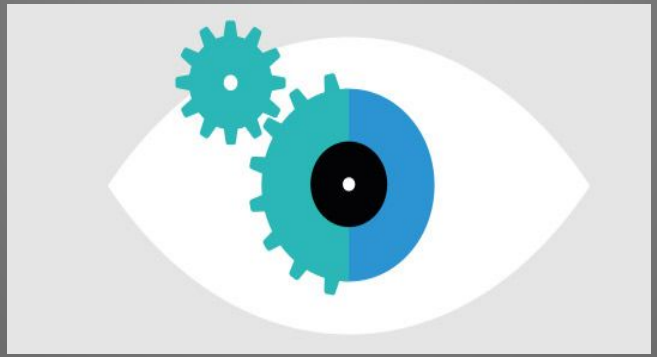
Intro to

Image Recognition API

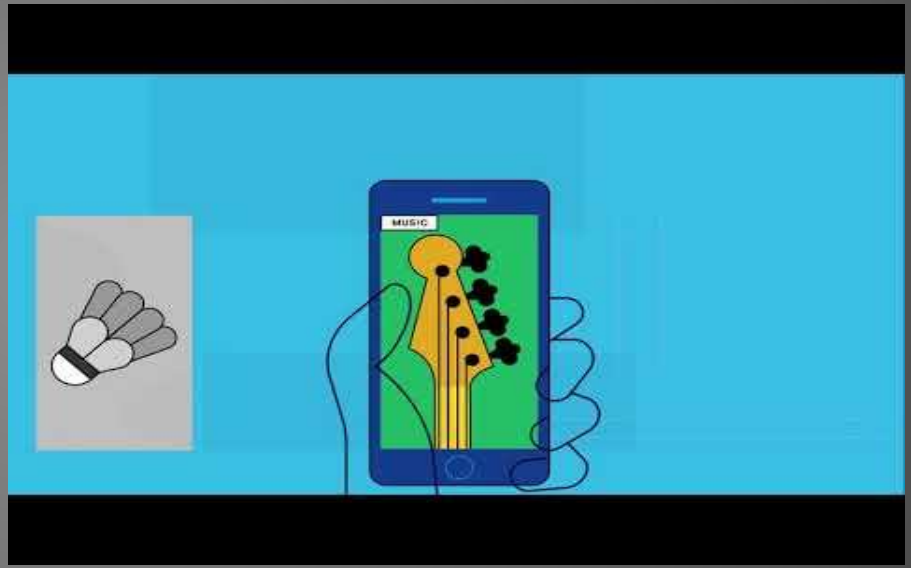
TCNJ Association for Computing Machinery
February 13, 2018

Computer Vision?

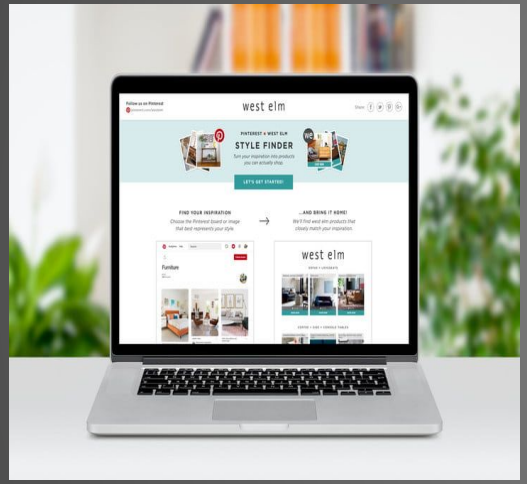


What is Clarifai?

- Founded in 2013
- Computer Vision API
- Ranked in the top 5 of the prestigious ImageNet image classification contest

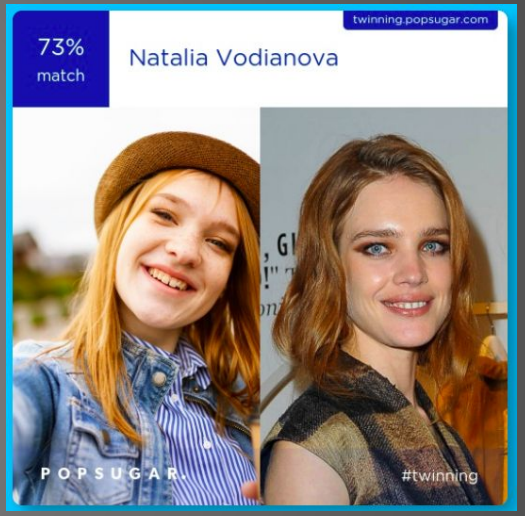


Applications



West Elm's Style Curator

9GAG NSFW Image/Video Filter



POPSUGAR's Twinning App

On to the tutorial...

- Frame-by-frame concept recognition in videos and gifs

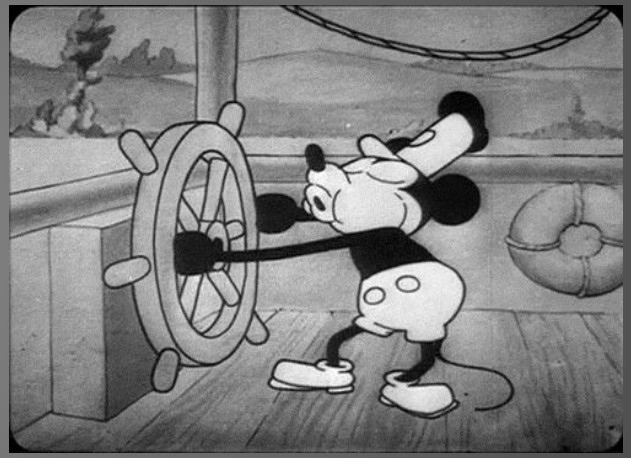
```
14.0s two
14.0s summer
14.0s nature
14.0s young
14.0s grass
15.0s people
15.0s portrait
15.0s outdoors
15.0s adult
15.0s woman
15.0s family
15.0s park
15.0s man
15.0s two
15.0s nature
15.0s summer
15.0s young
15.0s grass
```

Components

- A Clarifai Developer account (free)
- Python
  - https://www.python.org/downloads/windows/
  - PyCharm highly recommended for Windows
  - Mac OS X has Python 2.7 already
- JSON
- A video (MP4, MKV, AVI, MOV, OGG, or GIF) or gif

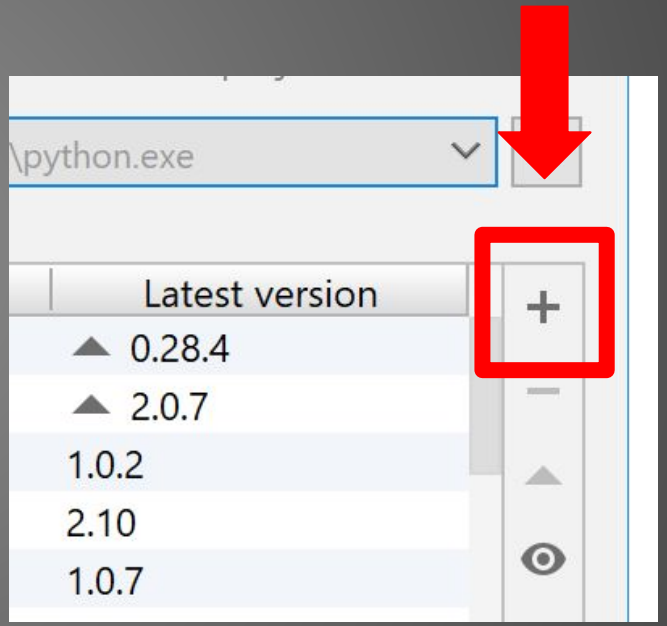
Some installation needed...

- Verify whether Python is installed by typing `-V` in the terminal
- Install Clarifai:
  - `$pip install clarifai`
- Install JSON:
  - `$pip install json`
  - Or... use https://jsoneditoronline.org/



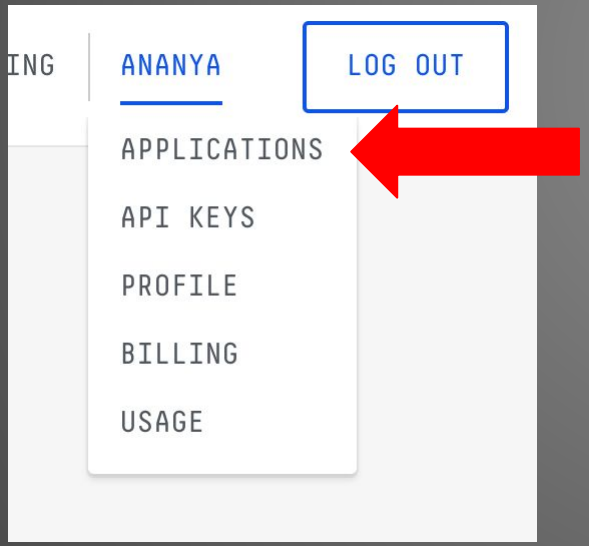
Some installation needed...

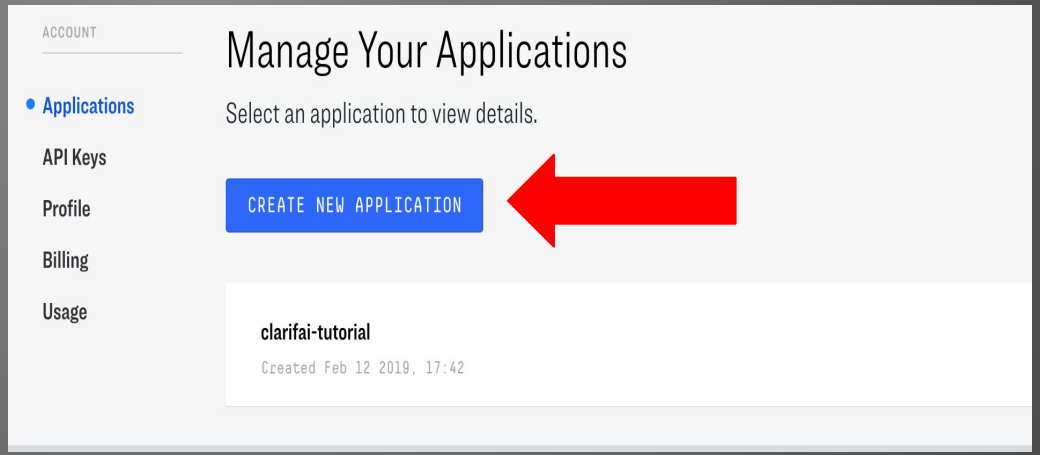
- On PyCharm:
  - File > Settings
  - Go to Project:whatever-you-named-it
  - Select Project Interpreter
    - Click the “+” sign and search for Clarifai
    - Select and check if it is in the list of packages



API Setup

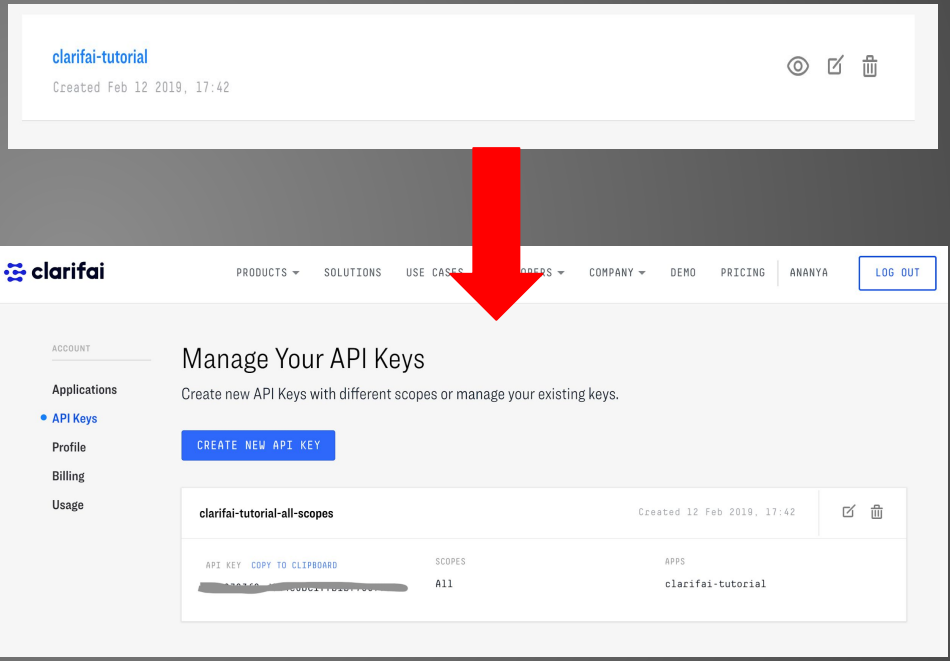
- Log into your Clarifai account > Applications > “Create New Application”





Retrieve Key

- API keys are used for **project identification** and **project authorization**
- Click on the project you just made > open the API Keys drop down



predictor.py

- Let’s make a Python file called predictor.py
- Import the following in the predictor.py file:

```
import os

from clarifai.rest import ClarifaiApp
from clarifai.rest import Video as ClVideo
```

predictor.py: Setting environmental variable

- Let’s set our environmental variable to the API key value
- Test if the value is set using a print statement

```
os.environ['CLARIFAI_API_KEY'] = '[redacted]'
print(os.environ)
print(os.environ["CLARIFAI_API_KEY"])
```

predictor.py: Call API using key

- Let’s call the API using your custom key from the Clarifai application
- Replace the INSERT_KEY_HERE value, but remember to keep the quotes around your key

```
api_key = os.environ.get('INSERT_KEY_HERE')
app = ClarifaiApp(api_key=api_key)
```

```
api_key = os.environ["CLARIFAI_API_KEY"]
app = ClarifaiApp(api_key=api_key)
```

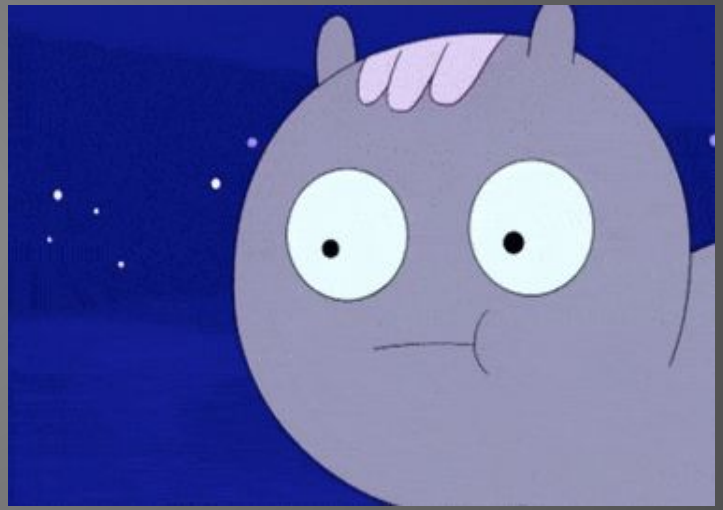
predictor.py: Send video/gif

- Let's set up the model (General concepts) we will use to recognize concepts in each frame
- Set the video to a gif/video of your choice
- Pass the video as input to the model

```
model = app.models.get('general-v1.3')
video = ClVideo(url='https://media.giphy.com/media/COoFzhnusyUVy/giphy.gif')
json = model.predict([video])
```

So what's happening?

- Sending a video/gif as a URL to the Clarifai API to process
- We are using the General v1.3 model by Clarifai for our annotator
- A JSON object is returned in response



predictor.py: Represent JSON output

- We need to represent the JSON output in a parsible manner

```
frames = json["outputs"][0]["data"]["frames"]
```

predictor.py: Parse JSON output

- Let's parse the JSON output for the video/gif
- Clarifai gives each tag is given a confidence score
  - We accept tags with scores > .85

```
for x in range(len(frames)):
    #convert time to seconds
    timeInSeconds = frames[x]['frame_info']['time']/1000
    concepts = frames[x]['data']['concepts']
    for i in range(len(concepts)):
        # filters out confidence scores <.85
        if concepts[i]['value'] > .85:
            print("{}s {}".format(timeInSeconds, concepts[i]['name']))
```

```
import os

from clarifai.rest import ClarifaiApp
from clarifai.rest import Video as ClVideo

os.environ['CLARIFAI_API_KEY'] = [redacted]
print(os.environ)
print(os.environ["CLARIFAI_API_KEY"])

api_key = os.environ["CLARIFAI_API_KEY"]
app = ClarifaiApp(api_key=api_key)

model = app.models.get('general-v1.3')
video = ClVideo(url='https://media.giphy.com/media/COoFzhnusyUVy/giphy.gif')
json = model.predict([video])

frames = json['outputs'][0]['data']['frames']

for x in range(len(frames)):
    timeInSeconds = frames[x]['frame_info']['time'] / 1000

    concepts = frames[x]['data']['concepts']

    for i in range(len(concepts)):

    # filters out all confidence scores <.94

        if concepts[i]['value'] > .94:

            print('{}s {}'.format(timeInSeconds, concepts[i]['name']))
```

predictor.py: Run program

- Command line: `python insert_name.py`
- PyCharm: Click the *Run* button and select the name of your file

```
Ananyas-MBP:clarifai-tutorial Akila$ python predictor.py
0.0s people
0.0s woman
```

Results

```
0.0s illustration
0.0s art
0.0s sketch
0.0s man
0.0s woman
0.0s vector
0.0s adult
0.0s people
```

```
0.0s illustration
0.0s art
0.0s sketch
0.0s man
0.0s woman
0.0s vector
0.0s adult
0.0s people
0.0s fantasy
0.0s fun
0.0s child
0.0s graphic
```

Questions?

Additional Resources

- https://clarifai.com/developer/guide/
- https://github.com/Clarifai