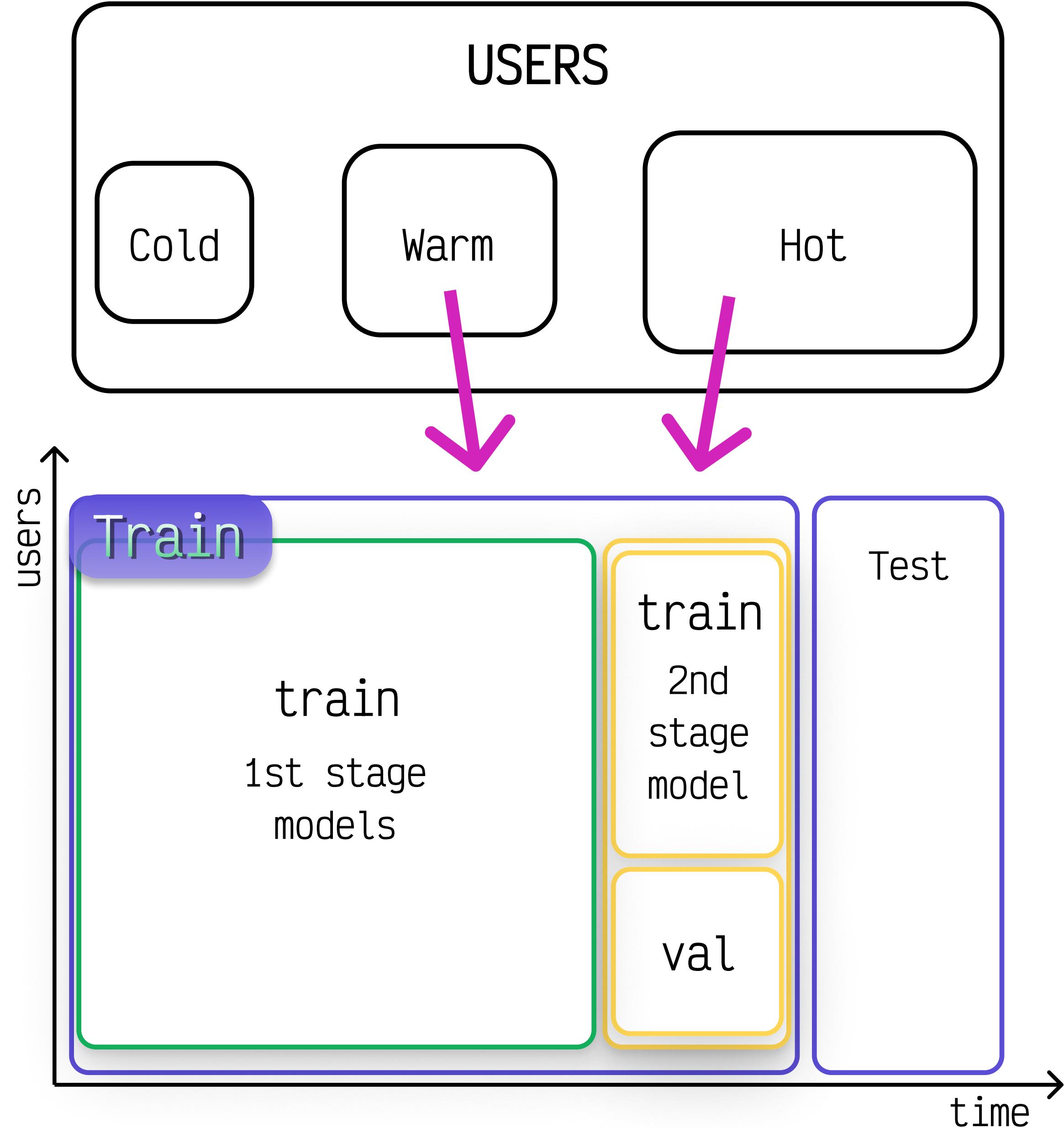


# RecSys

Stage\_6

# Валидация

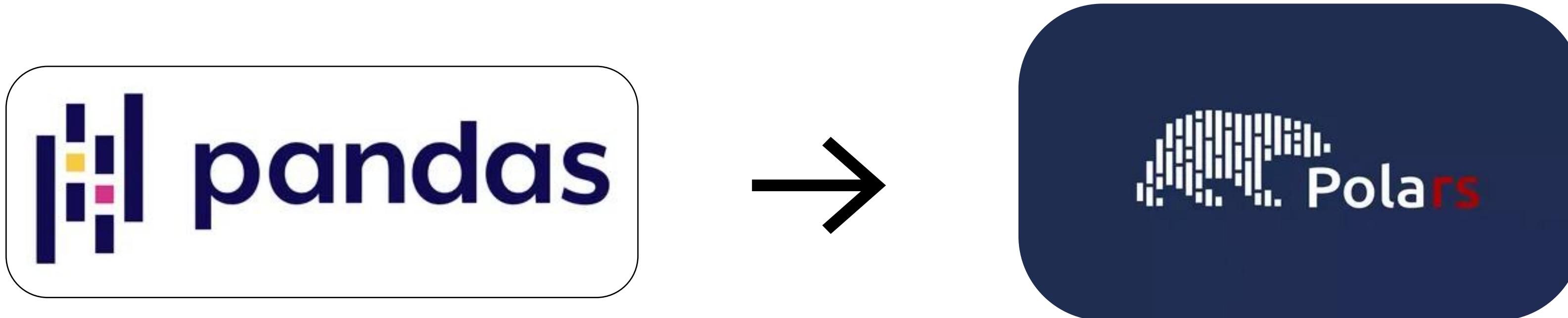


При использовании Time-based валидации, естественно, что появятся Cold-users. Так как у нас отсутствует информация о пользователях, то для холодного старта будет использоваться модель TopPopular.

На оставшихся пользователях и будет обучаться наша модель.

# Data Processing

# Переход от Pandas к Polars



Выигрыш по памяти и скорости выполнения операций:

Потребление RAM:

- pandas: avg ~ 8 GB, max = 14.5 GB
- polars: avg ~ 4 GB, max = 9 GB

Время выполнения по самой затратной операции:

- pandas: ~ 5 min
- polars: ~ 1.5 min

# Модели

# Модели

## Модели первого уровня

- KNN (cosine, BM25, TD-IDF)
- LightFM

## Модели второго уровня

- CatBoost Ranker

## Модели cold users

- TopPopular(topk=50) + BanditRecommender

## Метрики

- Recall@k
- NDCG@k
- MAP@k

# Время

## Таргитированные рекомендации

Этап	Время (мин)
Тренировка + инференс моделей 1 уровня (для трен. ранкера)	15
Обработка результатов инференса моделей 1 уровня	1
Тренировка ранкера	4
Тренировка + инференс моделей 1 уровня (для инф. ранкера)	18
Инференс ранкера	1.5
<b>ИТОГО</b>	<b>40</b>

## Холодные рекомендации

Этап	Время
Тренировка + инференс популярной модели	10 секунд
Тренировка Бандита	≤ 1 минута
Инференс Бандита	1 минута
<b>ИТОГО</b>	<b>≤ 2 минуты 10 секунд</b>

# Модель второго уровня (метрики расчетаны на тестовых данных)

ДО:

```
k = 3: {'ndcg@k': 0.2066009927816457, 'recall@k': 0.12447302675108117, 'map@k': 0.14127237787969413}
```

```
k = 5: {'ndcg@k': 0.24092300833035926, 'recall@k': 0.14942065114748218, 'map@k': 0.13471953670207448}
```

```
k = 10: {'ndcg@k': 0.2951177111158031, 'recall@k': 0.19172486787876636, 'map@k': 0.13414947362623275}
```

```
k = 15: {'ndcg@k': 0.3262363181849842, 'recall@k': 0.2175279962499981, 'map@k': 0.13528035914656342}
```

ПОСЛЕ:

```
k = 3: {'ndcg@k': 0.23283668145585287, 'recall@k': 0.1507586238930032, 'map@k': 0.16838806967687284}
```

```
k = 5: {'ndcg@k': 0.2672899635214155, 'recall@k': 0.1784753701591795, 'map@k': 0.16011627018530403}
```

```
k = 10: {'ndcg@k': 0.3140544182833436, 'recall@k': 0.21806060763468477, 'map@k': 0.1566453522770767}
```

```
k = 15: {'ndcg@k': 0.3387634076939192, 'recall@k': 0.24039676206392474, 'map@k': 0.1565731543568892}
```

\*оставил CatBoost тюнится optuna'ой на ~5 часов

как следствие, уменьшил число итераций ⇒ быстрее инференс:  
время сократилось с 5 минут до 2 минут

# Взаимодействие

# Докер образ

3.12 GB / 5.13 GB in use 1 images

Last refresh: 44 minutes ago ⏱

□	Name	Tag	Created	Size	Actions	Image ID
□	stage_6-app	latest	● 39 minutes ago	4.58 GB	▷ : 🗑	d7452d91bf54

```
dockerfiles > api.Dockerfile > ...
1 # Use an official Python image as a parent image
2 FROM python:3.11.9
3
4 # Install necessary dependencies (including poetry)
5 RUN apt-get update && apt-get install -y --no-install-recommends apt-utils
6 RUN apt-get -y install curl
7 RUN apt-get install libgomp1
8
9 # Set the working directory in the container
10 WORKDIR /app
11
12 # Copy project files into the container
13 COPY pyproject.toml poetry.lock ./ 
14
15 # Upgrade pip and install poetry
16 RUN pip install --upgrade pip setuptools wheel
17 RUN pip install poetry
18
19 # Install project dependencies
20 RUN poetry config virtualenvs.create false && \
21     poetry install --no-interaction --no-ansi --no-root -vvv && \
22     poetry run pip install --upgrade pip setuptools wheel && \
23     poetry run pip install lightfm && \
24     poetry cache clear pypi --all -n
25
26
27 # Copy the rest of the project files (including the source code)
28 COPY . /app
29
30 # Set the entrypoint
31 CMD ["poetry", "run", "uvicorn", "src.api.main:app", "--host", "0.0.0.0", "--port", "8000"]
```

```
docker-compose.yml
  ▷ Run All Services
1 services:
  ▷ Run Service
2   app:
3     build:
4       context: .
5       dockerfile: ./dockerfiles/api.Dockerfile
6     ports:
7       - "8000:8000"
8     volumes:
9       - ./app
10      - ./data:/data
```

# Использован пакетный менеджер poetry

```
7 ]
8 readme = "README.md"
9 requires-python = "=3.11.9"
10 dependencies = [
11     "fastapi (>=0.115.6,<0.116.0)",
12     "uvicorn (>=0.34.0,<0.35.0)",
13     "orjson (>=3.10.14,<4.0.0)",
14     "pydantic-settings (>=2.7.1,<3.0.0)",
15     "# torch (>=2.5.1,<3.0.0)",
16     "# torchvision (>=0.20.1,<0.21.0)",
17     "pyarrow (>=18.1.0,<19.0.0)",
18     "dill (>=0.3.9,<0.4.0)",
19     "# transformers (>=4.48.0,<5.0.0)",
20     "streamlit (>=1.41.1,<2.0.0)",
21     "rectools (>=0.9.0,<0.10.0)",
22     "polars (>=1.20.0,<2.0.0)",
23     "scikit-learn (>=1.6.1,<2.0.0)",
24     "catboost (>=1.2.7,<2.0.0)",
25     "pylint (>=3.3.4,<4.0.0)",
26     "mab2rec (>=1.3.1,<2.0.0)"
```

# codestyle проверен pylint

```
) poetry run pylint ./src/**/*.py
-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)
~/Code/wb/recsys_project/stage_6 main*
)
```

# Реализован API (фреймворк FastAPI)

Есть ручки как для отдельных этапов решения, так и пайплайны

## Логирование

```
✓ Network stage_6_default Created
✓ Container stage_6-app-1 Created
Attaching to app-1
app-1 | Skipping virtualenv creation, as specified in config file.
app-1 | Error 35 (CUDA driver version is insufficient for CUDA runtime version) ignored wh
app-1 | INFO: Started server process [1]
app-1 | INFO: Waiting for application startup.
app-1 | INFO: Application startup complete.
app-1 | INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
app-1 | [2025-01-27 14:10:02,965: INFO] Inferencing 1st stage models: started...
app-1 | [2025-01-27 14:10:02,965: INFO] Inferencing 1st stage KNN models: started...
app-1 | [2025-01-27 14:10:02,965: INFO] CosineRecommender models initialization: started..
app-1 | [2025-01-27 14:10:02,966: INFO] CosineRecommender models initialization: finished!
app-1 | [2025-01-27 14:10:02,966: INFO] CosineRecommender models inference: started...
app-1 | [2025-01-27 14:11:07,257: INFO] CosineRecommender models inference: finished!
app-1 | [2025-01-27 14:11:07,257: INFO] BM25Recommender models initialization: started...
app-1 | [2025-01-27 14:11:07,257: INFO] BM25Recommender models initialization: finished!
app-1 | [2025-01-27 14:11:07,257: INFO] BM25Recommender models inference: started...
app-1 | [2025-01-27 14:12:07,896: INFO] BM25Recommender models inference: finished!
app-1 | [2025-01-27 14:12:07,896: INFO] TFIDFRecommender models initialization: started...
app-1 | [2025-01-27 14:12:07,896: INFO] TFIDFRecommender models initialization: finished!
app-1 | [2025-01-27 14:12:07,896: INFO] TFIDFRecommender models inference: started...
app-1 | [2025-01-27 14:13:07,168: INFO] TFIDFRecommender models inference: finished!
app-1 | [2025-01-27 14:13:07,168: INFO] Inferencing 1st stage KNN models: finished!
app-1 | [2025-01-27 14:13:07,168: INFO] Inferencing 1st stage LFM model: started...
app-1 | [2025-01-27 14:13:07,168: INFO] LFM model initialization: started...
app-1 | [2025-01-27 14:13:07,168: INFO] LFM model initialization: finished!
app-1 | [2025-01-27 14:13:07,168: INFO] LFM model inference: started...
app-1 | [2025-01-27 14:21:15,445: INFO] LFM model inference: finished!
app-1 | [2025-01-27 14:21:15,445: INFO] Inferencing 1st stage LFM model: finished!
app-1 | [2025-01-27 14:21:15,445: INFO] Inferencing 1st stage models: finished!
app-1 | INFO: 172.18.0.1:55908 - "POST /wbtech_proj/inference_models/first_stage_model
s_path=.%2Fdata%2Fmodels%2F&candidates_data_path=.%2Fdata%2Fmodels%2Fcandidates_data%2F HTT

```

v View in Docker Desktop o View Config W Enable Watch  
1 nvim - 2 make □

WBTech proj 0.0.1 OAS 3.1  
[/openapi.json](#)

PreprocessRawData

PreprocessDataForRanker

TrainningFirstStageForRankerTraining

TrainningFirstStageForRankerInference

TrainningRanker

TrainningColdRecsModels

InferenceFirstStageForRankerTraining

InferenceFirstStageForRankerInference

InferenceRanker

InferenceColdRecsModels

Pipelines

POST /wbtech\_proj/pipelines/raw\_data/preprocess/ Preprocess Raw Data Pipeline

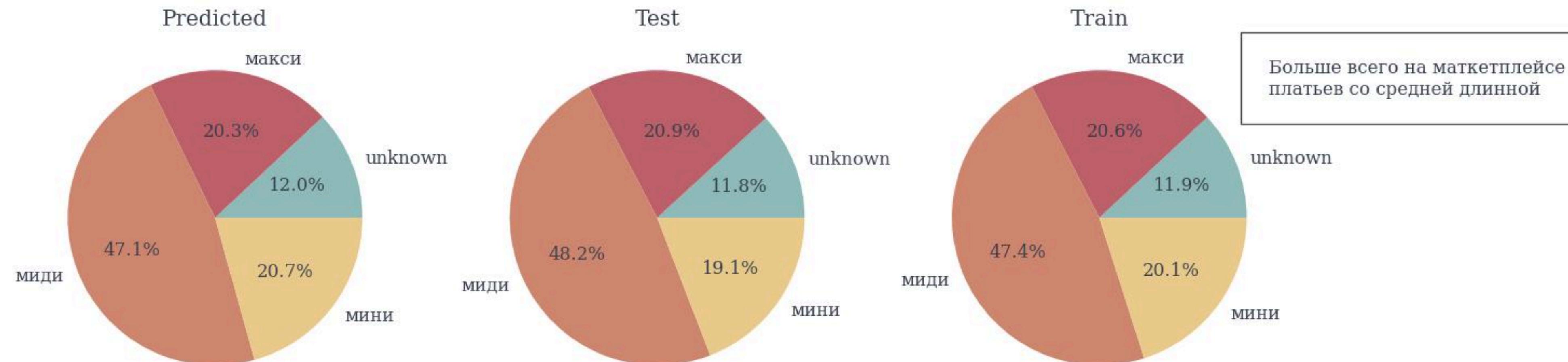
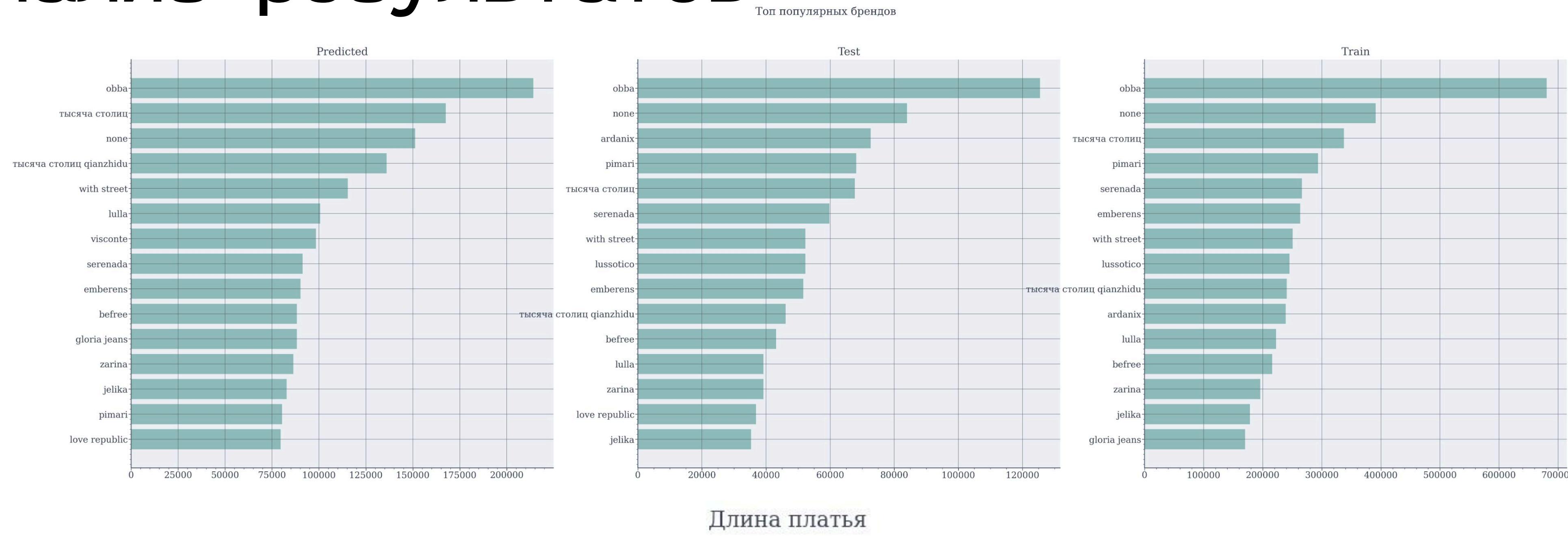
POST /wbtech\_proj/pipelines/train/ranker/ Trainning Ranker Pipeline

POST /wbtech\_proj/pipelines/train/cold\_recbs/ Trainning Cold Recs Pipeline

POST /wbtech\_proj/pipelines/inference/ranker/ Inference Ranker Pipeline

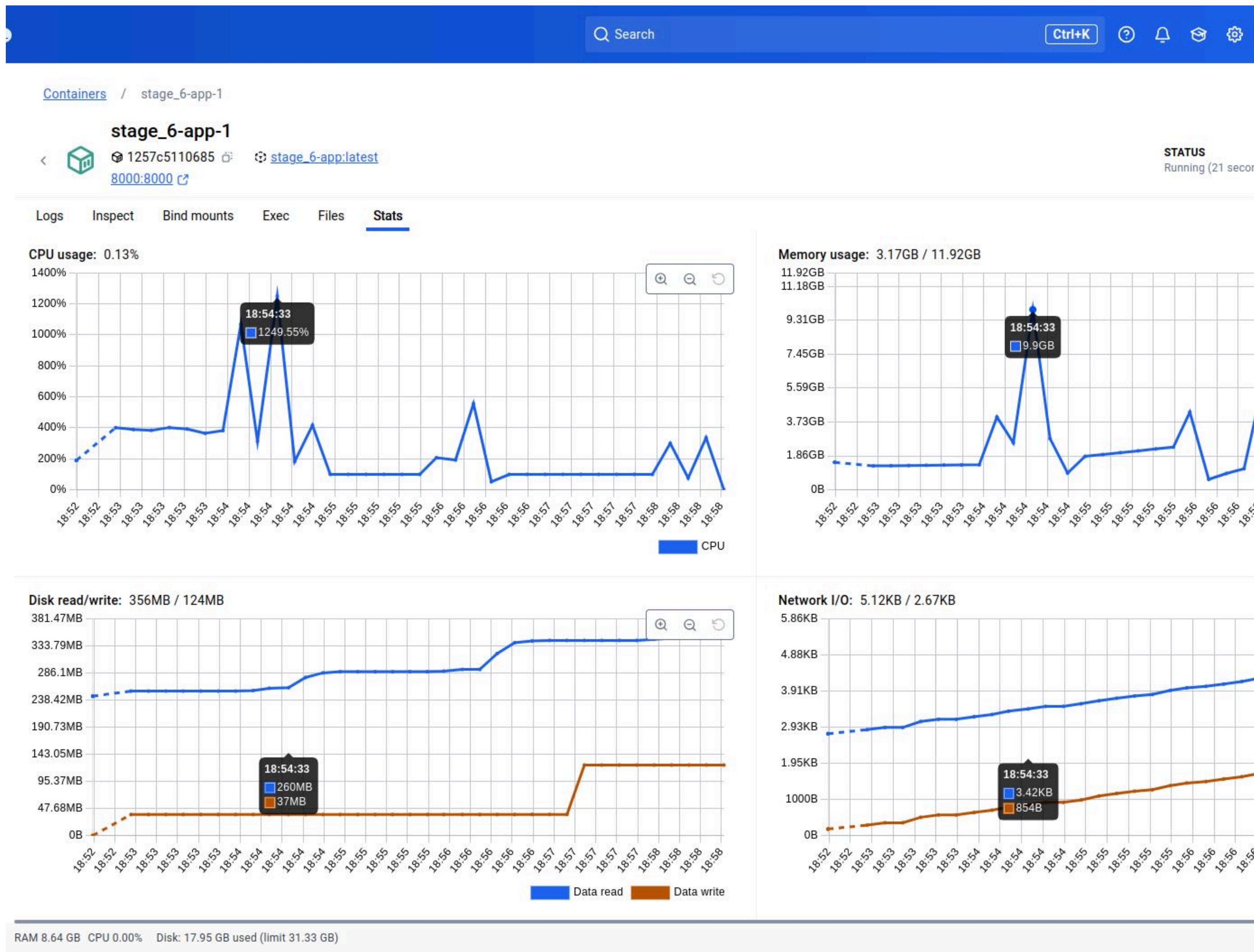
POST /wbtech\_proj/pipelines/inference/cold\_recbs/ Inference Pop Mab Pipeline

# Анализ результатов

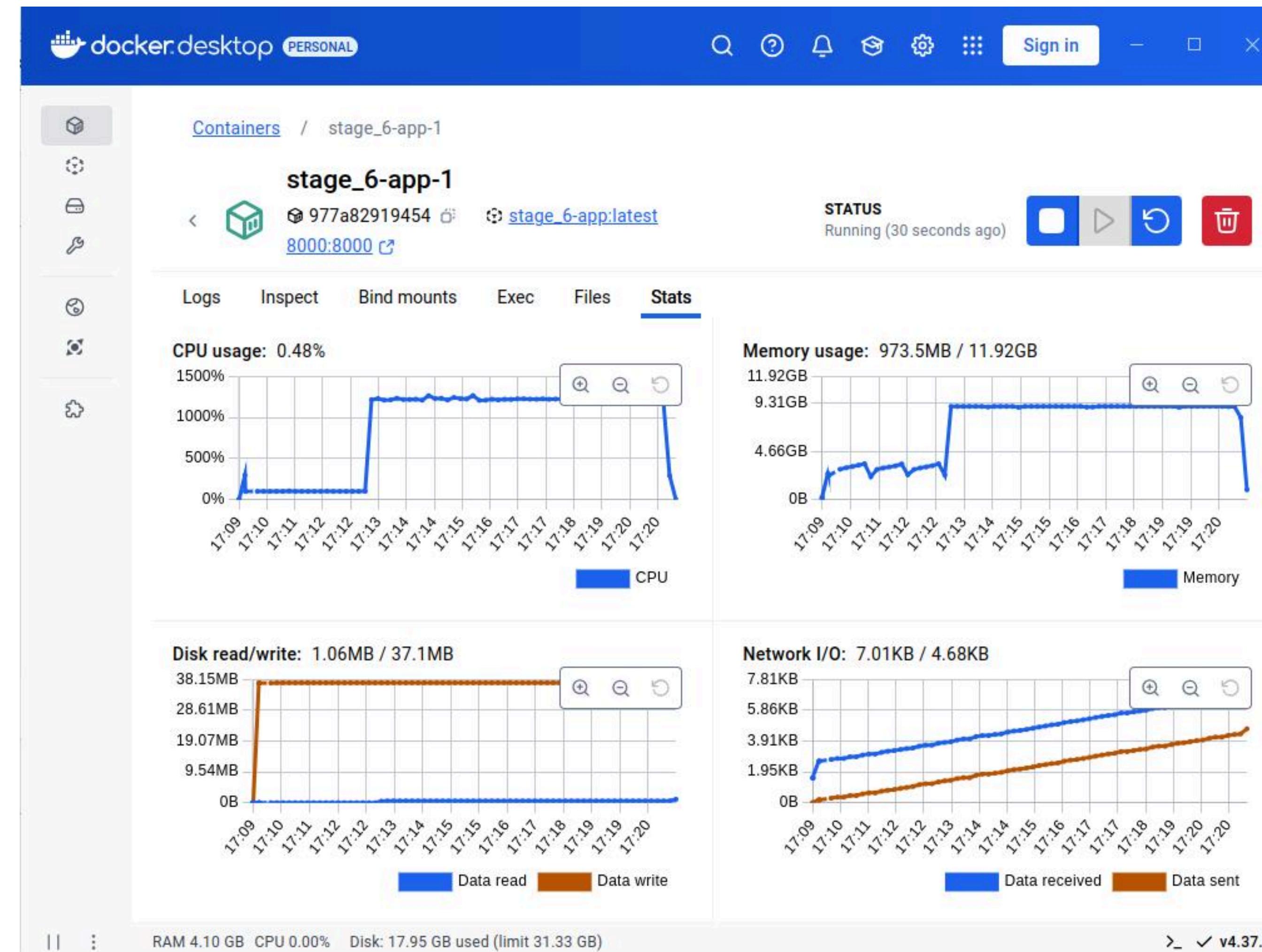


# Приложение

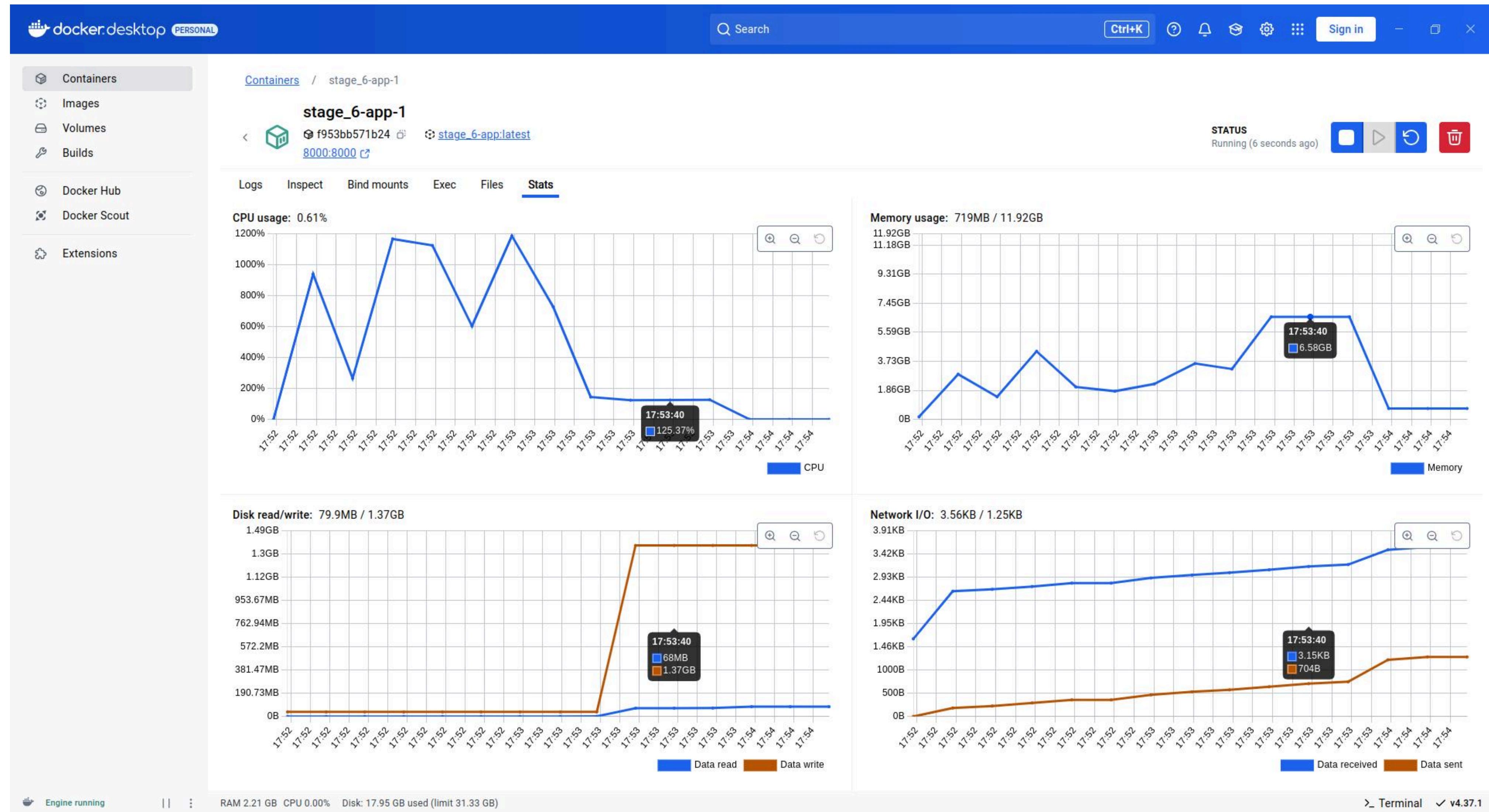
# Предобработка данных



# Обучение моделей 1 уровня



# Обработка данных моделей 1 уровня



# Обучение ранкера

