

## Part 2

OK. How to recognise a cat or dog? Easy - for the people but what the computer can do it? The easiest way is to use Google Photos (if you have photos with cats or dogs write cat or dog in search bar :) but we are developers - the easiest way isn't fun.

Let's do something which is not available straitly:

1. Open Google Cloud Shell
2. Go to folder with lab:  
`cd ~/gdgwarsaw2019/vision`
3. Run:  
`python visionapi.py`

More information:

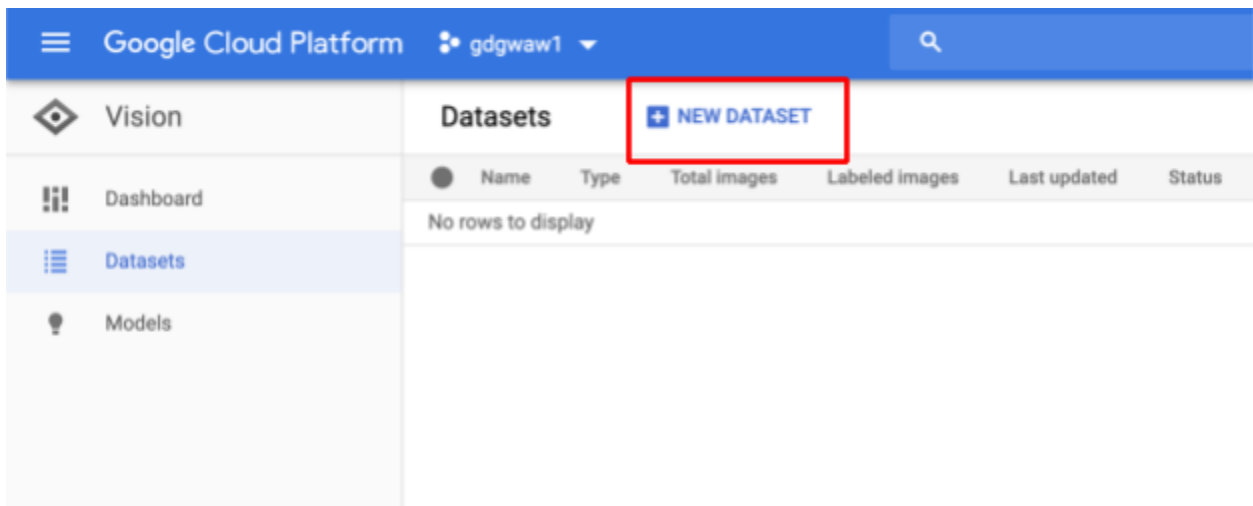
<https://cloud.google.com/vision/docs/quickstart-client-libraries>

Let's build own ML model which will recognize cats and dogs only. How to do it? Easy :)

1. Login to <https://console.cloud.google.com>
2. Choose Vision and Datasets:



3. Click NEW DATASET:



4. Name it ***cats\_and\_dogs*** and click CREATE DATASET:

## Create new dataset

Dataset name \*

cats\_and\_dogs

Use letters, numbers and underscores up to 32 characters.

Select your model objective



☒ **Single-Label Classification**  
Predict the one correct label that you want assigned to an image.



☐ **Multi-Label Classification**  
Predict all the correct labels that you want assigned to an image.



☐ **Object detection**  
Predict all the locations of objects that you're interested in.

CANCEL

CREATE DATASET

5. Mark **Select a CSV file on Cloud Storage**, write path to file with bucket name which is [PROJECT\_ID]+vcm/cats-dogs.csv and click CONTINUE:

← cats\_and\_dogs

|| LABEL STATS

EXPORT DATA

IMPORT IMAGES TRAIN EVALUATE TEST & USE

To build a custom model, you first need to import a set of images to train it. Each image should be categorized with a label. (Labels are essential for telling the model how to identify an image.)

- Each label should have at least 100 images for best results.

☐ Upload images from your computer

☒ **Select a CSV file on Cloud Storage**

### Select a CSV file on Cloud Storage

If you haven't already, upload your files to [Cloud storage](#). The CSV file should be a list of GCS paths to your images. Images can be in JPG, PNG, GIF, BMP or ICO formats. Optionally, you can specify the TRAIN, VALIDATE, or TEST split.

Sample CSV format

```
[set, ]image_path[, label]
TRAIN,gs://My_Bucket/sample1.jpg,cat
TEST,gs://My_Bucket/sample2.jpg,dog
```

gs:// \*

☒ gdgwaw1-vcm/cats-dogs.csv

BROWSE

CONTINUE

6. Select TRAIN and click START TRAINING

[←](#) cats\_and\_dogs [LABEL STATS](#) [EXPORT DATA](#)

IMPORT IMAGES **TRAIN** EVALUATE TEST & USE

### You have enough images to start training

Unlabeled images aren't used. Your dataset will be automatically split into [Train, Validation, and Test sets](#).

Ideally, each label should have at least 10 images. Fewer images often result in inaccurate precision and recall. You must also have at least 8, 1, 1 images each assigned to your Train, Validation and Test sets.

Labels	Images		Train	Validation	Test
cat	<div><div></div></div> 105		80	16	9
dog	<div><div></div></div> 115		96	6	13

START TRAINING

7. Fill the name **cats\_and\_dogs** and click CONTINUE

## Train new model

---

### 1 Define your model

Model name \*

cats\_and\_dogs

☒ Cloud hosted

Host your model on Google Cloud for online predictions

☐ Edge

Download your model for offline/mobile use

CONTINUE

### 2 Set a node hour budget

START TRAINING

CANCEL

8. Change budget to **4** and mark **Deploy model to 1 node after training**

## Train new model

---

### ✓ Define your model

### 2 Set a node hour budget

Specify the maximum number of node hours to spend training your model. If your model stops improving before then, AutoML Vision will stop training and you'll only be charged for the actual node hours used.

For cloud-hosted model. You can train for 40 node hours (per billing account) for free. Standard pricing applies afterwards. [Pricing guide](#)

Budget \*

4

?

☒ Deploy model to 1 node after training

Make your model available for REST API requests immediately after training. Deployment pricing applies.

START TRAINING

CANCEL

9. Wait for four hours :)