**FATE的部署及测试**

## 云上创建虚拟机

-选择在阿里云上面开发 选择8vCPU+64GB内存+Ubuntu20.04+200G硬盘 各自两台

## 登录到实例上

-在阿里云登录到各自两台机器的后端，分别是一号机172.28.12.233和二号机172.28.12.232

## 若在VMware上部署的补充

由于阿里云已经在镜像系统中更新到阿里云的源和ssh和vim等命令，所以有些操作可以不用做，以以下操作为主即可。但是由于在VMware上部署，内存可能不足，由此会引发一系列不可见的问题

如果是在自己VMware的虚拟机，则需要通过以下命令：

-首先安装vim，方便可视化及修改

apt-get install vim\_

-修改阿里云源（可选）

vi /etc/apt/sources.list

把原有的显色句子都都用#注释掉

然后在最后加入以下十条句子

deb http://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ bionic-proposed main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic-proposed main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse

然后终端上输入更新源  
apt-get update

apt-get upgrade

-安装ssh服务

apt-get install openssh-server

ps -e | grep ssh

如果返回以下结果及服务启动成功

文本

描述已自动生成

## 免密互通

-完成两台机器的免密互通 阿里云一个账户创建的实例可以相互通信，只要把密码填写好嘞就可以互通，但是实际上我们之后创建集群，需要各台机器能够免密互通，这样就不用每次都去输入密码

在一号机172.28.12.233的机器上实行，将秘钥发送给二号机172.28.12.232

ssh-keygen -t rsa  
cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys

chmod 600 ~/.ssh/authorized\_keys

service sshd restart

ssh-copy-id -i ~/.ssh/id\_rsa.pub root@ 172.28.12.232

-同样的操作发生在二号机上，只是最后的一句改成

ssh-copy-id -i ~/.ssh/id\_rsa.pub root@ 172.28.12.233

-如果发生密码多次登入也进不去，需要进行配置修改  
vim /etc/ssh/sshd\_config

找到对应的下面三项，进行取消注释和修改为yes

PermitRootLogin yes

PasswordAuthentication yes

ChallengeResponseAuthentication yes

## 安装docker

sudo apt-get update

sudo apt-get install \

apt-transport-https \

ca-certificates \

curl \

gnupg-agent \

software-properties-common

curl -fsSL http://mirrors.aliyun.com/docker-ce/linux/ubuntu/gpg | sudo apt-key add

sudo apt-key fingerprint 0EBFCD88

sudo add-apt-repository \

"deb [arch=amd64] http://mirrors.aliyun.com/docker-ce/linux/ubuntu \

$(lsb\_release -cs) \

stable"

sudo apt-get install docker-ce docker-ce-cli containerd.io

-验证运行helloword测试，会返回结果如下图

docker run hello-world

文本

描述已自动生成

## 安装docker-compose

sudo curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

chmod +x /usr/local/bin/docker-compose

-运行进行验证

文本

描述已自动生成

## 安装FATE的镜像文件

wget <https://webank-ai-1251170195.cos.ap-guangzhou.myqcloud.com/fate_1.4.2-images.tar.gz>

docker load -i fate\_1.4.2-images.tar.gz

docker images

-最后一步是验证fate的docker是否已经完全加载下来

图形用户界面, 文本, 聊天或短信

描述已自动生成

## 安装kubefate-docker-compose

wget https://github.com/FederatedAI/KubeFATE/releases/download/v1.4.2/kubefate-docker-compose.tar.gz

tar -zxvf kubefate-docker-compose.tar.gz

## 部署构造集群

-只在一号机操作即可修改ip地址，一般谁发起任务谁就是guest端，另外一个是host端记住user从fate改完root

cd docker-deploy

vim parties.con

bash generate\_config.sh

bash docker\_deploy.sh all --training

bash docker\_deploy.sh all --serving

文本

描述已自动生成

-查看部署情况

docker ps

图形用户界面, 文本

描述已自动生成

## FATE Board的设置

-允许公网上部分IP地址访问您的网站，只要在阿里云入方向上面增加安全组规则就可以，从而让本机的浏览器能够访问到云上虚拟机（公网ip）的端口，开放8080端口和8059

图形用户界面, 应用程序

描述已自动生成

## 测试集群部署和在线推理功能

-集群部署的验证，如果跑完了，可以在fateboard上查看获取情况

docker exec -it confs-10000\_python\_1 bash

cd /data/projects/fate/python/examples/toy\_example

python run\_toy\_example.py 10000 9999 1

文本

描述已自动生成

图形用户界面, 文本, 应用程序

描述已自动生成

-在线推理serving 的验证如下

一号机上执行

docker exec -it confs-10000\_python\_1 bash

cd fate\_flow

vi examples/upload\_host.json //修改参数 如下图

python fate\_flow\_client.py -f upload -c examples/upload\_host.json //上传数据

文本

描述已自动生成

二号机执行如下

docker exec -it confs-9999\_python\_1 bash

cd fate\_flow

vi examples/upload\_guest.json //修改如上

进去把“name:breast-xxxxxx”改成"name": "breast"，然后上传数据

python fate\_flow\_client.py -f upload -c examples/upload\_guest.json

vi examples/test\_hetero\_lr\_job\_conf.json //修参

python fate\_flow\_client.py -f submit\_job -d examples/test\_hetero\_lr\_job\_dsl.json -c examples/test\_hetero\_lr\_job\_conf.json//提交任务

图形用户界面, 应用程序, Teams

描述已自动生成

-查看训练任务状态，记得复制上面的"model\_id"和"model\_version"

python fate\_flow\_client.py -f query\_task -j 202011281703531795652 | grep f\_status

-修改加载模型的配置，将"model\_id"和"model\_version"进行替换，还有相关的guest替换成9999，如图

vi examples/publish\_load\_model.json

文本

描述已自动生成

-加载模型，将修改的模型公开出来

python fate\_flow\_client.py -f load -c examples/publish\_load\_model.json

-修改绑定模型的配置

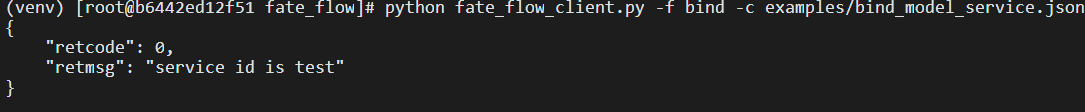
vi examples/bind\_model\_service.json

**文本

描述已自动生成**

-绑定模型

python fate\_flow\_client.py -f bind -c examples/bind\_model\_service.json



-绑定之后我们就可以在线测试

curl -X POST -H 'Content-Type: application/json' -i 'http://8.135.100.181:8059/federation/v1/inference' --data '{

"head": {

"serviceId": "test"

},

"body": {

"featureData": {

"x0": 0.254879,

"x1": -1.046633,

"x2": 0.209656,

"x3": 0.074214,

"x4": -0.441366,

"x5": -0.377645,

"x6": -0.485934,

"x7": 0.347072,

"x8": -0.287570,

"x9": -0.733474

},

"sendToRemoteFeatureData": {

"id": "123"

}

}

}'

得到返回结果如下：

文本

描述已自动生成

或者可以将serving的可视化，我们选择在本机的Chrome浏览器上安装[RESTED](https://github.com/RESTEDClient/RESTED)的拓展程序，输入如下内容，得到返回的结果

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成