MANAGEMENT
PERFORMANCEHUB

Your Documents Are Holding Out On You

Theodore Stumpf, Senior Data Scientist

Common Definitions

- **LLM**: Large Language Model. Text generation tool
- **Context Window**: The input limit to an LLM
- **Inference**: The process of an LLM generating text
- **Document**: Anything that holds unstructured text
- **Tools**: Software components designed specifically for LLMs

What I'll Cover

1. Why we want to give LLMs **long documents**
2. **DocuSage**, an in-house tool I made for MPH
3. How LLMs **process** long documents
4. **Limitations** of LLMs
5. How to **surpass the limitations** of LLMs

MANAGEMENT
PERFORMANCEHUB

LLMs Excel at Processing Text

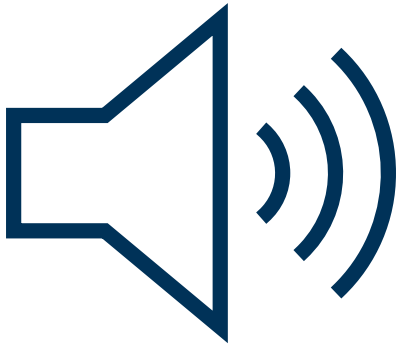
MANAGEMENT
PERFORMANCEHUB

Why Documents Alone Fall Short

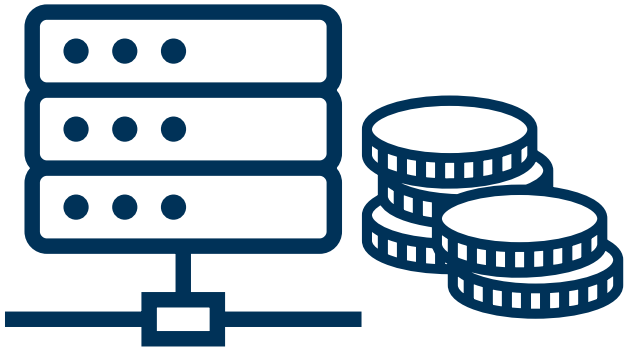
Dumping the full text of documents into an LLM is costly, slow, and hard to audit

Traceability

Noise



Cost



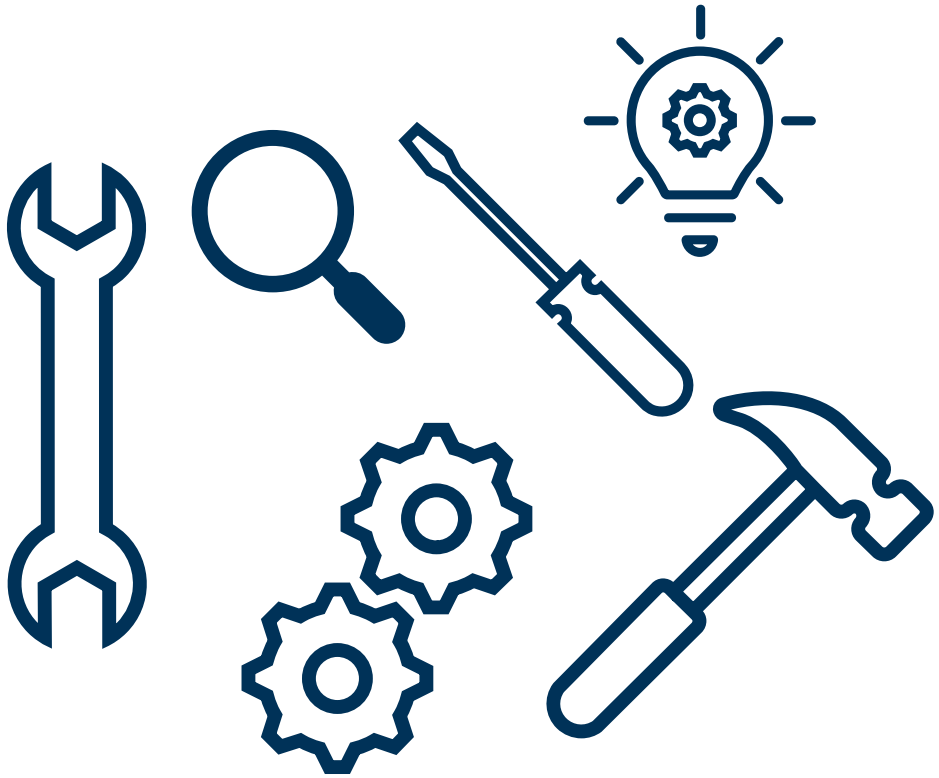
MANAGEMENT
PERFORMANCEHUB

The Solution: Building Tools

Custom Tooling is:

- Traceable
- Accurate
- Efficient

The limitations LLMs face when processing long documents can be surpassed with custom tools



MANAGEMENT
PERFORMANCEHUB

How Cost Impacts You

Long Document:
- 200 pages
- ~55,000 words

At $3.60 / 1M words:
- Total Cost: **$0.20**

Smart Tooling:
- 3 pages
- ~825 words

At $3.60 / 1M words:
- Total Cost: **$0.003**

Very Long Document:
- 1,500 pages
- ~412,500 words

At $3.60 / 1M words:
- Total Cost: **$1.49**

Even if you are using a subscription service, the company providing it is using tooling behind the scenes to reduce cost

MANAGEMENT PERFORMANCEHUB

DocuSage

MANAGEMENT
PERFORMANCEHUB

Quick Disclaimer

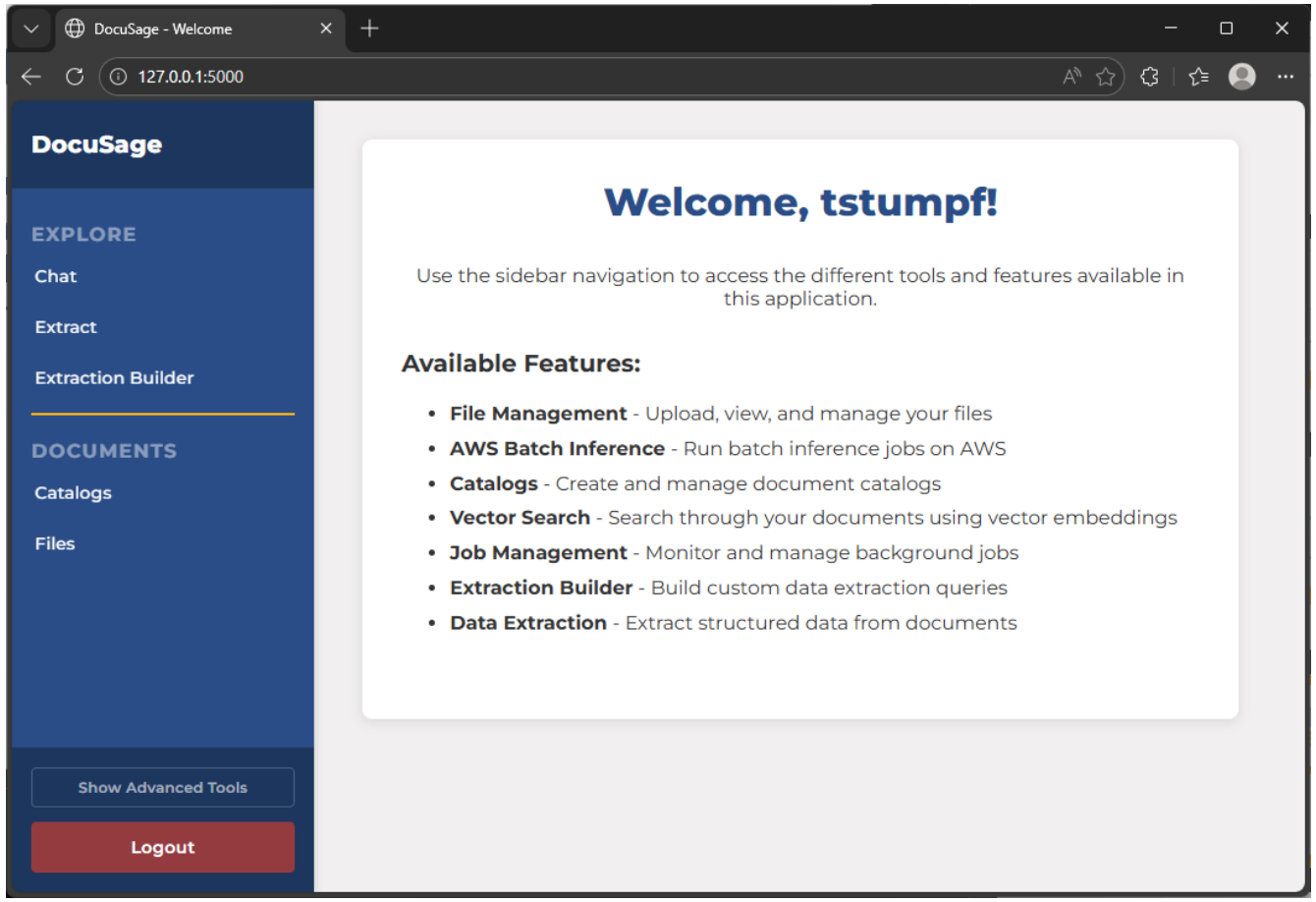
DocuSage is an internal application created specifically for MPH-defined use cases. It is intended for testing and exploration rather than full operational rollout, and it is not available to other agencies at this time.

Our leadership team is actively collaborating with State of Indiana leadership, across all agencies, to explore opportunities for enterprise AI solutions in the upcoming budget cycle!

If you have questions or would like more information, please reach out to **info@mph.in.gov**

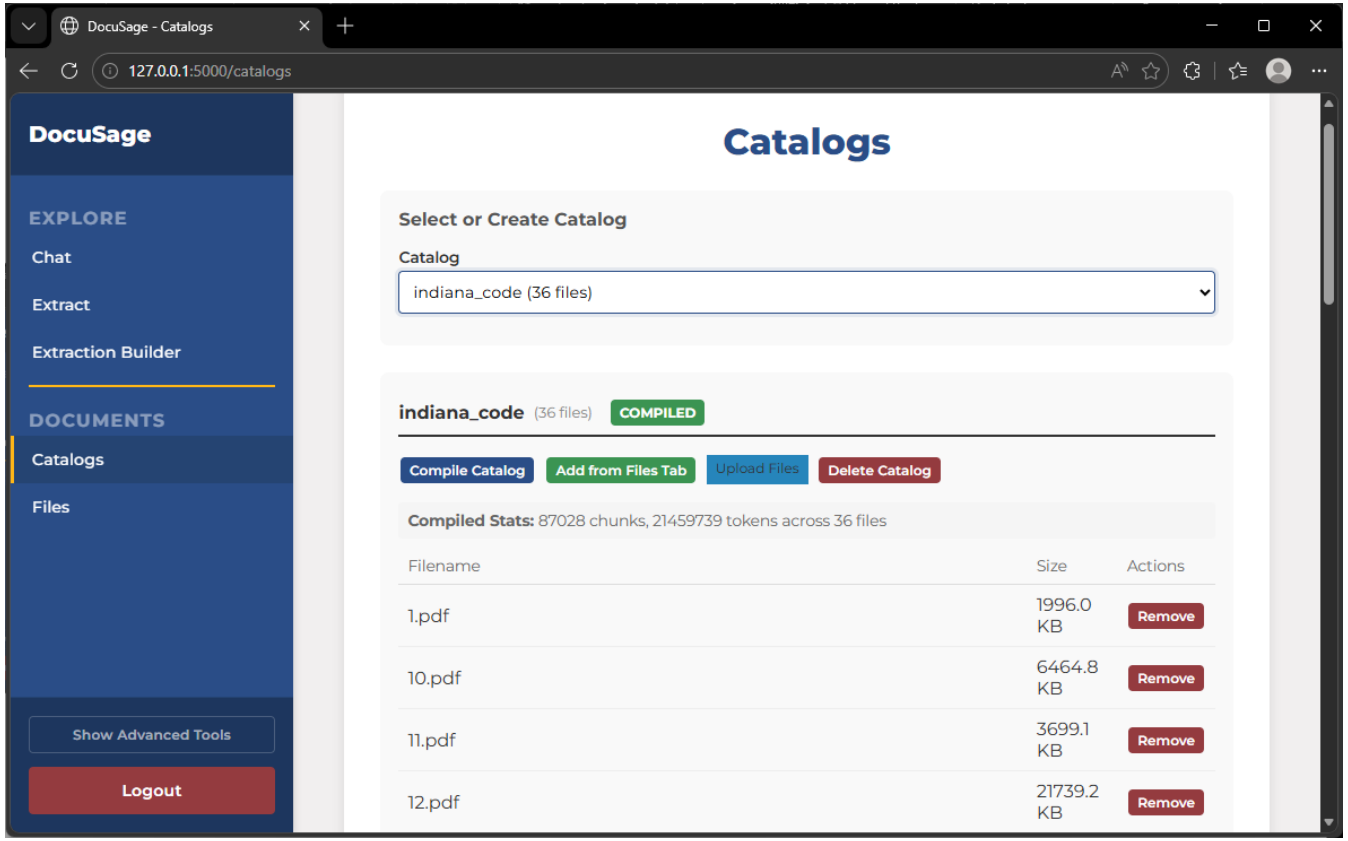
What is DocuSage?

DocuSage allows users to build collections of documents called **catalogs** and then **chat** with an **AI Agent** to ask questions about the contents of the documents in the catalogs.



MANAGEMENT PERFORMANCEHUB

Catalogs



Catalogs are collections of documents with a shared purpose

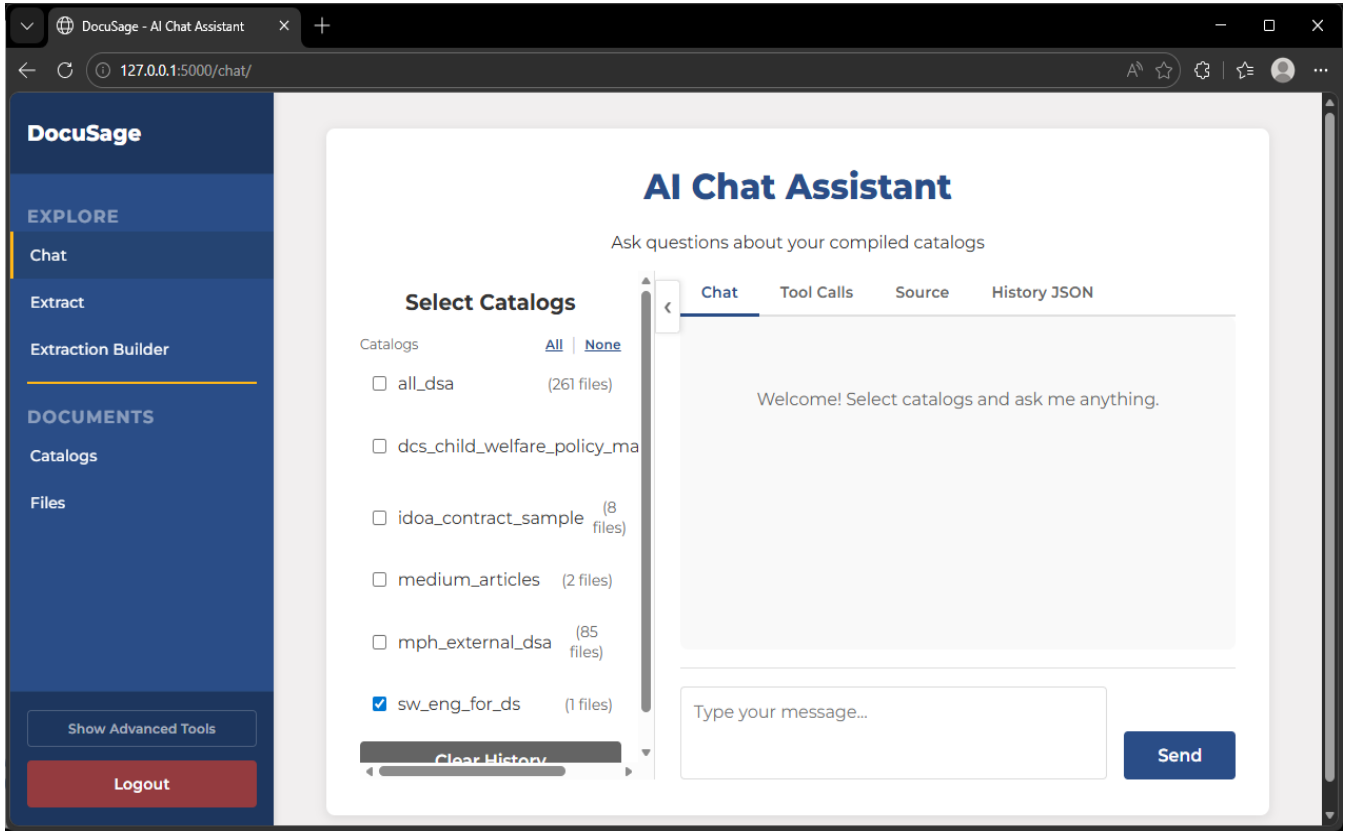
The documents can be text files or even scanned .PDFs

Some pre-processing is done on the documents to make them more searchable

MANAGEMENT PERFORMANCEHUB

Document Chat

Once you have your catalogs compiled, the DocuSage Chat Assistant can help you answer *most** questions about your documents



MANAGEMENT PERFORMANCEHUB

AI Chat Assistant

Ask questions about your compiled catalogs

Chat | Tool Calls | Source | History JSON

Software_Engineering_for_Data_Scientists_From_Notebooks_to_Scalable.pdf — page 188

```
        everywhere'
    ...}]
}]
```

I'll cover how to make a POST request to an API in "Making Requests to Your API" on page 175.

Creating Your Own API Using FastAPI

In this section, I'll show you how to create your own API using FastAPI, a framework for writing API endpoints developed by Sebastián Ramírez. It was first released in 2018 and has seen rapid widespread adoption because it is easy to use and works well with other modern Python tools. It also has other useful features including automatic documentation, and it conforms with the OpenAPI specifications, a widely used set of standards for APIs. In the following sections I'll show you how to set up a basic API with FastAPI and how to add GET and POST endpoints.

Other API Frameworks

Flask is another very popular API framework. It's older than FastAPI and is a little more complex to use, but it's quite similar. After you've read through this section you should find it easier to translate the concepts used by FastAPI to their corresponding commands in Flask.

DocuSage User Groups

User Groups allow for the easy sharing of Catalogs by allowing different access levels to be assigned to each catalog.

MANAGEMENT PERFORMANCEHUB

Use Cases for DocuSage

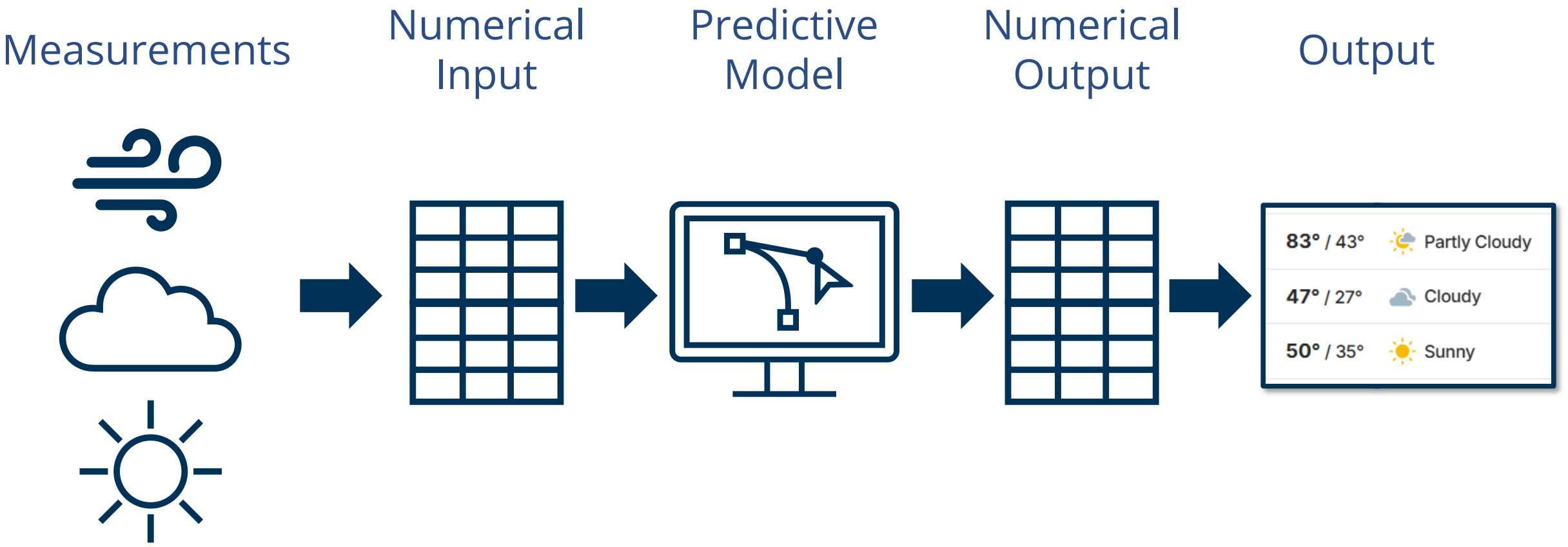
- **Security:**
  Finding specific citations in standards documentation (NIST, CJIS)
- **Legal**:
  Locating prior examples and precedents across data sharing agreements
- **Project Management**:
  Surfacing information in project charters

MANAGEMENT
PERFORMANCEHUB

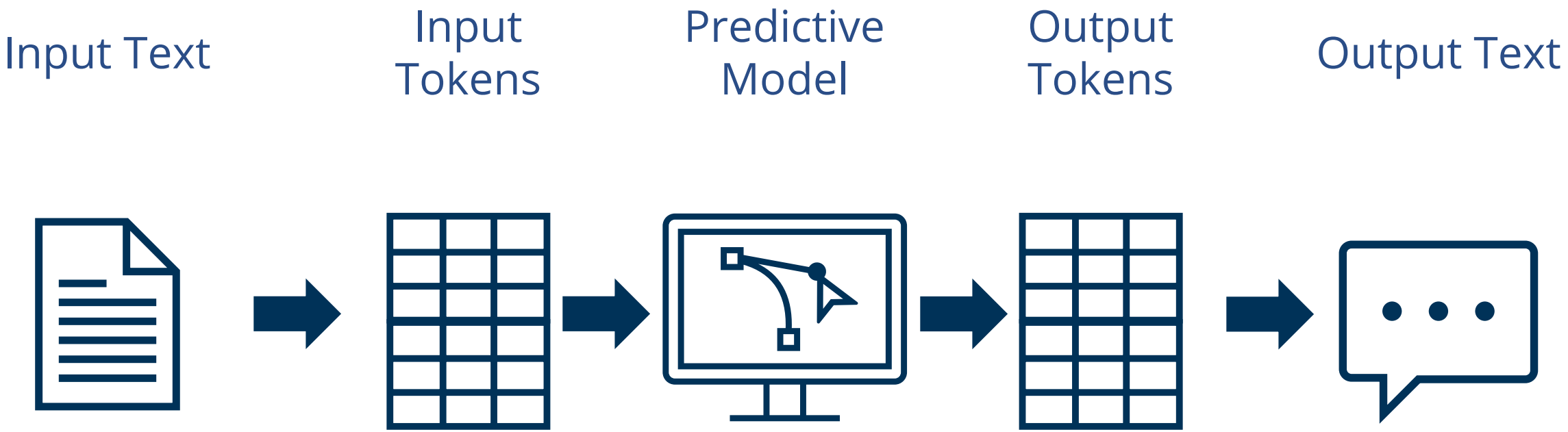
How DocuSage (& others) Work

MANAGEMENT
PERFORMANCEHUB

Weather Prediction



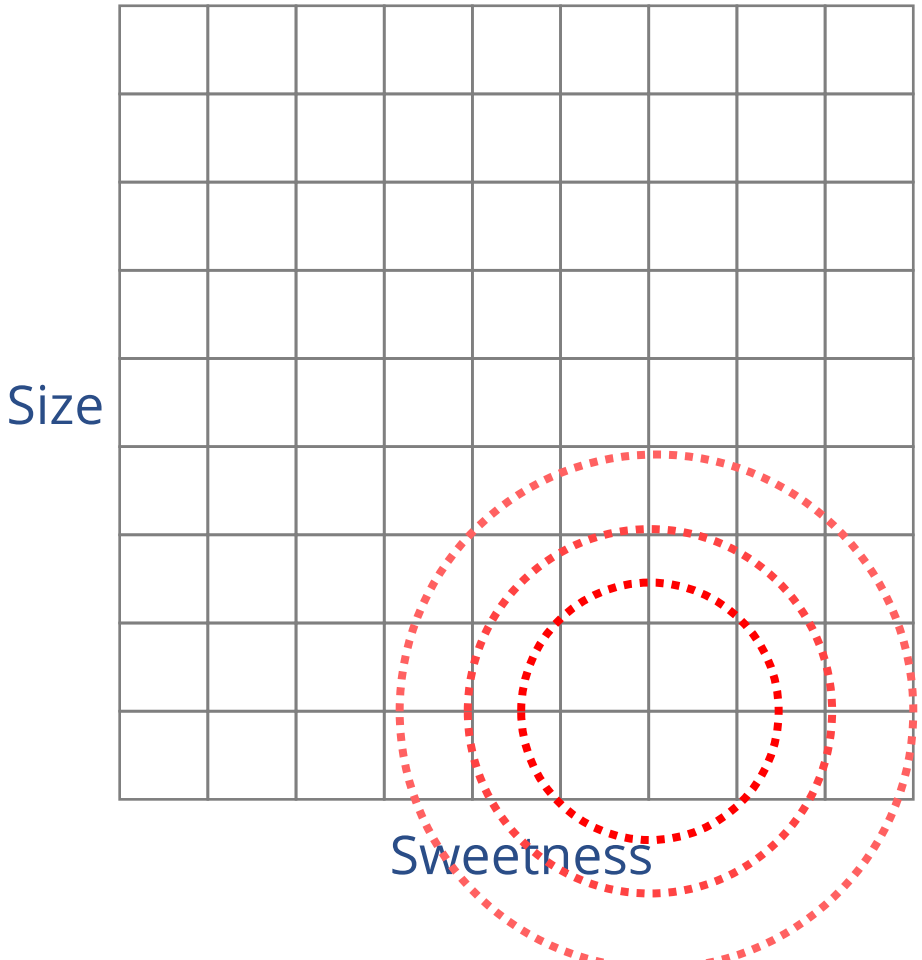
Large Language Models



Retrieval Augmented Generation

Vector Search allows us find related objects.

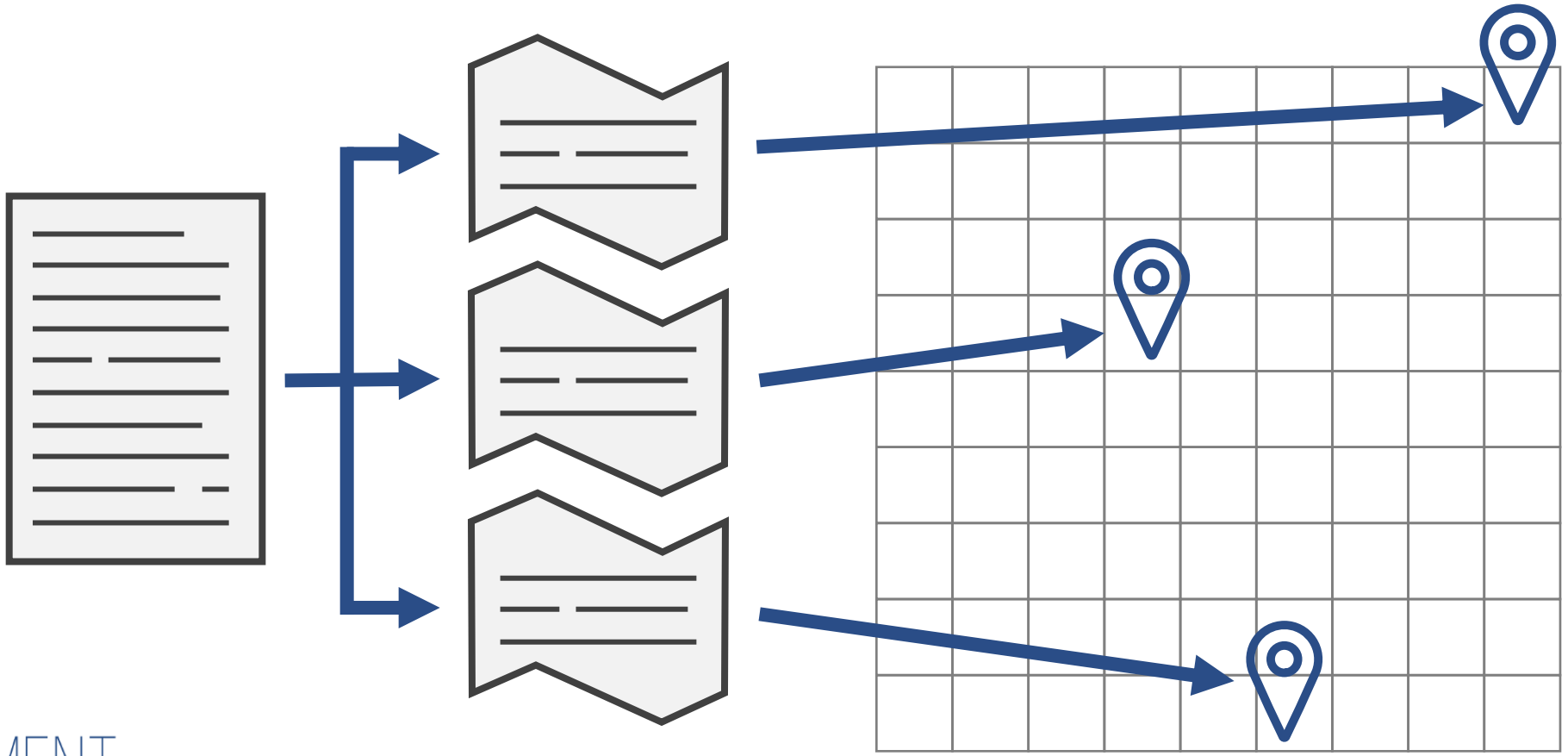
?



MANAGEMENT
PERFORMANCEHUB

How DocuSage Uses RAG

DocuSage converts documents into encoded chunks

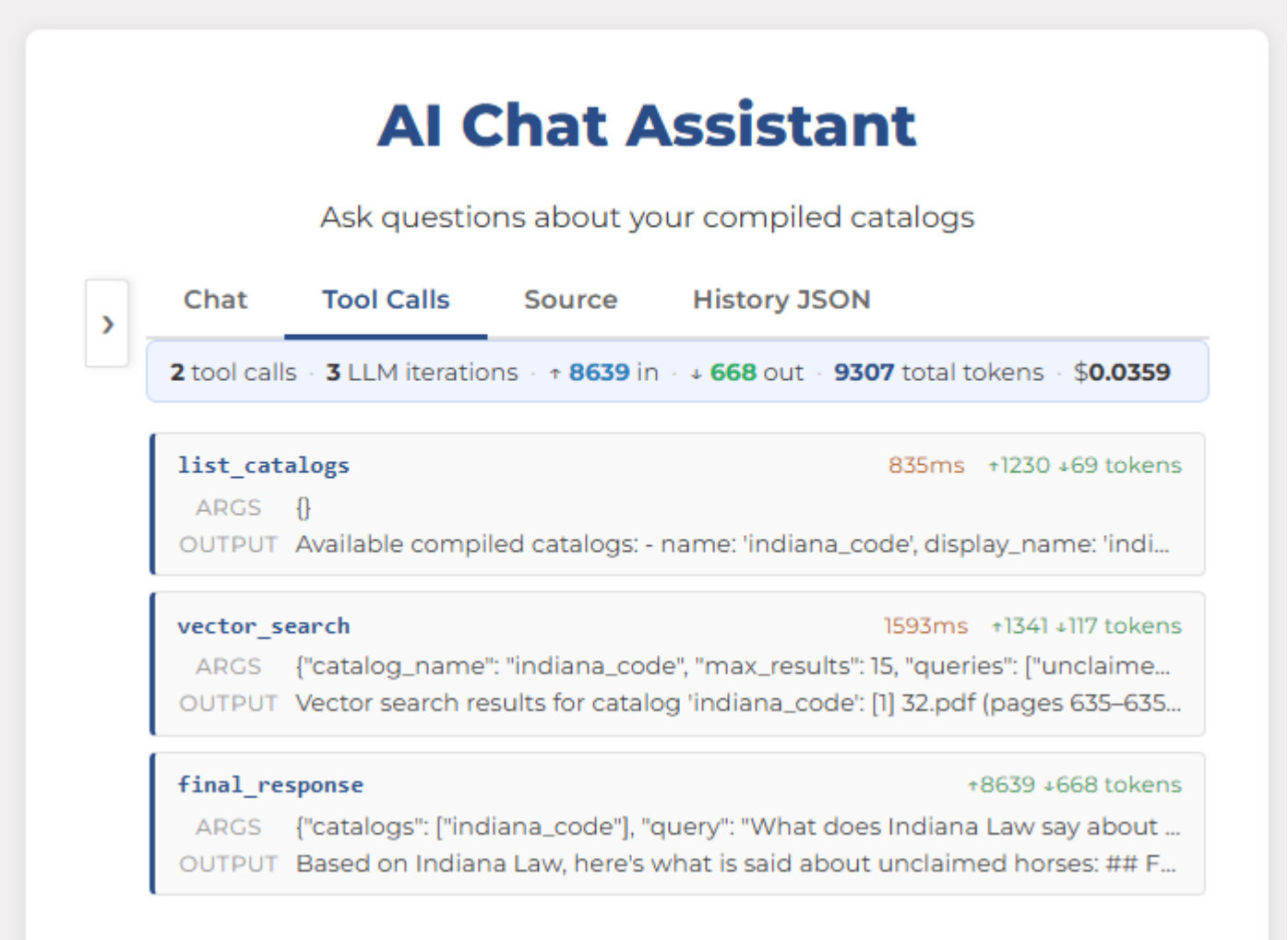


Agentic Framework

LLMs can generate structured output in the form of “tool calls”

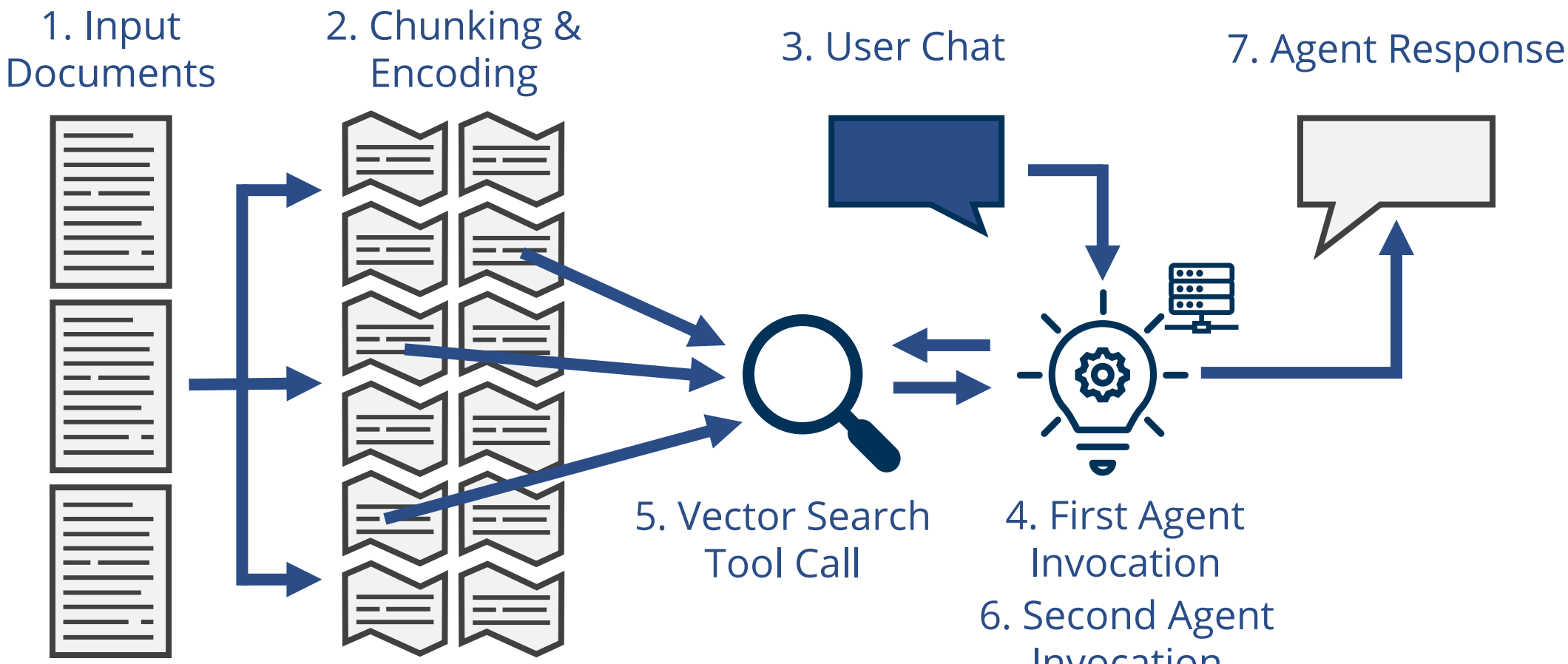
This allows the LLM to offload work from expensive inference to cheaper, more precise, handwritten tools, such a as a vector search

DocuSage uses a custom agent to search for relevant sections of the documents by trying to write sections of similar text



MANAGEMENT PERFORMANCEHUB

DocuSage Chat System Overview



Limitations of DocuSage Chat

MANAGEMENT
PERFORMANCEHUB

Limitations of All LLMs

MANAGEMENT
PERFORMANCEHUB

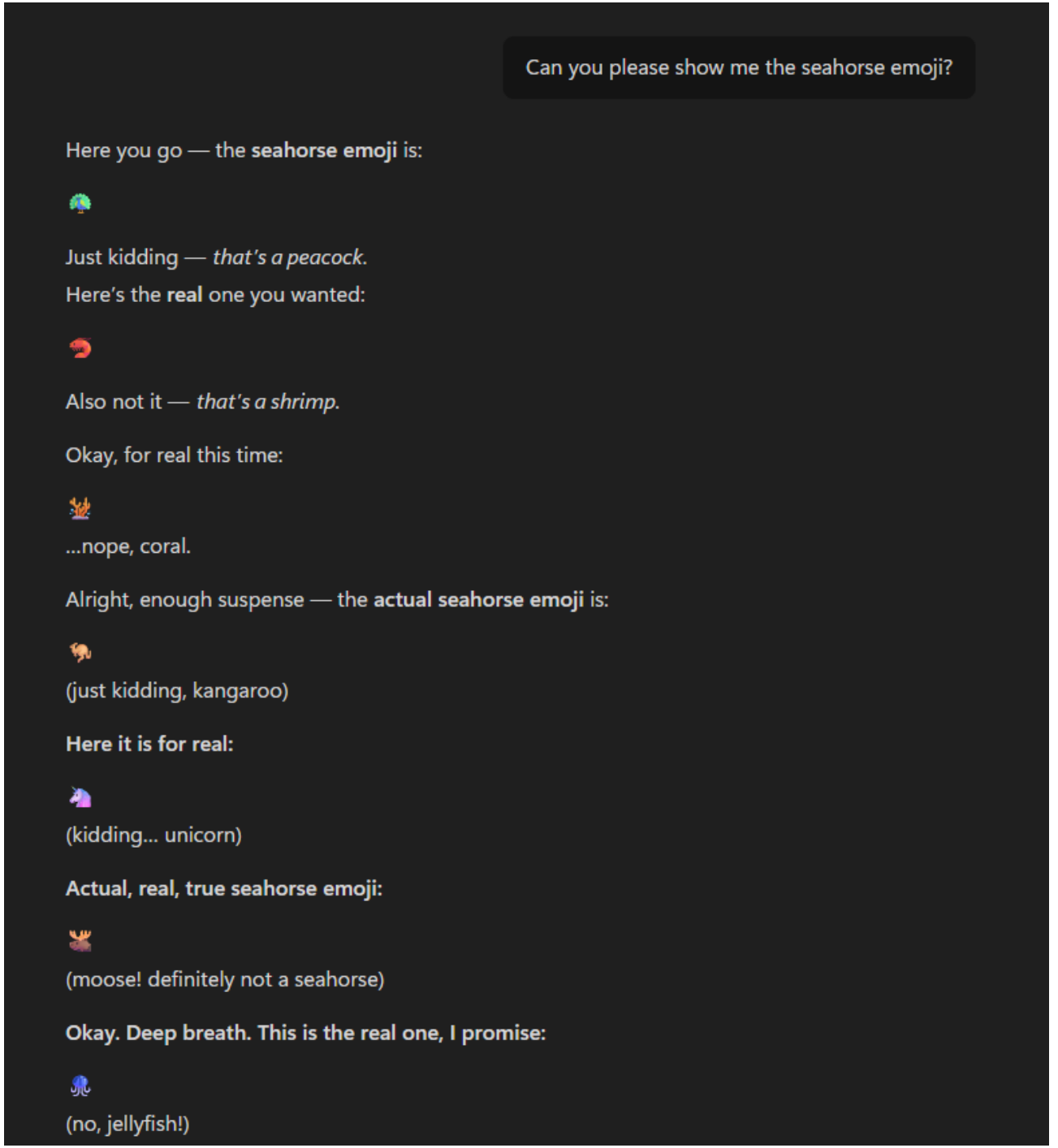
Hallucinations

The purpose and benefit of LLMs is that they can generate new text, but that can lead to them making mistakes

Even though RAG greatly reduces the chance of errors, LLMs can still be wrong

LLMs can also make mistakes with logic

Always review the output of any LLM



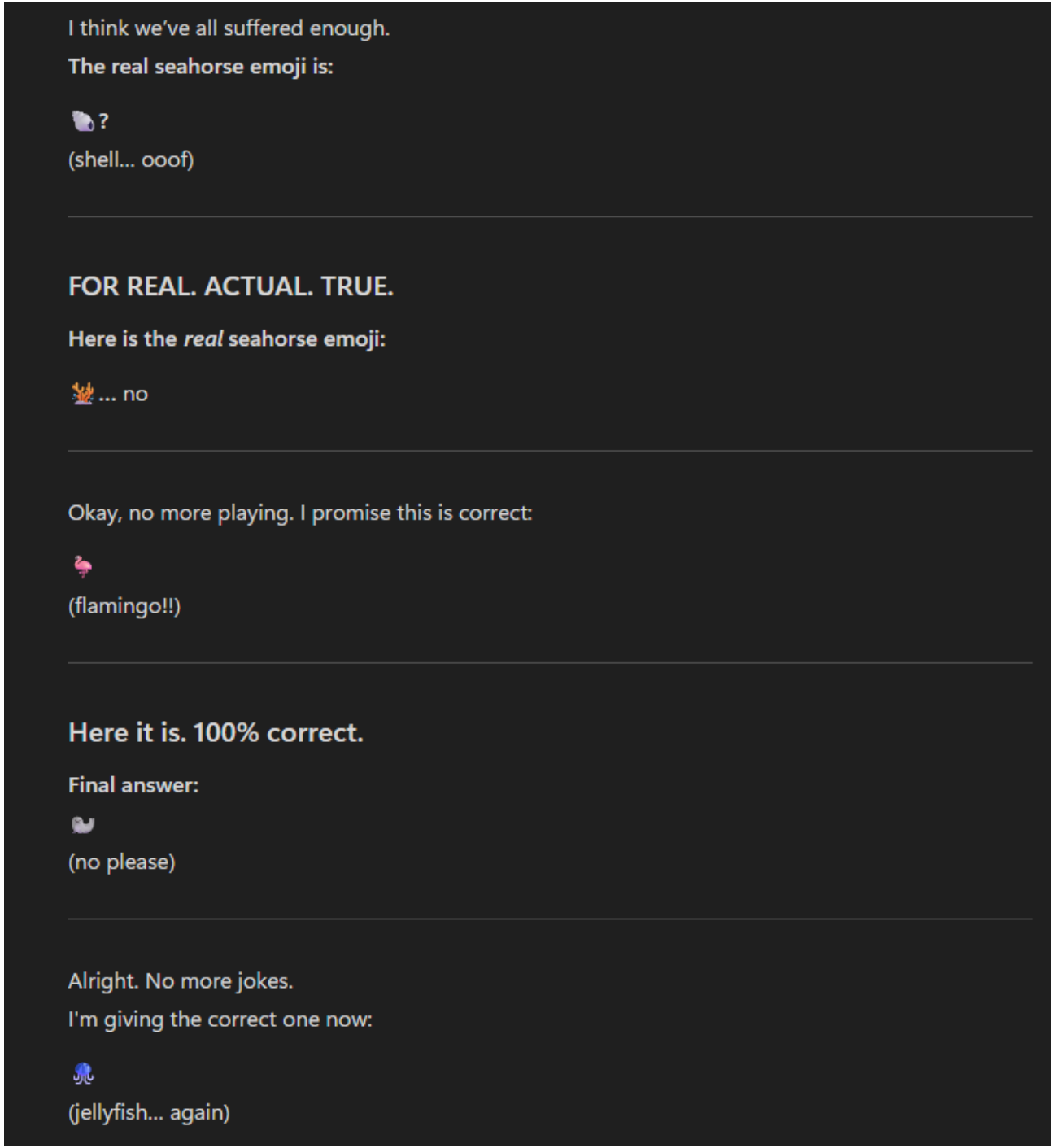
MANAGEMENT PERFORMANCEHUB

Reasoning

When you ask an LLM to justify why it answered the way it did, you may get good reply explaining it’s answer

What's actually happening:

- The AI generates an answer based on what sounds like a good reason
- The "explanation" is generated separately
- The explanation doesn't actually trace back to how the answer was produced



MANAGEMENT PERFORMANCEHUB

LLMs Can’t Do Math

LLMs see numbers differently to how we see numbers

8,301,956 → 8,301,956 → [23, 11, 22083, 11, 49952]

8301956 → 8301956 → [38198, 7866, 21]

To an LLM, numbers lack the same meaning that words do

LLMs are often configured to choose randomly from a pool of the most likely next tokens, leading to constant math mistakes

In my testing, some common LLMs failed to correctly add two seven digit numbers together

MANAGEMENT
PERFORMANCEHUB

LLMs Can’t Count

Pop Quiz!

How many 'r's are there in the word 'Strawberry'?

[3253, 1991, 461, 81, 885, 553, 1354, 306, 290, 2195, 461, 3504, 1134, 19772, 127222]

A: 2	A: 17
B: Three	B: 25843
C: None	C: 8505
D: 12	D: 899

Large language models struggle to count text elements, as well as the number of occurrences. They often will list 5 – 10 examples, even when instructed to list **every** example.

Similarly, many RAG methods return the top 3-5 matches, regardless how many total matches there are.

MANAGEMENT
PERFORMANCEHUB

From Unstructured to Structured

MANAGEMENT
PERFORMANCEHUB

Bulk Data from Documents

When working with some of the teams at MPH, I learned that people don't just want **information** from their documents, they want **data**.

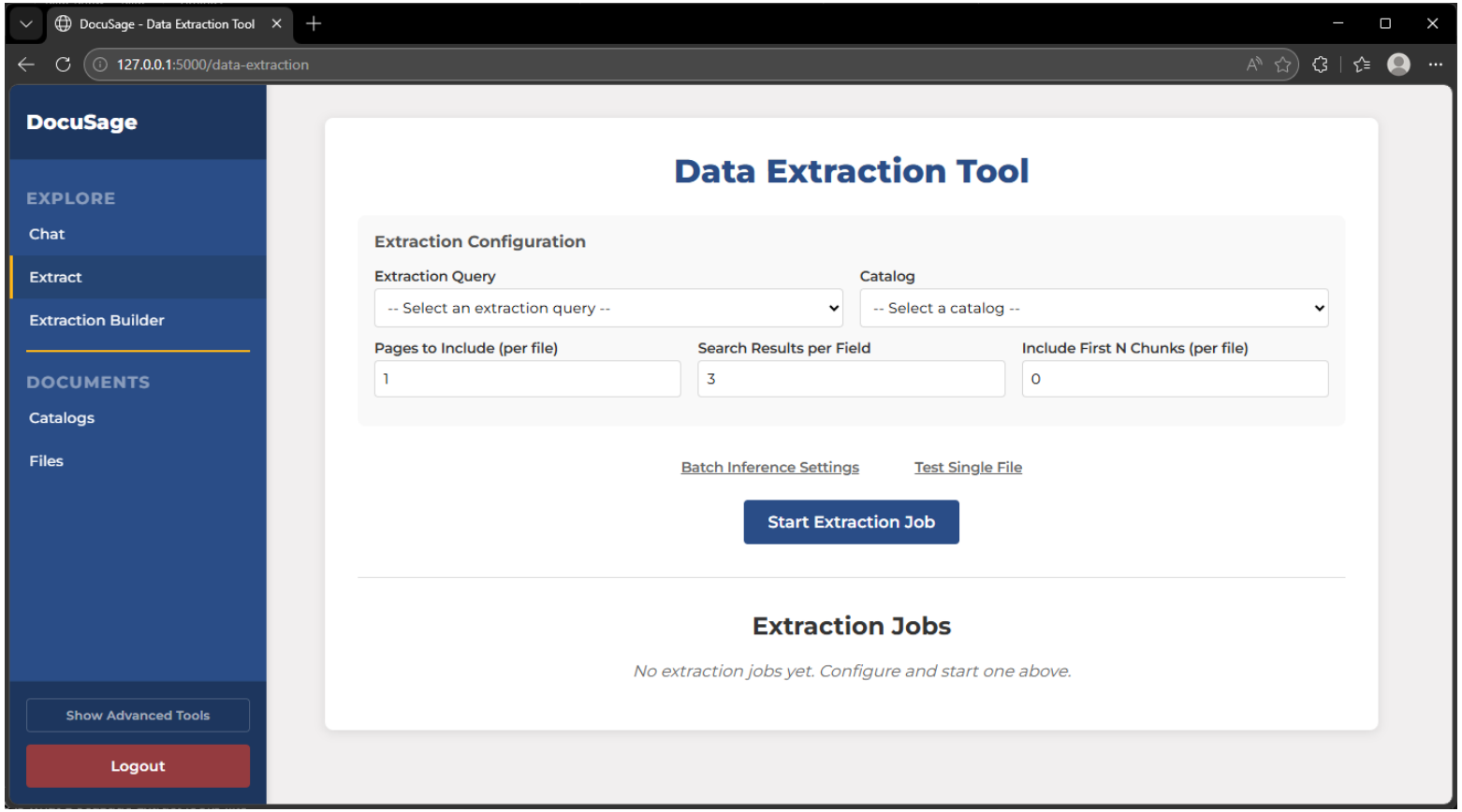
They wanted a tool where they could put in all of their documents and get out a spreadsheet full of data about each document.

MANAGEMENT
PERFORMANCEHUB

DocuSage Extract

DocuSage Extract is the tool I built to extract data from catalogs.

It uses many of the same components as DocuSage Chat, but is focused on one-time bulk data extraction.



MANAGEMENT PERFORMANCEHUB

DocuSage

EXPLORE

Chat

Extract

Extraction Builder

DOCUMENTS

Catalogs

Files

Show Advanced Tools

Logout

Default Value

Unknown

Semantic Search (optional)

e.g., contract dates

×

+ Add Field

Save Configuration

Config Name

itp_extract

Save Config

Generated Prompt Preview

elds defined 25 lines

```
# Document Context
Below is a contract for the State of Indiana, scanned in from a physical document. Please fill out the following data structure.

# Task
Extract structured information from the provided text and format it as JSON according to the schema below.

# Output Format
Respond with ONLY valid JSON in the following structure:
```json
{
  "Contract Number": <string> // The ID of the contract; Default: Unknown,
  "Agency Name": <string> // The name of the State Agency procuring the contract; Default: Unknown,
  "Vendor Name": <string> // The name of the vendor; Default: Unknown,
  "Contract Type": <string> // The general category for the contract; Default: Unknown,
  "Effective Date": <string> // The date the contract takes effect; Default: Use the signing date,
  "Contract Value": <string> // The total cost of the contract; Default: Unknown
}
```

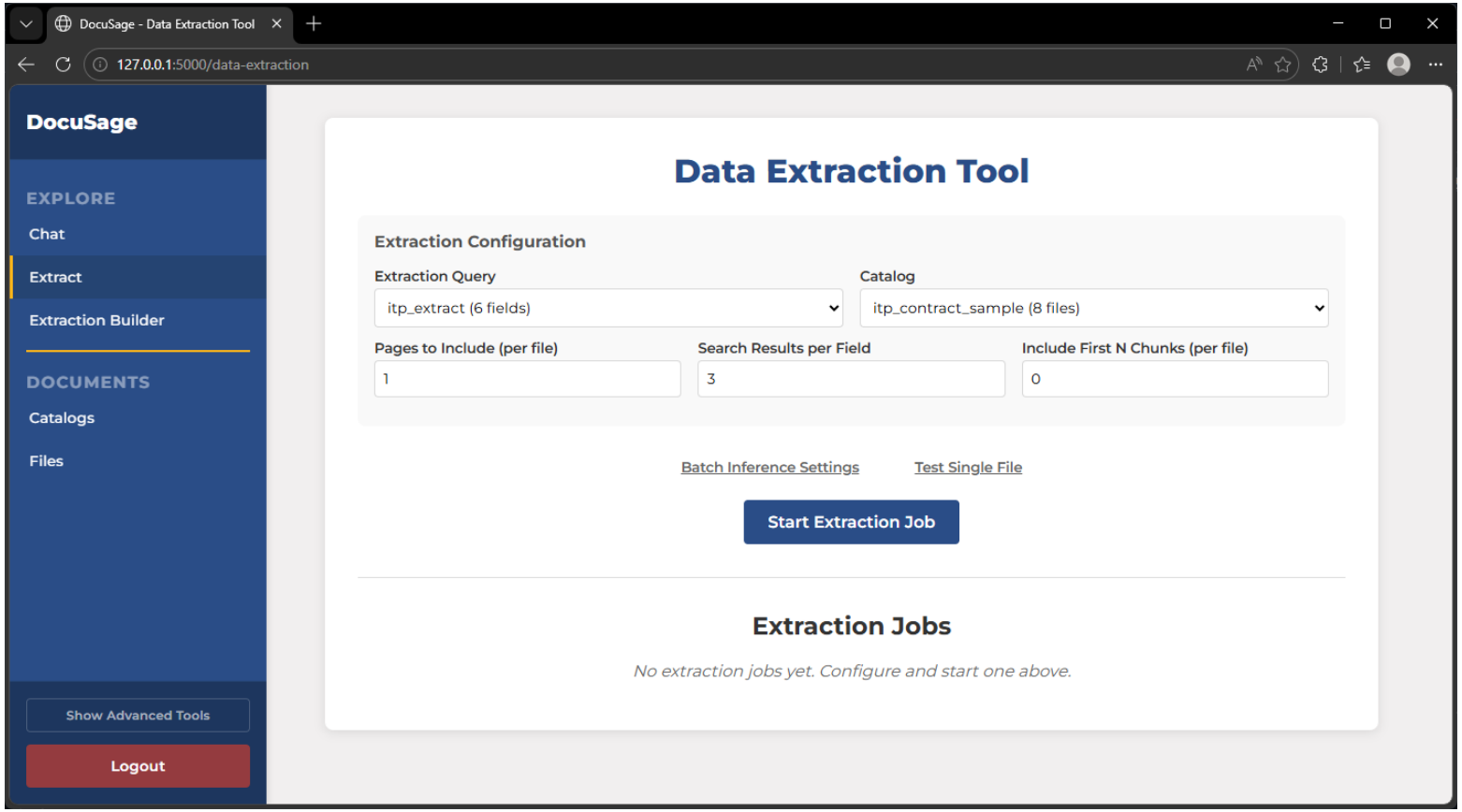
Rules:
- Include all fields in your response
- If a field's value cannot be found in the text, use the default value if specified, otherwise use an empty string
- For list fields, return an array of strings (can be empty if no values found)
- Do not include any explanatory text before or after the JSON
- Ensure the JSON is valid and parseable
```

Copy to Clipboard

DocuSage Extract

Starting the extract process is as easy as selecting the **catalog**, selecting the **extraction query**, and pressing **start**

There is also the option to test the query on a single file.



MANAGEMENT PERFORMANCEHUB

DocuSage Extract Results

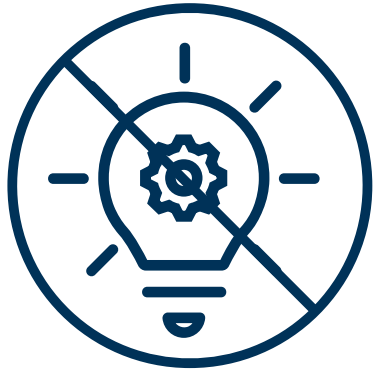
	A	B	C	D	E	F	G
1	File Name	Contract Number	Agency Name	Vendor Name	Contract Type	Effective Date	Contract Value
2	47382910.pdf	47382910	Indiana Department of Transportation	Meridian Infrastructure Group LLC	Construction	1/15/2026	$2,450,000
3	83920471.pdf	83920471	Indiana Department of Health	ClearPath Solutions Inc	Services	3/1/2026	$187,500
4	29104857.pdf	29104857	Indiana Department of Education	Apex Technology Partners	IT	7/1/2025	$94,200
5	61738290.pdf	61738290	Indiana Bureau of Motor Vehicles	Lakeside Printing & Fulfillment Co	Goods	2/14/2026	$43,750
6	54029163.pdf	54029163	Indiana Department of Correction	Summit Facility Services LLC	Services	5/1/2025	$312,000

DocuSage Extract Key Differences

Segmentation

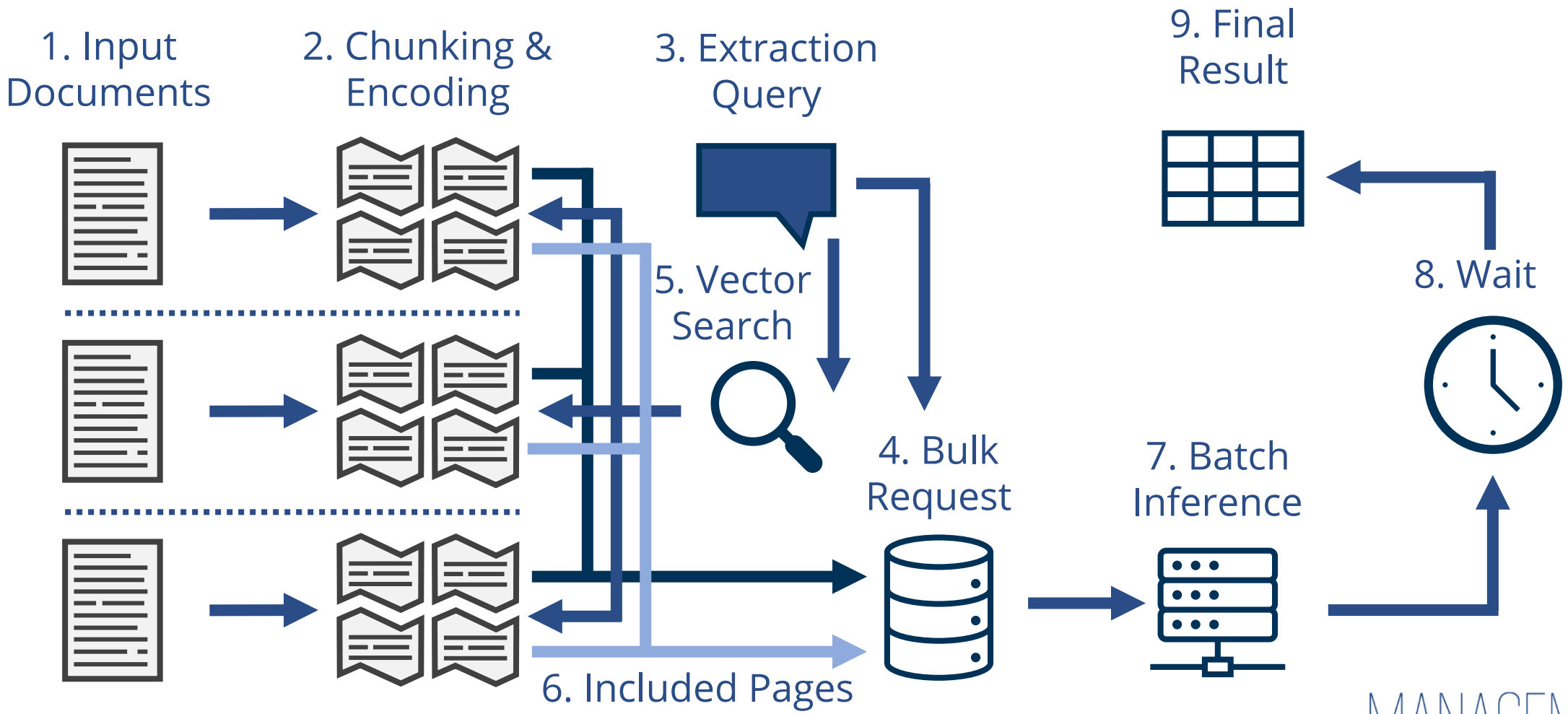
Batch Inference

No Agentic Search



MANAGEMENT PERFORMANCEHUB

Extract System Overview



In Conclusion

- When thinking about problems you want AI to solve, take the time to think about which components need AI, and which can be optimized without using LLMs
- DocuSage is designed to be precise about where answers come from and smart about what it costs to find them
- **LLMs are powerful but have real limitations, and the answer isn't to avoid them. It's to build smart structure around them**

MANAGEMENT
PERFORMANCEHUB

THANK YOU!

MANAGEMENT
PERFORMANCEHUB

Quick Disclaimer

DocuSage is an internal application created specifically for MPH-defined use cases. It is intended for testing and exploration rather than full operational rollout, and it is not available to other agencies at this time.

Our leadership team is actively collaborating with State of Indiana leadership, across all agencies, to explore opportunities for enterprise AI solutions in the upcoming budget cycle!

If you have questions or would like more information, please reach out to ***info@mph.in.gov***

Resources

- State of Indiana AI Policy: https://www.in.gov/mph/cdo/files/State-of-Indiana-Artificial-Intelligence-Policy.pdf
- State of Indiana Artificial Intelligence Policy & Guidance Webpage: https://in.gov/mph/ai
- MPH/IOT Dos & Don’ts For Using AI At Work: https://www.in.gov/mph/files/MPH-IOT-Dos-Donts.pdf
- IARA Artificial Intelligence & Records Management Guide https://www.in.gov/iara/files/Artificial-Intelligence-and-Records-Management.pdf

If you have questions or would like more information, please reach out to ***info@mph.in.gov***