

HOW TO BUILD A DOCKER IMAGES TO RUN FEDERATED LEARNING ON KUBEEDGE ENVIRONMENT

Download the Federated Learning project using gdown:

```
$pip install gdown
```

```
$gdown 1eACMOFFHPakqQ_JUb4AV37cJgmTfmsK5
```

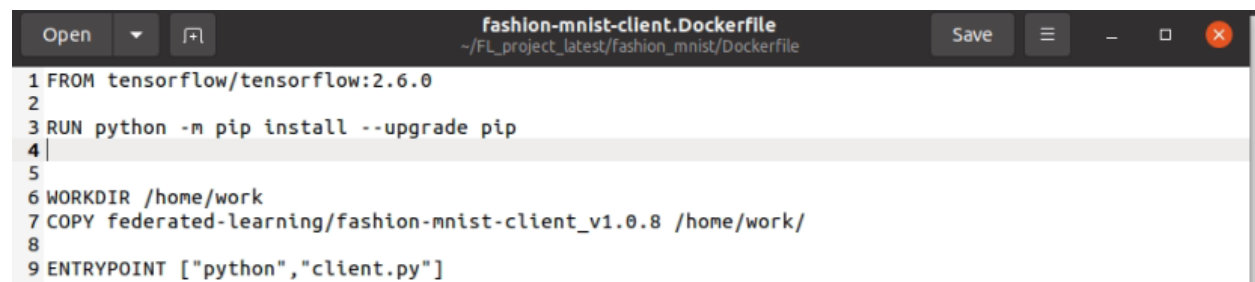
```
$unzip FL_project_latest.zip
```

Content of fashion-mnist-aggregator.Dockerfile:



```
1 FROM tensorflow/tensorflow:2.6.0
2
3 RUN python -m pip install --upgrade pip
4
5
6 WORKDIR /home/work
7 COPY federated-learning/fashion-mnist-cloud_v1.0.8 /home/work/
8
9
10 ENTRYPOINT ["python", "server.py"]
```

Content of fashion-mnist-client.Dockerfile:



```
1 FROM tensorflow/tensorflow:2.6.0
2
3 RUN python -m pip install --upgrade pip
4
5
6 WORKDIR /home/work
7 COPY federated-learning/fashion-mnist-client_v1.0.8 /home/work/
8
9 ENTRYPOINT ["python", "client.py"]
```

```
cd FL_project_latest/fashion-mnist/Dockerfile
```

Build the docker file:

```
sudo docker build -f fashion-mnist-aggregator.Dockerfile -t kubeedge/fashion-mnist-aggregator:v1.0.0 --
label FL_project_latest=fashion_mnist/federated-learning ..
```

```

cloud@cloud:~/FL_project/Dockerfile$ sudo docker build -f fashion-mnist-aggregator.Dockerfile -t kubeedge/fashion-mnist-aggregator:v1.0.0 --label FL_project=federated-learning ..
[sudo] password for cloud:
Sending build context to Docker daemon 22.53kB
Step 1/8 : FROM tensorflow/tensorflow:2.6.0
--> 94fc08a3795e
Step 2/8 : RUN python -m pip install --upgrade pip
--> Using cache
--> b221892db032
Step 3/8 : RUN pip install flwr
--> Using cache
--> 5418597dba98
Step 4/8 : RUN pip install tensorflow
--> Using cache
--> 1c20f1264375
Step 5/8 : WORKDIR /home/work
--> Using cache
--> c2725df32100
Step 6/8 : COPY federated-learning/tensorflow_fashion_mnist /home/work/
--> 5805e4b144a2
Step 7/8 : ENTRYPOINT ["python", "server.py"]
--> Running in 8f31c8f808cf
Removing intermediate container 8f31c8f808cf
--> 35b177f11e5a
Step 8/8 : LABEL FL_project=federated-learning
--> Running in 7b83dfac0ef6
Removing intermediate container 7b83dfac0ef6
--> 34f0c6504574
Successfully built 34f0c6504574
Successfully tagged kubeedge/fashion-mnist-aggregator:v1.0.0

```

sudo docker build -f fashion-mnist-client.Dockerfile -t kubeedge/fashion-mnist-client:v1.0.0 --label FL_project_latest=fashion_mnist/federated-learning ..

```

cloud@cloud:~/FL_project/Dockerfile$ sudo docker build -f fashion-mnist-client.Dockerfile -t kubeedge/fashion-mnist-client:v1.0.0 --label FL_project_latest=fashion_mnist/federated-learning ..
[sudo] password for cloud:
Sending build context to Docker daemon 22.53kB
Step 1/8 : FROM tensorflow/tensorflow:2.6.0
--> 94fc08a3795e
Step 2/8 : RUN python -m pip install --upgrade pip
--> Using cache
--> b221892db032
Step 3/8 : RUN pip install flwr
--> Using cache
--> 5418597dba98
Step 4/8 : RUN pip install tensorflow
--> Using cache
--> 1c20f1264375
Step 5/8 : WORKDIR /home/work
--> Using cache
--> c2725df32100
Step 6/8 : COPY federated-learning/tensorflow_fashion_mnist /home/work/
--> Using cache
--> 5805e4b144a2
Step 7/8 : ENTRYPOINT ["python","client.py"]
--> Running in a30a31d8a50e
Removing intermediate container a30a31d8a50e
--> aba994e4a46e
Step 8/8 : LABEL FL_project=federated-learning
--> Running in 35f8b222c2a3
Removing intermediate container 35f8b222c2a3
--> 38f0e2cee5c8
Successfully built 38f0e2cee5c8
Successfully tagged kubeedge/fashion-mnist-client:v1.0.0

```

Check the docker images on cloud:

sudo docker images

```
cloud@cloud:~$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kubeedge/fashion-mnist-client	v1.0.0	38f0e2cee5c8	2 minutes ago	2.92GB
kubeedge/fashion-mnist-aggregator	v1.0.0	34f0c6504574	About an hour ago	2.92GB
kubeedge/edgemesh-server	latest	a4812893d5c0	7 days ago	50.6MB
kubeedge/edgemesh-server	<none>	1244970211b6	12 days ago	50.6MB
k8s.gcr.io/kube-apiserver	v1.21.8	a5a584eef959	2 weeks ago	126MB
k8s.gcr.io/kube-scheduler	v1.21.8	4cd11f55d2ec	2 weeks ago	50.9MB
k8s.gcr.io/kube-controller-manager	v1.21.8	74e3bdf53cd3	2 weeks ago	120MB
k8s.gcr.io/kube-proxy	v1.21.8	f70621d55c05	2 weeks ago	104MB
calico/node	v3.21.2	f1bca4d4ced2	4 weeks ago	214MB
calico/pod2daemon-flexvol	v3.21.2	7778dd57e506	4 weeks ago	21.3MB
calico/cni	v3.21.2	4c5c32530391	4 weeks ago	239MB
calico/kube-controllers	v3.21.2	b20652406028	4 weeks ago	132MB
tensorflow/tensorflow	2.6.0	94fc08a3795e	4 months ago	1.32GB
k8s.gcr.io/pause	3.4.1	0f8457a4c2ec	11 months ago	683kB
k8s.gcr.io/coredns/coredns	v1.8.0	296a6d5035e2	14 months ago	42.5MB
k8s.gcr.io/etcd	3.4.13-0	0369cf4303ff	16 months ago	253MB

How to send fashion-mnist-client images to edge node:

sudo apt-get install openssh-server

At Cloud Node:

sudo docker save -o client.tar 38f0e2cee5c8

sudo scp client.tar edge1@192.168.27.129:/home/edge1

sudo scp client.tar edge2@192.168.27.130:/home/edge2

At Edge Node:

sudo docker load -i client.tar

sudo docker image tag 38f0e2cee5c8 kubeedge/fashion-mnist-client:v1.0.0

sudo docker images

```
edge1@edge1:~$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kubeedge/fashion-mnist-client	v1.0.0	38f0e2cee5c8	3 hours ago	2.92GB
kubeedge/edgemesh-server	latest	a4812893d5c0	7 days ago	50.6MB
kubeedge/edgemesh-agent	latest	9d055bb5d3ee	7 days ago	61.9MB
kubeedge/edgemesh-server	<none>	1244970211b6	12 days ago	50.6MB
kubeedge/edgemesh-agent	<none>	b5569b947f1c	12 days ago	61.9MB
k8s.gcr.io/kube-apiserver	v1.21.8	a5a584eef959	2 weeks ago	126MB
k8s.gcr.io/kube-controller-manager	v1.21.8	74e3bdf53cd3	2 weeks ago	120MB
k8s.gcr.io/kube-scheduler	v1.21.8	4cd11f55d2ec	2 weeks ago	50.9MB
k8s.gcr.io/kube-proxy	v1.21.8	f70621d55c05	2 weeks ago	104MB
calico/cni	v3.21.2	4c5c32530391	4 weeks ago	239MB
calico/kube-controllers	v3.21.2	b20652406028	4 weeks ago	132MB
k8s.gcr.io/coredns/coredns	v1.8.0	296a6d5035e2	14 months ago	42.5MB
k8s.gcr.io/etcd	3.4.13-0	0369cf4303ff	16 months ago	253MB
kubeedge/pause	3.1	da86e6ba6ca1	4 years ago	742kB

STRUCTURE OF A YAML FILE TO RUN FEDERATED LEARNING PODS

cloud.yaml:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: fashion-mnist
  labels:
    app: federated-learning
spec:
  selector:
    matchLabels:
      app: federated-learning
  template:
    metadata:
      labels:
        app: federated-learning
    spec:
      nodeName: cloud
      containers:
        - name: federated-learning
          image: kubeedge/fashion-mnist-aggregator:v1.0.0
          resources:
            requests:
              memory: "2Gi"
              cpu: "2"
            limits:
              memory: "4Gi"
              cpu: "4"
          args: ["--rounds=5", "--sample_fraction=0.5", "--min_sample_size=2", "--min_num_clients=2", "--server_address=0.0.0.0:8888"]
          ports:
            - containerPort: 8888
      hostNetwork: true #false: provide IP for cloud pod, true: use its cloud IP
      dnsPolicy: Default
```

service.yaml:

###SERVICE###

apiVersion: v1

kind: Service

metadata:

name: federated-learning-svc

spec:

selector:

app: federated-learning

ports:

- port: 12345

protocol: TCP

targetPort: 8888

client.yaml:

#####EDGE 1#####

apiVersion: v1

kind: Pod

metadata:

name: fashion-mnist-edge1

spec:

nodeName: edge1

containers:

- image: kubeedge/fashion-mnist-client:v1.0.0

name: federated-learning-client

args: ["--partition=0", "--clients=1000", "--server address=192.168.27.128:8888"]

hostNetwork: true

dnsPolicy: Default

restartPolicy: Never

Cloud IP address

Position of final element training dataset - 1

#####EDGE 2#####

apiVersion: v1

kind: Pod

metadata:

name: fashion-mnist-edge2

spec:

nodeName: edge2

containers:

- image: kubeedge/fashion-mnist-client:v1.0.0

name: federated-learning-client

Position of first element training dataset


```
args: ["--partition=2000", "--clients=4000", "--server_address=192.168.27.128:8888"]
hostNetwork: true
dnsPolicy: Default
restartPolicy: Never
```

Deploy:

```
$sudo chmod +x run-federated.sh
$sudo chmod +x stop-federated.sh
./run-federated.sh
```

Check results:

```
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide -A
NAMESPACE   NAME                                     READY   STATUS    RESTARTS   AGE   IP             NODE
default     fashion-mnist-6dcb58996c-r56dl         1/1     Running   1          2m14s  192.168.27.128  cloud
default     fashion-mnist-edge1                   0/1     Completed 0          2m4s   192.168.27.129  edge1
default     fashion-mnist-edge2                   0/1     Completed 0          2m4s   192.168.27.130  edge2

cloud@cloud:~/FL_project/yaml$ kubectl get svc -A
NAMESPACE   NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
default     federated-learning-svc ClusterIP    10.108.46.5   <none>        12345/TCP        18m
default     kubernetes           ClusterIP    10.96.0.1     <none>        443/TCP          13d
kube-system kube-dns              ClusterIP    10.96.0.10    <none>        53/UDP,53/TCP,9153/TCP 13d
kube-system metrics-server        ClusterIP    10.99.30.186  <none>        443/TCP          5d18h

cloud@cloud:~/FL_project/yaml$ kubectl get ep -A
NAMESPACE   NAME                 ENDPOINTS                                          AGE
default     federated-learning-svc 192.168.27.128:8888                             18m
default     kubernetes             192.168.27.128:6443                             13d
kube-system kube-dns               192.168.41.4:53,192.168.41.5:53,192.168.41.4:53 + 3 more... 13d
kube-system metrics-server        192.168.27.128:4443                             5d18h

cloud@cloud:~/FL_project/yaml$ kubectl logs -f fashion-mnist-edge1
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
32768/29515 [=====] - 0s 2us/step
40960/29515 [=====] - 0s 2us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26427392/26421880 [=====] - 3s 0us/step
26435584/26421880 [=====] - 3s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
16384/5148 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4423680/4422102 [=====] - 2s 0us/step
4431872/4422102 [=====] - 2s 0us/step
DEBUG flower 2022-01-04 12:49:39,060 | connection.py:36 | ChannelConnectivity.IDLE
DEBUG flower 2022-01-04 12:49:39,061 | connection.py:36 | ChannelConnectivity.CONNECTING
INFO flower 2022-01-04 12:49:39,061 | app.py:61 | Opened (Insecure) gRPC connection
DEBUG flower 2022-01-04 12:49:39,062 | connection.py:36 | ChannelConnectivity.READY
2022-01-04 12:49:50.179215: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization
2)
x_train shape (full): (60000, 28, 28)
y_train shape (full): (60000,)
x_test shape (full): (10000, 28, 28)
y_test shape (full): (10000,)
[50916  47 45254 19771 34478 48596 15781 30481 41153  8964 57386 36186
 44992 57226 34887 36164 54905 13973 12014 48853]
[4436 3829 2434 4449  43 9961 4509 5114 9374 3273 6875  265 2811 9596
 8932 5826 9162 4798 1079 7150]
0
10000
x_train shape: (1000, 28, 28)
y_train shape: (1000,)
x_test shape: (10000, 28, 28)
y_test shape: (10000,)
```

```

cloud@cloud:~/Fl_project/yanl$ kubectl logs -f fashion-mnist-edge2
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
32768/29515 [=====] - 0s 3us/step
40960/29515 [=====] - 0s 2us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26427392/26421880 [=====] - 3s 0us/step
26435584/26421880 [=====] - 3s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
16384/5148 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4423680/4422102 [=====] - 1s 0us/step
4431872/4422102 [=====] - 1s 0us/step
DEBUG flower 2022-01-04 12:49:40,745 | connection.py:36 | ChannelConnectivity.IDLE
DEBUG flower 2022-01-04 12:49:40,746 | connection.py:36 | ChannelConnectivity.CONNECTING
DEBUG flower 2022-01-04 12:49:40,746 | connection.py:36 | ChannelConnectivity.READY
INFO flower 2022-01-04 12:49:40,747 | app.py:61 | Opened (insecure) gRPC connection
2022-01-04 12:49:50.172063: I tensorflow/compiler/mlir/mlir_graph_optimization_pass.cc:185] None of the MLIR Optimization
2)
x_train shape (full): (60000, 28, 28)
y_train shape (full): (60000,)
x_test shape (full): (10000, 28, 28)
y_test shape (full): (10000,)
[50916 47 45254 19771 34478 48596 15781 30481 41153 8964 57386 36186
 44992 57226 34887 36164 54905 13973 12014 48853]
[4436 3829 2434 4449 43 9961 4509 5114 9374 3273 6875 265 2811 9596
 8932 5826 9162 4798 1079 7150]
2000
4000
x_train shape: (2000, 28, 28)
y_train shape: (2000,)
x_test shape: (10000, 28, 28)
y_test shape: (10000,)

```

To stop:

./stop-federated.sh

HOW TO ADD METRIC SERVER/HPA ON KUBEEDGE ON FL POD

sudo nano /etc/kubernetes/manifests/kube-apiserver.yaml

Add line: **--enable-aggregator-routing=true** after kube-apiserver

Follow this instruction to install metric server.

<https://kubedge.io/en/docs/advanced/metrics/>

Check results:

```
cloud@cloud:~/FL_project/yami$ kubectl top node
W0106 21:38:38.435673 1534974 top_node.go:119] Using json format to get metrics. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME      CPU(cores)   CPU%    MEMORY(bytes)  MEMORY%
cloud     722m        9%      4186Mi         54%
edge1     47m         2%      2428Mi         31%
edge2     42m         2%      2381Mi         30%
```

```
cloud@cloud:~/FL_project/yami$ kubectl get pod -o wide -A
NAMESPACE   NAME                                                    READY   STATUS    RESTARTS   AGE   IP              NODE
kube-system  calico-kube-controllers-6b9fbfff44-7k788             1/1     Running   7          7d8h   192.168.41.3    cloud
kube-system  calico-node-4gj8l                                     1/1     Running   0          7d8h   192.168.27.128  cloud
kube-system  coredns-558bd4d5db-d2xz8                             1/1     Running   0          7d11h   192.168.41.1    cloud
kube-system  coredns-558bd4d5db-p9gsc                             1/1     Running   0          7d11h   192.168.41.2    cloud
kube-system  etcd-cloud                                             1/1     Running   0          7d11h   192.168.27.128  cloud
kube-system  kube-apiserver-cloud                                 1/1     Running   0          6m     192.168.27.128  cloud
kube-system  kube-controller-manager-cloud                       1/1     Running   2          7d11h   192.168.27.128  cloud
kube-system  kube-proxy-bmx9c                                     1/1     Running   0          7d8h   192.168.27.128  cloud
kube-system  kube-scheduler-cloud                                1/1     Running   2          7d11h   192.168.27.128  cloud
kube-system  metrics-server-794f9b5fd-bwbb5                      1/1     Running   0          9s     192.168.27.128  cloud
```

Check results while running FL pods:

```
cloud@cloud:~/FL_project/yami$ kubectl top node
W0106 21:55:06.931525 1556057 top_node.go:119] Using json format to get metrics. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME      CPU(cores)   CPU%    MEMORY(bytes)  MEMORY%
cloud     827m        10%     4483Mi         58%
edge1     194m         9%      2599Mi         33%
edge2     220m        11%     2542Mi         32%
```

```
cloud@cloud:~/FL_project/yami$ kubectl get po -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE   NOMINATED NODE   READINESS GATES
fashion-mnist-69c89966bb-x6k8g      1/1     Running   0          93s   192.168.27.128  cloud   <none>            <none>
fashion-mnist-edge1-879b848cd-ngkl6 1/1     Running   0          82s   192.168.27.129  edge1   <none>            <none>
fashion-mnist-edge2-8455d6df66-d52m6 1/1     Running   0          82s   192.168.27.130  edge2   <none>            <none>
```


Add HPA Autoscaling into FL pods running on edge1/edge2:

```
####HPA AutoScaler####
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  name: fashion-mnist-edge1
  namespace: default
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: fashion-mnist-edge1
  minReplicas: 1
  maxReplicas: 5
  metrics:
  - type: Resource
    resource:
      name: cpu #memory
      target:
        type: Utilization
        averageUtilization: 110 #Must higher than the CPU request resources of FL pod
  behavior:
    scaleDown:
      stabilizationWindowSeconds: 120
      policies:
      - type: Pods
        value: 5
        periodSeconds: 120
      selectPolicy: Min #Max
```

Check HPA status on FL pods (edge1/edge2):

```
cloud@cloud:~/FL_project/yaml$ kubectl get hpa
```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
fashion-mnist-edge1	Deployment/fashion-mnist-edge1	<unknown>/60%	1	5	1	100s
fashion-mnist-edge2	Deployment/fashion-mnist-edge2	<unknown>/60%	1	5	1	100s

```
cloud@cloud:~/FL_project/yaml$ kubectl get hpa
```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
fashion-mnist-edge1	Deployment/fashion-mnist-edge1	189%/60%	1	5	1	82s
fashion-mnist-edge2	Deployment/fashion-mnist-edge2	189%/60%	1	5	1	82s

```
cloud@cloud:~/FL_project/yaml$ kubectl get hpa
```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
fashion-mnist-edge1	Deployment/fashion-mnist-edge1	189%/60%	1	5	4	94s
fashion-mnist-edge2	Deployment/fashion-mnist-edge2	189%/60%	1	5	4	94s

Desired Pods = ceil(current_pods * (current value/ target value))
 = 1 * (1.89 / 0.6) = 3.15 ~~ 4 pods for each node.

```
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED	READ
fashion-mnist-69c89966bb-pz784	1/1	Running	0	5m23s	192.168.27.128	cloud	<none>	<non
fashion-mnist-edge1-59f96fc4cc-8kk4k	1/1	Running	0	10s	192.168.27.129	edge1	<none>	<non
fashion-mnist-edge1-59f96fc4cc-bbgh6	0/1	Error	0	10s	192.168.27.129	edge1	<none>	<non
fashion-mnist-edge1-59f96fc4cc-lj67b	1/1	Running	0	85s	192.168.27.129	edge1	<none>	<non
fashion-mnist-edge1-59f96fc4cc-qkmxw	0/1	Error	0	10s	192.168.27.129	edge1	<none>	<non
fashion-mnist-edge2-74c67b88cd-5kdld	0/1	Pending	0	10s	<none>	edge2	<none>	<non
fashion-mnist-edge2-74c67b88cd-bjj2t	1/1	Running	0	85s	192.168.27.130	edge2	<none>	<non
fashion-mnist-edge2-74c67b88cd-cthgs	0/1	Pending	0	10s	<none>	edge2	<none>	<non
fashion-mnist-edge2-74c67b88cd-dw2nr	0/1	Pending	0	10s	<none>	edge2	<none>	<non

Recheck after a couple of minutes:

```
cloud@cloud:~/FL_project/yaml$ kubectl get hpa
```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
fashion-mnist-edge1	Deployment/fashion-mnist-edge1	19%/60%	1	5	5	12m
fashion-mnist-edge2	Deployment/fashion-mnist-edge2	19%/60%	1	5	5	12m

```
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED	READIN
fashion-mnist-69c89966bb-pz784	1/1	Running	0	17m	192.168.27.128	cloud	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-8kk4k	1/1	Running	0	11m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-bbgh6	1/1	Running	1	11m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-f924t	1/1	Running	0	11m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-lj67b	1/1	Running	0	13m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-qkmxw	1/1	Running	2	11m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge2-74c67b88cd-5kdld	1/1	Running	2	11m	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-bjj2t	1/1	Running	0	13m	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-cthgs	1/1	Running	2	11m	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-d7g6s	1/1	Running	0	11m	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-dw2nr	1/1	Running	2	11m	192.168.27.130	edge2	<none>	<none>

- ⇒ Check logs from every pods and all of them are working normal and all pods in 1 node are training synchronously (the same round).
- ⇒ Although the number of CPU percent is decreased, but the number of replicas is still unchanging. (it should be decreased??? – after 5 minutes for scale down).

<https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/>

```
cloud@cloud:~/FL_project/yami$ kubectl get hpa
```

NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
fashion-mnist-edge1	Deployment/fashion-mnist-edge1	11%/60%	1	5	5	28m
fashion-mnist-edge2	Deployment/fashion-mnist-edge2	11%/60%	1	5	5	28m

It's lead to high % CPU performance at each node because the HPA didn't scale down:

```
cloud@cloud:~/FL_project/yami$ kubectl top node
```

NAME	CPU(cores)	CPU%	MEMORY(bytes)	MEMORY%
cloud	498m	6%	5097Mi	66%
edge1	1992m	99%	6092Mi	77%
edge2	2000m	100%	5015Mi	64%

New results: now HPA can scale up and scale down.

```
cloud@cloud:~/FL_project/yami$ kubectl get pod -o wide
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE	READIN
ESS GATES								
fashion-mnist-69c89966bb-zzpwj	1/1	Running	2	28m	192.168.27.128	cloud	<none>	<none>
fashion-mnist-edge1-59f96fc4cc-rzszh	1/1	Running	4	28m	192.168.27.129	edge1	<none>	<none>
fashion-mnist-edge2-74c67b88cd-9rbvn	1/1	Running	1	15m	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-g65wz	1/1	Running	0	48s	192.168.27.130	edge2	<none>	<none>
fashion-mnist-edge2-74c67b88cd-n9phq	1/1	Running	1	15m	192.168.27.130	edge2	<none>	<none>

```
cloud@cloud:~/FL_project/yami$ kubectl describe hpa fashion-mnist-edge2
```

```
Name: fashion-mnist-edge2
Namespace: default
Labels: <none>
Annotations: <none>
CreationTimestamp: Tue, 11 Jan 2022 17:36:14 -0800
Reference: Deployment/fashion-mnist-edge2
Metrics: ( current / target )
  resource cpu on pods (as a percentage of request): 93% (930m) / 110%
Min replicas: 1
Max replicas: 5
Deployment pods: 3 current / 3 desired
Conditions:
  Type            Status  Reason                        Message
  ----            -
  AbleToScale     True    ReadyForNewScale             recommended size matches current size
  ScalingActive   True    ValidMetricFound             the HPA was able to successfully calculate a replica count from cpu resource utilization (percentage of request)
  ScalingLimited  False   DesiredWithinRange           the desired count is within the acceptable range
Events:
  Type            Reason                        Age                    From                    Message
  ----            -
  Warning         FailedGetResourceMetric        28m (x2 over 28m)      horizontal-pod-autoscaler  failed to get cpu utilization: unable to get metrics for resource cpu: no metrics returned from resource metrics API
  Warning         FailedComputeMetricsReplicas  28m (x2 over 28m)      horizontal-pod-autoscaler  invalid metrics (1 invalid out of 1), first error is: failed to get cpu utilization: unable to get metrics for resource cpu: no metrics returned from resource metrics API
  Normal          SuccessfulRescale             15m (x2 over 26m)      horizontal-pod-autoscaler  New size: 4; reason: cpu resource utilization (percentage of request) above target
  Normal          SuccessfulRescale             6m10s (x2 over 20m)    horizontal-pod-autoscaler  New size: 2; reason: All metrics below target
  Normal          SuccessfulRescale             55s                    horizontal-pod-autoscaler  New size: 3; reason: cpu resource utilization (percentage of request) above target
```


Edge 1: 10K training samples, Edge 2: 20K training samples.

(Cloud node completed in aggregation phase, edge nodes is training at round 4).

[illegible]

Check CPU/Memory performance on each pod/each node at end of a loop (1 loop = 5 rounds).

```
cloud@cloud: ~/FL_project/yaml
fashion-mnist-edge1-5495c469dc-qn4l5 1862m 661Mi
fashion-mnist-edge2-5d69f9b795-hlqxz 1917m 791Mi
cloud@cloud:~/FL_project/yaml$ kubectl top pod
W0117 18:12:16.487590 901761 top_pod.go:140] Using json format to get metrics
. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME                                CPU(cores)    MEMORY(bytes)
fashion-mnist-69c8996bb-7g5bv      0m            0Mi
fashion-mnist-edge1-5495c469dc-qn4l5      0m            0Mi
fashion-mnist-edge2-5d69f9b795-hlqxz      0m            0Mi
cloud@cloud:~/FL_project/yaml$ kubectl top node
W0117 18:12:20.754588 901833 top_node.go:119] Using json format to get metrics
. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME                                CPU(cores)    CPUS%    MEMORY(bytes)    MEMORY%
cloud  975m      12%      5111Mi          66%
edge1  19m        0%      2918Mi          37%
edge2  55m        2%      2222Mi          28%
cloud@cloud:~/FL_project/yaml$

cloud@cloud: ~/FL_project/yaml
INFO flower 2022-01-18 02:11:33,547 | app.py:120 | app_fit: metrics_distributed {}
INFO flower 2022-01-18 02:11:33,547 | app.py:121 | app_fit: losses centralized [(0, 2.3160588239739691), (0, 0.3543362319469452), (2, 0.30772948265075684), (3, 0.26319000124931335), (4, 0.2637775242328644), (5, 0.2706083642463684)]
INFO flower 2022-01-18 02:11:33,547 | app.py:122 | app_fit: metrics centralized {'accuracy': [(0, 0.03759999945759773), (1, 0.0733999729156494), (2, 0.8883000016122463), (3, 0.9035999774932861), (4, 0.9082000255584717), (5, 0.910099983215332)]}]
DEPRECATION WARNING: deprecated 'eval_fn' return format
    loss, accuracy
move to
    loss, {"accuracy": accuracy}
instead. Note that compatibility with the deprecated return format will be removed in a future release.
cloud@cloud:~/FL_project/yaml$

cloud@cloud: ~/FL_project/yaml
313/313 [=====] - 20s 63ms/step - loss: 0.2430 - accuracy: 0.9103
Epoch 2/2
313/313 [=====] - 18s 57ms/step - loss: 0.2088 - accuracy: 0.9234
157/157 - 2s - loss: 0.2814 - accuracy: 0.8966
Epoch 1/2
313/313 [=====] - 20s 63ms/step - loss: 0.1913 - accuracy: 0.9301
Epoch 2/2
313/313 [=====] - 18s 56ms/step - loss: 0.1665 - accuracy: 0.9385
157/157 - 2s - loss: 0.3047 - accuracy: 0.8981
Epoch 1/2
313/313 [=====] - 19s 62ms/step - loss: 0.1565 - accuracy: 0.9409
Epoch 2/2
313/313 [=====] - 18s 57ms/step - loss: 0.1353 - accuracy: 0.9488
157/157 - 2s - loss: 0.3159 - accuracy: 0.9005
DEBUG flower 2022-01-18 02:11:34,141 | connection.py:68 | Insecure gRPC channel closed
INFO flower 2022-01-18 02:11:34,142 | app.py:72 | Disconnect and shut down
cloud@cloud:~/FL_project/yaml$
```


Edge 1: 10K training samples, Edge 2: 20K training samples.

=> Training time on each loop: 4m39s

+ **With HPA:** Check CPU/Memory performance on each pod/each node.

Edge 1: 10K training samples, Edge 2: 20K training samples.

Test case: Apply HPA on edge 1, resources on edge 1 pod: (Request: 1 vCPU, 1Gi memory, Limits: 2vCPU, 2Gi memory), HPA info: (Min:1, Max: 5, CPU target: 120% ~ 1.2vCPU, scale down time: 120s).

```
cloud@cloud: ~/FL_project/yaml
edge2 1290m 64% 3050Mi 39%
cloud@cloud:~/FL_project/yaml$ kubectl top pod
W0117 22:46:35.707941 1260820 top_pod.go:140] Using json format to get metrics
. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME CPU(cores) MEMORY(bytes)
fashion-mnist-69c8996bb-lxvr8 1811m 502Mi
fashion-mnist-edge1-5495c469dc-27dhc 281m 446Mi
fashion-mnist-edge1-5495c469dc-thtfx 207m 635Mi
fashion-mnist-edge2-5d69f9b795-z89b6 779m 665Mi
cloud@cloud:~/FL_project/yaml$ kubectl top node
W0117 22:46:41.550511 1260872 top_node.go:119] Using json format to get metrics
. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME CPU(cores) CPU% MEMORY(bytes) MEMORY%
cloud 683m 8% 5040Mi 66%
edge1 242m 12% 3521Mi 45%
edge2 677m 33% 3132Mi 40%
cloud@cloud:~/FL_project/yaml$

cloud@cloud:~/FL_project/yaml
4642/5000 [=====] - ETA: 0s - loss: 0.2759 - accuracy:
4670/5000 [=====] - ETA: 0s - loss: 0.2757 - accuracy:
4698/5000 [=====] - ETA: 0s - loss: 0.2753 - accuracy:
4725/5000 [=====] - ETA: 0s - loss: 0.2756 - accuracy:
4753/5000 [=====] - ETA: 0s - loss: 0.2764 - accuracy:
4781/5000 [=====] - ETA: 0s - loss: 0.2764 - accuracy:
4809/5000 [=====] - ETA: 0s - loss: 0.2773 - accuracy:
4832/5000 [=====] - ETA: 0s - loss: 0.2782 - accuracy:
4854/5000 [=====] - ETA: 0s - loss: 0.2800 - accuracy:
4882/5000 [=====] - ETA: 0s - loss: 0.2797 - accuracy:
4909/5000 [=====] - ETA: 0s - loss: 0.2791 - accuracy:
4937/5000 [=====] - ETA: 0s - loss: 0.2795 - accuracy:
4964/5000 [=====] - ETA: 0s - loss: 0.2793 - accuracy:
4989/5000 [=====] - ETA: 0s - loss: 0.2790 - accuracy:
5000/5000 [=====] - ETA: 0s - loss: 0.2792 - accuracy: 0.892
INFO flower 2022-01-18 06:46:23,521 | server.py:154 | fit progress: (3, 0.27921
098470687866, {'accuracy': 0.8981999754905701, 157.43164375401102})
INFO flower 2022-01-18 06:46:23,521 | server.py:199 | evaluate_round: no client
s selected, cancel
DEBUG flower 2022-01-18 06:46:23,521 | server.py:255 | fit_round: strategy samp
led 2 clients (out of 3)

cloud@cloud:~/FL_project/yaml
Epoch 2/2
157/157 [=====] - 10s 63ms/step - loss: 0.4067 - accuracy: 0.8531
157/157 - 3s - loss: 0.3995 - accuracy: 0.8561
Epoch 1/2
157/157 [=====] - 11s 73ms/step - loss: 0.3378 - accuracy: 0.8765
Epoch 2/2
157/157 [=====] - 11s 69ms/step - loss: 0.2800 - accuracy: 0.8984
157/157 - 2s - loss: 0.3436 - accuracy: 0.8700
Epoch 1/2
157/157 [=====] - 17s 108ms/step - loss: 0.2818 - accuracy: 0.8922
Epoch 2/2
157/157 [=====] - 17s 109ms/step - loss: 0.2353 - accuracy: 0.9121
157/157 - 4s - loss: 0.2922 - accuracy: 0.8941

cloud@cloud:~/FL_project/yaml
NunPyClient (recommended) or Client. NunPyClient is recommended because it
is conceptually very similar to KerasClient.

Epoch 1/2
1/313 [.....] - ETA: 2:05 - loss: 2.3262 - accuracy: 0.015
2/313 [.....] - ETA: 18s - loss: 2.2879 - accuracy: 0.1875
313/313 [=====] - 20s 62ms/step - loss: 0.5440 - accuracy: 0.8024
Epoch 2/2
313/313 [=====] - 18s 58ms/step - loss: 0.3424 - accuracy: 0.8763
157/157 - 2s - loss: 0.3405 - accuracy: 0.8742
Epoch 1/2
313/313 [=====] - 20s 65ms/step - loss: 0.3083 - accuracy: 0.8879
Epoch 2/2
313/313 [=====] - 18s 59ms/step - loss: 0.2598 - accuracy: 0.9039
157/157 - 2s - loss: 0.3010 - accuracy: 0.8892
Epoch 1/2
313/313 [=====] - 20s 63ms/step - loss: 0.2373 - accuracy: 0.9120
Epoch 2/2
```

Catch some errors since Cloud pod is still not ready but Edge pods have tried to connect => lead to failed to connect to all address (Cloud IP address).

```
Activities Terminal Jan 17 23:05
cloud@cloud: ~/FL_project/yaml
NAME CPU(cores) CPU% MEMORY(bytes) MEMORY%
cloud 695m 68% 4614Mi 60%
edge1 57m 2% 2340Mi 29%
edge2 52m 2% 2392Mi 30%
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide
NAME READY STATUS RESTARTS AGE IP
fashion-mnist-69c8996bb-lxvr8 1/1 Running 2 21m 192
fashion-mnist-edge1-5495c469dc-27dhc 0/1 Error 7 19m 192
fashion-mnist-edge1-5495c469dc-7v9qd 0/1 Error 6 13m 192
fashion-mnist-edge1-5495c469dc-thtfx 0/1 Error 6 21m 192
fashion-mnist-edge2-5d69f9b795-z89b6 0/1 Error 7 21m 192
cloud@cloud:~/FL_project/yaml$

cloud@cloud:~/FL_project/yaml
apiVersion: apps/v1

cloud@cloud:~/FL_project/yaml
dart.so.11.0: cannot open shared object file: No such file or directory
2022-01-18 06:52:46.393065: I tensorflow/stream_executor/cuda/cudart_stub.cc:29
] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datas
ts/train-labels-idx1-ubyte.gz
32768/29515 [=====] - 0s 15us/step
40960/29515 [=====] - 0s 12us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datas
ts/train-images-idx3-ubyte.gz
Ac
cloud@cloud:~/FL_project/yaml$ kubectl logs -f fashion-mnist-69c8996bb-lxvr8
2022-01-18 06:52:46.393028: W tensorflow/stream_executor/platform/default/dso.l
oader.cc:64] Could not load dynamic library 'libcudart.so.11.0': dlerror: libc
dart.so.11.0: cannot open shared object file: No such file or directory
2022-01-18 06:52:46.393065: I tensorflow/stream_executor/cuda/cudart_stub.cc:29
] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datas
ts/train-labels-idx1-ubyte.gz
32768/29515 [=====] - 0s 15us/step
40960/29515 [=====] - 0s 12us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datas
ts/train-images-idx3-ubyte.gz

cloud@cloud:~/FL_project/yaml$

cloud@cloud:~/FL_project/yaml
File "/usr/local/lib/python3.6/dist-packages/flwr/client/grpc_client/connection.py", line 60, in <lambda>
receive: Callable[[ServerMessage]] = lambda: next(server_message_iterator)
File "/usr/local/lib/python3.6/dist-packages/grpc_channel.py", line 426, in _next
return self._next()
File "/usr/local/lib/python3.6/dist-packages/grpc_channel.py", line 809, in _next
raise self
grpc_channel._MultiThreadedRendezvous: <MultiThreadedRendezvous of RPC that termina
ted with:
status = StatusCode.UNAVAILABLE
details = "failed to connect to all addresses"
debug_error_string = "[{"created": "01642489297.482068417", "description": "Fai
led to pick subchannel", "file": "src/core/ext/filters/client_channel/client_channel.cc",
"file_line": 3008, "referenced_errors": [{"created": "01642489297.482065891", "description
": "failed to connect to all addresses", "file": "src/core/ext/filters/client_channel/lb
_policy/pick_first/pick_first.cc", "file_line": 397, "grpc_status": 14}]]>
cloud@cloud:~/FL_project/yaml$

cloud@cloud:~/FL_project/yaml
server_message = receive()
File "/usr/local/lib/python3.6/dist-packages/flwr/client/grpc_client/connection.p
y", line 60, in <lambda>
receive: Callable[[ServerMessage]] = lambda: next(server_message_iterator)
File "/usr/local/lib/python3.6/dist-packages/grpc_channel.py", line 426, in _ne
xt
return self._next()
File "/usr/local/lib/python3.6/dist-packages/grpc_channel.py", line 809, in _next
raise self
grpc_channel._MultiThreadedRendezvous: <MultiThreadedRendezvous of RPC that termi
nated with:
status = StatusCode.UNAVAILABLE
details = "failed to connect to all addresses"
debug_error_string = "[{"created": "01642488872.535399829", "description": "Fai
led to pick subchannel", "file": "src/core/ext/filters/client_channel/client_channel.
cc", "file_line": 3008, "referenced_errors": [{"created": "01642488872.535397708", "desc
ription": "failed to connect to all addresses", "file": "src/core/ext/filters/client_ch
annel/lb_policy/pick_first/pick_first.cc", "file_line": 397, "grpc_status": 14}]]>
cloud@cloud:~/FL_project/yaml$
cloud@cloud:~/FL_project/yaml$
```


Try Edge 1 & Edge 2 has the same amount of training data samples. HPA applied on Edge 2.

Problem 1: Incorrect pod which chosen to train

```
cloud@cloud:~$ kubectl top pod
W0118 00:08:14.522188 32135 top_pod.go:140] Using json format to get metrics
. Next release will switch to protocol-buffers, switch early by passing --use-
protocol-buffers flag
NAME                                CPU(cores)   MEMORY(bytes)
fashion-mnist-69c89966bb-t27rp      1m           444Mi
fashion-mnist-edge1-7b5dcf65f4-fqpww 3m           261Mi
fashion-mnist-edge2-5d69f9b795-mdj5c 988m         517Mi
fashion-mnist-edge2-5d69f9b795-qlf74 2m           562Mi
fashion-mnist-edge2-5d69f9b795-vl2lx 992m         652Mi
fashion-mnist-edge2-5d69f9b795-wvpbx 2m           292Mi
```

Picture 1: HPA scaling up and 2 FL training pods are running on the same node (Edge 2)

```
cloud@cloud:~$ kubectl top pod
W0118 00:17:29.237209 44687 top_pod.go:140] Using json format to get metr
ics. Next release will switch to protocol-buffers, switch early by passing
--use-protocol-buffers flag
NAME                                CPU(cores)   MEMORY(bytes)
fashion-mnist-69c89966bb-t27rp      1m           618Mi
fashion-mnist-edge1-7b5dcf65f4-fqpww 1908m        870Mi
fashion-mnist-edge2-5d69f9b795-crdfz 2m           261Mi
fashion-mnist-edge2-5d69f9b795-mcc9m 2m           268Mi
fashion-mnist-edge2-5d69f9b795-qlf74 1909m        850Mi
fashion-mnist-edge2-5d69f9b795-vl2lx 2m           690Mi
```

Picture 2: HPA scaling up and FL pod is running on the different node -> this is correct.

```
cloud@cloud:~$ kubectl top pod
W0118 00:28:37.982169 59401 top_pod.go:140] Using json format to get metr
ics. Next release will switch to protocol-buffers, switch early by passing
--use-protocol-buffers flag
NAME                                CPU(cores)   MEMORY(bytes)
fashion-mnist-69c89966bb-t27rp      1m           329Mi
fashion-mnist-edge1-7b5dcf65f4-fqpww 2m           261Mi
fashion-mnist-edge2-5d69f9b795-qlf74 993m         528Mi
fashion-mnist-edge2-5d69f9b795-vl2lx 990m         539Mi
cloud@cloud:~$ kubectl top node
W0118 00:28:49.256901 59642 top_node.go:119] Using json format to get met
rics. Next release will switch to protocol-buffers, switch early by passing
--use-protocol-buffers flag
NAME      CPU(cores)   CPU%   MEMORY(bytes)   MEMORY%
cloud     245m         3%     3079Mi           40%
edge1     14m          0%     1314Mi           16%
edge2     1998m        99%     2141Mi           27%
```

Picture 3: HPA scaling down and FL pods are running on the same node (Edge 2)

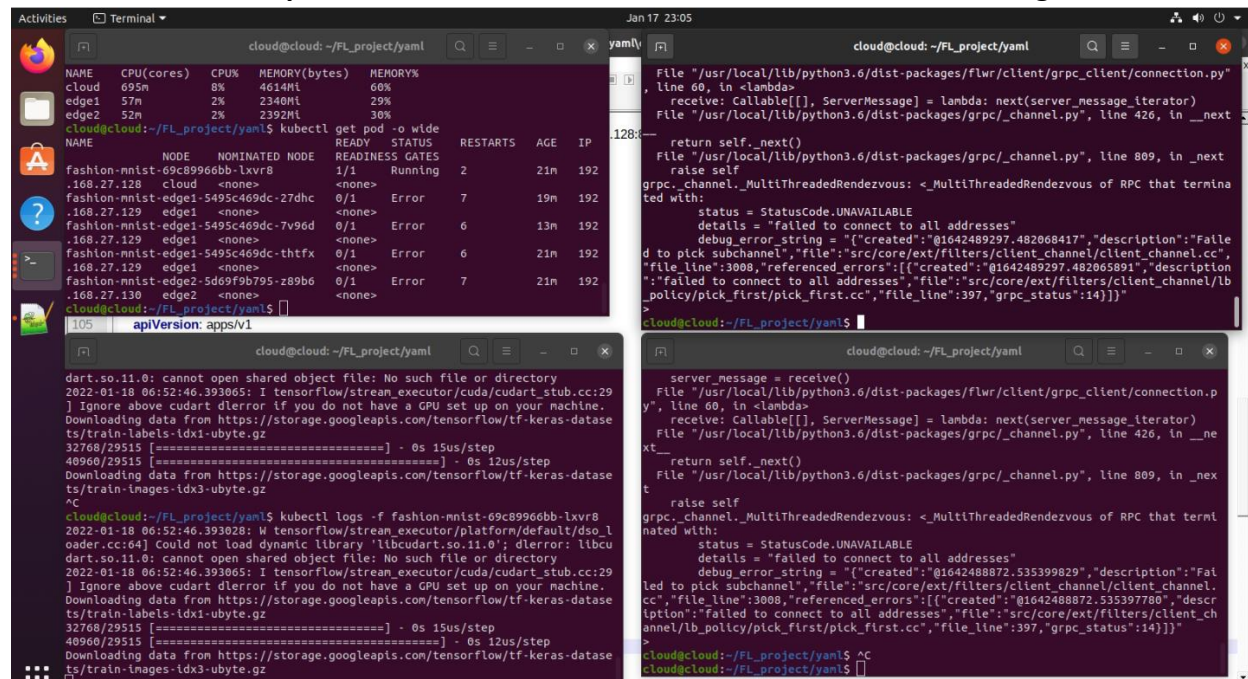
Why?

Because I set `min_num_client = 2` so when the HPA scale up & scale down the number of pod, the FL application will understand that each pod will represent for each client, they just only choose at least 1 pair of pod and start training, the rest pods will wait or training in the next round.

This lead to the problem that FL_application may choose 2 pods in the same node like Picture 1 or different node like Picture 2.

=> Solution: Not yet.

Problem 2: Failed to pick subchannel/Failed to connect to all addresses on edge nodes



The screenshot shows three terminal windows. The top-left window displays the output of `kubectl get pod -o wide`, showing pods for `fashion-mnist` on nodes `cloud` and `edge1`. The top-right window shows application logs with a `grpc_channel.MuliThreadedRendezvous` error: "failed to connect to all addresses". The bottom window shows logs from `kubectl logs -f fashion-mnist-69c89966bb-lxvr8`, indicating a failure to load the dynamic library `libcudart.so.11.0` and a warning about not having a GPU set up.

Why?

Because Cloud pod is still not ready state(Cloud pod still didn't finish the previous training loop but some pods in Edge 2 (HPA Pods) have already tried to connect => lead to failed to connect to all address (Cloud IP address).

Check logs on cloud: It was stuck at this step.



The screenshot shows a terminal window with the command `kubectl logs -f fashion-mnist-69c89966bb-t27rp`. The logs show a warning about not loading the dynamic library `libcudart.so.11.0` and a message about downloading data from `https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz`.

Cloud logs at ready state should be:

```
cloud@cloud:~/FL_project/yanl$ kubectl logs -f fashion-mnist-69c89966bb-kxd54
2022-01-18 11:22:29.073117: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'libcudart.so.11.0'; dLError: libcudart.so.11.0
: cannot open shared object file: No such file or directory
2022-01-18 11:22:29.073154: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dLError if you do not have a GPU set up on your machine.
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-labels-idx1-ubyte.gz
32768/29515 [=====] - 0s 2us/step
40960/29515 [=====] - 0s 2us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/train-images-idx3-ubyte.gz
26427392/26421880 [=====] - 3s 0us/step
26435584/26421880 [=====] - 3s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-labels-idx1-ubyte.gz
16384/5148 [=====] - 0s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/t10k-images-idx3-ubyte.gz
4423680/4422102 [=====] - 1s 0us/step
4431872/4422102 [=====] - 1s 0us/step
INFO flower 2022-01-18 11:22:35.164 | app.py:80 | Flower server running (Insecure, 5 rounds)
INFO flower 2022-01-18 11:22:35.164 | server.py:118 | Initializing global parameters
INFO flower 2022-01-18 11:22:35.164 | server.py:304 | Requesting initial parameters from one random client
```

Check HPA on both edge 1 and edge 2, the same training samples = 20K. (Below picture)

At first round, Cloud node received 3 parameters from clients, 1 failures, total 8 client (8 pods) are training, 4 pods for each node.

At second round, Cloud node received 4 parameters from clients, 0 failures, total 8 client (8 pods) are training, 4 pods for each node.

```
x_test shape: (10000, 28, 28)
y_test shape: (10000,)
5000/5000 [=====] - 10s 2ms/step - loss: 2.3161 - accuracy: 0.0376
INFO flower 2022-01-18 11:32:47.595 | server.py:127 | initial parameters (loss, other metrics): 2.316058397293091, {'accuracy': 0.03759999945759773}
INFO flower 2022-01-18 11:32:47.596 | server.py:133 | FL starting
DEBUG flower 2022-01-18 11:32:47.596 | server.py:255 | fit_round: strategy sampled 2 clients (out of 4)
DEBUG flower 2022-01-18 11:33:32.148 | server.py:264 | fit_round received 2 results and 0 failures

DEPRECATION WARNING: deprecated 'eval_fn' return format

    loss, accuracy

move to

    loss, {"accuracy": accuracy}

instead. Note that compatibility with the deprecated return format will be
removed in a future release.

5000/5000 [=====] - 9s 2ms/step - loss: 0.3304 - accuracy: 0.8819
INFO flower 2022-01-18 11:33:41.581 | server.py:154 | fit progress: (1, 0.3304124176502228, {'accuracy': 0.8819000124931335}, 53.985757448001095)
INFO flower 2022-01-18 11:33:41.582 | server.py:199 | evaluate_round: no clients selected, cancel
DEBUG flower 2022-01-18 11:33:41.582 | server.py:255 | fit_round: strategy sampled 4 clients (out of 8)
DEBUG flower 2022-01-18 11:35:05.551 | server.py:264 | fit_round received 3 results and 1 failures

DEPRECATION WARNING: deprecated 'eval_fn' return format

    loss, accuracy

move to

    loss, {"accuracy": accuracy}

instead. Note that compatibility with the deprecated return format will be
removed in a future release.

5000/5000 [=====] - 9s 2ms/step - loss: 0.2717 - accuracy: 0.9038
INFO flower 2022-01-18 11:35:15.064 | server.py:154 | fit progress: (2, 0.2716974914073944, {'accuracy': 0.9038000106811523}, 147.46861769499992)
INFO flower 2022-01-18 11:35:15.064 | server.py:199 | evaluate_round: no clients selected, cancel
DEBUG flower 2022-01-18 11:35:15.064 | server.py:255 | fit_round: strategy sampled 4 clients (out of 8)
DEBUG flower 2022-01-18 11:36:37.739 | server.py:264 | fit_round received 4 results and 0 failures

DEPRECATION WARNING: deprecated 'eval_fn' return format

    loss, accuracy
```

StatefulSet:

```
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP            NODE   NOMINATED NODE   READINESS GATES
fashion-mnist-7476c566c6-zj5gm      1/1     Running   21          163m  192.168.27.128  cloud  <none>           <none>
fashion-mnist-edge1-594865456c-ct9bf 1/1     Running   4           10m   192.168.27.129  edge1  <none>           <none>
fashion-mnist-edge1-594865456c-dtftf 1/1     Running   4           10m   192.168.27.129  edge1  <none>           <none>
fashion-mnist-edge2-0                1/1     Running   21          163m  192.168.27.130  edge2  <none>           <none>
fashion-mnist-edge2-1                1/1     Running   4           10m   192.168.27.130  edge2  <none>           <none>
```

Advantage: using StatefulSet can keep the original pod when applying HPA because the StatefulSet will terminate pod by order from biggest to smallest during the scale down process. (The original pod won't be terminated by HPA because minReplicas is 1).

Disadvantage: if the original pod (fashion-mnist-edge1-0 || fashion-mnist-edge2-0) are in error state and they can't be running/ready state by themselves, the system will be shutdown. In below picture, they meet **the problem 2** during training and the system can't be fixed by itself.

```
cloud@cloud:~/FL_project/yaml$ kubectl get pod -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP            NODE   NOMINATED NODE   READINESS GATES
fashion-mnist-7476c566c6-x4brh      1/1     Running   67          10h   192.168.27.128  cloud  <none>           <none>
fashion-mnist-edge1-0                0/1     Error     78          10h   192.168.27.129  edge1  <none>           <none>
fashion-mnist-edge2-0                0/1     Error     77          10h   192.168.27.130  edge2  <none>           <none>
```

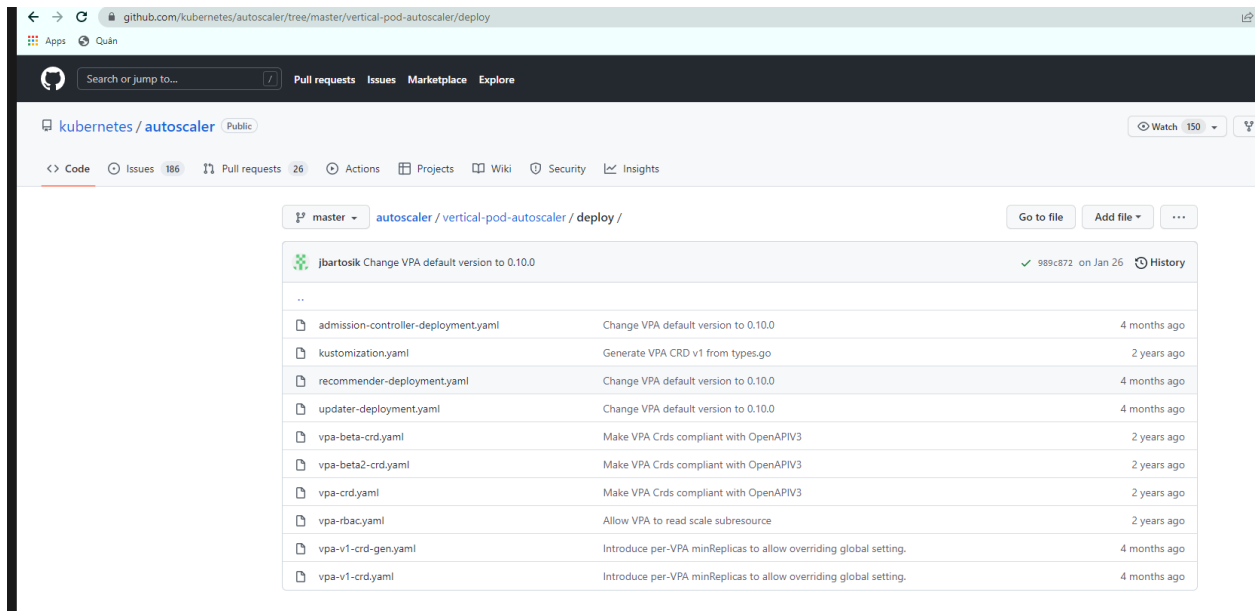
```
grpc_channel_MultiThreadedRendezvous: <MultiThreadedRendezvous of RPC that terminated with:
  status = StatusCode.UNAVAILABLE
  details = "failed to connect to all addresses"
  debug_error_string = "[{"created":"@1644196050.393040682","description":"Failed to pick subchannel","file":"src/core/ext/filters/client_channel/client_channel.cc","file_line":3008,"referenced_errors":[{"created":"@1644196050.393038347","description":"failed to connect to all addresses","file":"src/core/ext/filters/client_channel/lb_policy/pick_first/pick_first.cc","file_line":397,"grpc_status":14}]]"
```

Below is 3 pods on each node, no autoscale, min_num_client = 6.

```
cloud@cloud:~/FL_project/yaml$ kubectl top pod
W0216 23:26:39.079657 2385976 top_pod.go:140] Using json format to get metrics. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME                                CPU(cores)   MEMORY(bytes)
fashion-mnist-cloud-0               1m           406Mi
fashion-mnist-edge1-0               2005m        1183Mi
fashion-mnist-edge1-1               2030m        1139Mi
fashion-mnist-edge1-2               2088m        1123Mi
fashion-mnist-edge2-0               2011m        1100Mi
fashion-mnist-edge2-1               2032m        1087Mi
fashion-mnist-edge2-2               2053m        1114Mi
```

```
cloud@cloud:~/FL_project/yaml$ kubectl top node
W0216 23:27:25.059295 2386929 top_node.go:119] Using json format to get metrics. Next release will switch to protocol-buffers, switch early by passing --use-protocol-buffers flag
NAME      CPU(cores)   CPU%   MEMORY(bytes)   MEMORY%
cloud     1950m        24%    4161Mi          53%
edge1     6111m        76%    5274Mi          67%
edge2     6270m        78%    5179Mi          66%
```

Install VPA:



The screenshot shows the GitHub repository for Kubernetes autoscaler, specifically the `vertical-pod-autoscaler/deploy` directory. The repository is public and has 186 issues, 26 pull requests, and 150 watchers. The current branch is `master`. The directory contains several YAML files for deploying VPA components.

Commit: jbartosik Change VPA default version to 0.10.0 (989c872 on Jan 26)

File	Description	Time
admission-controller-deployment.yaml	Change VPA default version to 0.10.0	4 months ago
kustomization.yaml	Generate VPA CRD v1 from types.go	2 years ago
recommender-deployment.yaml	Change VPA default version to 0.10.0	4 months ago
updater-deployment.yaml	Change VPA default version to 0.10.0	4 months ago
vpa-beta-crd.yaml	Make VPA Crds compliant with OpenAPIV3	2 years ago
vpa-beta2-crd.yaml	Make VPA Crds compliant with OpenAPIV3	2 years ago
vpa-crd.yaml	Make VPA Crds compliant with OpenAPIV3	2 years ago
vpa-rbac.yaml	Allow VPA to read scale subresource	2 years ago
vpa-v1-crd-gen.yaml	Introduce per-VPA minReplicas to allow overriding global setting.	4 months ago
vpa-v1-crd.yaml	Introduce per-VPA minReplicas to allow overriding global setting.	4 months ago

Change nodeName: cloud