual facility that aims to provide an ecosystem in which academia, commerce, and the public sector can study and experiment with LLMs to further enhance AI autonomy.

All in all—considering the leaderboard—Zephyr-7b-beta [137] and geitje-7b-chat-v2 [121] are suitable to serve as the backbone LLM for this thesis. This consideration is due to these models having

[1] `https://huggingface.co/spaces/BramVanroy/open_dutch_llm_leaderboard`

the highest average scores on multiple — into Dutch translated — benchmarks such as ARC [28], HellaSwag [156], MMLU [53] and TruthfulQA [84].

2.2. Retrieval Augmented Generation

Retrieval Augmented Generation (RAG) essentially enhances the capabilities of pre-trained parametric-memory generation models. The most common examples of such models are Large Language Models (LLMs). The methodology involves the integration of non-parametric memory into LLMs through a versatile fine-tuning approach, as outlined by Lewis et al. [81]. Non-parametric memory could represent a database, vector store [146], or index from which information can be retrieved to augment LLM outputs. A vector index could, for example, reference a Wikipedia article based on its vector embedding. The distinctive benefit of this framework is its capacity to avoid the necessity of retraining LLMs when encountering new information. Additionally, an adequately large LLM can demonstrate versatility across various purposes, tasks, and domains.

In the context of RAG, the terminology, introduced by Lewis et al. [2020], designates two roles for LLMs. The first role is as a Generator and the second role is as a Retriever. The generator, on the one hand, is characterized by a decoder-encoder model and its role is to generate responses based on the context and a prompt template. The context here refers to the retrieved documents and chat history which is then formatted into the prompt template. The retriever, on the other hand, is an encoder-only model that encodes tokenized snippets of text and the user's query. The choice and motivation for both the Generator and Retriever will be elaborated in chapter 4. Moreover, to ensure that the generator receives the appropriate context, we need to consider retrieval techniques and optimizations. The framework that ties all of these components together is called an orchestrator. All in all, the orchestration of generators, retrievers, information retrieval techniques, and retrieval optimizations, allows for a system that can retrieve information and answer the user's query on the fly with new information that the foundational LLM has not been trained on.

Section 2.2.1 provides an overview of currently available orchestration tools. The following subsections highlight each of the orchestration components one by one.

2.2.1. Orchestration tools

There are various orchestration tools available [76, 133, 119] that all demonstrate different levels of abstraction. Langchain [76] provides a high-level abstraction by providing the option of chaining tools and agents to collaborate to solve a task whilst being able to call upon external data sources (e.g. vector stores). Tools are API endpoints that the LLM can call upon to perform certain tasks (e.g. a calculator) and agents can be considered as sub-processes where the RAG system is being prompted with a more specific prompt to perform a chain of tasks (e.g. ordering a pizza[2]). LlamaIndex [133] focuses on the efficient retrieval of vectors from databases by abstracting away the complex task of manual vector retrieval and supports advanced retrieval techniques such as pre-retrieval optimization (e.g. sentence window retrieval) and retrieval optimization (hybrid search). We will explore these techniques and optimizations further in sections 2.2.4 and 2.2.5 respectively. Ragna's [119] strong suit is providing an intuitive API for rapid experimentation. It also includes scalable built-in tools for deploying apps to production. All these frameworks share common features such as the ability to call upon many different LLMs to generate responses [111, 5, 47] and to embed documents [107]. Another crucial feature that they have in common is an integration with a vector store such as ChromaDB [26].

2.2.2. Generator

The generator creates responses based on a prompt template with three parameters: the user query (`{query}`), the context (`{context}`), and the chat history (`{chat_history}`). `{Your_Prompt}` refers to some presets affecting the output. The prompt designer (e.g. AI/ML engineer) considers the business context and safety precautions to prevent the model from generating undesired or harmful output. The answer, `ANSWER`, follows as a token completion task, see listing 2.1 as a prompt template.

Listing 2.1: A prompt template for RAG

```
{Your_Prompt}
```

[2] https://www.youtube.com/watch?v=CPnKwEg2TlE

```
CONTEXT:
{context}

QUESTION:
{query}

CHAT HISTORY:
{chat_history}

ANSWER:
```

The user query is usually either a question or an instruction. On the one hand, users usually ask questions – within RAG – with the intent to retrieve information [16] given the context of their data. On the other hand, users write instructions to draft a new piece of text that adheres to a format that has syntactic and semantic constraints. An example of such instructions can be found in section 5.3.

In our definition of context, the retrieved documents make up the context. Additionally, the user provides their chat history by interacting with the RAG system over multiple messages. Their chat history, in turn, ascertains that the LLM knows how to understand the user when they refer to the information contained in earlier messages. The retrieved documents as context and the chat history together allow RAG to execute any task and answer any question given enough knowledge and context similar to a newbie on the job.

That leaves us with the answer as a parameter. The LLM then fills the answer parameter and gradually forms the most likely user-prompted outcome. In the conversation API, this section is then stripped from the prompt template and returned as a response.

All in all, those parts form the generator, prompting us to investigate the origins of our data-driven context further. We expand on this in the next section.

2.2.3. Retriever

The retriever has three key aspects that we will highlight here: the embedding model and tokenization methods, retrieval databases, and retrieval techniques and optimizations.

Embeddings Model and Tokenization Methods The embeddings model is, usually, an encoder-only LLM that converts tokens into a vector embedding. To obtain tokens from a text source, it necessitates extracting the text from the document together with encoded metadata and tokenizing it. Tokenization is the process of splitting up a string of text into smaller pieces that are semantically related. We distinguish tokens in three major categories [127]: word-based, character-based, and sub-word-based. Word-based tokenization aims to extract single words from the text by splitting according to whitespace, punctuation, and other delimiters which are each considered tokens as well. As each language treats words differently, rule-based tokenizers are often used to split up words based on the target language. This comes with the drawback that the tokenization process is not uniform across languages that tokenize based on words. Character-based tokenization is on the other side of the spectrum with a small vocabulary size which comes at a disadvantage of losing the semantics in each token. The vocabulary size is the number of permutated tokens you can create from the target language. At the character level, this comes down to approximately 200 tokens for the English language based on ASCII. Predicting the next token based on the letter ‘b’ using this method is equivalent to the distribution of words starting with the letter ‘b’, which is not useful for most NLP applications. Sub-word-based tokenization is in the middle and the most effective of the three. This category houses some commonly used tokenization algorithms such as Byte-Pair Encoding (BPE) [67], Word Piece [48], N-grams [65], and Sentence Piece [70].

In section 4.1.3, we mention which tokenization method we employ for the experiment. Due to the context window of the embeddings model, we must split larger documents into smaller paragraphs of a few hundred tokens[3] each. Each list of tokens is then encoded into numbers referencing the original tokens, performing math operations on these numbers through multiple passes of the deep neural network to eventually end up with a single vector that represents the list of tokens. We finally put the resulting vectors in a vector database, so that we can access them when a user queries information contained in the representation of these vectors. Besides vector databases, RAG can query other

[3] For the sake of argument, one token roughly equals one word

types of databases through prompt engineering techniques to either call upon prepared queries or create custom SQL or SPARQL queries. We highlight some of them below.

Retrieval Databases The main type of database that is commonly used with RAG is a vector database. They store vector embeddings which are high-dimensional representations of objects such as documents, data points, or items [12]. Assuming that the vector embeddings have been created using the same embeddings model, they are designed in a way that related objects are closer in vector space (i.e. have a smaller distance) and unrelated objects are farther apart in vector space. This makes them efficient in fields where similarity search [12] is desired such as recommendation systems, image and video retrieval, and facial recognition. As vector databases are optimized for dealing with high-dimensional data, they are used to cluster and classify [12] spatial and geographical data as well as proteins and DNA. In short, they are a great way to store unstructured data based on similar pieces of data, but not for (semi-) structured data due to a loss of semantic information carried in the values of the fields (e.g. columns) upon embedding these into a vector.

Besides vector databases, we briefly mention databases containing structured data, because they are important within a wider context. We do not consider them for the context of the thesis, but they are future work for being able to connect enterprise data to RAG. One such example of structured databases is graph databases [51]. Unlike vector databases, there is no direct semantic comparison between the user's prompt and the text represented in these documents. Instead, it is necessary to compose a query according to the language specification of the database query language (DQL) [153]. Firstly, to achieve this, the author creates a template with exposed parameters to ensure that the composed query is accurate in the DQL. Secondly, they extract entities from the prompt to fill in the template. Lastly, the results from the execution of the query will need to be filtered for the most similar entries by some arbitrary criteria.

Moreover, these steps are similar when connecting RAG to a relational database [59]. This demonstrates that it is feasible to employ RAG with structured data but to enforce security and specificity, it is highly recommended to expose query templates with strongly-typed parameters to ensure to avoid SQL injection attacks and to ensure only sound queries are being executed.

Nonetheless, as this thesis deals with unstructured data contained in PDFs for context, we only regard vector databases when discussing retrieval techniques and optimizations.

2.2.4. Retrieval Techniques

In this section, we distinguish between techniques and optimizations. Techniques comprise retrieval algorithms that are combined with LLMs to make up RAG. Optimizations include re-ranking strategies and changing parameters for more efficient retrieval.

We distinguish between two main techniques for retrieving vectors from a vector store. The first technique is **cosine similarity (CoSim)**. CoSim works by embedding the prompt template (see Listing 2.1) using the embedding model and comparing this embedding to the existing vectors in the vector store.

Performing this calculation for each pair, results in a list of normalized scores (i.e. with values between zero and one). Subsequently, we pick the top-k vectors with the highest scores. Hereafter, we retrieve the original tokens from the text of each vector from V_k and put these in the `{context}` field, so that the generator can provide an answer.

The second technique is **maximal marginal relevance (MMR)** [22, 41] that embeds **CoSim** to find vectors that are either most similar to the query vector or an existing vector in the vector store. The diversity parameter λ defines how much weight is assigned to each of the two possibilities. As such, MMR is considered as an augmentation on top of cosine similarity, since it requires a similarity metric to work. The key difference, though, is how the vectors are ranked between both methods. Carbonell and Goldstein [1998] define the diversity parameter λ as:

> "MMR computes incrementally the standard relevance-ranked list when the parameter λ=1, and computes a maximal diversity ranking among the documents in [V_k] when λ=0." (Carbonell and Goldstein, 1998)

Having discussed some retrieval techniques such as CoSim and MMR, optimizing the aspects around these techniques is equally important. We acknowledge that none of the optimizations described in the next sections were taken into account when designing the proposed RAG system (outlined in chapter 4) because an important survey paper [46] was published after the literature phase of the thesis was completed. In the context of Gao et al. [2024], we consider the RAG system to be naive-RAG with little optimizations and foresee the future work of this thesis to include these new strategies. In subsection 2.2.5, we outline these optimizations.

2.2.5. Retrieval Optimizations

Tuning optimizations turn naive-RAG into advanced- or even modular-RAG systems [46] [4]. Like Gao et al. [2024], we consider three stages: **pre-retrieval**, **retrieval** and **post-retrieval** optimizations.

For advanced RAG, pre-retrieval optimizations involve improving the indexing and the original user query. Indexing improvements include strategies such as enhancing data granularity, optimizing index structures, adding metadata, alignment optimization of the user query, and mixed retrieval. Other methods for optimizing the original query include query transformation and expansion, aimed at clarifying and enhancing the original query for the retrieval task. Post-retrieval optimizations consist of reranking retrieved chunks and compressing context to efficiently utilize the available context window of an LLM. In the context of modular RAG, we categorize optimizations between pre-retrieval, in-retrieval, and post-retrieval.

Pre-retrieval

One important pre-retrieval optimization involves selecting an optimal granularity level at retrieval time to enhance retrieval performance and downstream tasks in dense retrievers. The granularity levels can range from the smallest, at the **token** level, to the largest, at the **document** level [46]. Improving the indexing strategy directly impacts the retrieval phase's ability to obtain the appropriate context. This includes setting a fitting chunk size where a larger chunk size captures more context which is set off by more noise in the chunk, longer processing time at inference, and an overall higher cost. Conversely, smaller chunk sizes offer less context but entail less noise, quicker processing times at inference, and lower costs. However, utilizing recursive splits and sliding window methods allows to achieve layered retrieval and amalgamate global information present across multiple chunks, despite mid- and between-sentence truncation.

Augmenting these chunks with metadata such as filenames, page numbers, authors, categories, and timestamps provides filter opportunities based on this information. In parallel, the Reverse HyDE method [45] generates metadata in the form of summaries and introduces hypothetical questions to reduce the gap between the query and the answer embedded in the various chunks. The process of generating hypothetical content to enhance RAG systems is referred to as HyDE (Hypothetical Document Embeddings) [9, 105] when creating hypothetical answers for the query, or Reverse HyDE [45] when generating hypothetical queries. Furthermore, incorporating a hierarchical structure for documents accelerates the retrieval process by establishing parent-child relationships with attached chunks, thus aiding the RAG system in finding similar documents based on relationship.

In-retrieval

To optimize the in-retrieval process, there are several available strategies. Firstly, query expansion is expanding a single query (i.e., query directly from the user prompt) into multiple queries to enrich the content of the query. Query expansion can be resolved in a multitude of ways including multi-query to generate multiple queries through prompt engineering. Another way is by generating simpler sub-queries that are more direct and later aggregating the answers similar to a divide-and-conquer approach in algorithms. Additionally, Chain-of-Verification (CoVe) aims to increase reliability [34] by validating each sub-outcome by the LLM.

Secondly, fine-tuning the embedding model can improve the performance in cases where the context significantly deviates from the corpus on which the foundation model was trained. Such contexts include legal practice, healthcare, and the government to a certain degree.

Post-retrieval

For optimization in the post-retrieval phase, certain aspects can be improved upon such as context curation and LLM fine-tuning (i.e., Chat model). Considering context curation, reranking effectively reduces the overall document pool whilst serving a dual purpose as an enhancement and a filter in information

[4] This paper was published when we finished the literature review period for this thesis and thus was not taken into account for the final solution. The proposed solution in chapter 4 can be considered as naive-RAG with some optimizations. However, we decided to include this paper anyway, since it contains useful configurations for RAG

retrieval. As mentioned earlier, retrieving more context also introduces more noise. To counteract this effect without compromising too much on quality, Jiang et al. [2023] employ small language models (SLMs) to detect and remove unimportant tokens. This effectively compresses and refines the context given to the model. Additionally, reducing the overall number of documents improves the accuracy of the model's answers. Ma et al. [2023] proposes a paradigm that embodies the strengths of LLMs and SLMs, the "Filter-Reranker". More specifically, the SLMs serve as filters whereas the LLMs act as reordering agents. Research demonstrates that instructing the LLMs to re-order the challenging samples sourced from SLMs leads to overall significant improvements in information extraction (IE) tasks. Regarding LLM fine-tuning, one can compensate for the lack of domain knowledge by fine-tuning the chosen foundation LLM. Moreover, one gains control over the model's input and output as an added benefit. A potential approach is to align LLM outputs through reinforcement learning with human or retriever preferences. For example, one could manually annotate the responses and provide feedback through reinforcement learning.

2.3. User Task Definition

In this thesis, we motivate and define two types of user tasks that are employed within the user experiment. The first task is summarization which aims to represent one source document in a limited amount of words. The second task is juxtaposition which essentially compares two source documents and outlines their differences and similarities. The documents on which the experiment participants perform tasks will be explained in section 2.4.4. We will further concretize these tasks in section 5.3.

2.3.1. Summarization

With regards to summarization, there are two ways to distinguish summaries: abstractive and extractive summaries [124]. Here, we focus on abstractive summaries as we are employing RAG, which embodies Natural Language Generation (NLG) to respond to prompts as opposed to extractive ones that aim to obtain key sentences from the original document and concatenate them to form these summaries. This task is limited to Single Document Summarization (SDS) [124], where the original input is a single document. This has been researched using LLMs [102, 55, 154] where they process the contents of the document in one or more passes in the context window of the LLM as opposed to using LLMs in combination with an RAG system to aggregate these snippets of context. Other research proposes a retrieval-enhanced framework [4] to generate abstractive summaries where previous summaries are used as extra context to generate new ones.

Therefore, we define summarization as a writing task where one source document is shortened and maintains the most relevant and important points of the source document. In general, a summarization task has this form:

> Summarize document X in no more than Y words and divide the summary into Z paragraphs, each with one theme. The summary is textual, so no bullet points. The summary begins with a title and has sub-headings for each paragraph.

2.3.2. Juxtaposition

The task of juxtaposition has not been explicitly defined in prior research, but it is closely aligned to Query-focused Multi-Document Summarization (QMDS) [124] where the goal is to summarize from multiple sources given a particular query. Adapting SDS solutions into MDS, specifically QMDS, comes with its own set of challenges including limited training data and a higher document redundancy for similar documents [124]. QMDS has been approached from different angles such as Reinforcement Learning (RL) [90, 131] and Unsupervised Learning (UL) [149, 126]. Through meetings with domain experts (discussed in chapter 2.4), we conclude that the ability of such a system to compare documents semantically is a highly desired feature for operations at the Ministry of Finance. In our case, we want to assess zero-shot prompting on an RAG setup that contains two documents and highlight their similarities and differences. We generally consider the following structure for a juxtaposition prompt:

> Juxtapose documents A and B, and describe in no more than Z words the similarities and differences. Put the similarities in one paragraph and the differences in the other paragraph. The juxtaposition is textual in nature, so no bullet points. The juxtaposition begins with a title and has sub-headings for each paragraph.

2.4. Experiment Context at the Ministry of Finance

This section outlines the experiment context that is the Ministry of Finance (MoF) in which this research is conducted. Firstly, we will briefly describe the MoF, its people partitioned by divisions, their main tasks, and how a chatbot – the user's perspective of an RAG setup – impacts their workflow. Secondly, from the context provided in the 'people' section, we will elaborate on the user tasks for each division. Thirdly, there will be a section dedicated to the influence of the context on the proposed RAG setup, to be explained in chapter 4, and onto the experimental setup, to be explained in chapter 5.

2.4.1. Introduction of the Ministry of Finance

The mission of the Ministry of Finance (MoF) is "Werken aan een financieel gezond Nederland" (i.e., Working towards a financially healthy Netherlands) [95]. They achieve this by collaborating with other ministries and partners within their international network to prepare the Netherlands and its economy for the ever faster-changing world of the future.

The MoF employs roughly 35,0000 civil servants [95] out of more than 120,000 civil servants [112] working for Rijksoverheid (i.e., the Government of the Netherlands). There are seven major departments [100] comprised of five Directoraten-Generaal (General Directorates; DG) and two other ones. This enumerates to: DG Belastingdienst (Tax Authorities), DG Douane (Customs), DG Toeslagen (Stipends), DG Fiscale Zaken (Fiscal Affairs), DG Rijksbegroting (Government Budget), Generale Thesaurie (General Treasury) and Secretaris-Generaal-Cluster (General Secretary Cluster; SG-Cluster). SG-Cluster oversees eleven directorates of which we would like to highlight two of them: Auditdienst Rijk (Government Audit Services) and Concerndirectie Informatievoorziening en Openbaarmaking (Corporate Directorate Information and Disclosure Services; **CdIO**). All in all, the Ministry of Finance's core values foster an environment of growth and innovation, and together with its size makes for an ideal breeding ground for testing new technologies such as Retrieval Augmented Generation which can impact hundreds of employees at once.

2.4.2. Selection of Domain Experts

To write this thesis and conduct our user experiment, we set out on a journey to find professionals from multiple divisions who were interested in participating in such a study. This was quite challenging due to the sheer size of this ministry keeping in mind that we – for the sake of making use of the experts – had to create user-tailored tasks for each division to ensure that the quality of the output could be evaluated accordingly. To accommodate these conditions, we limited our search to five of these divisions.

By conducting interviews with several employees throughout the ministry, participating in brainstorming meetings, preparing pitch decks, and presenting the idea of a chatbot assistant that's able to alleviate work pressure caused by menial tasks, we managed to eventually settle on three divisions spread across the ministry: **Sector-IT within Auditdienst Rijk, Beleid [Policy] within CdIO, and Begrotingsbeheer (Budget Management) within DG Rijksbegroting**. For the remainder of this thesis, these groups will be referred to as **ADR/IT, CdIO/Beleid, and DGRB/BBH**. Their interest in optimizing their workload and availability of human and time resources have been the main reasons these divisions were ultimately chosen. The background of these domain experts will be described in section 2.4.3.

2.4.3. Domain Expert Background

For each of the three divisions, we outline the general activities of the domain experts. Subsequently, we highlight one or more main activities where we juxtapose how these are conducted at present and how we foresee that these tasks will be performed when an RAG system is available.

Auditdienst Rijk - Sector IT

Auditdienst Rijk (ADR) is responsible for auditing the government's business by performing financial audits, IT-audits, operational audits, and others. Specifically, we are highlighting Sector IT (IT-sector). Their focus lies on performing IT-audits and performing data analytics to discover anomalies and forecast trends regarding the IT-implementation within Rijksoverheid.

One such use case is that yearly audit reports require a change report on what has changed between the current year and the previous year. This requires auditors to read both documents [7, 8] of at least 20 pages each. Using the proposed RAG-solution, they would save time by having access to a chatbot that can generate a change report in minutes rather than hours. The procedure necessitates storing and indexing the reports in a designated repository. Subsequently, the user can log into the RAG-app, submit a prompt for a change report, adhering to a specific format that juxtaposes two files: A and B. The response generated by the system is then to be copied, subjected to required modifications, and ultimately saved into the file.

This approach reallocates valuable time, permitting a greater focus on data analysis rather than drafting reports.

Concerndirectie Informatievoorziening en Openbaarmaking - Beleid

The Department of Policy, with a specific emphasis on digital and IT-related policy, is structured around four main pillars. These pillars, with the exception of the last one, are developed in collaboration with other departments within the Dutch Government.

The first pillar is concerned with the detailed interpretation and application of parliamentary legislation, such as the European AI Act [37]. The second pillar involves the collaborative development of government-wide policies and the supervision of their implementation at the ministry level. The third pillar involves working with the Ministry of Justice and Security to discuss potential security measures that support the enacted legislation.

The fourth pillar, unique to the Department of Policy, involves the development of internal policies specific to the Ministry of Finance.

Focusing on the second pillar, policy officers often spend significant time extracting pertinent information from documents drafted from motions submitted to the House of Representatives (Tweede Kamer in Dutch). These documents are then summarized into narrower scopes. An example of such a document is the letter [56] submitted by parliament representatives Huffelen, Adriaansens, and Dijkgraaf on the topic of Generative AI, which spans 15 pages excluding attachments.

With the proposed RAG-application, a policy officer can draft this document in less than an hour, provided they carefully evaluate the output of a large language model. This approach shifts the focus from writing to conceptualizing policy from legislation, thereby optimizing the use of time.

DG Rijksbegroting - Begrotingsbeheer

Begrotingsbeheer's main focus is managing the budget of the state (i.e., the Netherlands) and assessing whether the state budget meets all regulations put forward by HAFIR [99]. Within this division, there are legal and financial experts who regularly answer questions from other departments and the public on HAFIR-related matters. Begrotingsbeheer receives on average 200 emails per year related to the state budget of which 80% contain questions that could be described as run-of-the-mill type content. A RAG-driven chatbot could help answer those questions whose answer can be derived from the data. Hereby, effectively reducing 200 emails to 30-40 emails per year that these experts can focus on and spend more time on their main assignment.

2.4.4. User Task Document Selection

This section briefly describes the type of documents that were used for the user experiment described in chapter 5 grouped by division. The actual contents of each user task – summarization or juxtaposition – are discussed in chapter 5.3.

Auditdienst Rijk - Sector IT

Initially, we wanted to utilize yearly audits – mentioned in section 2.4.3 – for creating the user tasks for the IT sector of ADR. However, given the time-consuming amount of roughly 8000 words – 30 minutes of reading – per report, it was infeasible to ask the control participants of this division to manually summarize a report or write a juxtaposition of two reports. As such, together with the domain experts, we collectively decided on four articles from the publicly available Audit Magazine[5] due to their widespread popularity among auditors. Additionally, these articles are around 2000 words – 10 minutes of reading – which was feasible for the experimental setup. From these articles, we summarize two [117, 116] and juxtapose another two [63, 79].

[5] `https://auditmagazine.nl/`

Concerndirectie Informatievoorziening en Openbaarmaking - Beleid

For the policy division, together with experts, we decided to summarize the handout for government organizations using generative AI (i.e., “handreiking voor overheidsorganisaties bij het gebruik van generatieve AI” in Dutch). This handout is – at the time of writing – considered a draft and is scheduled to be published by the end of 2024. We only consider the summarization task due to a lack of available domain experts in this division.

Besides experts’ interest in the summarizing capabilities of an RAG system, they were keen to investigate the capability of such a system in assessing the sentiment and the document author’s stance (e.g. conservative/progressive policy). These factors affect how government organizations such as the MoF are to implement policies passed down by parliament.

DG Rijksbegroting - Begrotingsbeheer

Even though Begrotingsbeheer (i.e., State Budget Management) could benefit from the same RAG setup used for question-answering tasks, utilizing the emails as source documents were unsuitable for the experiment due to the focus on summarization and juxtaposition.

An expert from the Begrotingsbeheer division, however, suggested to juxtapose a fiche on Budgetrecht (Budget law) [98] and Het Grote Begrotingsboek (The Big Budget Book) [138] and summarize Rijksbegrotingsvoorschriften 2024 (Government Budget Regulations 2024) [97]. Ministerie van Financiën [2024] defines fiches as (directly translated from Dutch):

> “[...] short fact sheets on the core topics of accounting laws and regulations such as budget laws, budget law, treasury banking, etc. They are intended to supplement regulations and are included as informational documents.”

The main idea, in this case, is to find similarities and differences regarding budget law according to the fiche drafted by them and The Big Budget Book provided by the Dutch House of Representatives (i.e., Tweede Kamer der Staten-Generaal).

3

Related Work

This chapter outlines related works in the field of generated summary evaluation, human-ai decision-making, user trust, user reliance, and retrieval-augmented generation. Regarding user reliance, we discuss the dynamics as well as a motivator for researching the temporal aspect.

3.1. Evaluating summary quality

Related works [132, 147, 89] investigate the summarizing capabilities in a medical context and another work considers LLM summarization quality [42]. We outline the common evaluation methods of papers to draw inspiration. For automatic evaluation, all papers [132, 147, 42] use BLEU [114], METEOR [11] and ROUGE-L [83] to quantitatively assess the quality of the reference text (i.e., domain-expert generated text) to the LLM-generated text. Regarding human evaluation, Tang et al. [2023] focus on four dimensions: coherence, factual consistency, comprehensiveness, and harmfulness; whereas Veen et al. [2024] center their survey around completeness, correctness and conciseness. Therefore, we consider, in the experimental setup, several aspects of summary quality based on human evaluation, such as coherence, consistency, completeness, correctness (accuracy), and conciseness among others. Moreover, for automatic evaluation, we look at BLEU, METEOR, and ROUGE-L.

3.2. Human-AI decision making

A recent overview of the design space in Human-AI (Artificial Intelligence) decision-making [71] shows that decision-making AI systems have been applied in a variety of fields including Law & Civics, Medicine & Healthcare, and Finance & Business. These studies have delved into decision tasks in the context of classification and regression where they focus on evaluating, understanding, and improving human performance and experience.

The experimental setup found in these studies typically includes human subjects and assesses the impact of the AI assistance compared to the standard experience. A lack of a common assessment framework increases the difficulty of recreating the expected findings in this study. Additionally, using different models, each designed for a different purpose, the interactions that they facilitate are different. In this thesis, we focus on LLMs and the interactions they bring.

3.3. User trust and reliance

In this section, we define what user trust and user reliance are in the context of human-AI systems. User trust requires careful consideration as too little trust (i.e., under-trust) may defeat the purpose of such a system being helpful, but too much trust (i.e., over-trust) prompts danger as people are likely to be less critical of a system that barely resembles cognitive self-reflection. Moreover, user trust is hard to quantify and thus measured using qualitative methods such as surveys and focus groups. User reliance, though, is easier to quantify as one method of measuring is by considering the outputs between an LLM and the user's post-edit of the LLM's output. Furthermore, the difference between these outputs can be inspected through different lenses, such as looking for semantic or syntactic similarity. The next subsections dive into user trust and user reliance, and what we derive from these related works for our user study.

3.3.1. User trust

The examination of human-AI decision-making trends [71], highlights a shift towards emphasizing decision trials rather than the overall user-AI interaction experience. Within this context, the temporal dimension of trust emerges as a crucial element influencing decision-making as users acclimate to the system over time. The current hypothesis, as proposed by Liu, Lai, and Tan [2021], suggests that undertrust is more likely to occur in situations of unfamiliarity. Adequate trust levels are associated with user familiarity and in-distribution data, while overtrust tends to emerge when users are familiar with a subject, but the data is out-of-distribution. This concept is depicted in Figure 3.1. In-distribution data refers to data that is similar to the training data that the model has been trained on. Whereas out-of-distribution refers to data that is not within the scope of the data that the model was trained on. Moreover, some cognitive biases such as the Dunning-Kruger Effect (DKE) manifest themselves over time. As such, DKE can develop to have a powerful effect on users' perception of reliance on human-AI systems [52]. To tackle these biases, one has to account for the time required to make users aware of these biases.

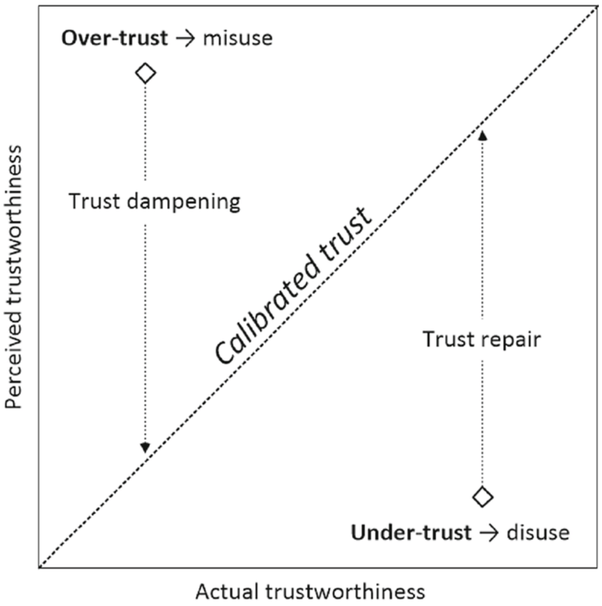


Figure 3.1: Overtrust, undertrust, and calibrated trust as a function of perceived trustworthiness versus actual trustworthiness. Courtesy of De Visser et al. [2020]

This thesis seeks to mitigate overtrust levels among end-users following repeated interactions with LLMs, emphasizing the need to explore temporal trust progression over multiple sessions, a facet currently underexplored in existing research.

Defining and measuring trust

Firstly, to measure a change in temporal trust, we need to define trust. According to a recent literature review of User Trust in AI-enabled Systems [10], there are two definitions [92, 80] commonly used. One provided by Mayer, Davis, and Schoorman [1995]:

> "The willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" (Mayer, Davis, and Schoorman, 1995)

Another by Lee and See [2004]:

> "The belief that an agent will help achieve an individual's goal in a situation characterized by uncertainty and vulnerability." (Lee and See, 2004)

What sets these two definitions apart is that Mayer, Davis, and Schoorman's [1995] definition tends to be used for cases where AI-system personally affects end-users whereas Lee and See's [2004] is employed when this is not the case. As our research focuses on AI systems employed in a large organization, we embrace the definition provided by Lee and See [2004].

Secondly, before measuring trust development over some time, accounting for confounding factors affecting user trust helps to set the baseline. The first key finding is that inherent user characteristics are the dominating factor for initial user trust [10, 163]. Zhou, Luo, and Chen [2020] found that from the big five personality traits [49]: "*Low Openness traits (practical, conventional, prefers routine) had the highest trust, followed by Low Conscientiousness (impulsive, careless, disorganized), Low Extraversion (quiet, reserved, withdrawn), and High Neuroticism (anxious, unhappy, prone to negative emotions)*" [10]. Providing users with a user interface that identifies and informs them of their personality traits helps users become aware of how these traits affect their decision-making upon interacting with an AI system. The second key finding is the acknowledgment that user trust can increase over time with more Human-AI interactions [36, 80]. The acknowledgment serves as a major motivation for researching temporal trust in our research. The third key finding is that the set of factors affecting end-users differs on the context, characteristics of users, and systems [10]. To mitigate this issue, selecting and tailoring features of the system to account for the user group's characteristics as well as emphasizing a selected set of technical and design features that fit the context of the end-user.

Thirdly, there is no common standard for measuring trust. This is demonstrated by the fact that over two-thirds of the included studies of Bach et al. [2022] came up with their surveys rather than used validated tools such as the General Attitudes towards Artificial Intelligence Scale (GAISS) and the Human-Computer Trust Scale (HCTS). In addition to these scales, there is also the Trust In Automation (TIA) questionnaire that aims to capture the overall perception of trust in automated systems as well as interaction with any particular system. Besides questionnaires, other qualitative methods — such as interviews or focus groups — are also slightly less popular methods for measuring trust. These methods, however, are subject to varied interpretations and are considered harder to compare results with. Therefore, in this thesis, we decide to incorporate elements from all three scales.

Fourthly, one considers the ideal situation to be some form of calibrated trust (see figure 3.1) where we would like to avoid the ends of the spectrum: under-trust and over-trust. On the one hand, if under-trust occurs, employing RAG's ability to show citations that the model used to derive its answers increases model trust [122]. On the other hand, if over-trust is at play, the designer of the HAI system has the responsibility to clearly communicate the purposes of the system and build guardrails to avoid misusage [32].

Temporal user trust

Development of user trust in AI over time has been investigated [66, 140] fairly recently. One paper by Cabiddu et al. [2022] put forward some prepositions for factors that affect user trust over time. Firstly, the authors point to social influence [20], because the people around a person define the likelihood of that person continuing to use an algorithm regardless of its intrinsic qualities. They state that:

> "The greater the positive social influence a user perceives over algorithms, the higher the probability of building trust in AI algorithms over time." (Cabiddu et al., 2022)

Secondly, on the topic of the usefulness of algorithms, they mention:

> "The greater the learned usefulness an algorithm demonstrates, the higher the probability of building trust in AI algorithms over time." (Cabiddu et al., 2022)

Lastly, the authors consider familiarity, summarizing their literary study as:

> "The greater the familiarity a user demonstrates with algorithms, the higher the probability of building trust in AI algorithms over time." (Cabiddu et al., 2022)

We mentioned the last quote in section 1.1 because it meaningfully connects to **RQ2**. Although Cabiddu et al. discuss trust and we seek to measure temporal differences in user reliance, their findings show that repeated interactions with AI over time cause noticeable differences in human-AI interaction aspects.

Overall, this thesis embodies trust as part of the efficacy (i.e. **RQ1**), because trust in a (RAG-)system is necessary for it to be considered effective. The next section describes the concept of user reliance in related works and how increased familiarity could affect user reliance.

3.3.2. User reliance

Whereas user trust is considered a subjective matter, we can measure user reliance objectively. User reliance can be defined as "*the degree of agreement between the final user response and the AI suggestion [...]*" [21]. If a system has a sufficient level of quality and is reliable, its users are likely to over-rely on its outputs. For example, Kim et al. [2024] found that purposefully adding uncertainty encouraged users to check their answers due to human-like factors such as impression management, maintaining credibility and avoiding liability, but they note that their intervention does not necessarily avoid over-reliance.

For measuring reliance Schemmer et al. [2022] considers relative positive AI reliance (RAIR) and relative positive self-reliance (RSR). RAIR is the ratio of cases where the user follows the correct AI advice given that their initial decision was incorrect versus these cases combined with cases where the human ignores the correct AI advice. RSR is the number of cases where the user trusts their own initial decision and receives incorrect advice from the AI over those cases and where the user does not trust their decision and follows the incorrect AI advice. The authors found that explainable AI (XAI) had a higher RAIR score and lower RSR score compared to the regular AI model.

In this thesis, we embrace the definition of user reliance given by Cao and Huang [2022], but differ from Schemmer et al.'s [2022] approach. Mainly, their measurement of reliance is binary (i.e. discrete), i.e. whether the user (dis)agrees with the AI's advice instead of a gradual, continuous measurement of agreement. This is an important distinction as user reliance is dependent on the learning task: classification, regression, etc. In the case of generative AI and specifically the context of this thesis, however, we focus on generating text and to what extent this text is usable downstream. Thus, as section 5.2.2 explains in more detail, we measure implicit agreement through the similarity between the output of the user's final response and the AI's generated response.

4 RAG System Implementation

The last chapter discussed related work to understand core concepts around the evaluation of summary quality, user trust, and user reliance. We contextualize these core concepts in chapter 5. However, before introducing the experimental setup in the next chapter, we first introduce our proposed RAG system that supports this thesis.

This chapter outlines our proposed RAG system, named DoRA: Document Retrieval and Analysis (in Dutch: Documenten Raadplegen en Analyseren). DoRA is the proposed chatbot driven by an RAG system with two main purposes for her [1] existence. Firstly, it is a prototype necessary to answer both research questions posed at the beginning of this thesis (see **RQ1** and **RQ2**). Secondly, we propose DoRA as a highly configurable and modular system. This means that environment variables dictate which databases DoRA uses for saving the chat history and embedded documents as well as which models she can use from either OpenAI [109, 107] or from local LLMs saved on the user's filesystem.

4.1. Document Retrieval and Analysis (DoRA)

In this section, we will walk through the RAG system that drives DoRA. She consists of seven components of which there are 2 Large Language Models (LLMs), 3 databases, 1 server, and 1 web app. These components are orchestrated by a Kubernetes [134] cluster. The code for all components, except for the web app, is publicly available on GitHub [129]. The reason for keeping the web app private is explained in section 2.4. We describe the web-app in section 4.2 See Figure 4.1 for a visual overview of these components.

4.1.1. Large language models

Like most chatbots, DoRA needs to understand the user's query. To that end, we supply her with a chat model. For the user study, we pick `GPT-3.5-turbo` [17] as our chat model. We justify our pick due to the model's stability [139], its training on aggregating text, and its lower price [108]. This is in favor of `GPT-4` [111] which has a larger 25K (instead of 16K) and can receive multi-modal input. Moreover, due to hardware resource limitations, we do not use any of the Dutch language models described in section 2.1. That said, DoRA's chat model – in theory – can be any LLM that can process text as input and output text. In practice, however, the most suitable LLMs are those that have been instruction-tuned as well as conversation-tuned so that they understand the user in a similar way that humans ask other humans for help.

Luckily, due to the inclusion of RAG, a model with a smaller context window should perform reasonably since only the relevant parts of each document are included in each prompt.

Additionally, to effectively use RAG, DoRA needs to embed all documents so that she can query them for context. To that end, we provide her with an embedding model. An embedding model can be any transformer model in which the encoding part is only used to go from text input to vector embedding.

[1] Unlike in English, there is no gender-neutral pronoun in Dutch for inanimate objects like "chatbot". "chatbot", is derived from "robot", and is a masculine noun in Dutch. However, as the name DoRA has a feminine ring to it (in European languages), I am referring to DoRA as she. In the UI, she is also being referred to by she/her (zij/haar in Dutch). Moreover, it makes this thesis more fun to read and anthropomorphizing chatbots is also common practice.

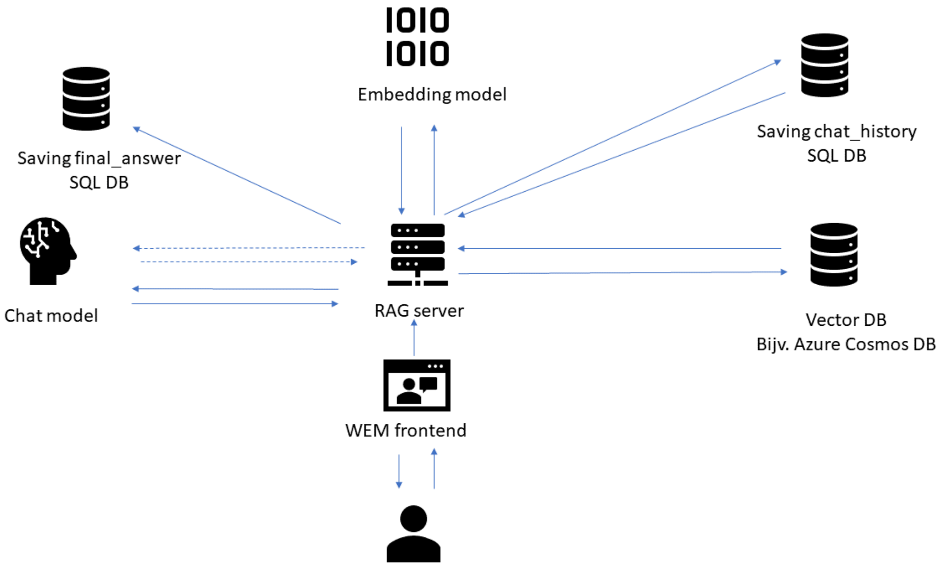


Figure 4.1: DoRA's architecture which centers around a RAG-server that communicates with a chat model, an embedding model, a vector database, and a relational database storing two tables.

For the user study, however, we opted for `text-embedding-ada-002` [107] as it was the only model available from OpenAI at the time of creating DoRA. To obtain the best results from an RAG setup, it is highly recommended that the embedding model and the chat model are similar so that the query vector (obtained by the chat model) is as close as possible to the embedded documents stored in the database. In this way, the retrieved documents are sure to be relatable to the query which allows the chat model to reason about the query augmented prompt.

4.1.2. Databases

Databases are essential to DoRA, because she needs all sorts of data. To reduce overhead, we practically decide to merge all the relational databases into one database with multiple tables to enable easier access. This amounts to one MariaDB [91] database container in the Kubernetes [134] cluster and one ChromaDB [26] server-embedded vector database. We choose this setup to ensure that the relational data is persistent and to quickly have a proof-of-concept to perform the user experiments with by avoiding external overhead by setting up a separate vector DB. Therefore, each database icon in Figure 4.1 represents either a relational table or a vector database. Below, we describe what each table and vector database does.

The first table logs the user's activity needed for analysis. It stores the user's UUID (`session_id`), the start time of the session (`start_time`), the end time of the session (`end_time`), the number of messages exchanged per session (`number_of_messages`), the original answer from the DoRA chatbot (`original_answer`) and the edited answer from the user (`edited_answer`). The last two pieces of data are relevant for capturing the user reliance (**RQ2**) as well as the quality of the answers (**RQ1**).

The second table stores the chat history of the user comprising of two columns: the user's UUID (`session_id`) and a serialized `ChatMessage` [73] object. By including these chat messages, we ensure that LLM has prior context that it can utilize to formulate the full prompt. We set the number of chat messages $m = 5$ to include for context to the LLM because referring to chat history is not highly relevant for summarization tasks, but important enough such that user is allowed to split their query in pieces rather than finding the perfect query at once.

The vector database (vector DB) stores the embedded vectors provided by the embedding model. The embedded vectors represent the documents that the user uploads for added context as described in section 4.1.3 – **Upload Documents**. Moreover, the retriever, described in the aforementioned section, allows for communication to the vector DB to query the most relevant documents based on an embedded prompt vector using either cosine similarity or Maximal Marginal Relevance (MMR). These methods have been described in Chapter 3. For the user study, we opt for MMR due to the tasks of summarization and juxtaposition that favor diversity in the set of documents over the closest match. Carbonell and Goldstein has used MMR been used since 1998 to generate summaries based on document contents [90, 22]. We opt for $\lambda = 0.2$ for the MMR algorithm as we favor the diversity of sources (i.e., chunks with high inter-similarity) over the chunk that has the highest similarity with the given query vector. This follows from the summarization task. Furthermore, we opt for $k = 10$, since that is the largest amount of context we can pass to `GPT-3.5-turbo` without exceeding the token window.

4.1.3. RAG server

The server consists of two main components: Langchain[2] and Flask[3]. Langchain is an LLM orchestration tool that encompasses different libraries to build agents, tools, and chains. Flask is used to quickly create several endpoints that are needed to support RAG. Below, we discuss the two most important endpoints:

Upload Documents

The first endpoint receives a set of uploaded documents to be embedded. Upon reception, we save each document temporarily on the file system of the Docker container in which the RAG server runs. Using Langchain's various document loaders [75], each document has its text and metadata extracted. For the scope of the experiment, we only work with PDF documents. Next, we use the `RecursiveTextSplitter` from Langchain [77] as our tokenization method. It operates by cutting up the extracted texts in chunks of 512 tokens which corresponds to roughly five paragraphs or 380 words of text [110]. Afterwards, the chunks are embedded using the embedding model (see section 4.1.1). We store the resulting vectors in a Chroma vector database [26] in a collection whose key equates to the user ID. By executing all of these steps, the documents are ready to be queried by an embedded prompt.

Send prompts

The second endpoint is to receive new prompts from the user. The prompt and corresponding user ID (in the form of a UUID) are sent to the `ConversationalRetrievalLangchain` (CRL) [74]. This chain recalls the chat history, prepares the `VectorStoreRetriever` [78] with the collection of embedded document vectors and looks up the predefined chat model. Using this information, the CRL enhances the prompt with the retrieved documents as described in section 2. Afterwards, CRL sends the prompt, receives an answer from the model, extracts the sources from this answer, and returns the response to the Flask server. The flask server wraps this response in a JSON object that is sent back.

4.2. Chatting with DoRA: face-to-face

We build DoRA's frontend interface using WEM [152], a no-code solution to quickly build web applications and connect them using REST APIs and relational data sources. We motivate our selection of WEM in section 7.2.

During this development process, we investigated the possibilities within WEM to re-create a similar interface to ChatGPT and Microsoft Copilot to ensure that users feel familiar with the interface. With the current widgets available within WEM, however, this was impossible. We, therefore, attempted to create a plug-in, but it was unsuccessful due to the lack of documentation on that part. Hence, we opted to use pop-up dialogs whenever DoRA sent her network request to inform users of the state of the application. Regardless, we acknowledge that deviating from the norm has affected the user experience and indirectly the qualitative feedback from the users in the study.

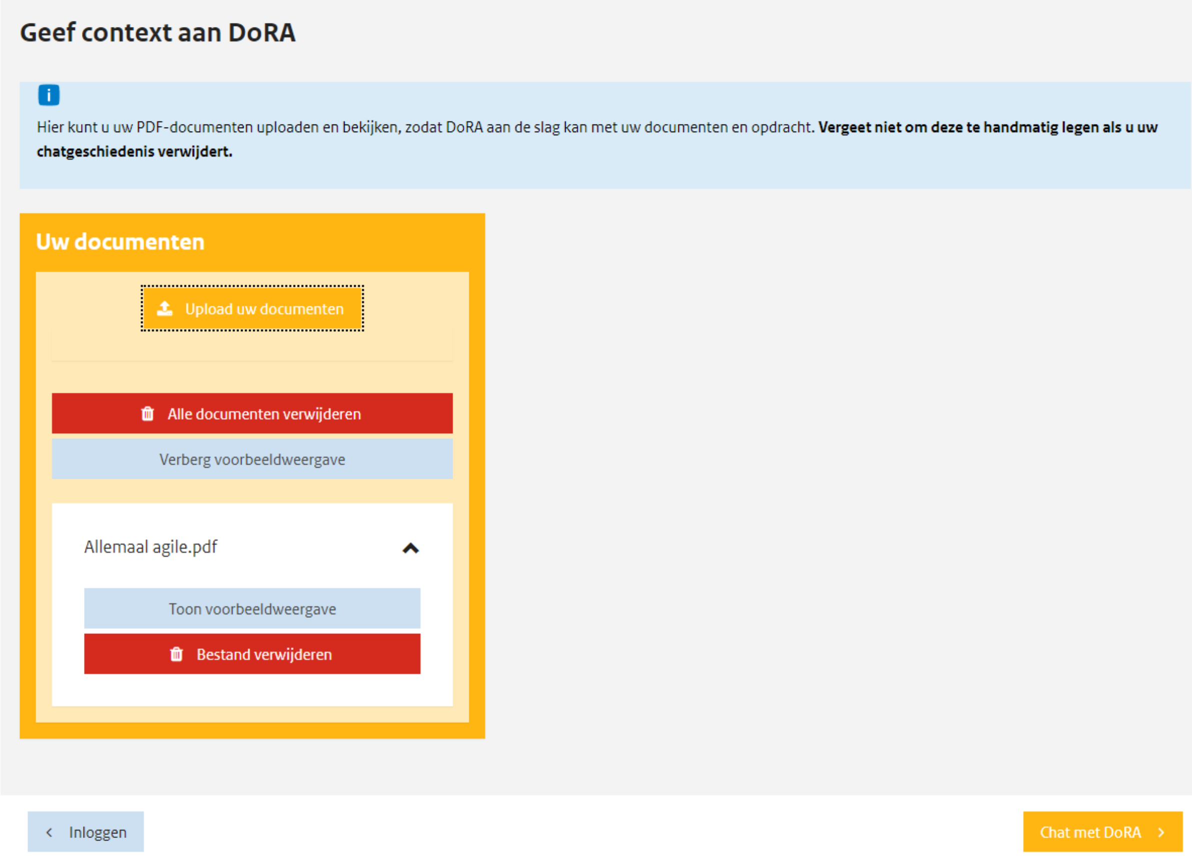
We iterated through some chat interface designs and started with an interface that allowed users to upload documents mid-conversation. That, in contrast to our expectations, seemed to not work as DoRA would forget that more context had been added. We attribute that to Langchain not re-indexing when adding documents later on.

Eventually, we settled on a user experience where users were first met with an upload screen, shown in figure 4.2, where they could upload documents before chatting. Naturally, this is not sustainable long-term as the number of documents grows, document sizes grow and the time-consuming task of

[2] https://python.langchain.com/docs/get_started/introduction

[3] https://flask.palletsprojects.com/en/3.0.x/

extracting text, structure, and metadata from documents becomes ever-more present. So, we circle back to this in section 7.3 as future work.



Geef context aan DoRA

Hier kunt u uw PDF-documenten uploaden en bekijken, zodat DoRA aan de slag kan met uw documenten en opdracht. **Vergeet niet om deze te handmatig legen als u uw chatgeschiedenis verwijdert.**

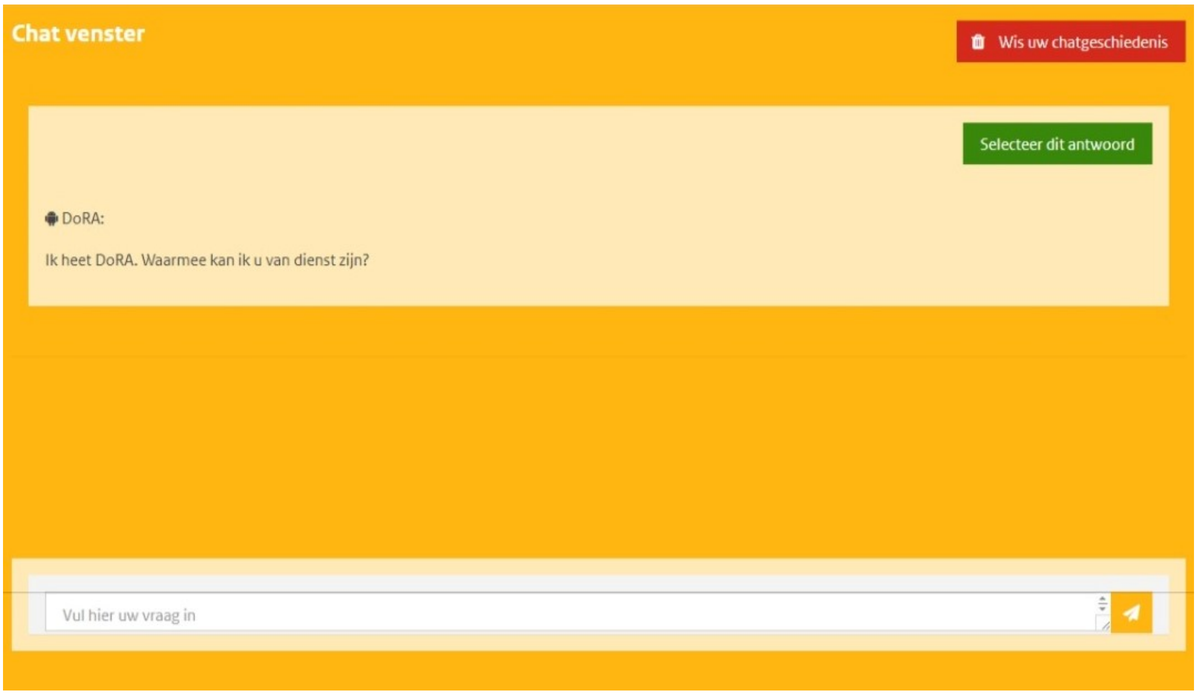
Figure 4.2: DoRA Chatbot – Upload screen. Here, the user uploads their source documents, and, if they are a verifier, they upload the outputs received from the control and experiment group participants.

After uploading their documents, DoRA greets her users with (in Dutch): “Ik heet DoRA. Waarmee kan ik u van dienst zijn?” Meaning: “My name is DoRA. How can I be of service to you?” This formal greeting is a conscious choice as many employees at the Ministry of Finance find this level of formal communication fitting within their organization. Figure 4.3 shows a visual example.

We fast-forward briefly to a user task (section 5.3) assigned to a participant. They may prompt DoRA to help them solve the task during their interaction (section 5.4). Assuming the user has uploaded a document describing the process of state budget management and its justification (i.e., Proces van Begrotingsbeheer en Verantwoording; in Dutch), they can ask it to summarize according to a custom output format. They send the following prompt:

> Summarize the document in 500 words and three paragraphs with each a bold header before the paragraph’s content.

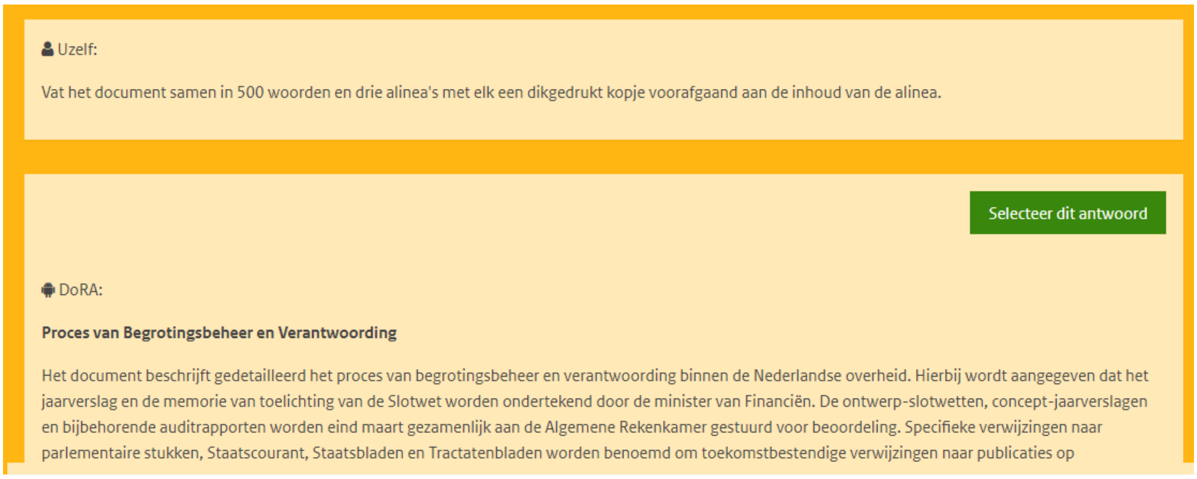
DoRA, in return, responds with a summary that adheres to the above prompt. Figure 4.4 displays this interaction.



DoRA:

Ik heet DoRA. Waarmee kan ik u van dienst zijn?

Figure 4.3: DoRA Chatbot interface – Empty. When the user starts a conversation with DoRA, they are greeted with: “Ik heet DoRA. Waarmee kan ik u van dienst zijn?” Meaning: “My name is DoRA. How can I be of service to you?”



Uzelf:

Vat het document samen in 500 woorden en drie alinea's met elk een dikgedrukt kopje voorafgaand aan de inhoud van de alinea.

DoRA:

Proces van Begrotingsbeheer en Verantwoording

Het document beschrijft gedetailleerd het proces van begrotingsbeheer en verantwoording binnen de Nederlandse overheid. Hierbij wordt aangegeven dat het jaarverslag en de memorie van toelichting van de Slotwet worden ondertekend door de minister van Financiën. De ontwerp-slotwetten, concept-jaarverslagen en bijbehorende auditrapporten worden eind maart gezamenlijk aan de Algemene Rekenkamer gestuurd voor beoordeling. Specifieke verwijzingen naar parlementaire stukken, Staatscourant, Staatsbladen en Tractatenbladen worden benoemd om toekomstbestendige verwijzingen naar publicaties op

Figure 4.4: DoRA Chatbot interface – Prompt: *Summarize the document in 500 words and three paragraphs with each a bold header before the paragraph’s content.* – Response has been cut off.

5 Experimental Setup

This chapter describes the variables, the metrics, and the procedure during the user sessions as well as a description of user profiles. Section 5.1 describes the study design, section 5.2 describes how we measure the dependent variables in the user study within both the between-subject design of **RQ1** and longitudinal design of **RQ2**. Section 5.3 describes the format of the assigned user tasks to the participants. We walk through the experiments in section 5.4.

5.1. Study Design

This section outlines the study design through an explanation of the participant roles within the experiment and how they relate to experimental conditions, two hypotheses accompanied by a conceptual model, a description of the measurements, and how these are affected by confounding factors.

5.1.1. Experiment roles

For the experimental setup, we categorize participants into one of three roles. The first role is that of a human agent who executes the assigned task manually without any AI assistance and represents the **control group**. In this work, we may use human agent, control group, and group A interchangeably. The second role is that of a RAG-assisted agent who uses DoRA to complete their assigned user task and represent the **experimental group**. The third and last role is that of **independent verifier** who rates the task outputs of the previous two roles and is **not** a conditional group in the experiment, but a human-driven means of measuring the difference w.r.t. the dependent variable. Though we aim to use consistent terminology, we may use human agent, control group, and group A interchangeably for the first role. For the second role, we synonymously use RAG-(assisted) agent, experiment group, and group B respectively. For the third, we use the terms independent verifier, verifier group, and group C accordingly.

Regarding the assignment of roles, we ensure that each user task (i.e. combination of domain document + task type) is executed in triples. As such some people may take on more than one task, but do not fulfill more than one role within a triple. The number of task outputs and division of tasks per group are shown in tables 5.1 and 5.2 respectively.

5.1.2. Hypotheses and conceptual model

Before diving into a conceptual model, we first state some hypotheses that help establish the conceptual model for **RQ1** and **RQ2**.

H1: A Naive-RAG system produces similar output compared to a human expert with regards to quality of work (QoW) on text aggregation tasks such as summarization and juxtaposition.

H2: Interacting with a Naive-RAG chat bot increases user reliance.

We conceptually model these in figures 5.1 and 5.3 respectively.

5.1.3. Comparing the outputs between conditions (RQ1)

Following from the illustration in figure 5.1 and hypothesis **H1**, we change the agent that performs the text aggregation tasks as an independent variable (IV) and we observe the quality of work (QoW) resulting from each agent as a dependent variable (DV) in between-subject design. Whereas the subjects refer to human and RAG-assisted agent roles, the independent verifier assesses their QoW. This is illustrated in figure 5.2. We discuss our approach using a variety of metrics to measure the quality of the output in section 5.2.1.

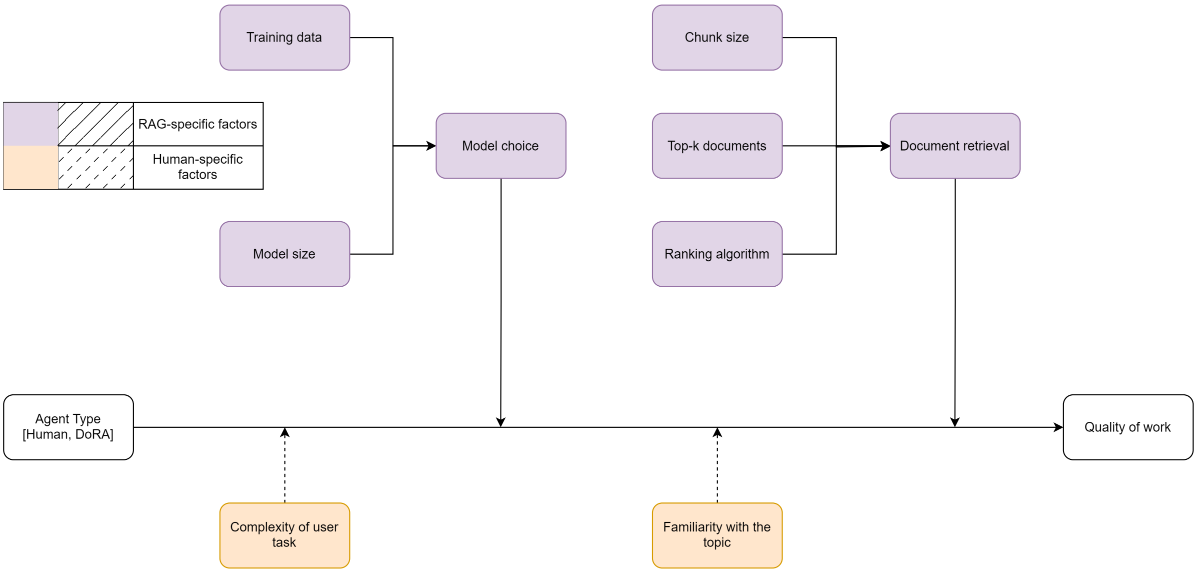


Figure 5.1: The conceptual model for RQ1. The left end shows the experimental conditions as the DV and the right end shows QoW as the IV. The nodes in the middle indicate confounding factors that are either RAG-specific or human-specific. The arrows indicate which factors affect other factors or the outcome of the IV (illustrated as a long horizontal line from left to right).

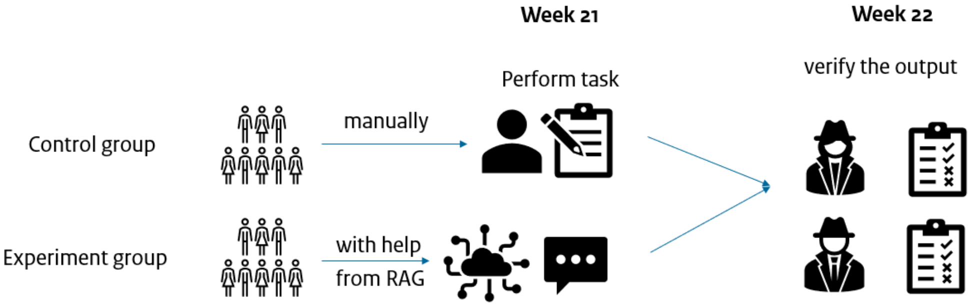


Figure 5.2: Experimental design for RQ1. The control group performs their assigned user task manually without assistance from an RAG system in week 21 of 2024 (i.e., at the time of session II). The experiment group completes their task using DoRA in the same week. The verifier group checks the outputs from both groups in the following week (22).

5.1.4. Capturing user reliance (RQ2)

From figure 5.3, it follows that the user's experience with DoRA affects their user reliance. We solely consider the experiment group for measuring temporal reliance, because they are the only ones interacting with DoRA during their task assignment. As we enable DoRA for use outside of the experiment, we allow users to gain experience with the RAG system. We capture the difference in user reliance between $t = 0$ and $t = 4$ weeks as two snapshots carried by the assumption stated in hypothesis **H2**. An illustration of the process is depicted in figure 5.4.

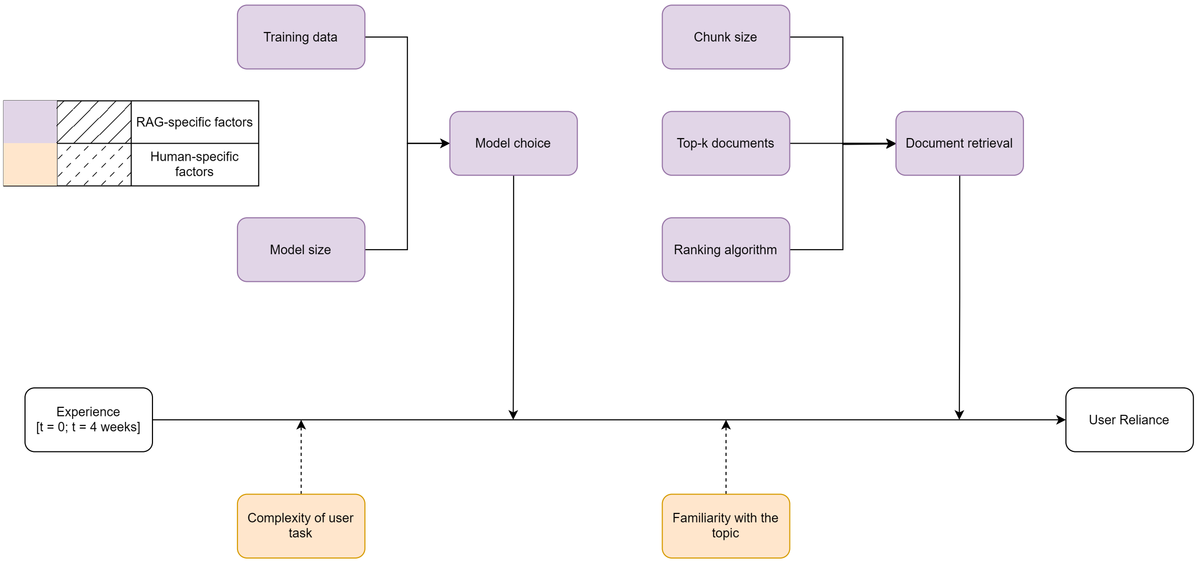


Figure 5.3: The conceptual model for RQ2. The left end shows the experience from the users as a function of time $t = 0 \rightarrow t = 4$ as the DV and the right end shows user reliance as the IV. The nodes in the middle indicate confounding factors that are either RAG-specific or human-specific. The arrows indicate which factors affect other factors or the outcome of the IV (illustrated as a long horizontal line from left to right).

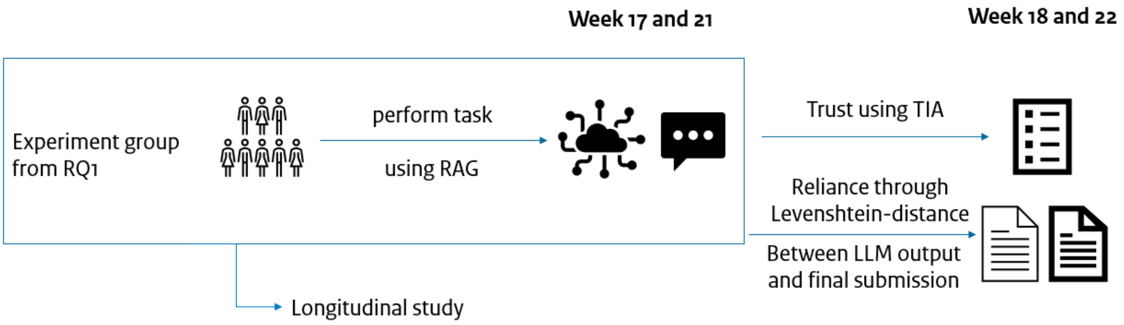

Week 17 and 21
Week 18 and 22
Experiment group from RQ1
perform task
using RAG
Trust using TIA
Reliance through Levenshtein-distance
Between LLM output and final submission
Longitudinal study

Figure 5.4: Experimental design for RQ2. The same experiment group (from RQ1) performs their user task twice during the experiment: in week 17 (i.e., at the time of session I) and week 21 (i.e., session II) of 2024 using DoRA. One week after each session (i.e., weeks 18 and 22), their outputs are analyzed manually and metrics indicative of user reliance are calculated and captured.

5.1.5. Confounding factors

The intermediate ‘boxes’ illustrated in figure 5.1 and 5.3 represent the confounding factors that affect IVs. We can map the confounding factors to the generation-and-retrieval paradigm (explained in section 2.2), where the choice of model and the document retrieval implementation can highly influence the user experience and outputs of an RAG system. Some of the factors that affect the model choice, and implicitly the generation part, are the training data and the model size as they directly affect whether an LLM can be instruction- and dialog-tuned which is imperative for an RAG system. As mentioned in chapter 4, we picked `GPT-3.5-turbo` from OpenAI, due to its accessibility and meeting the requirements of being tuned for any task you provide it.

Another subset of factors that influence the retrieval part are the chunk size, the top-k variable, and the (re-)ranking algorithm. We established the influence of these parameters earlier on in section 2.2 and provided these parameters with values in chapter 4. Nonetheless, we emphasize the effect these parameters have on the outcome below as well as which values they attain. Firstly, the chunk size s determines how large each document becomes from the source files and introduces more noise with larger sizes. As we want to perform summarization tasks, we want to provide the model with as much context as fits within the context window in favor of potential noise generation to minimize the odds of missing a piece of information from a document. Therefore, we set our chunk size $s = 512$ tokens. Secondly, the top-k variable decides how many documents are used to derive the response to an instruction where the token window w caps the number of documents and chunk size: $s \cdot k \leq w$. Along the same line of reasoning, we tried to max out our token window and ended up selecting $k = 10$ document vectors to use for creating the summary. Lastly, different ranking algorithms affect the performance of the RAG system for certain instructions where **CoSim** can increase accuracy for QA-tasks and **MMR** likewise for summarization tasks (discussed in section 2.2.4). As such, we pick **MMR** to re-rank and select $k = 10$ document vectors that encompass the diversity of chunks from the source file.

5.2. Variables and Metrics

This section describes what metrics we use to capture the independent variables in hypotheses **H1** and **H2**. We categorize these by method rather than by hypothesis or research question to avoid repetition of information. The first subsection describes the survey used in detail to capture the quality of the outputs from DoRA and trust in DoRA as an RAG system through a human-evaluated survey. The second subsection discusses several metrics to compare the generator’s initial output with the human-adjusted output to indicate user reliance.

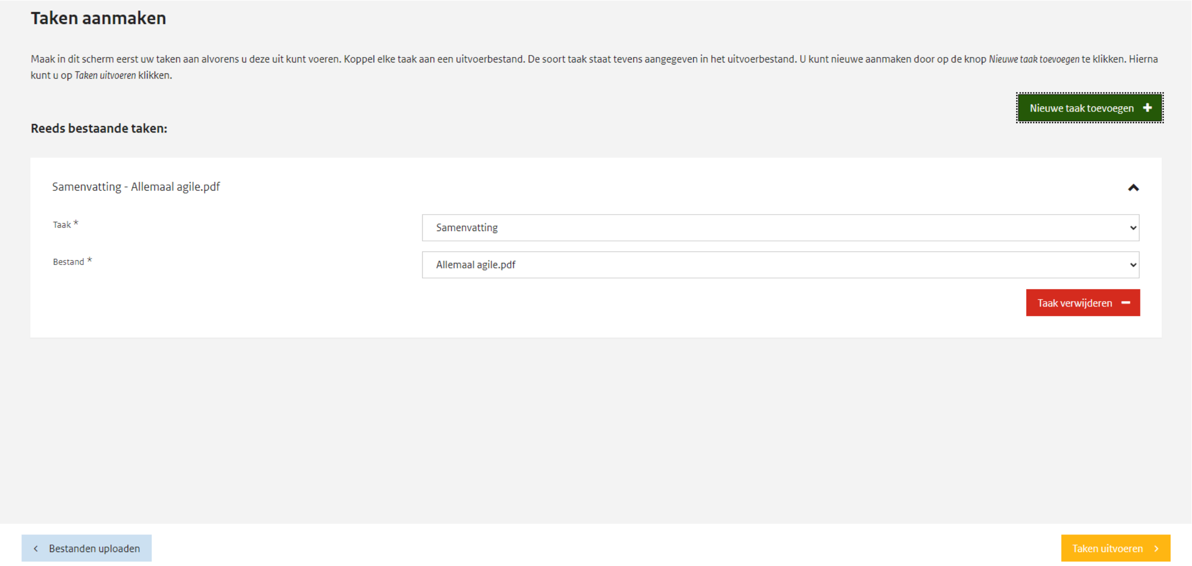
5.2.1. Qualitative assessment

To capture the QoW from DoRA and the user’s trust in DoRA, we make the experiment and the verifier group complete different surveys. The questionnaire for each survey is attached in appendix A. Both surveys tackle 9 dimensions of summary and juxtaposition quality which are clarity, completeness, relevance, accuracy, consistency, structure, conciseness, grammar and spelling, and coherence. On top of that, we measure the overall satisfaction of the output quality. They are assessed as 7-point Likert-scale questions varying from fully `<negative adjective>` to fully `<positive adjective>` (e.g., fully unclear to fully clear) with a free-form field to elaborate on one’s answers. These answer options are defined in appendix B.

The one survey assigned to the experiment group, consisting of 35 questions[1], additionally considers experience and trust in RAG systems as well as the user’s perceived time. Regarding experience and trust in RAG-systems, we partition these into 5 different aspects consisting of: the understanding of RAG-systems, comparing RAG and human expert outcomes, challenges and limitations, user trust and confidence, and the user’s perceived duration.

The other survey assigned to the verifier group, consisting of 20 questions, has one additional question asking for the overall satisfaction of the quality of the output document. This survey is repeated for each document subject to verification. To facilitate the replication of the survey, the verifier group needs to indicate whether they are checking a summary or juxtaposition and which uploaded file it refers to. This is illustrated in figure 5.5.

[1] including both Likert-scale questions and free-form elaborations



Taken aanmaken

Maak in dit scherm eerst uw taken aan alvorens u deze uit kunt voeren. Koppel elke taak aan een uitvoerbestand. De soort taak staat tevens aangegeven in het uitvoerbestand. U kunt nieuwe aanmaken door op de knop *Nieuwe taak toevoegen* te klikken. Hierna kunt u op *Taken uitvoeren* klikken.

Reeds bestaande taken:

Samenvatting - Allemaal agile.pdf

Taak * Samenvatting

Bestand * Allemaal agile.pdf

Figure 5.5: DoRA Chatbot - Task Creation Screen. By creating a task for each output, one verifier can check multiple output texts.

A screenshot of the survey screen is depicted in figure 5.6. We juxtapose the outcomes of both surveys from the verifier, observe the quantitative differences of the Likert scores, and perform a thematic analysis on the collected user feedback from the experiment group. The thematic analysis is discussed in section 6.1. Having discussed the method for gathering data to answer hypothesis **H1**, the next section, 5.2.2, describes the method for hypothesis **H2** on how we observe user reliance.

5.2.2. User reliance assessment

In this section, we describe different metrics as indicators of user reliance. We can divide these metrics into two groups. The first group consists of two metrics related to session metadata, namely: time spent per session and the number of messages *exchanged*[2] per session. The second group consists of metrics related to the similarities between the DoRA's generated output and the final user's output. The indicators in from the second group rely on the assumption that a higher degree of similarities implies a higher user reliance on DoRA. We can further partition these indicators into semantic and syntactical similarity metrics.

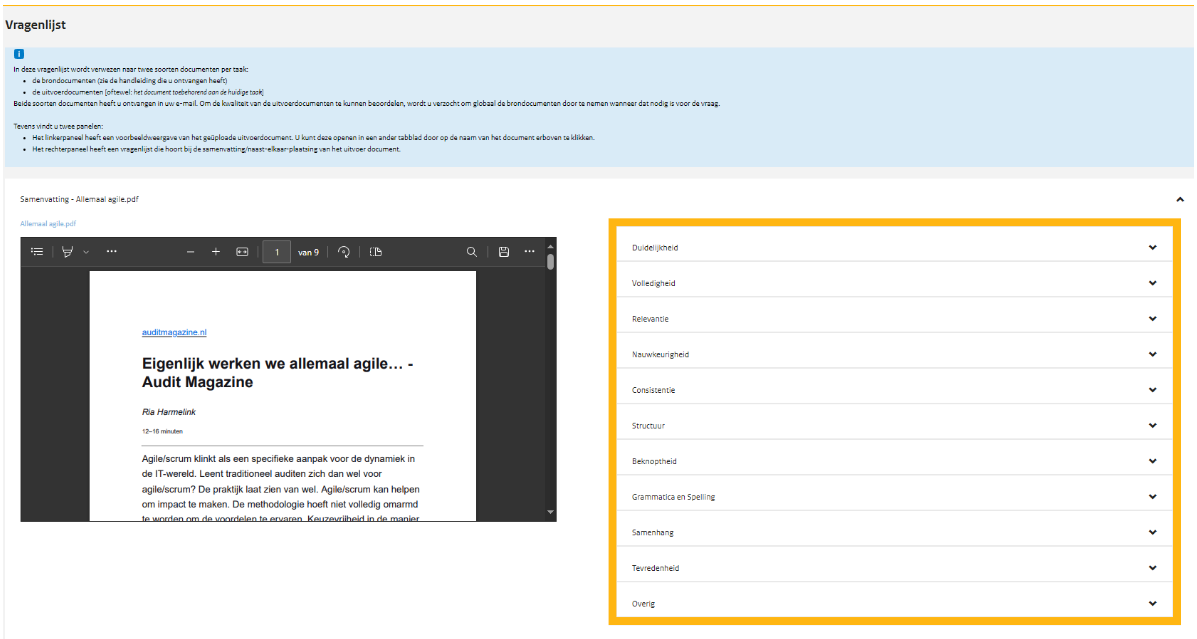
Semantic similarity metrics

To capture semantic similarity metrics, we turn our focus to embedding models [33, 107] and their capability to encode a chunk of text into a vector representation. Vector representations are mappings between chunks of texts and n-dimensional numerical lists that semantically position these chunks in numerical space. Since Word2Vec [93] proposed these representations, they have been useful, because a human-effort intensive ontology is no longer necessary to model the semantic relationship between words as they are approximated using vectors. From these vector representations, we can use cosine similarity, **CoSim**, or neural networks such as BERTScore to capture the vectors' semantic proximity. Whereas **CoSim** expresses this on an embedding level, BERTScore [160] does this on a token level and takes an aggregate to calculate its score.

Syntactic similarity metrics

Syntactic similarity has been captured in prior research with BLEU [114] and ROUGE [83] metrics. Whereas ROUGE-N considers the overlap of n-grams between two texts and takes into account the F1-score, BLEU focuses on precision only by calculating precision for each N-gram size (i.e., unigrams, bigrams, trigrams, tetragrams). A high BLEU score indicates that the user's modifications are similar to the reference text and imply a stronger user reliance. Likewise, a high ROUGE-N score signifies

[2] This includes replies from DoRA.



Vragenlijst

In deze vragenlijst wordt verwezen naar twee soorten documenten per taak:

- de brondocumenten (zie de handleiding die u ontvangen heeft)
- de uitvoerdocumenten [oftewel: *het document toebehorend aan de huidige taak*]

Beide soorten documenten heeft u ontvangen in uw e-mail. Om de kwaliteit van de uitvoerdocumenten te kunnen beoordelen, wordt u verzocht om globaal de brondocumenten door te nemen wanneer dat nodig is voor de vraag.

Tevens vindt u twee panelen:

- Het linkerpaneel heeft een voorbeeldweergave van het geüploade uitvoerdocument. U kunt deze openen in een ander tabblad door op de naam van het document erboven te klikken.
- Het rechterpaneel heeft een vragenlijst die hoort bij de samenvatting/naast-elkaar-plaatsing van het uitvoer document.

Samenvatting - Allemaal agile.pdf

Allemaal agile.pdf

auditmagazine.nl

Eigenlijk werken we allemaal agile... - Audit Magazine

Ria Harmelink

12–16 minuten

Agile/scrum klinkt als een specifieke aanpak voor de dynamiek in de IT-wereld. Leent traditioneel auditen zich dan wel voor agile/scrum? De praktijk laat zien van wel. Agile/scrum kan helpen om impact te maken. De methodologie hoeft niet volledig omarmd te worden om de voordelen te ervaren. Keuzevrijheid in de manier

Figure 5.6: DoRA Chatbot - Survey Screen: Verifier Group. On the left is a PDF-viewer showing the output from a participant in the control or experiment group and on the right are the questions categorized by dimensional quality.

that the user has kept the structure and overall content the same and points to a deeper reliance on the LLM's output.

In addition to NLP metrics, we include **Levenshtein distance** [104] as a metric to assess the edit distance between the two texts. With a smaller distance, this means that the user edited fewer characters. The reason for using this metric is that we can underline what may have caused a higher BLEU or ROUGE-N score. Namely, if the edited text has a low distance (e.g., ≈10 characters), then this immediately explains the BLEU and ROUGE-N score being close to 1, as little edits to the text do not change n-gram-related scores by much.

This section has explained the metrics involved with evaluating summary quality and measuring user reliance. Section 5.3 enumerates the user tasks employed as an input for the output on which these metrics are then applied.

5.3. Examples of user tasks

The control group and experiment group received the same execution task to provide to the LLM. The execution task could be used as a starting prompt for the experiment group, but this was only recommended in passing and not at all required. We highlight the task templates for each working division below.

Summarization Task - General:

> Summarize the document <X.pdf> in no fewer than 500 words and partition the summary in four paragraphs with each one theme. The summary is of textual nature, so no bullet points. The summary starts with a title and has sub-headers for each paragraph.

Juxtaposition Task - General:

> Juxtapose <X.pdf> and <Y.pdf> and describe in no fewer than 500 words the similarities and differences. Put the similarities in one paragraph and the differences in the other paragraph. The juxtaposition is of textual nature, so no bullet points. The juxtaposition starts with a title and has sub-headers for each paragraph.

Summarization Task - Beleid:

> Summarize <X.pdf> in no fewer than 500 words. Of these 500 words, use approximately 100 words to express the general sentiment and 100 words to indicate the attitude (e.g.

> conservative/progressive). Use the remaining 300 words to summarize the contents of the document. Thus, there are three paragraphs with each a header.

Verifier group:

> You have received n output files. You are asked to upload these output files into the application and create a (virtual) task, summary and/or juxtaposition, for each file. Subsequently, you will rate these according to a survey. You can use the source files to compare the texts (from the output files) with the sources that they are based on.

The interface for creating these virtual tasks is shown in figure 5.5. Table 5.1 shows the number of task outputs for both summarization and juxtaposition categorized by the three participant groups. Having defined these user tasks, the next section outlines the procedure for the experiment.

Table 5.1: Factorial design: Number of Task Outputs Categorized by Participant Group

$\#(\textbf{task output}) = 18$	**Summarization**	**Juxtaposition**
Control	4	2
Experiment	4	2
Verifiers	4	2

5.4. Procedure

This section presents the undertaken procedure during the user study. Table 5.2 shows how many participants were included in each group and how many participants completed one or two tasks. Before allowing the users to partake in the study, we invited them to join a training session to get familiar with the user interface. During the training, we briefed them on potential questions that they could encounter during the experiment.

Table 5.2: Factorial design: Number of People Working One or Two Tasks Category Participant Group

	$\#(\textbf{people}) = 13$	×**1 task**	×**2 tasks**
Control	4	2	2
Experiment	5	4	1
Verifiers	4	2	2

After these training sessions, we organized two experiment sessions. Due to the varying time schedules of these participants, we opted to conduct the sessions in an asynchronous format through email lists. This format allowed participants to complete their assigned task at their convenience without necessitating a group intervention such as a focus group. The first session, **S1**, only included the experiment group, because **S1** acts as the anchoring point, $t = 0$, for the longitudinal aspect of measuring temporal reliance (**RQ2**) from the experiment group. However, the second session, **S2**, included all three groups and served two purposes. Its first purpose served as another measuring moment, $t = 4$, where we use the data from the experiment group and compare the DV reliance indicators with the ones from $t = 0$ to measure the temporal development of user reliance (see **RQ2**). Its second purpose served as a period in which we collected data from both the control and experiment groups and had their outputs rated by the verifier group to assess the output quality of the summaries (see **RQ1**).

Outline of the procedure

Figure 5.7 illustrates the procedures for the control, experiment, and verifier groups. In all cases, the participants start by receiving an e-mail with a PDF manual (see appendix C) with a task instruction (see section 5.3) together with any supporting documents. The exact set of supporting documents differs depending on the user task, department, and which group they belong to. Between the control and experiment groups, the supporting documents are the same consisting of one or two source documents depending on the user task. That is one document for summarization and two documents for juxtaposition respectively. The verifier group, on the other hand, gets both the source documents and the outputs from the control and experiment groups.

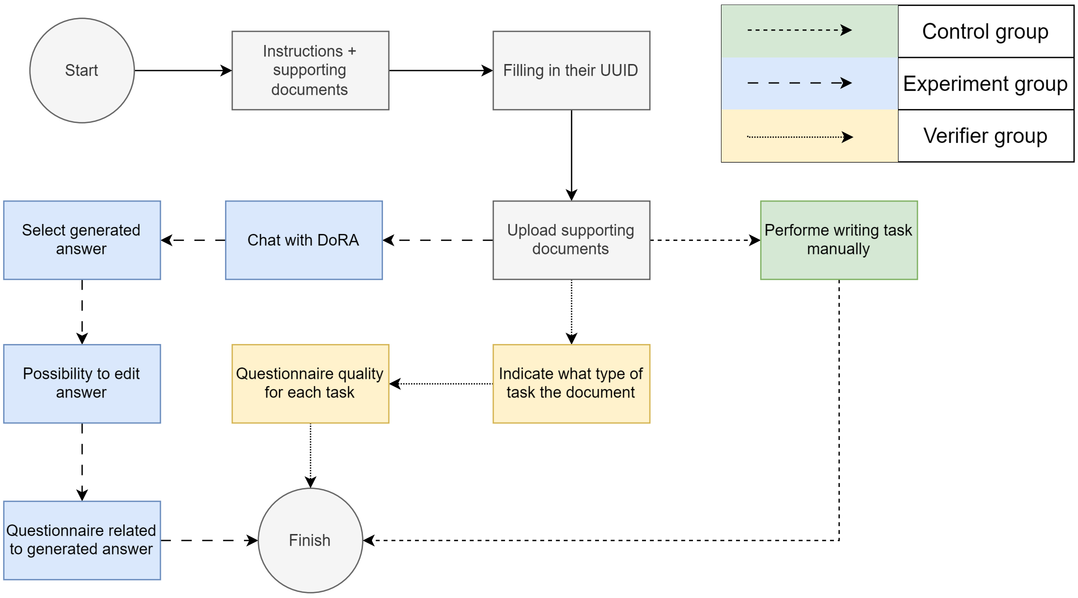

Start
Instructions + supporting documents
Filling in their UUID
Control group
Experiment group
Verifier group
Select generated answer
Chat with DoRA
Upload supporting documents
Performe writing task manually
Possibility to edit answer
Questionnaire quality for each task
Indicate what type of task the document
Questionnaire related to generated answer
Finish

Figure 5.7: The procedures for all participant groups. Each group is made distinct using different colors and arrow types: solid, long-dashed, short-dashed, and dotted

The manual instructs the participants then to click on a link to the web app and fill in their UUIDs. The experiences are made distinct for each participant group by the last letter of their UUID; that is A (control group), B (experiment group), and C (verifier group). All participants are then asked to upload either their source documents (for groups A and B) or their output documents (group C). The next step is distinct for each group.

Firstly, the control group can complete their user task in a text field in the DoRA app. When they are finished, they submit their task and the experiment ends for the control participant.

Secondly, in the experiment group, they are forwarded to a chat screen where they can chat with DoRA for as long as they like. They are tasked to keep chatting with it until they find a response from DoRA that is as close as possible to the task at hand and their expectations. When they have found a desirable response, they select it and are presented with a new screen where they can edit their selected response until it meets their quality standards. After submission of their human-edited answer, they are shown a post-task questionnaire where they are asked to rate the quality of the original response from DoRA as well as some questions about their knowledge, experience, and trust in RAG systems. The experiment participant ends their session on completion of filling out this questionnaire. The experiment group is then expected to complete the same task four weeks later. This is due to the measurement of **H2** of which the participants are not informed.

Thirdly, the verifier group, they are shown a screen where they can indicate if the output documents are summaries or juxtapositions. When each document is assigned to a task, we display the questionnaire where they are asked to rate the quality of the outputs 'blindly'[3]. Upon finishing the survey, that's where the procedure ends for the verifier group.

[3] Despite the quality of GPT-3.5 and RAG, it cannot be avoided that people who use ChatGPT know what outputs are likely to be machine-generated.

6
Results

This section describes the results of our user experiment. We begin this chapter by outlining the results in three parts. The first part is a thematic analysis of the user feedback from the free-form questions in the quality-assessment survey. The second part is an observation of the Likert scores across the nine dimensions (see section 5.2.1) and overall satisfaction. The third part considers the development of user reliance by considering the semantic and syntactical similarity between DoRA's initial output and the user-edited output. We close this chapter by coalescing the results from all the channels and testing these against our defined hypotheses. Here, the first and second parts will be tested against **H1** and the third part will be tested against **H2**.

6.1. Qualitative thematic analysis

To evaluate the quality of the outputs, we set out to perform a thematic analysis of the user feedback from the open questions of the survey. We started by reading all the responses and started coding according to Braun and Clarke's method [2006] of thematic analysis. We consider a combined inductive and deductive approach as there is no research available setting a baseline for common themes about the human evaluation of summaries; or more generally, generated texts. In a deductive approach, we anticipate that user feedback will correspond to the 9 dimensions introduced in section 5.2.1, some of which align with best practices for human evaluation of generated text [141]. Conversely, in an exploratory manner, we analyze user responses to questions that ask them to elaborate on their ratings for each dimension. Extracting codes from the responses required filtering out all the responses where the participants had left no feedback. Some answers contained multiple codes suited codes. From the open coding exercise and merging commonalities, we initially ended up with 20 codes that were not distinct. For these 20 codes, we rank them based on frequency and keep merging. When codes appeared in the same frequency, we judged subjectively based on how related the code was to domain expertise and the experiment tasks, because some codes were related to limitations of the RAG interface rather than the system. After applying this process, we eventually distill these codes into four themes:

1. Prompt Composition
2. Exact Compliance
3. Efficiency over Quality
4. Prioritization of Content

The next subsections will each describe each emerging theme supported by quotes from the participants from the experiment group as well as from the independent evaluators. In consideration of legibility and consistency, we translated the responses of the participants from Dutch to English and added any missing words between brackets and commentary as footnotes.

6.1.1. Prompt composition

Participants demonstrate difficulty composing a prompt that results in the desired answer. The next examples mention both knowledge-related aspects as well as linguistic aspects. Participant P11 states that: **"Posing specific questions to reach the answer is complicated"** in response to the question: *"Which challenges do you face while evaluating the text quality of DoRA?"* This answer can be linked to a knowledge-related aspect where knowing what to ask is considered a challenge especially when dealing with an RAG system that cannot infer the knowledge level of the average user without more data.

When asked *"How do these challenges influence the overall efficacy of DoRA for text aggregation tasks?"* P4, another participant, responds: "**Finding the 'magic' words which cause DoRA to have access**[1]**, makes the process rather frustrating and especially in the case of a short document, a feeling creeps up that summarizing by oneself is faster.**

Participant P5 reflected on *"Elaborate on your rating regarding the clarity of the original answer from DoRA. Provide an example if applicable."*, mentioning, **"It took a while to find the correct formulation of the question which enables it [(DoRA)] to extract the differences and similarities from the texts"**. Moreover, when asked which challenges they faced during the assessment of the text quality, P5 states: **"I had to formulate the task in a slightly different [manner] before the right [output] was achieved."**. As for how this challenge influences the overall efficacy of DoRA, they say the following: **"it is sometimes difficult to create the correct query. This may take [some] time."**

P4's response could be knowledge-related, but may also be due to the initial set-up prompt from DoRA being strict. P5's response can be characterized as having linguistic issues, where DoRA expects instructions to be formulated using instructional language which this user was not aware of. All in all, these examples demonstrate that there is a gap in knowledge between the user's ability to instruct the model to format the output consistently. In chapter 7, we suggest some future work to narrow the gap between domain knowledge and the context from RAG. Assuming the formulation of a prompt is successful, sending it to DoRA leads to exact compliance. The next section describes exact compliance which focuses on how the prompt is interpreted by DoRA as opposed to the task of coming up with a suitable prompt.

6.1.2. Exact compliance

Exact compliance refers to the system doing exactly what is being instructed to do but does not assume much domain knowledge. As Participant P12 states: **"Currently, one must steer towards an answer. Whereas one needs to do that less in question mode [as opposed to juxtaposing or summarizing mode]."**, in response to, *"How can RAG-systems increase the user trust in its outputs, especially compared to human-written content?"*. Furthermore, they state: **"[...] For example, I had to ask separately for four paragraphs which a system may be able to devise on its own."** Participant P9 mentions that **"Getting to the point is a quality of DoRA that humans find more difficult."**, when asked: *"Can you give examples where DoRA's output in terms of quality outshines humans?"* In short, this exact compliance can be considered both desirable and undesirable simultaneously. As such, we briefly describe in Chapter 7 what may have caused this phenomenon and what factors need to be considered before instructing a model in a domain context.

6.1.3. Efficiency over quality

Multiple participants have expressed that they appreciate the gain in perceived efficiency even if DoRA lacks in quality. We emphasize **perceived** here as the efficiency is measured qualitatively and therefore subject to the participant's experience of DoRA instead of a time-bound measurement of task completion. The next paragraphs support this claim with the first paragraph on the ADR/IT department and the latter on the other two departments.

Participant P3 estimates that **"...the efficiency of the usage of DoRA is many times higher than the deployment of domain experts despite still achieving a sufficient quality of the summary."**, prompted by: *"How do you see the trade-offs between content generated by RAG and human-written content in terms of quality?"* Continuing, P3 marks that they did not face any challenges, but when asked: *"How do these challenges influence the overall effectiveness of DoRA for text aggregation tasks?"*, they state that: **"...The deployment of DoRA in my daily routine would save much time**

[1] This refers to the retriever part failing to provide context to the LLM.

and give me the possibility to focus on more important business." A participant from the same department, P4, ponders about the trade-offs as **"RAG-systems win in terms of efficiency. But not yet in terms of quality.** P5, a colleague of P4, agrees: **"For the best answer qualitatively, currently, humans are necessary. In terms of speed, a RAG system is superior. I expect that it will be executed by RAG-systems in the future because this [method] can generate answers more quickly and better."**

Participants P9 and P12 who each belong to different departments than the participants before, when asked about the trade-offs, that efficiency is gained. On this note, P9 remarks **"[It is] definitely time-saving in any case. It guides you to formulate papers. However, the human touch is always necessary, but that is much less effort due to a RAG system.** P12 notes that **"Efficiency is much higher when using DoRA, provided you ask the right questions."** Additionally, they say that: **"...Quality [remains] considerably similar [to humans]. Aside from prioritizing.**

All in all, the perceived and factual quality of the output is lacking compared to humans, but efficiency makes up for it. Notably, the participants were working with publicly available documents and were not able to use the RAG system with their documents at work due to logistical and security constraints. We explain in chapter 7 what the constraints are, but the perceived increase in efficiency was nothing short of surprising. We consider the prioritization of content to play a major role in the perception of output quality. The next section describes this issue in detail.

6.1.4. Prioritization of content

We define prioritization of content as the elements that are present in DoRA's response in an abstractive summary sense. That is finding a subset of elements from the sources to use for summarizing and juxtaposing documents and the order in which they are presented. From a high-level overview, a response may seem correct syntactically and the vocabulary seems fitting, but on closer inspection, the summary appears not faithful to its sources. As participant P4 discusses which factors influence their trust in DoRA, they state: **"DoRA gives a summary that contains falsehoods. By and large, it is correct, but it is precisely the details which make me distrustful."** Participant P12 shared a similar experience; when asked: *"On the contrary, are there any moments where human domain experts perform better than DoRA?"*; **"Prioritizing. [The] subjects for the four paragraphs seem random."** Having tested DoRA with similar prompts as P12, we understood what they meant with 'random'.

In line with exact compliance, DoRA appears to find the greediest way to fulfill the requirements stated in the user task. This method disregards any sensible format guided by the source documents. For example, if the document has three sections, but the prompt asks it to divide it up into four sections, the LLM will still attempt to draft a document even if the outcome is less sensible by considering a random fact that is not related to the other three. Conversely, if the document has more than four sections, it starts to pick chunks of text with the highest scores (assuming higher means more suitable to the prompt). We attribute this to a lack of structure that comes with extracting PDF documents and embedding only the texts with some overlap. We mention a potential method of overcoming this issue in section 7.3 on future work. In conclusion, the omitted context of the document structure as a whole contributes to a random assignment of importance in the output text which is perceived as a decline in quality.

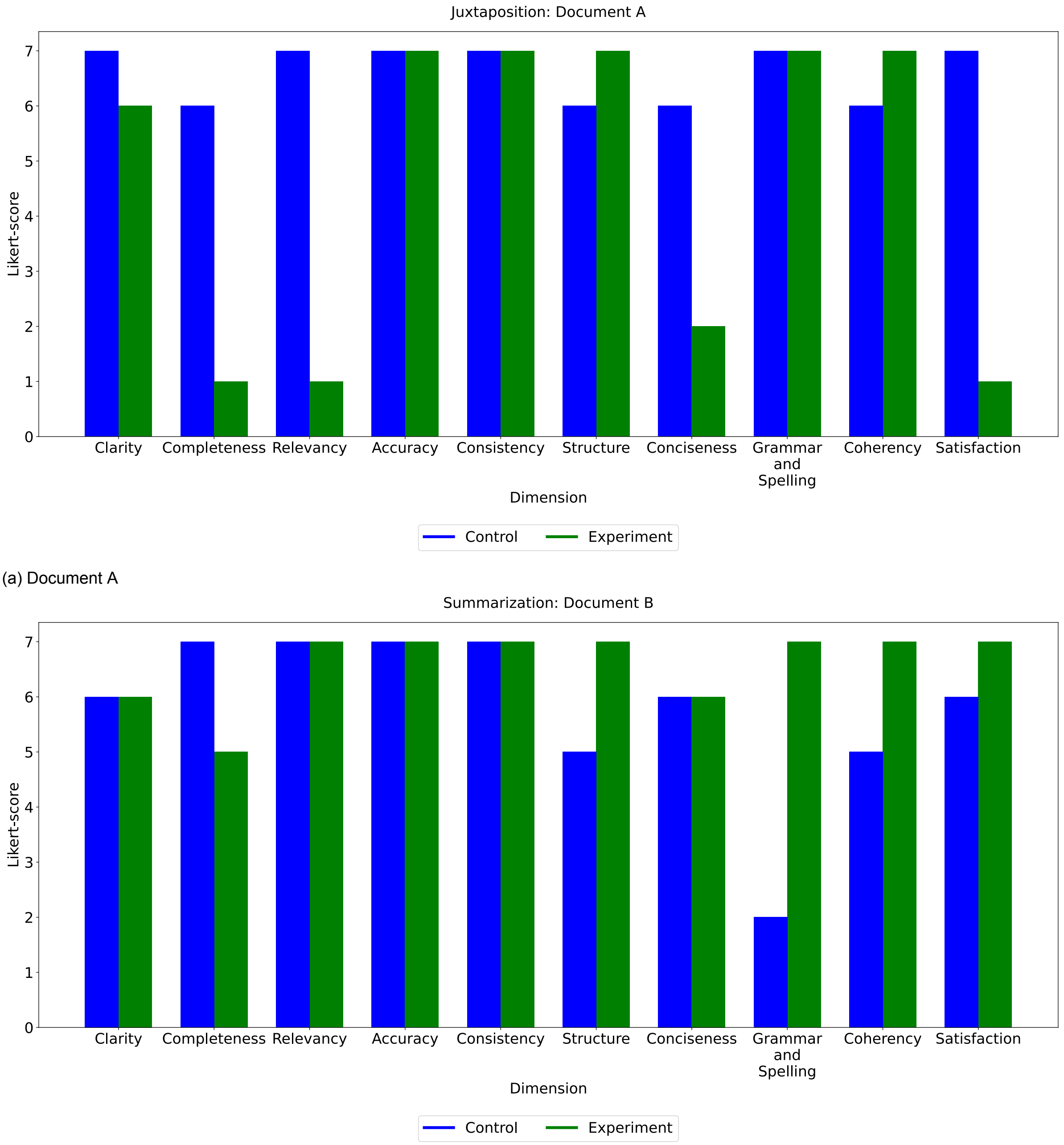
Table 6.1: Supporting table for figure 6.1 outlining the type of document, a description of its content, and which department proposed it.

Document Name	Source Document Description	User Task Category	Department
Document A	1) Article on the adoption of agile practices in auditing 2) Article on the adoption of the agile practice at Jumbo (a Dutch supermarket)	Juxtaposition	ADR/IT IT-auditors
Document B	Article on excessive risk management in auditing	Summarization	ADR/IT IT-auditors
Document C	A handbook on generative AI policy	Summary	CdIO/Beleid (Policy department)
Document D	1) Budget law outline sheet 2) State Budget Handbook Dutch Parliament	Juxtaposition	DGRB/BBH (State Budget Management)
Document E	State Budget Regulations 2024	Summary	DGRB/BBH (State Budget Management)

6.2. Exploratory findings of task output quality

This section will discuss the exploratory findings of the task output quality as captured in the filled-out survey. By observing the 9 dimensions of task output quality and the overall satisfaction rating from the independent evaluator (i.e. the verifier), we establish if there was any significant difference between the control and experiment groups. Each triple of control, experiment, and independent evaluator was involved with the same task and the same document. To refrain from having to remember long document names, we opted to name the documents A, B, C, etc., We provide a convenient mapping in table 6.1.

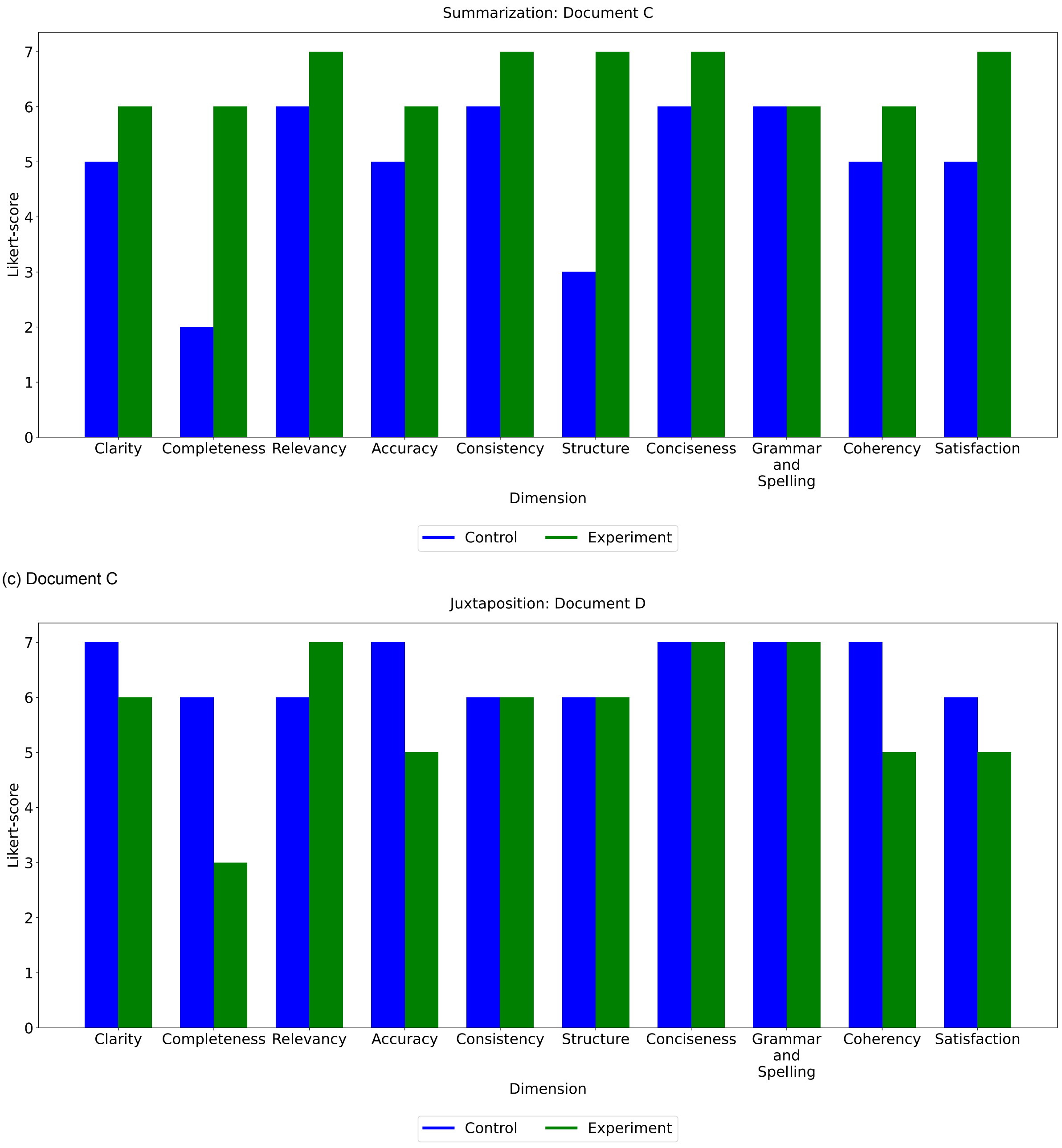
As illustrated by figure 6.1, the DoRA-assisted summary scored higher on the 7-point Likert scale across all dimensions except for Spelling and Grammar, where the score is equal, for document A. For documents B and C, the control group scored equally if not higher than the experiment condition except for conciseness in document C. Generally, across all documents, the difference between the two groups is only one Likert point except for completeness and overall satisfaction of the output. The difference between the two groups varies between 3-4 Likert points indicating that in some cases a complete summary or juxtaposition of the document(s) may depend in part on subjective interpretation (see section 6.1).



(a) Document A

(b) Document B

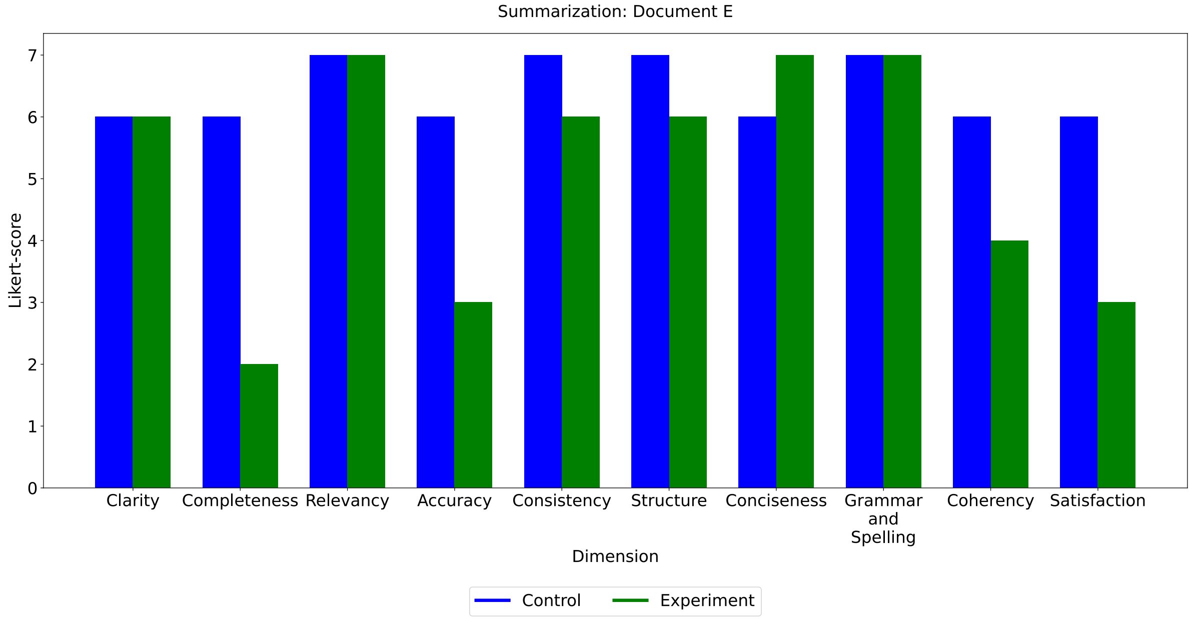
Figure 6.1: Results of the survey across 9 dimensions and overall satisfaction between control and experiment group for documents A and B



(c) Document C

(d) Document D

Figure 6.1: Results of the survey across 9 dimensions and overall satisfaction between control and experiment group for documents C and D



(e) Document E

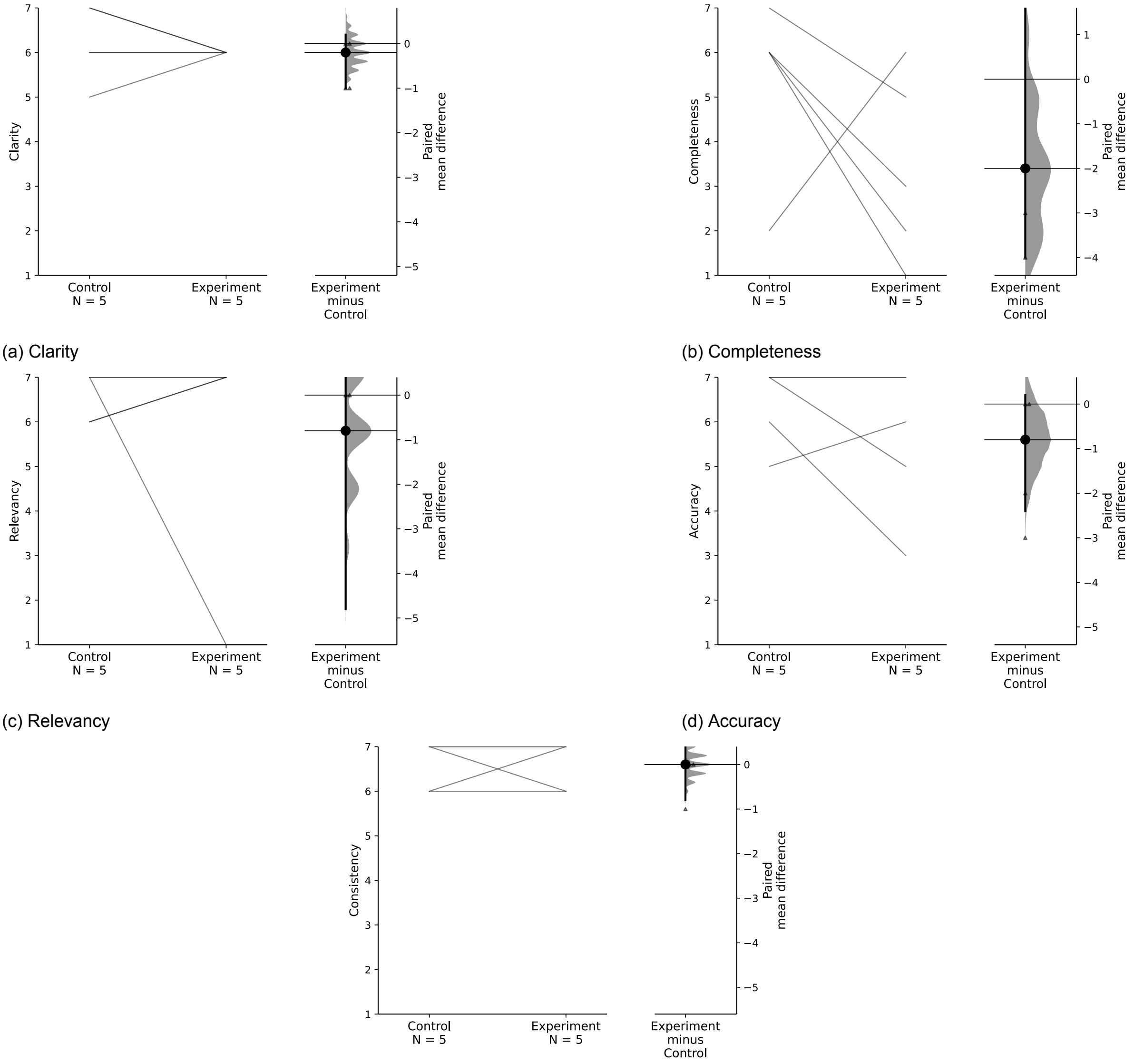
Figure 6.1: Results of the survey across 9 dimensions and overall satisfaction between control and experiment group for document E

Table 6.2: Task output quality results (7-point Likert-scale; see appendix B)

Dimension ($n = 5$)	Control μ	Experiment μ	δ	p-value ($\alpha = 0.05$)	$t(4)^a$
Clarity	6.2	6.0	-0.2	0.62	0.53
Completeness	5.4	3.4	-2.0	0.27	1.26
Relevancy	6.6	5.8	-0.8	0.58	0.61
Accuracy	6.4	5.6	-0.8	0.34	1.09
Consistency	6.6	6.6	0.0	1.0	0.0
Structure	5.4	6.6	1.2	0.24	-1.39
Conciseness	6.2	5.8	-0.4	0.69	0.43
Grammar and Spelling	5.8	6.8	1.0	0.37	-1.0
Coherency	5.8	5.8	0.0	1.0	0.0
Satisfaction	6.0	4.6	-1.4	0.31	1.16

[a] Student's T-test with $n - 1$ degrees of freedom (dependent samples)

Figure 6.2 shows an estimation plot of task output quality across nine dimensions and the overall satisfaction. These plots show the control and experiment groups as paired as the independent evaluators check the outputs of both control and experimental groups based on the same source document(s). Assuming an $\alpha = 0.05$ and a resampling rate of 10K, figure 6.2 shows that the mean differences between the control and experiment groups are not equal to zero. As table 6.2 shows, none of the dimensions are statistically significant.



(a) Clarity

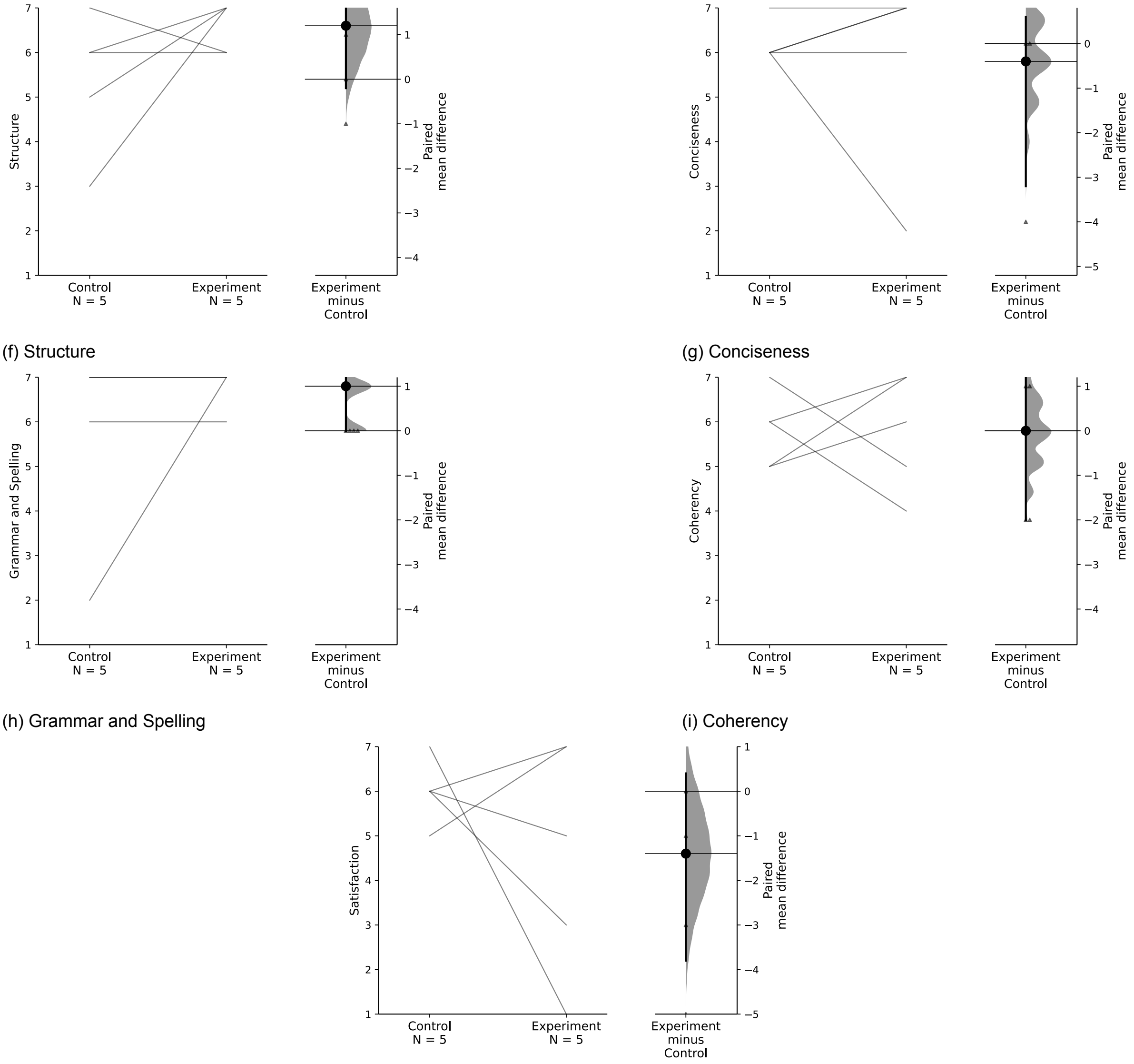
(b) Completeness

(c) Relevancy

(d) Accuracy

(e) Consistency

Figure 6.2: Estimation plots of task output quality between control and experiment group for dimensions: Clarity, Completeness, Relevancy, Accuracy, and Consistency.



(f) Structure

(g) Conciseness

(h) Grammar and Spelling

(i) Coherency

(j) Satisfaction

Figure 6.2: Estimation plots of task output quality between control and experiment group for dimensions: Structure, Conciseness, Grammar and Spelling, and Coherency; and the overall satisfaction.

Table 6.3: User reliance results including metadata metrics (e.g., mins p/session, number of messages exchanged), syntactic and semantic similarity metrics. The bold-faced numbers indicate a statistically significant difference between the averages of each session μ_{S1} and μ_{S2}

Metric $(n = 6)$	μ_{S1}	μ_{S2}	δ	p-value $(\alpha = 0.05)$	$t(5)$ [a]
Minutes per session	40.0	13.33	-26.67	0.0913	2.0863
Number of messages exchanged	13.33	16.0	2.67	0.552	-0.6373
BLEU	0.91	1.0	0.09	**0.0078**	-4.2911
ROUGE-L	0.96	1.0	0.04	**0.0373**	-2.8156
METEOR	0.97	1.0	0.03	0.0886	-2.1106
COSIM	0.99	1.0	0.01	0.0763	-2.2286
LD-ratio[b]	0.06	0.01	-0.05	**0.0353**	2.8626
BERTScore	0.92	1.0	0.08	**0.0001**	-11.0535

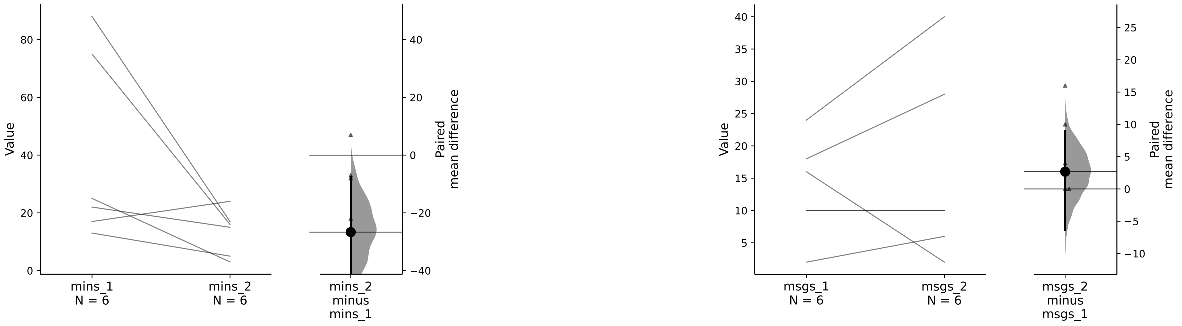
[a] Student's T-test with $n - 1$ degrees of freedom (dependent samples)

[b] Levenshtein distance (LD) ratio: the Levenshtein distance between RAG-output and human-edited output divided by the number of characters in the RAG-output

6.3. User reliance

This section outlines the results for measuring user reliance on the generated output from the LLM in DoRA. Here, we distinguish between syntactic and semantic similarity to assess to what extent the human-edited user task is equivalent to the generated text.

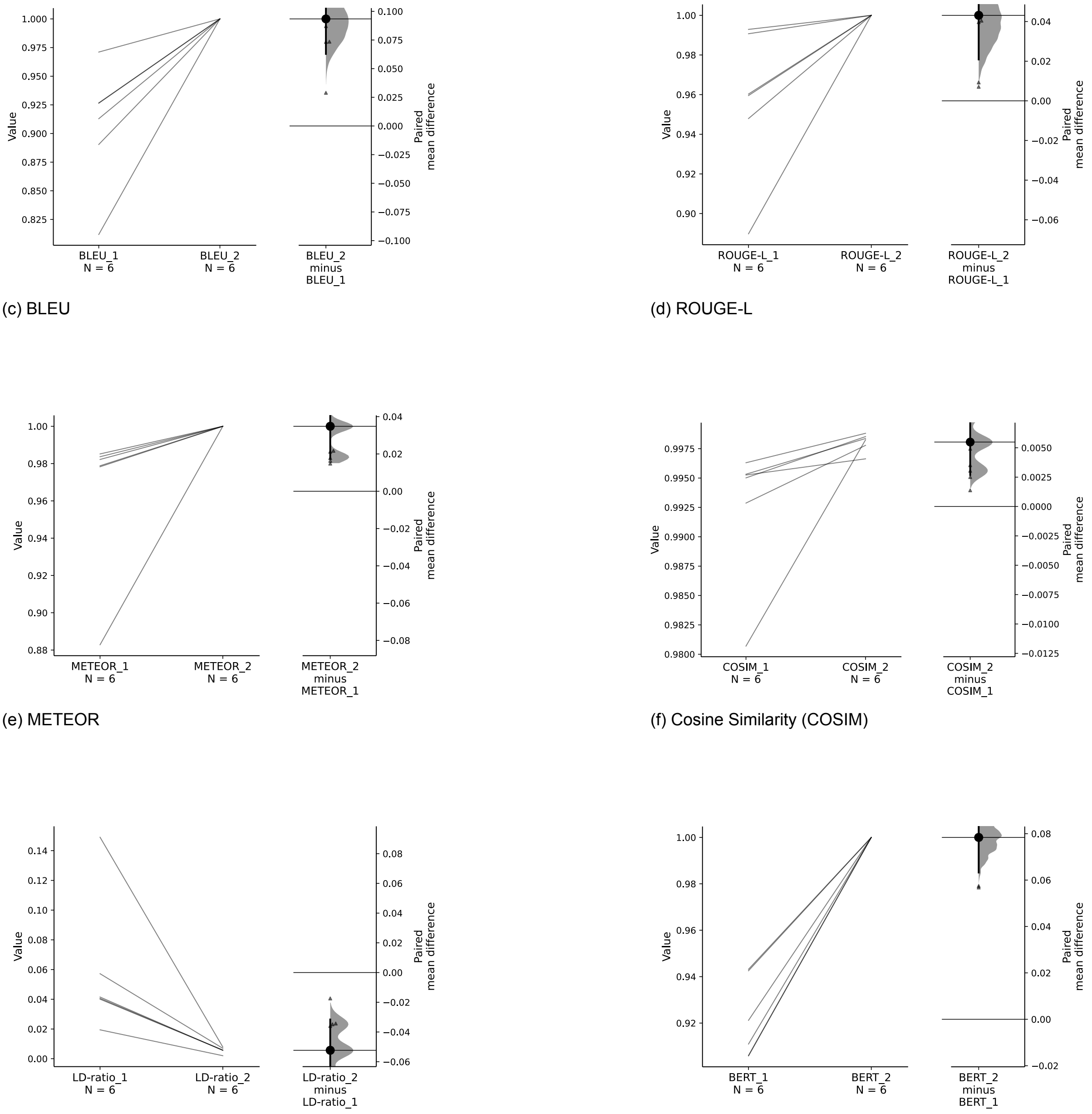
Figure 6.3 shows an estimation plot for user reliance among both types of metrics as well as the time spent in minutes per session and the number of messages exchanged. The suffixes '_1' and '_2' refer to the sessions respectively. Here, we sample about 10K times to perform statistical bootstrapping. From this figure, we infer that multiple semantic and syntactic metrics display a significant difference between sessions I and II. For BLEU ($\delta = 0.09$, $ci = 95\%$, $p = 0.007 < 0.05$, $t(5) = -4.29$), ROUGE-L ($\delta = 0.04$, $ci = 95\%$, $p = 0.037 < 0.05$, $t(5) = -2.82$), BERT ($\delta = 0.08$, $ci = 95\%$, $p = 0.001 < 0.05$, $t(5) = -11.03$), LD-ratio ($\delta = -0.05$, $ci = 95\%$, $p = 0.035 < 0.05$, $t(5) = 2.86$) the difference between session I and session II are significant. LD-ratio refers to the ratio of the Levenshtein distance to the length of the generated text. These results serve as an indicator that the human-edited text and the generated text are highly similar both syntactically as well as semantically in the second session compared to the first session. Moreover, figure 6.3 shows that the LD-ratio in the second session was close to zero. This means that the users barely edited the generated output from DoRA in the second session for the same documents. On top of that, table 6.3 shows the average for the metrics for both sessions, the difference between the sessions, and the result of the student's t-test applied to the data.



(a) Number of minutes spent per session

(b) Number of messages exchanged

Figure 6.3: Estimation plots highlighting the effect of temporal difference between the first ('_1') and the second ('_2') session on user reliance.



(c) BLEU

(d) ROUGE-L

(e) METEOR

(f) Cosine Similarity (COSIM)

(g) Levenshtein distance (LD) ratio

(h) BERTScore

Figure 6.3: Estimation plots highlighting the effect of temporal difference between the first (‘_1‘) and the second (‘_2‘) session on user reliance.

Table 6.4: Acceptance of research hypotheses **H1** (**RQ1**) and **H2** (**RQ2**).

Hypothesis	Accept
H1: A Naive-RAG system produces similar output compared to a human expert with regards to quality of work (QoW) on text aggregation tasks such as summarization and juxtaposition.	No
H2: Interacting with a Naive-RAG chatbot increases user reliance.	Yes

6.4. Summary

As illustrated by table 6.4, we find that for hypothesis **H1**, the quality of DoRA is in some ways comparable to humans, but lacks in some aspects. As such, the result remains inconclusive and we cannot accept this hypothesis. For hypothesis **H2**, we find that user reliance has increased over four weeks between the sessions where the participants have barely edited the responses from the LLM in the second session. Therefore, we reject the null hypothesis (i.e., user reliance is not affected by temporal user interaction) and conclude that this difference is significant.

7 Discussion

This chapter outlines the discussion of the results. We first describe our findings, how these are in line, and their implications. Next, we outline our encountered limitations during the execution of the user experiment. Lastly, we dedicate a section to potential future work where we distinguish between work aiming at the enhancement of the quality of the RAG system and directions on the research of RAG interaction in professional contexts.

7.1. Findings and implications

This section describes our findings, how these relate to earlier works, and what they imply for future research.

Firstly, we find that the difference in output quality between experts using DoRA (experiment group) and experts drafting a document themselves (control group) to complete the task of summarizing one document or juxtaposing multiple documents is statistically insignificant. The insignificance resulting from the magnitude of difference does **not** equate that both groups produce the *same* output quality, but rather that there is no significant trend that suggests that one group performs better than another or that both groups perform equally. Even though the difference is negligible, the thematic analysis demonstrates that there are steps towards improvement. Users of DoRA generally remark that DoRA in its current form already causes a productivity increase, because the RAG system can generate a convincing summary in under a minute. As DoRA scores high on structure and language, through RAG, she's able to create a summary that looks decent at first sight, but the devil is in the details. This is one of the reasons for advocating for more user studies related to RAG contextualized in a text aggregation task such as summarization. While earlier works [132, 147, 89, 42] investigate the summarizing capabilities of LLMs, there remains a notable gap in the literature around generative query-focused multi-document summarization. These studies have primarily focused on assessing the intrinsic summarization abilities of LLMs, evaluating their performance in generating summaries from provided medical documentation as input to these LLMs. However, they have not extended their analysis to the context of retrieval augmented generation (RAG). The omission of integration of retrieval mechanisms with generation processes limits the potential enhancement of the accuracy and relevance of summaries by leveraging external information sources. This thesis contributes by addressing the gap in investigating subjective summary quality through human evaluation with a basic RAG system. Moreover, it underscores the essence of benchmarking (automated) methods of completing text-related user tasks in a field study due to the gain in ecological validity.

Secondly, our results show that users barely edit the RAG-generated output during the second session compared to the first one. Even though the texts already demonstrated high levels of similarity during the first experiment session, we found a significant difference between the sessions. As such, this implies user reliance can be considered a function over time and it can be affected through repeated interaction spaced out over four weeks. Therefore, this thesis has put down the first step to close the gap between user reliance on AI and the accompanying behavior. On top of that, prior literature [52, 10] focuses primarily on user reliance and trust in decision-making AI systems which are mostly comprised of classification models. We argue that user reliance in generative AI models, such as RAG, is more layered, because reliance cannot be simply measured as the binary agreement ratio between system and user, but needs consideration of a way to quantify the extent to which a user takes on the outcome from the system. In prior literature [52, 10], there is a major focus on user reliance and trust in decision-making AI systems. There is a noticeable gap, however, in researching user reliance on LLM and RAG-model outputs. Especially, the impact of over-reliance on these outputs in generic work processes and how these unexamined outputs eventually affect decision-making systems at the higher level. That is LLM-output-driven evidence building to support human (-AI) decision-making process at the higher levels. Therefore, this work makes the first move by measuring user reliance in an evidence-building supporting decision-making process through generative AI.

7.2. Limitations

This section discusses the limitations of this study. We first start with some stakeholder insight that gives context to the next subsections regarding challenges around the user experiment and DoRA as an online tool.

7.2.1. Stakeholder insight

As section 2.4 mentions, the employees at the Ministry of Finance (MoF) have their use-cases that each profit from the increased productivity that an RAG setup provides. The aforementioned departments were already sold by the idea of DoRA before its construction. They have expressed the desire to see this turn into a full-fledged, resources-allocated project. Thus, here we define the scope of the RAG setup with its requirements, expectations, and limitations imposed by the MoF.

The scope, in addition to answering the the research questions posed in chapter 1, is a web-application that allows users to upload their documents and subsequently interact with a chatbot to give it instructions related to their documents. The front end of this application should be accessible to all of the employees of the Ministry of Finance. The users want a chatbot that can understand most queries in Dutch, answer questions given sufficient context, and perform text-related tasks. We acknowledge that these supplemental qualities stem from the initial idea of being pitched as an assistant capable of document retrieval and analysis.

7.2.2. Technical challenges

In the initial stages of app development, we determined that utilizing the resources of the Ministry of Finance (MoF) for hosting the application was the most suitable option. However, this decision came with the requirement to comply with the ministry's stringent security regulations concerning the development of internal applications, without differentiation between apps intended for pilot use, user experiments, or production purposes. Consequently, the implementation of authorization and authentication mechanisms became imperative.

The integration with the Government of the Netherlands' authentication Single-Sign-On (SSO) service required a comprehensive testing process and a significant duration to obtain the necessary credentials for connecting to the API, resulting in delays. Fortunately, the MoF possessed a license for WEM[152], a low-code solution that offered integration with the government's SSO module. Nevertheless, this necessitated a complete redesign of the app's frontend and modifications to the backend, requiring it to accept JSON requests instead of the initial web-form data.

The adoption of WEM as a frontend service precluded the implementation of a Streaming API-like service [6], potentially impacting the user experience compared to other interfaces such as ChatGPT[1] and Microsoft Copilot[2]. Additionally, during the first user experiment session, a front-end bug allowed users to initiate a chat session with DoRA before the documents were indexed in the vector database. This miscalculation in real-time chunking speed meant that the Language Model (LLM) could not retrieve chunks from the documents for necessary context, leading participants to pose multiple questions until the chunking was completed and loaded up. Subsequently, this issue was resolved during the interim between the sessions.

Furthermore, at the outset of the user experiment, the lack of suitable hardware, such as a GPU, both on-premises and in the ministry's cloud environment compelled the usage of an LLM accessible

[1] https://chat.openai.com/

[2] https://copilot.microsoft.com/

through an API, which, at that time, was exclusively provided by OpenAI. This approach raised security concerns, particularly regarding data gathering outside of the organization and OpenAI's APIs not aligning with GDPR standards. Nevertheless, considering the capabilities of OpenAI's models and their ease of integration into RAG-orchestrators like Langchain and LlamaIndex, a decision was made, in collaboration with stakeholders at the Ministry of Finance, to utilize GPT-3.5-turbo, subject to specific conditions outlined in the subsequent section.

In summary, these technical hurdles not only impacted the user interface of DoRA and the user experience but also introduced numerous challenges, as further discussed in the section on ”Experimental Challenges,” which bear relevance to the validity of this research in certain aspects.

7.2.3. Bureaucratic challenges

As we advocated for the integration of a chatbot across the organization as a productivity tool and as a long-term strategy, issues of business continuity began to arise. This was a requirement stipulated by the Ministry of Finance, as they expressed great enthusiasm for GenAI and its capabilities, including RAG. Despite positioning the user experiment as a pilot, it was imperative to provide project-related documentation about data storage, privacy, and post-pilot management (i.e. the 'Ops' in DevOps).

Furthermore, the Ministry of Finance typically outsources the development of any applications that cannot be constructed using WEM to external suppliers, who generally operate with a dedicated team working on a project simultaneously. Completing a pilot (i.e., building DoRA, deploying it, and conducting a user experiment) within four months is considered a noteworthy achievement, given the size of the organization. This is evident in the numerous stakeholder meetings, extensive paperwork, and the coordination of participant schedules required to carry out the experiment.

In contrast, developing a Minimum Viable Product (MVP) typically takes anywhere between 8-16 months at the Ministry of Finance due to the aforementioned processes. All in all, significant effort was necessary to secure approval for deploying DoRA on their intranet, on top of the completion of this thesis.

7.2.4. Configuration challenges

In light of limited resources and the goal of ensuring that a RAG (Retrieval-Augmented Generation) application remains generalizable to the average Ministry of Finance employee, we decided not to customize certain aspects of the model. One of these aspects involved selecting a pre-trained model trained on domain-specific data from the Ministry of Finance or specialized in document summarization. Instead, we opted for a multilingual model, easy to deploy, and trained for instructions and dialogues. This way, the participant's prompts did not need to be formatted specifically for the model to generate reliable outputs.

Another aspect was an in-depth exploration of different prompt templates to identify the most suitable one for summarization and comparison tasks. Initially, we focused on zero-shot prompting to mirror the human usage of chat assistants. Additionally, we aimed to compare the RAG system's output with its default settings, as specifying numerous specific requirements could lead the language model to generate erroneous content. Therefore, we utilized the domain-agnostic default RAG template provided by Langchain. In conclusion, as discussed in section 2.2.5, a potential research avenue is to replicate the user experiment using advanced or modular RAG systems [46]. Further details on this are provided in Section 7.3.

7.2.5. Experimental challenges

The following section discusses the experimental challenges encountered during our study.

The first challenge stemmed from the selection of the LLM vendor, OpenAI, and its associated security implications, which led to several restrictions on hosting the user experiment. Participants were informed that, apart from the designated experiment documents, they were only permitted to use DoRA with publicly labeled Government of the Netherlands documents, and were instructed not to include any personally identifiable information (PII) in their use. Furthermore, it was mandated that the documents used in the experiment had to be classified as public information. This constraint posed a significant obstacle as it limited the potential user base of the RAG system within the daily tasks of interested employees. Additionally, access to the tool was restricted to those who had completed a consent form, leading us to present the RAG tool concept through slides and demos without providing prior access to potential participants.

The second challenge arose from the arduous and time-consuming recruitment of participants, impacting the validity of the experimental setup. Several threats to internal and external validity were identified. Internal validity was influenced by the participant selection process and the social interactions between the control and experiment groups. The participant selection was not completely randomized, as we conducted instruction sessions and meetings to inform participants about their specific roles. Social interaction between groups was unavoidable due to participants from the same department being involved with the same source documents. In terms of external validity, sampling bias was noted as participants were primarily recruited through networking events and meetings, resulting in a non-random sample. Most participants were either tech-savvy or intrigued by the potential of an RAG system, seeking assistance from our innovation department. However, it is important to note that the hypotheses and experimental setup maintained ecological validity, as the results were obtained from a real-life corporate environment using familiar documents.

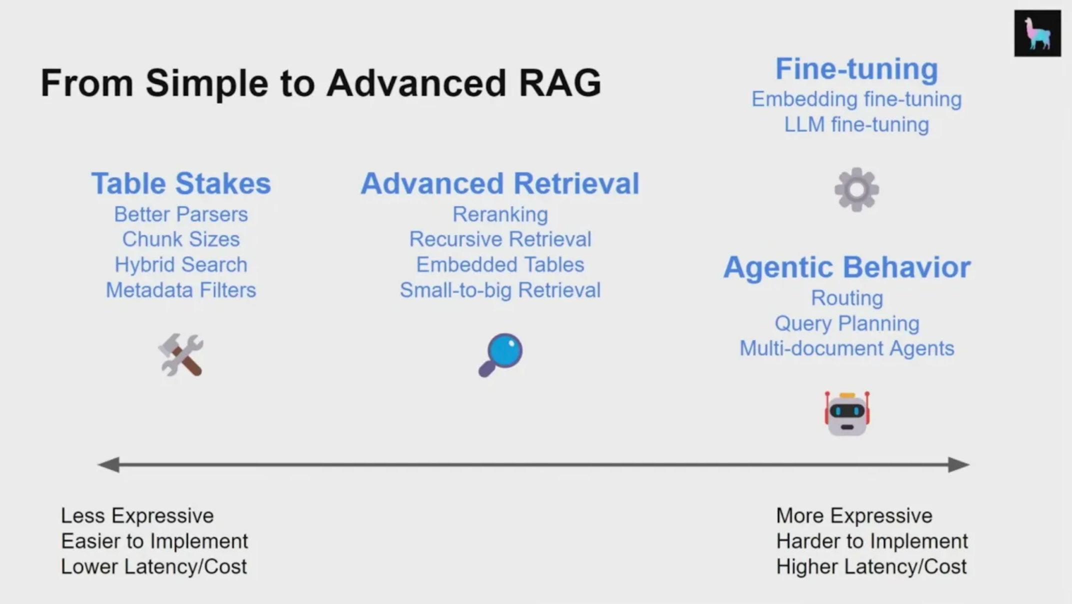
The above resulted in the final challenge: the limited sample size posed a challenge to the statistical validity of our study. To evaluate the paired samples in quality (control vs. experiment group) and user reliance (experiment group S1 vs. S2), a paired Student's t-test was utilized. While this test accounts for a small sample size, with only $n = 6$ samples, cautious consideration of these metrics is required to draw conclusive outcomes. Nevertheless, these findings are presented in Chapter 6 to establish a threshold for rejecting the null hypothesis.

7.3. Future work

This section discusses potential research directions related to RAG quality and user interaction with RAG. Firstly, we briefly outline how RAG quality can be improved using various methods. Secondly, we elaborate on incorporating domain knowledge into RAG.

7.3.1. Improving RAG-quality

As mentioned in section 2.2.5, much more information has become available on RAG since the start of this thesis. We have mentioned some of the improvements in chapter 2 such as the inclusion of relational or graph databases to obtain more grounded answers (section 2.2.3) and adding Chain-of-Verification [34] and the Filter-Reranker paradigm [90] (section 2.2.5). A selection of these improvements is illustrated in figure 7.1.



From Simple to Advanced RAG

Fine-tuning
Embedding fine-tuning
LLM fine-tuning

Table Stakes
Better Parsers
Chunk Sizes
Hybrid Search
Metadata Filters

Advanced Retrieval
Reranking
Recursive Retrieval
Embedded Tables
Small-to-big Retrieval

Agentic Behavior
Routing
Query Planning
Multi-document Agents

Less Expressive
Easier to Implement
Lower Latency/Cost

More Expressive
Harder to Implement
Higher Latency/Cost

Figure 7.1: Overview of aspects in RAG; courtesy of Liu and AI Engineer [2023]

In addition to these, we can employ an agent (section 2.2.1) for each document where it extracts information from the document using a transformed query through an LLM in a divide-and-conquer approach. Here, the head agent instructs multiple sub-agents to make summaries of each document given the original prompt, combine their outputs, and reason over them through Chain-of-Thought (CoT). Moreover, the transformed query is the product of query planning where the idea is to cut the original prompt into multiple prompts in two ways: augmenting the sub-queries through context and recognition of the sub-queries.

Besides agents, adding structure [39] and ontologies [14, 69, 3] increases the output quality for retrieving information for question-answering. In the context of the user tasks from our study, Edge et al. [2024] have constructed a pipeline to perform query-focused summarization using graph databases and RAG.

7.3.2. Incorporating domain knowledge

During the thematic analysis (section 6.1), we found that crafting the right prompt was difficult for most users. We suspect that this is two-fold because some of these users are underestimating the level of specificity needed for an LLM to understand its instruction; for others, it lacked the wider domain knowledge to be used effectively (e.g. terminology used). Fine-tuning large language models is a potential solution to overcome this issue [159, 128]. However, it comes with the overhead of creating a few-shot data set for the LLM to perform in-context learning. Additionally, through query rewriting and creating intention-based agents, one can obtain information from different perspectives (i.e. intentions) to obtain a more nuanced answer [23].

7.3.3. Calibrating user reliance

Our data suggests that the temporal development of user reliance tends towards over-reliance. Whereas under-reliance causes decreased productivity and disuse, over-reliance causes inferior performance due to overtrust of the system's capabilities.

Buçinca, Malaya, and Gajos [2021] conducted a study and found that applying cognitive forcing functions in decision-making AI helps reduce over-reliance. Cognitive forcing functions encompass interventions that prompt users to consciously think at decision-making times. Listed are some examples of cognitive forcing functions. Firstly, one way to force users to think consciously is to ask them to decide on a classification matter before being shown the AI's recommendation [18, 50]. This function works, because, conversely, if the outcome is presented to the user before they decide, the user may be influenced due to anchoring bias. Secondly, another way might be to purposefully delay showing users the AI's outcome [115] as this subconsciously impresses the user into thinking that the AI system is *human* by emulating a careful examination process. Thirdly, giving users autonomy over whether and when they want to see the AI's recommendation avoids the development of aversion towards taking on the AI's advice [40]. Fourthly, despite user preference, creating complex systems with visual difficulties improves participants' understanding and recall of the presented content [57]. Lastly, despite students' perception of having learned more through passive instruction, a paper in education research has found that students conversely learn better with cognitively demanding, active instruction [31].

Though these cognitive forcing functions do not carry over one-to-one to a generative AI dialog system, we can draw some mappings. The first mapping is shifting from the order of showing the AI's recommendation and the user's decision to enabling the user to think about the requirements that the answer needs to fulfill as well as facilitating the validation of the answer to the pre-emptive 'checklist' to allow the user to cognitively assess the degree of agreement with their expectations. The second mapping is trivially implemented by adding a frontend delay before showing the results. The third one is the most difficult one to contextualize in generative AI due to the nature of the motivation that attracts users to generative AI solutions. Most users make use of generative AI tools because they are keen to find out what it will produce for them. The fourth strategy is similar to the second in that they both focus on visual cues. However, this strategy focuses on obfuscation of the content so that extra cognitive effort is required to understand and use the output. The last method comprising active instruction as opposed to passive can be exemplified by building an interrogation mode of sorts that prompts the user at random intervals during the dialog conversation to think of what the ideal output would look like. In concrete terms, the chatbot asks the user to elaborate and motivate more on why they think this might be the right result in line with the behavior of Eliza [151]; considered one of the first AI-driven social chatbots. All in all, we can employ these five strategies to calibrate user reliance to appropriate levels.

8 Conclusion

In this thesis, we explored the efficacy of a RAG system in an organizational setting relative to the organization's experts, focusing on text aggregation tasks such as summarization and juxtaposition. Our investigation led to several key insights and directions; paving the way for further research.

Our first research question (**RQ1**) investigates the extent to which the outputs of a RAG-based LLM system, particularly our Naive-RAG chatbot DoRA (Document Retrieval and Analysis), align with the quality of outputs generated by human experts on text aggregation tasks. Our findings present a mixed bag: while the study revealed that DoRA demonstrates capability in chunking, indexing, and generating document summaries and juxtapositions; the subjective quality as measured against human experts' output — spanning accuracy, relevancy, completeness, and consistency — remains partially unresolved. This lack of resolve primarily stems from the limitations imposed by our small sample size of experts. Despite this, our thematic analysis uncovered relevant insights from user interaction with an RAG-based LLM system, including challenges in prompt composition, exact compliance, perceived efficiency over quality, and content prioritization. These insights suggest that enhancements in prompt engineering and retrieval mechanisms could potentially elevate the system's output quality. This exposes new avenues to refine these aspects, possibly through incorporating advanced RAG techniques to obtain a better understanding and replicate expert-level summarization and juxtaposition tasks.

Turning to our second research question (**RQ2**) concerning how familiarity or experience with an RAG-based LLM system influences user reliance over time, the experimental findings indicate a nuanced shift in user behavior. Users edited the outputs of DoRA less in subsequent sessions, suggesting a growth in acceptance. However, the alterations focused on syntactical adjustments rather than substantive content changes. Although these findings hint at an increasing reliance on the system, the small sample size and the nature of the edits prevent us from drawing definitive conclusions about the development of user reliance. To holistically capture the temporal evolution of user reliance on RAG-based systems, we recommend that future studies include larger and more user groups, over extended periods.

In summary, while this thesis contributes to bridging the knowledge gap between human experts and RAG systems in professional settings, the findings underscore the need for ongoing research. Enhancing system design based on user feedback and experimental parameters will be crucial for advancing our understanding of the potential and limitations of RAG-based LLM systems in attaining human-like efficiency and quality in text processing tasks.

References

[1] Anum Afzal et al. *Towards Optimizing and Evaluating a Retrieval Augmented QA Chatbot using LLMs with Human in the Loop.* 2024. arXiv: `2407.05925 [cs.CL]`. URL: `https://arxiv.org/abs/2407.05925`.

[2] Sayed Fayaz Ahmad et al. "Impact of artificial intelligence on human loss in decision making, laziness and safety in education". In: *Humanities and Social Sciences Communications* 10.1 (June 2023). DOI: `10.1057/s41599-023-01787-8`.

[3] Dean Allemang and Juan Sequeda. *Increasing the LLM Accuracy for Question Answering: Ontologies to the Rescue!* 2024. arXiv: `2405.11706 [cs.AI]`. URL: `https://arxiv.org/abs/2405.11706`.

[4] Chenxin An et al. "RetrievalSum: A Retrieval Enhanced Framework for Abstractive Summarization". In: *CoRR* abs/2109.07943 (2021). arXiv: `2109.07943`. URL: `https://arxiv.org/abs/2109.07943`.

[5] Anthropic. *Claude Anthropic*. Mar. 2023. URL: `https://www.anthropic.com/claude`.

[6] Alejandro AO. *How to use Streaming in LangChain and Streamlit*. Mar. 2024. URL: `https://alejandro-ao.com/how-to-use-streaming-in-langchain-and-streamlit/`.

[7] Auditdienst Rijk | Ministerie van Financiën. *Auditrapport 2021 Ministerie van Sociale Zaken en Werkgelegenheid (XV)*. 2022. URL: `https://www.rijksoverheid.nl/onderwerpen/rijksoverheid/documenten/rapporten/2022/03/15/auditrapport-2021-ministerie-van-sociale-zaken-en-werkgelegenheid-xv`.

[8] Auditdienst Rijk | Ministerie van Financiën. *Auditrapport 2022 Ministerie van Sociale Zaken en Werkgelegenheid (XV)*. 2023. URL: `https://www.rijksoverheid.nl/onderwerpen/rijksoverheid/documenten/rapporten/2023/03/15/auditrapport-2022-ministerie-van-sociale-zaken-en-werkgelegenheid-xv`.

[9] azhar. *Power of Hypothetical Document Embeddings: An In-Depth Exploration of HyDE*. Oct. 2023. URL: `https://medium.com/ai-insights-cobet/power-of-hypothetical-document-embeddings-an-in-depth-exploration-of-hyde-92601a335e5f`.

[10] Tita Alissa Bach et al. "A systematic literature review of user trust in AI-enabled systems: An HCI perspective". In: *International Journal of Human–Computer Interaction* (2022), pp. 1–16. DOI: `10.1080/10447318.2022.2138826`.

[11] Satanjeev Banerjee and Alon Lavie. "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments". In: *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 2005, pp. 65–72. URL: `https://aclanthology.org/W05-0900.pdf`.

[12] Pavan Belgatti. *Vector Databases: A Beginner's Guide!* July 2023. URL: `https://medium.com/data-and-beyond/vector-databases-a-beginners-guide-b050cbbe9ca0`.

[13] Natalie C Benda et al. "Trust in AI: why we should be designing for APPROPRIATE reliance". In: *Journal of the American Medical Informatics Association* 29.1 (2022), pp. 207–212.

[14] Filippo Bianchini et al. "Enhancing Complex Linguistic Tasks Resolution Through Fine-Tuning LLMs, RAG and Knowledge Graphs (Short Paper)". In: *Advanced Information Systems Engineering Workshops*. Ed. by João Paulo A. Almeida, Claudio Di Ciccio, and Christos Kalloniatis. Cham: Springer Nature Switzerland, 2024, pp. 147–155. ISBN: 978-3-031-61003-5.

[15] Virginia Braun and Victoria Clarke. "Using thematic analysis in psychology". In: *Qualitative Research in Psychology* 3 (2006), pp. 77–101. DOI: `10.1191/1478088706qp063oa`.

[16] Andrei Broder. "A taxonomy of web search". In: *SIGIR Forum* 36.2 (Sept. 2002), pp. 3–10. ISSN: 0163-5840. DOI: `10.1145/792550.792552`.

[17] Tom Brown et al. “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[18] Zana Buçinca, Maja Barbara Malaya, and Krzysztof Z. Gajos. “To Trust or to Think: Cognitive Forcing Functions Can Reduce Overreliance on AI in AI-assisted Decision-making”. In: *Proc. ACM Hum.-Comput. Interact.* 5.CSCW1 (Apr. 2021). DOI: 10.1145/3449287. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3449287.

[19] Jason W. Burton, Mari-Klara Stein, and Tina Blegind Jensen. “A systematic review of algorithm aversion in augmented decision making”. In: *Journal of Behavioral Decision Making* 33.2 (Oct. 2019), pp. 220–239. DOI: 10.1002/bdm.2155.

[20] Francesca Cabiddu et al. “Why do users trust algorithms? A review and conceptualization of initial trust and trust over time”. In: *European Management Journal* 40.5 (2022), pp. 685–706. ISSN: 0263-2373. DOI: https://doi.org/10.1016/j.emj.2022.06.001. URL: https://www.sciencedirect.com/science/article/pii/S0263237322000846.

[21] Shiye Cao and Chien-Ming Huang. “Understanding User Reliance on AI in Assisted Decision-Making”. In: *Proceedings of the ACM on Human-Computer Interaction* 6.CSCW2 (Nov. 2022), pp. 1–23. DOI: 10.1145/3555572.

[22] Jaime Carbonell and Jade Goldstein. “The use of MMR, diversity-based reranking for reordering documents and producing summaries”. In: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR ’98. Melbourne, Australia: Association for Computing Machinery, 1998, pp. 335–336. ISBN: 1581130155. DOI: 10.1145/290941.291025.

[23] Guanhua Chen, Wenhan Yu, and Lei Sha. *Unlocking Multi-View Insights in Knowledge-Dense Retrieval-Augmented Generation*. 2024. arXiv: 2404.12879 [cs.CL]. URL: https://arxiv.org/abs/2404.12879.

[24] I-Chun Chern et al. *FacTool: Factuality Detection in Generative AI – A Tool Augmented Framework for Multi-Task and Multi-Domain Scenarios*. 2023. arXiv: 2307.13528 [cs.CL]. URL: https://arxiv.org/abs/2307.13528.

[25] Aakanksha Chowdhery et al. “PaLM: Scaling Language Modeling with Pathways”. In: *Journal of Machine Learning Research* 24.240 (2023), pp. 1–113. URL: http://jmlr.org/papers/v24/22-1144.html.

[26] Chroma. *Langchain | Chroma*. 2024. URL: https://docs.trychroma.com/integrations/langchain.

[27] Kevin Clark et al. “ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators”. In: *CoRR* abs/2003.10555 (2020). arXiv: 2003.10555. URL: https://arxiv.org/abs/2003.10555.

[28] Peter Clark et al. “Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge”. In: *CoRR* abs/1803.05457 (2018). arXiv: 1803.05457. URL: http://arxiv.org/abs/1803.05457.

[29] Databricks. *Retrieval Augmented Generation*. [Online; accessed 2023-12-21]. URL: https://www.databricks.com/glossary/retrieval-augmented-generation-rag.

[30] Ewart J De Visser et al. “Towards a theory of longitudinal trust calibration in human–robot teams”. In: *International journal of social robotics* 12.2 (2020), pp. 459–478.

[31] Louis Deslauriers et al. “Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom”. In: *Proceedings of the National Academy of Sciences* 116.39 (2019), pp. 19251–19257.

[32] Jayati Dev et al. “Building Guardrails in AI Systems with Threat Modeling”. In: *Digit. Gov.: Res. Pract.* (2024). Just Accepted. DOI: 10.1145/3674845. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3674845.

[33] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *CoRR* abs/1810.04805 (2018). arXiv: `1810.04805`. URL: `http://arxiv.org/abs/1810.04805`.

[34] Shehzaad Dhuliawala et al. *Chain-of-Verification Reduces Hallucination in Large Language Models*. 2023. arXiv: `2309.11495 [cs.CL]`.

[35] Darren Edge et al. *From Local to Global: A Graph RAG Approach to Query-Focused Summarization*. 2024. arXiv: `2404.16130 [cs.CL]`. URL: `https://arxiv.org/abs/2404.16130`.

[36] Aaron C Elkins and Douglas C Derrick. "The sound of trust: voice as a measurement of trust during interactions with embodied conversational agents". In: *Group decision and negotiation* 22.5 (2013), pp. 897–913. DOI: `10.1007/s10726-012-9339-x`.

[37] European Parliament. *EU AI Act: first regulation on artificial intelligence*. 2023. URL: `https://www.europarl.europa.eu/topics/en/article/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence`.

[38] Run-Ze Fan et al. *RIGHT: Retrieval-augmented Generation for Mainstream Hashtag Recommendation*. 2023. arXiv: `2312.10466 [cs.CL]`. URL: `https://arxiv.org/abs/2312.10466`.

[39] Masoomali Fatehkia, Ji Kim Lucas, and Sanjay Chawla. *T-RAG: Lessons from the LLM Trenches*. 2024. arXiv: `2402.07483 [cs.AI]`. URL: `https://arxiv.org/abs/2402.07483`.

[40] Gavan J Fitzsimons and Donald R Lehmann. "Reactance to recommendations: When unsolicited advice yields contrary responses". In: *Marketing Science* 23.1 (2004), pp. 82–94.

[41] Jan Frederik Forst, Anastasios Tombros, and Thomas Roelleke. "Less Is More: Maximal Marginal Relevance as a Summarisation Feature". In: *Advances in Information Retrieval Theory*. Ed. by Leif Azzopardi et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 350–353. ISBN: 978-3-642-04417-5. DOI: `10.1007/978-3-642-04417-5_37`.

[42] Bharathi Mohan G et al. "Comparative Evaluation of Large Language Models for Abstractive Summarization". In: *2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. 2024, pp. 59–64. DOI: `10.1109/Confluence60223.2024.10463521`.

[43] Deep Ganguli et al. "Predictability and Surprise in Large Generative Models". In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '22. Seoul, Republic of Korea: Association for Computing Machinery, 2022, pp. 1747–1764. ISBN: 9781450393522. DOI: `10.1145/3531146.3533229`.

[44] Catherine A. Gao et al. "Comparing scientific abstracts generated by ChatGPT to real abstracts with detectors and blinded human reviewers". In: *npj Digital Medicine* 6.1 (Apr. 2023). DOI: `10.1038/s41746-023-00819-6`.

[45] Yunfan Gao. *Modular RAG and RAG Flow: Part I*. Jan. 2024. URL: `https://medium.com/@yufan1602/modular-rag-and-rag-flow-part-%E2%85%B0-e69b32dc13a3`.

[46] Yunfan Gao et al. *Retrieval-Augmented Generation for Large Language Models: A Survey*. 2024. arXiv: `2312.10997 [cs.CL]`.

[47] Google. *Gemini*. Dec. 2023. URL: `https://gemini.google.com/`.

[48] Google. *The WordPiece Algorithm in Open Source BERT*. Retrieved on May 6th 2024. 2018. URL: `https://github.com/google-research/bert/blob/master/tokenization.py#L335-L358`.

[49] Samuel D Gosling, Peter J Rentfrow, and William B Swann. "A very brief measure of the Big-Five personality domains". In: *Journal of Research in Personality* 37.6 (2003), pp. 504–528. ISSN: 0092-6566. DOI: `https://doi.org/10.1016/S0092-6566(03)00046-1`. URL: `https://www.sciencedirect.com/science/article/pii/S0092656603000461`.

[50] Ben Green and Yiling Chen. "The Principles and Limits of Algorithm-in-the-Loop Decision Making". In: *Proc. ACM Hum.-Comput. Interact.* 3.CSCW (Nov. 2019). DOI: `10.1145/3359152`. URL: `https://doi-org.tudelft.idm.oclc.org/10.1145/3359152`.

[51] Katia Gil Guzman. *RAG with a Graph database*. Dec. 2023. URL: `https://cookbook.openai.com/examples/rag_with_graph_db`.

[52] Gaole He, Lucie Kuiper, and Ujwal Gadiraju. "Knowing About Knowing: An Illusion of Human Competence Can Hinder Appropriate Reliance on AI Systems". In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. CHI '23. Hamburg, Germany: Association for Computing Machinery, 2023. ISBN: 9781450394215. DOI: `10.1145/3544548.3581025`.

[53] Dan Hendrycks et al. "Measuring Massive Multitask Language Understanding". In: *CoRR* abs/2009.03300 (2020). arXiv: `2009.03300`. URL: `https://arxiv.org/abs/2009.03300`.

[54] Lei Huang et al. *A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions*. 2023. arXiv: `2311.05232 [cs.CL]`. URL: `https://arxiv.org/abs/2311.05232`.

[55] Luyang Huang et al. "Efficient Attentions for Long Document Summarization". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2021. DOI: `10.18653/v1/2021.naacl-main.112`. URL: `http://dx.doi.org/10.18653/V1/2021.NAACL-MAIN.112`.

[56] Alexandra C. van Huffelen, M.A.M. Adriaansens, and R.H. Dijkgraaf. *Kamerbrief bij overheidsbrede visie generatieve AI*. 2024. URL: `https://www.rijksoverheid.nl/documenten/kamerstukken/2024/01/18/kamerbrief-bij-overheidsbrede-visie-generatieve-ai-artificiele-intelligentie`.

[57] Jessica Hullman, Eytan Adar, and Priti Shah. "Benefitting InfoVis with Visual Difficulties". In: *IEEE Transactions on Visualization and Computer Graphics* 17.12 (Dec. 2011), pp. 2213–2222. ISSN: 1077-2626. DOI: `10.1109/TVCG.2011.175`. URL: `https://doi-org.tudelft.idm.oclc.org/10.1109/TVCG.2011.175`.

[58] Intel. *Neural Chat*. Oct. 2023. URL: `https://huggingface.co/Intel/neural-chat-7b-v3-1`.

[59] Anton Ioffe. *Integrating RAG with SQL Databases: Techniques and Best Practices*. Jan. 2024. URL: `https://borstch.com/blog/development/integrating-rag-with-sql-databases-techniques-and-best-practices`.

[60] Albert Q. Jiang et al. *Mistral 7B*. 2023. arXiv: `2310.06825 [cs.CL]`.

[61] Huiqiang Jiang et al. *LongLLMLingua: Accelerating and Enhancing LLMs in Long Context Scenarios via Prompt Compression*. 2023. arXiv: `2310.06839 [cs.CL]`.

[62] Xiaoqi Jiao et al. "TinyBERT: Distilling BERT for Natural Language Understanding". In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, 2020. DOI: `10.18653/v1/2020.findings-emnlp.372`. URL: `http://dx.doi.org/10.18653/v1/2020.findings-emnlp.372`.

[63] Pieter Jolen et al. *Eigenlijk werken we allemaal agile… - Audit Magazine*. Sept. 2023. URL: `https://auditmagazine.nl/artikelen/eigenlijk-werken-we-allemaal-agile/`.

[64] Mandar Joshi et al. "SpanBERT: Improving Pre-training by Representing and Predicting Spans". In: *Transactions of the Association for Computational Linguistics* 8 (Jan. 2020), pp. 64–77. ISSN: 2307-387X. DOI: `10.1162/tacl_a_00300`. eprint: `https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl\_a\_00300/1923170/tacl\_a\_00300.pdf`. URL: `https://doi.org/10.1162/tacl%5C_a%5C_00300`.

[65] Daniel Jurafsky and James H. Martin. *Speech and Language Processing*. To be published; draft. 2024. Chap. 3.1. URL: `https://web.stanford.edu/~jurafsky/slp3/ed3bookfeb3_2024.pdf`.

[66] Patricia K. Kahr et al. “It Seems Smart, but It Acts Stupid: Development of Trust in AI Advice in a Repeated Legal Decision-Making Task”. In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. IUI ’23. Sydney, NSW, Australia: Association for Computing Machinery, 2023, pp. 528–539. DOI: 10.1145/3581641.3584058. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3581641.3584058.

[67] Takuya Kida et al. “Byte Pair Encoding: a Text Compression Scheme That Accelerates Pattern Matching”. In: 1999. URL: https://api.semanticscholar.org/CorpusID:18801509.

[68] Sunnie S. Y. Kim et al. ““I’m Not Sure, But...”: Examining the Impact of Large Language Models’ Uncertainty Expression on User Reliance and Trust”. In: *The 2024 ACM Conference on Fairness, Accountability, and Transparency*. FAccT ’24. ACM, June 2024. DOI: 10.1145/3630106.3658941. URL: http://dx.doi.org/10.1145/3630106.3658941.

[69] Vamsi Krishna Kommineni, Birgitta König-Ries, and Sheeba Samuel. *From human experts to machines: An LLM supported approach to ontology and knowledge graph construction*. 2024. arXiv: 2403.08345 [cs.CL]. URL: https://arxiv.org/abs/2403.08345.

[70] Taku Kudo and John Richardson. “SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing”. In: *CoRR* abs/1808.06226 (2018). arXiv: 1808.06226. URL: http://arxiv.org/abs/1808.06226.

[71] Vivian Lai et al. “Towards a Science of Human-AI Decision Making: An Overview of Design Space in Empirical Human-Subject Studies”. In: *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. FAccT ’23. Chicago, IL, USA: Association for Computing Machinery, 2023, pp. 1369–1385. DOI: 10.1145/3593013.3594087.

[72] Zhenzhong Lan et al. “ALBERT: A Lite BERT for Self-supervised Learning of Language Representations”. In: *CoRR* abs/1909.11942 (2019). arXiv: 1909.11942. URL: http://arxiv.org/abs/1909.11942.

[73] Langchain. *ChatMessage*. 2024. URL: https://api.python.langchain.com/en/latest/messages/langchain_core.messages.chat.ChatMessage.html.

[74] Langchain. *Conversational Retrieval Chain*. 2024. URL: https://sj-langchain.readthedocs.io/en/latest/chains/langchain.chains.conversational_retrieval.base.ConversationalRetrievalChain.html?highlight=conversationalretrievalchain#langchain.chains.conversational_retrieval.base.ConversationalRetrievalChain.

[75] Langchain. *Document loaders | Langchain*. 2024. URL: https://python.langchain.com/docs/modules/data_connection/document_loaders.

[76] Langchain. *Langchain Documentation*. 2023. URL: https://python.langchain.com/docs/get_started/introduction.

[77] Langchain. *Recursively split by character | Langchain*. 2024. URL: https://python.langchain.com/docs/modules/data_connection/document_transformers/recursive_text_splitter.

[78] Langchain. *VectorStoreRetriever*. 2024. URL: https://sj-langchain.readthedocs.io/en/latest/vectorstores/langchain.vectorstores.base.VectorStoreRetriever.html#langchain.vectorstores.base.VectorStoreRetriever.

[79] Annemarie van de Langenberg-de Reuver and Joost van Beijsterveld. *Agile Auditing @ Jumbo - Audit Magazine*. Jan. 2022. URL: https://auditmagazine.nl/artikelen/agile-auditing-jumbo-a-good-practice/.

[80] John D. Lee and Katrina A. See. “Trust in Automation: Designing for Appropriate Reliance”. In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 46.1 (2004), pp. 50–80. DOI: 10.1518/hfes.46.1.50_30392.

[81] Patrick Lewis et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[82] Xiaonan Li et al. *LLatrieval: LLM-Verified Retrieval for Verifiable Generation*. 2024. arXiv: 2311.07838 [cs.CL]. URL: https://arxiv.org/abs/2311.07838.

[83] Chin-Yew Lin. “ROUGE: A Package for Automatic Evaluation of Summaries”. In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. URL: https://aclanthology.org/W04-1013.

[84] Stephanie Lin, Jacob Hilton, and Owain Evans. “TruthfulQA: Measuring How Models Mimic Human Falsehoods”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2022. DOI: 10.18653/v1/2022.acl-long.229. URL: http://dx.doi.org/10.18653/v1/2022.acl-long.229.

[85] Han Liu, Vivian Lai, and Chenhao Tan. “Understanding the Effect of Out-of-Distribution Examples and Interactive Explanations on Human-AI Decision Making”. In: *Proc. ACM Hum.-Comput. Interact.* 5.CSCW2 (Oct. 2021). DOI: 10.1145/3479552.

[86] Jerry Liu and AI Engineer. *Building Production-Ready RAG Applications: Jerry Liu*. Nov. 2023. URL: https://youtu.be/TRjq7t2Ms5I?si=-PAxWEPbv9fAib-i&t=519.

[87] Yinhan Liu et al. “RoBERTa: A Robustly Optimized BERT Pretraining Approach”. In: *CoRR* abs/1907.11692 (2019). arXiv: 1907.11692. URL: http://arxiv.org/abs/1907.11692.

[88] Yubo Ma et al. “Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples!” In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics, 2023. DOI: 10.18653/v1/2023.findings-emnlp.710. URL: http://dx.doi.org/10.18653/v1/2023.findings-emnlp.710.

[89] S. S. Manathunga and Y. A. Illangasekara. *Retrieval Augmented Generation and Representative Vector Summarization for large unstructured textual data in Medical Education*. 2023. arXiv: 2308.00479 [cs.CL].

[90] Yuning Mao et al. “Multi-document Summarization with Maximal Marginal Relevance-guided Reinforcement Learning”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020. DOI: 10.18653/v1/2020.emnlp-main.136. URL: http://dx.doi.org/10.18653/v1/2020.emnlp-main.136.

[91] MariaDB. *MariaDB Server (SQL Database Server) – MariaDB Documentation*. 2024. URL: https://mariadb.com/docs/server/.

[92] Roger C. Mayer, James H. Davis, and F. David Schoorman. “An Integrative Model of Organizational Trust”. In: *The Academy of Management Review* 20.3 (1995), pp. 709–734. ISSN: 03637425. URL: http://www.jstor.org/stable/258792 (visited on 01/09/2024).

[93] Tomas Mikolov et al. *Efficient Estimation of Word Representations in Vector Space*. 2013. arXiv: 1301.3781 [cs.CL]. URL: https://arxiv.org/abs/1301.3781.

[94] Sewon Min et al. *FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation*. 2023. arXiv: 2305.14251 [cs.CL]. URL: https://arxiv.org/abs/2305.14251.

[95] Ministerie van Binnenlandse Zaken en Koninkrijkrelaties. *Jaarrapportage Bedrijfsvoering Rijk 2022*. Table (Tabel) 34. Rijksoverheid, 2022, p. 115. URL: https://open.overheid.nl/documenten/ronl-b46def6afb3da457fe5330739244a42c8126a324/pdf.

[96] Ministerie van Financiën. *Fiches*. 2024. URL: https://www.rijksfinancien.nl/hafir/fiches.

[97] Ministerie van Financiën. *Rijksbegrotingsvoorschriften 2024*. 2024. URL: https://rbv.rijksfinancien.nl/sites/default/files/pdf/Rijksbegrotingsvoorschriften%202024%20%285%29.pdf.

[98] Ministerie van Financiën - Begrotingszaken/BBH. *Fiche: budgetrecht*. Nov. 2018. URL: https://www.rijksfinancien.nl/sites/default/files/hafir/Fiches/Fiche-Budgetrecht-definitief.pdf.

[99] Ministerie van Financiën - Rijksoverheid. *HAFIR*. 2024. URL: https://www.rijksfinancien.nl/hafir.

[100] Ministerie van Financiën - Rijksoverheid. *Organogram ministerie van Financiën*. 2024. URL: https://www.rijksoverheid.nl/ministeries/ministerie-van-financien/organisatie/organogram.

[101] Arindam Mitra et al. *Orca 2: Teaching Small Language Models How to Reason*. 2023. arXiv: 2311.11045 [cs.AI].

[102] Mudasir Mohd, Rafiya Jan, and Muzaffar Shah. "Text document summarization using word embedding". In: *Expert Systems with Applications* 143 (2020), p. 112958. DOI: https://doi.org/10.1016/j.eswa.2019.112958. URL: https://www.sciencedirect.com/science/article/pii/S0957417419306761.

[103] Khalid Nassiri and Moulay Akhloufi. "Transformer models used for text-based question answering systems". In: *Applied Intelligence* 53.9 (2023), pp. 10602–10635. DOI: 10.1007/s10489-022-04052-8.

[104] Gonzalo Navarro. "A guided tour to approximate string matching". In: *ACM Comput. Surv.* 33.1 (Mar. 2001), pp. 31–88. ISSN: 0360-0300. DOI: 10.1145/375360.375365. URL: https://doi.org/10.1145/375360.375365.

[105] Plaban Nayak. *Advanced RAG — Improving retrieval using Hypothetical Document Embeddings(HyDE)*. Nov. 2023. URL: https://medium.aiplanet.com/advanced-rag-improving-retrieval-using-hypothetical-document-embeddings-hyde-1421a8ec075a.

[106] Jeroen Nikkel. *Chatbot ChatGPT bereikt bijna anderhalf miljoen personen in Nederland*. Apr. 2023.

[107] OpenAI. *Embeddings - OpenAI API*. 2024. URL: https://platform.openai.com/docs/guides/embeddings/embeddings.

[108] OpenAI. *Pricing*. 2024. URL: https://openai.com/pricing.

[109] OpenAI. *Text Generation - OpenAI API*. 2024. URL: https://platform.openai.com/docs/guides/text-generation/chat-completions-api.

[110] OpenAI. *What are tokens and how to count them? | OpenAI Help Center*. 2024. URL: https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them.

[111] OpenAI et al. *GPT-4 Technical Report*. Tech. rep. OpenAI, 2023. arXiv: 2303.08774 [cs.CL].

[112] Organisatie en Personeel Rijk. *Wat is de Rijksoverheid? - Werken voor Nederland*. 2024. URL: https://www.werkenvoornederland.nl/over-de-rijksoverheid/wat-is-de-rijksoverheid.

[113] Long Ouyang et al. "Training language models to follow instructions with human feedback". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 27730–27744. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

[114] Kishore Papineni et al. "Bleu: a Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Ed. by Pierre Isabelle, Eugene Charniak, and Dekang Lin. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, July 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040.

[115] Joon Sung Park et al. "A Slow Algorithm Improves Users' Assessments of the Algorithm's Accuracy". In: *Proc. ACM Hum.-Comput. Interact.* 3.CSCW (Nov. 2019). DOI: 10.1145/3359204. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3359204.

[116] Marinus de Pooter. *Is risicomanagement overtollig? - Audit Magazine*. July 2022. URL: https://auditmagazine.nl/artikelen/is-risicomanagement-overtollig/.

[117] Karen Punter. *Actieve openbaarmaking auditrapporten Auditdienst Rijk - Audit Magazine*. Apr. 2022. URL: `https://auditmagazine.nl/artikelen/actieve-openbaarmaking-auditrapporten-auditdienst-rijk/`.

[118] Colin Raffel et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". In: *The Journal of Machine Learning Research* 21.1 (Oct. 2020), pp. 5485–5551. ISSN: 1532-4435. URL: `https://www.jmlr.org/papers/volume21/20-074/20-074.pdf`.

[119] Ragna Development Team at Quansight LLC. *Home - Ragna*. 2023. URL: `https://ragna.chat/en/stable/`.

[120] Vipula Rawte, Amit Sheth, and Amitava Das. *A Survey of Hallucination in Large Foundation Models*. 2023. arXiv: `2309.05922 [cs.AI]`. URL: `https://arxiv.org/abs/2309.05922`.

[121] Edwin Rijgersberg and Bob Lucassen. *GEITje: een groot open Nederlands taalmodel*. Dec. 2023. URL: `https://github.com/Rijgersberg/GEITje`.

[122] Seo-hyun Kim Ha-rin Lee. *Bring Retrieval Augmented Generation to Google Gemini via External API: An Evaluation with BIG-Bench Dataset*. 2024. DOI: `10.21203/rs.3.rs-4394715/v1`.

[123] Antoine Ross. *RAG vs. Fine-Tuning: Navigating the Best Path for Your AI Project*. [Online; accessed 2023-12-21]. Dec. 2023. URL: `https://medium.com/@antoineross/rag-vs-fine-tuning-navigating-the-best-path-for-your-ai-project-fcd65d395537`.

[124] Prasenjeet Roy and Suman Kundu. "Review on Query-focused Multi-document Summarization (QMDS) with Comparative Analysis". In: *ACM Computing Surveys* 56.1 (Aug. 2023). ISSN: 0360-0300. DOI: `10.1145/3597299`. URL: `https://doi.org/10.1145/3597299`.

[125] Max Schemmer et al. *Should I Follow AI-based Advice? Measuring Appropriate Reliance in Human-AI Decision-Making*. 2022. arXiv: `2204.06916 [cs.HC]`. URL: `https://arxiv.org/abs/2204.06916`.

[126] Ori Shapira and Ran Levy. "Massive Multi-Document Summarization of Product Reviews with Weak Supervision". In: *CoRR* abs/2007.11348 (2020). arXiv: `2007.11348`. URL: `https://arxiv.org/abs/2007.11348`.

[127] Sharvil. *Tokenization Algorithms Explained*. July 2021. URL: `https://towardsdatascience.com/tokenization-algorithms-explained-e25d5f4322ac`.

[128] Shamane Siriwardhana et al. "Improving the Domain Adaptation of Retrieval Augmented Generation (RAG) Models for Open Domain Question Answering". In: *Transactions of the Association for Computational Linguistics* 11 (Jan. 2023), pp. 1–17. ISSN: 2307-387X. DOI: `10.1162/tacl_a_00530`. eprint: `https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl\_a\_00530/2067834/tacl\_a\_00530.pdf`. URL: `https://doi.org/10.1162/tacl%5C_a%5C_00530`.

[129] Iodine98 (Rohan Sobha). *Iodine98/dora-back: A Python backend for Document Retrieval and Analysis (DoRA)*. 2023. URL: `https://github.com/Iodine98/dora-back`.

[130] Irene Solaiman et al. "Release Strategies and the Social Impacts of Language Models". In: *CoRR* abs/1908.09203 (2019). arXiv: `1908.09203`. URL: `http://arxiv.org/abs/1908.09203`.

[131] DiJia Su et al. "Optimizing Multidocument Summarization by Blending Reinforcement Learning Policies". In: *IEEE Transactions on Artificial Intelligence* 4.3 (2023), pp. 416–427. DOI: `10.1109/TAI.2022.3201807`.

[132] Liyan Tang et al. "Evaluating large language models on medical evidence summarization". In: *npj Digital Medicine* 6.1 (2023), p. 158. DOI: `10.1038/s41746-023-00896-7`.

[133] LlamaIndex Team. *LlamaIndex*. [Accessed on 2024-05-01]. 2024. URL: `https://docs.llamaindex.ai/en/stable/`.

[134] The Kubernetes Authors. *Kubernetes Documentation | Kubernetes*. 2024. URL: `https://kubernetes.io/docs/home/`.

[135] TNO. *The Netherlands starts realisation GPT-NL, its own open AI-language mo*. [Online; accessed 2024-01-04]. Nov. 2023. URL: https://www.tno.nl/en/newsroom/2023/11/netherlands-starts-realisation-gpt-nl/.

[136] Hugo Touvron et al. *Llama 2: Open Foundation and Fine-Tuned Chat Models*. 2023. arXiv: 2307.09288 [cs.CL].

[137] Lewis Tunstall et al. *Zephyr: Direct Distillation of LM Alignment*. 2023. arXiv: 2310.16944 [cs.LG].

[138] Tweede Kamer der Staten-Generaal. *Het Grote Begrotingsboek*. 2021. URL: https://www.tweedekamer.nl/sites/default/files/atoms/files/het_grote_begrotingsboek.pdf.

[139] u/m1n1crusher. *GPT-3.5 vs GPT4: Same prompt, unbelievable difference*. 2023. URL: https://www.reddit.com/r/ChatGPT/comments/11zouvn/gpt35_vs_gpt4_same_prompt_unbelievable_difference/.

[140] Daniel Ullrich, Andreas Butz, and Sarah Diefenbach. "The Development of Overtrust: An Empirical Simulation and Psychological Analysis in the Context of Human–Robot Interaction". In: *Frontiers in Robotics and AI* 8 (2021). ISSN: 2296-9144. DOI: 10.3389/frobt.2021.554578. URL: https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2021.554578.

[141] Chris Van Der Lee et al. "Best practices for the human evaluation of automatically generated text". In: *Proceedings of the 12th International Conference on Natural Language Generation*. 2019, pp. 355–368. DOI: 10.18653/v1/W19-8643.

[142] Bram Vanroy. *Language Resources for Dutch Large Language Modelling*. 2023. arXiv: 2312.12852 [cs.CL].

[143] Bram Vanroy. *Llama-2-13b-chat-dutch*. [Model]. 2023. URL: https://huggingface.co/BramVanroy/Llama-2-13b-chat-dutch.

[144] Bram Vanroy. *llama2-13b-ft-mc4_nl_cleaned_tiny*. [Model]. 2023. URL: https://huggingface.co/BramVanroy/llama2-13b-ft-mc4_nl_cleaned_tiny.

[145] Helena Vasconcelos et al. "Explanations Can Reduce Overreliance on AI Systems During Decision-Making". In: *Proc. ACM Hum.-Comput. Interact.* 7.CSCW1 (Apr. 2023). DOI: 10.1145/3579605. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3579605.

[146] Github users: evchaki et al. Explanation page on Vector databases. Mar. 2024. URL: https://learn.microsoft.com/en-us/semantic-kernel/memories/vector-db.

[147] Dave van Veen et al. "Adapted large language models can outperform medical experts in clinical text summarization". In: *Nature Medicine* 30.4 (Feb. 2024), pp. 1134–1142. ISSN: 1546-170X. DOI: 10.1038/s41591-024-02855-5.

[148] Cunxiang Wang et al. *Survey on Factuality in Large Language Models: Knowledge, Retrieval and Domain-Specificity*. 2023. arXiv: 2310.07521 [cs.CL]. URL: https://arxiv.org/abs/2310.07521.

[149] Kexiang Wang, Baobao Chang, and Zhifang Sui. "A Spectral Method for Unsupervised Multi-Document Summarization". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber et al. Online: Association for Computational Linguistics, Nov. 2020, pp. 435–445. DOI: 10.18653/v1/2020.emnlp-main.32. URL: https://aclanthology.org/2020.emnlp-main.32.

[150] Zheng Wang et al. *M-RAG: Reinforcing Large Language Model Performance through Retrieval-Augmented Generation with Multiple Partitions*. 2024. arXiv: 2405.16420 [cs.CL]. URL: https://arxiv.org/abs/2405.16420.

[151] Joseph Weizenbaum. "ELIZA—a computer program for the study of natural language communication between man and machine". In: *Communications of the ACM* 9.1 (1966), pp. 36–45.

[152] WEM. *WEM No-code development platform*. 2024. URL: https://wem.io/.

[153] The authors of Wikipedia. *Query language*. 2024. URL: https://en.wikipedia.org/wiki/Query_language.

[154] Jeff Wu et al. "Recursively Summarizing Books with Human Feedback". In: *CoRR* abs/2109.10862 (2021). arXiv: `2109.10862`. URL: `https://arxiv.org/abs/2109.10862`.

[155] Xingjiao Wu et al. "A survey of human-in-the-loop for machine learning". In: *Future Generation Computer Systems* 135 (2022), pp. 364–381. ISSN: 0167-739X. DOI: `https://doi.org/10.1016/j.future.2022.05.014`. URL: `https://www.sciencedirect.com/science/article/pii/S0167739X22001790`.

[156] Rowan Zellers et al. "HellaSwag: Can a Machine Really Finish Your Sentence?" In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Ed. by Anna Korhonen, David Traum, and Lluís Màrquez. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 4791–4800. DOI: `10.18653/v1/P19-1472`. URL: `https://aclanthology.org/P19-1472`.

[157] Chunpeng Zhai, Santoso Wibowo, and Lily D. Li. "The effects of over-reliance on AI dialogue systems on students' cognitive abilities: a systematic review". In: *Smart Learning Environments* 11.1 (2024). DOI: `10.1186/s40561-024-00316-7`.

[158] Jiebin Zhang et al. *Retrieval-based Full-length Wikipedia Generation for Emergent Events*. 2024. arXiv: `2402.18264 [cs.CL]`. URL: `https://arxiv.org/abs/2402.18264`.

[159] Tianjun Zhang et al. *RAFT: Adapting Language Model to Domain Specific RAG*. 2024. arXiv: `2403.10131 [cs.CL]`. URL: `https://arxiv.org/abs/2403.10131`.

[160] Tianyi Zhang et al. *BERTScore: Evaluating Text Generation with BERT*. 2020. arXiv: `1904.09675 [cs.CL]`.

[161] Yue Zhang et al. *Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models*. 2023. arXiv: `2309.01219 [cs.CL]`. URL: `https://arxiv.org/abs/2309.01219`.

[162] Wayne Xin Zhao et al. *A Survey of Large Language Models*. 2023. arXiv: `2303.18223 [cs.CL]`. URL: `https://arxiv.org/abs/2303.18223`.

[163] Jianlong Zhou, Simon Luo, and Fang Chen. "Effects of personality traits on user trust in human–machine collaborations". In: *Journal on Multimodal User Interfaces* 14.4 (2020), pp. 387–400. DOI: `10.1007/s12193-020-00329-9`.

A
Surveys

A.1. Experiment Group

Please find below the survey for the experiment group in table A.1. It contains 35 questions.

Table A.1: Survey for experiment group

Number	Question	Subject	Tooltip
01.1	How clear is the original answer from DoRA?	Clarity	The answer can be considered extremely clear, provided that the language level is a maximum of B1 and each sentence logically follows the previous one.
01.2	Elaborate on your rating of the clarity of the original answer from DoRA. Use an example if applicable.	Clarity	
02.1	How complete is the original answer from DoRA in relation to the source documents?	Completeness	The answer can be considered as fully complete if the main points from the source documents have been mentioned.
02.2	Elaborate on your rating of completeness of the original answer from DoRA. Use an example if applicable.	Completeness	
03.1	To what extent does the original answer from DoRA reflect what is present in the source documents?	Relevance	The answer can be considered relevant if most of the points in the answer come from the quoted sources. The sources can be found in the upper panel. You are requested to consult the corresponding sources and the source documents.

03.2	Elaborate on your rating above of the relevance of the original answer from DoRA. Use an example if applicable.	Relevance	
04.1	How accurate are the points included in the original answer from DoRA in relation to the source documents?	Accuracy	The answer can be considered accurate provided it also represents the content of the source documents.
04.2	Elaborate on your rating of the accuracy of the original answer from DoRA. Use an example if applicable.	Accuracy	
05.1	How consistent is the style and tone of the original answer from DoRA?	Consistency	The answer can be considered consistent provided that: • the same tense (present tense/past tense) • the same level of formality (jij/u) are maintained throughout the text.
05.2	Elaborate on your rating of the consistency of the original answer from DoRA. Use an example if applicable.	Consistency	
06.1	How structured is the original answer from DoRA?	Structure	The answer can be considered fully structured, provided it is logically divided into multiple paragraphs, possibly with headings.
06.2	Elaborate on your rating of the structure of the original answer from DoRA. Use an example if applicable.	Structure	
07.1	How concise is the original answer from DoRA?	Conciseness	The answer can be considered fully concise, provided it is ”straightforward” and the information is provided in as few words as possible.

07.2	Elaborate on your rating of the conciseness of the original answer from DoRA. Use an example if applicable.	Conciseness	
08.1	Rate the grammar and spelling quality of the original answer from DoRA.	Grammar and Spelling	The answer can be considered extremely good provided there are no spelling or grammar mistakes to be found.
08.2	Elaborate on your rating of the grammar and spelling quality. In case there are one or more grammar or spelling errors are present, highlight the most remarkable error.	Grammar and Spelling	
09.1	How coherent is the original answer from DoRA?	Coherence	The answer can be considered fully coherent provided all the points mentioned in the text logically flow into each other and it comes across as one piece of text.
09.2	Elaborate on your rating of the coherence of the original answer from DoRA. Use an example if applicable.	Coherence	
12.1	To what extent do you have an understanding of RAG systems and their goals?	Understanding of RAG Systems	
12.2	Elaborate on your understanding in 1-2 sentences.	Understanding of RAG Systems	
12.3	To what extent have you ever worked with RAG-systems or evaluated these previously?	Understanding of RAG Systems	
12.4	Elaborate on your experience(s) with RAG-systems in 1-2 sentences. If you have no experience, please fill in ’n.v.t.’ (niet van toepassing; NA; Not Applicable)	Understanding of RAG Systems	

13.1	In your opinion, does DoRA achieve comparable quality to human domain experts?	Quality Measurement of RAG Outcomes	You can consider your close colleagues as human domain experts. How does the quality of DoRA compare to them?
13.2	By means of an example (if applicable), explain your rating in 1-2 sentences.	Quality Measurement of RAG Outcomes	You can consider your close colleagues as human domain experts.
14.1	Can you provide examples where DoRA's outcome excels in quality compared to human-written content?	Comparing RAG and Human Expert Outcomes	You can consider your close colleagues as human domain experts.
14.2	Conversely, are there moments where human experts perform better than DoRA?	Comparing RAG and Human Expert Outcomes	You can consider your close colleagues as human domain experts.
14.3	How do you see the trade-offs between content generated by RAG and human-written content in terms of quality and efficiency?	Comparing RAG and Human Expert Outcomes	A RAG system is a system like DoRA, where a large language model (e.g., GPT-3.5/GPT-4/Gemini/ChatGPT) is supplemented with the ability to search for information from a corpus of documents. You can consider your close colleagues as human domain experts.
15.1	What challenges do you encounter as you assess the text quality of DoRA? Are there specific limitations that affect performance of DoRA?	Challenges and Limitations	Challenges refer to how the application or other internal factors make it more difficult to apply DoRA, at this time, in the current processes. Limitations refer to which external factors are currently at play that affect DoRA's performance.
15.2	How do these challenges influence the overall effectiveness of DoRA for text aggregation tasks?	Challenges and Limitations	Text aggregation tasks refer to summaries, juxtapositions, and other tasks where pieces of text from source documents are combined and paraphrased to create a new piece of text.
16.1	How much do you trust RAG-systems such as DoRA?	User Trust and Confidence	A RAG system is a system like DoRA, where a large language model (e.g., GPT-3.5/GPT-4/Gemini/ChatGPT) is supplemented with the ability to search for information from a corpus of documents.

16.2	Which factors influence your trust? Please elaborate on your trust rating.	User Trust and Confidence	
16.3	How can RAG systems increase user confidence in their outcomes, especially compared to human-written content?	User Trust and Confidence	A RAG system is a system like DoRA, where a large language model (e.g., GPT-3.5/GPT-4/Gemini/ChatGPT) is supplemented with the ability to search for information from a corpus of documents.
17.1	How did you perceive the time it took to complete the writing task?	Duration	According to your own judgement, did it go fast or rather slow?
17.2	Imagine you had to perform the assigned task manually, how would you estimate your speed at completing this task?	Duration	Manually refers to not having access to a RAG-system such as DoRA.
18.2	Feel free to leave miscellaneous comments here that do not fit the questions above.	Miscellaenous	

A.2. Verifier Group

Please find below the survey for the verifier group in table A.2. It contains 20 questions.

Table A.2: Survey for verifier group

Number	Question	Subject	Tooltip
01.1	How clear is the document belonging to the current task?	Clarity	The document can be considered extremely clear, provided that the language level is a maximum of B1 and each sentence logically follows the previous one.
01.2	Elaborate on your rating of the clarity of the document belonging to the current task. Use an example if applicable.	Clarity	
02.1	How complete is the document belonging to the current task in relation to the source documents?	Completeness	The document can be considered as complete if the main points from the source documents have been mentioned.
02.2	Elaborate on your rating of completeness of the document belonging to the current task. Use an example if applicable.	Completeness	
03.1	To what extent does the the document belonging to the current task reflect what is present in the source documents?	Relevance	The document can be considered relevant if most of the points in the document come from the quoted sources. The sources can be found in the upper panel. You are requested to consult the corresponding sources and the source documents.
03.2	Elaborate on your rating above of the relevance of the document belonging to the current task. Use an example if applicable.	Relevance	
04.1	How accurate are the points included in the document belonging to the current task in relation to the source documents?	Accuracy	The document can be considered accurate provided it also represents the content of the source documents.

04.2	Elaborate on your rating of the accuracy of the document belonging to the current task. Use an example if applicable.	Accuracy	
05.1	How consistent is the style and tone of the document belonging to the current task?	Consistency	The document can be considered consistent provided that: • the same tense (present tense/past tense) • the same level of formality (jij/u) are maintained throughout the text.
05.2	Elaborate on your rating of the consistency of the document belonging to the current task. Use an example if applicable.	Consistency	
06.1	How structured is the document belonging to the current task?	Structure	The document can be considered fully structured, provided it is logically divided into multiple paragraphs, possibly with headings.
06.2	Elaborate on your rating of the structure of the document belonging to the current task. Use an example if applicable.	Structure	
07.1	How concise is the document belonging to the current task?	Conciseness	The document can be considered fully concise, provided it is ”straightforward” and the information is provided in as few words as possible.
07.2	Elaborate on your rating of the conciseness of the document belonging to the current task. Use an example if applicable.	Conciseness	
08.1	Rate the grammar and spelling quality of the document belonging to the current task.	Grammar and Spelling	The document can be considered extremely good provided there are no spelling or grammar mistakes to be found.

08.2	Elaborate on your rating of the grammar and spelling quality. In case there are one or more grammar or spelling errors are present, highlight the most remarkable error.	Grammar and Spelling	
09.1	How coherent is the document belonging to the current task?	Coherence	The document can be considered fully coherent provided all the points mentioned in the text logically flow into each other and it comes across as one piece of text.
09.2	Elaborate on your rating of the coherence of the document belonging to the current task. Use an example if applicable.	Coherence	
10.1	How satisfied are you with the overall quality of the document belonging to the current task?	Satisfaction	
18.1	Feel free to leave miscellaneous comments here that do not fit the questions above.	Miscellaenous	

B
Answer Options

Please find below the tables with with answer options and their corresponding Likert-scales for:

- clarity (table B.1)
- completeness (table B.2)
- relevancy (table B.3)
- accuracy (table B.4)
- consistency (table B.5)
- structure (table B.6)
- conciseness (table B.7)
- grammar and spelling quality (table B.8)
- coherency (table B.9)
- satisfaction (table B.10)
- familiarity (table B.11)
- experience level (table B.12)
- comparability (table B.13)
- trust (table B.14)
- perceived speed (table B.15)

Table B.1: Answer options: Clarity

Number	**Scale**	**Label**
01.1	1	Completely unclear
01.1	2	Mostly unclear
01.1	3	Somewhat unclear
01.1	4	Neither clear or unclear
01.1	5	Somewhat clear
01.1	6	Mostly clear
01.1	7	Completely clear

Table B.2: Answer options: Completeness

Number	Scale	Label
02.1	1	Entirely incomplete
02.1	2	Mostly incomplete
02.1	3	Somewhat incomplete
02.1	4	Neither complete or incomplete
02.1	5	Somewhat complete
02.1	6	Mostly complete
02.1	7	Entirely complete

Table B.3: Answer options: Relevancy

Number	Scale	Label
03.1	1	Completely irrelevant
03.1	2	Mostly irrelevant
03.1	3	Somewhat irrelevant
03.1	4	Neither relevant or irrelevant
03.1	5	Somewhat relevant
03.1	6	Mostly relevant
03.1	7	Completely relevant

Table B.4: Answer options: Accuracy

Number	Scale	Label
04.1	1	Completely inaccurate
04.1	2	Mostly inaccurate
04.1	3	Somewhat inaccurate
04.1	4	Neither accurate or inaccurate
04.1	5	Somewhat accurate
04.1	6	Mostly accurate
04.1	7	Completely accurate

Table B.5: Answer options: Consistency

Number	Scale	Label
05.1	1	Completely inconsistent
05.1	2	Mostly inconsistent
05.1	3	Somewhat inconsistent
05.1	4	Neither consistent or inconsistent
05.1	5	Somewhat consistent
05.1	6	Mostly consistent
05.1	7	Completely consistent

Table B.6: Answer options: Structure

Number	Scale	Label
06.1	1	Completely unstructured
06.1	2	Mostly unstructured
06.1	3	Somewhat unstructured
06.1	4	Neither structured nor unstructured
06.1	5	Somewhat structured
06.1	6	Mostly structured
06.1	7	Completely structured

Table B.7: Answer options: Conciseness

Number	Scale	Label
07.1	1	Completely wordy
07.1	2	Mostly wordy
07.1	3	Somewhat wordy
07.1	4	Neither wordy nor concise
07.1	5	Somewhat concise
07.1	6	Mostly concise
07.1	7	Completely concise

Table B.8: Answer options: Grammar and Spelling Quality

Number	Scale	Label
08.1	1	Very bad
08.1	2	Bad
08.1	3	Insufficient
08.1	4	Moderate
08.1	5	Sufficient
08.1	6	Good
08.1	7	Very good

Table B.9: Answer options: Coherency

Number	Scale	Label
09.1	1	Completely incoherent
09.1	2	Mostly incoherent
09.1	3	Somewhat incoherent
09.1	4	Not coherent or incoherent
09.1	5	Somewhat coherent
09.1	6	Mostly coherent
09.1	7	Completely coherent

Table B.10: Answer options: Satisfaction

Number	Scale	Label
10.1	1	Completely dissatisfied
10.1	2	Mostly dissatisfied
10.1	3	Somewhat dissatisfied
10.1	4	Not satisfied or dissatisfied
10.1	5	Somewhat satisfied
10.1	6	Mostly satisfied
10.1	7	Completely satisfied

Table B.11: Answer options: Familiarity

Number	Scale	Label
12.1	1	Completely unfamiliar
12.1	2	Mostly unfamiliar
12.1	3	Somewhat unfamiliar
12.1	4	Not familiar or unfamiliar
12.1	5	Somewhat familiar
12.1	6	Mostly familiar
12.1	7	Completely familiar

Table B.12: Answer options: Experience level

Number	Scale	Label
12.3	1	Completely inexperienced
12.3	2	Mostly inexperienced
12.3	3	Somewhat inexperienced
12.3	4	Not experienced or inexperienced
12.3	5	Somewhat experienced
12.3	6	Mostly experienced
12.3	7	Completely experienced

Table B.13: Answer options: Comparability

Number	Scale	Label
13.1	1	Completely incomparable
13.1	2	Mostly incomparable
13.1	3	Somewhat incomparable
13.1	4	Not comparable or incomparable
13.1	5	Somewhat comparable
13.1	6	Mostly comparable
13.1	7	Completely comparable

Table B.14: Answer options: Trust

Number	Scale	Label
16.1	1	Completely untrustworthy
16.1	2	Mostly untrustworthy
16.1	3	Somewhat untrustworthy
16.1	4	Not trustworthy or untrustworthy
16.1	5	Somewhat trustworthy
16.1	6	Mostly trustworthy
16.1	7	Completely trustworthy

Table B.15: Answer options: Perceived speed

Number	Scale	Label
17.1 + 17.2	1	Very slow
17.1 + 17.2	2	Mostly slow
17.1 + 17.2	3	Somewhat slow
17.1 + 17.2	4	Neither slow or fast
17.1 + 17.2	5	Somewhat fast
17.1 + 17.2	6	Mostly fast
17.1 + 17.2	7	Very fast

C

User Task Manual

Please find a (redacted) example of a user task manual (in Dutch) attached on the next pages. It starts with a thank-you note and an introduction of the details needed to participate. This is then followed by a section the details of the participant.

Below this section, it lists the names of files that the user should have received. These are partitioned into source files (Bronbestanden) and output files (Uitvoerbestanden). Source files are the original documents which are used to generate summaries and juxtapositions from whereas output files are the the summaries/juxtapositions generated by the control and experiment groups. The output files are only filled when this document is sent to the verifier group as they need to verify the automated output as well as the manual output.

Lastly, it shows the task description (Taakomschrijving) and a link to the website where they can perform this task.

Additionally, the next pages contain prompting tips (Prompt-tips) and frequently asked questions (Veelgestelde vragen) for commonly occuring situations.

DoRA-experiment handleiding

Beste deelnemer,

Hartelijk dank voor uw deelname aan het experiment. Dankzij uw inzet kunnen wij de effectiviteit van DoRA nauwkeurig bepalen. Deze handleiding bevat uw gegevens die u nodig heeft, de bestanden die u ontvangen zou moeten hebben in uw e-mail, een basisprompt (indien van toepassing), uw taakinstructie en extra tips voor het schrijven van prompts in DoRA.

Gegevens

[redacted]

[redacted]	[redacted]
[redacted]	[redacted]
[redacted]	[redacted]

Bronbestanden

Rijksbegrotingsvoorschriften 2024.pdf

Uitvoerbestanden

Taak

Taakomschrijving

Vat het document "Rijksbegrotingsvoorschriften 2024.pdf" samen in maximaal 500 woorden en deel de samenvatting op in vier alinea's met elk één thema. De samenvatting is tekstueel van aard, dus geen bullet points. De samenvatting begint met een titel en heeft subkoppen voor elke alinea.

Voer de bovenstaande taak uit op deze website: [redacted]

Prompt-tips (indien van toepassing)

Experimenteer met verschillende instructies en stel vervolg- en verduidelijkingsvragen.

Experimenteer met het geven van verschillende instructies aan het generatieve AI-model en hanteer een iteratief proces. Herformuleer indien nodig en vergelijk antwoorden met elkaar. Het stellen van vervolg- en verduidelijkingsvragen verhoogt in sommige gevallen de kans op een goede uitkomst. Ook kan dit helpen om erachter te komen of het model een inhoudelijk juist antwoord geeft.

Wees voorzichtig met vooringenomen en suggestieve vragen.

Zorg ervoor dat je instructies zo neutraal mogelijk zijn. Wees je bewust van de aannames die je maakt in een vraag. Let erop dat deze geen ongewenste vooringenomenheden bevatten. Deze kunnen namelijk worden bevestigd in het antwoord. Wanneer je een suggestieve vraag stelt aan een generatief AI-model, is de kans groot dat je een bevestigend antwoord ontvangt. Blijf dus altijd kritisch in het beoordelen van de antwoorden en raadpleeg de originele (wetenschappelijke) bron.

Wees specifiek en nauwkeurig.

Gebruik begrijpelijke taal en geef voldoende details. Vermijd het gebruik van jargon of complexe termen wanneer dit niet strikt nodig is voor de beantwoording van de prompt. Wanneer je een taak geeft aan een generatief AI-model, geef dan een duidelijke en specifieke omschrijving van de taak. Verdeel een complexe taak op in verschillende simpelere sub-taken. Wanneer je een vraag stelt, stel deze dan op een open manier. Stel één vraag tegelijk, om verwarring te voorkomen.

Geef context en voorbeelden mee in je instructies.

Generatieve AI-modellen kennen geen context. Geef dus voldoende achtergrondinformatie mee in je instructies, en gebruik voorbeelden waar mogelijk. Als je een instructie schrijft met een specifiek doel, vermeld dan wat dit doel (en eventueel de doelgroep) is. Je zou een generatief AI-model kunnen zien als een nieuwe collega die net is gestart en heel heldere instructies nodig heeft omdat zij/hij de organisatie en jou nog niet kent.

Geef regels aan het generatieve AI-model.

Geef duidelijk aan wat je verwacht van het generatieve AI-model. Geef bijvoorbeeld aan hoe lang het antwoord moet zijn en in wat voor stijl en format het antwoord gegeven moet worden (bijvoorbeeld in een stap-voor-stap uitleg, tabel of lijst). Soms kan het helpen om het model een bepaalde rol toe te wijzen in je instructie. Hiermee geef je context mee in je prompt, wat de stijl en inhoud van de uitkomst kan beïnvloeden.

[Bron: Handreiking voor overheidsorganisaties bij het gebruik van generatieve AI; versie 0.77]

Veelgestelde vragen

Het model zegt: "Sorry, ik kan je vraag niet beantwoorden met de gegeven context." / "Ik kan je niet helpen met die vraag"

Dat kan één van twee mogelijke oorzaken hebben:

1) Het document is niet goed ingeladen: ga terug naar de vorige pagina, verwijder het document / de documenten, upload deze opnieuw.
2) Het antwoord op jouw vraag is niet direct af te leiden uit de documenten en vraagt meer context.

Het model reageert niet op: "Gedraag je als….."

Het model is, voor nu, dusdanig ingesteld dat het in principe geen prompt aanneemt die vraagt om een rol aan te nemen. Probeer je opdracht en/of vraag direct te formuleren.

Ik krijg de boodschap: "….meld dit aan het onderzoeksteam"

Stuur dan een e-mail naar [redacted]