Peer-Review: 20.04.2024

From data to design

LLM-enabled information extraction across industries

Robert Becker, Laura Steffny, Thomas Bleistein, Dirk Werth, August-Wilhelm Scheer Institut für digitale Produkte und Prozesse gGmbH

This paper explores the application of Large Language Models (LLMs) in the automotive and supplier industries, with a particular focus on the use of retrieval-augmented generation (RAG) systems to streamline information retrieval from technical documentation. The research, part of the CoLab4DigiTwin project, investigates how digital twins supported by smart services can enhance interdisciplinary collaboration and reduce the reliance on manual data searches. We developed a pipeline utilizing a RAG architecture which uses a vector database for efficient data management and fast access to relevant information, eliminating the need for expensive computational resources. The performance of various open-source LLMs, which are fine-tuned on German, was evaluated, focusing on readability, clarity, and accuracy. The results show decent performance of the system without the need for model fine-tuning. Future research will aim to refine these processes and extend the applicability of RAG systems, highlighting the potential of Large Language Models to transform industrial data interaction.

#Large Language Model #Retrieval-Augmented Generation #Python #Efficiency

Von Daten zu Design

LLM-gestützte Informationsextraktion über Branchen hinweg

In diesem Beitrag wird die Anwendung von Großsprachmodelle (Large Language Models, LLMs) in der Automobil- und Zuliefererindustrie untersucht, wobei der Schwerpunkt auf dem Einsatz von Retrieval-Augmented-Generating-Systemen (RAG) zur Rationalisierung der Informationsbeschaffung aus technischen Unterlagen liegt. Die Forschungsarbeit, die Teil des CoLab4-DigiTwin-Projekts ist, untersucht, wie digitale Zwillinge, die von intelligenten Diensten unterstützt werden, die interdisziplinäre Zusammenarbeit verbessern und die Abhängigkeit von der manuellen Datensuche verringern können. Wir haben eine Pipeline mit einer RAG-Architektur entwickelt, die eine Vektordatenbank für ein effizientes Datenmanagement und einen schnellen Zugriff auf relevante Informationen nutzt, wodurch teure Rechenressourcen überflüssig werden. Die Leistung verschiedener Open-Source-LLMs, die auf Deutsch abgestimmt sind, wurde bewertet, wobei der Schwerpunkt auf Lesbarkeit, Klarheit und Genauigkeit lag. Die Ergebnisse zeigen eine gute Leistung des Systems, ohne dass eine Feinabstimmung des Modells erforderlich ist. Zukünftige Forschung wird darauf abzielen, diese Prozesse zu verfeinern und die Anwendbarkeit von RAG-Systemen zu erweitern, um das Potenzial von Large Language Models für die Interaktion mit industriellen Daten zu verdeutlichen.

#Large Language Model #Retrieval-Augmented Generation #Python #Effizienz

1. Introduction

The advancing digitalization and automation significantly shape the industry, especially the automotive and its supplier industry. In this dynamic environment the collaboration of different companies from different sectors, including architects, structural engineers, engineers, plant designers, and automation technicians has become increasingly important. In addition, these sectors face the challenge of processing extensive specification sheets and requirement catalogues, which often contain several thousand pages in different formats such as PDF, Word, and CSV. Furthermore, these documents are often redundant or contradictory. Besides, the requirements vary from project to project, further complicating the search and extraction processes. Studies have shown that these professional groups spend more than 20% of their time searching for information from e-mails or other internal sources [1]. Within the research project CoLab4DigiTwin (Grant number: 13IK013F), which aims to enable interdisciplinary collaboration through a Digital Twin, this problem will be addressed. With the help of smart services, it should be possible in the future to save personnel resources and process unstructured data. Automated support can be a valuable addition to automotive plant engineering, but it can also be beneficial in other sectors. For instance, artificial intelligence (AI) can be used to extract specific information from a large amount of unstructured or structured data. This allows for more efficient use of time, directing it towards the processes that require skilled workers and mitigating the impact of their shortage.

In this context, pre-trained Large Language Models (LLMs) [2] offer a promising solution. These AI models are designed to understand natural language and recognize complex relationships. If the data is present LLMs can be trained to fulfill several tasks, including question answering. Thus, LLMs can assist engineers and planners by providing in-domain

knowledge in natural language. This opens new possibilities for more efficient work processes and better resource utilization, particularly in the field of automation technology.
However, the integration of LLMs into the field of automation technology raises various research questions and challenges [3]. For example, the development and/or training of specific models for handling technical knowledge is required to ensure precise extraction of relevant information. Here vast amount of training data must be available and pre-processed before it can be used for training an LLM. Moreover, mechanisms for ensuring data integrity and security must be implemented to minimize potential risks associated with the use of sensitive corporate data. Additionally, optimizing question answering processes of LLM systems is crucial to ensuring high accuracy and reliability of extraction results while reducing the need for manual verification. Recent developments have introduced the so-called retrieval-augmented generation (RAG) approach (Figure 1) [4]. RAG enables the use of data from external sources, thereby simplifying data management and decreasing the potential for security risks when training LLMs with sensitive data or using other third-party LLMs like the GPT series from OpenAI. These aspects are the focus of current scientific investigations and developments in the field of AI-supported data processing in the automotive and supplier industry.
Overall, the use of LLMs offers promising perspectives for enhancing efficiency and optimizing work processes within the automotive and supplier industry. Through the intelligent processing of unstructured data, engineers and planners can access relevant information more quickly and make informed decisions. Current research activities focus on maximizing the potential of LLMs while addressing potential challenges to successfully integrate this technology into industrial applications. This not only offers the opportunity to significantly reduce the effort involved in information retrieval but also contributes to increasing the competitiveness and innovation capability of companies in the automotive and supplier industry.

2. State-of-the-Art

Tasks requiring the gathering or access of knowledge and information from documentation, manuals etc., i.e. involving knowledge management, have a significant time loss due to non-productive queries. According to a McKinsey study, employees use an average of 1.8 hours daily for searching and gathering information [1]. This equates to approximately one-fifth of employees being non-productive, engaged solely in information retrieval rather than focusing on productive tasks. Additionally, an IDC report indicates that employees spend about 2.5 hours per day, or 30% of their workday, on similar tasks [5]. The time investment in information searches highlights a critical point for improvements within organizations, particularly within sectors like manufacturing where efficient information access and management are directly tied to productivity and operational efficacy.
Generative AI is a promising and rapidly evolving approach that has already demonstrated its enormous potential in

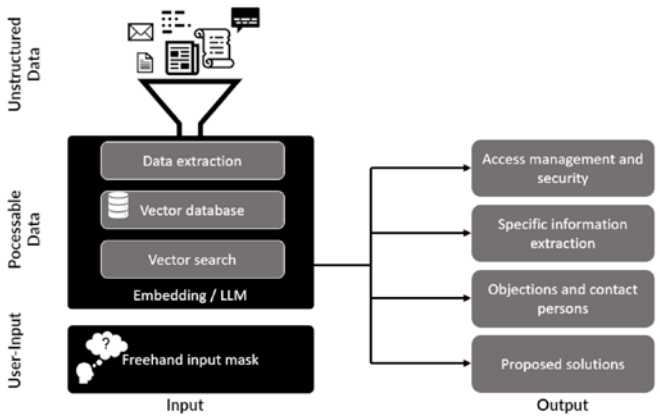


Figure 1: A simplified architecture for a retrieval-based system. Documents can be stored in a vector database and then used by a LLM to create answers to a query.

recent years. It can address the challenges associated with information search and gathering, which are crucial activities during the daily work. Especially, in sectors such as academia, engineering and communications, AI is already seen as a valuable tool for increasing work productivity by not only searching for information but also for writing reports and other text-heavy tasks. A recent review summarized the potentials in different areas, such as agriculture, healthcare, communication, and business management. Usually, the AI is used as a chatbot agent, where a user is able to communicate in natural language with the AI agent by asking relevant questions [6].
The backbone of such chatbots is most commonly a LLM, which themselves are based on the transformer architecture of neural networks. LLMs are advanced AI systems trained on extensive text corpora to generate, understand, and interact with human language. These models leverage deep learning techniques, particularly transformer architectures, to process and produce text in a contextually relevant manner. LLMs are instrumental in a variety of applications, ranging from natural language processing tasks to complex decision-making processes in diverse fields [7], [8].
The transformer architecture was first introduced by Vaswani et al. in the paper “Attention is All You Need” [2]. The innovation of the transformer is the self-attention mechanism that allows the model to weigh the significance of different words in a sentence, irrespective of their positional distance from each other. This feature enables the model to capture complex syntactic and semantic dependencies more effectively than previous architectures that relied on recurrent layers [9].
The architecture of transformers is composed of an encoder and/or a decoder. The encoder processes the input text by generating representations for both the content and context of the input sequence. Then, the decoder predicts the output sequence, guided by the encoder’s output and previous decoder outputs. This design allows for significantly parallelized processing, which decreases training times and enables the handling of long data sequences more efficiently than earlier models.

Models, such as the Generative Pre-trained Transformer (GPT) series by OpenAI [10], are built on this architecture. These models are pre-trained on a diverse corpus using unsupervised learning techniques to generate a model that can then be fine-tuned for a variety of specific tasks. The ability of GPT models to generate coherent and contextually relevant text has made them highly effective across a range of applications, including automated text completion and question answering (QA) systems. The GPT models from OpenAI have demonstrated remarkable performance improvements, underscoring the impact of the transformer architecture on the field of natural language processing.
However, the use of systems like ChatGPT raises concerns about both ethical implications and data security. For instance, when such systems incorporate a reinforcement system, the system can use the user's interactions with the chatbot to improve its own performance. This means that personal information might get built into the model's weights, which is often not desired for sensitive information. In this context, local open-source models represent a good alternative, as they can be tailored to specific domains and deployed in an encapsulated environment, preventing the leakage of sensitive information to third parties. However, the process of fine-tuning these models locally can be a very expensive task, requiring significant computational resources and large amount of training data.
Most recent practice use a RAG architecture for LLM-based smart document search, where the task is mostly question-answering on domain-specific knowledge. The advantage of such a system is, that the LLM which is used serves as a conversation agent to make retrieved information more accessible by the user. Although, RAG has its weaknesses, e.g., the limitation of the number of input tokens and the accuracy of the similarity search, the RAG architecture is quite attractive, because of the simple implementation and the out-of-the-box usage of LLMs without sharing sensitive data.
In a RAG system, the LLM in use gets access to external knowledge, usually stored in a vector database, to get information that the LLM has not been trained on. This is particularly useful and beneficial when dealing with unstructured data in an ever-changing industrial environment, because instead training the LLM onto new up-to-date data, it is sufficient to store the new data into a connected database or update the old database. Furthermore, it addresses two of the most challenging aspects of LLMs: the data security issue and the need for vast amounts of data and computational resources to fine-tune models.
In industrial settings, RAG systems can be particularly effective in searching through and making sense of vast amounts of unstructured data. For example, in the manufacturing sector, such systems can be used to quickly search through maintenance logs, operational manuals, and incident reports to provide troubleshooting support or operational insights. By querying the RAG system, engineers and technicians can obtain synthesized information that directly addresses their specific queries, thus speeding up decision-making and improving operational efficiency.
The latest research is using LLMs and, especially RAG systems in manufactural use-cases. For example, Freire et al. have tested a RAG system to deliver information from standard work instructions and machine manuals [11]. The research showed the superiority of the GPT-4 model over several other models. The user study noted the system's potential to modernize factory operations and accelerate tasks, although concerns were expressed about safety, efficiency, and comparisons to human experts.
Makatura et al. [12] investigated the application of LLMs in design and manufacturing, noting their potential to transform product development processes. The study addresses the integration challenges of balancing creativity with precise verification and outlines methods to reduce LLM limitations, such as user guidance and external APIs. Additionally, it explores workflow integration strategies and discusses ethical risks like job displacement and data security, proposing mitigation measures. The authors call for further research to fully harness LLM capabilities in this sector.
Also, Bornea et al. [13] introduced Telco-RAG, a novel RAG framework tailored for processing 3GPP telecommunications standards and enhancing LLM applications in telecom scenarios. The study emphasizes on possible performance improvements. They can be achieved through optimizations of chunk sizes, embedding models, indexing strategies, and query structuring. These refinements have proven effective in addressing common challenges in constructing RAG pipelines for technical domains. The Telco-RAG framework, made publicly available, is expected to significantly advance the integration of AI within the telecommunications sector.

3. Methology/Conception and Results

This study aims to develop and evaluate a pipeline for a document QA- system based on the RAG architecture in Python. A key aspect of the study is the use of open-source tools and models for all steps in a multilingual setting, focusing on German language. No fine-tuning or larger computational clusters are required. The solution runs on a single NVIDIA Tesla V100 with 16 GB VRAM. The pipeline consists of several steps: data extraction, preprocessing, text chunking, embedding generation, data retrieval and LLM response generation.

3.1 Architecture

In the following section, the individual steps will be explained in further detail. The detailed architecture is shown in Figure 2.
The **data corpus** consisted of 48 PDF documents of which 23 documents (47.92 %) consist of two to six pages and 25 documents (52.08 %) consist of 20 to 44 pages. The documents included unstructured text, images, and tables. The use of PDF documents was justified by two main reasons: 1. Most of the provided data in the project consists of PDF documents and 2. the PDFLoader of Langchain provides the most meta-data during text splitting (such as page number etc.). In addition, PDF files are very common in all domains and other text-based document types can usually be converted into PDF files without any information loss.
Data extraction is done via Langchain [14], an open-source tool designed for extracting and structuring text data from

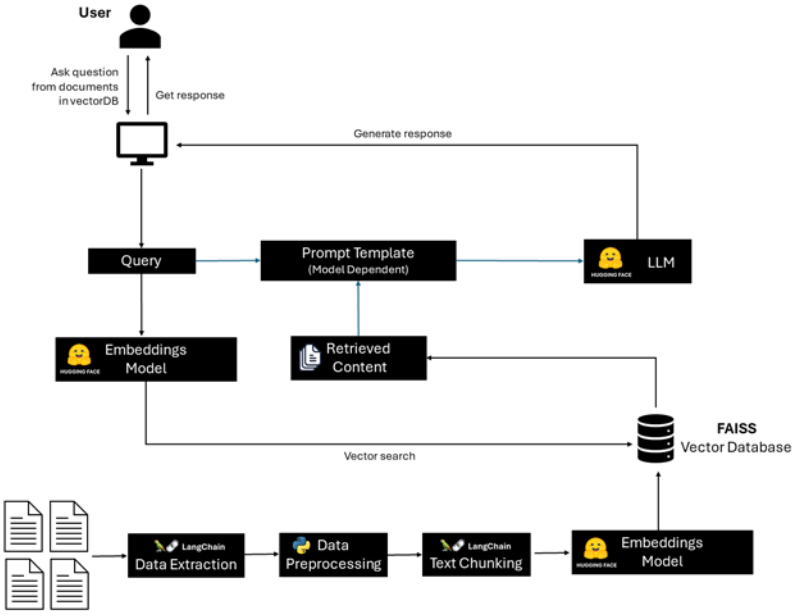


Figure 2: Detailed architecture of the QA-system developed in this study.

documents. Langchain is highly versatile and can be used on a wide range of document types, including PDF files, Word files, HTML etc. For this study, we stick with the PDFLoader of Langchain which is specialized in loading the embedded text from PDF files with additional metadata such as document name, page number etc.

Data Preprocessing is conducted via different natural language processing methods. First, we normalized the extracted text using Normalization Form Compatibility Composition (NFKC) to avoid any issues with Unicode. The Natural Language Toolkit (NLTK) was used to tokenize the text into sentences for further preprocessing. Various augmentations are then applied to the sentences, including the removal of redundant characters, the correction of split words across newlines etc., mostly making use of regular expressions. This method results in a clean and formatted raw text, which is essential for correct embedding.

Text chunking is the process of splitting the text into further smaller units, or “chunks”, to enhance efficiency in storage and retrieval in and from a vector database. This process is crucial for handling larger documents, which, if handed over in their entirety to an LLM, would exceed the maximum token input limit of LLMs. For example, the current State-of-the-Art maximum number of input tokens is at 8192 tokens for ChatGPT (GPT 4.0), which translates to roughly 6150 words. In theory, the concept of chunking will improve the retrieval process later by making it retrieve more relevant passages from a document to a certain query and making the retrieval process more precise by pinpointing to smaller chunks of information and more efficient by lowering the computational cost and by parallelization of the retrieval. Langchain was used for text chunking and the text was split by a chunk size of 500 with an overlap of 100.

Embedding text is used to capture the semantic meaning of a sentence or text in general. This is achieved by converting each chunk into a vector, using an embeddings model. In this study, the sentence-transformers/distiluse-base-multilingual-cased-v1 [15] embeddings model from Hugging Face [16] is used to convert each chunk into its vector representation which takes the semantics of sentences into account. This embedding model was chosen, because of its ability to create valid embedding vectors for German texts. These vectors are crucial as they will be used during the retrieval process. During the similarity search on the vector database, the calculated distance of the two vectors in the vector space determines the proximity of a query and a chunk in real space, i.e., how the degree of relation of the two texts. The embeddings are then indexed with their metadata in a Facebook AI Similarity Search (FAISS) [17] vector database. This indexing is structured to allow for efficient searching, using algorithms and data structures optimized for high-dimensional vector spaces (e.g., KD-trees, Ball-trees, or approximate nearest neighbor (ANN) search techniques).

Retrieval is the process during inference in which the RAG system hands relevant context to the LLM. This allows the knowledge to be accessed as an external information source without the need to train it in the LLM parameters. Here, the query of the user is transformed with the same embedding model as before into a vector. Then, a similarity search is carried out, where the distance between the query vector and all other vectors in the vector database is calculated. The vector in the vector database with the lowest distance to the query vector is then extracted, transformed back to text form, and written inside the query prompt.

LLM answer generation is carried out after the original query prompt is altered with the context provided by the similarity search. In our case, we handed over the three chunks of context with the highest similarity, i.e., the lowest distance to the query. Pre-testing revealed that the number of retrieved chunks has a high impact on the performance of the RAG pipeline and three chunks has shown to be a sweet spot for the number of documents which was used in this study. For

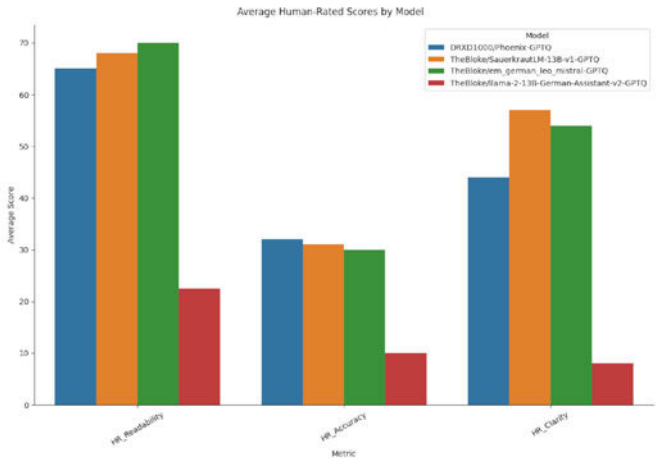


Figure 3: Comparison of the human-rated (HR) metrics of the four tested open-source German LLMs.

Figure 4: Comparison between answers of the Phoenix model (top "Antwort") and the Llama 2 model (bottom "Antwort") to the same question.

example, if too less chunks are provided, the model may miss important information. On the other hand, if too many chunks are provided, the model may get confused and starts to merge non-relevant information.

We tested four different open-source LLM from Hugging Face, which were already pre-trained on German language, namely TheBloke/llama-2-13B-German-Assistant-v2-GPTQ (Llama 2) [18], DRXD1000/Phoenix-GPTQ (Phoenix) [19], TheBloke/em_german_leo_mistral-GPTQ (Leo) [20] and TheBloke/SauerkrautLM-13B-v1-GPTQ (Sauerkraut) [21].

3.2 Evaluation

The performance of the four different LLMs was assessed through human-evaluated metrics — readability, clarity, and accuracy — in relation to the context provided by the text chunks. Here, we manually prepared 120 questions and (reference) answer pairs. The questions were prompted into the pipeline and then evaluated, taking the retrieved context and the reference answer into account. Additionally, two numerical metrics were employed: the Flesch Reading Ease Score (Flesch Score) [22] for readability assessment, and the BERTScore [23] for semantic similarity between the LLM output and reference text. The scores are designed to evaluate the following properties of texts:

- **Accuracy:** How correct is the answer according to the given context? (0-100)
- **Readability:** How easy is the text to read, i.e. how complex are the sentences? (0-100)
- **Clarity:** Which portion of the sentences in the answer refer directly to the question? (0-100)
- **FleschScore:** The average sentence length (the number of words divided by the number of sentences) and the average number of syllables per word (the number of syllables divided by the number of words). (0-100)
- **BERTScore:** Leverages the contextual embeddings from models like BERT (Bidirectional Encoder Representations from Transformers) to perform a nuanced and semantic evaluation. (0.0-1.0)

This methodology and metrics provide a comprehensive framework for evaluating the applicability of LLMs in a multilingual retrieval setting, with a particular focus on enhancing accessibility and accuracy of information retrieval in the German language.

3.3 Results

As mentioned, each model was rated based on accuracy, readability, and clarity with scores ranging from 0 to 100, where a higher score indicated a better performance in the respective category.

Figure 3 shows the results of the human-rated evaluation of the four different LLMs. In general, the Sauerkraut and the Leo model performed similar across all three scores, indicating a small difference between their performance. During evaluation, it was observed that the Sauerkraut model answered more direct and using less words, making it a suited model for tasks were precise and short answers are desired. DRXD1000/Phoenix-GPTQ achieved the highest accuracy score, indicating that it provided the most precise information in relation to the questions asked based on the retrieved context. However, the difference in accuracy score between this model and Sauerkraut and Leo was minimal.

According to the evaluation, the Llama 2 model performed worse on all three scores. This was due to unexpected behavior of the model during question answering, which resulted in significant challenges in avoiding repetition and delivering sensible information.

Overall, Phoenix, Sauerkraut and Leo performed good by giving mostly clear and simple answers. For example, in Figure 4 is shown one answer each of the Phoenix model and of the Llama 2 model, respectively. The answers were generated on the same question, having access to the same context information from the retrieval system. Despite having the same information, the answer of the Llama 2 model lacks important detail in answering the question. The Phoenix model, on the other hand, provides an answer which agrees with the provided context. Thus, the model hands the user all necessary information for answering the question accurately.

Figure 5: (Top) Bad retrieval case. The similarity search performed poorly on the given query. (Bottom) Good retrieval case. The similarity search made a good matching of query and context.

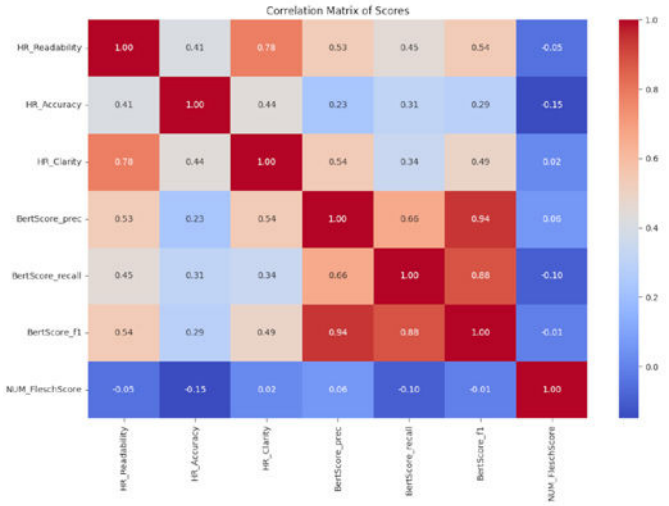


Figure 6: Correlation matrix of all used scores, including precision, recall and f1 of the BERTScore.

However, the models' accuracy was only moderate according to the given context. The lower accuracy is most likely due to the retrieval system, as it occasionally returned smaller text chunks with limited context. Two different extreme cases of a retrieval output during inference with the RAG system are shown in Figure 5. The first question, which is the same as in the examples of Figure 4, leads to a list of relevant chunks. With these chunks, the model has a chance to answer the query correctly. However, in the second case, the retrieval did not provide any meaningful context chunks, making it impossible for the model to answer the query. Thus, the whole system is not able to provide any useful answers when giving irrelevant context information, rendering the similarity search as a bottle neck in the given architecture. Hence, the accuracy of the answers directly correlates with the accuracy of the chunk retrieval. A robust retrieval system is essential for a robust question answering pipeline. Similar observations were addressed by Bornea et al. [13].
Hence, the performance of the retrieval from the vector database via similarity search is essential for the overall performance of a RAG pipeline.
To improve the retrieval, three potential modifications are proposed:

1. Using a different embeddings model, which is fine-tuned solely on German. The currently used embeddings model is a multi-language model and might perform worse than other embedding models which are only trained on the German language. This could lead to a better mapping of the query vectors with the relevant content in the vector database, which ultimately would lead to better context retrieval for the LLM.

2. Using a different vector database with a different similarity search algorithm. In this study, FAISS was used as a vector database, but other databases, such as Chroma or Pinecone, might work better in terms of performing similarity searches. They differ in storing and querying speed and can also provide different similarity search techniques.

3. Using a different chunking technique before storing the text into the vector database. This might be the most promising next step in improving the pipeline. Retrieved chunks were sometimes without any clear context, rendering them useless, when providing them to the LLM. Here, semantic chunking or chunking-by-title from Unstructured.io promises better results already. The investigation of the impact of the chunking techniques will be the next step in this project.

As shown in Figure 6, the evaluation demonstrates a moderate correlation between the BERTScore and the human-rated readability and clarity, with a correlation coefficient ranging from 0.49 to 0.54. The moderate correlation coefficient indicates that the BERTScore is a decent alternative for the evaluation of the LLMs when data, time and workforce for the evaluation is limited. In contrast, the Flesch score showed no correlation at all. Nevertheless, it can still be a valuable insight when the complexity and sentence structure of an LLM output are of particular importance.

4. Conclusion & Future Work

In recent months, LLMs and systems like RAG have been undergoing constant evolution and improvement. They have become more domain-specific and more versatile in terms of development. Also, the libraries of open-source pre-trained models are expanding daily, with more models tailored to specific language and tasks. Thus, building a RAG system based on a pre-trained model can be an attractive and cheap alternative for companies, which want to develop ChatGPT-like assistants despite limited computational power and limited training data.
Even relatively small German models (up to 13 billion parameters) already provide decent capabilities in text generation,

making it possible to chat with them without noticing larger inconsistencies or hallucinations in their answers. However, as shown in this study, correct retrieval in a RAG system can still be challenging in real-world scenarios, especially with the expansion of the underlying data basis, e.g. the number of documents.
In a RAG system, there are numerous parameters that can be altered to enhance the performance of the pipeline. A lot of studies concentrate on fine-tuning or the selection of the optimal LLM, which, in the case of fine-tuning, can be exceedingly costly. The choice of the embedding model and the retrieval algorithm and vector database is at least as crucial and changing or improving them provides a cost-effective alternative to fine-tuning. Also, the document pre-processing, such as text transformation and chunking, has a tremendous impact on the overall performance of a RAG pipeline and should not be neglected.
In future works, we will examine these various properties in depth, with a particular focus on the retrieval system. Our findings indicate that the retrieval system may be a current bottleneck, given that the LLMs themselves perform well in text generation.

Acknowledgements

The study was part of the research project CoLab4DigiTwin and funded by the German Federal Ministry of Economic Affairs and Climate Action (Bundesministerium für Wirtschaft und Klimaschutz, BMWK, grant number # 13IK013F).

References

[1] Chui, M., Manyika, J., Bughin, J., Dobbs, R., Roxburgh, C., Sarrazin, H., Sands, G., Westergren, M. (2012*). The social economy: Unlocking value and productivity through social technologies*. Retrieved from: https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/the-social-economy, accessed: 20. October 2023

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2023). Attention is all you need. arXiv 2017. *arXiv preprint arXiv:1706.03762, 3762.* Retrieved from: http://arxiv.org/abs/1706.03762

[3] Mangaonkar, M., Penikalapati, V. K. (2024). Enhancing Production Data Pipeline Monitoring and Reliability through Large Language Models (LLMs). *Eduzone: International Peer Reviewed/Refereed Multidisciplinary Journal, 13*(1), 51-56.

[4] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems, 33*, 9459-9474., Retrieved from: http://arxiv.org/abs/2005.11401, accessed: 19. April 2024.

[5] IDC. (2018). *Equalizing Time Spent on Data Management vs. Analytics.* Retrieved from: https://blogs.idc.com/2018/08/23/time-crunch-equalizing-time-spent-on-data-management-vs-analytics/, accessed: 15. April 2024.

[6] Al Naqbi, H., Bahroun, Z., Ahmed, V. (2024). Enhancing Work Productivity through Generative Artificial Intelligence: A Comprehensive Literature Review. *Sustainability, 16*(3), 1166. doi: 10.3390/su16031166.

[7] Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., ... Wen, J. R. (2023). A survey of large language models. arXiv preprint arXiv:2303.18223. Retrieved from: https://arxiv.org/abs/2303.18223, accessed 17. May 2024.

[8] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. doi: 10.48550/arXiv.2312.10997.

[9] Chizhikova, A., Murzakhmetov, S., Serikov, O., Shavrina, T., Burtsev, M. (2022). Attention understands semantic relations. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 4040-4050). Retrieved from: https://aclanthology.org/2022.lrec-1.430, accessed: 17. May 2024.

[10] Open AI. (2022). Introducing ChatGPT. Retrieved from: https://openai.com/blog/chatgpt, accessed: 19. April 2024.

[11] Kernan Freire, S., Wang, C., Foosherian, M., Wellsandt, S., Ruiz-Arenas, S., Niforatos, E. (2024). Knowledge sharing in manufacturing using LLM-powered tools: user study and model benchmarking. *Frontiers in Artificial Intelligence, 7*, 1293084. doi: 10.3389/frai.2024.1293084.

[12] Makatura, L., Foshey, M., Wang, B., Hähnlein, F., Ma, P., Deng, B., ... Matusik, W. (2024). Large Language Models for Design and Manufacturing. doi: 10.21428/e4baedd9.745b62fa.

[13] Bornea, A. L., Ayed, F., De Domenico, A., Piovesan, N., Maatouk, A. (2024). Telco-RAG: Navigating the Challenges of Retrieval-Augmented Language Models for Telecommunications. *arXiv preprint arXiv:2404.15939*. doi: 10.48550/arXiv.2404.15939.

[14] Chase, H. (2022). *LangChain*. Retrieved from: https://github.com/langchain-ai/langchain

[15] Sentence Transformers. (2021). *Sentence-transformers/distiluse-base-multilingual-cased-v1*. Retrieved from: https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1, accessed 19. April 2024.

[16] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... Rush, A. M. (2020). HuggingFace's Transformers: State-Of-The-Art Natural Language Processing. ArXiv191003771 Cs. doi: 10.48550/arXiv.1910.03771.

[17] Johnson, J., Douze, M., Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data, 7*(3), 535-547. Retrieved from: http://arxiv.org/abs/1702.08734, accessed: 19. April 2024.

[18] Jobbins, T. (2023). *TheBloke/llama-2-13B-German-Assistant-v2-GPTQ*. Retrieved from: https://huggingface.co/TheBloke/llama-2-13B-German-Assistant-v2-GPTQ, accessed 19. April 2024.

[19] Uhlig, M. (2024). *DRXD1000/Phoenix-GPTQ*. Retrieved from: https://huggingface.co/DRXD1000/Phoenix-GPTQ, accessed 19. April 2024.

[20] Jobbins, T. (2023). *TheBloke/em_german_leo_mistral-GPTQ*. Retrieved from: : https://huggingface.co/TheBloke/em_german_leo_mistral-GPTQ, accessed 19. April 2024.

[21] Jobbins, T. (2023). *TheBloke/SauerkrautLM-13B-v1-GPTQ*. Retrieved from: https://huggingface.co/TheBloke/SauerkrautLM-13B-v1-GPTQ, accessed 19. April 2024.

[22] Flesch, R. (1948). A new readability yardstick. *Journal of Applied Psychology, 32*(3), 221–233. doi: 10.1037/h0057532.

[23] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*. doi: 10.48550/arXiv.1904.09675.

AUTHORS

Dr. Robert Becker is a data scientist at the August-Wilhelm Scheer Institute. He is specialised in the fields of data processing/analysis, neural networks, and conventional machine learning methods. He has experience in innovative cross-disciplinary projects in field of basic research as well as in development of practical applications.

Dr. Robert Becker
August-Wilhelm Scheer Institut gGmbH
Campus D5 1
66123 Saarbrücken
+49 681 93511-279
@ robert.becker@aws-institut.de

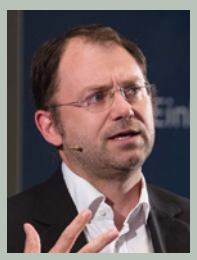
Laura Steffny is a researcher at the August-Wilhelm Scheer Institute. Her expertise lies in data preparation and analysis and AI development. She has extensive experience in implementing sensor-based and workflow-optimizing algorithms.

Laura Steffny
August-Wilhelm Scheer Institut gGmbH
Campus D5 1
66123 Saarbrücken
Telefon: +49 681 93511-121
@ laura.steffny@aws-institut.de

Dr.-Ing. Thomas Bleistein is a team lead and a scientist at the August-Wilhelm Scheer Institute, specializing in the field of Smart Energy. His expertise lies in data analytics, optimization, and mechanical engineering. He has extensive experience in conducting multiple projects on planning and operational optimization of energy systems for commercial and industrial sites, as well as the use of digital twins in the context of manufacturing systems.

Dr.-Ing. Thomas Bleistein
August-Wilhelm Scheer Institut gGmbH
Campus D5 1
66123 Saarbrücken
Telefon: +49 681 93511-127
@ thomas.bleistein@aws-institut.de

Dr. Dirk Werth is Managing Director and Scientific Director of the August-Wilhelm Scheer Institute for Digital Products and Processes, a private digitalization institute that transforms interdisciplinary and cross-industry digital innovations into marketable products.



Dr. Dirk Werth
August-Wilhelm Scheer Institut gGmbH
Scientific Director
Campus D5 1
66123 Saarbrücken
@ dirk.werth@aws-institut.de