Creating a Taxonomy for Retrieval Augmented Generation Applications

Anonymous ACL submission

Abstract

In this research, we develop a taxonomy to conceptualize a comprehensive overview of the constituting characteristics that define retrieval augmented generation (RAG) applications, facilitating the adoption of this technology for different application domains. To the best of our knowledge, no holistic RAG application taxonomies have been developed so far. We employ the method foreign to ACL and thus contribute to the set of methods in the taxonomy creation. It comprises four iterative phases designed to refine and enhance our understanding and presentation of RAG's core dimensions. We have developed a total of five meta-dimensions and sixteen dimensions to comprehensively capture the concept of RAG applications. Thus, the taxonomy can be used to better understand RAG applications and to derive design knowledge for future solutions in specific application domains.

1 Introduction

Large Language Models (LLMs) have been identified to have several core limitations. These include a tendency to generate incorrect or misleading information (hallucinations) (Liang et al., 2024; Nonkes et al., 2024), poor arithmetic capabilities, a lack of interpretative power, the high costs associated with model revisions, limitations in handling less popular or low-resource concepts and entities, and an inability to reference sources accurately (Barnett et al., 2024; Soudani et al., 2024; Zhao et al., 2024). Several approaches have been developed to mitigate the limitations of LLMs, while retrieval augmented generation (RAG) is as of now deemed as one of the most promising (Gao et al., 2024). RAG primarily enhances LLMs by incorporating contextual information during the retrieval process, significantly improving the generated content's accuracy and consistency. Consequently, RAG improves LLM tasks and applications in various ways, as evidenced by recent studies (Asai et al., 2024; Jiang et al., 2023; Martino et al., 2023). Given its potential, this paper aims to develop a conceptualization for RAG applications, illustrating how RAG can be systematically implemented to improve LLM tasks and applications across various domains. Recent studies, such as those by Asai et al. (2024); Jiang et al. (2023); Martino et al. (2023), have shown various ways in which RAG can enhance LLMs, underscoring its attributes as explainable, scalable, and adaptable in nature (Siriwardhana et al., 2023).

Given this value of RAG for real-world applications, still there is a dearth of systematization of the field. This is particularly evident in surveys, which emphasize technological aspects over practical applications (Zhao et al., 2024). Therefore, we aim to create a taxonomy that conceptualizes a comprehensive overview of the constituting characteristics that define RAG applications, facilitating the adoption of this technology. Current research on RAG is distributed across various disciplines, and since the technology is evolving very quickly, its unit of analysis is mostly on technological innovations, rather than applications in business contexts. To the best of our knowledge, there have not been any holistic RAG application taxonomies. Thus, our research question is as follows: *"How can RAG applications be conceptualized in a taxonomy?"*

Therefore, the main contributions of the paper are as follows:

- We present a RAG taxonomy offering a comprehensive framework to define and categorize the core characteristics of Retrieval-Augmented Generation (RAG) applications, promoting their broader adoption and practical use.
- We contribute to taxonomy creation methods within the ACL community by adapting the approach of Nickerson et al. (2013) from the field of Information Systems.

- We validate the practical applicability of the created taxonomy by reviewing various papers on RAG applications and also by using the automatic domain classification with ChatGPT.

2 Related Work

In this section, we discuss the peculiarities of the current survey before the other survey papers (Zhao et al., 2024; Gao et al., 2024; Li et al., 2022; Zhao et al., 2023) and discuss the origins of our taxonomy creation approach.

While there exist already various surveys on RAG, we would like to specify, how the current survey paper is different from others. Li et al. (2022) provide one of the first survey papers on the topic that covers early works on RAG with application to NLP tasks such as Dialogue Systems, Neural Machine Translation, Paraphrase Generation, Text Style Transfer, etc. In Zhao et al. (2024), the authors comprehensively review existing efforts that integrate RAG techniques into AI-generated content scenarios, exhibiting how RAG contributes to current generative models. Gao et al. (2024) contextualize the broader scope of RAG research within the landscape of LLMs. The paper presents various RAG components, datasets, and evaluation setups being the perfect source for both understanding the RAG concept as well as delving into the topic. Zhao et al. (2023) specialize in multimodal research for images, videos, code, text, etc. The main idea is to structure our knowledge of RAG by creating a taxonomy for different characteristics of the RAG system in order to make their development easier for further RAG applications.

Classification research and typologies are used for the scientific pursuit of differences to theorize about commonalities (Beaulieu et al., 2015). Classification schemes and theories of typologies originate in biology to study and classify species, but have since gained widespread adoption in application-oriented research domains like the information systems community (Nickerson et al., 2013). While sometimes typologies are synonymous with the term framework, in this paper we will use the term taxonomy to structure the novel technological artifact known as RAG. Thus, we describe our methodology to develop a RAG application taxonomy following a systematic approach based on Nickerson et al. (2013).

3 Taxonomy Development

In this section, we describe the methodology used for developing the taxonomy of RAG. We describe the process of paper selection, as well as the iterations of the methodology applied. Moreover, we describe the additional approach for identifying the domains where RAG was applied – an analysis of papers found with specific queries in Google Scholar and ACL Anthology using ChatGPT.

3.1 Methodology

Following the approach of Nickerson et al. (2013), we defined our meta-characteristic as "structure and applications of retrieval augmented generation". In doing so, we included conceptual work and case studies that aimed at specific application domains or use cases. Due to the high dynamic in the research area, we also included pre-print articles. To ensure minimal quality standards, we reviewed each article by two independent researchers. To define a level of saturation, we used the objective and subjective ending conditions as proposed by Nickerson et al. (2013). So, we examined a representative sample of the most recent literature, and those dimensions were stable for an iteration. Thus, no extensions, merges, or splits of characteristics were performed. We also ensured that dimensions have at least one characteristic, and those are directly derived from papers while being unique. Subjective ending conditions were considered as proposed. We aimed for comprehensiveness, robustness, conciseness, extensibility, and explainability. This is reflected in the adaptation of dimensions throughout the iterations as we joined, reorganized, and split categories. As we tackle a recent and ever-changing phenomenon, we conclude that an expanded set of dimensions, namely 16, is still useful for research and practice in the current state. Regarding robustness, we checked for a strict separation between dimensions as well as characteristics. Comprehensibility is ensured by our extensive approach. Extensibility and explainability were tackled by repeatedly applying examples to the taxonomy.

To start the development process, we choose a twofold approach. First, we used aspects of conceptual-to-empirical to catalyze the initial iteration (Nickerson et al., 2013). This was done by identifying relevant domains with the help of ChatGPT. Second, we also incorporated an empirical-to-conceptual approach that specifically used surveys

of RAG, as those already provide some cumulative knowledge of the field. Still, most are rather new and have not gone through a proper double-blind review process. Thus, in combining both approaches, we propose a new angle to deal with emerging topics. Afterward, we strictly follow the empirical-to-conceptual regime.

3.2 Paper Selection

As the object of interest is relatively novel, we employed a naïve approach for identifying and selecting papers. As a first step, we started with the search string "RAG & application" at Google Scholar. The results encompassed several articles from the Association of Computational Linguistics, which is a driver of the general development of LLMs, as well as RAG. Thus, we deemed the general search strategy as useful. Despite high-quality results, most papers are in pre-print status and have not been part of a thorough peer review. Still, we include those, after quality checks by the author team. If articles are perceived as questionable by one of the researchers, we employed a cross-check by at least another researcher on the team. Also, we excluded published work with publishers that do not comply with the minimal standards of our research community.

Due to the iterative approach taken, we reviewed twenty-eight papers in the taxonomy development process. While 20 papers have been reviewed and published in several venues, 8 papers are still in pre-print phase. However, we need to emphasize that all 8 papers were written in 2024 which means that they still can be accepted in the future.

3.3 ChatGPT Domain Identification

To facilitate the time-consuming process of paper analysis, we decided to apply it for identifying the application domains and further compare the outcome to see, whether such an automated technique applies to the taxonomy construction. First, we created two queries for Google Scholar – a large search system for academic papers[1]. The first query "application of rag" for the papers dating from 2023 (anywhere in the article) returned ninety-three results. The "rag application" query from 2023 (anywhere in the article) returned seventy-four papers. Additionally, we searched these two queries in ACL Anthology[2], and the first one returned two results, while five results are returned with the second one.

[1]All searches are done on a date, 11.09.2024

[2]https://aclanthology.org/

Then, inspired by Rafailov et al. (2023), we created a prompt to ChatGPT to cluster the extracted papers into domains. We formulated the prompt as follows:

```
You are a scientific assistant writing
a survey. Here below is a list of paper
names. Your task is to cluster those pa-
pers into domains. Name those domains
(it might be something like NLP,
or medicine).
```

After the prompt, we pasted names from the first query separated with the line separator. Based on the titles provided, the ChatGPT model identified the following 10 classes: (1) Artificial Intelligence and Natural Language Processing (AI & NLP); (2) Cybersecurity; (3) Medicine and Healthcare; (4) Business and Economics; (5) Education and Programming; (6) Social Sciences and Ethics; (7) Disaster and Risk Management; (8) Physics and Engineering; (9) Data Science and Knowledge Discovery; (10) Adversarial AI and Machine Learning.

When providing ChatGPT papers from the "rag application" query, the output is: (1) Artificial Intelligence and Language Models; (2) Legal and Justice; (3) Medicine and Healthcare; (4) Education and Pedagogy; (5) Engineering and Construction; (6) Data Science and Information Retrieval.

It is also important to emphasize that in addition to the class names, the model returned paper examples for each class, therefore, we were able to primarily check the correctness of the identified classes. Furthermore, during iterations, we expected to use a manual paper check to prove the ChatGPT clustering efficiency.

3.4 Iterations

We performed four iterations to build our initial RAG application taxonomy. Overall, we analyzed 28 papers, including 4 surveys on RAG, that already performed the extensive analysis and generalization of previous works (Li et al., 2022; Zhao et al., 2023; Gao et al., 2024; Zhao et al., 2024). Those papers already comprise over 2000 citations, resulting in more than twenty-eight papers involved in our study in total. Moreover, a major part of the dimensions was added during the first iteration, where two survey papers were analyzed. Therefore, we consider this number of iterations reasonable, which was also confirmed by meeting the following objective condition – no new dimensions or

characteristics were added, merged, or split in the iteration.

Our iterative development process for the RAG application taxonomy, as illustrated by Figure 2 in Appendix A started with an initial set of eleven dimensions, mapped across five meta-dimensions. In this first iteration, the key dimensions of the RAG Phase, Application Domain, Application Task, RAG Process, Paradigm, Retrieval Type, Retrieval Process, RAG Role, Modality, Evaluation Metrics, and Failures of RAG were established, each annotated with a specific number of characteristics indicated by the numbers in parentheses. In the second iteration, we enriched the taxonomy by analyzing seven additional papers, leading to the introduction of three new dimensions: LLM Status, Granularity, and Dataset. Concurrently, we refined several existing dimensions by either merging or adding new characteristics, reflecting deeper insights and broader coverage. By the third iteration, only two new dimensions were necessary: Application Architecture and Future Directions, indicating a nearing saturation in the scope of the taxonomy. Adjustments were made to five dimensions during this phase, demonstrating a trend toward the stabilization of the taxonomy's structure. The fourth iteration confirmed the saturation, as no new changes were made to the taxonomy, suggesting that the existing structure sufficiently captured the relevant aspects of RAG applications as evidenced by the literature. Across all iterations, the taxonomy has evolved to accommodate and anticipate the dynamic nature of RAG applications, ensuring its relevance and utility in future research.

4 Results

In this section, we present the final RAG taxonomy created from twenty-eight papers in four iterations. Figure 1 illustrates the entire taxonomy divided into five main components. In the following subsections, we describe each component in more detail.

4.1 General

The first group of dimensions is devoted to general aspects of RAG systems, regardless of specific structural aspects. Within this group, we identified three dimensions to represent the general aim and motivation of applied RAG.

D_1 Phase: The dimension subsumes the focus of RAG in place. It also relates to the evolving discourse – despite being a novel phenomenon. Research shows that there are three primary areas of application for RAG. The resulting characteristics are pre-training, inference, and fine-tuning (Gao et al., 2024).

D_2 Application Domain: In total, eight application domains are found in the discourse. The most frequent are health, law, biology, general AI and NLP, as well as ecology (Gao et al., 2024). Further application domains are education and research (Barnett et al., 2024) as well as media (Siriwardhana et al., 2023).

D_3 Application Task: RAG methods can be applied to different applications or downstream tasks. In this dimension, the characteristics are the prominent tasks (Gao et al., 2024). We consider eight characteristics, i.e. Question-Answering (QA), Information Extraction (IE), Dialog, Reasoning, Slot Filling (Glass et al., 2021), Machine Translation (Li et al., 2022), Summarization (Zhao et al., 2024), others. QA has several different types, e.g. open domain QA, abstractive QA (Lewis et al., 2020), GraphQA (He et al., 2024), etc. Other contains some other tasks, for example, Fact Checking/Verification, Question Generation (Lewis et al., 2020), Code search (Gao et al., 2024), and many more.

4.2 Structure

This includes an examination of the underlying technologies that form the architecture of the RAG application, determining whether the RAG acts as the principal system or merely a component within a larger system. We further delineate the structure of RAG systems by analyzing different RAG paradigms—such as naive, advanced, and modular RAG—which reflect varying levels of complexity and integration. Additionally, we consider the specific contexts or processes where the RAG retrieval is realized. In total, the structure includes key characteristics that distinguish RAG systems.

D_4 Retrieval Process: The retrieval process represents to what extent the RAG uses retrieval. Single retrieval, multiple retrieval, and adaptive retrieval are the identified characteristics (Gao et al., 2024). Single retrieval thus solely relies on a single retrieval sequence in a RAG, while multiple retrieval is an iterative or sequential approach. Adaptive retrieval is the most contextual approach, as it integrates the results of prior retrieval to adapt the next iteration of retrieval.

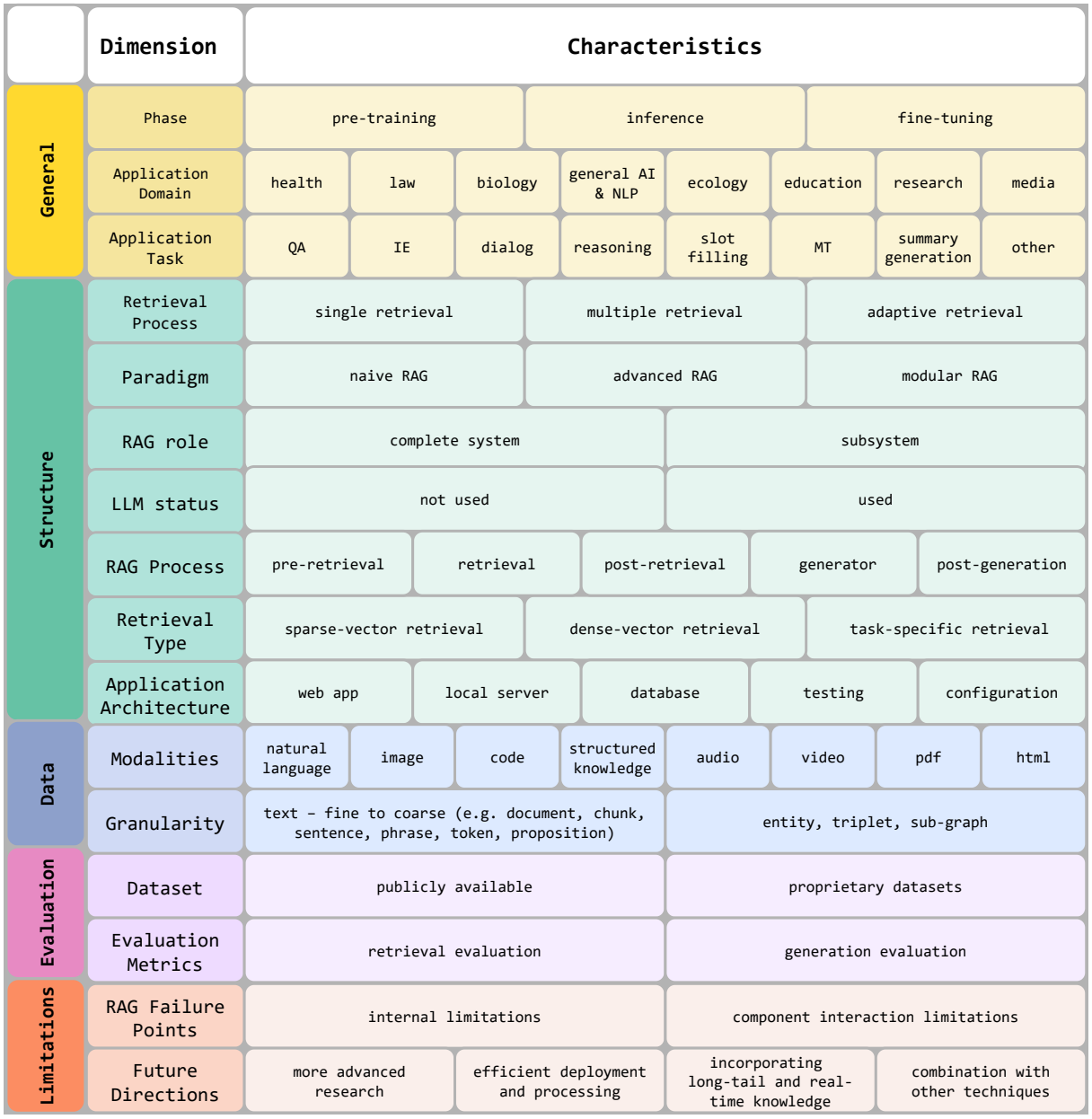

Figure 1: RAG Taxonomy created from twenty-eight papers within four iterations.

D_5 Paradigm: Gao et al. (2024) categorize the RAG research paradigm into three Naive RAG, Advanced RAG, and Modular RAG. In this dimension, these three categories are the characteristics. In Naive RAG, there are three parts, i.e., indexing, retrieval, and generation. Advance RAG also includes pre-retrieval and post-retrieval parts before and after retrieval. Modular RAG provides flexibility with different modules, e.g., search module, memory module, etc.

D_6 RAG Role: Within the application landscape, the role of RAG systems can vary significantly: they can operate as dedicated, monolithic systems or as modular components integrated within other application systems (Zhao et al., 2024). According to the authors, subsystems can be part of larger architectures that employ multiple frameworks, i.e. RetDream for 3D Generation (Seo et al., 2024), R-ConvED for video captioning (Chen et al., 2023), and kNN-TRANX (Zhang et al., 2023) for text-to-code tasks. In the above-mentioned systems, RAG is used as an additional step to the pipeline, enhancing generation with the retrieved data.

D_7 LLM Status: This dimension is binary, it checks for the adaptability of the LLM in place. So it can either be not used, meaning no further approach is taken to improve the LLM performance, or it can be used (Chen et al., 2024). Used thus leads to different forms. It can be trainable to be adjusted in each context or it can be looped as a specification of the paradigm modular RAG.

D_8 RAG Process: In this dimension, processes in RAG models are discussed mostly based on the information by Gao et al. (2024). Five characteristics are considered, i.e., pre-retrieval, retrieval, post-retrieval, generation, and post-generation. The pre-retrieval step involves some techniques applied before the retrieval step, for instance, chunking, vectorizing, indexing, and some other strategies to e.g., optimize indexing, enhance user input, etc. In the retrieval step, the relevant information to the user input is retrieved. Post-retrieval includes methods to improve the retrieved information during integration with user input, e.g. re-rank the information or subgraph construction (He et al., 2024). In the generation step, LLM provides a response to the prompt that contains the retrieved information and user input. Post-generation contains strategies that can be applied after generation, e.g. output rewrite (Zhao et al., 2024). Note that there exist various modules to enhance different components (see Gao et al. (2024) for more information).

D_9 Retrieval Type: There are different types of retrieval augmentation methods (Li et al., 2022). In this dimension, three characteristics are considered, i.e., sparse-vector retrieval, dense-vector retrieval, and task-specific retrieval (Li et al., 2022). Sparse-vector retrieval involves methods, e.g., TF-IDF, BM25, etc. Dense-vector retrieval contains models based mostly on low-dimensional dense vectors, e.g., BERT-encoders, and relying often on vector databases. In task-specific retrieval, the retrieval module is based on task (Li et al., 2022) and might comprise a database (Radeva et al., 2024). Some research works directly using the edit distance between natural language texts (Hayati et al., 2018) or abstract syntax trees (AST) of code snippets (Poesia et al., 2022).

D_{10} Application Architecture: When developing a RAG system, in addition to the RAG structure, we also need to consider the structure of the final application and the interaction of the components. (Radeva et al., 2024) present a web application RAG system that consists of the "local or server-based installation", "web application", "vector storage" (database), as well as the testing and configurations. Therefore, this dimension comprises the web app, local server, database, testing, and configuration characteristics.

4.3 Data

D_{11} Modalities: Although the concept of RAG was originally developed for text-based generation, its use has been adapted for a variety of other generation modalities (Chen et al., 2024; Gao et al., 2024; Lewis et al., 2020; Zhao et al., 2024). This includes programming code, audio, visual content, such as images and videos, 3D models, and other knowledge structures. The latter can include table structures, higher-level modeling languages, graphs, textual graphs (He et al., 2024), or knowledge graphs. The fundamental principles of RAG remain similar across these different modalities, even though slight modifications to the augmentation methods are sometimes required.

D_{12} Granularity: This dimension is for different granularities of retrieved data based on the information by Gao et al. (2024). The modality can be natural language (or text), yet still, the retrieved granularity might vary from fine to coarse, e.g., document, chunk, sentence, proposition, etc. (Gao et al., 2024). Similarly, there exist several granularities in structured data, e.g., sub-graph, triplet, entity, etc. (Gao et al., 2024).

4.4 Evaluation

D_{13} Dataset: Regarding the datasets used for RAG, most RAG surveys consistently list the datasets used regardless of the application task, the RAG step to be evaluated on as well as the dataset availability. Thus, we focused on the matter of availability and considered two characteristics, i.e., publicly available, and proprietary datasets. Some examples of publicly available datasets are e.g. FEVER (Thorne et al., 2018), SQuAD (Rajpurkar et al., 2016) etc., and the dataset, e.g. by Bondarenko et al. (2020), is an example for proprietary datasets.

D_{14} Evaluation Metrics: When reviewing papers discussing separate models and architectures, we can see that the authors mostly use task-specific metrics (Thakur and Vashisth, 2024) or the generation output quality only (Chen et al., 2024). However, Gao et al. (2024) split evaluation metrics into

two groups: retrieval evaluation and generation evaluation metrics, which are the base parts of RAG. The first group evaluates the relevance of the retrieved data to the query and is mostly represented with the ranking evaluation metrics: Precision@k, Recall@k, F@1, MRR, MAP (Gao et al., 2024). The second group involves generation evaluation metrics, such as BLEU, METEOR, ROUGE, PPL (Radeva et al., 2024) and Accuracy, Rejection Rate, Error Detection Rate, Error Correction Rate (Chen et al., 2024).

4.5 Limitation

Despite multiple advantages and ubiquitous application, Zhao et al. (2024) outline limitations and possible directions of RAG. We describe the last two dimensions in more detail, also considering failures from Barnett et al. (2024).

D_{15} RAG Failure Points: RAG limitations can be divided into two groups: internal (related to the system component efficiency) and integration (related to the problems of RAG components interaction). Here, we discuss each type separately.

The most evident and the most frequent failure point for RAG is the retrieval step. Noises in retrieval results or missing relevant content may drastically decrease the final performance, as the information provided to the generator may contain irrelevant objects or misleading information. Barnett et al. (2024) also state that the reason for that might be the missing content, e.g., "when asking a question that cannot be answered from the available documents". The next failure point is called "not in context" Barnett et al. (2024). In this case, the extracted documents were not correctly consolidated during the post-retrieval process. The last three failure points relate to the generated output: the incorrect format of the output, incorrect specificity ("not specific enough or is too specific to address the user's need"), and incomplete output that misses essential information even though being extracted by the retriever.

When combining RAG with another system, the most common limitation is extra overhead: additional retrieval and interaction processes lead to increased latency of the system. Moreover, speed time also depends on the gap between retrievers and generators: the integration process and increased system complexity might be other bottlenecks that should be considered. When applying RAG to LLMs or other generators with a limited context size, lengthy context might become a problem: the models might not be able to accept the whole retrieved data as input and the generation process will take much more time than expected.

D_{16} Future Directions: The last dimension outlines future directions for the RAG systems based on the findings of Zhao et al. (2024). The most straightforward directions are further development of RAG methodologies, enhancements, and applications. This might include new interactions between the retriever and generator, various modular RAG architectures with looping, and more advanced pre- and post-processing steps. Another direction is efficient deployment and processing. When discussing limitations, most of the integration limitations were related to efficiency and latency. Hopefully, future research on RAG capacities will allow shorter system response time and easier deployment. Another important research direction is the incorporation of long-tail and real-time knowledge. With the rapid growth of the data, it is extremely difficult to constantly update large retrievals in RAG. Many existing works apply a static database for knowledge retrieval, which requires re-indexing and/or computing additional representations. Zhao et al. (2024) expect newer techniques to solve the issue, as well as provide better retrieval of less commonly referenced data. Lastly, the combination of other techniques might be also seen as a promising direction, e.g. integration of RAG with the new state space model architecture like Mamba (Gu and Dao, 2024) or RWKV (Peng et al., 2023).

5 Discussion

RAG application systems are an emerging technology that has received considerable attention outside the NLP community, which addresses the limitations of LLM applications (Gao et al., 2024; Leiser et al., 2024). While recent studies address both the applications of LLMs and methods to mitigate their shortcomings, RAGs have not yet been fully recognized outside NLP community or explored as a potential solution to these limitations. Thus, our taxonomy provides a basis for applying RAGs as an emerging technology for novel fields of application. Based on our taxonomy, we see that the broader community can engage in a socio-technical perspective for guiding future RAG applications.

Domain-specific applications: Our taxonomy shows that different mediums of generation are

gaining interest, including conceptual modeling approaches (Baumann et al., 2024). For example, process modeling already leverages generative AI for improving and (re-) designing organizational processes (van Dun et al., 2023). RAGs appear to make such application systems much more viable, as our taxonomy provides us an example, where knowledge structure already incorporates conceptual process modeling types (Baumann et al., 2024). Thus, we see potential in incorporating RAGs into design science research endeavors (Hevner et al., 2004; Peffers et al., 2007; Teixeira et al., 2019) to address a variety of domain-specific applications. In our analyses, we identified proof-of-concepts and have seen little research on practical RAG applications. Thus, we call for future research to address this lack of proof-of-value (Nunamaker et al., 2015).

Business Value of RAG Applications: The impacts of AI have largely been due to increasing business value following business processes (Davenport and Ronanki, 2018). Research into conversational agents has shown that they can lead to tangible business value (Kull et al., 2021; Mariani et al., 2023; McLean et al., 2021), whereas the assessment of LLM-based impact of business value remains under-studied (Storey et al., 2024). This could be attributed to current restraints of LLMs. However, with RAGs, there might be legitimate potential to create tangible business value, be it by also addressing knowledge- and labor-intensive services or improving work conditions.

Ethical, legal, social implications: Most current research is moving towards sustainability, including calls for studying social value (Nunamaker et al., 2015) and putting the need for reflecting on values in system designs (e.g.: Bednar and Spiekermann (2023); Friedman et al. (2013)). With LLMs already being discussed with ecological inefficiencies due to their potential high carbon emissions, integrating RAGs to improve LLM applications might have unforeseen consequences. Similarly, we see ongoing discussions on the societal implications of improving works systems, leading to potential job losses (e.g. Brynjolfsson et al. (2023)) or legal disputes about leveraging copyrighted content to generate new content (Golatkar et al., 2024; Samuelson, 2023). Integrating RAGs can improve either, yet its ethical, legal, and social implications require careful consideration, exemplifying its socio-technical nature.

Digital transformation and RAG implementations: The challenges of adopting technologies, including AI (Grønsund and Aanestad, 2020), firms are facing include high resistance to change and organizational barriers (Vial, 2019). Since RAG applications address traditionally knowledge-intensive tasks, such as analysis and interpretation of data, aiming to outperform human capabilities, we see the potential that RAG applications can lead to increased organizational resistance, especially when integrated into existing Enterprise applications. RAGs can play a considerable role as orchestrators of enterprise application systems (Böhmann et al., 2014) to call each functionality as part of its retrieval and generate respective outputs. This increasing complexity of heterogeneous applications might lead to new challenges for digital transformation.

Thus, the taxonomy is broad with sixteen dimensions. As the field is still evolving, we deem this initial breadth beneficial to shape our community understanding. While the field is settling and maturing, a narrowed-down taxonomy could be the next step, to further increase the conciseness and applicability, especially for practice. As of now, expert knowledge is still needed to assess several details within the taxonomy.

6 Conclusion

Our RAG taxonomy provides a structured way to categorize and analyze the diverse approaches, system features, and technologies that constitute RAG applications. Thus, we contribute to a clearer understanding of its components and their interactions. The taxonomy has five meta-dimensions, sixteen dimensions, and sixty-one characteristics, reflecting the inherent complexities of current RAGs. This systematic classification is essential for different researchers and practitioners to identify gaps in the current technology, facilitate research and development efforts, and identify potential use cases for real-world applications. Based on our taxonomy, we also present several avenues for future research, accommodating the RAG characteristics for different application types. Overall, the taxonomy not only enriches the academic discourse by providing a foundational framework for study and discussion but also guides practical implementations and innovations within the field.

7 Limitations

Our taxonomy development approach has several limitations, which can be attributed to the novelty of our phenomenon of interest. Additionally, while dealing with pre-prints in a fast-moving research field, papers get updated while working with them, leading to inconsistencies for the research team, that need to be reworked afterward. Furthermore, we do not claim completeness, as the field is quickly moving forward, and we aim to capture an initial view of the emerging phenomenon, calling for future taxonomy extensions.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *ICLR*. OpenReview.net.

Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. 2024. Seven failure points when engineering a retrieval augmented generation system. In *CAIN*, pages 194–199. ACM.

Nils Baumann, Juan Sebastian Diaz, Judith Michael, Lukas Netz, Haron Nqiri, Jan Reimer, and Bernhard Rumpe. 2024. Combining retrieval-augmented generation and few-shot learning for model synthesis of uncommon dsls. Modellierung 2024 Satellite Events.

Tanya Beaulieu, Suprateek Sarker, and Saonee Sarker. 2015. A conceptual framework for understanding crowdfunding. *Communications of the Association of Information Systems*, 37. [Online; accessed 2024-05-01].

Kathrin Bednar and Sarah Spiekermann. 2023. The power of ethics: Uncovering technology risks and positive value potentials in it innovation planning. *Business & Information Systems Engineering*. [Online; accessed 2024-05-01].

Alexander Bondarenko, Pavel Braslavski, Michael Völske, Rami Aly, Maik Fröbe, Alexander Panchenko, Chris Biemann, Benno Stein, and Matthias Hagen. 2020. Comparative web search questions. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, WSDM '20, page 52–60, New York, NY, USA. Association for Computing Machinery.

Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. 2023. Generative ai at work. Technical report, NATIONAL BUREAU OF ECONOMIC RESEARCH, Cambridge, MA. DOI: 10.3386/w31161.

Julia Bräker, Julia Hertel, and Martin Semmann. 2022. Conceptualizing interactions of augmented reality solutions. In *Proceedings of the 55th Hawaii International Conference on System Sciences*. Accepted: 2021-12-24T17:22:28Z.

Tilo Böhmann, Jan Marco Leimeister, and Kathrin Möslein. 2014. Service systems engineering. *Business & Information Systems Engineering*, 6(2):73–79. Tex.ids= Böhmann2014Service, bohmannServiceSystemsEngineering2014, bohmannServiceSystemsEngineering2014b.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17754–17762.

Jingwen Chen, Yingwei Pan, Yehao Li, Ting Yao, Hongyang Chao, and Tao Mei. 2023. Retrieval augmented convolutional encoder-decoder networks for video captioning. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 19(1s):48:1–48:24.

Thomas H. Davenport and Rajeev Ronanki. 2018. Artificial intelligence for the real world. *Harvard business review*, 96(1):108–116.

Batya Friedman, Peter H. Kahn, Alan Borning, and Alina Huldtgren. 2013. Value sensitive design and information systems. In Neelke Doorn, Daan Schuurbiers, Ibo van de Poel, and Michael E. Gorman, editors, *Early engagement and new technologies: Opening up the laboratory*, pages 55–95. Springer Netherlands, Dordrecht. DOI: 10.1007/978-94-007-7844-3_4.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Michael R. Glass, Gaetano Rossiello, Md. Faisal Mahbub Chowdhury, and Alfio Gliozzo. 2021. Robust retrieval augmented generation for zero-shot slot filling. In *EMNLP (1)*, pages 1939–1949. Association for Computational Linguistics.

Aditya Golatkar, Alessandro Achille, Luca Zancato, Yu-Xiang Wang, Ashwin Swaminathan, and Stefano Soatto. 2024. CPR: Retrieval Augmented Generation for Copyright Protection. *Preprint*, arXiv:2403.18920.

Tor Grønsund and Margunn Aanestad. 2020. Augmenting the algorithm: Emerging human-in-the-loop work configurations. *The Journal of Strategic Information Systems*, 29(2):101614.

Albert Gu and Tri Dao. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *Preprint*, arXiv:2312.00752.

Shirley Anugrah Hayati, Raphael Olivier, Pravalika Avvaru, Pengcheng Yin, Anthony Tomasic, and Graham Neubig. 2018. Retrieval-based neural code generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 925–930, Brussels, Belgium. Association for Computational Linguistics.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *Preprint*, arXiv:2402.07630.

Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. Design science in information systems research. *Mis Quarterly*, 28(1):75–105.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *EMNLP*, pages 7969–7992. Association for Computational Linguistics.

Alexander J. Kull, Marisabel Romero, and Lisa Monahan. 2021. How may i help you? driving brand engagement through the warmth of an initial chatbot message. *Journal of Business Research*, 135:840–850.

Florian Leiser, Sven Eckhardt, Valentin Leuthe, Merlin Knaeble, Alexander Mädche, Gerhard Schwabe, and Ali Sunyaev. 2024. HILL: A hallucination identifier for large language models. In *CHI*, pages 482:1–482:13. ACM.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *NeurIPS*.

Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A survey on retrieval-augmented text generation. *Preprint*, arXiv:2202.01110.

Yuxin Liang, Zhuoyang Song, Hao Wang, and Jiaxing Zhang. 2024. Learning to trust your feelings: Leveraging self-awareness in LLMs for hallucination mitigation. In *Proceedings of the 3rd Workshop on Knowledge Augmented Methods for NLP*, pages 44–58, Bangkok, Thailand. Association for Computational Linguistics.

Marcello M. Mariani, Novin Hashemi, and Jochen Wirtz. 2023. Artificial intelligence empowered conversational agents: A systematic literature review and research agenda. *Journal of Business Research*, 161:113838.

Ariana Martino, Michael Iannelli, and Coleen Truong. 2023. Knowledge injection to counter large language model (llm) hallucination. In *The Semantic Web: ESWC 2023 Satellite Events*, pages 182–185, Cham. Springer Nature Switzerland.

Graeme McLean, Kofi Osei-Frimpong, and Jennifer Barhorst. 2021. Alexa, do voice assistants influence consumer brand engagement? – examining the role of ai powered voice assistants in influencing consumer brand engagement. *Journal of Business Research*, 124:312–328.

Robert C Nickerson, Upkar Varshney, and Jan Muntermann. 2013. A method for taxonomy development and its application in information systems. *European Journal of Information Systems*, 22(3):336–359. ZSCC: 0000597.

Noa Nonkes, Sergei Agaronian, Evangelos Kanoulas, and Roxana Petcu. 2024. Leveraging graph structures to detect hallucinations in large language models. In *Proceedings of TextGraphs-17: Graph-based Methods for Natural Language Processing*, pages 93–104, Bangkok, Thailand. Association for Computational Linguistics.

Jay F. Nunamaker, Robert O. Briggs, Douglas C. Derrick, and Gerhard Schwabe. 2015. The last research mile: Achieving both rigor and relevance in information systems research. *Journal of Management Information Systems*, 32(3):10–47.

Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2007. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Gv, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan Wind, Stanisław Woźniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. 2023. RWKV: Reinventing RNNs for the transformer era. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077, Singapore. Association for Computational Linguistics.

Gabriel Poesia, Alex Polozov, Vu Le, Ashish Tiwari, Gustavo Soares, Christopher Meek, and Sumit Gulwani. 2022. Synchromesh: Reliable Code Generation from Pre-trained Language Models. In *International Conference on Learning Representations*.

Irina Radeva, Ivan Popchev, Lyubka Doukovska, and Miroslava Dimitrova. 2024. Web application for retrieval-augmented generation: Implementation and testing. *Electronics*, 13(7).

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Gerrit Remane, Rob Nickerson, Andre Hanelt, Jan Tesch, and Lutz Kolbe. 2016. A taxonomy of carsharing business models. In *ICIS 2016 Proceedings*.

Pamela Samuelson. 2023. Generative ai meets copyright. *Science*, 381(6654):158–161.

Junyoung Seo, Susung Hong, Wooseok Jang, Inès Hyeonsu Kim, Minseop Kwak, Doyup Lee, and Seungryong Kim. 2024. Retrieval-augmented score distillation for text-to-3d generation. In *ICML*. OpenReview.net.

Shamane Siriwardhana, Rivindu Weerasekera, Elliott Wen, Tharindu Kaluarachchi, Rajib Rana, and Suranga Nanayakkara. 2023. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17. Publisher: MIT Press One Broadway, 12th Floor, Cambridge, Massachusetts 02142, USA ....

Heydar Soudani, Evangelos Kanoulas, and Faegheh Hasibi. 2024. Fine tuning vs. retrieval augmented generation for less popular knowledge. *Preprint*, arXiv:2403.01432.

Veda C. Storey, Alan R. Hevner, and Victoria Yoon. 2024. The design of human-artificial intelligence systems in decision sciences: A look back and directions forward. *Decision Support Systems*, page 114230.

Jorge Grenha Teixeira, Lia Patrício, and Tuure Tuunanen. 2019. Advancing service design research with design science research. *Journal of Service Management*, 30(5):577–592. Citation Key: Teixeira.2019.

Ayush Thakur and Rashmi Vashisth. 2024. Loops On Retrieval Augmented Generation (LoRAG). *Preprint*, arXiv:2403.15450.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Christopher van Dun, Linda Moder, Wolfgang Kratsch, and Maximilian Röglinger. 2023. Processgan: Supporting the creation of business process improvement ideas through generative machine learning. *Decision Support Systems*, 165:113880.

Gregory Vial. 2019. Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems*, 28(2):118–144.

Xiangyu Zhang, Yu Zhou, Guang Yang, and Taolue Chen. 2023. Syntax-aware retrieval augmented code generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1291–1302, Singapore. Association for Computational Linguistics.

Penghao Zhao, Hailin Zhang, Qinhan Yu, Zhengren Wang, Yunteng Geng, Fangcheng Fu, Ling Yang, Wentao Zhang, Jie Jiang, and Bin Cui. 2024. Retrieval-augmented generation for ai-generated content: A survey. *Preprint*, arXiv:2402.19473.

Ruochen Zhao, Hailin Chen, Weishi Wang, Fangkai Jiao, Xuan Long Do, Chengwei Qin, Bosheng Ding, Xiaobao Guo, Minzhi Li, Xingxuan Li, and Shafiq Joty. 2023. Retrieving multimodal information for augmented generation: A survey. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4736–4756, Singapore. Association for Computational Linguistics.

A Appendix

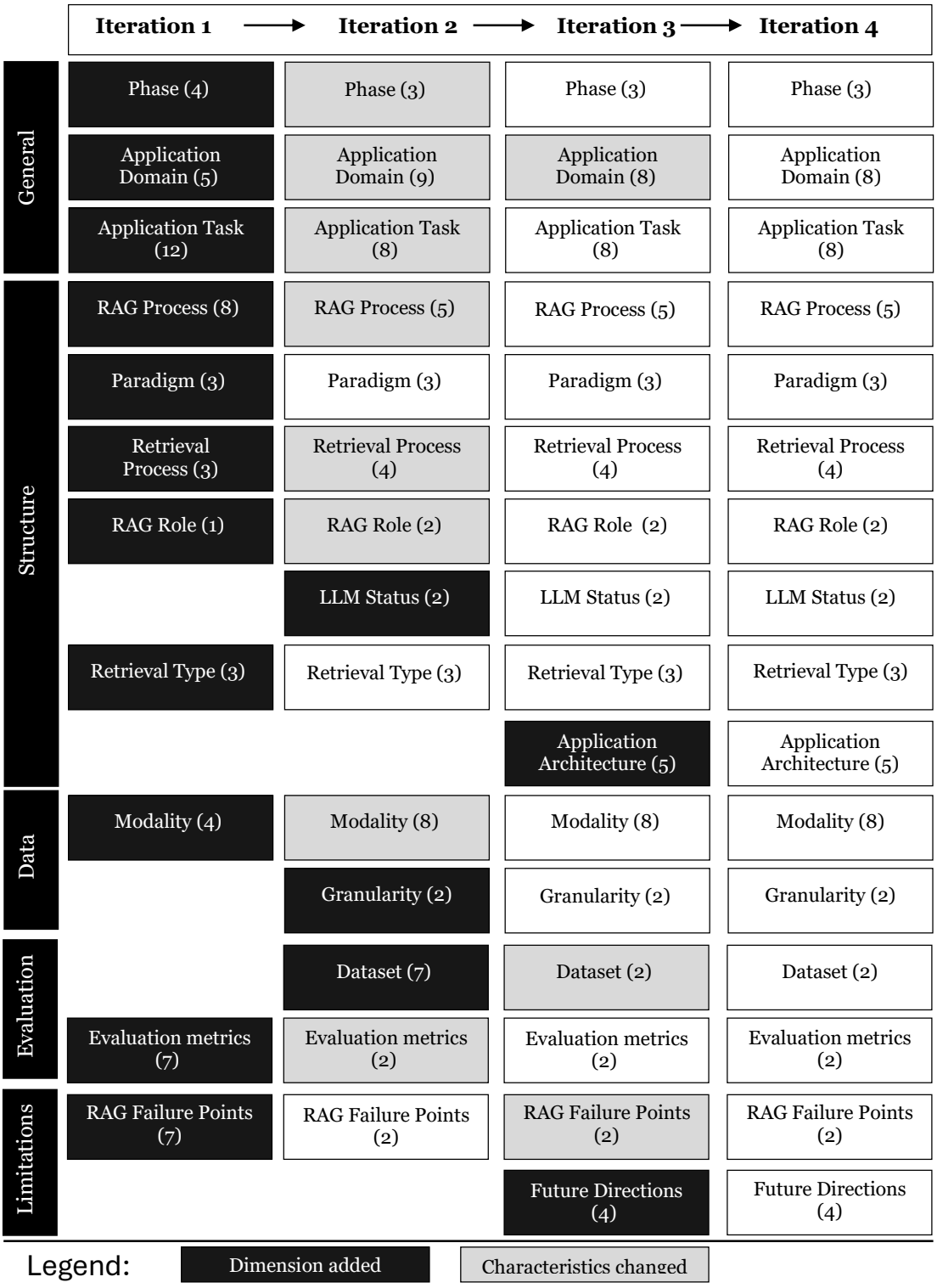

Figure 2: Development of taxonomy dimensions and characteristics (adapted from Bräker et al. (2022); Remane et al. (2016))