Ollama

Run Llama 3, Phi 3, Mistral, Gemma, and other models. Customize and create your own.

- https://ollama.com/
- GitHub: https://github.com/ollama/ollama
- Doc: https://github.com/ollama/ollama/tree/main/docs
- Video: ????????????? AI ?? Ollama ???????????????? - YouTube

Installation

ollama + open webui

```
mkdir ollama-data download open-webui-data
```

docker-compose.yml:

```
services:
  ollama:
    image: ollama/ollama:latest
    ports:
      - 11434:11434
    volumes:
      - ./ollama-data:/root/.ollama
      - ./download:/download
    container_name: ollama
    pull_policy: always
    tty: true
    restart: always
    networks:
      - ollama-docker

  open-webui:
    image: ghcr.io/open-webui/open-webui:main
    container_name: open-webui
    volumes:
```

```
      - ./open-webui-data:/app/backend/data
    depends_on:
      - ollama
    ports:
      - 3000:8080
    environment:
      - 'OLLAMA_BASE_URL=http://ollama:11434'
    extra_hosts:
      - host.docker.internal:host-gateway
    restart: unless-stopped
    networks:
      - ollama-docker

networks:
  ollama-docker:
    external: false
```

ollama

```
mkdir ollama-data download


docker run --name ollama -d --rm \
    -v $PWD/ollama-data:/root/.ollama \
    -v $PWD/download:/download \
    -p 11434:11434 \
    ollama/ollama
```

K8s Deployment

- Ollama Kubernetes: Run AI Models Seamlessly on K8s
- Ollama Kubernetes ??????? ????????????? ????????????? - ?????
- ? Kubernetes ??? llama3 | Kubernetes ????
- Enable GPU Support in Kubernetes: Complete Guide

1. ?? *hostpath-storage*

```
microk8s enable hostpath-storage
microk8s status
```

Verify the Storage Class

```
❯ kubectl get storageclass
NAME                          PROVISIONER            RECLAIMPOLICY   VOLUMEBINDINGMODE
ALLOWVOLUMEEXPANSION   AGE
microk8s-hostpath (default)   microk8s.io/hostpath   Delete          WaitForFirstConsumer
false                  17m
```

2. `ollama-pvc.yaml` :

- PVC ????????? *Pending*??????????????? *Bound*?
- PersistentVolume ???????????????

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: ollama-pvc
  namespace: ollama
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 3Gi
```

3. `ollama-deployment.yaml` :

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: ollama
  namespace: ollama
spec:
  replicas: 1
  selector:
    matchLabels:
      app: ollama
  template:
    metadata:
      labels:
        app: ollama
    spec:
```

```
      containers:
        - name: ollama
          image: ollama/ollama:latest
          env:
            - name: OLLAMA_HOST
              value: 0.0.0.0:11434
          ports:
            - name: http
              containerPort: 11434
              protocol: TCP
          volumeMounts:
            - name: ollama-data
              mountPath: /root/.ollama
      volumes:
        - name: ollama-data
          persistentVolumeClaim:
            claimName: ollama-pvc
```

4. `ollama-svc.yaml` :

```
apiVersion: v1
kind: Service
metadata:
  name: ollama-service
  namespace: ollama
spec:
  selector:
    app: ollama
  ports:
  - protocol: TCP
    port: 11434
    targetPort: 11434
  type: ClusterIP
```

Testing with curl

```
curl -s http://<NODE_IP>:<nodeport>/api/generate -d '{
  "model": "llama2",
```

```
  "prompt": "Why is the sky blue?"
}' | jq -r '.response' | tr -d '\n'
```

Verify GPU support

`kubectl logs -n ollama -l name=ollama`

The last line in the example output above shows that Ollama is using a single Tesla V100-SXM2-16GB GPU.

```
2024/09/27 18:51:55 routes.go:1153: INFO server config env="map[CUDA_VISIBLE_DEVICES:
GPU_DEVICE_ORDINAL: HIP_VISIBLE_DEVICES: HSA_OVERRIDE_GFX_VERSION: HTTPS_PROXY: HTTP_PROXY:
NO_PROXY: OLLAMA_DEBUG:false OLLAMA_FLASH_ATTENTION:false OLLAMA_GPU_OVERHEAD:0
OLLAMA_HOST:http://0.0.0.0:11434 OLLAMA_INTEL_GPU:false OLLAMA_KEEP_ALIVE:5m0s
OLLAMA_LLM_LIBRARY: OLLAMA_LOAD_TIMEOUT:5m0s OLLAMA_MAX_LOADED_MODELS:0 OLLAMA_MAX_QUEUE:512
OLLAMA_MODELS:/root/.ollama/models OLLAMA_NOHISTORY:false OLLAMA_NOPRUNE:false
OLLAMA_NUM_PARALLEL:0 OLLAMA_ORIGINS:[http://localhost https://localhost http://localhost:*
https://localhost:* http://127.0.0.1 https://127.0.0.1 http://127.0.0.1:* https://127.0.0.1:*
http://0.0.0.0 https://0.0.0.0 http://0.0.0.0:* https://0.0.0.0:* app://* file://* tauri://*]
OLLAMA_SCHED_SPREAD:false OLLAMA_TMPDIR: ROCR_VISIBLE_DEVICES: http_proxy: https_proxy:
no_proxy:]"
time=2024-09-27T18:51:55.719Z level=INFO source=images.go:753 msg="total blobs: 0"
time=2024-09-27T18:51:55.719Z level=INFO source=images.go:760 msg="total unused blobs removed:
0"
time=2024-09-27T18:51:55.719Z level=INFO source=routes.go:1200 msg="Listening on [::]:11434
(version 0.3.12)"
time=2024-09-27T18:51:55.720Z level=INFO source=common.go:49 msg="Dynamic LLM libraries"
runners="[cpu_avx cpu_avx2 cuda_v11 cuda_v12 cpu]"
time=2024-09-27T18:51:55.720Z level=INFO source=gpu.go:199 msg="looking for compatible GPUs"
time=2024-09-27T18:51:55.942Z level=INFO source=types.go:107 msg="inference compute" id=GPU-
d8c505a1-8af4-7ce4-517d-4f57fa576097 library=cuda variant=v12 compute=7.0 driver=12.2
name="Tesla V100-SXM2-16GB" total="15.8 GiB" available="15.5 GiB"
```

Models

List Models Installed

```
ollama list
```

Load a GGUF model manually

```
ollama create <my-model-name> -f <modelfile>
```

Page Assist

Page Assist is an open-source Chrome Extension that provides a Sidebar and Web UI for your Local AI model.

- Video: This Chrome Extension Surprised Me - YouTube

Revision #22
Created 2024-04-29 20:14:29 CST by A-Lang (Admin)
Updated 2026-02-01 10:15:17 CST by A-Lang (Admin)