Project Group (SS26+WS26/27)

Benchmarking and Optimizing Mobile RAG Systems

Prof. Dr. Lin Wang, Huzaifa Shaaban Kabakibo

Computer Networks Group
Paderborn University

LLMs and RAG

Large language models (LLMs) have enabled many cool applications

ChatGPT

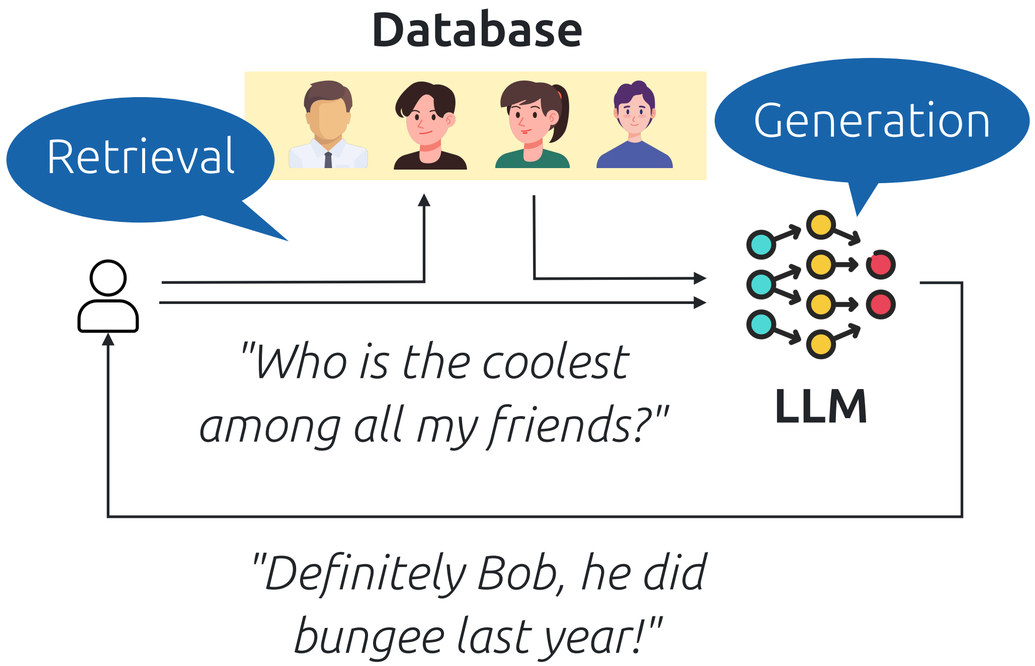
manus

Claude

Gemini

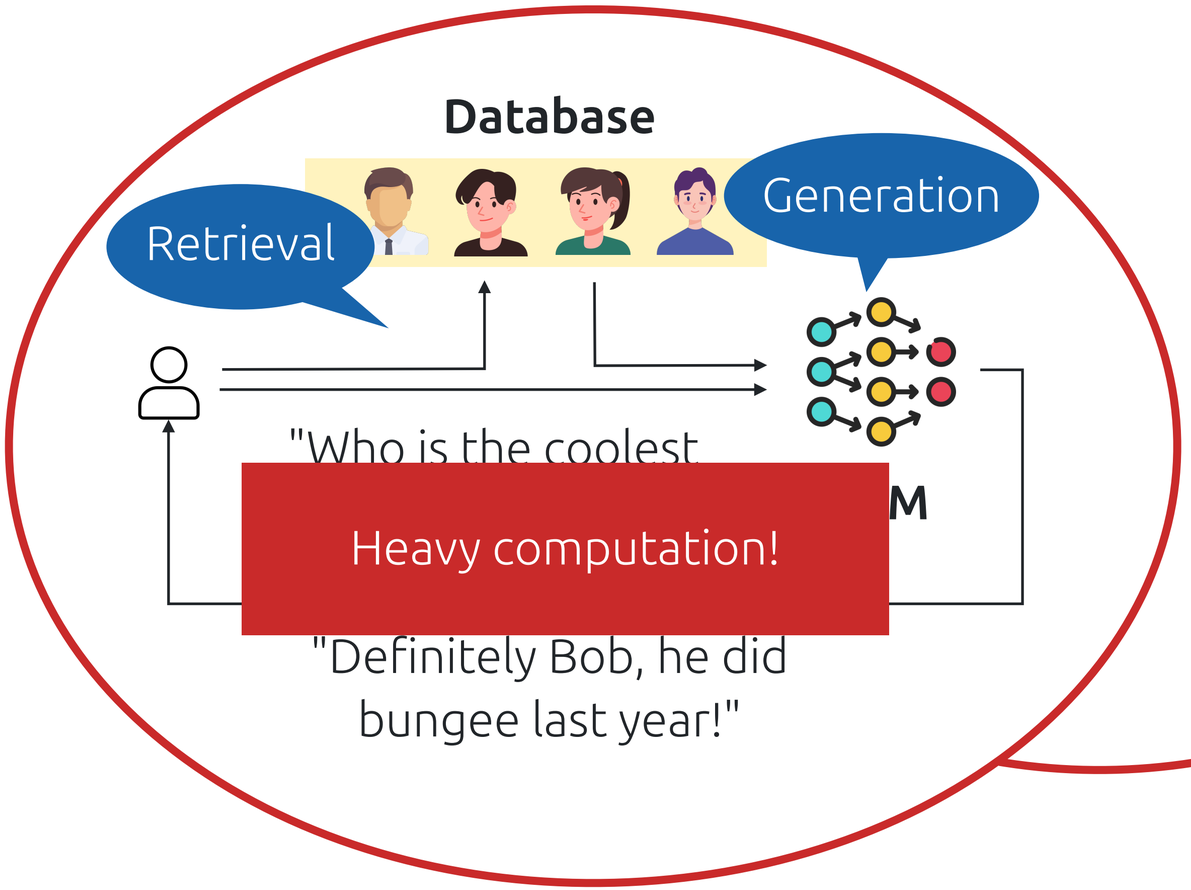
deepseek

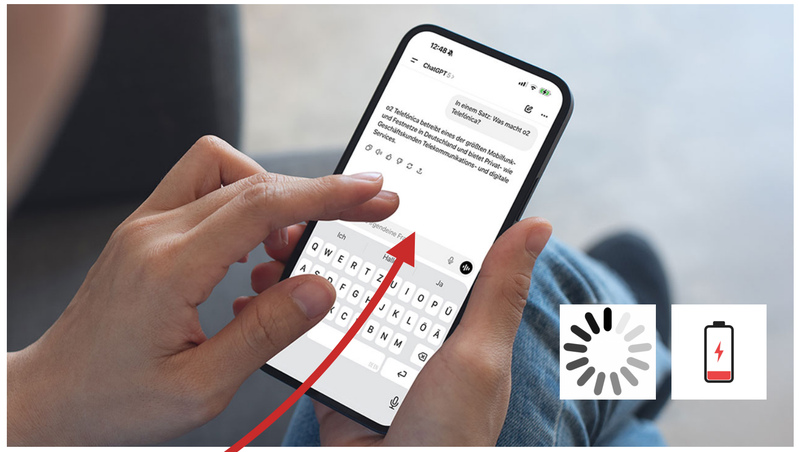
Retrieval augmented generation (RAG) makes LLMs even better!



MobileRAG

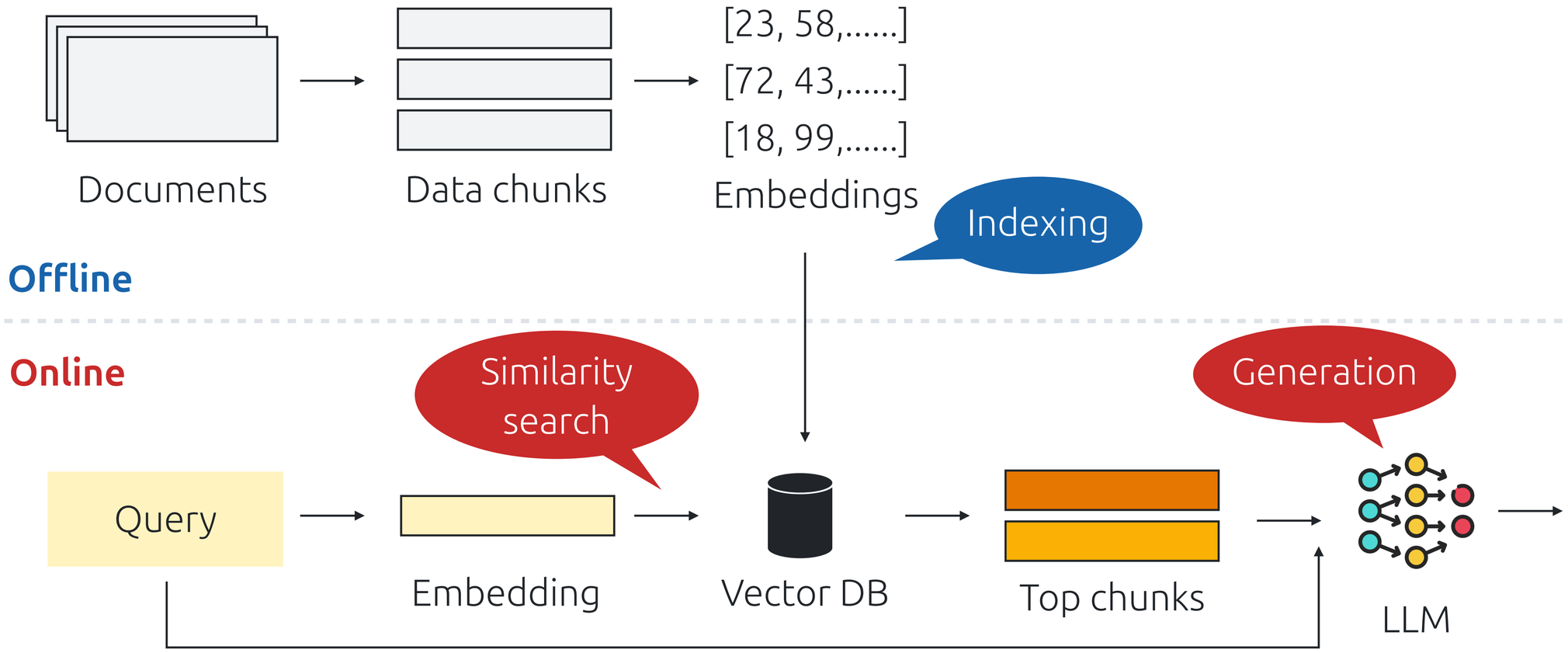
Everything on-device: **better privacy** and **more usable** under poor network



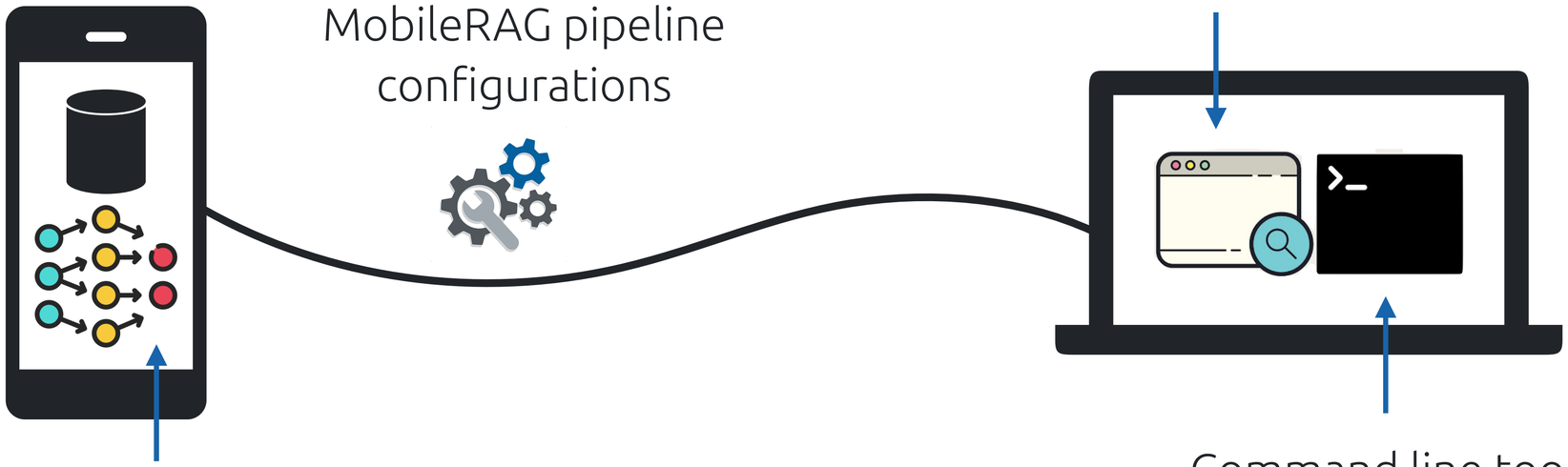


Question: how to make MobileRAG fast and efficient?

MobileRAG pipeline



Benchmarking framework



Current status

Configurable parameters

- Supports multiple datasets
- Supports different LLMs
- Supports 3 different indexing methods
- User-tunable paramters

Automated benchmarking for different datasets

Results generation and reporting for different metrics

- Per-component latency, throughput
- Accuracy
- Memory

```
{
    "downstream_task": {
        "name": "squad",
        "sampling_method": "first_n",
        "limit": 2000
    },
    "rag_pipeline": {
        "embedding": {
            "backend": "cpu",
            "model_name": "all-minilm-l6-v2",
            "dtype": "int8",
            "chunker": {
                "method": "token",
                "size": 256,
                "overlap_enabled": true,
                "overlap_size": 50
            }
        },
        "faiss": {
            "method": "ivf",
            "backend": "cpu",
            "metric": "IP",
            "top_k": 3,
            "config": {
                "nprobe": 8,
                "nlist": 16,
                "num_training_vectors": 150
            },
            "use_cache": false
        },
        "llm": {
            "aug_method": "concatenation",
            "backend": "cpu",
            "model_name": "qwen2.5-0.5B",
            "use_sampling": false,
            "dtype": "int8",
            "kv_window": 4096,
            "prefill_chunk_size": 1024,
            "max_tokens": 96,
            "ignore_eos": false,
            "generate_until": ["\n", "\n\n"],
            "system_prompt": "You are a helpful assistant."
        }
    }
}
```

Ongoing and future directions

Basic

- Add support for more embedding models and LLMs, more datasets, more indexing methods
- Add hyper-parameters for each component
- Add UI to the configuration and result reporting

Intermediate

- Add augmentation methods (e.g., re-ranking, filtering)
- Add support for more metrics (CPU/GPU utilization, power consumption)
- Intensive benchmarking on multiple mobile phones

Advanced

- Code migration to native languages (e.g., C)
- Add support for GPU/NPU acceleration for each component


GOOD COFFEE.
GREAT RESEARCH.
COME JOIN US!