1. 第一步老生常谈的配置环境太简单了不多赘述了

在网页终端中配置环境：

2、重点来了，最下面输入，代码：python $GEMINI\_RUN/DogsVsCats.py --num\_epochs 5 --data\_dir $GEMINI\_DATA\_IN1/DogsVsCats/ --train\_dir $GEMINI\_DATA\_OUT

代码什么含义呢？

这个特定的命令调用了一个名为 DogsVsCats.py 的Python脚本，并且传递了一些参数给它。以下是每个部分的意义：

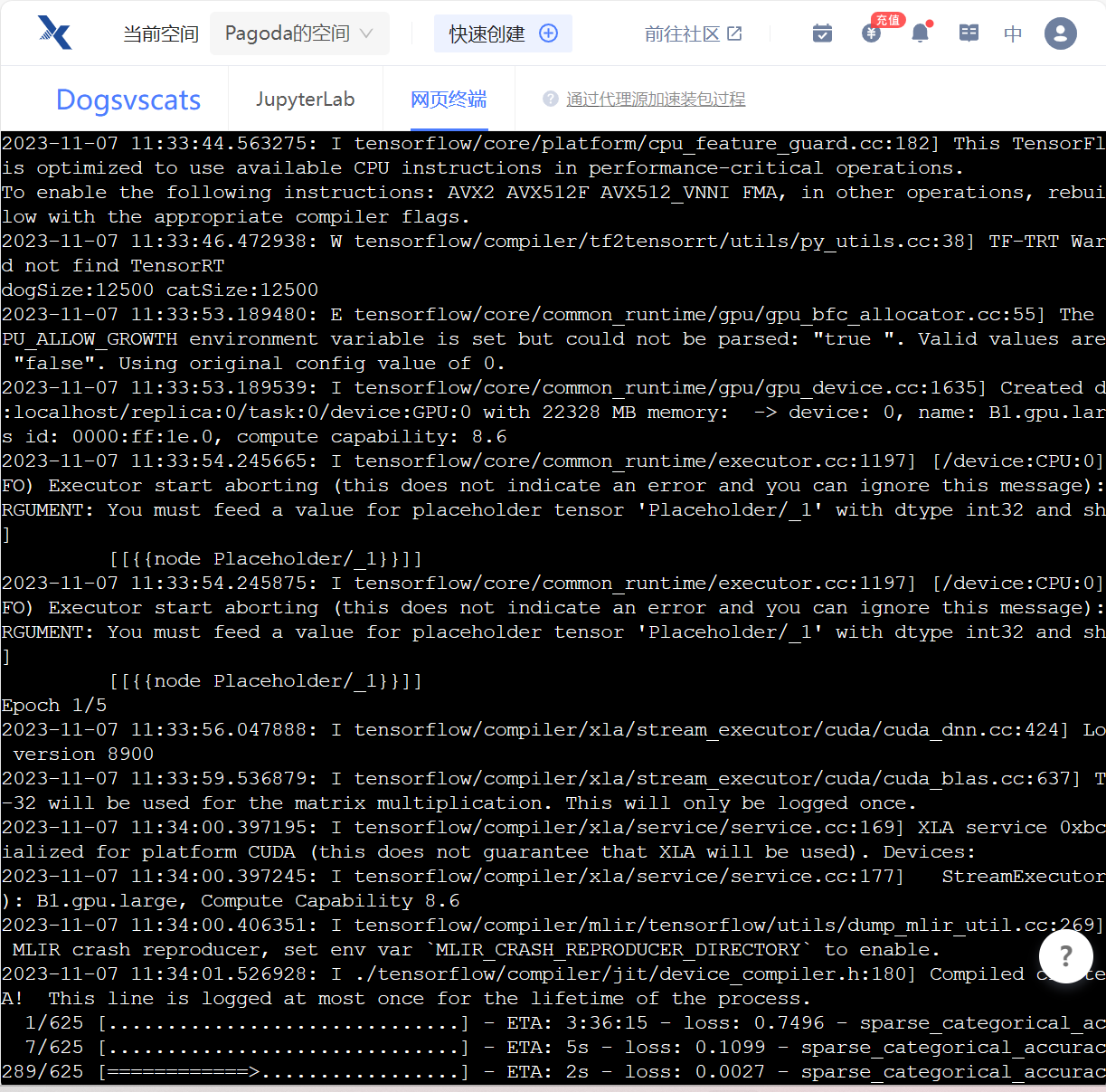
python: 这是调用Python解释器的命令，它用来执行 .py 文件中的Python代码。

$GEMINI\_RUN/DogsVsCats.py: 这是要执行的Python脚本的路径。$GEMINI\_RUN 可能是一个环境变量，指向存放脚本的目录。

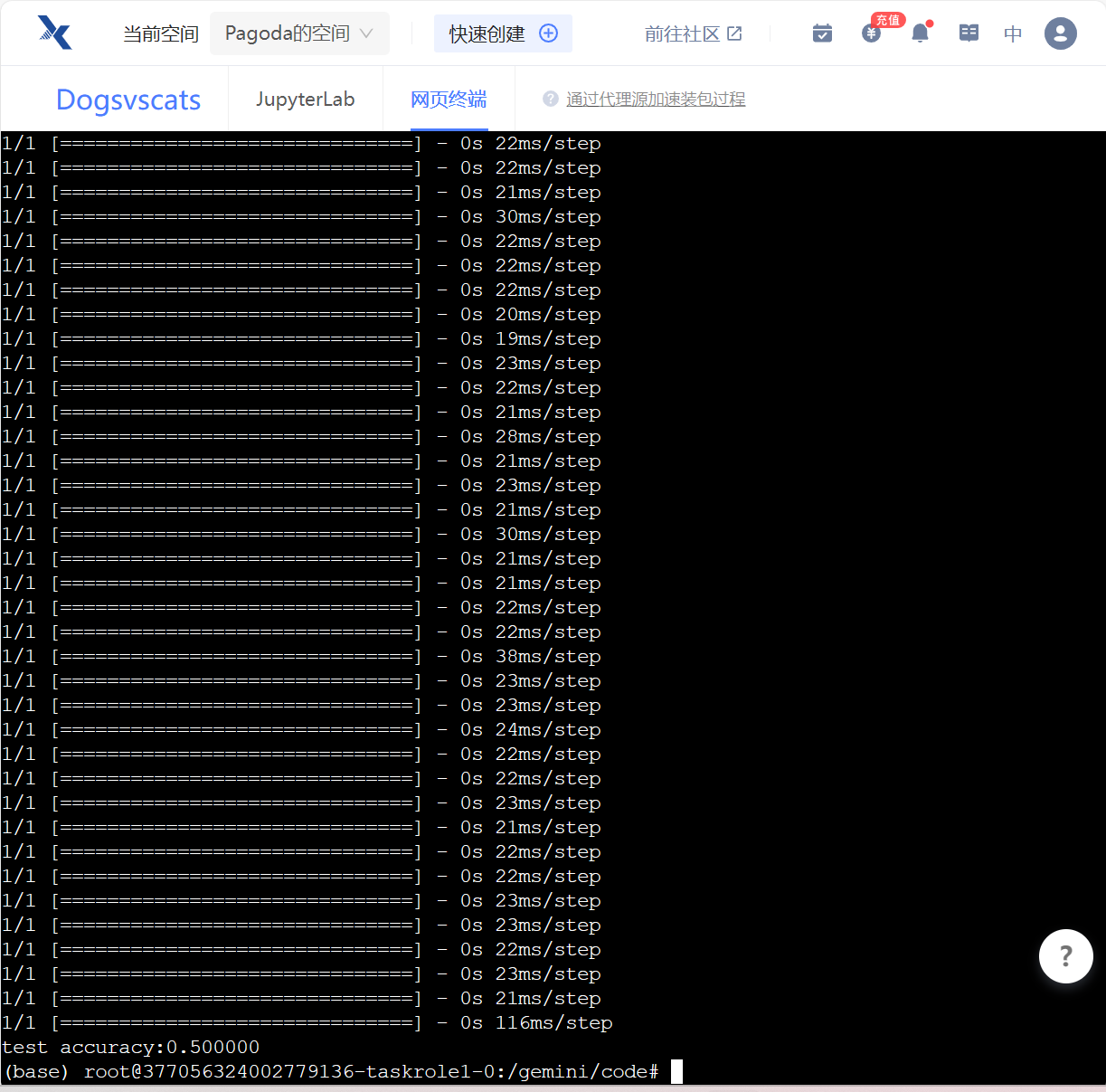
--num\_epochs 5: 这是一个命令行参数，--num\_epochs 表示要运行的训练周期数，在这里设置为5。训练周期是指在整个数据集上进行一次前向和后向传播的过程。

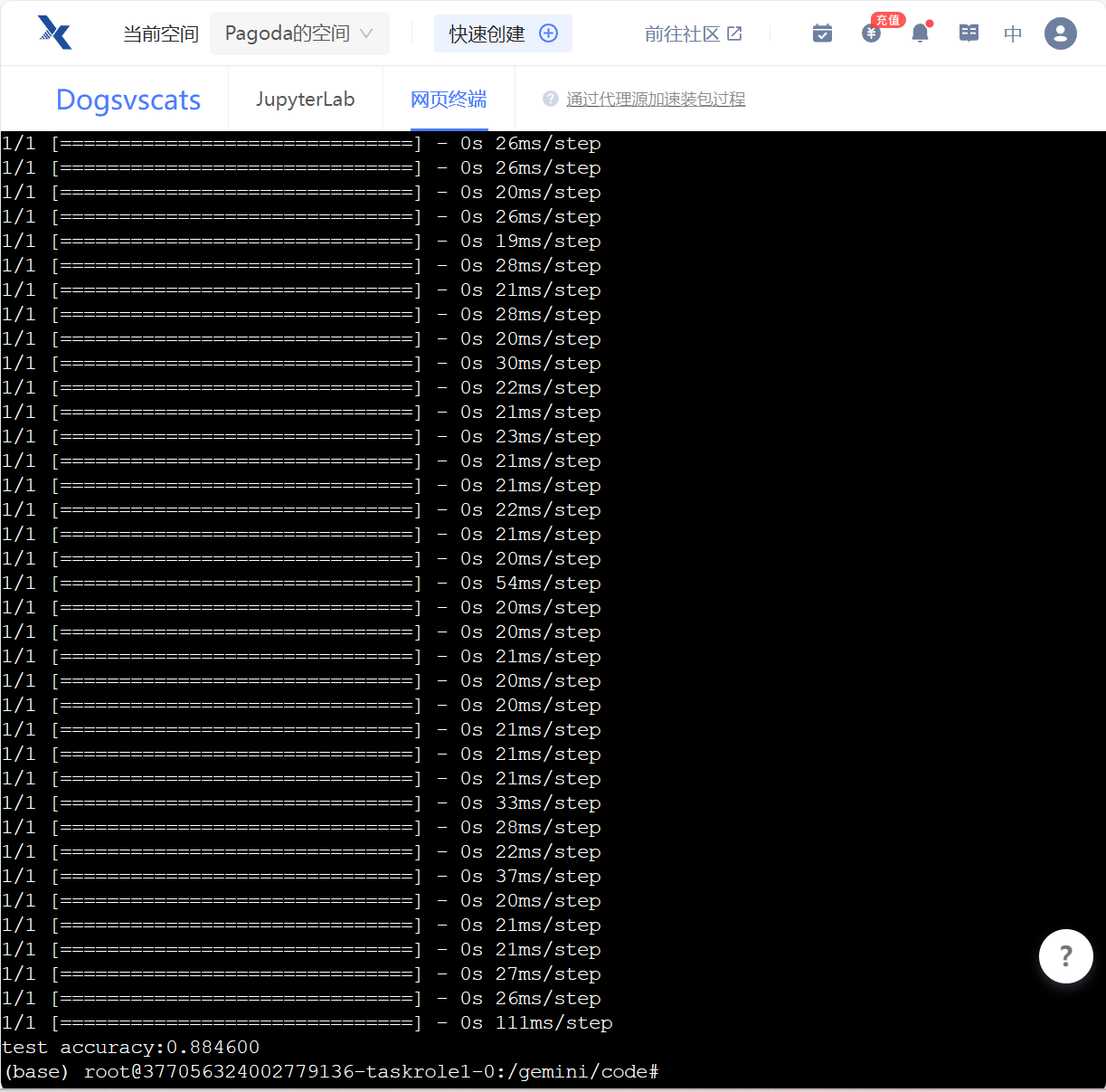
--data\_dir $GEMINI\_DATA\_IN1/DogsVsCats/: 这是另一个命令行参数，--data\_dir 指定了数据集的目录。$GEMINI\_DATA\_IN1 可能是另一个环境变量，指向存放输入数据的位置。

--train\_dir $GEMINI\_DATA\_OUT: 这个参数指定了训练过程中生成的数据（如模型权重、日志文件等）应该保存的目录。$GEMINI\_DATA\_OUT 可能是一个指向输出目录的环境变量。



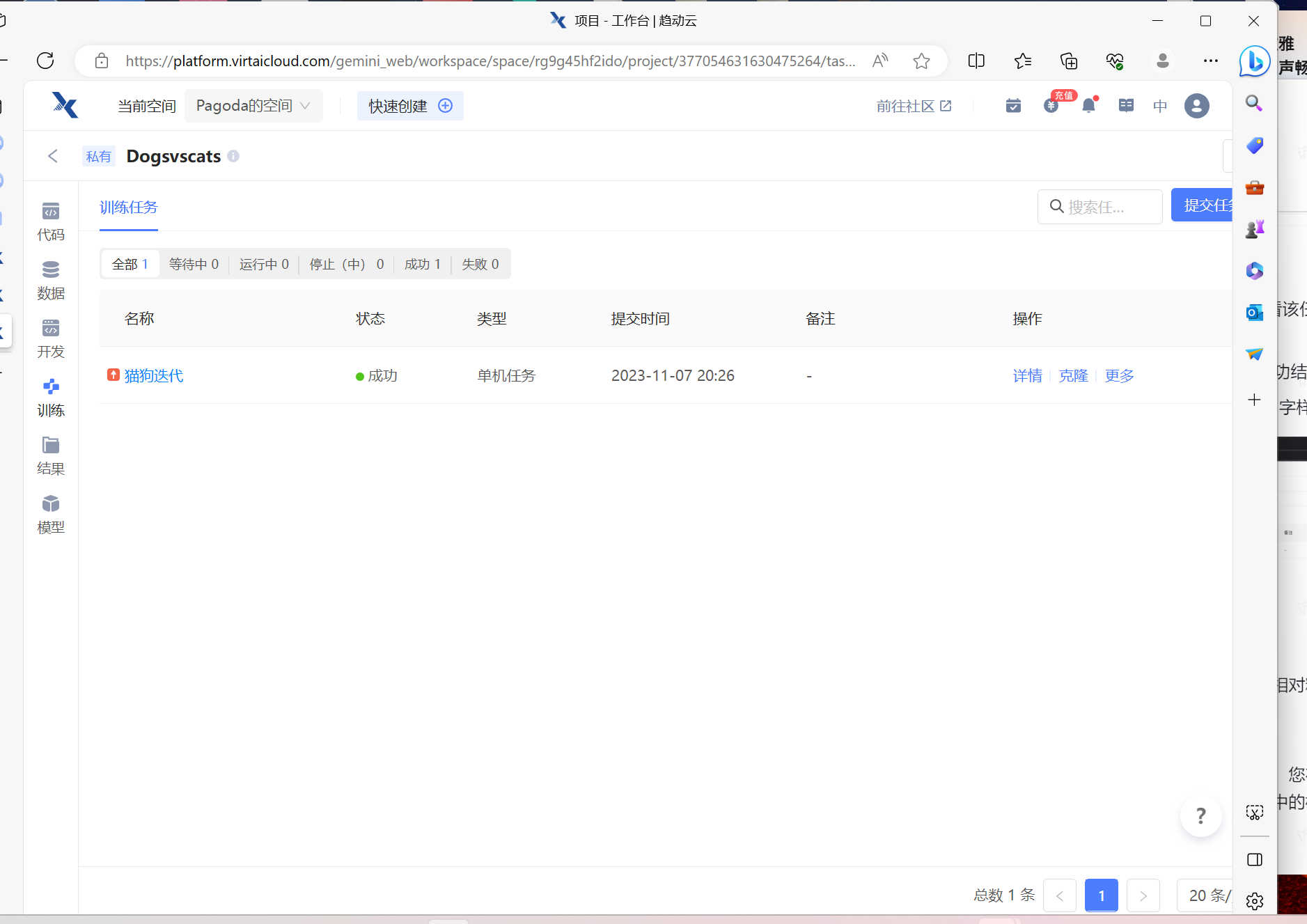
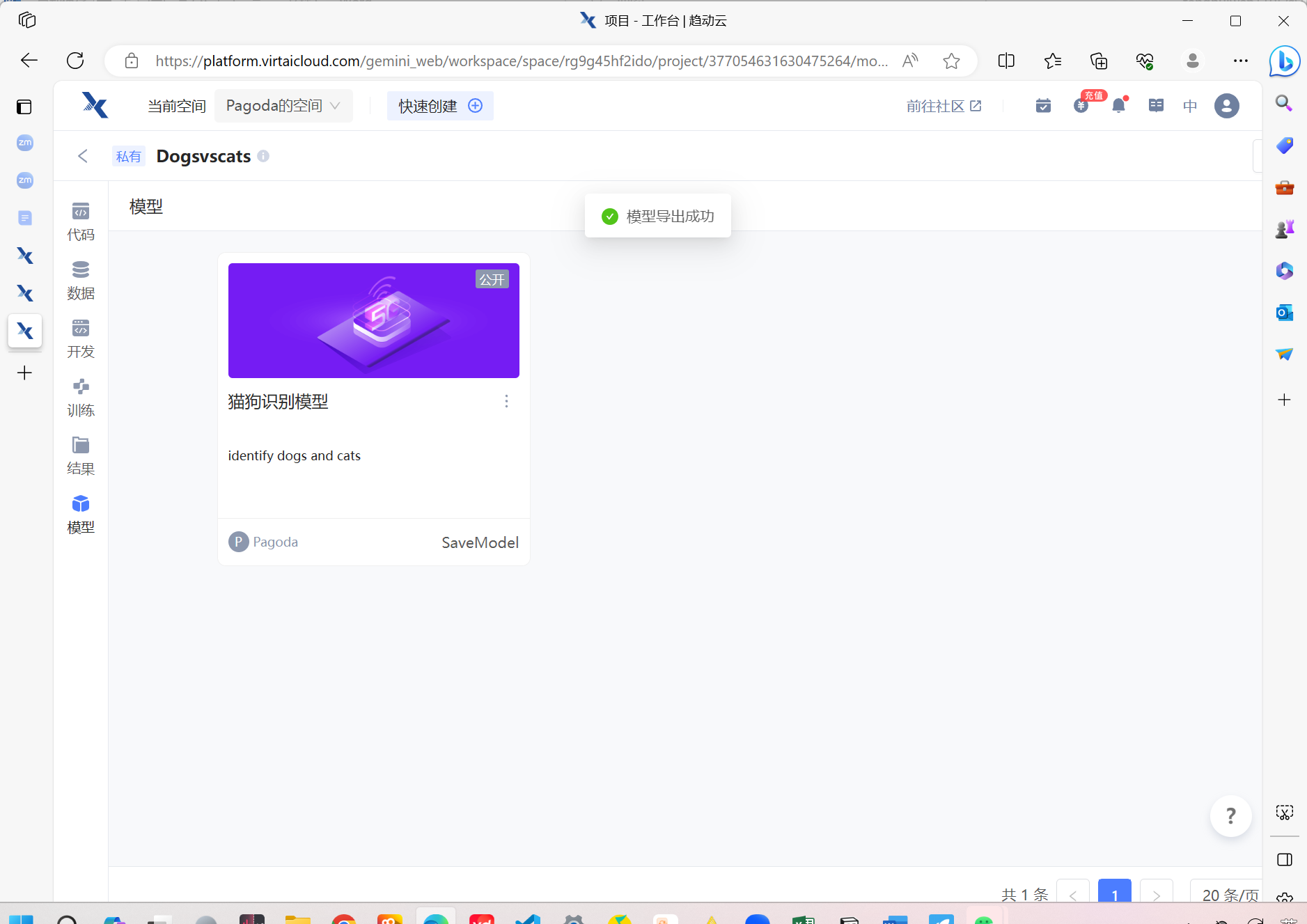
结果如上图。最终显示





train\_dataset = train\_dataset.shuffle(buffer\_size=20000) 多了这一行代码为啥准确度提升了很多。

保存好模型输出：

1. 提交离线文件
2. 新建代码名字xxx
3. 配置环境，输入启动代码
4. 镜像复制代码环境然后直接启动下面看结果。
5. 

这就是导出成功了！！