Codeware-RAG: Rethinking Retrieval in RAG Systems for Code

Seminar Programming Experience

Hendrik Droste

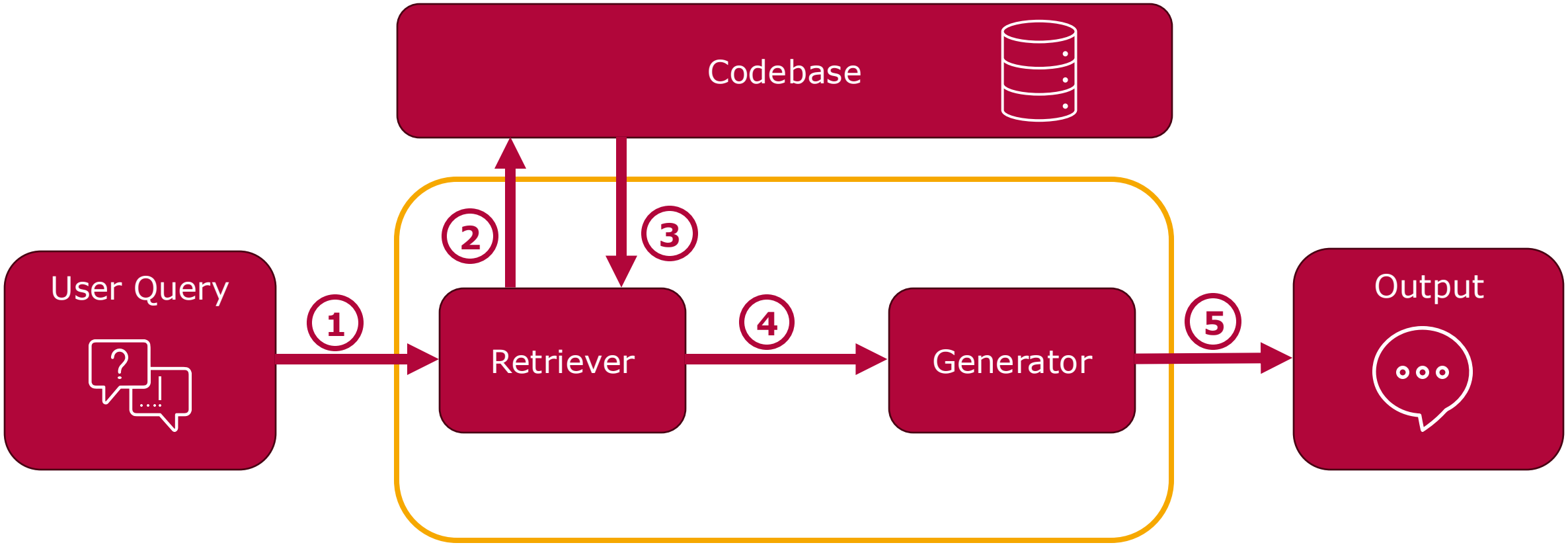
Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Design IT.
Create Knowledge.

Agenda

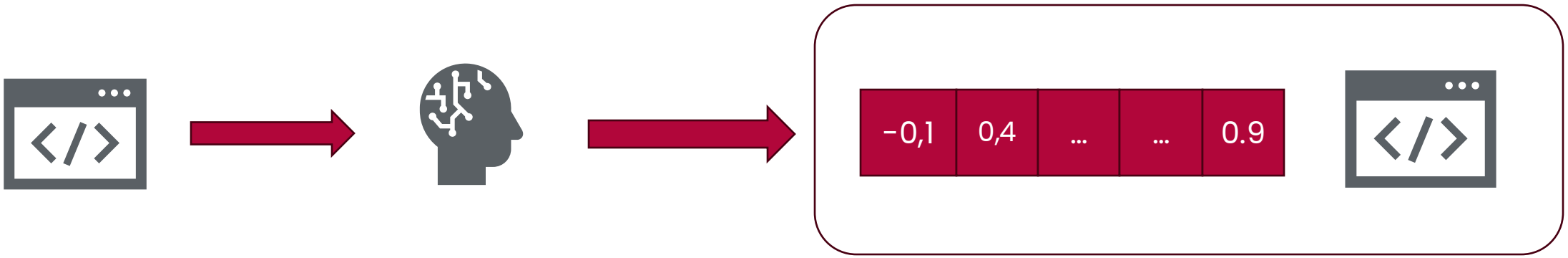
1. Background & Motivation
   - How does RAG Work
   - How to create the Codebase
   - Why Code is Different
2. Conceptional Approach
   - Building the Dataset
   - Prototype RAG Pipeline
   - Mean Reciprocal Rank for Evaluation
   - Experiments
3. Next Steps
4. Summary

How does RAG work?



RAG System

How to create the codebase?



Code Fragment

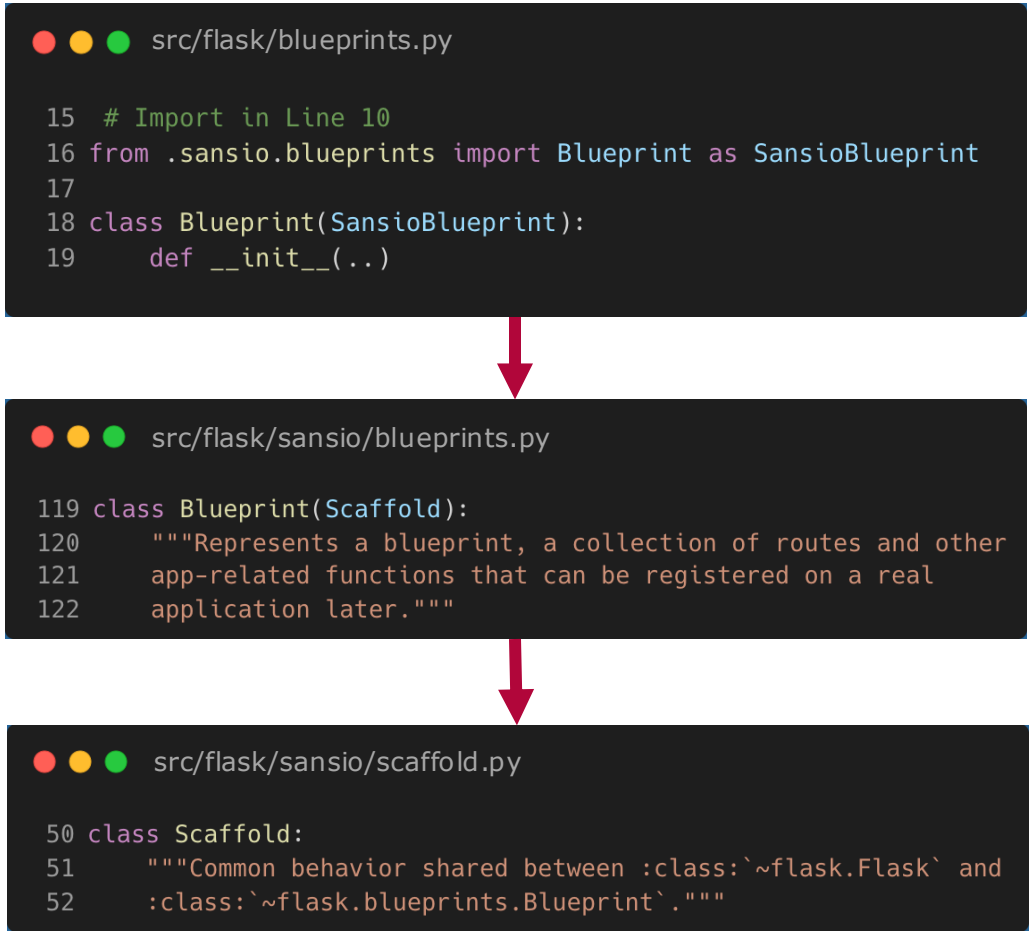
Embedding Model

Record with vector and code

Why Code is Different

- Code semantics span multiple files and modules
- Chunking can break function and class boundaries
- Docstrings are easier to understand for regular LLMs and Humans

Task: Describe the inheritance tree of blueprint!



Components Overview – What do we need?

Codebase

Embedding Model

Splitter

Retriever

Evaluation

Building the Dataset

- Existing Datasets focus on:
  - code generation [1,3]
  - finding similar code [1]
  - mapping of documentation to code [2]

- Our Approach
  - Use Flask: a popular, well-documented codebase
  - Create custom dataset
    - Inspired by questions from Stack Overflow
    - Natural Language (NL) -> Code
    - 18 Questions

Example Questions
Describe the class Scaffold! Where is it used in the project?
How can I send a file?
Where does the app load the default values?
How to divide flask app into multiple files?
How can I redirect a to a URL?

Baseline Embedding Models & Splitter

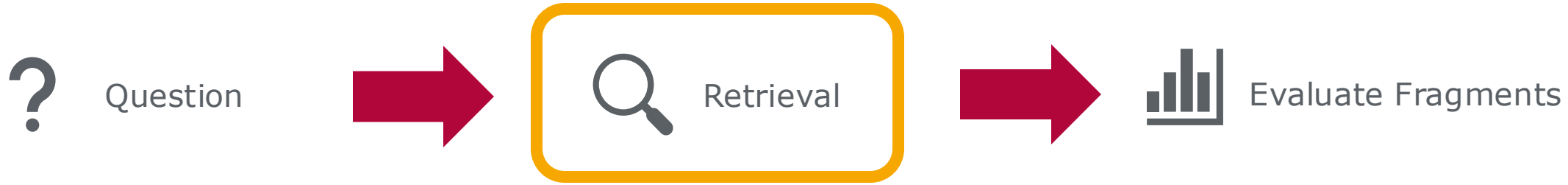
- Chunking with LangChain python text splitter [4]
  - Split in code fragments of maximum 900 characters
  - Splits based on regular expressions (class, func, \n\n)

- Compare Multiple Embedding Models
  - Selection based on Huggingface Massive Text Embedding Benchmark [5]
  - 4 Models trained for Code Embedding
  - 3 Models trained for NL Embedding
  - Configurable via config file

Prototype RAG Pipeline

Codebase generation

Querying



Abstract Class Base-Pipeline with function for splitting and retrieval

Evaluation Metric – Mean Reciprocal Rank (MRR)

- K: rank position of the first relevant code fragment
- $Reciprocal\ Rank\ Score = \frac{1}{K}$
- MRR is the mean RR across multiple queries
- Common evaluation metric for RAG Systems (higher is better)

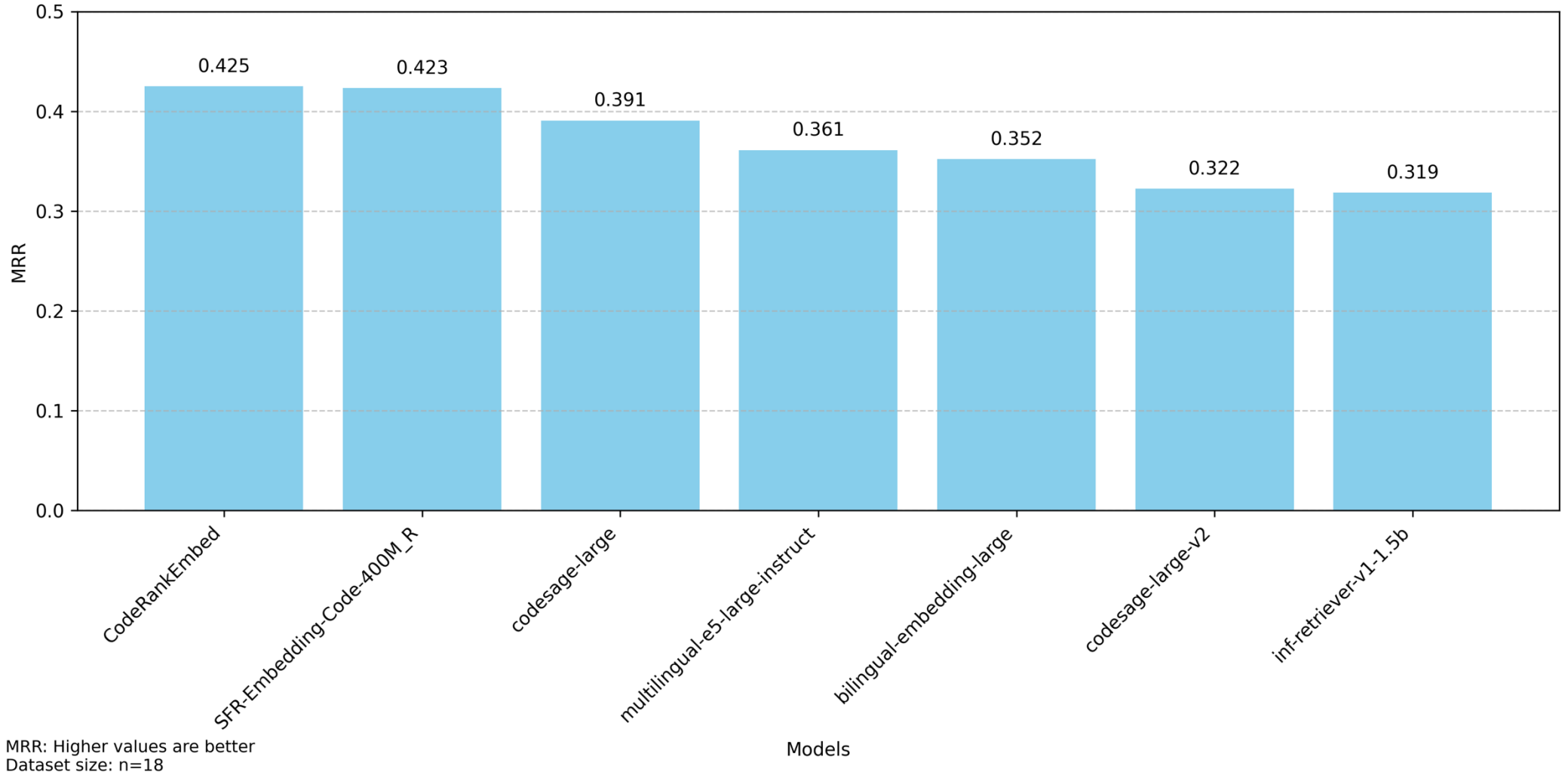
$K = 3$ 

$K = 1$ 

$K = 2$ 

$$MRR = \frac{\left(\frac{1}{3} + \frac{1}{1} + \frac{1}{2}\right)}{3} = \frac{11}{18}$$

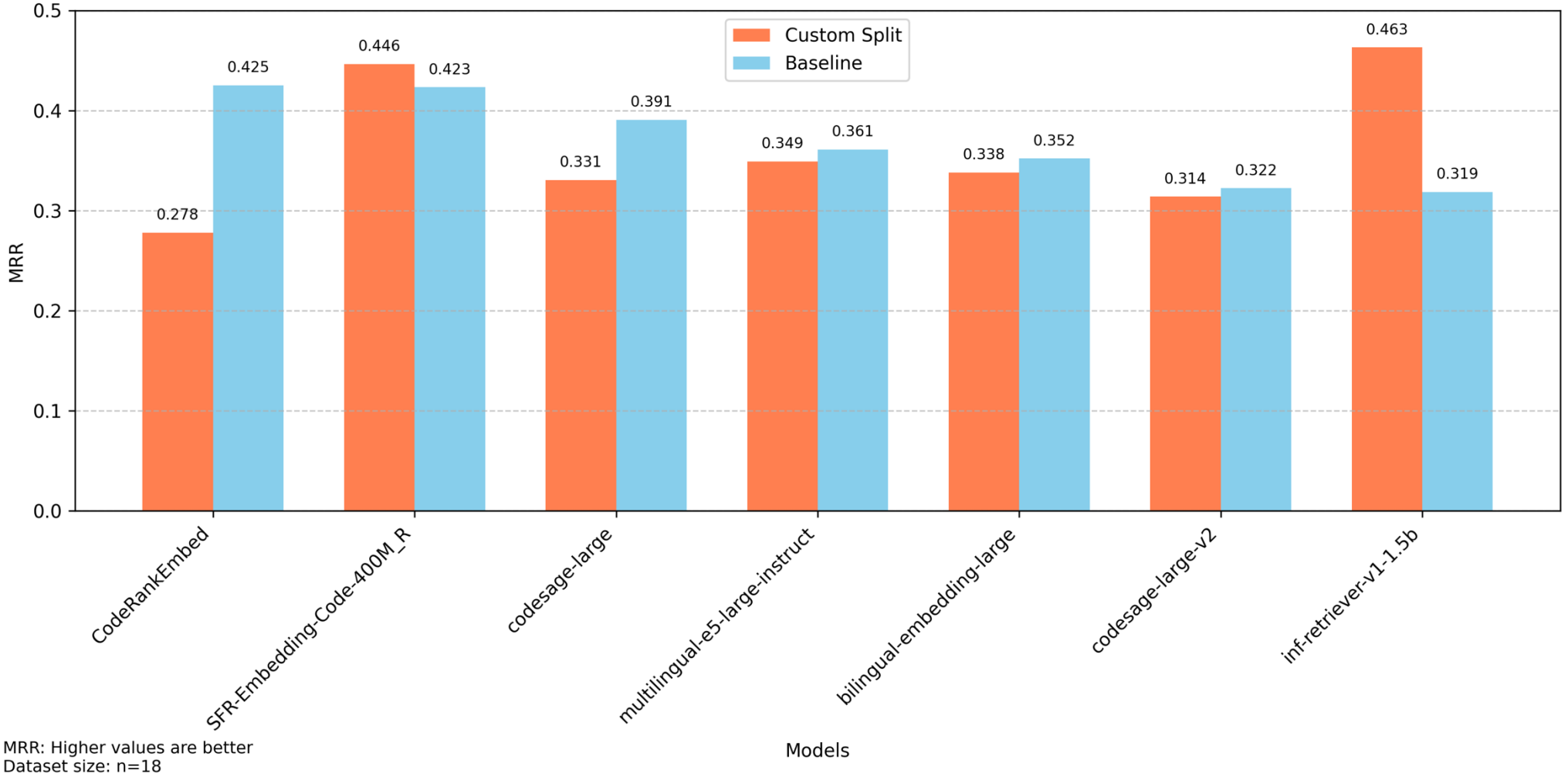
Baseline MRR by Model



MRR: Higher values are better
Dataset size: n=18

Custom Chunking with Tree-Sitter

- Idea: Use Tree-Sitter [6] to create better splits than LangChain

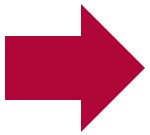


Chunk Summarization with LLM

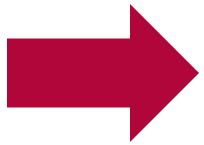
- Embedding Models are finetuned on Code or NL but not on both
- Use LLM to create text summary of code and embed these but return the code!
  - Use Llama-3.2-1B-Instruct [7] for summary
- Calculate similarity between NL and NL and not NL and Code

src/flask/sansio/app.py

```
605 def add_url_rule(..):
606     if endpoint is None:
607         endpoint = _endpoint_from_view_func(view_func)
608     options["endpoint"] = endpoint
609     methods = options.pop("methods", None)
```

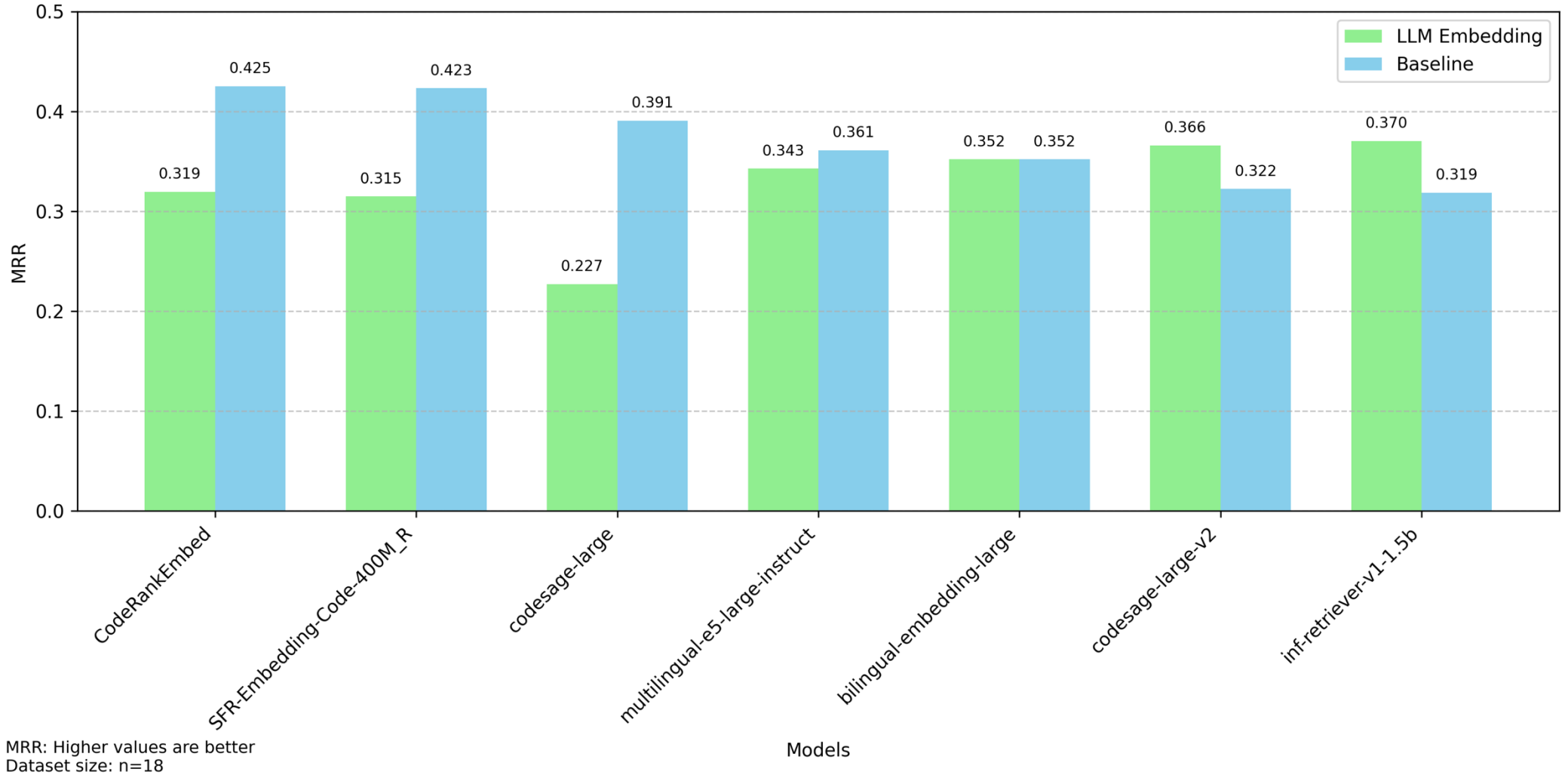


Function Purpose:
The add_url_rule function adds a URL rule to a Flask application.
Key Functionality:
[...]



Embed

Chunk Summarization with LLM



Discussion

Very Small Dataset (use LLM to generate more questions)

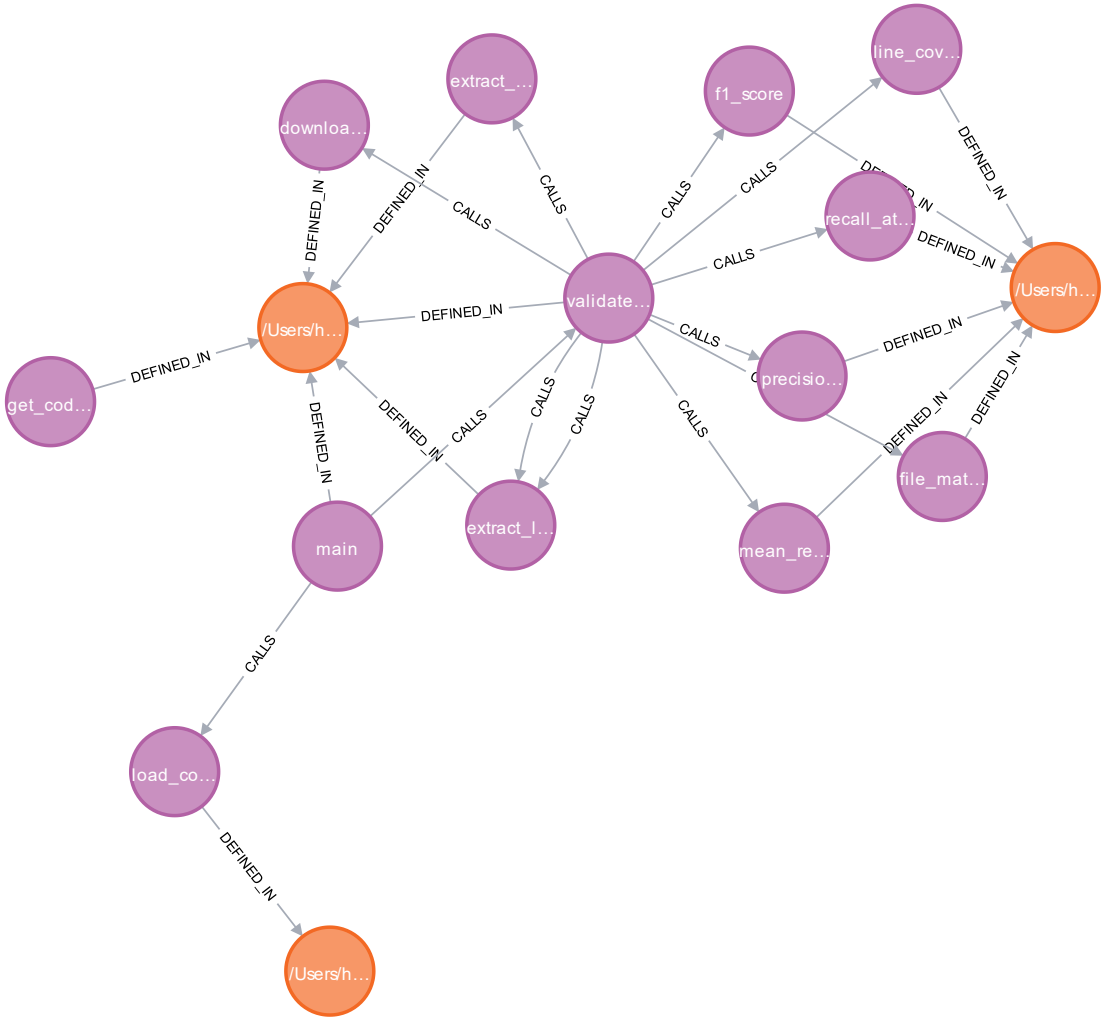
MRR should not be the only metric as some questions require multiple code snippets

LLM chat output is not measured

Influence of chunking on the generator is not measured

Future Work: Reference Graph Retrieval

- Idea: Create Neo4j [8] Graph Database for Code
  - Function Calls
  - Definitions
  - Inheritance Structure
- Retrieval
  - Use embedding to get first anchor fragment
  - Call Graph Database to find other relevant fragments for retrieval



Future Work: Search

- Idea: Implement search
  - Text
  - Definitions of variables
- Retrieval
  - Use LLM to filter out key search terms
  - Find relevant fragments (e.g. TF-IDF)

Task: Describe the inheritance tree of blueprint!

src/flask/blueprints.py

```
15  # Import in Line 10
16 from .sansio.blueprints import Blueprint as SansioBlueprint
17
18 class Blueprint(SansioBlueprint):
19     def __init__(..)
```

src/flask/sansio/blueprints.py

```
119 class Blueprint(Scaffold):
120     """Represents a blueprint, a collection of routes and other
121     app-related functions that can be registered on a real
122     application later."""
```

src/flask/sansio/scaffold.py

```
50 class Scaffold:
51     """Common behavior shared between :class:`~flask.Flask` and
52     :class:`~flask.blueprints.Blueprint`."""
```

Key Takeaways from Codeware-RAG

What I Built

- Custom Flask Dataset
- Modular RAG Pipeline
- Multiple splitter and corresponding retriever
- Evaluation via MRR

What I Learned

- Code need semantic-aware chunking
- Existing code splitting methods provide already strong results
- Summarization adds abstraction but no performance gains

What's Next

- Use graph database with function definitions and usages
- Add search functionality for retriever

Sources

[1] H. Husain, H.-H. Wu, T. Gazit, M. Allamanis, and M. Brockschmidt, 'CodeSearchNet Challenge: Evaluating the State of Semantic Code Search', arXiv [cs.LG]. 2020.

[2] S. Lu et al., 'CodeXGLUE: A Machine Learning Benchmark Dataset for Code Understanding and Generation', arXiv [cs.SE]. 2021.

[3] Y. Li et al., 'Competition-level code generation with AlphaCode', Science, vol. 378, no. 6624, pp. 1092 1097, 2022.

[4] "PythonCodeTextSplitter — 🦜🔗 LangChain documentation." https://python.langchain.com/api_reference/text_splitters/python/langchain_text_splitters.python.PythonCodeTextSplitter.html

[5] "MTEB Leaderboard - a Hugging Face Space by mteb." https://huggingface.co/spaces/mteb/leaderboard

[6] Max Brunsfeld, „tree-sitter/tree-sitter: v0.25.6". Zenodo, Juni 04, 2025. doi: 10.5281/zenodo.15594630.

Sources

[7] A. Grattafiori et al., ‘The Llama 3 Herd of Models’, arXiv [cs.AI]. 2024.

[8] Neo4j, “NEO4J Graph Database & Analytics | Graph Database Management System,” Graph Database & Analytics, Jul. 31, 2024. https://neo4j.com/