

# MLOps Lab2 Assignment

## Task Summary

Create main component of your system from the assignment 1.

1. For this assignment you should create a baseline model for your system, create an API wrap around it (FastAPI, Seldon, BentoML, etc. but you can use other frameworks as well), put it in the docker container and make a demo how your API works.
2. Train a baseline model. You should organise your data for training in DB or object storage (this may be S3, minio or other storage). Create an Airflow job for putting new data to your storage (training dataset).

**Note:** all artefacts for lab results demo are located in the `MLOps/reranker_pipeline_demo` folder

## Baseline Model

The baseline model is a sentence-level CrossEncoder based on the pre-trained `cross-encoder/ms-marco-MiniLM-L-6-v2` checkpoint. Its main task is pairwise relevance ranking — given a user query and a candidate document (or sentence), it estimates the relevance score.

In this project, the CrossEncoder is fine-tuned on custom training data consisting of `(query, document, label)` triples to adapt it to the domain-specific reranking task.

The trained model is wrapped into a **FastAPI service**, containerized with Docker, and deployed as a microservice. The system is integrated with **MinIO** for storing training data and uses **Airflow** to automate the ingestion of new data into the object store.

For the sake of convenience we will move all the files for the demo to the separated folder and create the next structure:

## Project Description and Component Overview

""

```
reranker_pipeline_demo/
    └── clients/
        ├── config.py
        ├── openai_client_wrapper.py
        └── qdrant_client_wrapper.py
```

```
|   └── supabase_client_wrapper.py
|── dags/
|   └── data_pipeline_dag.py
|── logs/
|── models/
|   └── model_registry.py
|── plugins/
|── prompts/
|   ├── answer_prompt.txt
|   ├── fallback_answer_prompt.txt
|   ├── generate_queries.txt
|   └── router_prompt.txt
|── demo.py
|── docker-compose.yml
|── Dockerfile
|── Dockerfile.airflow
|── generate_training_data.py
|── load_reranker_data.py
|── parse_raw_batch_data.py
|── README.md
|── requirements.txt
|── reranker_fast_api_app.py
|── train_reranker.py
└── utils.py
```

This project implements a reranking pipeline for a RAG-based system. Its main objective is to train and deploy a reranker model that scores the relevance of document-query pairs and serves it via an API for inference tasks.

## Core Components

- **Baseline Model:**

The reranker is based on the pretrained model `cross-encoder/ms-marco-MiniLM-L-6-v2` from HuggingFace. It is fine-tuned on a custom dataset for binary relevance classification (relevant vs. irrelevant documents for a given query).

- **Training Pipeline:**

- `generate_training_data.py` , `parse_raw_batch_data.py` ,  
`load_reranker_data.py` : responsible for building the training dataset from external sources and preparing it for training.
- `train_reranker.py` : performs fine-tuning of the CrossEncoder model using PyTorch/Transformers.

- **Model Serving:**

- `reranker_fast_api_app.py` : exposes the trained reranker model through a FastAPI service.
  - `Dockerfile` : builds a production-ready container for model serving.
  - `docker-compose.yml` : orchestrates the deployment of services including FastAPI, Airflow, MinIO, and PostgreSQL.
- **Airflow Integration:**
    - `dags/data_pipeline_dag.py` : defines a DAG that automates the retraining pipeline, loading of new data, and pushing it to object storage (MinIO).
    - `Dockerfile.airflow` : builds a production-ready container for data scripts orchestration.
  - **Supporting Modules:**
    - `clients/` : wrappers for interacting with OpenAI, Supabase, Qdrant, etc.
    - `prompts/` : prompt templates for generating queries or responses as part of the RAG workflow.
    - `model_registry.py` : utility for saving/loading trained model versions.

## System Deployment

Navigate to the root folder of the reranker pipeline project

```
cd PycharmProjects/rag-news-assistant/reranker_pipeline_demo
```

Run a one-time initialization of the Airflow database (creates metadata DB and admin user)

```
docker compose up airflow-init
```

```
[+] Running 4/4
 ✓ Network reranker_pipeline_demo_default          Created      0.1s
 ✓ Volume "reranker_pipeline_demo_postgres_data"   Created      0.0s
 ✓ Container reranker_pipeline_demo-postgres-1     Created      0.1s
 ✓ Container reranker_pipeline_demo-airflow-init-1  Created      0.1s

Attaching to airflow-init-1
airflow-init-1  | /home/airflow/.local/lib/python3.8/site-packages/airflow/cli/commands/db_command.py:43 DeprecationWarning: 'db init' is deprecated. Use 'db migrate' instead to migrate the db and/or airflow conne
ctor. airflow create-default-connections to create the default connections
airflow-init-1  | DB: postgres://airflow:@32.156.587.0:8000 (migration.py:213) INFO - Context impl PostgresImpl.
airflow-init-1  | [2025-07-11T02:32:56.588+0000] (migration.py:216) INFO - Will assume transactional DDL.
airflow-init-1  | INFO [alembic.runtime.migration] Context impl PostgresImpl.
airflow-init-1  | INFO [alembic.runtime.migration] Will assume transactional DDL.
airflow-init-1  | INFO [alembic.runtime.migration] Running stamp_revision -> 405de8318b3a
airflow-init-1  | WARN [airflow.models.crypto] empty cryptography key - values will not be stored encrypted.
airflow-init-1  | Initialization done
```

Start all services (MinIO, Reranker API, Airflow, PostgreSQL) in detached/background mode

```
docker compose up -d
```

```
[+] Building 2.1s (14/14) FINISHED docker:desktop-linux
=> [reranker-api internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 562B 0.0s
=> [reranker-api internal] load metadata for docker.io/library/python:3. 1.6s
=> [reranker-api auth] library/python:pull token for registry-1.docker.i 0.0s
=> [reranker-api internal] load .dockerignore 0.0s
=> => transferring context: 2B 0.0s
```

[+] Running 7/7

✓ Volume "reranker_pipeline_demo_minio_data"	Created	0.0s
✓ Container reranker_pipeline_demo-minio-1	Started	0.6s
✓ Container reranker_pipeline_demo-reranker-api-1	Started	0.6s
✓ Container reranker_pipeline_demo-postgres-1	Healthy	1.4s
✓ Container reranker_pipeline_demo-airflow-init-1	Exited	6.4s
✓ Container reranker_pipeline_demo-airflow-scheduler-1	Started	0.6s
✓ Container reranker_pipeline_demo-airflow-webserver-1	Started	6.6s

List all running containers, their statuses, ports, and resource usage

`docker compose ps`

NAME	IMAGE	COMMAND	SERVICE	CREATED	STATUS	PORTS
reranker_pipeline_demo-airflow-scheduler-1	apache/airflow:2.7.3	"/usr/bin/dumb-init -- airflow-scheduler	airflow-scheduler	36 seconds ago	Up 35 seconds	8080/tcp
reranker_pipeline_demo-airflow-webserver-1	apache/airflow:2.7.3	"/usr/bin/dumb-init -- airflow-webserver	airflow-webserver	36 seconds ago	Up 29 seconds (health: starting)	0.0.0.0:8080->8080/tcp
reranker_pipeline_demo-minio-1	minio/minio:latest	"/usr/bin/docker-entrypoint.s...	minio	36 seconds ago	Up 35 seconds	0.0.0.0:9000->9000-9001/tcp
reranker_pipeline_demo-postgres-1	postgres:13	=	postgres	About a minute ago	Up About a minute (healthy)	5432/tcp

Provides real-time metrics for all running containers

`docker stats`

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
b90d4425cef0	reranker_pipeline_demo-minio-1	0.00%	217.9MiB / 7.654GiB	2.78%	12.4kB / 4.19kB	38.7MB / 41kB	16
a59f4315353b	reranker_pipeline_demo-reranker-api-1	0.38%	544.2MiB / 7.654GiB	6.94%	93.7MB / 1.85MB	2.55MB / 269MB	40
59545d336792	reranker_pipeline_demo-postgres-1	1.10%	31.53MiB / 7.654GiB	0.40%	1.53MB / 2.08MB	3.58MB / 1.59MB	12
57949c87472a	reranker_pipeline_demo-airflow-scheduler-1	2.56%	401MiB / 7.654GiB	5.12%	780kB / 941kB	3.47MB / 39.6MB	73
c27c3a0bd25d	reranker_pipeline_demo-airflow-webserver-1	0.11%	574.4MiB / 7.654GiB	7.33%	933kB / 396kB	0B / 103MB	9

Containers statuses can also be seen in Docker Desktop

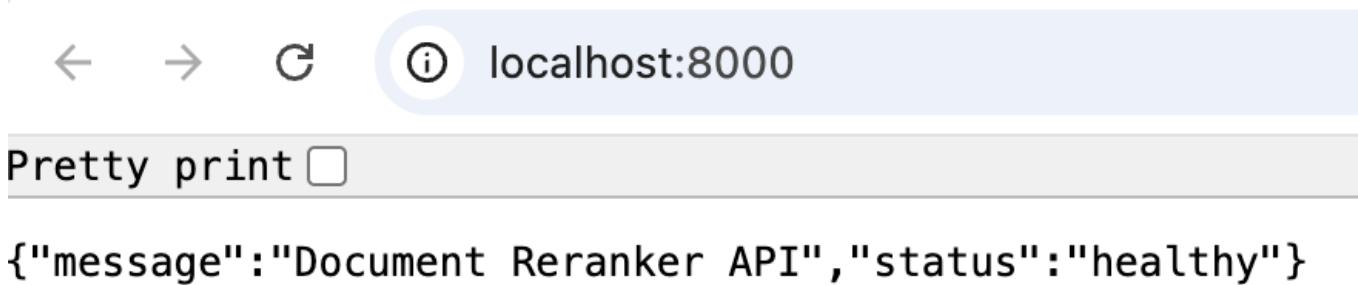
<input type="checkbox"/>	<input checked="" type="checkbox"/>	reranker_pipeline_demo	-	-	7.87%	2 minutes ago	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	postgres-1	7d556f973b2d	postgres:13	0.44%	3 minutes ago	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	airflow-init-1	16ff636d3f62	apache/airflow:2.7.3	0%	2 minutes ago	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	minio-1	1f1c44f5e2a4	minio/minio:latest	9000:9000	Show all ports (2)	0.02%	2 minutes ago	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	reranker-api-1	46114cb24681	reranker_pipeline_demo-reranker-api	8000:8000		0%	2 minutes ago	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	airflow-scheduler-1	ac90a75827c7	apache/airflow:2.7.3			7.24%	2 minutes ago	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	airflow-webserver-1	3112c02697d5	apache/airflow:2.7.3	8080:8080		0.17%	2 minutes ago	<input type="checkbox"/>

The following services are launched in separate containers:

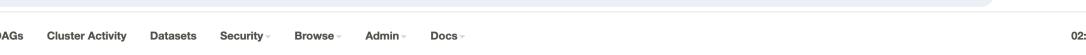
- Reranker API (FastAPI) — available at <http://localhost:8000>
- Airflow UI — available at <http://localhost:8080>, login: admin/admin
- MinIO Object Store — <http://localhost:9001>, login: minioadmin/minioadmin
- PostgreSQL — used by Airflow backend and logs

- Airflow Scheduler & Webserver — for running and monitoring DAGs

## Reranker API



Airflow UI



A screenshot of the Apache Airflow web interface. The top navigation bar includes links for Airflow, DAGs, Cluster Activity, Datasets, Security, Browse, Admin, and Docs. The top right shows the date and time as 02:37 UTC and a user icon labeled AU. The main title "DAGs" is displayed above a search bar and filter options. Below the search bar, there are buttons for "All" (1), "Active" (0), and "Paused" (1). There are also buttons for "Running" (0) and "Failed" (1). A "Filter DAGs by tag" input field and a "Search DAGs" input field are present. To the right, there is an "Auto-refresh" toggle switch and a refresh button. The main content area displays a single DAG entry for "reranker\_data\_pipeline". The table columns include: DAG (with a dropdown arrow), Owner (data-team), Runs (with a dropdown arrow), Schedule (1 day, 0:00:00), Last Run (2025-07-10, 02:37:02), Next Run (with a dropdown arrow), Recent Tasks (10 circular icons), Actions (with a play and stop button), and Links (with a three-dot menu). At the bottom, there is a navigation bar with icons for back, forward, and search, and a message indicating "Showing 1-1 of 1 DAGs".

# MinIO Object Store

The screenshot shows the MinIO Object Store Community Edition web interface. The top navigation bar includes standard browser controls (back, forward, search) and a URL bar showing "localhost:9001/browser". On the right side of the header are various system icons. The main content area is titled "Object Browser". On the left, there's a sidebar with a "Create Bucket" button. The central panel is titled "Buckets" and contains the following text: "MinIO uses buckets to organize objects. A bucket is similar to a folder or directory in a filesystem, where each bucket can hold an arbitrary number of objects." Below this text is a call-to-action button labeled "To get started, Create a Bucket!".

Just commands that have been used 100500 times to debug and restart everything

**docker compose down**

```
docker compose build reranker-api
```

```
docker compose up -d
```

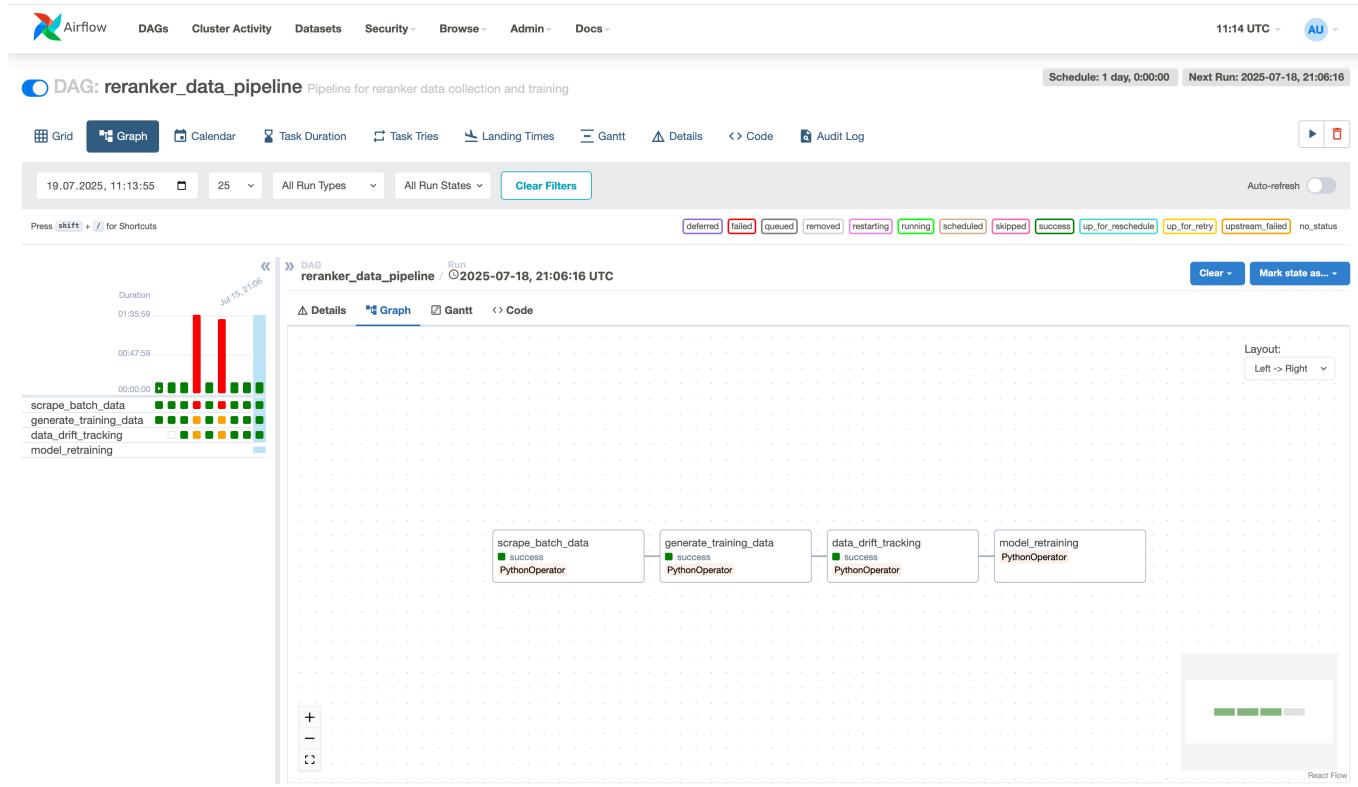
```
docker compose down -v
```

```
docker compose build --no-cache
```

```
docker compose up airflow-init
```

```
docker compose up -d
```

## Data Pipeline Logs



localhost:8080/taskinstance/list/?\_flt\_3\_dag\_id=reranker\_data\_pipeline&\_flt\_3\_state=success

	State	Dag Id	Task Id	Run Id	Map Index	Logical Date	Operator	Start Date	End Date	Duration	Note	Job Id	Hostname
	SUCCESS	reranker_data_pipeline	scrape_batch_data	manual__2025-07-11T20:39:33.631597+00:00		2025-07-11, 20:39:33	PythonOperator	2025-07-11, 20:54:45	2025-07-11, 20:55:18	33s		5	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	manual__2025-07-11T20:39:33.631597+00:00		2025-07-11, 20:39:33	PythonOperator	2025-07-11, 20:55:19	2025-07-11, 20:55:30	10s		6	9f9112c51c27
	SUCCESS	reranker_data_pipeline	scrape_batch_data	scheduled__2025-07-10T20:55:51.669566+00:00		2025-07-10, 20:55:51	PythonOperator	2025-07-11, 20:55:52	2025-07-11, 20:56:11	19s		7	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	scheduled__2025-07-10T20:55:51.669566+00:00		2025-07-10, 20:55:51	PythonOperator	2025-07-11, 20:56:13	2025-07-11, 20:56:23	9s		8	9f9112c51c27
	SUCCESS	reranker_data_pipeline	scrape_batch_data	scheduled__2025-07-11T20:55:51.669566+00:00		2025-07-11, 20:55:51	PythonOperator	2025-07-12, 20:59:50	2025-07-12, 21:00:42	51s		9	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	scheduled__2025-07-11T20:55:51.669566+00:00		2025-07-11, 20:55:51	PythonOperator	2025-07-12, 21:00:44	2025-07-12, 21:00:55	11s		10	9f9112c51c27
	SUCCESS	reranker_data_pipeline	data_drift_tracking	scheduled__2025-07-11T20:55:51.669566+00:00		2025-07-11, 20:55:51	PythonOperator	2025-07-12, 21:00:55	2025-07-12, 21:00:55	<1s		11	9f9112c51c27
	SUCCESS	reranker_data_pipeline	scrape_batch_data	scheduled__2025-07-13T21:06:16.075593+00:00		2025-07-13, 21:06:16	PythonOperator	2025-07-14, 21:06:18	2025-07-14, 21:06:57	38s		13	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	scheduled__2025-07-13T21:06:16.075593+00:00		2025-07-13, 21:06:16	PythonOperator	2025-07-14, 21:06:59	2025-07-14, 21:07:09	10s		14	9f9112c51c27
	SUCCESS	reranker_data_pipeline	data_drift_tracking	scheduled__2025-07-13T21:06:16.075593+00:00		2025-07-13, 21:06:16	PythonOperator	2025-07-14, 21:07:10	2025-07-14, 21:07:10	<1s		15	9f9112c51c27
	SUCCESS	reranker_data_pipeline	scrape_batch_data	scheduled__2025-07-15T21:06:16.075593+00:00		2025-07-15, 21:06:16	PythonOperator	2025-07-16, 21:08:22	2025-07-16, 21:13:23	5M:1s		17	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	scheduled__2025-07-15T21:06:16.075593+00:00		2025-07-15, 21:06:16	PythonOperator	2025-07-16, 21:13:25	2025-07-16, 21:13:38	12s		18	9f9112c51c27
	SUCCESS	reranker_data_pipeline	data_drift_tracking	scheduled__2025-07-15T21:06:16.075593+00:00		2025-07-15, 21:06:16	PythonOperator	2025-07-16, 21:13:40	2025-07-16, 21:13:40	<1s		19	9f9112c51c27
	SUCCESS	reranker_data_pipeline	scrape_batch_data	scheduled__2025-07-16T21:06:16.075593+00:00		2025-07-16, 21:06:16	PythonOperator	2025-07-17, 21:16:14	2025-07-17, 21:17:08	54s		20	9f9112c51c27
	SUCCESS	reranker_data_pipeline	generate_training_data	scheduled__2025-07-16T21:06:16.075593+00:00		2025-07-16, 21:06:16	PythonOperator	2025-07-17, 21:17:14	2025-07-17, 21:17:25	11s		21	9f9112c51c27

localhost:8080/dags/reranker\_data\_pipeline/grid?tab=logs&dag\_run\_id=manual\_2025-07-11T20%3A39%3A33.631597%2B00%3A00&task\_id=scrape\_batch\_data

Airflow DAGs Cluster Activity Datasets Security Browse Admin Docs 21:53 UTC AU

11.07.2025, 20:56:52 25 All Run Types All Run States Clear Filters

Press shift + / for Shortcuts deferred failed queued removed restarting running scheduled skipped success up\_for\_reschedule up\_for\_retry upstream\_failed no\_status

DAG reranker\_data\_pipeline Run 2025-07-11, 20:39:33 UTC Task scrape\_batch\_data

Details Graph Gantt Code Logs

Clear task Mark state as... Filter Tasks...

Duration (by attempts)

scrape\_batch\_data generate\_training\_data

Logs

[2025-07-11, 20:54:45 UTC] [logging.mixin.py:1297] INFO - ravcap=0.24e-1  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - uvicorn=0.24.4  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - sentence-transformers==4.1.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - streamlit==1.45.1  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - torch==2.3.1  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - transformers==4.42.4  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - datasets==2.20.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - pandas==2.2.2  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - numpy==1.26.4  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - pydantic==2.5.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - python-multipart==0.0.6  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - psutil==5.9.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - requests==2.31.1  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - supabase==2.16.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] INFO - openai==1.79.0  
[2025-07-11, 20:54:45 UTC] [logging.mixin.py:154] WARNING - /home/\*\*\*/.local/lib/python3.11/site-packages/context.py:314 AirflowContextDeprecationWarning: Accessing 'execution\_date' from the template is deprecated.  
[2025-07-11, 20:54:57 UTC] [config.py:58] INFO - PyTorch version 2.3.1 available.  
[2025-07-11, 20:55:05 UTC] [SentenceTransformer.py:211] INFO - Use pytorch device\_name: cpu  
[2025-07-11, 20:55:12 UTC] [CrossEncoder.py:224] INFO - Using SentenceTransformer: BAAI/bge-base-en-v1.5  
[2025-07-11, 20:55:12 UTC] [CrossEncoder.py:224] INFO - Use pytorch device: cpu  
[2025-07-11, 20:55:17 UTC] [logging.mixin.py:154] INFO - Scraping data from 2025-07-09 to 2025-07-11  
[2025-07-11, 20:55:17 UTC] [`_client.py:1025`] INFO - HTTP Request: GET https://e3e9y79b-283b-4aeb-8057-6f4c761d5b89.europy-west3-0.gcp.cloud.qdrant.io:6333 "HTTP/1.1 200 OK"  
[2025-07-11, 20:55:17 UTC] [parse\_raw\_batch\_data.py:276] INFO - Deleting points in qdrant for range 2025-07-09 to 2025-07-11  
[2025-07-11, 20:55:17 UTC] [`_client.py:1025`] INFO - HTTP Request: GET https://e3e9y79b-283b-4aeb-8057-6f4c761d5b89.europy-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"  
[2025-07-11, 20:55:18 UTC] [logging.mixin.py:154] INFO - Collected data does not exist.  
[2025-07-11, 20:55:18 UTC] [parse\_raw\_batch\_data.py:293] INFO - Points 0 selected for processing.  
[2025-07-11, 20:55:18 UTC] [parse\_raw\_batch\_data.py:380] INFO - Total subtuples to embed: 0  
[2025-07-11, 20:55:18 UTC] [parse\_raw\_batch\_data.py:313] INFO - Uploading 0 points to qdrant.  
[2025-07-11, 20:55:18 UTC] [logging.mixin.py:154] INFO - No points to upload.  
[2025-07-11, 20:55:18 UTC] [parse\_raw\_batch\_data.py:313] INFO - Pipeline completed.  
[2025-07-11, 20:55:18 UTC] [`python.py:194`] INFO - Done. Returned value was: Scrapped data for 2025-07-09 to 2025-07-11  
[2025-07-11, 20:55:18 UTC] [taskinstance.py:1408] INFO - Marking task as SUCCESS, dag\_id=reranker\_data\_pipeline, task\_id=scrape\_batch\_data, execution\_date=20250711T203933, start\_date=20250711T205445, end\_date=20250711T205519  
[2025-07-11, 20:55:19 UTC] [local\_task\_job\_runner.py:228] INFO - Task exited with return code 0  
[2025-07-11, 20:55:19 UTC] [taskinstance.py:2778] INFO - 1 downstream tasks scheduled from follow-on schedule check

localhost:8080/dags/reranker\_data\_pipeline/grid?tab=logs&dag\_run\_id=manual\_\_2025-07-11T20%3A39%3A33.631597%2B00%3A00&task\_id=generate\_training\_data

Airflow DAGs Cluster Activity Datasets Security Browse Admin Docs 21:53 UTC AU

11.07.2025, 20:56:52 25 All Run Types All Run States Clear Filters

Press shift + / for Shortcuts

Auto-refresh

Duration

DAG reranker\_data\_pipeline / Run 2025-07-11, 20:39:33 UTC Task generate\_training\_data

Details Graph Gantt Code Logs

(by attempts)

All Levels

scrape\_batch\_data generate\_training\_data

Deferred Failed Queued Removed Restarting Running Scheduled Skipped Success Up\_for\_reschedule Up\_for\_retry Upstream\_failed No\_Status

Clear task Mark state as... Filter Tasks

Wrap Download See More

Logs

```
[2025-07-11, 20:55:28 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"
[2025-07-11, 20:55:28 UTC] {logging_mixin.py:154} INFO - Collection does not exist.
[2025-07-11, 20:55:28 UTC] {generate_training_data.py:78} INFO - Generating queries and inserting results...
[2025-07-11, 20:55:28 UTC] {logging_mixin.py:154} INFO - 
# Running pipeline from 2025-05-14 to 2025-05-28
[2025-07-11, 20:55:28 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333 "HTTP/1.1 200 OK"
[2025-07-11, 20:55:28 UTC] {generate_training_data.py:67} INFO - Retrieving documents from Odrant for date range 2025-05-14 to 2025-05-21
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - Collection does not exist.
[2025-07-11, 20:55:29 UTC] {generate_training_data.py:78} INFO - Generating queries and inserting results...
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - 
# Running pipeline from 2025-05-29 to 2025-06-04
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333 "HTTP/1.1 200 OK"
[2025-07-11, 20:55:29 UTC] {generate_training_data.py:67} INFO - Retrieving documents from Odrant for date range 2025-05-21 to 2025-05-28
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - Collection does not exist.
[2025-07-11, 20:55:29 UTC] {generate_training_data.py:78} INFO - Generating queries and inserting results...
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - 
# Running pipeline from 2025-05-28 to 2025-06-04
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333 "HTTP/1.1 200 OK"
[2025-07-11, 20:55:29 UTC] {generate_training_data.py:67} INFO - Retrieving documents from Odrant for date range 2025-05-28 to 2025-06-04
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - Collection does not exist.
[2025-07-11, 20:55:29 UTC] {generate_training_data.py:78} INFO - Generating queries and inserting results...
[2025-07-11, 20:55:29 UTC] {logging_mixin.py:154} INFO - 
# Generating training data from 2025-07-09 to 2025-07-11
[2025-07-11, 20:55:29 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333 "HTTP/1.1 200 OK"
[2025-07-11, 20:55:30 UTC] {generate_training_data.py:67} INFO - Retrieving documents from Odrant for date range 2025-07-09 to 2025-07-11
[2025-07-11, 20:55:30 UTC] {client.py:1025} INFO - HTTP Request: GET https://e93979b-283b-4aeb-8857-614c761d5b89.eu-west3-0.gcp.cloud.qdrant.io:6333/collections/None/exists "HTTP/1.1 200 OK"
[2025-07-11, 20:55:30 UTC] {logging_mixin.py:154} INFO - Collection does not exist.
[2025-07-11, 20:55:30 UTC] {generate_training_data.py:78} INFO - Generating queries and inserting results...
[2025-07-11, 20:55:30 UTC] {logging_mixin.py:154} INFO - 
# Done. Return value: None. Generated training data for 2025-07-09 to 2025-07-11
[2025-07-11, 20:55:30 UTC] {taskinstance.py:1406} INFO - Marking task as SUCCESS. dag_id=reranker_data_pipeline, task_id=generate_training_data, execution_date=20250711T203933, start_date=20250711T205519, end_date=2025-07-11T20:55:30, duration=0.000000
[2025-07-11, 20:55:30 UTC] {local_task_job_runner.py:228} INFO - Task exited with return code 0
[2025-07-11, 20:55:30 UTC] {taskinstance.py:2778} INFO - 0 downstream tasks scheduled from follow-on schedule check
```

# Data Storages Overview

In this project, two types of storages were used to support the RAG pipeline — Qdrant and Supabase. By separating responsibilities — unstructured retrieval to Qdrant and structured storage to Supabase — the system was made more modular and easier to maintain or scale.

Qdrant was selected for storing dense vector embeddings and enabling fast semantic similarity search.

The screenshot shows the Qdrant Cloud interface. On the left, a sidebar navigation includes 'DASHBOARD', 'Clusters' (selected), 'Hybrid Cloud', 'Backups', 'ACCOUNT', 'Access Management', 'Billing', and 'Settings'. The main content area is titled 'batch\_articles' with status 'HEALTHY' and 'FREE TIER'. It features tabs for 'Overview', 'API Keys', 'Metrics', 'Logs', 'Backups', and 'Configuration'. The 'Overview' tab displays cluster details: Cluster ID e03e979b-283b-4aeb-8057-6f4c761d5b89, Version v1.14.0, Cloud Provider europe-west3, Endpoint https://e03e979b-283b-4aeb-8057-6f4c761d5b89.europe-west3-0.gcp.cloud.qdrant.io, and resource usage (1 node, 4GiB disk, 1GiB RAM, 0.5 vCPU). A 'Scale Cluster' button is available. To the right, sections for 'Access the Cluster', 'Try Sample Datasets', 'Explore Tutorials', and 'Use the API' are shown. The 'Use the API' section includes an endpoint URL and an 'Examples' button. At the bottom, a callout 'Get All Qdrant Cloud Features' encourages upgrading to a paid cluster.

This Qdrant collection stores vectorized documents along with metadata in the payload.

```
Qdrant Point
└── id : string
└── vector : list[float]
└── payload
    ├── title : string
    ├── type : string
    ├── date : string
    ├── date_int : int
    ├── url : string
    ├── text : string
    ├── image
    │   ├── image_url : string
    │   ├── image_caption : string
    │   └── image_embedding : list[float]
    └── combined_text : string
```

- Text embeddings are used for semantic search.
- Metadata fields help with filtering or advanced ranking.
- Image-related fields allow for multimodal querying or reranking.

Supabase, built on PostgreSQL, was used to store structured information such as annotated samples of training data.

The screenshot shows the Supabase Schema Visualizer interface. On the left, a sidebar navigation includes Database, Schema Visualizer (selected), Tables, Functions, Triggers, Enumerated Types, Extensions, Indexes, Publications, Replication (Coming Soon), ACCESS CONTROL (Roles, Policies), PLATFORM (Backups, Migrations, Wrappers, Webhooks), and TOOLS (Security Advisor, Performance Advisor, Query Performance). The main area displays the 'reranker\_dataset' table structure:

	reranker_dataset	
◆	<b>id</b>	int4
◆	<b>query</b>	text
◇	<b>document_id</b>	uuid
◆	<b>document</b>	text
◆	<b>relevance</b>	int4
◆	<b>source</b>	varchar
◇	<b>created_at</b>	timestamp
◇	<b>document_date</b>	date

At the bottom, there are icons for Primary key, Identity, Unique, Nullable, and Non-Nullable.

The screenshot shows the Supabase Table Editor interface for the 'reranker\_dataset' table. The table has the following columns: id (int4), query (text), document\_id (uuid), document (text), relevance (int4), source (varchar), and created\_at (timestamp). The data consists of 151 rows, each containing a unique ID, a query string, a document ID, a document text, a relevance score, a source type, and a timestamp. The first few rows are:

	query	document_id	document	relevance	source	created_at
1	AI-assisted coding for software prototype	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
2	How AI is making software development	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
3	Using AI to build software prototypes qui	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
4	Benefits of generative AI for prototype de	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
5	AI tools for rapid app deployment	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
6	How does AI improve software prototypi	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
7	AI in software development 2025 predicti	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
8	Generative AI for flash card app develop	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
9	Prototyping with AI: Andrew's insights	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
10	AI and agentic workflows for code deploy	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	1	generated	2025-06-29 20:58:34
11	History of AI since the teenage years	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
12	Flash card apps for kids	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
13	Foreign exchange rate monitoring tool	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
14	International bank account management	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
15	User review analysis software	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
16	Benefits of using Bolt and Replit Agent	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
17	Andrew holding a sparkle stick	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
18	Learning to code with AI	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
19	Developing reliable software systems with	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
20	Challenges in AI system architecture desi	622a2c57-bddc-4f21-a289-e93c32d8524b	[ARTICLE]: Happy sum(i**3 for i in range(1, 10))	0	generated	2025-06-29 20:58:34
21	AI strategies to combat climate change	1818dd84-5268-4acf-9806-fbc2603226b6	[ARTICLE]: How can AI help to fight clim	1	generated	2025-06-29 21:04:24
22	Innovation for Cool Earth Forum AI roadm	1818dd84-5268-4acf-9806-fbc2603226b6	[ARTICLE]: How can AI help to fight clim	1	generated	2025-06-29 21:04:24
23	how AI can reduce greenhouse gas emissi	1818dd84-5268-4acf-9806-fbc2603226b6	[ARTICLE]: How can AI help to fight clim	1	generated	2025-06-29 21:04:24

## Model API Demo Preparings

Check if API is actually available

```
docker compose logs reranker-api
```

```
reranker-api-1 | INFO:     Started server process [1]
reranker-api-1 | INFO:     Waiting for application startup.
reranker-api-1 | INFO: [reranker_fast_api_app] No trained model found, using base model.
Downloading config.json: 100%[=====] 385/385 [00:00<00:00, 114KB/s]
Downloaded (.).173d8f8dedc65152fae7_100%[=====] 139M/133M [00:13<00:00, 9.97MB/s]
reranker-api-1 | Some weights of BertForSequenceClassification were not initialized from the model checkpoint at microsoft/MiniLM-L12-H384-uncased and are newly initialized: ['classifier.weight', 'classifier.bias']
reranker-api-1 | You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
Downloading tokenizer_config.json: 100%[=====] 2.00/2.00 [00:00<00:00, 8.37KB/s]
Downloading vocab.txt: 232kB [00:00, 2.73MB/s]
Downloading (.).cial_tokens_map.json: 100%[=====] 112/112 [00:00<00:00, 493kB/s]
reranker-api-1 | INFO:[sentence_transformers.cross_encoder.CrossEncoder]Use pytorch device: cpu
reranker-api-1 | INFO:[reranker_fast_api_app]Model loaded successfully
reranker-api-1 | INFO: Application startup complete.
reranker-api-1 | INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
```

The screenshot shows the OpenAPI documentation for the Document Reranker API. The main page has a header "Document Reranker API 1.0.0 OAS 3.1" and a link to "/openapi.json". Below the header, there's a "default" section with three GET methods: "/Root", "/health", and "/test". The "/test" method is collapsed. The "/rerank" endpoint is expanded, showing its description "Rank documents based on query relevance". It has a "Parameters" section (empty) and a "Request body" section marked as required, with a schema example:

```
{  "query": "string",  "documents": [    "string"  ],  "top_k": 10}
```

The "Responses" section shows a single 200 status code with the description "Successful Response". There are no links listed.

## Specific point requests to check everything one more time

```
/Users/anastasiiamazur/PycharmProjects/rag-news-assistant/rag_assistant3.11/bin/python /Users/anastasiiamazur/PycharmProjects/rag-news-assistant/api_test.py
INFO:__main__:Run http://localhost:8000/test: {'status': 'success', 'test_score': 0.48815932869911194}
INFO:__main__:Run http://localhost:8000/rerank: {'results': [{'document': 'Deep learning models require large datasets', 'score': 0.48848041892051697, 'original_index': 0, 'rank': 1}, {'document': 'Neural ...']}
INFO:__main__:Run http://localhost:8000/score: {'scores': [0.4879770278930664]}

Process finished with exit code 0
```

Create virtual environment to install requests library and run demo from terminal

```
python3 -m venv venv
source venv/bin/activate
pip install requests
```

Setup realtime logs checker

```
docker-compose logs -f reranker-api
```

## Model API Demo

Run demo from venv

```
python demo.py
```

```
anastasiiamazur@bttrm-amazur-pro16 reranker_pipeline_demo % docker-compose logs -f reranker-api — 119x18
reranker-api-1 | INFO:     Started server process [1]
reranker-api-1 | INFO:     Waiting for application startup.
reranker-api-1 | INFO:reranker_fast_api_app:No trained model found, using base model
reranker-api-1 | INFO:reranker_fast_api_app:Model loaded successfully
reranker-api-1 | INFO:     Application startup complete.
reranker-api-1 | INFO:     Uvicorn running on http://0.0.0:8000 (Press CTRL+C to quit)
reranker-api-1 | INFO:     172.19.0.1:60716 - "GET /health HTTP/1.1" 200 OK
reranker-api-1 | INFO:reranker_fast_api_app:Reranking 8 documents for query: machine learning model training...
reranker-api-1 | INFO:reranker_fast_api_app:Generated 8 scores
reranker-api-1 | INFO:reranker_fast_api_app:Returning top 5 results
reranker-api-1 | INFO:     172.19.0.1:60728 - "POST /rerank HTTP/1.1" 200 OK
reranker-api-1 | INFO:     172.19.0.1:60732 - "POST /score HTTP/1.1" 200 OK
reranker-api-1 | INFO:reranker_fast_api_app:Reranking 8 documents for query: transformer architecture deep learning...
reranker-api-1 | INFO:reranker_fast_api_app:Generated 8 scores
reranker-api-1 | INFO:reranker_fast_api_app:Returning top 3 results
reranker-api-1 | INFO:     172.19.0.1:60742 - "POST /rerank HTTP/1.1" 200 OK
```



reranker-pipeline --zsh-- 119x60

Document Reranker API Demo

=====

Testing API health...

Status: 200

Response: {'status': 'healthy', 'model\_loaded': True}

Testing document reranking...

Reranking successful!

Query: machine learning model training

Top ranked documents:

1. [Score: 1.4837]  
Document: Deep learning models require large datasets for effective training and validation....2. [Score: 1.2512]  
Document: Transfer learning allows pre-trained models to be fine-tuned on specific tasks....3. [Score: -4.2058]  
Document: Cross-validation is an important technique for evaluating model performance during training....4. [Score: -5.6689]  
Document: Neural networks can be trained using supervised learning techniques with labeled data....5. [Score: -11.1419]  
Document: The weather forecast shows rain tomorrow with temperatures around 20 degrees....

Testing query-document scoring...

Scoring successful!

Query-Document relevance scores:

1. [Score: -10.2019]  
Query: artificial intelligence applications  
Document: AI is transforming various industries including healthcare, finance, and transpo...2. [Score: -11.0884]  
Query: artificial intelligence applications  
Document: The recipe for chocolate cake requires flour, sugar, eggs, and cocoa powder....3. [Score: 5.4903]  
Query: data science techniques  
Document: Statistical analysis and machine learning are core components of data science....

Real-world demo: Finding relevant AI articles...

Found 3 most relevant articles:

Search query: 'transformer architecture deep learning'

Relevant articles:

🏆 Rank 1 (Score: 4.2985)  
The Transformer architecture revolutionized natural language processing with self-attention mechanisms.🏆 Rank 2 (Score: -1.1254)  
BERT and GPT models are based on the Transformer architecture and have achieved state-of-the-art results.🏆 Rank 3 (Score: -2.4213)  
Vision Transformers adapt the Transformer architecture for computer vision tasks.

All tests completed successfully!

(venv) anastasiiamazur@bttrm-amazur-pro16 reranker-pipeline %