|  |
| --- |
| Close-up image showing the leaf-sides of two oversized books side-by-side on a bookshelf, with additional books in soft focus background |
| LipReading using Deep Learning  Presented by: Ponnapalli Damarukeswar, Pujari N Thanmayee |
| |  |  |  | | --- | --- | --- | | thanmayee pujara |  |  | |

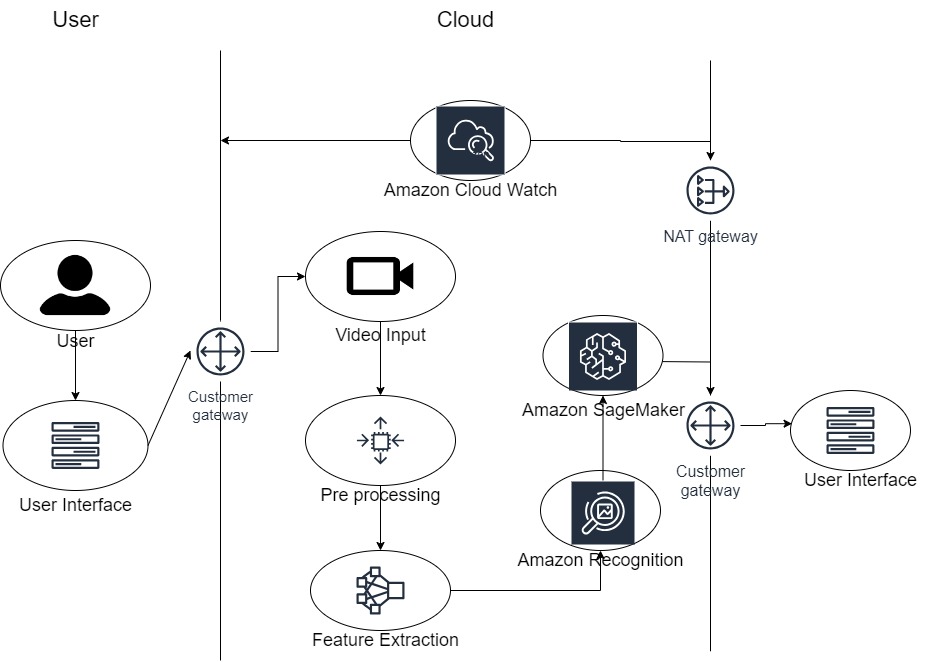
**Lip Reading deep learning project**

The objective of this project is to develop an end-to-end machine learning solution to detect words from a video of a person speaking. The proposed solution involves the use of Deep learning algorithms like LSTM, Neural Networks to predict the accurate output.

Lip reading using machine learning can offer several benefits:

1. **Improved Speech Recognition:** Lip reading can complement audio-based speech recognition systems, especially in noisy environments or scenarios where the audio signal is unclear. Integrating lip reading with traditional speech recognition can enhance accuracy and robustness.
2. **No Need for Audio Data**: Traditional speech recognition models require large amounts of transcribed audio data for training. In contrast, an end-to-end lip reading system can be trained solely on video data, eliminating the need for transcribed audio, which can be expensive and time-consuming to obtain.
3. **Multi-Modal Applications:** End-to-end lip reading can be combined with audio-based systems to create multi-modal applications. For example, in video conferencing, it can help improve real-time communication by providing more accurate transcriptions.
4. **Accessibility for Hearing-Impaired Individuals:** Lip reading can be an essential communication tool for individuals with hearing impairments. An accurate lip reading system can enhance their ability to understand spoken language and participate in conversations

**Technical Architecture**



**Project Flow:**

* + The user interacts with the UI to select the file.
  + Selected input is analyzed by the model which is integrated/developed through the customer gateway it will get pre-processed and required features will be extracted.
  + Once the model analyzes the input, the prediction is showcased on the UI. To accomplish this, we have to complete all the activities listed below,
  + Define Problem / Problem Understanding
    1. Specify the business problem ○ Business requirements ○ Literature Survey.

○ Social or Business

Impact. ● Data Collection &

Preparation

○ Collect the dataset

○ Data Preparation

* + Exploratory Data Analysis
    1. Descriptive statistical

○ Visual Analysis

* + Model Building ○ Building a model.
    1. Training the model.

○ Testing the model

* + Model Deployment
    1. Save the best model

○ Integrate with Web Framework

* + Project Demonstration & Documentation
    1. Record explanation Video for project end to end solution

○ Project Documentation-Step by step project development procedure

**Prior Knowledge:**

To complete this project, you must require following software’s , concepts and packages

* + VS Code: o Refer to the link below to download VS Code. o Link : <https://code.visualstudio.com/download>● Create Project: o Open VS Code. o Create a Folder and name it “Lip Reading”.
  + Machine Learning Concepts o Deep Learning:

[https://towardsdatascience.com/introducing-deep-learning-and-neural-networks-deep-learning-forr](https://towardsdatascience.com/introducing-deep-learning-and-neural-networks-deep-learning-for-rookies-1-bd68f9cf5883) [cookies](https://towardsdatascience.com/introducing-deep-learning-and-neural-networks-deep-learning-for-rookies-1-bd68f9cf5883)o NLP Models :

<https://medium.com/voice-tech-podcast/an-overview-of-rnn-lstm-gru-79ed642751c6>

* + Web concepts: o Get the gist on streamlit :

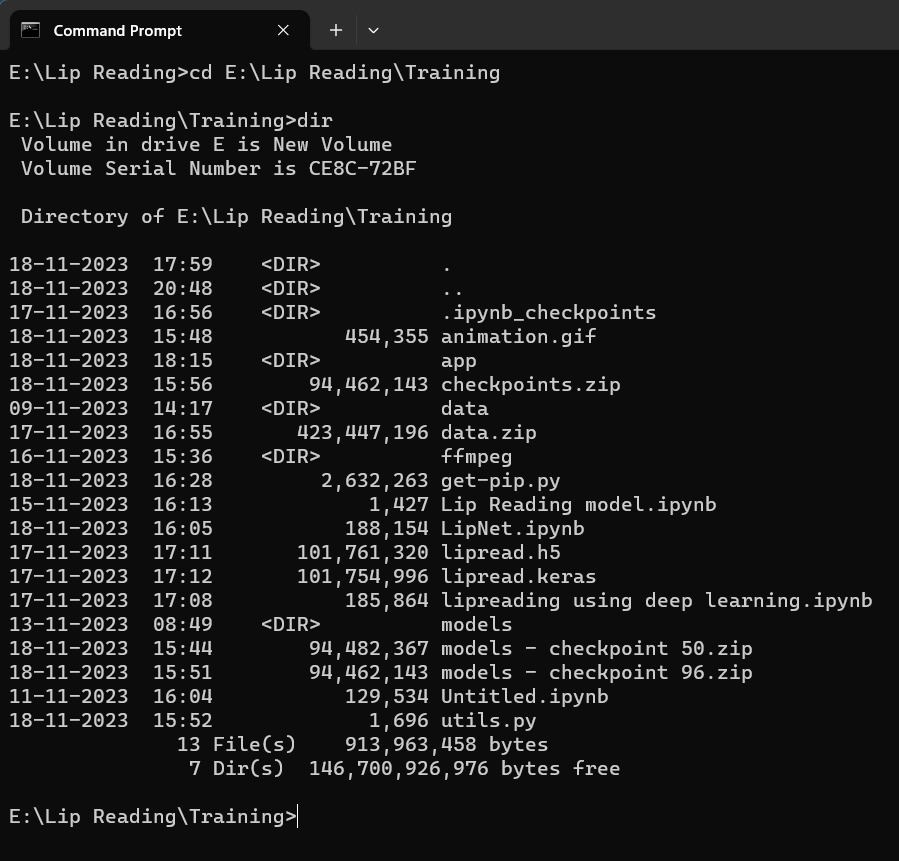
<https://www.geeksforgeeks.org/a-beginners-guide-to-streamlit/>

**Project Objectives:**

* + Know fundamental concepts and techniques of Deep Learning.
  + Gain a broad understanding of Lip Reading.
  + Know how to train models in an efficient way.
  + Know how to build a web application using the Streamlit framework.

**Project Structure:**

Project folder which contains files as shown below



* + We are building a Streamlit application which needs an app folder for a website.
  + models folder contains your saved models.
  + The Training folder contains the code for building/training/testing the model. ● The Dataset folder contains the videos and alignments.

# Milestone 1: Define Problem / Problem Understanding

**Activity 1: Specify the business problem**

Refer Project Description

## Activity 2: Business requirements

Here are some potential business requirements for lip reading using deep learning:

1. Accurate prediction: The predictor must be able to accurately predict the words. The accuracy of the prediction is crucial for the client. The client could be a corporate or a business person or anyone.
2. User-friendly interface: The predictor must have a user-friendly interface that is easy to navigate and understand. The interface should present the results of the predictor in a clear and concise manner.
3. Scalability: The predictor must be able to scale up based on the prediction from our product. The model should be able to handle any size of data without compromising on its accuracy or efficiency.

## Activity 3: Literature Survey (Student Will Write)

To identify the current state-of-the-art methods for lip reading using deep learning, their strengths and weaknesses, and gaps in knowledge. We have selected different paths by using the internet resources accurately like ChatGPT, Google search engine, Google scholar we have gone through many internet resources. Evaluate sources based on credibility, relevance, authority, and recency. Extract key findings, methodologies, and performance metrics. Like Reading alignment and mp4 files and converting them into integers by preprocessing for analysis. Synthesize information by summarizing, identifying themes, and categorizing research. Identify gaps and research opportunities, such as developing robust models and incorporating additional modalities.

**Activity 4: Social or Business Impact.**

1. **Privacy and Security:** Audio-based speech recognition systems may raise privacy concerns, as they capture and process audio data, potentially infringing on individuals' privacy. Lip reading systems, on the other hand, rely on visual information and might be considered less intrusive in this regard.
2. **Use in Noisy Environments:** In environments with high background noise, audio-based speech recognition can be challenging. Lip reading can help provide context and improve accuracy in these noisy scenarios.
3. **Cross-Lingual Applications:** Lip reading is language-agnostic, which means the same model can potentially be applied to lip reading in different languages without requiring language-specific training data.

# Milestone 2: Data Collection & Preparation

DL depends heavily on data. It is the most crucial aspect that makes algorithm training possible. So, this section allows you to acquire the required dataset.

**NOTE : If you don't have a GPU, then train your model on kaggle.**

**Activity 1: Loading the data.**

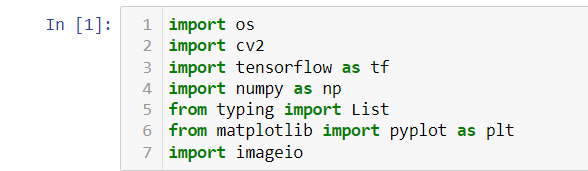
Link for the required dataset : <https://www.kaggle.com/datasets/rishisrdy/lipreading>

Download the provided data and load the data into the data folder

## Activity 2: Data Preparation

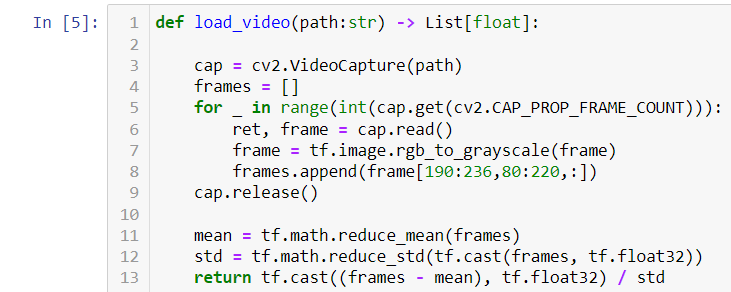
As we have understood how the data is, let's pre-process the collected data.

Let’s import the required packages

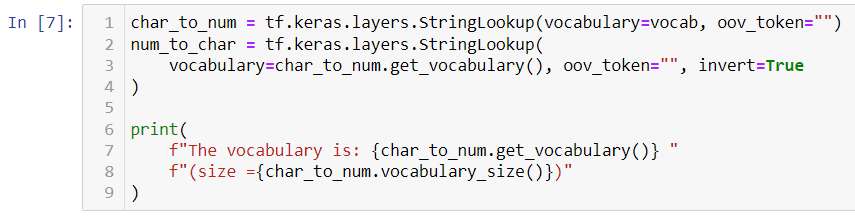


We require 5 functions to process the data

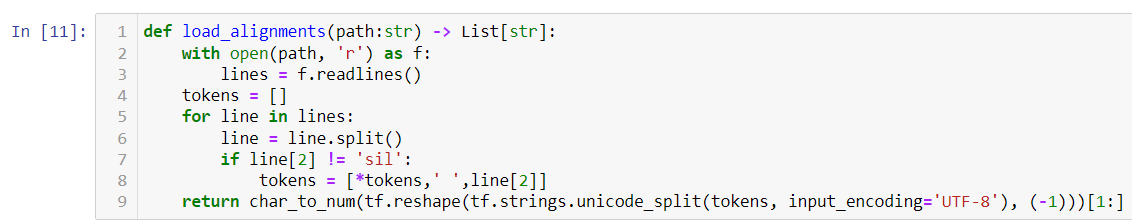
* load\_video()
* char\_to\_num
* num\_to\_char
* load\_alignments()
* load\_data()
* **load\_video():**



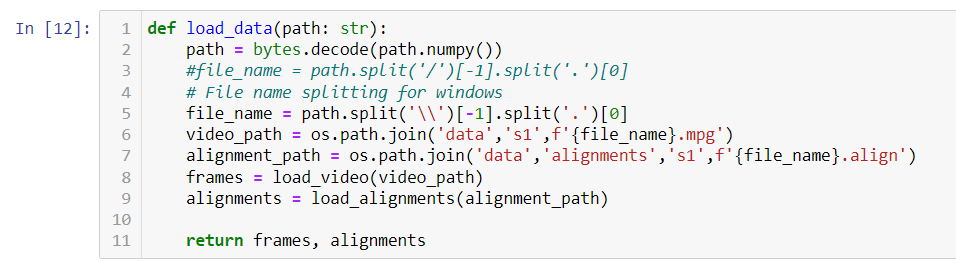
* **char\_to\_num:**
* **num\_to\_char:**



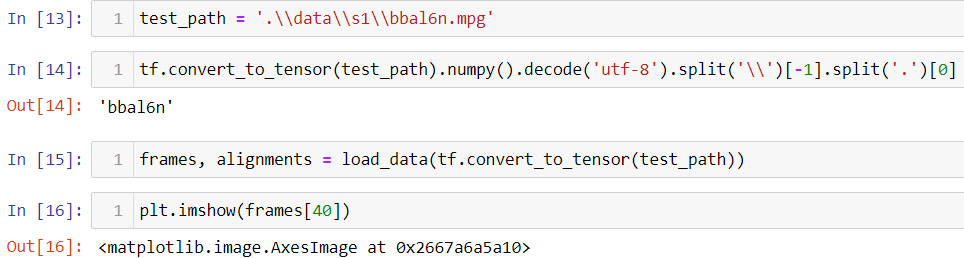
* **load\_alignments():**



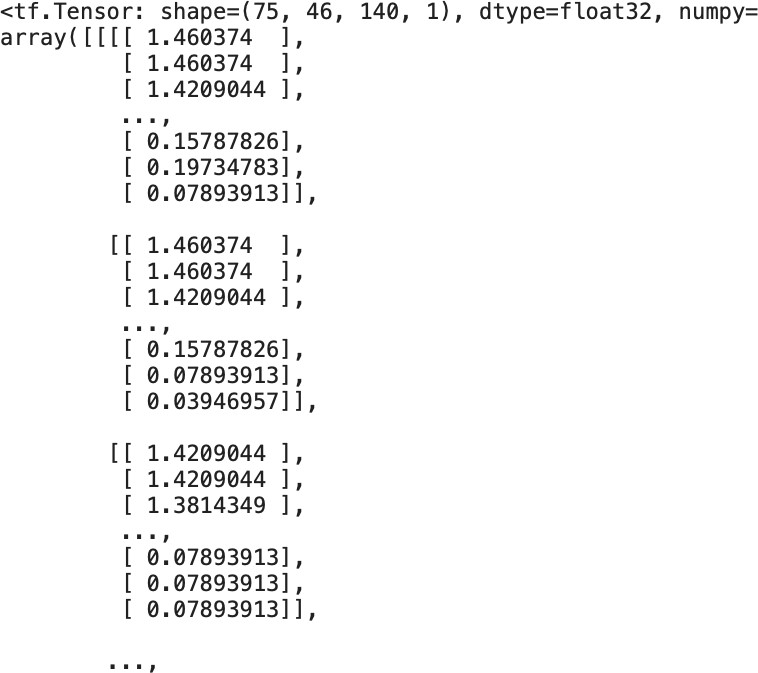
* **load\_data():**



Let’s convert the data into tensors using the above functions



The output will be :



and the output goes on….

# Milestone 3: Exploratory Data Analysis

## Activity 1: Descriptive statistical

Descriptive analysis is to study the basic features of data with the statistical process. With this describe function we can understand the unique, top and frequent values of categorical features. And we can find mean, std, min, max and percentile values of continuous features.

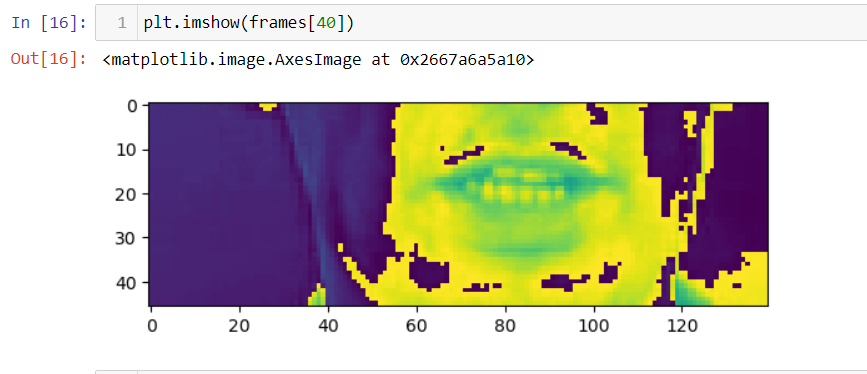
Which are not suitable for our dataset.

As our data consists of videos and alignments, There is no unnecessary data which can be eliminated.

## Activity 2: Visual analysis

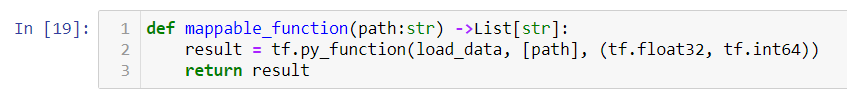
Visual analysis is the process of using visual representations, such as charts, plots, and graphs, to explore and understand data. It is a way to quickly identify patterns, trends, and outliers in the data, which can help to gain insights and make informed decisions.

Let’s plot the converted frames

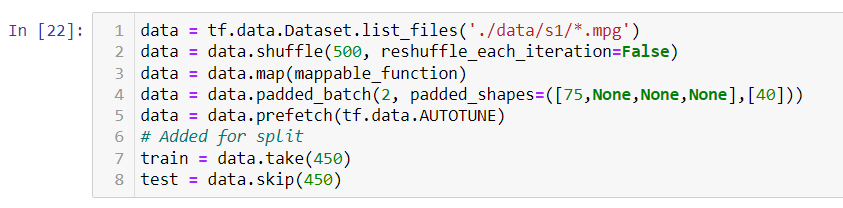
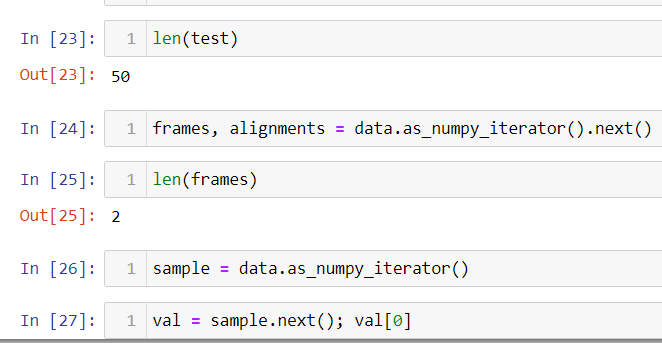


**Activity 3: Splitting data into train and test and validation sets**

First we will create a mapable function



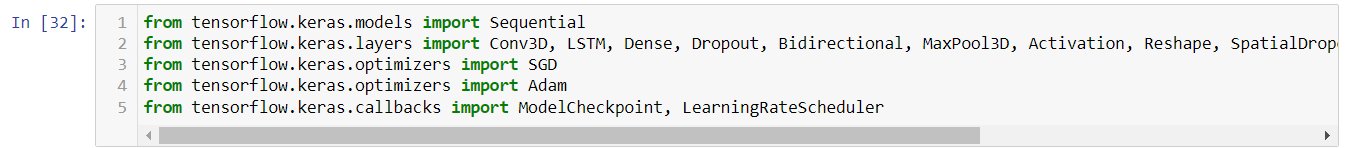
The following code is to divide the preprocessed data into train and test data equally.

The below codes are to print the preprocessed data

# Milestone 4: Model Building

## Activity 1: Importing necessary libraries

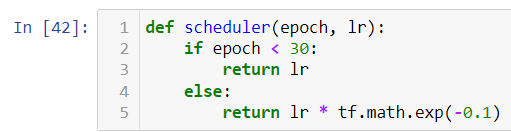
We’ll import some necessary libraries using the following code



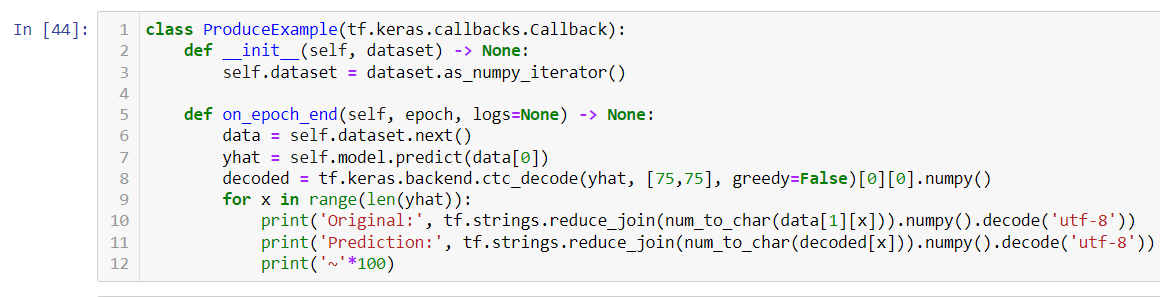
**Activity 2: Defining callbacks, Loss function and building the model**

Let’s define the following: ● Callbacks :

○ scheduler()

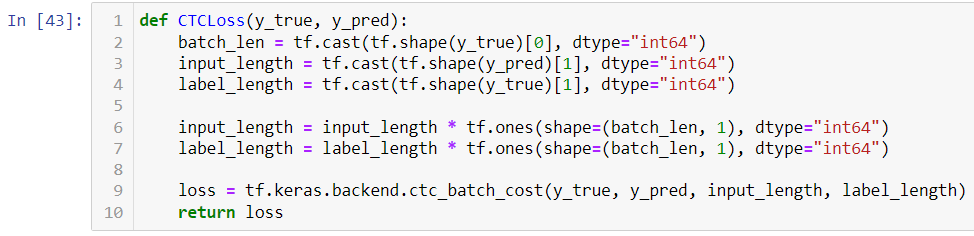


○ ProduceExample()



* Loss Function:

○ CTCLoss() :



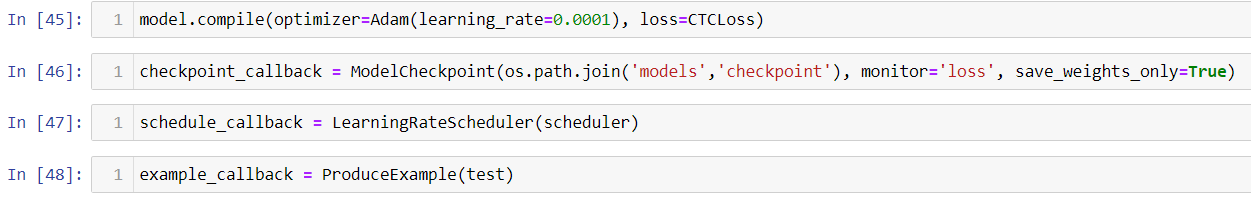
* Model Building :



## Activity 3: Train the models

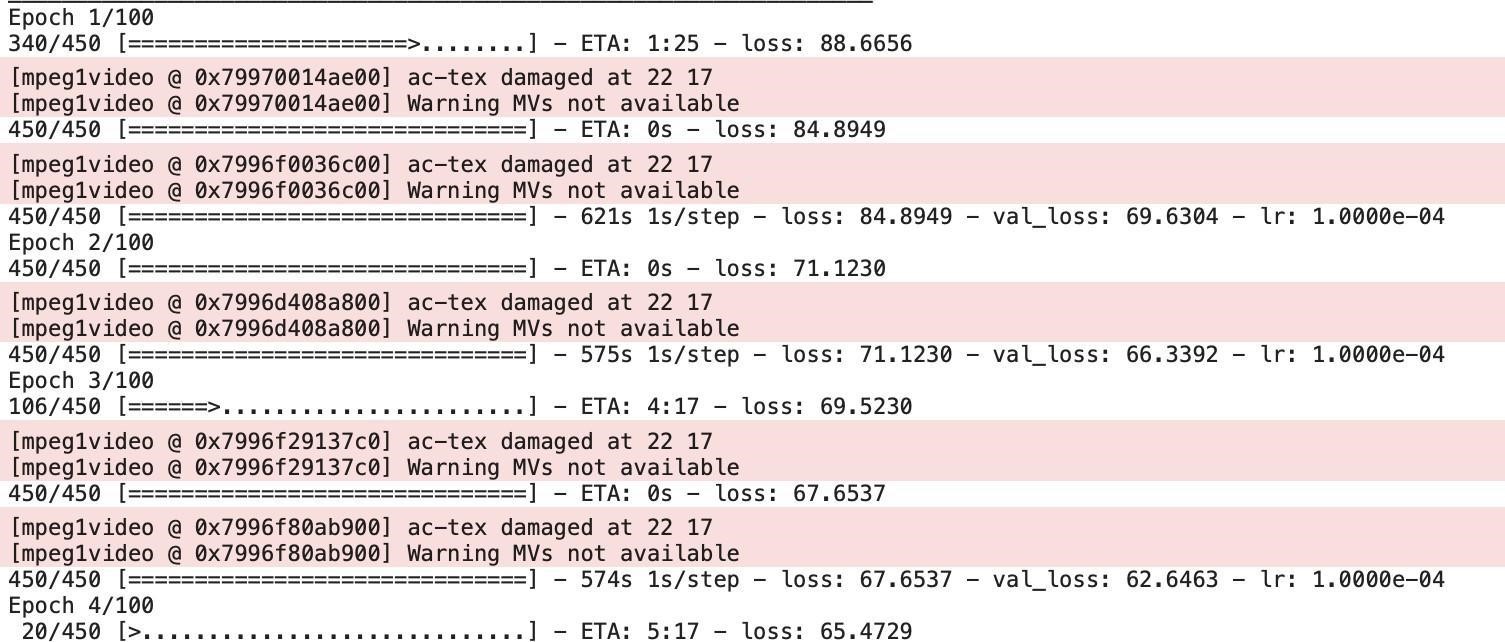
First initialize the callback functions

Then we compile the model



Now it’s time to train the model





Let’s wait for the model to train until 100 epochs.

# Milestone 5: Model Deployment

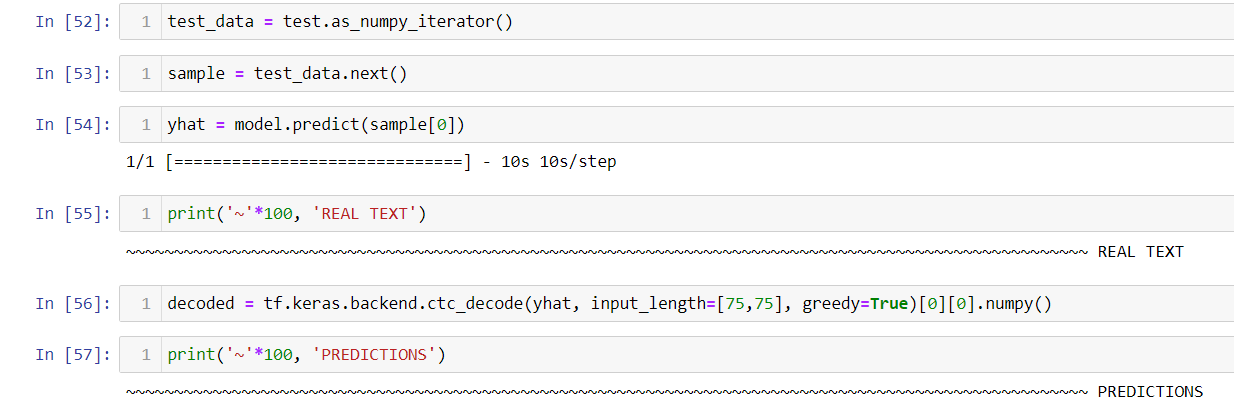
## Activity 1: Save the model

The model is saved in checkpoints

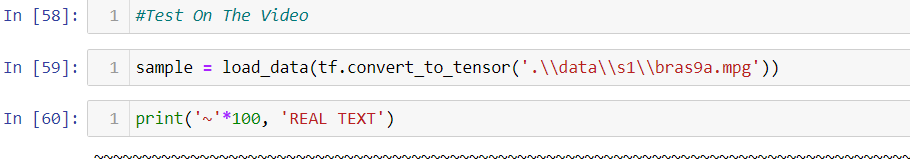
## Activity 2: Make a Prediction

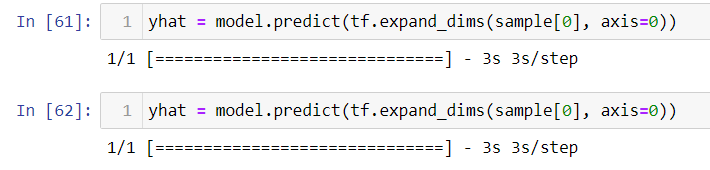


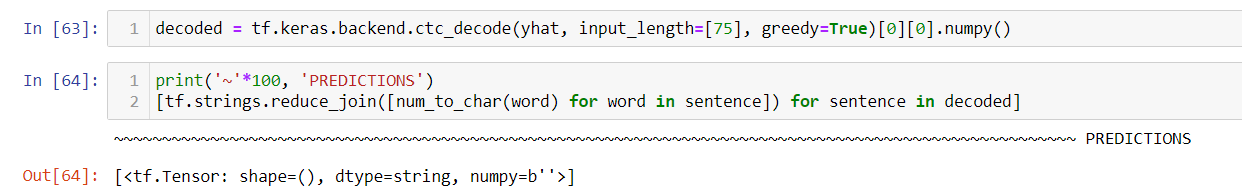
Now, Let's make the prediction on text data.



Now, Let’s make prediction on video







## Activity 3: Integrate with Web Framework

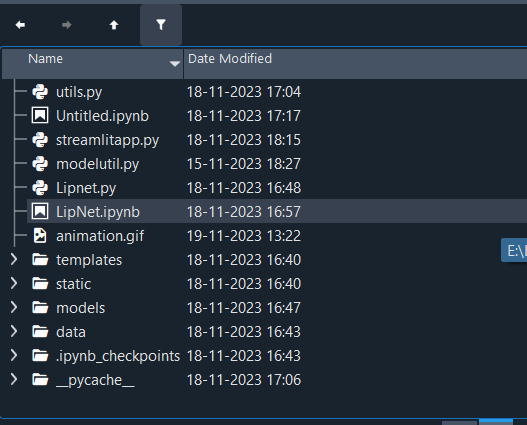
In this section, we will be building a web application that is integrated to the model we

built.

We will be using the streamlit package for our website development.

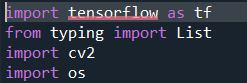
Streamlit is a free and open-source framework to rapidly build and share beautiful machine learning and data science web apps.

**Activity 4.1: Create an app folder and create following .py files in it:**

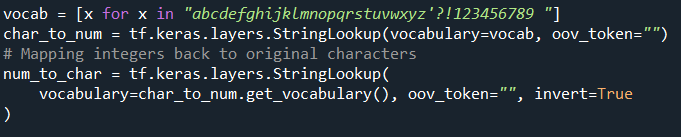


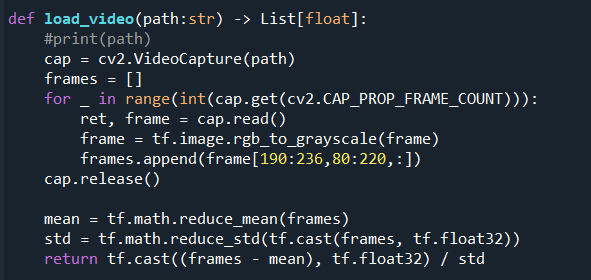
Let’s write the code for **“utils.py”**

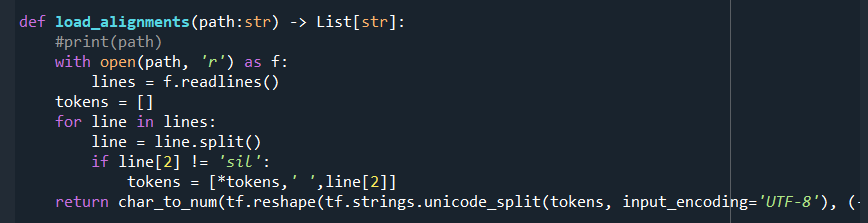
* **Importing necessary packages:**

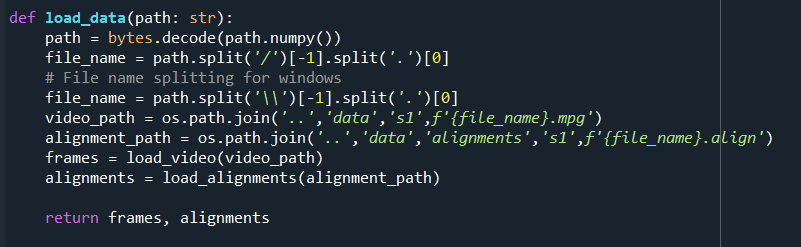


* **Defining char\_to\_num, num\_to\_char, load\_video(), load\_alignments(), load\_data() functions:**









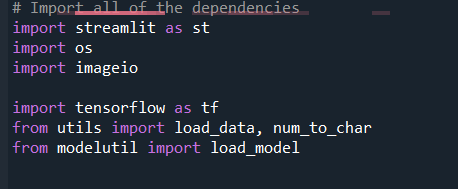
Let’s write code for **“modelutils.py” ● Import all the necessary packages:**



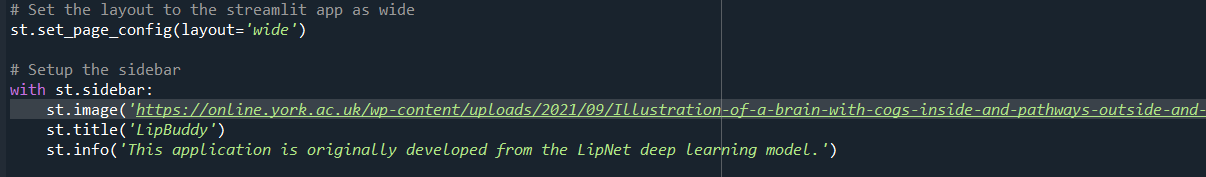
* **Defining load\_model() function :**

Let’s write code for **“streamlitapp.py”**

* **Importing necessary packages and functions:**

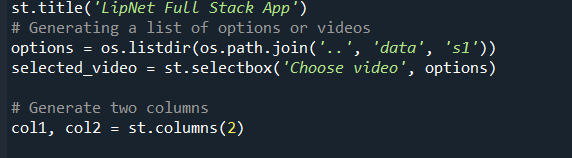


* **Creating sidebar:**



You can choose any image of your wish.

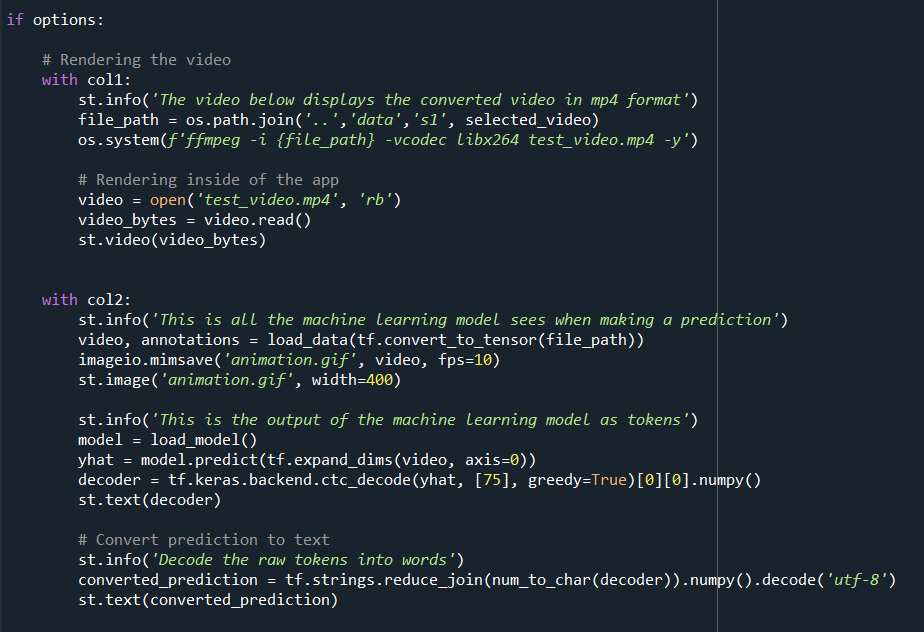
* **Drop down menu for selecting the input :**



* **Creating two columns :**



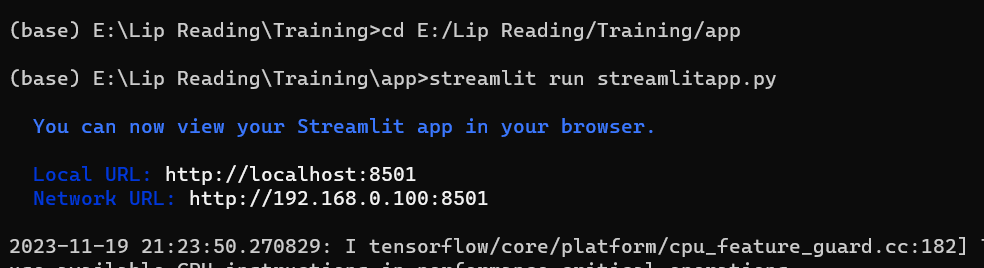
* **col1 and col2 :**



* col1 will display the selected video and play it.
* col2 will display :
  + animation of lips.
  + raw output of your model
  + result prediction

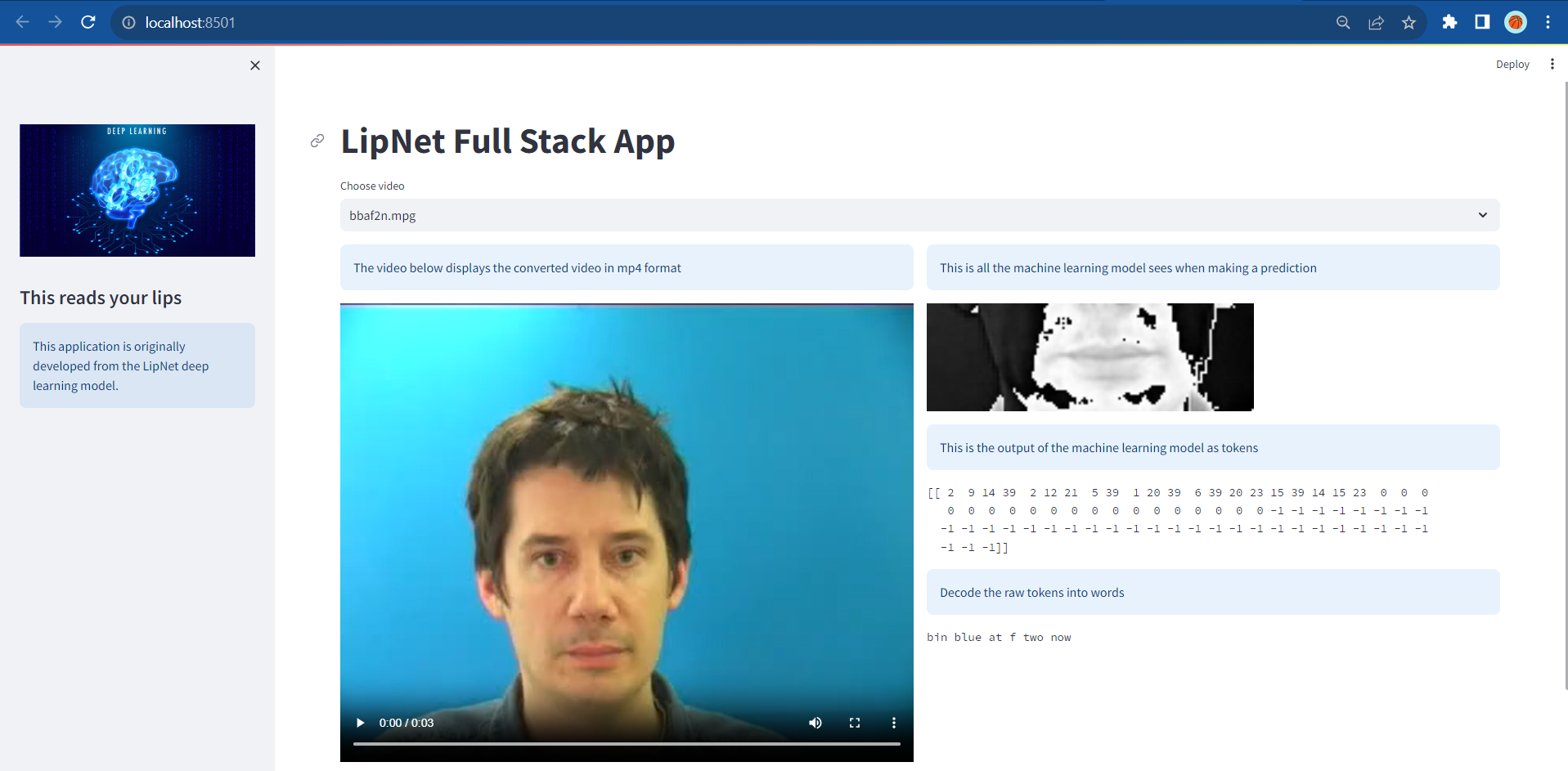
**To run this website go to terminal with your respective directory and run the command :**

* “streamlit run streamlitapp.py”



On successful execution you’ll get to see this in your terminal.

**HOW DOES YOUR WEBSITE LOOK LIKE ?**



## Milestone 6: Project Demonstration & Documentation

Below mentioned deliverables to be submitted along with other deliverables.

**Activity 1: - Record explanation Video for project end to end solution**

<https://drive.google.com/file/d/1PNgZXUPZ5iNq36aw5EuAkP350D9EGoFl/view?usp=drive_link>