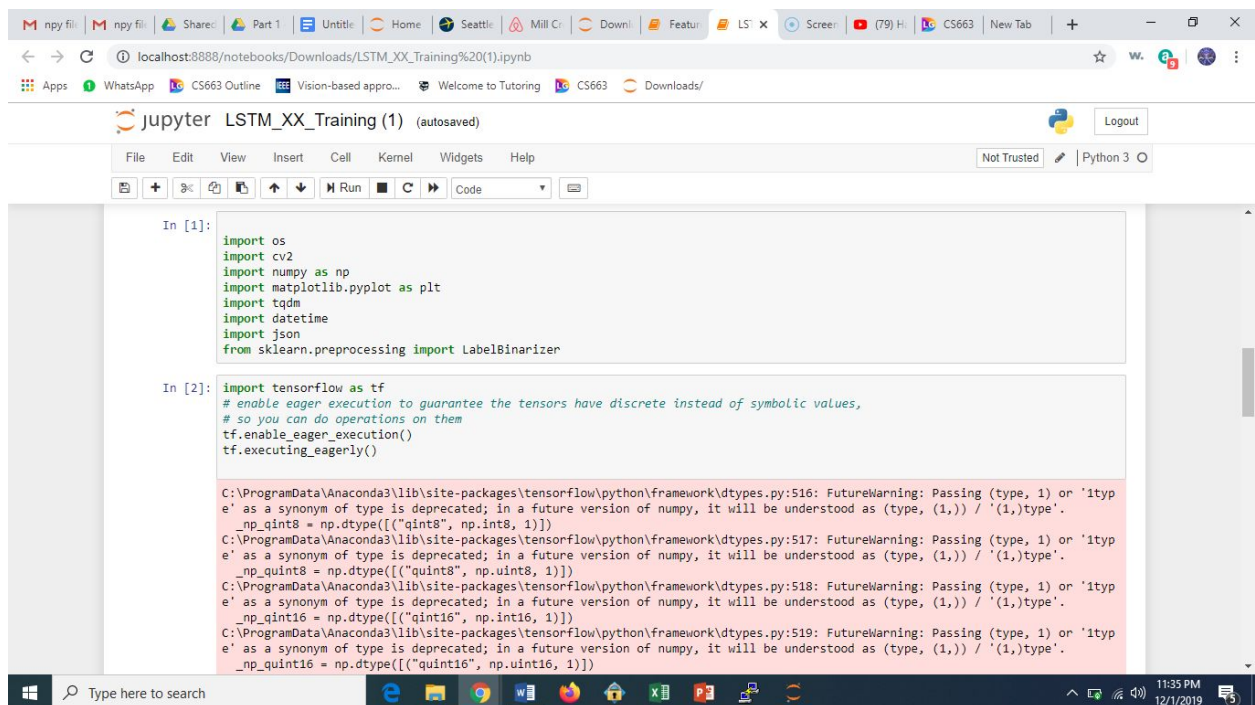


```
In [1]: import sys
!{sys.executable} -m pip install tensorflow==1.14

Collecting tensorflow==1.14
  Using cached https://files.pythonhosted.org/packages/f7/08/25e47a53692c2e0dcd2211a493ddf9007a5cd92e175d6dffa6169ab392/tensorflow-1.14.0-cp37-cp37m-win_amd64.whl
Requirement already satisfied: wheel>=0.26 in c:\programdata\anaconda3\lib\site-packages (from tensorflow==1.14) (0.33.6)
Requirement already satisfied: numpy<2.0,>=1.14.5 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorflow==1.14) (1.17.4)
Requirement already satisfied: google-pasta>=0.1.6 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorflow==1.14) (0.1.8)
Collecting tensorboard<1.15.0,>=1.14.0 (from tensorflow==1.14)
  Using cached https://files.pythonhosted.org/packages/91/2d/2ed263449a078cd9c8a9ba50ebd50123ad1f8cfbea1492f9084169b89d9/tensorboard-1.14.0-py3-none-any.whl
Requirement already satisfied: keras-applications>=1.0.6 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorboard==1.14) (1.0.8)
Requirement already satisfied: keras-preprocessing>=1.0.5 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorboard==1.14) (1.1.0)
Collecting tensorflow-estimator<1.15.0rc0,>=1.14.0rc0 (from tensorflow==1.14)
  Using cached https://files.pythonhosted.org/packages/3c/d5/21860a5b11caf0678fbc8319341b0ae21a07156911132e0e71bffd0510d/tensorflow-estimator-1.14.0-py2.py3-none-any.whl
Requirement already satisfied: six>=1.10.0 in c:\programdata\anaconda3\lib\site-packages (from tensorflow-estimator==1.14) (1.12.0)
Requirement already satisfied: absl-py>=0.7.0 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorflow-estimator==1.14) (0.8.1)
Requirement already satisfied: grpcio>=1.8.6 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorflow-estimator==1.14) (1.25.0)
Requirement already satisfied: gast>=0.2.0 in c:\users\stsc\appdata\roaming\python\python37\site-packages (from tensorflow-estimator==1.14) (0.2.2)
```

## Install Tensorflow V : 1.14



```
In [1]: import os
import cv2
import numpy as np
import matplotlib.pyplot as plt
import tqdm
import datetime
import json
from sklearn.preprocessing import LabelBinarizer

In [2]: import tensorflow as tf
# enable eager execution to guarantee the tensors have discrete instead of symbolic values,
# so you can do operations on them
tf.enable_eager_execution()
tf.executing_eagerly()

C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\framework\dtypes.py:516: FutureWarning: Passing (type, 1) or 'intype' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\framework\dtypes.py:517: FutureWarning: Passing (type, 1) or 'intype' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.uint16, 1)])
C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\framework\dtypes.py:518: FutureWarning: Passing (type, 1) or 'intype' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\framework\dtypes.py:519: FutureWarning: Passing (type, 1) or 'intype' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint64 = np.dtype([("qint64", np.int64, 1)])
```

## Import all dependencies

The screenshot shows a Jupyter Notebook titled "LSTM\_XX\_Training (1)" running on a local host. The code in the first two cells is as follows:

```
In [3]: # base path of video dataset
BASE_PATH = 'C:\\Users\\STSC\\Desktop\\Project3\\Training_Set\\new data'
VIDEOS_PATH = os.path.join(BASE_PATH, '**', '*.mp4')

# sequence length LSTM will process
SEQUENCE_LENGTH = 40
BATCH_SIZE = 16

In [4]: def frame_generator():
    video_paths = tf.io.gfile.glob(VIDEOS_PATH)
    np.random.shuffle(video_paths)
    for video_path in video_paths:
        frames = []
        cap = cv2.VideoCapture(video_path)
        num_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
        sample_every_frame = max(1, num_frames // SEQUENCE_LENGTH)
        current_frame = 0

        label = os.path.basename(os.path.dirname(video_path))

        max_images = SEQUENCE_LENGTH
        while True:
            success, frame = cap.read()
            if not success:
                break

            if current_frame % sample_every_frame == 0:
                frame = frame[:, :, :-1]
                img = tf.image.resize(frame, (224, 224))
```

Set the path of the Video files of the Dataset

The screenshot shows the same Jupyter Notebook with additional code in the third and fourth cells:

```
In [5]: dataset = tf.data.Dataset.from_generator(frame_generator,
        output_types=(tf.float32, tf.string),
        output_shapes=((224, 224, 3), ()))

dataset = dataset.batch(BATCH_SIZE, drop_remainder=True).prefetch(tf.data.experimental.AUTOTUNE)

WARNING:tensorflow:From C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\data\ops\dataset_ops.py:494: py_func (from
tensorflow.python.ops.script_ops) is deprecated and will be removed in a future version.
Instructions for updating:
tf.py_func is deprecated in TF V2. Instead, there are two
options available in V2.
- tf.py_function takes a python function which manipulates tf eager
tensors instead of numpy arrays. It's easy to convert a tf eager tensor to
an ndarray (just call tensor.numpy()) but having access to eager tensors
means `tf.py_function`s can use accelerators such as GPUs as well as
being differentiable using a gradient tape.
- tf.numpy_function maintains the semantics of the deprecated tf.py_func
(it is not differentiable, and manipulates numpy arrays). It drops the
stateful argument making all functions stateful.

In [6]: print(dataset)

<DatasetV1Adapter shapes: ((16, 224, 224, 3), (16,)), types: (tf.float32, tf.string)>

In [7]:
```

Create and assign Dataset for processing