WOR102

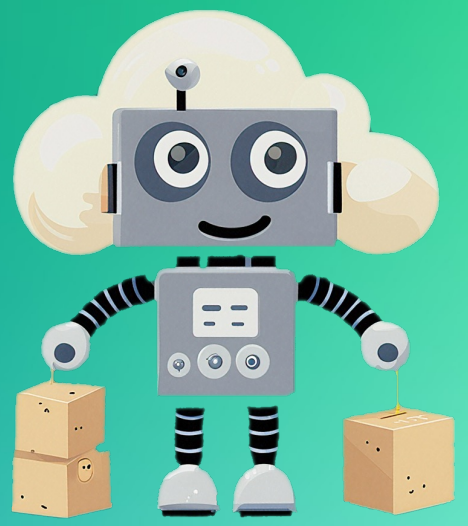
Extracting Information From Complex Documents with Amazon Bedrock

Nikita Kozodoi, PhD

(he/him)
Sr Applied Scientist
Amazon Web Services

Aiham Taleb, PhD

(he/him)
Applied Scientist
Amazon Web Services



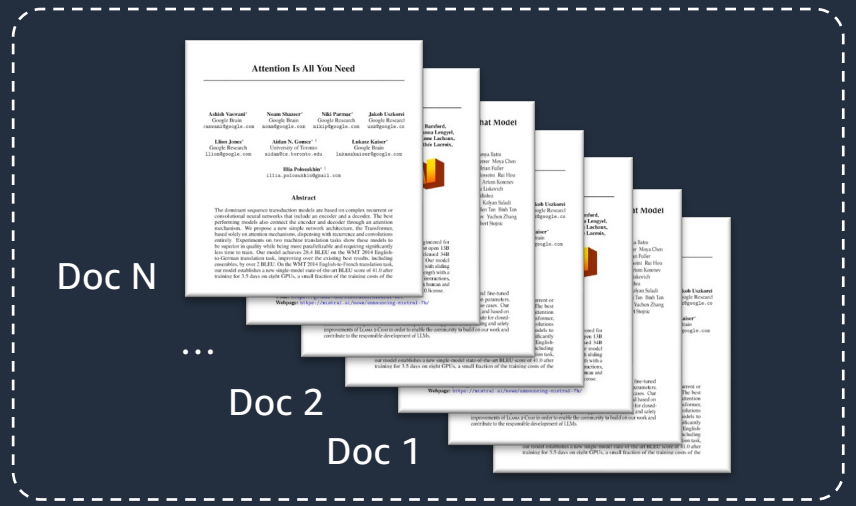
Overview

Recurring Business Task

- 🎯 Transform custom documents into a structured database
  - Extracting entities: e.g., names, dates, IDs, etc.
  - Assigning scores: e.g., sentiment, readability, urgency, etc.
  - Adding free-form content: e.g., summary, suggested response, etc.

- 👥 Example use cases:
  - Extract information from financial news articles (e.g., sentiment, summary, date)
  - Get entities from customer emails (e.g., issue type, urgency score, suggested reply)
  - Extract medical & nutrition entities from research papers (e.g., chemical elements, drug names)

Suggested Approach



PDF Documents

Name	Description
author	Author name
sentiment	Sentiment score between 0 and 1
…	…
summary	Document summary

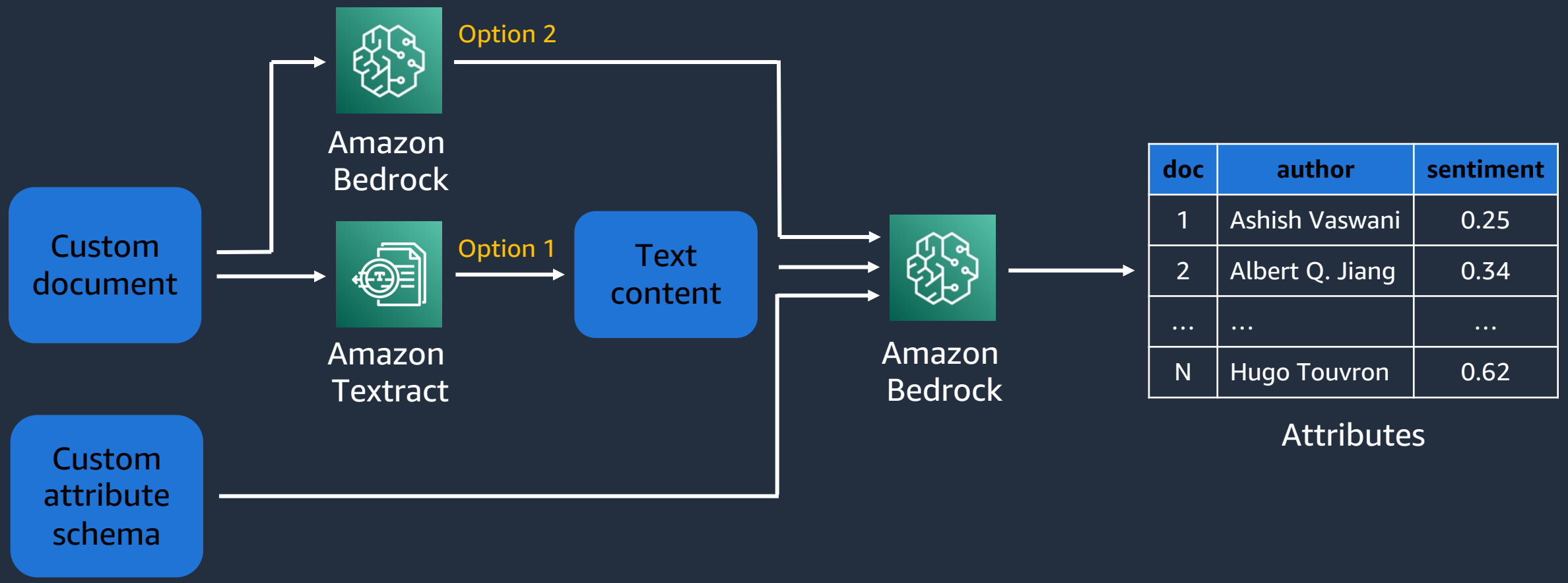
Attributes Schema

Extract Information with Amazon Bedrock

doc	author	sentiment	…	summary
1	Ashish Vaswani	0.25	…	The text desc…
2	Albert Q. Jiang	0.34	…	This docume…
…	…	…	…	…
N	Hugo Touvron	0.62	…	The paper int…

Generated Table

High-Level Architecture



doc	author	sentiment
1	Ashish Vaswani	0.25
2	Albert Q. Jiang	0.34
…	…	…
N	Hugo Touvron	0.62

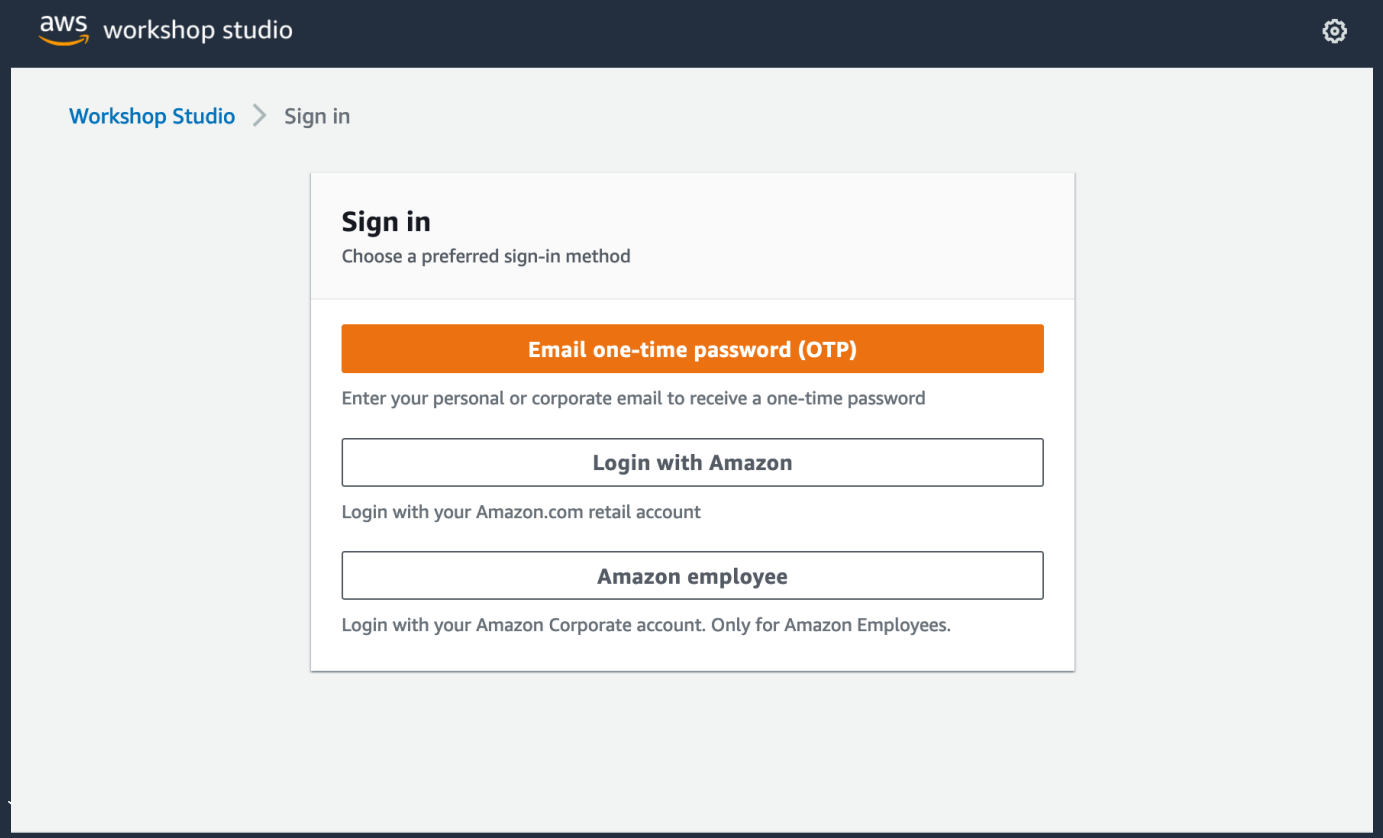
Attributes

Getting Started

1. Join the Workshop Session

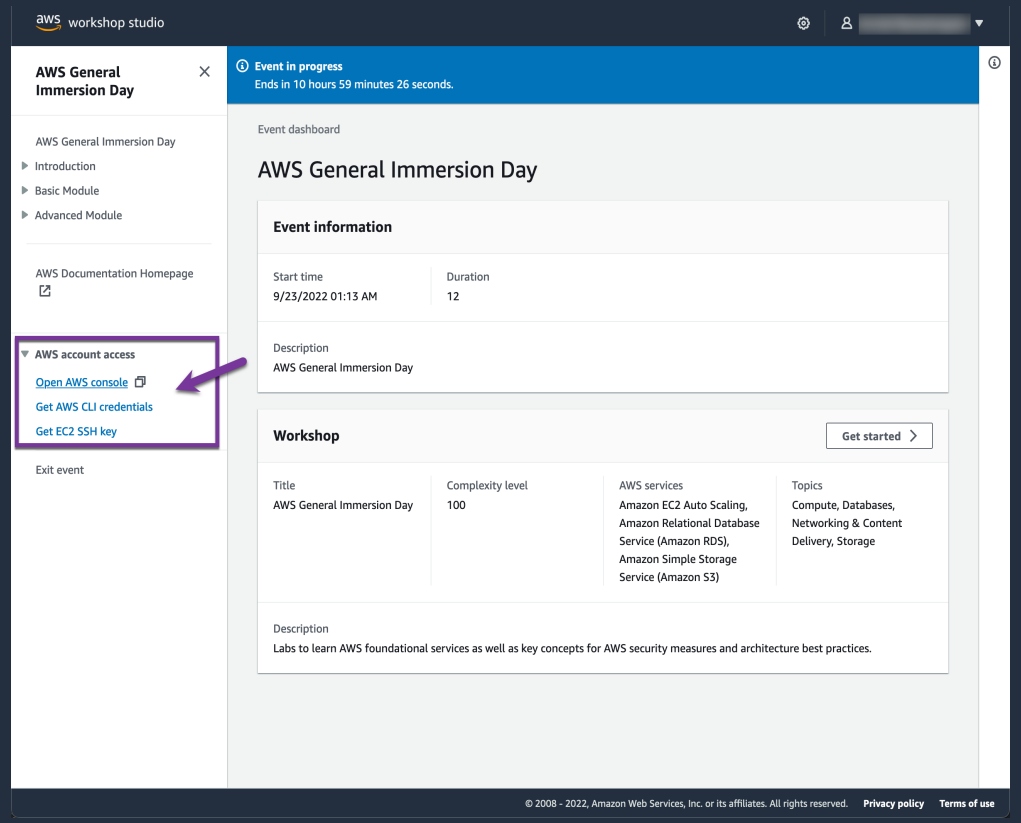
Open this link on your laptop and log in with the one-time-password

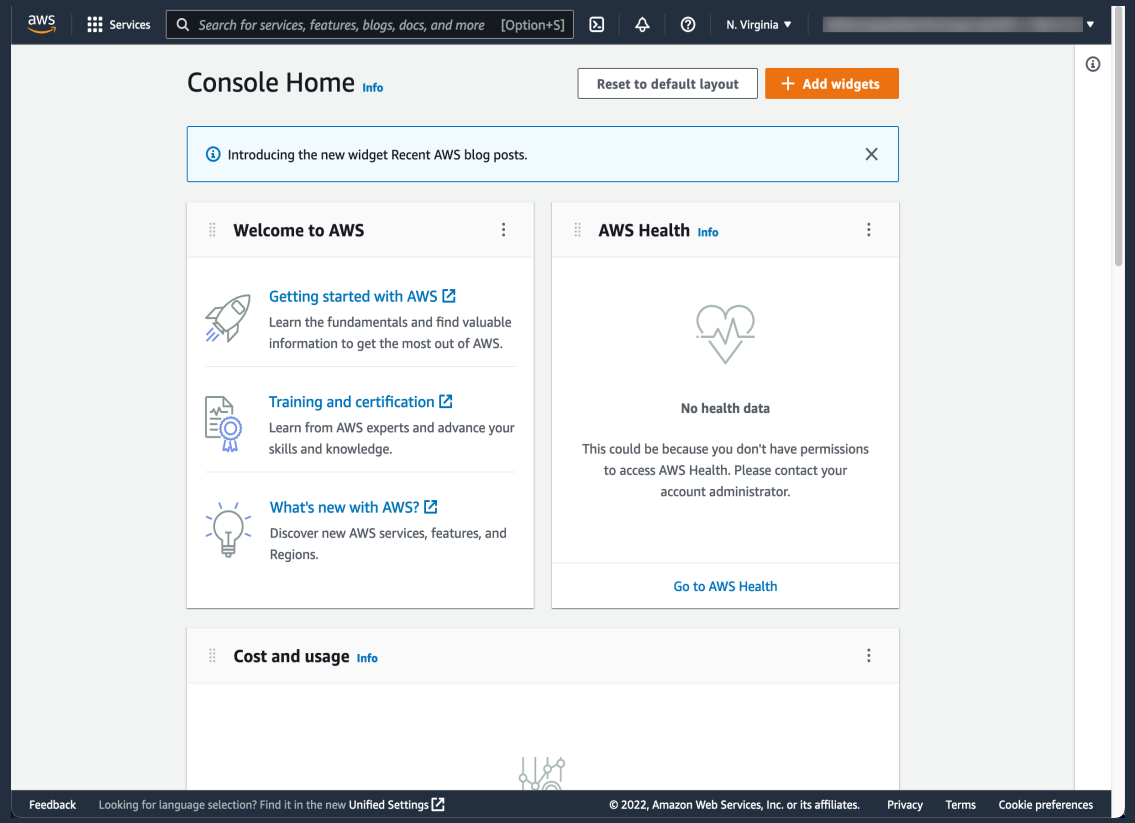
https://bit.ly/3Nyd3a4



2. Open the Workshop Environment

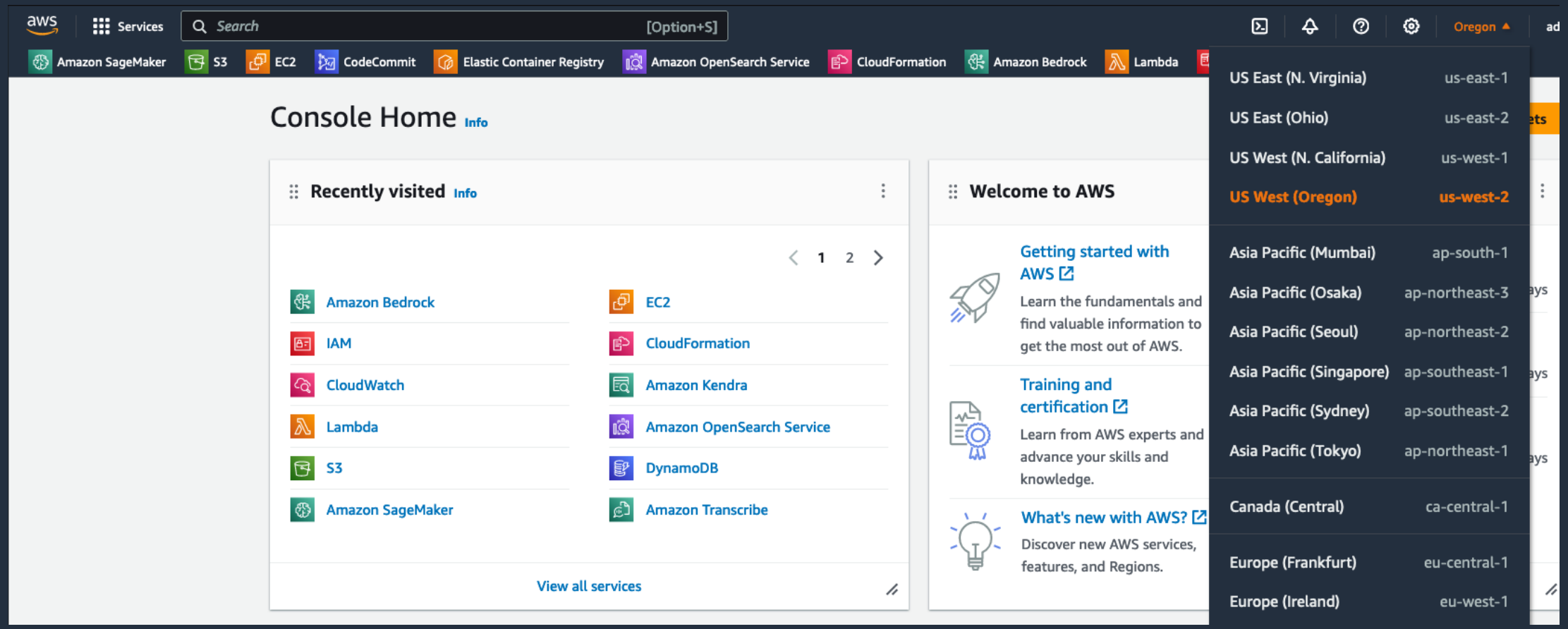
Open the AWS console from the workshop page (left sidebar)





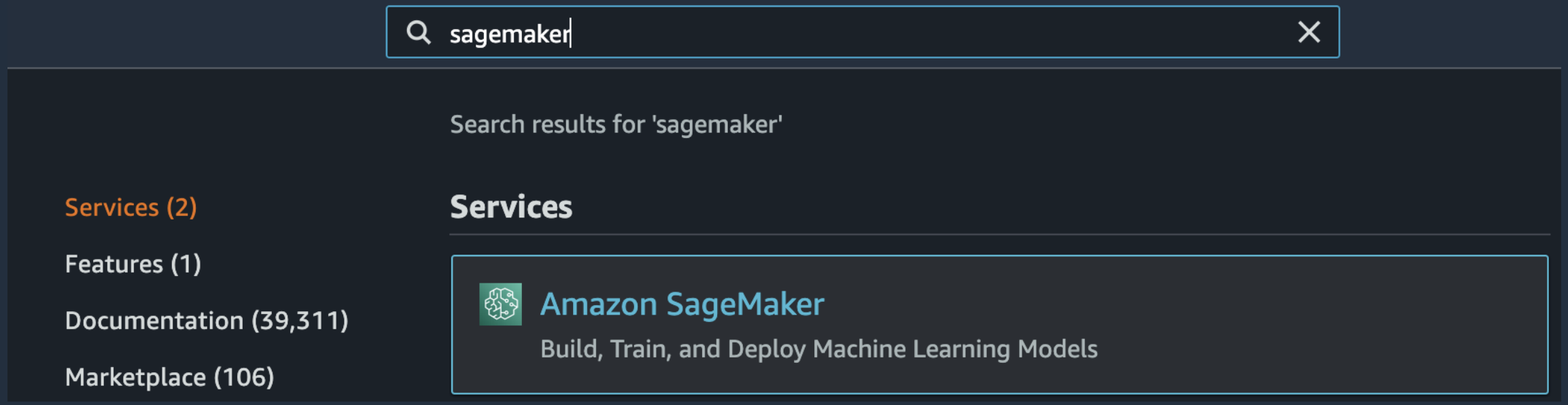
2. Open the Workshop Environment

Open the AWS console and switch to the ***us-west-2*** region if you are not already there



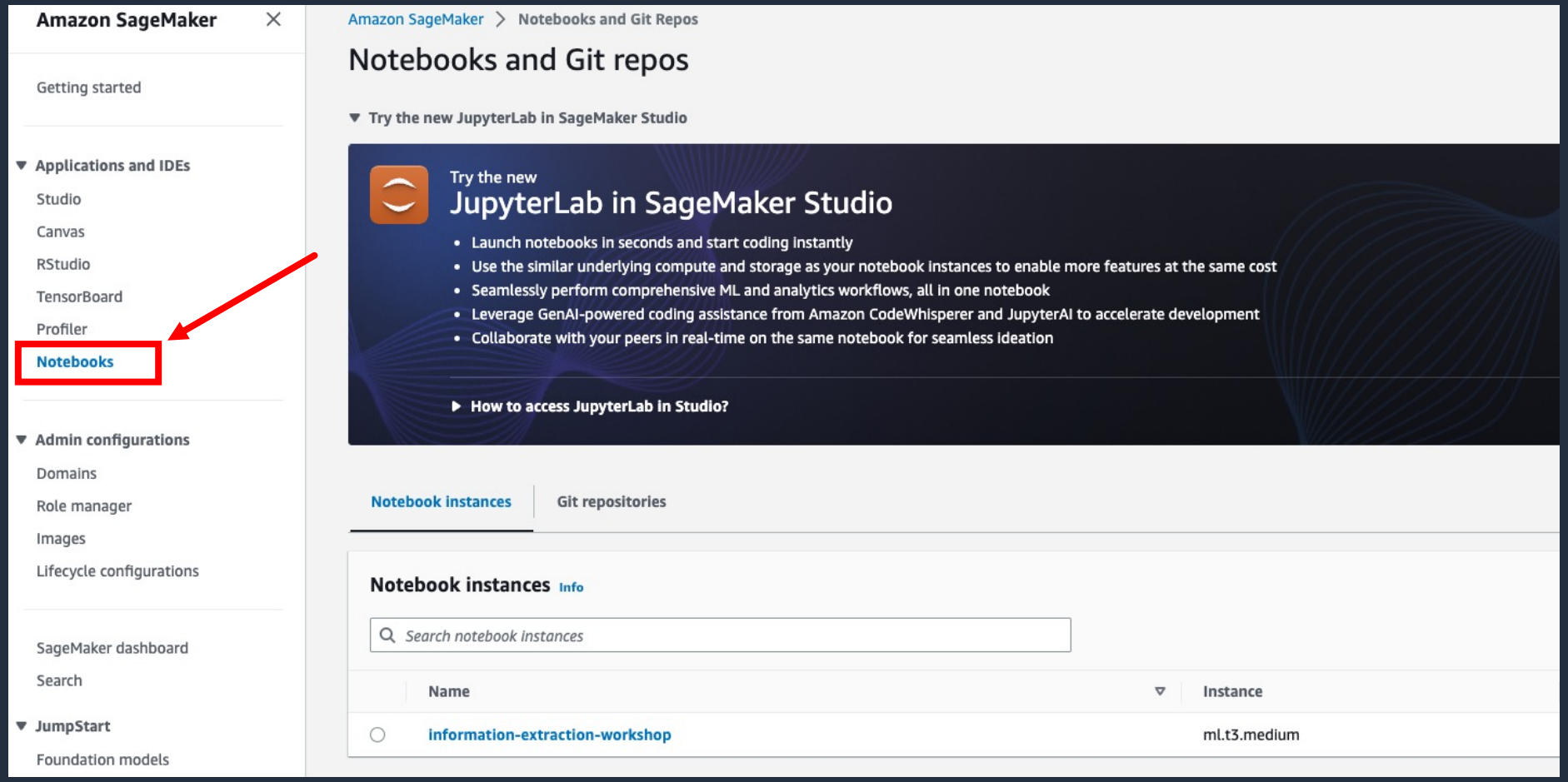
2. Open the Workshop Environment

Under ***services***, search for ***Amazon SageMaker***



2. Open the Workshop Environment

In the side bar, find the ***Applications and IDEs*** section, and click on ***Notebooks***

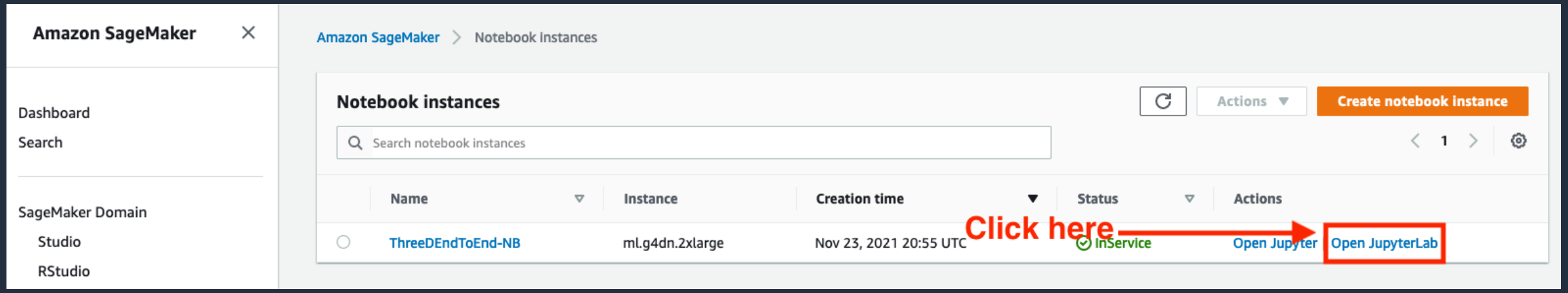


2. Open the Workshop Environment

A Notebook instance should already be provisioned

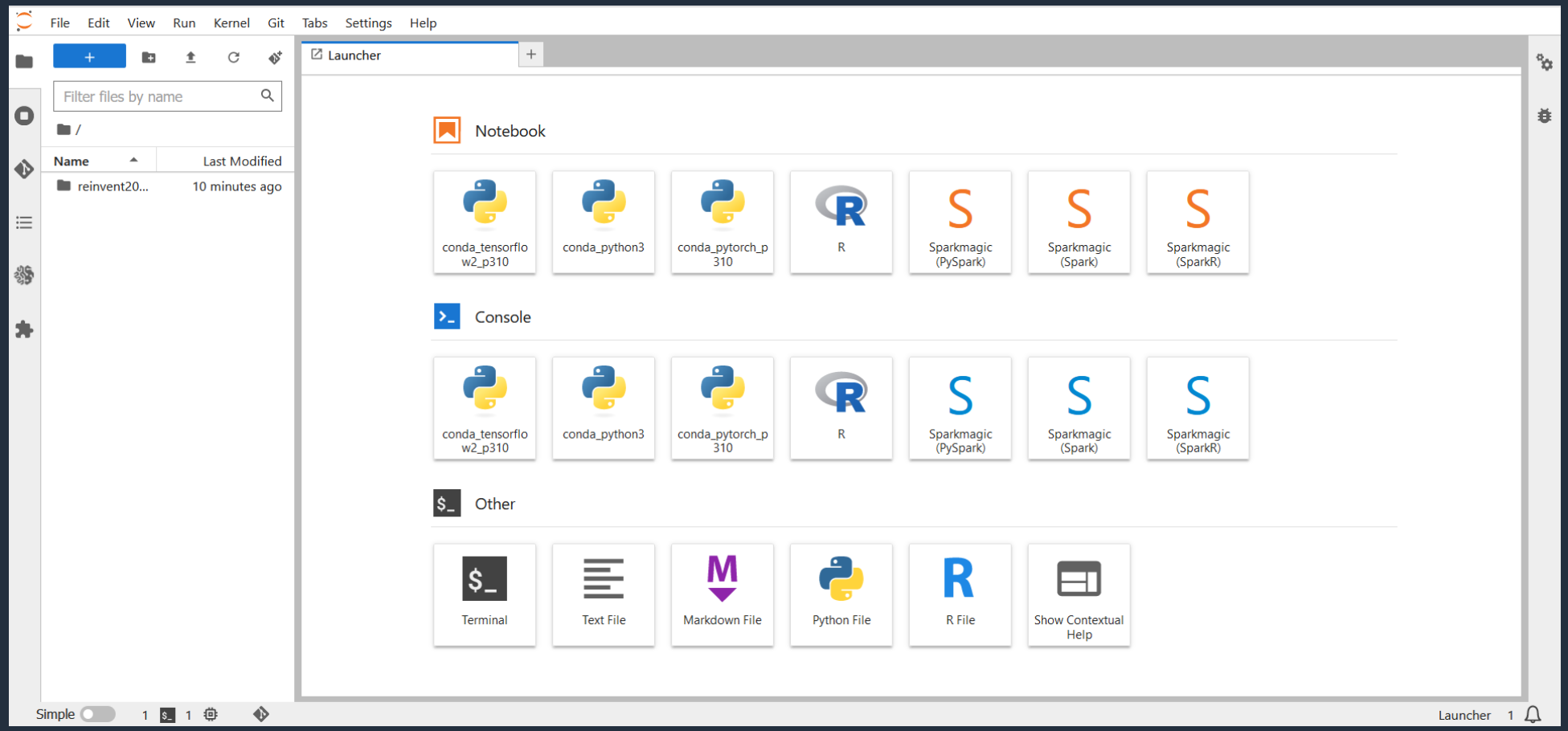
If the notebook status is “Stopped”, click **Actions** => **Start**

Click on ***Open JupyterLab*** (on the right side of the provisioned Notebook instance under ***Actions***)



2. Open the Workshop Environment

You will be redirected to a new browser tab that looks like this:



3. Work Through the Lab

Start working through the notebooks one by one (together with us or at your own pace)

You will find three notebooks:

- 00_prerequisites.ipynb: contains workshop prerequisites **[~5-10 min]**
- 01_process_pdf.ipynb: processes PDF documents and saves processed files **[~30 min]**
- 02_extract_info.ipynb: extracts custom information from the processed files **[~30 min]**

Hands-On Session

Go through *00_prerequisites.ipynb* [~10 min]

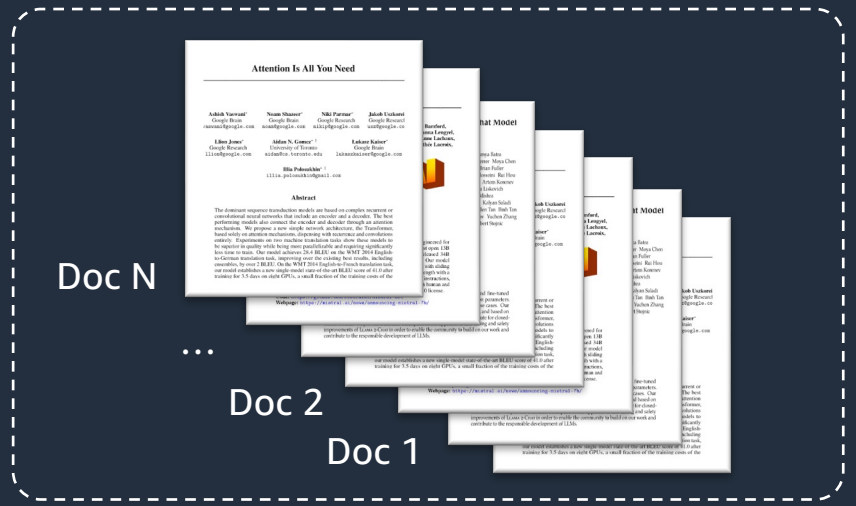
Hands-On Session

Work on *01_process_pdf.ipynb* [~30 min]

Hands-On Session

Work on *02_extract_attributes.ipynb* [~30 min]

Solution Recap



PDF Documents

Name	Description
author	Author name
sentiment	Sentiment score between 0 and 1
...	...
summary	Document summary

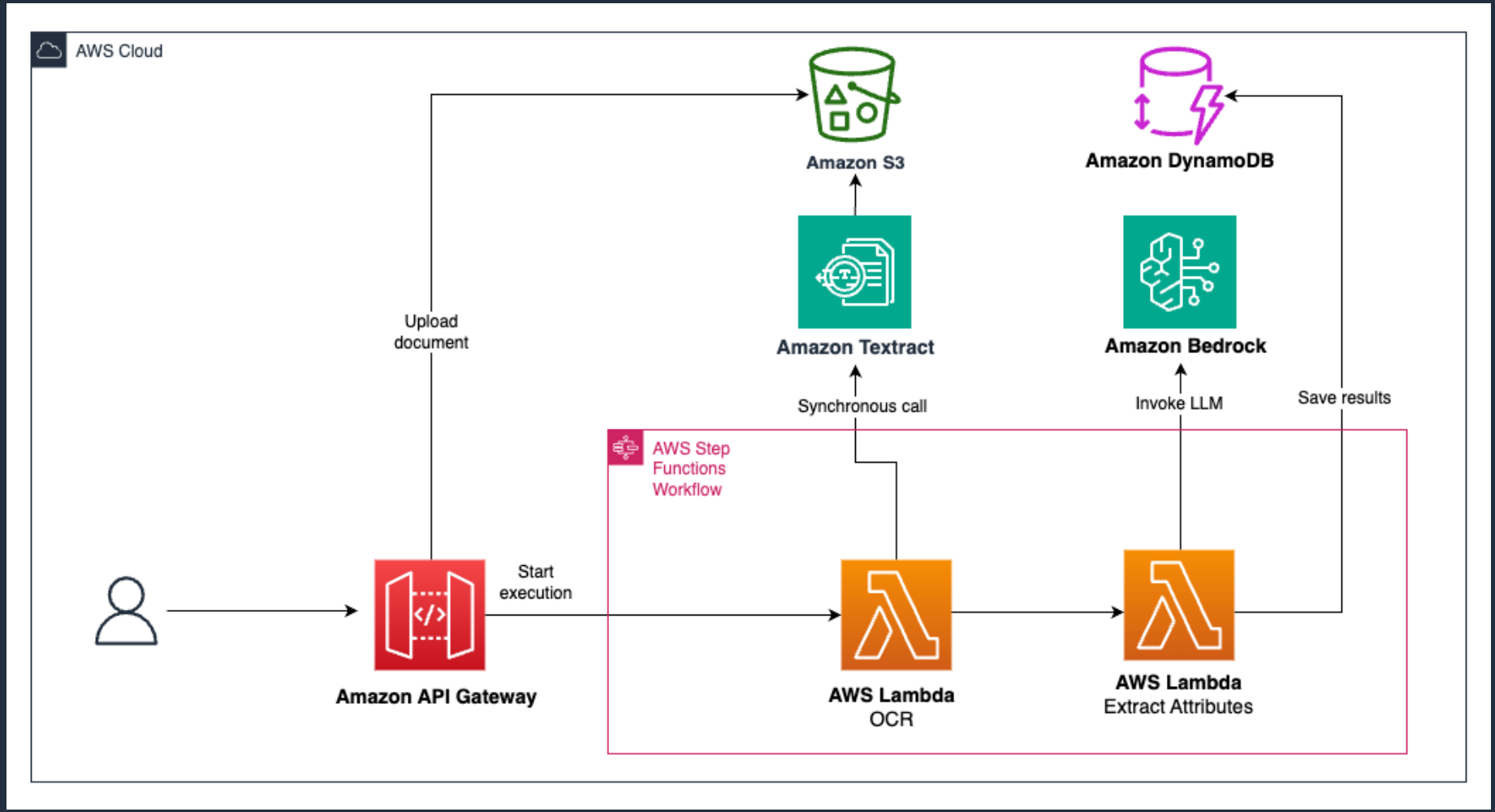
Attributes Schema

Extract Information with Amazon Bedrock

doc	author	sentiment	...	summary
1	Ashish Vaswani	0.25	...	The text desc...
2	Albert Q. Jiang	0.34	...	This docume...
...	...	...	...	...
N	Hugo Touvron	0.62	...	The paper int...

Generated Table

Scaling Up



Thank you!

Please complete the session survey.

Nikita Kozodoi, PhD

(he/him)
Sr Applied Scientist
Amazon Web Services

Aiham Taleb, PhD

(he/him)
Applied Scientist
Amazon Web Services

Appendix

LLM Prompt Template

```
Human: You are an expert AI assistant who can extract information from texts.
Read the document below and extract the entities or assign scores listed below in <entities></entities> tags.
The answer must contain the extracted entities and scores in JSON format. Do not include any other
information in the answer.
If the entity has multiple values, provide them as a list in this format: ['value1', 'value2', 'value3'].

Here is an example:
<example>
...
</example>

Document:
<document>
{document}
</document>

Entities and scores to be extracted:
<entities>
{entities}
</entities>


Assistant:
```

Few-shot examples

Input document(s)

User-provided entities in this format:

- {name_1}: {description_1} (must be {type_1})
- {name_2}: {description_2} (must be {type_2})
- ...

Extracting Different Information Types

- 👥 Explicit well-defined entities:
  - Document title
  - Product price
- 📊 Implicit entities and scores:
  - Sentiment score
  - Priority
- 🚀 Complex generated content:
  - Document summary
  - Translation

Problem Setting

- ⚙️ Existing tools like Amazon Comprehend use entity recognition models with limitations:
  - Pre-trained models only support a fixed limited set of entities
  - Using custom entities requires collecting annotated data
  - Adding or modifying an entity requires retraining the model

- 🚀 Solution: LLM-based entity recognition and scoring
  - Does not require labeled data (optional few-shot examples)
  - Supports custom set of entities without retraining
  - Goes beyond NER: also allows assigning numeric scores (e.g. sentiment)
  - Research shows LLAMA-based NER matches performance of fine-tuned BERT (Goel et al. 2023)