Make your LLM app sane again: forgetting incorrect data in real time.

OLIVIER RUAS
olivier@pathway.com

.pathway

TODAY:

Write your realtime LLM app in Pathway:

- Create your chatbot
- Make it learn on realtime data
- Forgetting incorrect data in real time.

AGENDA

1

Intro

Challenges

2

Common solutions

- Finetuning
- RAG

3

RAG with Pathway

Reactive Vector Index

4

Demo

Forgetting data in real-time

LLMs

Powerful AI models are available for developers through API.

01 **EMBEDDINGS**

Text embeddings measure the relatedness of text strings.

02 **CHAT COMPLETIONS**

Take a list of messages as input and return a model-generated message as output.

LLM LIMITATIONS

LLMs excel at answering questions,
but only on topics they remember from their training data.

UNFAMILIAR TOPICS

- Recent events after Dec 2023.
  - Real-time data.
- Personal and confidential data
  - Copyrighted and personal data
  - Your non-public documents.

LLM LIMITATIONS

LLMs model cannot forget: what is learned is learned.

COMPROMISED DATA AS GROUND TRUTH

- Outdated data (Pluto status?)
- False public rumors not corrected yet
- Deliberate Misinformation
- Copyrighted and personal data

.pathway

HOW TO CORRECT THE MODEL?

You can improve its knowledge by providing the additional data.

1. **New information**
2. **Patches**

HOW TO CORRECT THE MODEL?

Waiting for a new LLM model is not an option.

1. **Finetuning**
2. **Prompt engineering**

.pathway

1. FINETUNING

Finetuning adapts a pre-trained model to your data.

- Batch training
- Requires an adapted dataset (query/answer pairs)
- Cannot forget
- Re-training at every request is costly

1. FINETUNING

Finetuning adapts a pre-trained model to your data.

- Batch training
- Requires an adapted dataset (query/answer pairs)
- Cannot forget
- Re-training at every request is costly

.pathway

2. PROMPT ENGINEERING

Prompt engineering provides the relevant information to the LLM to answer correctly the query: you can add a patch.

Given the following data: {input_data} answer this query: {user_query}

PROBLEMS WITH MANUAL PROMPTING

01 MANUAL WORK

02 LIMITED CONTEXT

03 UNABLE TO SAVE DATA FOR FUTURE USE

04 IMPOSSIBLE TO STORE THE RESULT OR EXPOSE

05 NO REAL-TIME DATA

RETRIEVAL-AUGMENTED GENERATION (RAG)

01 COMPUTING THE EMBEDDINGS OF THE QUERY

02 FINDING THE MOST SIMILAR DOCUMENTS
(USING VECTOR INDEX/DATABASE)

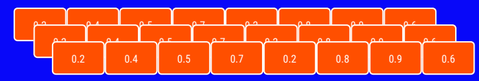
03 PROMPT ENGINEERING

WHAT ARE VECTOR EMBEDDINGS?

pathway

Audio

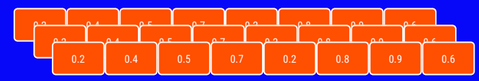
Audio model



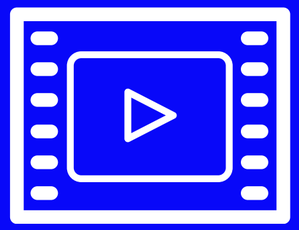
Audio Vector Embeddings

Text

Text model

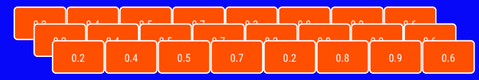


Text Vector Embeddings



Videos

Videos model



Video Vector Embeddings

.pathway

POPULAR RAG USE CASES:

- Build your own AI chatbot over your own data
- Include real time data
- Provide the context to queries to avoid potential incorrect answers

RAG EXAMPLE: CONFIDENTIAL DATA

- Your company has confidential data.
- Ask questions about this data: “What is this company's budget?”
- Find all the most relevant data about this using the vector index.
- Prompt engineering with the summary of the news.

WHAT HAPPENS IF THE RAG DATA IS COMPROMISED?

As previously:

- Outdated data
- False public rumors not corrected yet
- Deliberate Misinformation
- Copyrighted and personal data

- Good news: context can be forgotten.
- Remove the data from your index store, and your next query will not include it.

RAG supports the removal of knowledge:

- Incorrect data
- Confidential data

RAG EXAMPLE: CONFIDENTIAL DATA

- Confidential data is heavily regulated.
- For legal reasons, you have to remove data from your system.
- Keeping those might result in lawsuits.

.pathway

RAG EXAMPLE: CONFIDENTIAL DATA

- Add new data, and it’ll be taken into account.
- Remove data and it’ll be forgotten by your query system.

Reactivity is key

.pathway

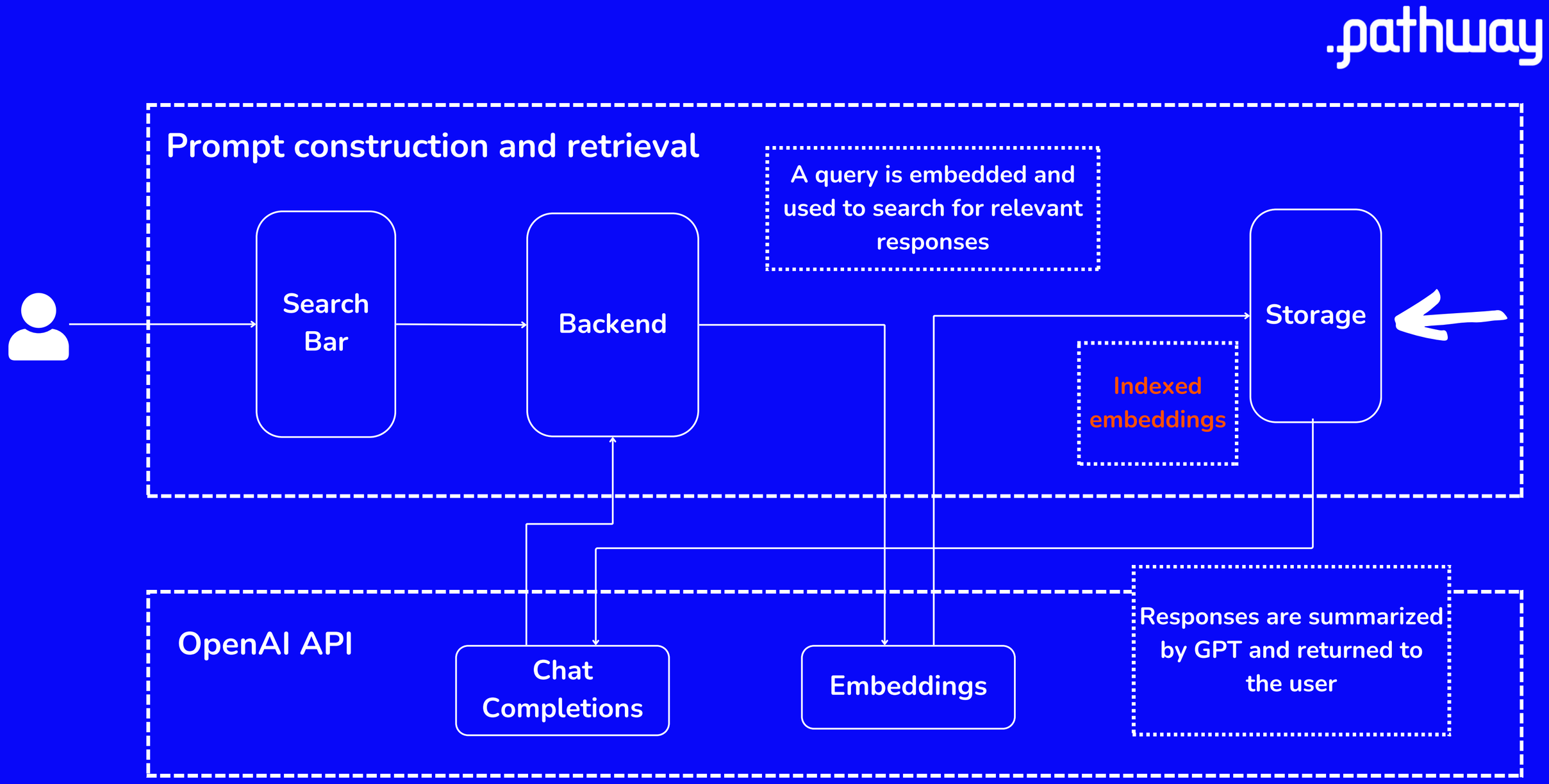
SOLUTION: USE A REAL TIME VECTOR INDEX

01 CAN FORGET, WELL ADAPTED TO LIVE DATA STREAM

02 SHOULD BE REACTIVE TO UPDATES

PRACTICE:
BUILD A CHATBOT

- Privately owned documents: financial documents.
- Need to remove the data fast once the legal period is over.
- Build in pure Python with pathway

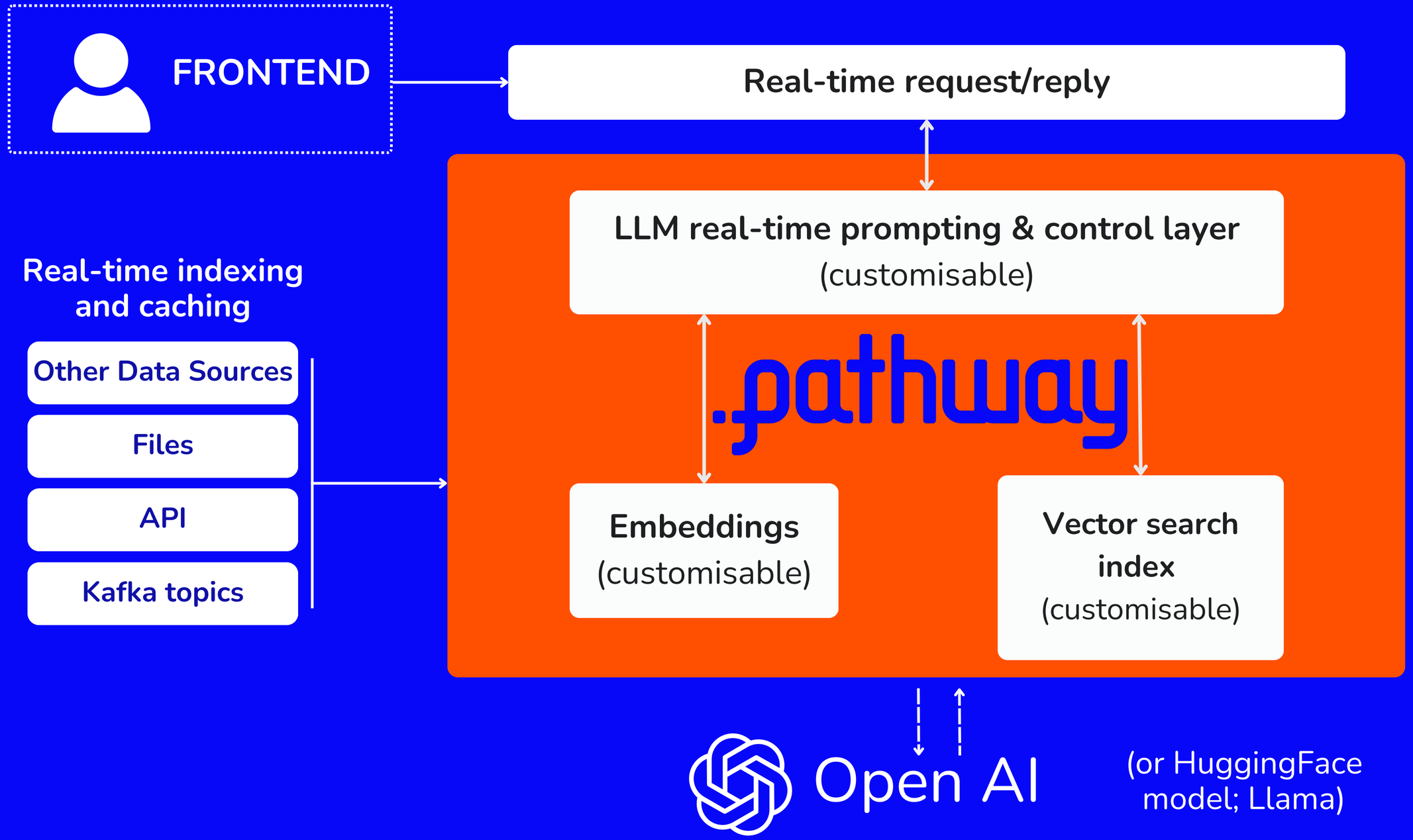

Prompt construction and retrieval
A query is embedded and used to search for relevant responses
Search Bar
Backend
Storage
Indexed embeddings
OpenAI API
Chat Completions
Embeddings
Responses are summarized by GPT and returned to the user

.pathway

The fastest data processing engine supporting unified workflows for batch, streaming data, and LLM applications

Simplicity of Python + Power of Rust

pathwaycom/pathway


FRONTEND
Real-time request/reply
LLM real-time prompting & control layer
(customisable)
Real-time indexing
and caching
Other Data Sources
Files
API
Kafka topics
.pathway
Embeddings
(customisable)
Vector search
index
(customisable)
Open AI
(or HuggingFace
model; Llama)

```python
documents = pw.io.fs.read("./documents/", format="binary", with_metadata=True)
embedder = embedders.OpenAIEmbedder(model="text-embedding-ada-002")
text_splitter = TokenCountSplitter(max_tokens=400)

chat = llms.OpenAIChat(model="gpt-3.5-turbo",temperature=0.05)
vector_server = VectorStoreServer(
    documents,
    embedder=embedder,
    splitter=text_splitter,
    parser=ParseUnstructured(),
)
webserver = pw.io.http.PathwayWebserver(host="0.0.0.0", port=8000)

queries, writer = pw.io.http.rest_connector(
    webserver=webserver,
    schema=PWAIQuerySchema,
    autocommit_duration_ms=50,
    delete_completed_queries=True,
)

results = queries + vector_server.retrieve_query(
    queries.select(
        query=pw.this.query,
        k=1,
        metadata_filter=pw.cast(str | None, None),
        filepath_globpattern=pw.cast(str | None, None)
    )
).select(
    docs=pw.this.result,
)

results += results.select(
    rag_prompt=prep_rag_prompt(pw.this.query, pw.this.docs)
)
results += results.select(
    result=chat(
        llms.prompt_chat_single_qa(pw.this.rag_prompt),
    )
)

writer(results)

pw.run(monitoring_level=pw.MonitoringLevel.NONE)
```

.pathway

.pathway

REACTIVITY IS KEY

- Garbage in, garbage out: you need to update your index ASAP.
- Streaming is the way to go: batch results in inconsistencies.
- Pathway Vector index is reactive and easy to use.

TAKEAWAYS

- LLMs can be wrong, the issue is mainly coming from the training data.
- RAG is the only existing solution for correcting LLM-limited knowledge that can adapt in real time.
- Reactivity is key: your indexing should reflect changes in your data in real time.

THANK YOU!

Try the demo yourself at:

https://github.com/pathway-labs/realtime-indexer-qa-chat

olivier@pathway.com