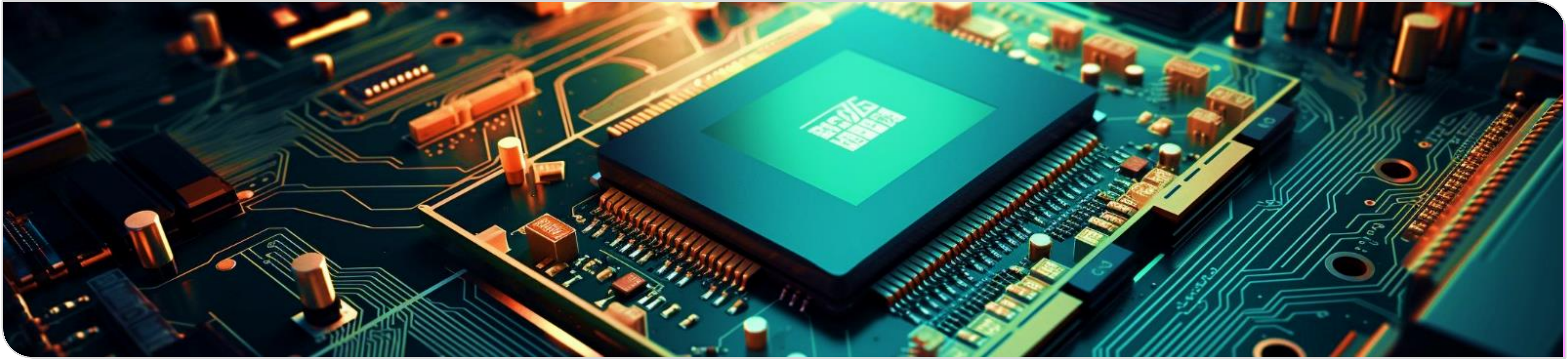
Karlsruhe Institute of Technology

Master Thesis – Overview:

Leveraging Embeddings and Knowledge Graphs for Enhanced Scholarly Information Retrieval: A Comparative Analysis of Retrieval Approaches Using Large Language Models

by Marco Schneider supervised by Angelika Kaplan

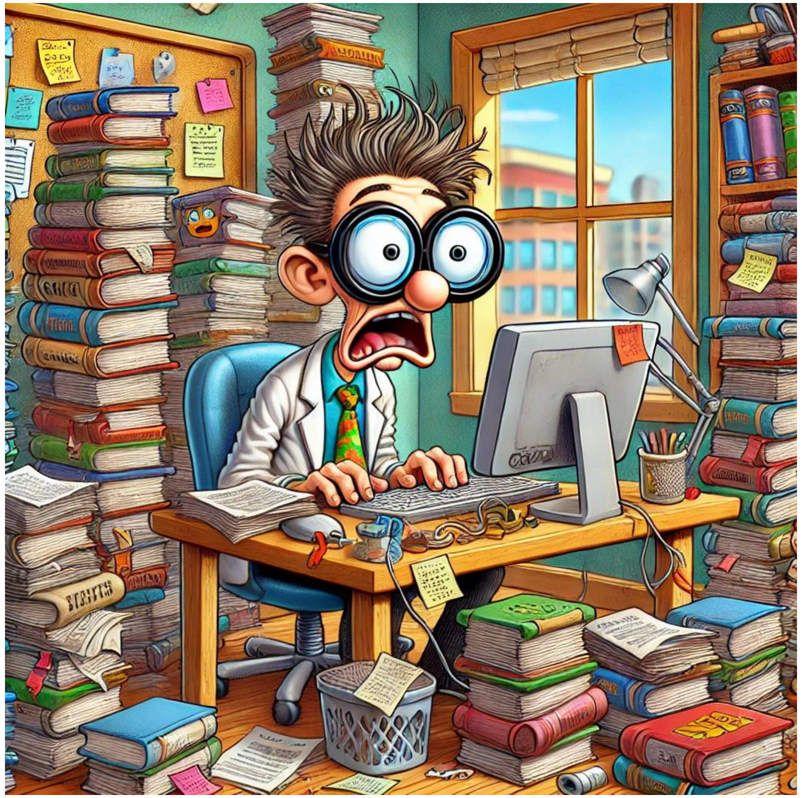


Overview for the Architecture Review

- Thank you for participating in the Architecture Review
- The following pages introduce you to the topic of the master thesis with the intention to prepare the background and fundamentals of the system we intend to create

Literature Research is hard

Who doesn't know this situation?



OpenAI. (2024). Cartoon-style image of a frustrated academic researcher. Created with DALL-E. Retrieved on July 4, 2024

In modern academic settings, most research results are published in scholarly **digital articles**, presenting difficulties for human and automated processing [Jaradeh19].

Academic search engines return a **set of ranked documents** and users have the tedious task of finding relevant information from it [Thambiand22].

Future generation academic search engines should **focus on understanding meaning** rather than simply matching keywords [Hippel23].

Literature Research could be easy

Large Language Models (LLMs) demonstrate **remarkable abilities** in natural language tasks [Yangetal24]

The application of LLMs to academic search has the potential to **reduce barriers** to accessing information and **speed up** research tasks

Just research by chatting!

OpenAI. (2024). Cartoon-style image of a happy academic researcher. Created with DALL-E. Retrieved on July 4, 2024

LLMs alone are not sufficient

OpenAI. (2024). Cartoon-style image of a hallucinating ChatGPT. Created with DALL-E. Retrieved on July 4, 2024

LLM is trained and its knowledge is **frozen** in time

Especially in knowledge specific tasks LLMs tend to **hallucinate** where they contradict existing sources or lack supporting evidence [Yangetal24]

Retrieval Augmented Generation (RAG) allows to **integrate external knowledge** to retrieve current knowledge and reduce hallucinations [Lewis20]

Overview for the Architecture Review

Main Goal

Investigate how Large Language Models (LLMs) can be used as retrieval agents on knowledge graphs in a Retrieval Augmented Generation (RAG) system to improve quality and reliability in question-answering for software architecture research.

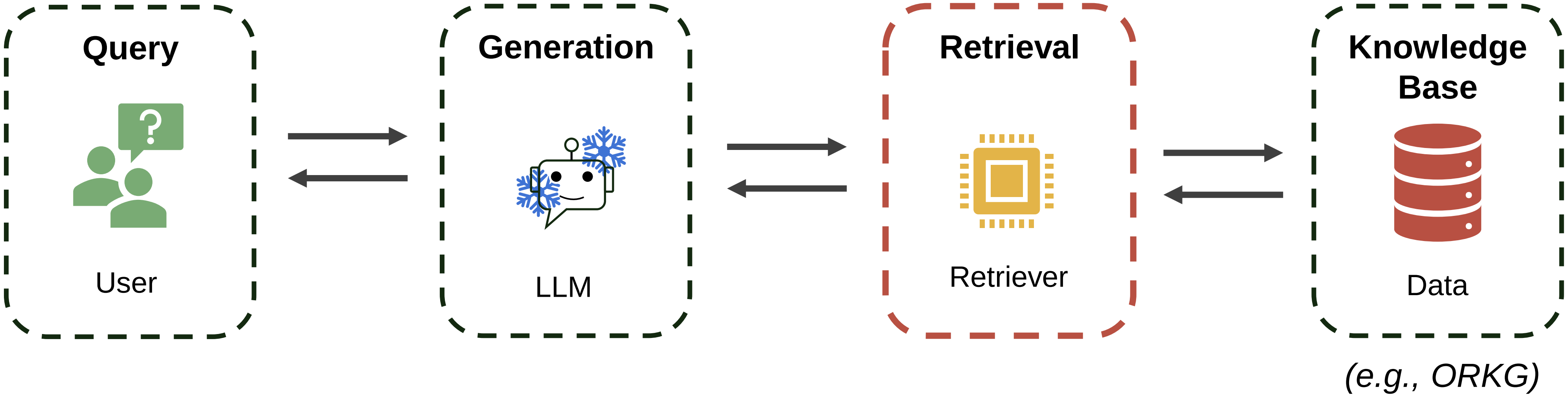
- To be able to compare different “retrieval” approaches (see next page) and allow users to ask questions and receive answers, we need a **flexible implementation of a question-answering system** with a focus on evaluation.
- We intend to implement this system using a process called “Retrieval Augmented Generation (RAG)” while leveraging a popular framework für LLM applications called LangChain.

Retrieval Augmented Generation (RAG)

What is RAG?

- RAG aims to **retrieve relevant data from an external knowledge base** to give the LLM recent and factual data for answer generation

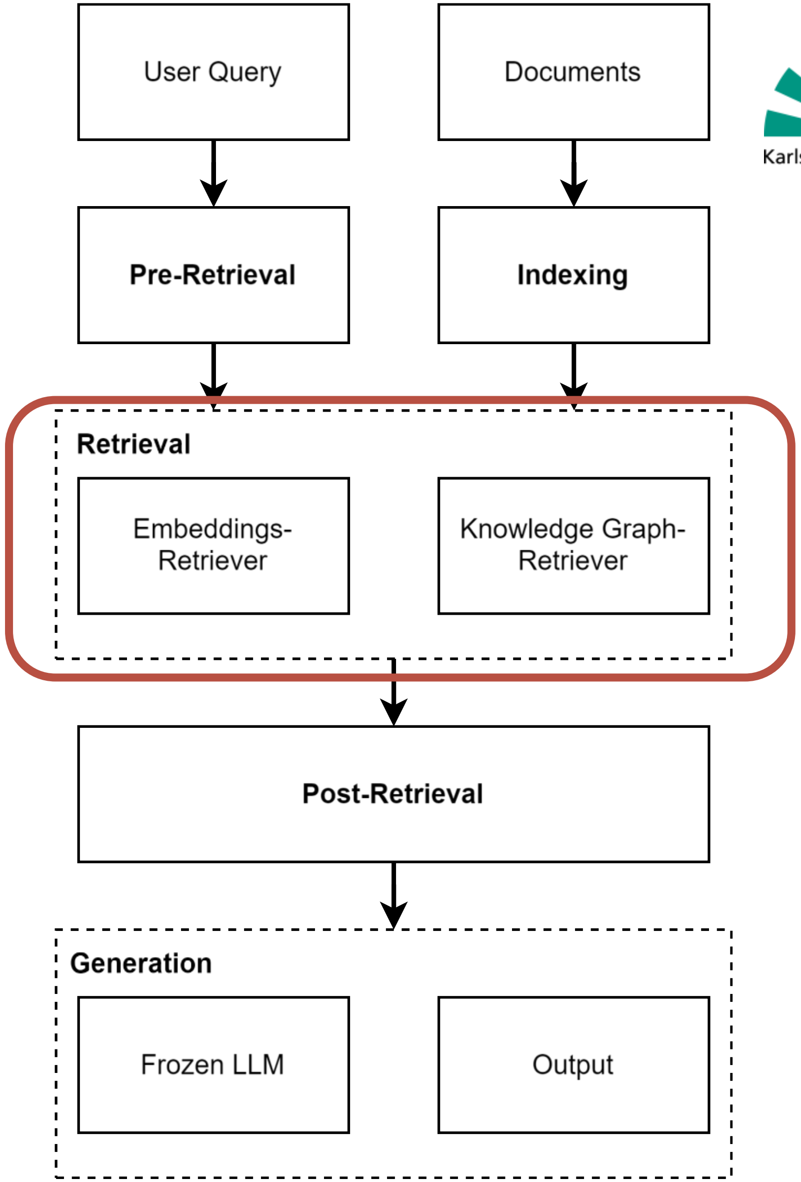
Overview RAG Process



Deeper look

- The RAG approach basically has 4 main phases

1. It starts with the **indexing** where data is saved in an appropriate format in a knowledge base (KB)
2. **Pre-Retrieval** are processing steps applied to a user query (question) to prepare the question for the retriever
3. **Retrieval** here a retriever object receives the query and retrieves the most relevant documents from the KB. This can be either embedding-based or knowledge graph-based depending on the KB used
4. **Generation** here a language model (like GPT) is tasked to generate an answer based on the documents retrieved



What should the system do?

- Ok, the RAG process is basically the high-level concept behind the question answering system that we intend to implement
- The benefits the system should provide are listed below. They are split between the user, developer, and the master thesis. This should indicate what the system will be used for

User Role

1. Ask questions on a scholarly dataset
2. Find relevant literature faster

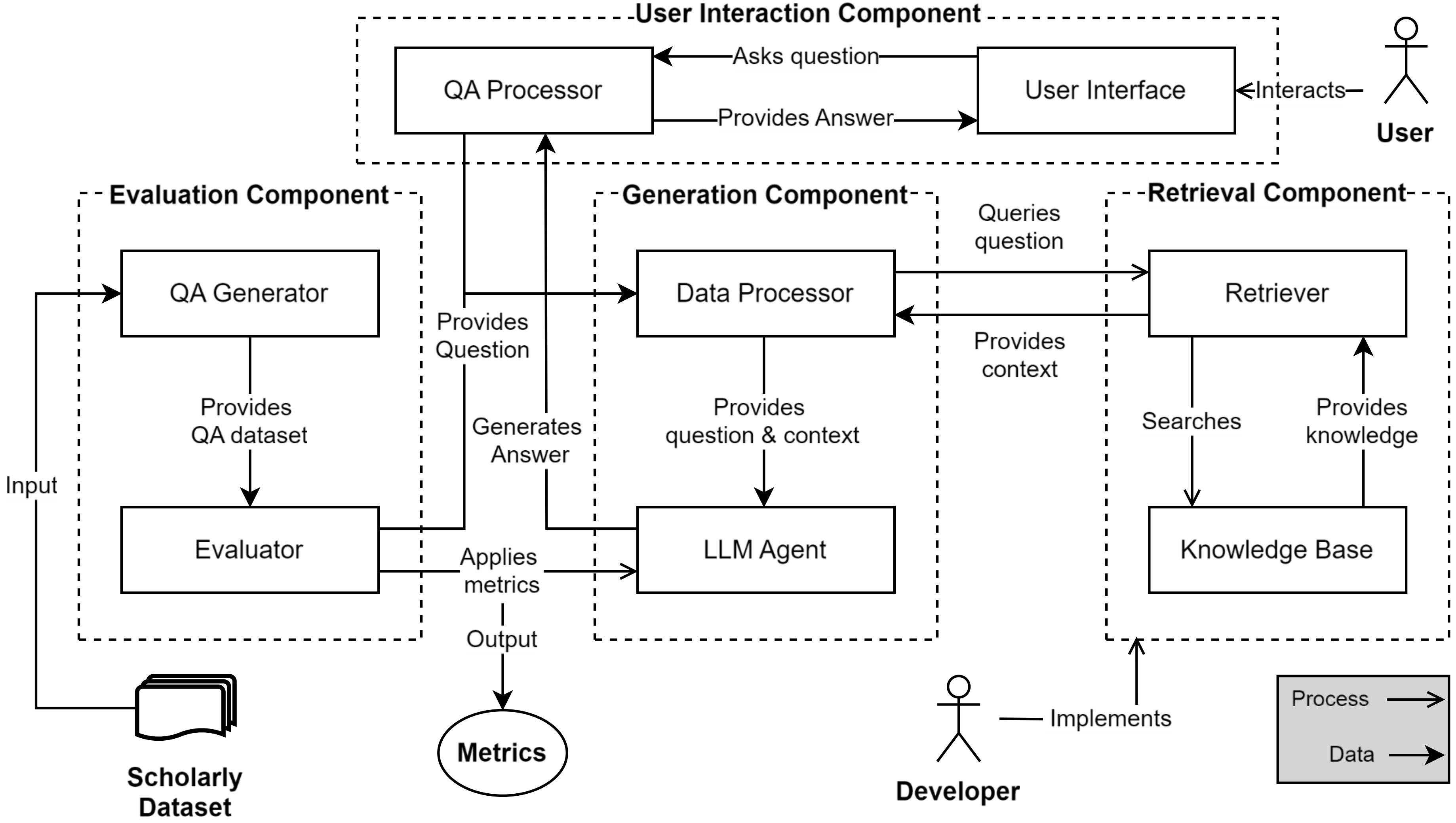
Developer Role

1. Tool for the implementation of a RAG process
2. Benchmarking of retrievers

Master Thesis

Tool for a consistent and empirical evaluation of embedding and KG-based retrieval approaches

Concept of the System



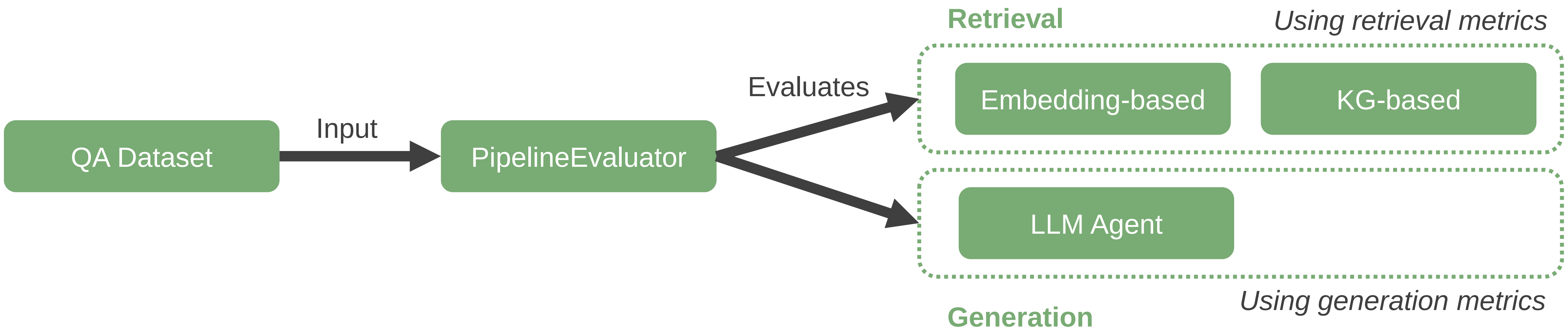
Overview for the Architecture Review

- More resources for the architecture of the question answering system can be found in the cloud folder
- These are work-in-progress! There may be changes up until the review of the architecture.

Evaluation

- The evaluation is done using the **Evaluator component**

- Current metrics used to evaluate a RAG system are applied on the components
  Retrieval (Accuracy, Precision, Recall, …) and
  Generation (Faithfulness, Answer Relevance, Answer Correctness, …)



Problem, Idea, Benefit, Actions (PIBA)

Karlsruhe Institute of Technology

Problem

Literature research for software architecture is a cumbersome process.

Idea

Investigate how LLMs can be leveraged to enhance the quality and reliability of content retrieved from KGs (e.g, ORKG) for software architecture research.

Benefit: User Role

Reduce barriers to accessing information and speed up research tasks.

Benefit: Dev Role

Provide tools and insights to enhance retrieval performance.

Actions

1. Development of a configurable **RAG-process Framework** and a **Question-Answering Dataset.**
2. Implementation and Evaluation of **existing Knowledge Graph-based RAG.**
3. Development of a **new Knowledge Graph-based RAG** retrieval approach.
4. Experimentation on **Improvement Techniques** for retrieval on Knowledge Graphs.