Training Machine Learning Models

In FlyElephant, you can train a model which was prepared in Jupyter Notebook very easily. It is just 4 steps. FlyElephant automates the training lifecycle, control all process, and deliver the result to you.

1. Prepare the model

Prepare the model in Jupyter Notebook and remember the folder in which it is located, as well as the name of the file.

Check the code in Notebook so that it doesn't have any functions that can be run only in Jupyter Notebook, for example magic commands. Jupyter Notebook file will be converted to python file, so it is necessary that all the code be compatible with Python 3. If such functions are, then you will get an error.

For example:

!pip3 install seaborn

can be change to

import os
os.system('pip3 install seaborn')

and

%matplotlib inline

can be change to

import matplotlib
matplotlib.use('Agg')

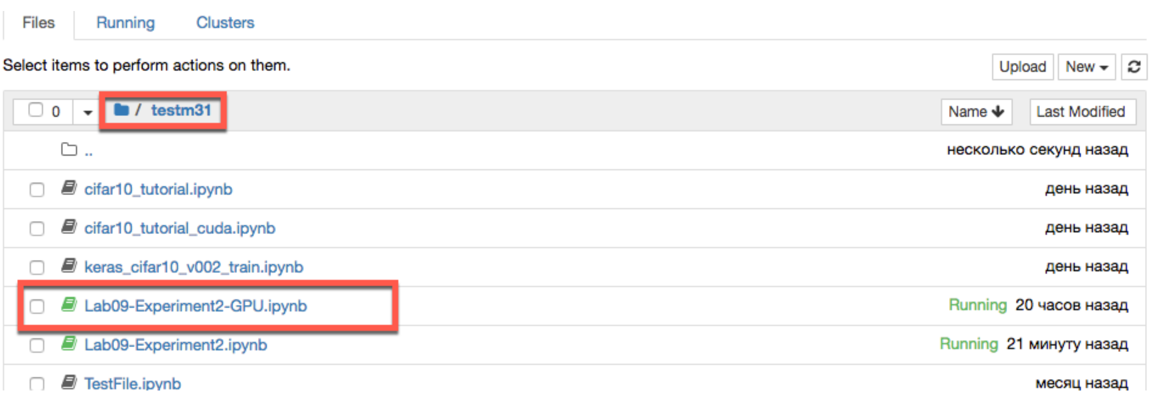
If you use data from a Datasets folder, change the relative path to full:

dataset_dir = '../Datasets/vgg_face_indian_dataset'

need change to

dataset_dir = '/notebooks/Datasets/vgg_face_indian_dataset'

When everything is ready, remember the directory and the name of the file.



Select items to perform actions on them.

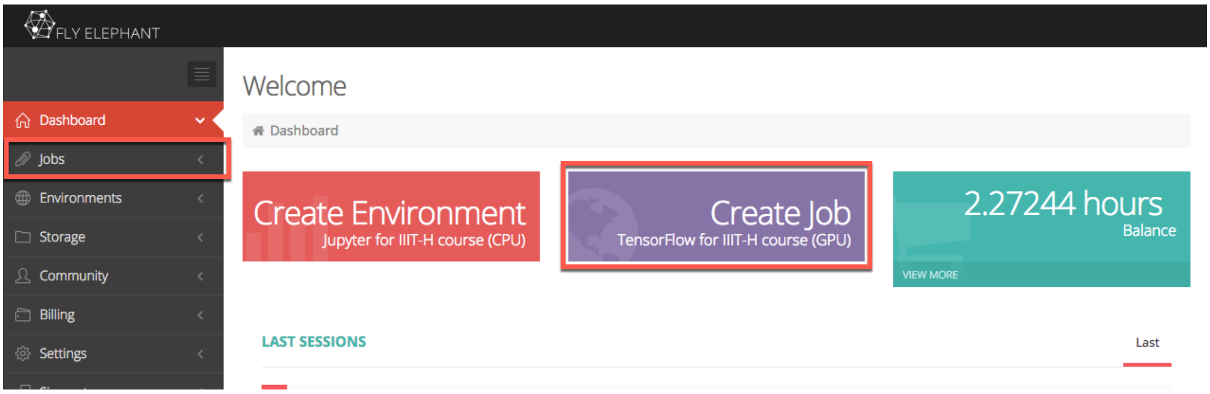
□ 0 / testm31

Name	Last Modified
..	несколько секунд назад
□ cifar10_tutorial.ipynb	день назад
□ cifar10_tutorial_cuda.ipynb	день назад
□ keras_cifar10_v002_train.ipynb	день назад
□ Lab09-Experiment2-GPU.ipynb	Running 20 часов назад
□ Lab09-Experiment2.ipynb	Running 21 минуту назад
□ TestFile.ipynb	месяц назад

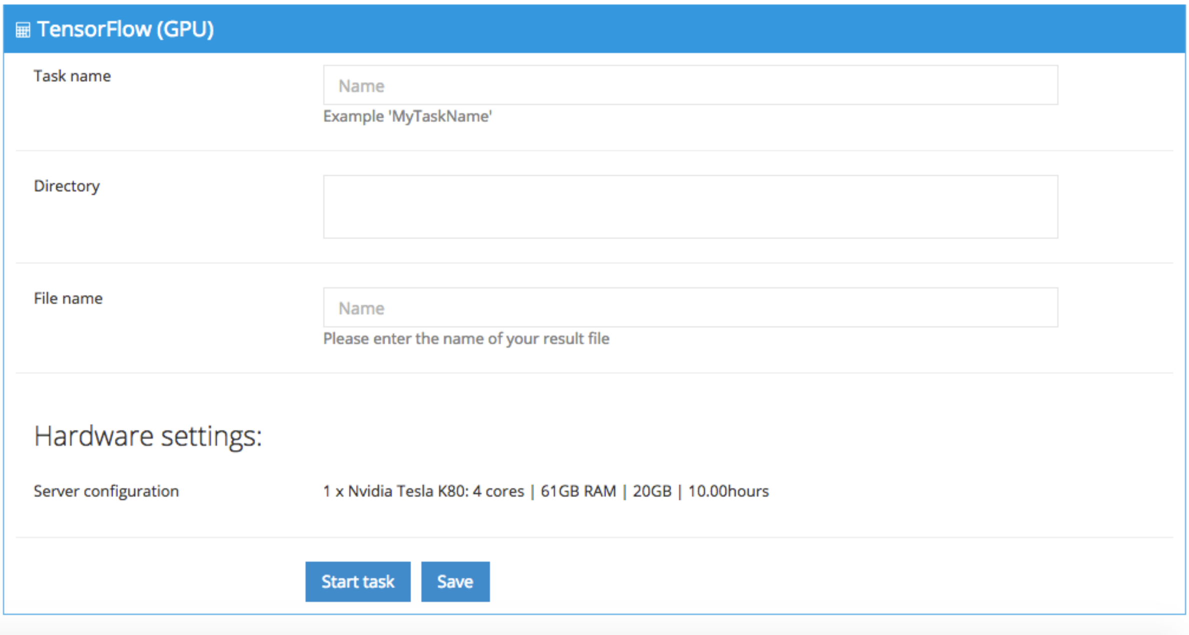
In our case this will be a folder - testm31 and a file - Lab09-Experiment2-GPU.ipynb.

2. Choose environment

We can click on the dashboard on the Create Job or through the menu go to Jobs - New Jobs - TensorFlow (GPU) - New Job.



3. Choose model and data



TensorFlow (GPU)

Task name	Name Example 'MyTaskName'
Directory	
File name	Name Please enter the name of your result file

Hardware settings:

Server configuration	1 x Nvidia Tesla K80: 4 cores | 61GB RAM | 20GB | 10.00hours

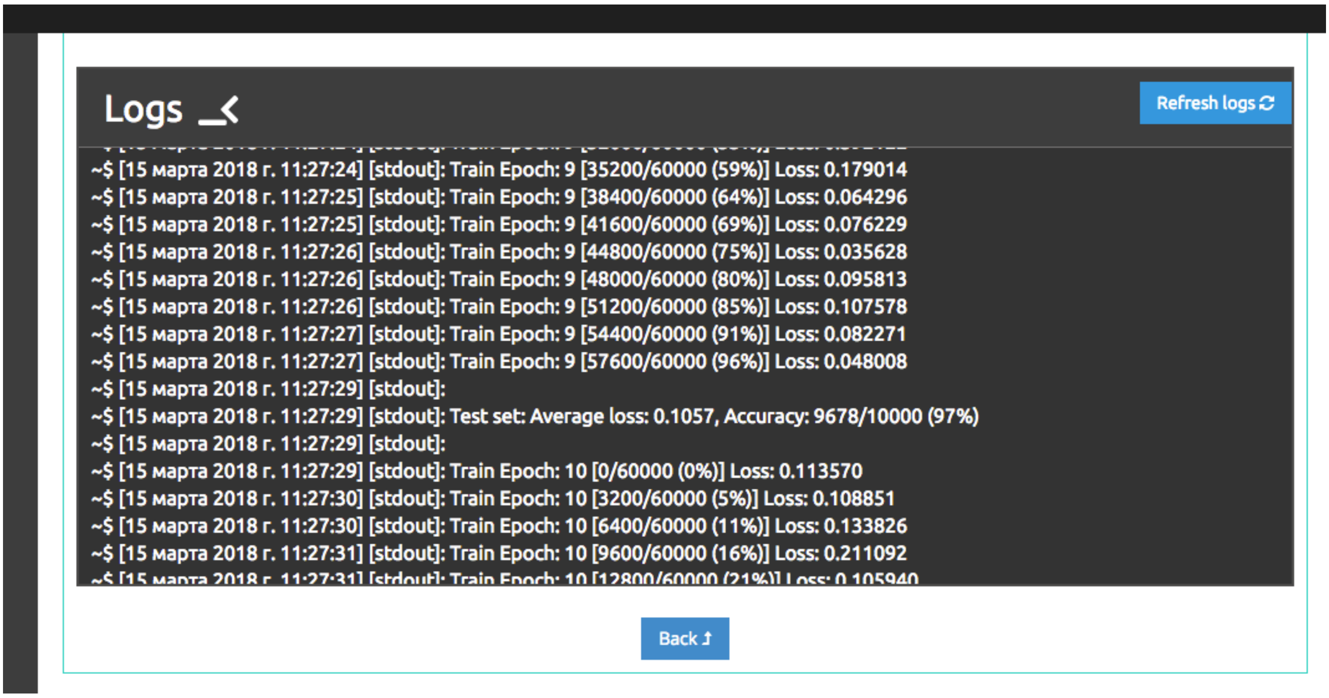
When the form is opened, you need to write the following information in it:

- Task name: Any task name
- Directory: Our directory. In our case this will be **testm31**.
- File name: Our Notebook file *without .ipynb*. In our case this will be **Lab09-Experiment2-GPU**

4. Start training model

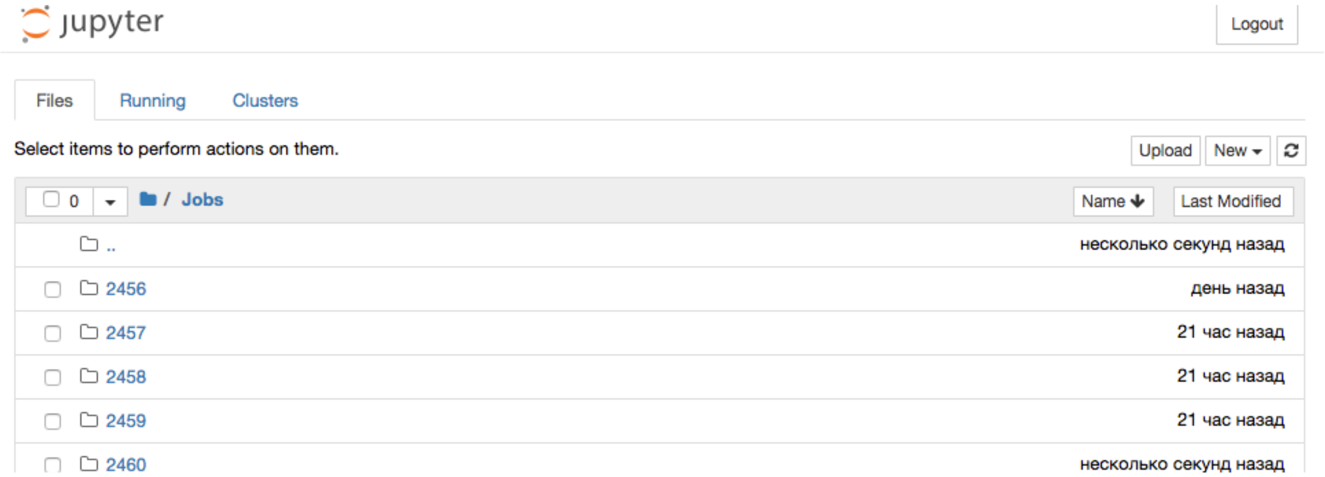
When everything is ready, click on the button Start task. FlyElephant will start the process. It will create a GPU instance, connect the user's storage, copy the necessary files, convert the file from *.ipynb to *.py, and launch it.

You can see all the details on the Job page. Also there will be all the logs. You need to press the button Refresh logs, to see the log update. Logs begin to appear only when the instance is running and the work of the file has already started. It takes a couple of minutes until everything starts.



```
~$ [15 марта 2018 г. 11:27:24] [stdout]: Train Epoch: 9 [35200/60000 (59%)] Loss: 0.179014
~$ [15 марта 2018 г. 11:27:25] [stdout]: Train Epoch: 9 [38400/60000 (64%)] Loss: 0.064296
~$ [15 марта 2018 г. 11:27:25] [stdout]: Train Epoch: 9 [41600/60000 (69%)] Loss: 0.076229
~$ [15 марта 2018 г. 11:27:26] [stdout]: Train Epoch: 9 [44800/60000 (75%)] Loss: 0.035628
~$ [15 марта 2018 г. 11:27:26] [stdout]: Train Epoch: 9 [48000/60000 (80%)] Loss: 0.095813
~$ [15 марта 2018 г. 11:27:26] [stdout]: Train Epoch: 9 [51200/60000 (85%)] Loss: 0.107578
~$ [15 марта 2018 г. 11:27:27] [stdout]: Train Epoch: 9 [54400/60000 (91%)] Loss: 0.082271
~$ [15 марта 2018 г. 11:27:27] [stdout]: Train Epoch: 9 [57600/60000 (96%)] Loss: 0.048008
~$ [15 марта 2018 г. 11:27:29] [stdout]:
~$ [15 марта 2018 г. 11:27:29] [stdout]: Test set: Average loss: 0.1057, Accuracy: 9678/10000 (97%)
~$ [15 марта 2018 г. 11:27:29] [stdout]:
~$ [15 марта 2018 г. 11:27:29] [stdout]: Train Epoch: 10 [0/60000 (0%)] Loss: 0.113570
~$ [15 марта 2018 г. 11:27:30] [stdout]: Train Epoch: 10 [3200/60000 (5%)] Loss: 0.108851
~$ [15 марта 2018 г. 11:27:30] [stdout]: Train Epoch: 10 [6400/60000 (11%)] Loss: 0.133826
~$ [15 марта 2018 г. 11:27:31] [stdout]: Train Epoch: 10 [9600/60000 (16%)] Loss: 0.211092
~$ [15 марта 2018 г. 11:27:31] [stdout]: Train Epoch: 10 [12800/60000 (21%)] Loss: 0.105940
```

When the training is finish and you need to access the files that have been saved, for example, the model, you can find it in the folder Job, where the folder name is the task id.



Select items to perform actions on them.

/ Jobs

```
..      несколько секунд назад
2456    день назад
2457    21 час назад
2458    21 час назад
2459    21 час назад
2460    несколько секунд назад
```

The best way to use this functionality is to prepare a model that you need to train in a separate file. Train model on the GPU, save the result in a file and then use this file on the CPU instance in Jupyter.