LLMs without the hype

How to leverage LLMs as a software developer

Felix Müller - January 2024

Who am I?

Senior Software Engineer at Stack Overflow

15 years in the industry

Worked in different roles from engineer to architect and consultant

Building LLM-backed applications last months

Moved past the hype

Believes they'll grow complexity overall

What could go wrong, eh?

Introduction

Goals

You can start building LLM-backed applications

LLMs are no magic

Learn the basics of large language models, embeddings and vector stores

Learn about Retrieval Augmented Generation

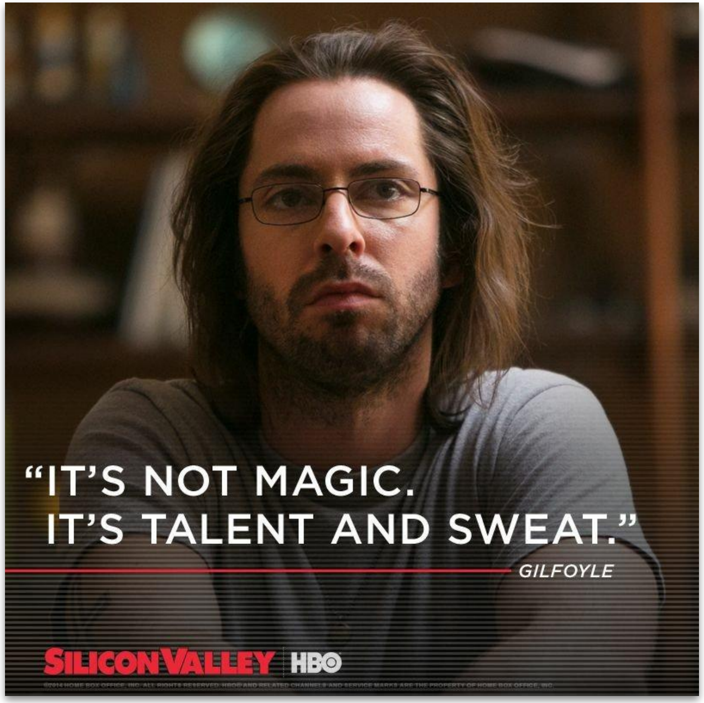
Some thoughts about maintenance

We go on a journey

Let's build an app!

InboxAgent

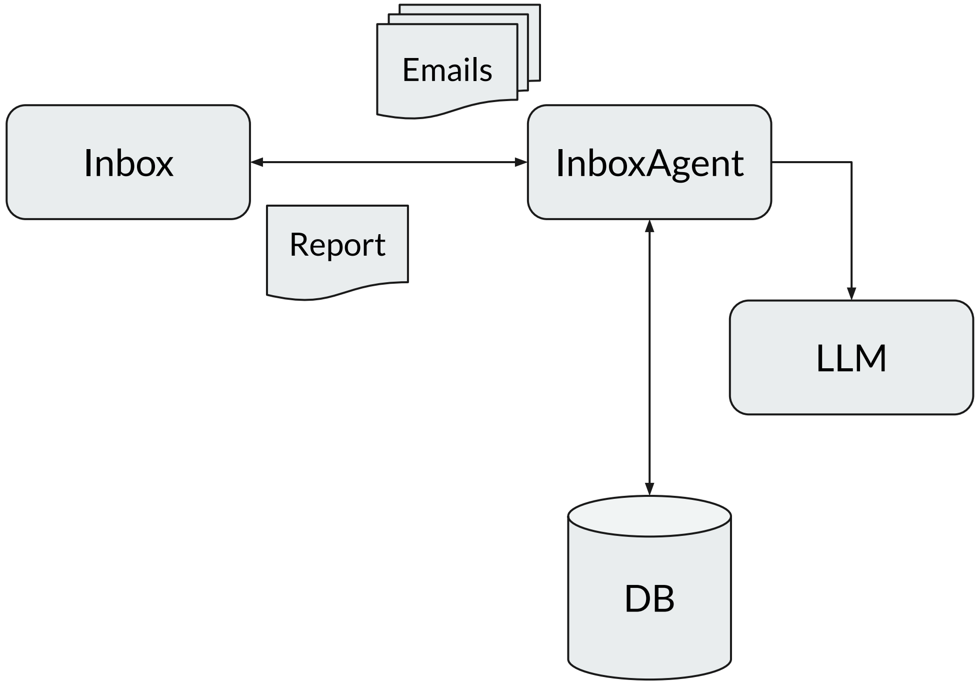
Get your inbox to zero by AI



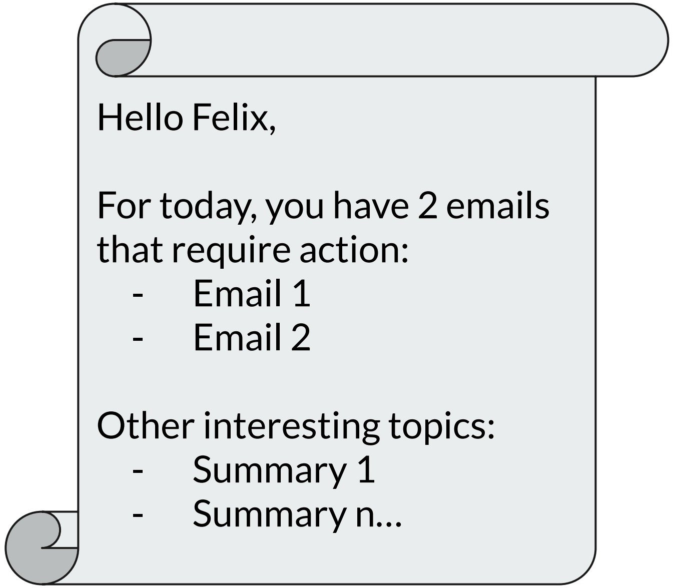
InboxAgent

Inbox Zero by AI:

- Check my emails daily
- Generate a report
  - general overview
  - action required, urgent stuff
  - summaries of important stuff



InboxAgent Report Example



Hello Felix,

For today, you have 2 emails that require action:

- Email 1
- Email 2

Other interesting topics:

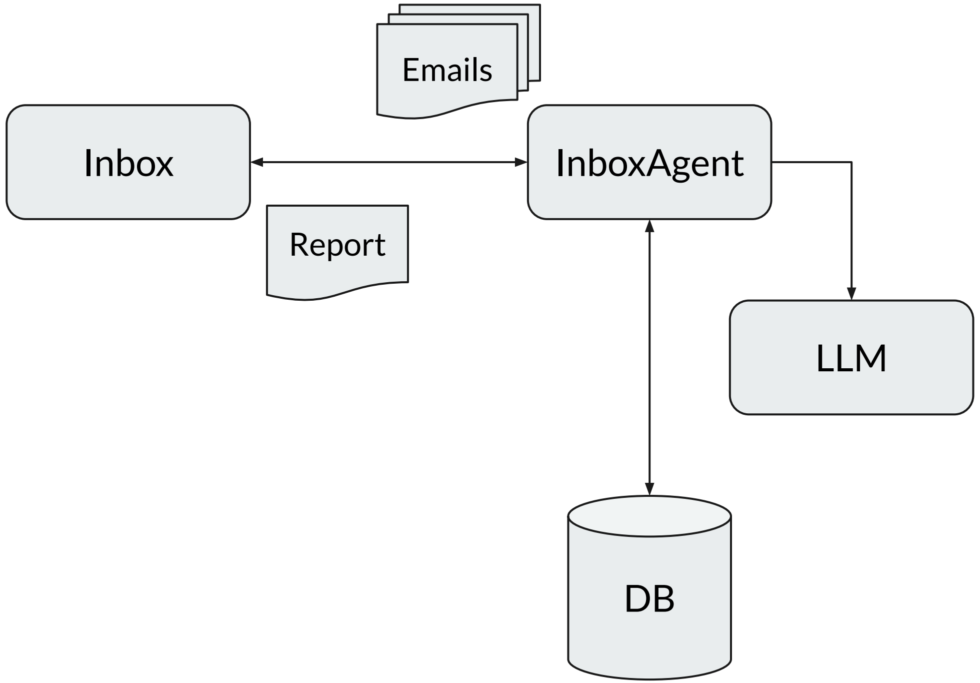
- Summary 1
- Summary n…

InboxAgent

How do we make the agent understand urgency and importance?

We need some form of categorization.

We need to think about email threads over time, referencing older emails.



Where to start?

Understanding LLMs

Large Language Models

Attention Is All You Need
https://arxiv.org/abs/1706.03762

Deep-neural networks for general-purpose language generation

2017: Transformer paper published

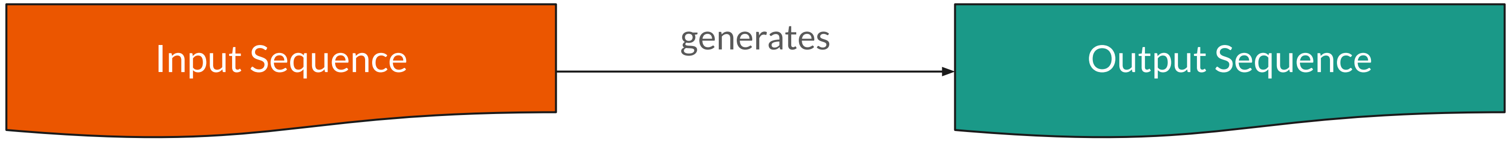
Encoder-Decoder architecture, **self-attention** mechanism

Is **parallelizable** and removes vanishing gradient problem of recurrent neural nets

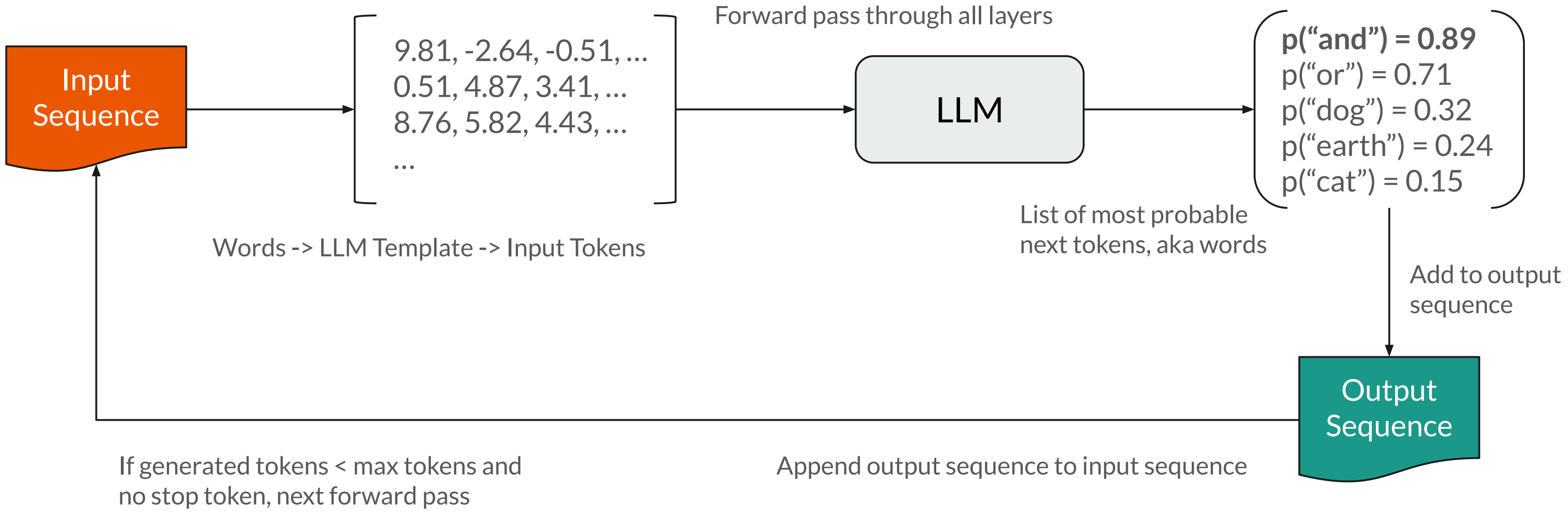
How does a LLM work?

Predict the next word in an output sequence

How does a LLM work?



How does a LLM work? (simplified)



Input Sequence
9.81, -2.64, -0.51, …
0.51, 4.87, 3.41, …
8.76, 5.82, 4.43, …
…
Words -> LLM Template -> Input Tokens
Forward pass through all layers
LLM
p("and") = 0.89
p("or") = 0.71
p("dog") = 0.32
p("earth") = 0.24
p("cat") = 0.15
List of most probable next tokens, aka words
Add to output sequence
Output Sequence
Append output sequence to input sequence
If generated tokens < max tokens and no stop token, next forward pass

How is it trained?

On huge GPU clusters

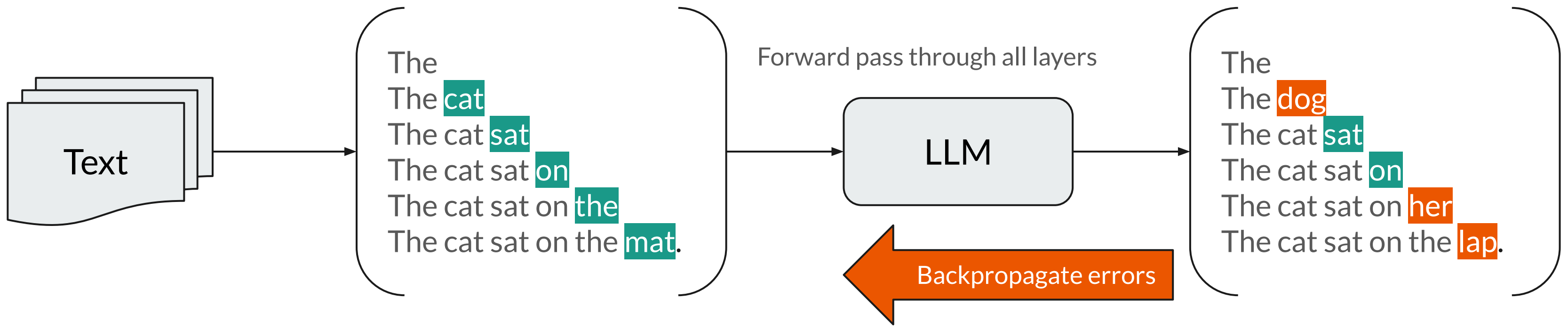
Vast amounts of text data

Self-supervised learning

Backpropagation and gradient descent

https://developer.nvidia.com/blog/nvidia-data-center-gpu-manager-cluster-administration/

How is it trained?



We are handling sensitive data. We can't just push it to a 3rd party API.

Running LLMs locally

No rocket science:

Use Python-based data science tooling, or inference optimized tools like llama.cpp

Plenty of models for free on **Hugging Face**

Challenges are sheer size of models:

Require GPUs with lots of RAM and high memory bandwidth

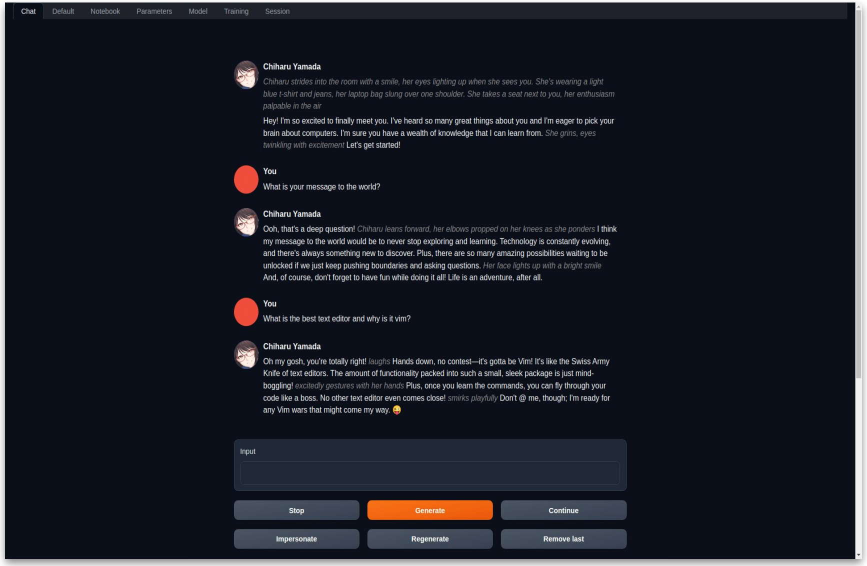
Quantized models for consumer hardware

Running LLMs locally

Text Generation Web UI: Windows, Mac, Linux, all-in-one solution - compatible OpenAI API extension

Ollama: Mac and Linux, strong focus on CLI and tooling around LLMs, all-in-one solution

LM Studio: Windows, Mac, Linux, great UI



Chiharu Yamada

Chiharu strides into the room with a smile, her eyes lighting up when she sees you. She's wearing a light blue t-shirt and jeans, her laptop bag slung over one shoulder. She takes a seat next to you, her enthusiasm palpable in the air

Hey! I'm so excited to finally meet you. I've heard so many great things about you and I'm eager to pick your brain about computers. I'm sure you have a wealth of knowledge that I can learn from. *She grins, eyes twinkling with excitement* Let's get started!

You

What is your message to the world?

Chiharu Yamada

Ooh, that's a deep question! *Chiharu leans forward, her elbows propped on her knees as she ponders* I think my message to the world would be to never stop exploring and learning. Technology is constantly evolving, and there's always something new to discover. Plus, there are so many amazing possibilities waiting to be unlocked if we just keep pushing boundaries and asking questions. *Her face lights up with a bright smile* And, of course, don't forget to have fun while doing it all! Life is an adventure, after all.

You

What is the best text editor and why is it vim?

Chiharu Yamada

Oh my gosh, you're totally right! *laughs* Hands down, no contest—it's gotta be Vim! It's like the Swiss Army Knife of text editors. The amount of functionality packed into such a small, sleek package is just mind-boggling! *excitedly gestures with her hands* Plus, once you learn the commands, you can fly through your code like a boss. No other text editor even comes close! *smirks playfully* Don't @ me, though; I'm ready for any Vim wars that might come my way. 😜

Text Generation Web UI

Practical tips

Running LLMs locally

Start with a quantized 3B model

Pick CPU-only tooling first

Keep model size and context window in mind - both determine RAM requirements

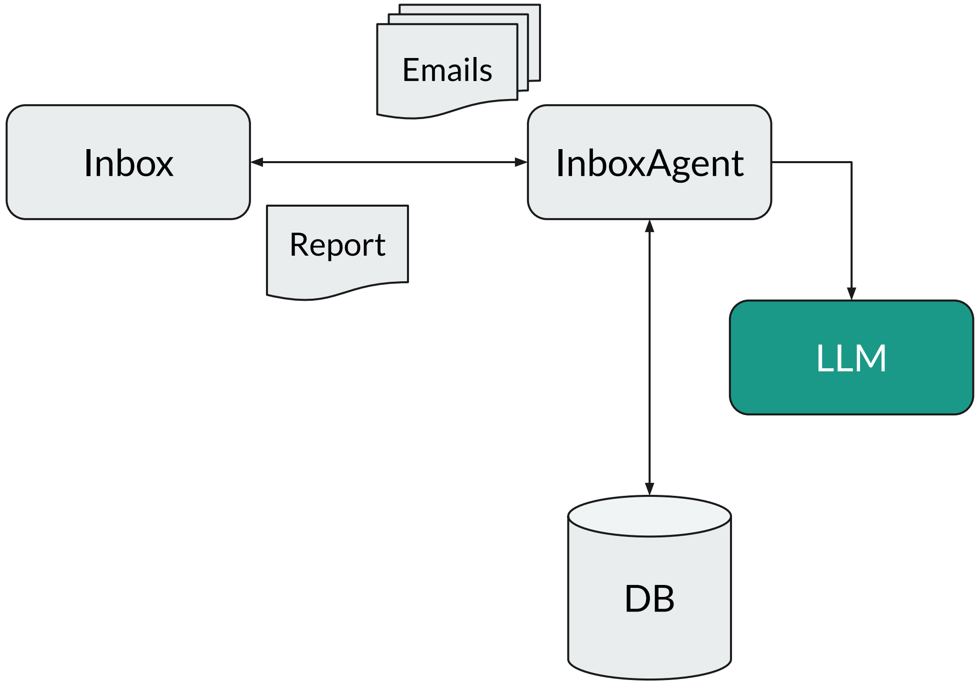
If using GPU, remember that your VRAM is also needed for your screens

InboxAgent

How do we make the agent understand urgency and importance?

We need some form of categorization.

We need to think about email threads over time, referencing older emails.



Our LLM runs, what's next?

Prompting

How do we make the agent understand urgency and importance?

We need some form of categorization.

We need to think about email threads over time, referencing older emails.

Prompt Engineering

The bigger the context, the more likely hallucinations.

Why is prompt engineering even a thing?

Remember: Attention Is All You Need

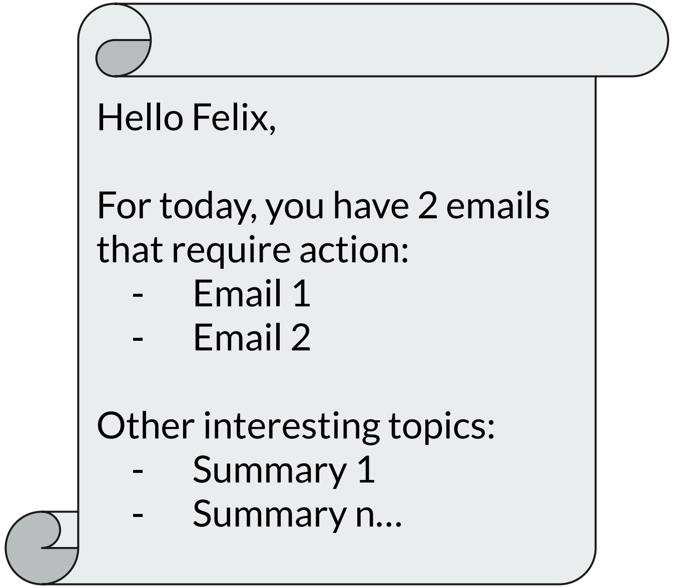
Accurate Responses: Ensures LLM understands and responds relevantly.

Maximizing LLM Potential: Unlocks full capabilities for diverse tasks.

Targeted Outcomes: Tailors AI output for specific styles and content.

Resource Efficiency: Saves time and computing power with precise prompts.

InboxAgent Report Example



Hello Felix,

For today, you have 2 emails that require action:

- Email 1
- Email 2

Other interesting topics:

- Summary 1
- Summary n…

Our Email Problem

Action-required and urgent?

The LLM needs to decide about two different categories.

Hard to put everything in context, it will confuse the LLM.

Important Topics

Is it important enough to be reported?

We only know which topic might be important by having access to recent emails.

Prompting Techniques

Recommended: promptingguide.ai

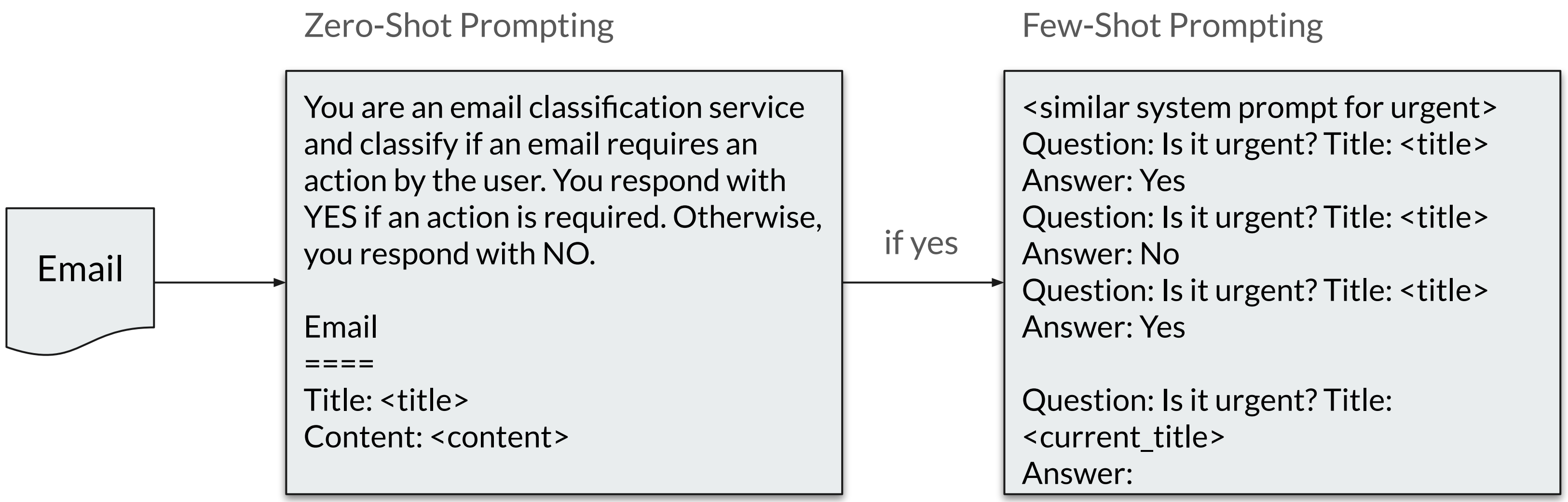
We gonna use a mix of:

Prompt Chaining

Zero-Shots Prompting

Few-Shots Prompting

InboxAgent: Prompt Pipeline Example



You are an email classification service and classify if an email requires an action by the user. You respond with YES if an action is required. Otherwise, you respond with NO.

Email
====
Title: <title>
Content: <content>

<similar system prompt for urgent>
Question: Is it urgent? Title: <title>
Answer: Yes
Question: Is it urgent? Title: <title>
Answer: No
Question: Is it urgent? Title: <title>
Answer: Yes

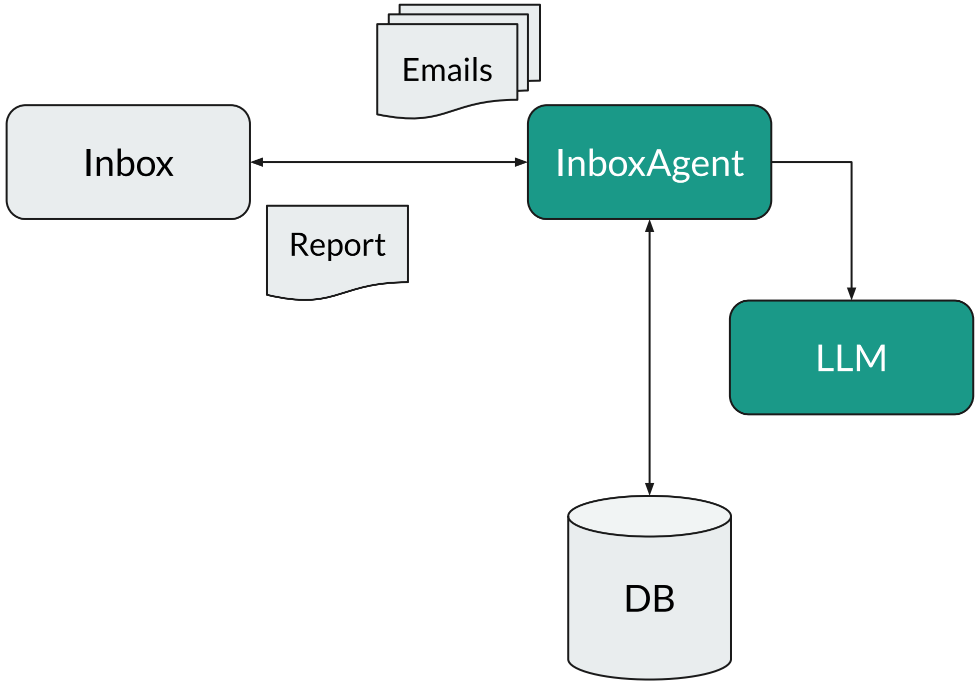
Question: Is it urgent? Title: <current_title>
Answer:

InboxAgent

~~How do we make the agent understand urgency and importance?~~

~~We need some form of categorization.~~

We need to think about email threads over time, referencing older emails.



We need some form of intelligent storage to find important email topics.

Retrieval-Augmented Generation

Retrieval Augmented Generation (RAG)

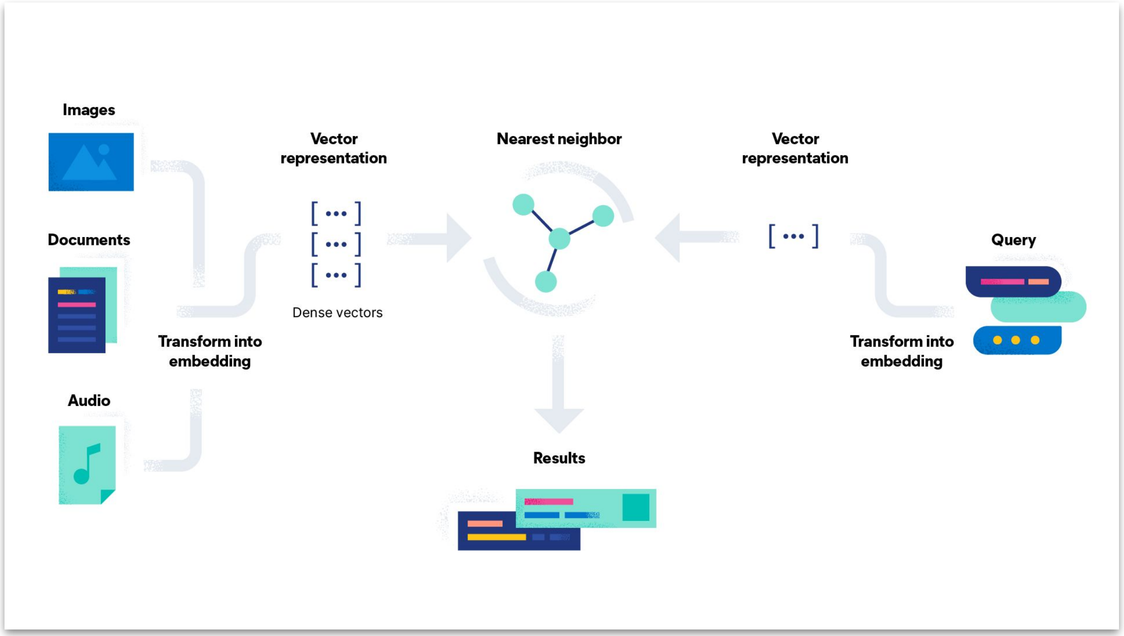
Basically augmenting the prompt with retrieved data from other sources

Combining semantic search and RAG to increase LLM performance and relevancy of its responses

RAG Example Prompt

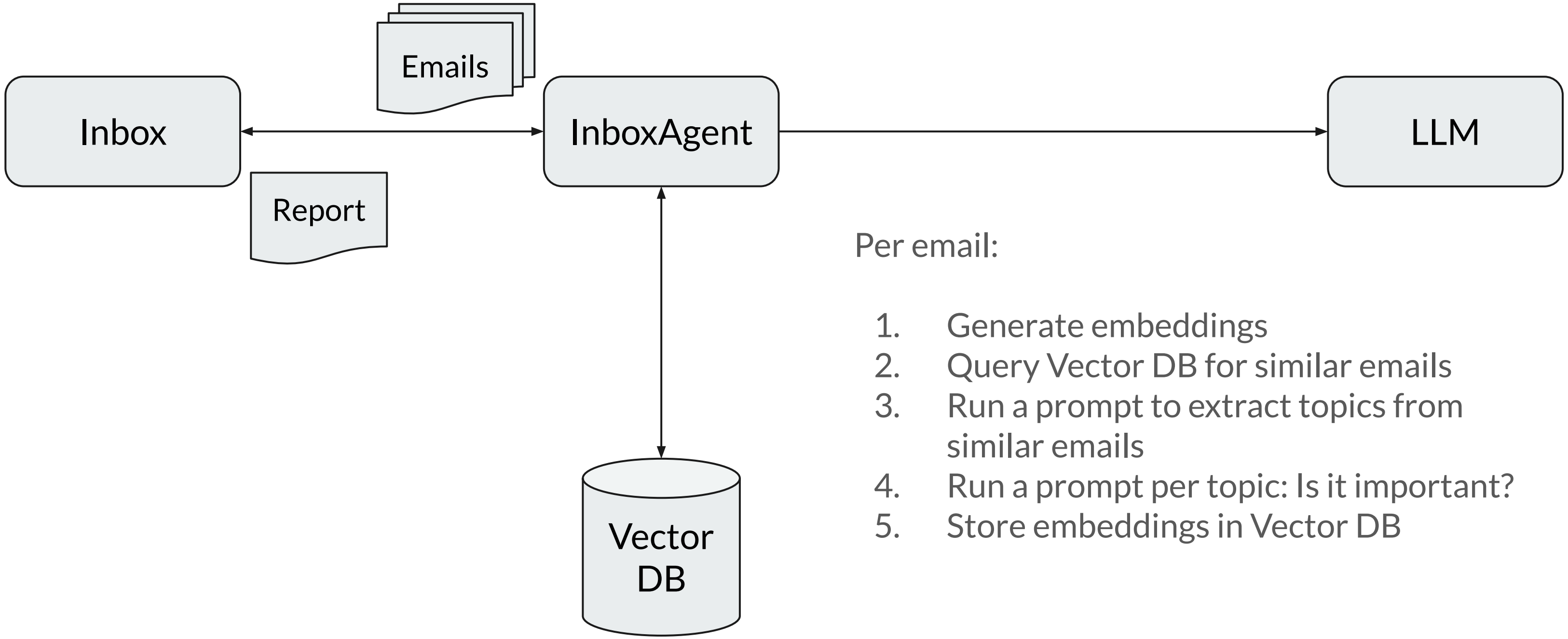
```
Answer the query by only referencing
the following search results.

<Search Result 1>
<Search Result 2>
<Search Result n>

<Query>
```

How does semantic search work?



https://www.elastic.co/what-is/semantic-search

InboxAgent with RAG



Per email:

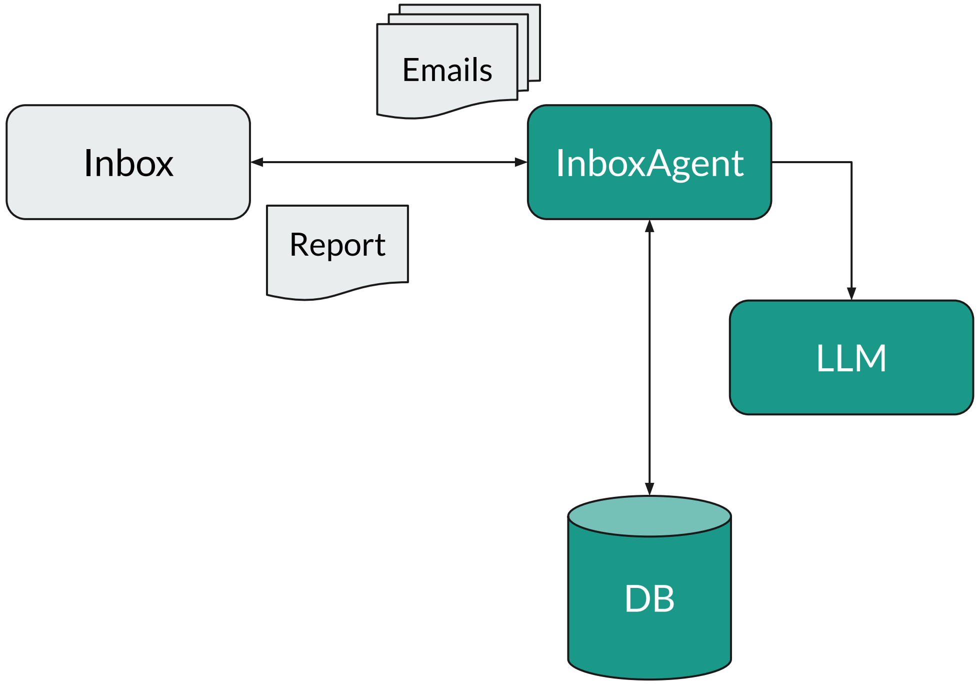
1. Generate embeddings
2. Query Vector DB for similar emails
3. Run a prompt to extract topics from similar emails
4. Run a prompt per topic: Is it important?
5. Store embeddings in Vector DB

InboxAgent

~~How do we make the agent understand urgency and importance?~~

~~We need some form of categorization.~~

~~We need to think about email threads over time, referencing older emails.~~



Tools to take a look at next

Biased selection, there is way more.

LangChain: Python, LLM orchestration

LlamaIndex: Python, specialized on RAGs

Semantic Kernel: .Net, LLM orchestration

LangChain4j: Java, LLM orchestration

Qdrant: Vector Database

Wait... How do we evaluate it?

Evaluation and Maintenance

Running LLMs in Production

Start with an evaluation data set from the beginning

Leverage GPT-4 to evaluate smaller, more specific models

Add guardrails, better focus on prompt pipelines instead of open chat

Monitor responses and latency

Conclusion

LLMs without the hype

LLMs: Predict next token

How are they trained?

How to run them locally?

Prompt Engineering: Chaining, Zero-Shots and Few-Shots Prompting

Retrieval Augmented Generation

Semantic Search

Vector Stores

Evaluation

Thank you

Feel free to ask questions

Slides:

https://fmueller.io/talks/llms-without-the-hype

Email: felix@fmueller.io

LinkedIn: linkedin.com/in/felix-mueller-bln/

Mastodon: @fmueller@mastodon.social

X: @fmueller_bln