

CS5830: Big Data Laboratory

Assignment 7

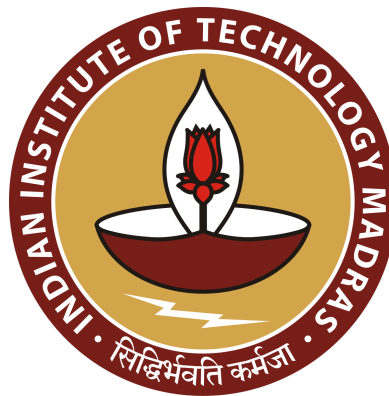
Report

Course Instructor: Balaraman Ravindran

Submitted By: Vishal V

Roll Number: ME20B204

Date: 5/05/2024



Indian Institute of Technology Madras

Chennai 600036, India

REPO: <https://github.com/visvig/cs5830-a7-2024>

Task 1: [25 pts]

Check the modified code for task1.py that now collects counters and gauges

```
app = FastAPI()

# Initialize Prometheus metrics
num_requests = Counter('num_requests', 'Number of requests received', ['method',
'endpoint', 'ip_address'])
processing_time_per_char = Gauge('processing_time_per_char', 'Processing time per
character in microseconds', ['method', 'endpoint'])
memory_usage = Gauge('memory_usage_bytes', 'Memory usage in bytes')
cpu_usage = Gauge('cpu_usage_percent', 'CPU usage percentage')
network_io_sent = Counter('network_io_sent_bytes_total', 'Total number of bytes sent
via network')
network_io_received = Counter('network_io_received_bytes_total', 'Total number of
bytes received via network')

# Setup Prometheus instrumentation for FastAPI
Instrumentator().instrument(app).expose(app)
```

Use prometheus_task1.yaml in the repo.

```
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every
1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1
minute.

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets:
          # Uncomment and modify if Alertmanager is configured
          # - "alertmanager:9093"
```

```
# Load rules once and periodically evaluate them according to the global
'evaluation_interval'.
rule_files:
  # Uncomment and list any rule files
  # - "first_rules.yml"
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  - job_name: "prometheus"
    static_configs:
      - targets: ["localhost:9090"]

  # Add a job to scrape metrics from Node Exporter
  - job_name: "node_exporter"
    static_configs:
      - targets: ["localhost:9100"]

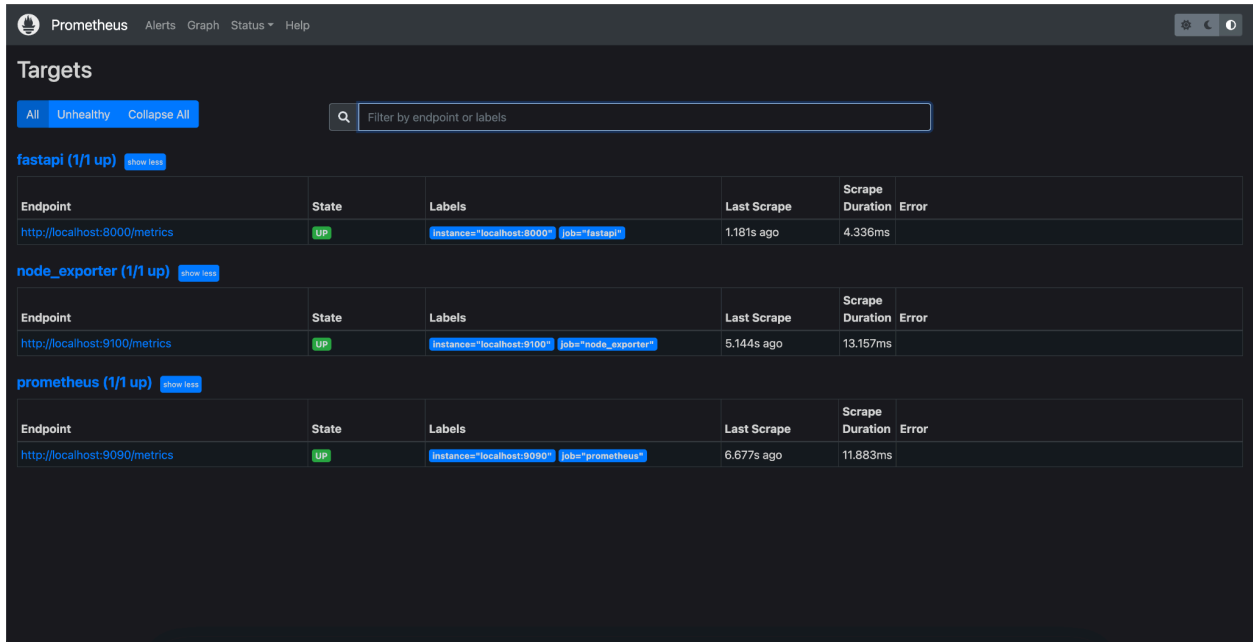
  # Add this job to scrape metrics from FastAPI application
  - job_name: "fastapi"
    scrape_interval: 5s # Scrape more frequently than the default
    static_configs:
      - targets: ["localhost:8000"] # FastAPI exposes metrics at this port
```

Run in terminal

To check the seamless working of Task 1

```
python task1.py mnist_model.h5
```

Prometheus Connection - Successful!



The screenshot shows the Prometheus web interface. At the top, there's a navigation bar with 'Prometheus', 'Alerts', 'Graph', 'Status', and 'Help'. Below this, the 'Targets' section is active. It features a filter bar with 'All', 'Unhealthy', and 'Collapse All' buttons, and a search input field. Three target groups are listed: 'fastapi (1/1 up)', 'node_exporter (1/1 up)', and 'prometheus (1/1 up)'. Each group has a 'show logs' button and a table of targets. All targets are in the 'UP' state.

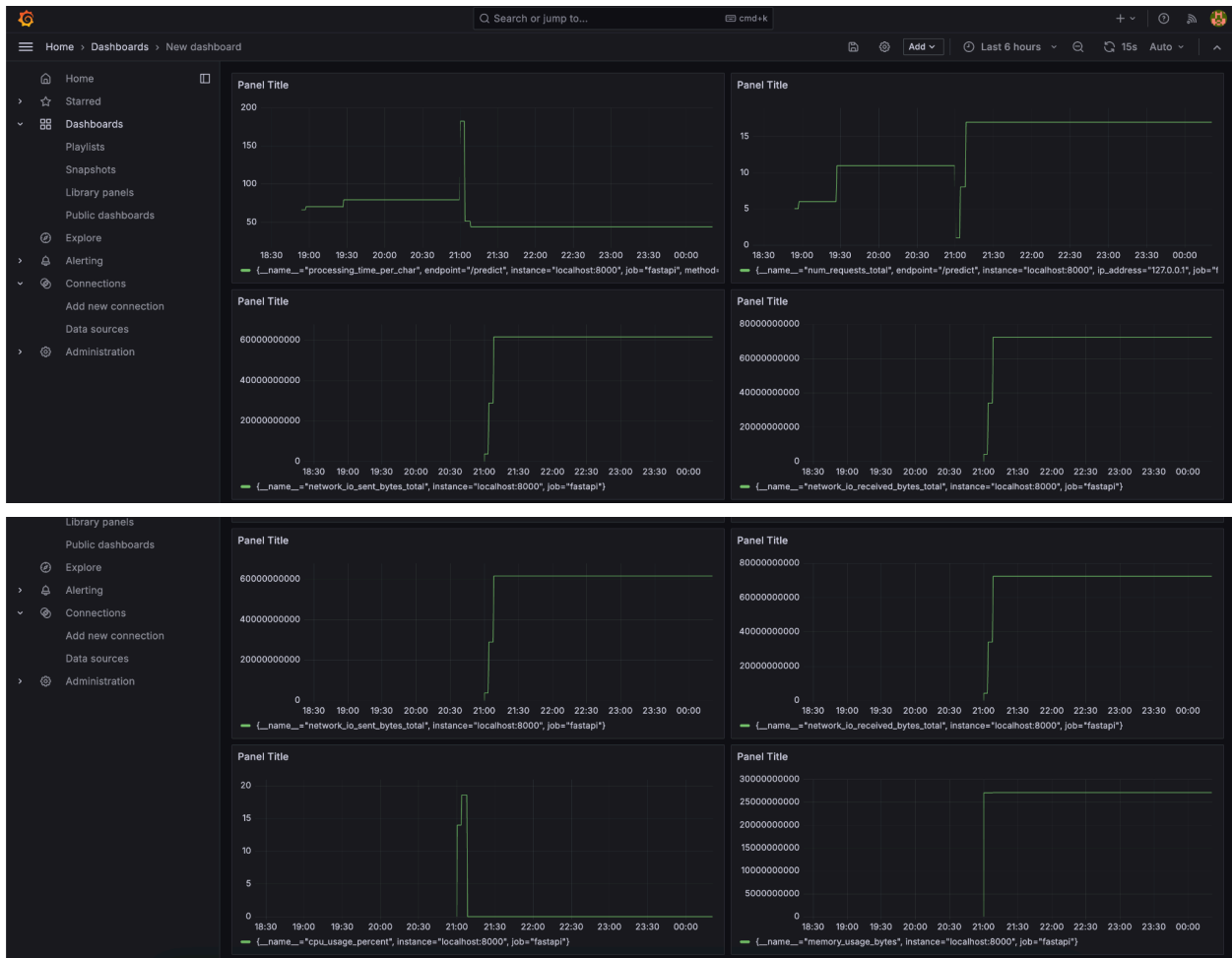
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
fastapi (1/1 up) show logs					
http://localhost:8000/metrics	UP	instance="localhost:8000" job="fastapi"	1.181s ago	4.336ms	
node_exporter (1/1 up) show logs					
http://localhost:9100/metrics	UP	instance="localhost:9100" job="node_exporter"	5.144s ago	13.157ms	
prometheus (1/1 up) show logs					
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	6.677s ago	11.883ms	

Rough Graphana Dashboard - Successful!

Now Collecting..

```
num_requests_total
processing_time_per_char
memory_usage_bytes
cpu_usage_percent
network_io_sent_bytes_total
```

Graphana Dashboard



Task 2: [25 pts]

Use prometheus_task2_docker.yaml from repo

Requirements.txt

```
fastapi==0.94.1
uvicorn==0.29.0
tensorflow==2.15.0
keras==2.15.0
numpy==1.24.4
Pillow==9.4.0
psutil==5.9.7
prometheus-fastapi-instrumentator==6.1.0
prometheus-client==0.16.0
h5py==3.8.0
python-multipart
```

Dockerfile

```
# Use an official Python runtime as a parent image
FROM python:3.10-slim

# Set the working directory in the container
WORKDIR /app

# Install system dependencies including HDF5 and pkg-config
RUN apt-get update && apt-get install -y \
    build-essential \
    libhdf5-dev \
    pkg-config # This package is necessary for h5py installation

# Optionally set HDF5_DIR if h5py cannot find the HDF5 installation
ENV HDF5_DIR=opt/homebrew

# Copy the current directory contents into the container at /app
COPY . /app

# Install any needed packages specified in requirements.txt
RUN pip install --no-cache-dir -r requirements.txt

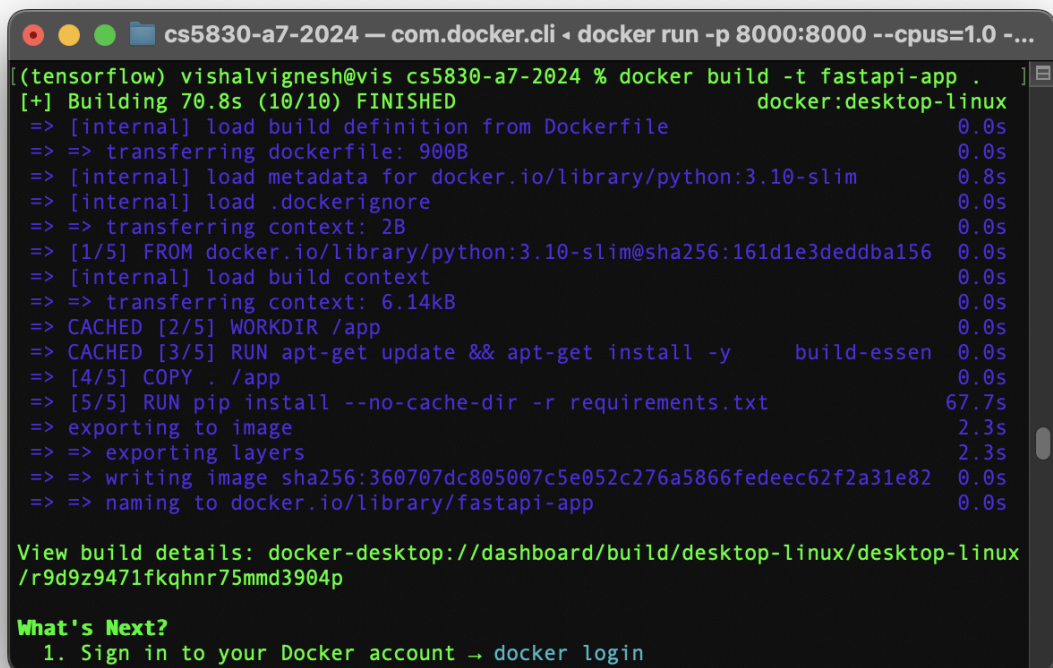
# Make port 8000 available to the world outside this container
```

```
EXPOSE 8000

# Define environment variable
ENV NAME World

# Run app.py when the container launches, including the model path
CMD ["python", "task1.py", "/app/mnist_model.h5"]
```

Run `docker build -t fastapi-app`



The screenshot shows a terminal window titled "cs5830-a7-2024 — com.docker.cli". The user has executed the command `docker build -t fastapi-app .`. The output shows the build process for "docker:desktop-linux" with a progress bar. The build is completed successfully, and the image is named "docker.io/library/fastapi-app". Below the build output, there is a link to view build details and a section titled "What's Next?" with the instruction "1. Sign in to your Docker account → docker login".

```
(tensorflow) vishalvignesh@vis cs5830-a7-2024 % docker build -t fastapi-app .
[+] Building 70.8s (10/10) FINISHED                                docker:desktop-linux
=> [internal] load build definition from Dockerfile                0.0s
=> => transferring dockerfile: 900B                                0.0s
=> [internal] load metadata for docker.io/library/python:3.10-slim 0.8s
=> [internal] load .dockerignore                                  0.0s
=> => transferring context: 2B                                       0.0s
=> [1/5] FROM docker.io/library/python:3.10-slim@sha256:161d1e3deddba156 0.0s
=> [internal] load build context                                  0.0s
=> => transferring context: 6.14kB                                   0.0s
=> CACHED [2/5] WORKDIR /app                                       0.0s
=> CACHED [3/5] RUN apt-get update && apt-get install -y          build-essen 0.0s
=> [4/5] COPY . /app                                              0.0s
=> [5/5] RUN pip install --no-cache-dir -r requirements.txt       67.7s
=> exporting to image                                              2.3s
=> => exporting layers                                              2.3s
=> => writing image sha256:360707dc805007c5e052c276a5866fedeeec62f2a31e82 0.0s
=> => naming to docker.io/library/fastapi-app                      0.0s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux
/r9d9z9471fkqhn75mmd3904p

What's Next?
1. Sign in to your Docker account → docker login
```

Docker Image Built successfully!

BUILDS

[Give feedback](#)

Build container images and artifacts from source code. [Learn more](#)

Selected builder
desktop-linux
Builder settings

[Build history](#)
Active builds

☒ Show only my builds

ID	Name	Builder	Status	Duration	Created	Action
PW0A0J	cs5830-a7-2024	default	Completed	1m 00s	4 hours ago	N/A

After completing the docker build, check run - Successful!

```
Run docker run -p 8000:8000 --name fastapi_app_instance_tf_7 -v /Users/vishalvignesh/codes/cs5830-a7-2024/mnist_model.h5:/app/mnist_model.h5 fastapi-app
```

[illegible]

Now 2 docker instances - Successful!

Run `docker run -p 8000:8000 --cpus="1.0" --name fastapi_app_instance_8 -v /Users/vishalvignesh/codes/cs5830-a7-2024/mnist_model.h5:/app/mnist_model.h5 fastapi-app`

Open another terminal

Run `docker run -p 8001:8000 --cpus="1.0" --name fastapi_app_instance_9 -v /Users/vishalvignesh/codes/cs5830-a7-2024/mnist_model.h5:/app/mnist_model.h5 fastapi-app`

```
cs5830-a7-2024 — com.docker.cli • docker run -p 8000:8000 --cpus=1.0 -...
[(tensorflow) vishalvignesh@vis cs5830-a7-2024 % docker run -p 8000:8000 --cpus="1.0" --name fastapi_app_instance_8 -v /Users/vishalvignesh/codes/cs5830-a7-2024/mnist_model.h5:/app/mnist_model.h5 fastapi-app

INFO:      Started server process [1]
INFO:      Waiting for application startup.
INFO:      Application startup complete.
INFO:      Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
Model loaded successfully.
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
INFO:      192.168.65.1:25227 - "GET /docs HTTP/1.1" 200 OK
INFO:      192.168.65.1:25227 - "GET /openapi.json HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
1/1 [=====] - 0s 44ms/step
INFO:      192.168.65.1:25228 - "POST /predict/ HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
1/1 [=====] - 0s 13ms/step
INFO:      192.168.65.1:25229 - "POST /predict/ HTTP/1.1" 200 OK
INFO:      192.168.65.1:25226 - "GET /metrics HTTP/1.1" 200 OK
1/1 [=====] - 0s 12ms/step
INFO:      192.168.65.1:25229 - "POST /predict/ HTTP/1.1" 200 OK
```

```
cs5830-a7-2024 — com.docker.cli • docker run -p 8001:8000 --cpus=1.0 -...
[(tensorflow) vishalvignesh@vis cs5830-a7-2024 % docker run -p 8001:8000 --cpus="1.0" --name fastapi_app_instance_9 -v /Users/vishalvignesh/codes/cs5830-a7-2024/mnist_model.h5:/app/mnist_model.h5 fastapi-app

INFO:      Started server process [1]
INFO:      Waiting for application startup.
INFO:      Application startup complete.
INFO:      Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
Model loaded successfully.
INFO:      192.168.65.1:56033 - "GET / HTTP/1.1" 404 Not Found
INFO:      192.168.65.1:56033 - "GET /favicon.ico HTTP/1.1" 404 Not Found
INFO:      192.168.65.1:56033 - "GET /docs HTTP/1.1" 200 OK
INFO:      192.168.65.1:56033 - "GET /openapi.json HTTP/1.1" 200 OK
1/1 [=====] - 0s 45ms/step
INFO:      192.168.65.1:56032 - "POST /predict/ HTTP/1.1" 200 OK
1/1 [=====] - 0s 14ms/step
INFO:      192.168.65.1:56035 - "POST /predict/ HTTP/1.1" 200 OK
1/1 [=====] - 0s 14ms/step
INFO:      192.168.65.1:56035 - "POST /predict/ HTTP/1.1" 200 OK
1/1 [=====] - 0s 13ms/step
INFO:      192.168.65.1:56035 - "POST /predict/ HTTP/1.1" 200 OK
1/1 [=====] - 0s 14ms/step
INFO:      192.168.65.1:56035 - "POST /predict/ HTTP/1.1" 200 OK
```

Port 8000

Request body **required** multipart/form-data

file **required**
string(\$binary) 9.jpg

Execute **Clear**

Responses

Curl

```
curl -X 'POST' \
  'http://0.0.0.0:8000/predict/' \
  -H 'accept: application/json' \
  -H 'Content-Type: multipart/form-data' \
  -F 'file=9.jpg;type=image/jpeg'
```

Request URL

http://0.0.0.0:8000/predict/

Server response

Code	Details
200	<p>Response body</p> <pre>{ "digit": "9" }</pre> <p>Response headers</p> <pre>content-length: 13 content-type: application/json date: Sun, 05 May 2024 08:28:41 GMT server: uvicorn</pre>

Responses

Code	Description	Links
200	Successful Response	No links

Port 8001

Parameters Cancel Reset

No parameters

Request body **required** multipart/form-data

file **required**
string(\$binary) 3.jpg

Execute **Clear**

Responses

Curl

```
curl -X 'POST' \
  'http://0.0.0.0:8001/predict/' \
  -H 'accept: application/json' \
  -H 'Content-Type: multipart/form-data' \
  -F 'file=3.jpg;type=image/jpeg'
```

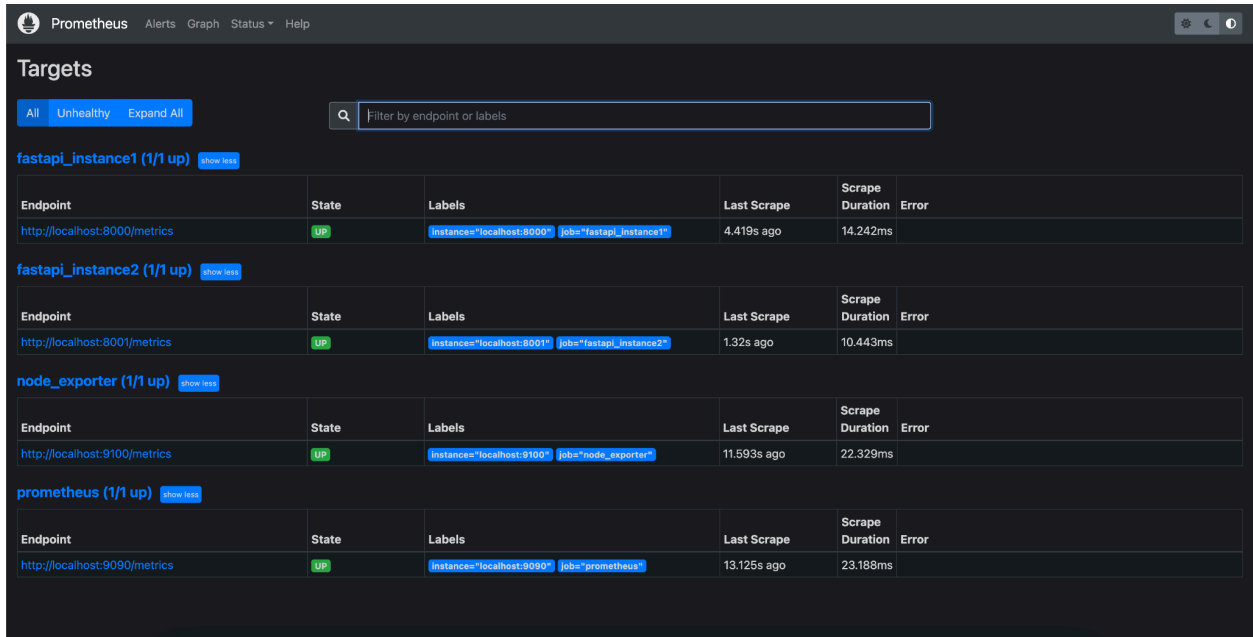
Request URL

http://0.0.0.0:8001/predict/

Server response

Code	Details
200	<p>Response body</p> <pre>{ "digit": "3" }</pre> <p>Response headers</p> <pre>content-length: 13 content-type: application/json date: Sun, 05 May 2024 08:27:40 GMT</pre>

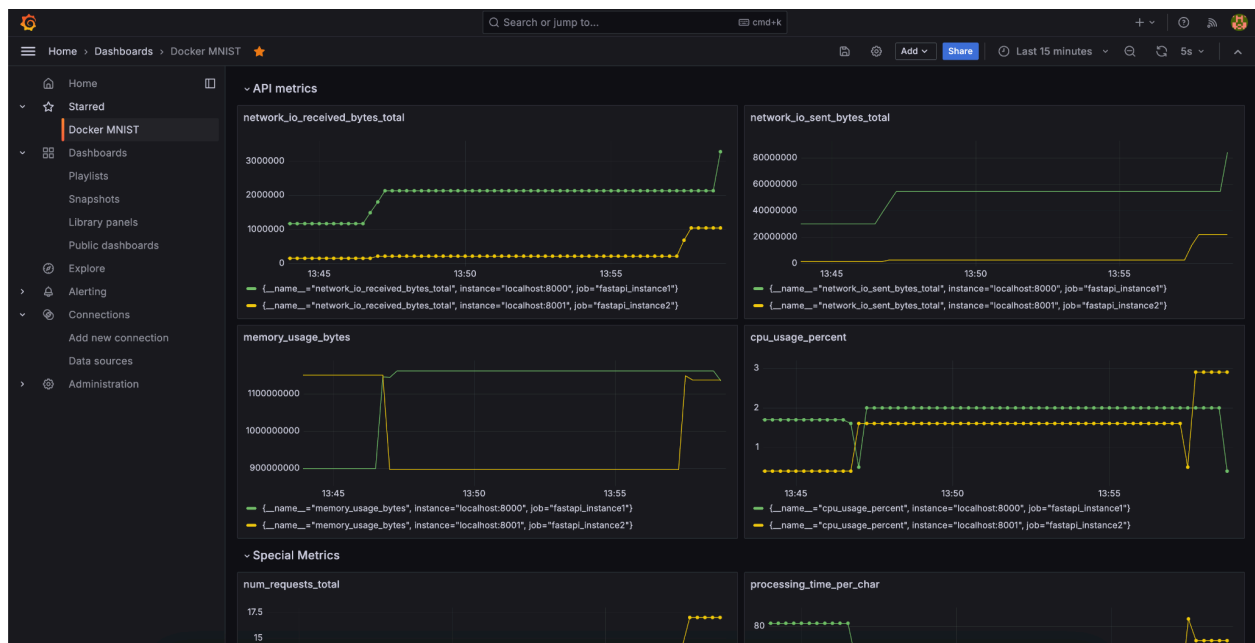
Prometheus now showing 2 instances - Successful!

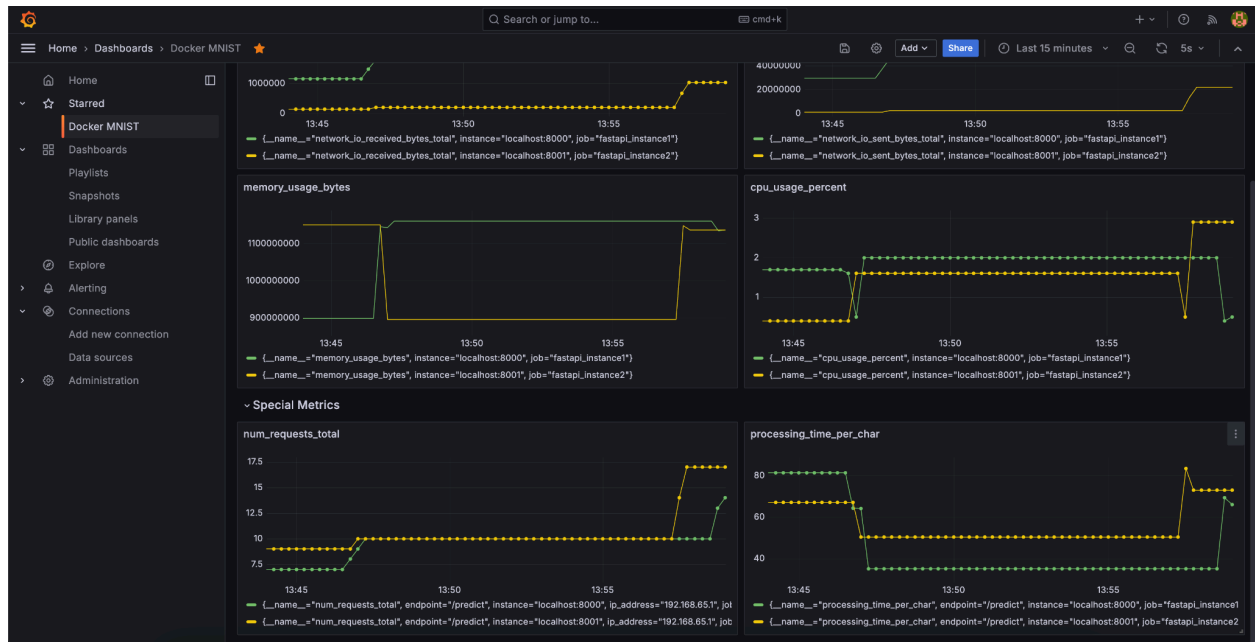


Prometheus Targets page showing four targets, all in 'UP' state. The targets are listed in a table with columns: Endpoint, State, Labels, Last Scrape, Scrape Duration, and Error.

Target	Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
fastapi_instance1 (1/1 up)	http://localhost:8000/metrics	UP	instance="localhost:8000" job="fastapi_instance1"	4.419s ago	14.242ms	
fastapi_instance2 (1/1 up)	http://localhost:8001/metrics	UP	instance="localhost:8001" job="fastapi_instance2"	1.32s ago	10.443ms	
node_exporter (1/1 up)	http://localhost:9100/metrics	UP	instance="localhost:9100" job="node_exporter"	11.593s ago	22.329ms	
prometheus (1/1 up)	http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	13.125s ago	23.188ms	

Graphana Dashboard





All Done! Fun!