**A REPORT**

**ON**

**Food Recognisation System Using Deep Learning and Computer Vision.**

**By**

Teja sai Siddharth – AP21110011421

Chaitanya Nath Gaddam – AP21110011422

Raja sekhar Naidu – AP21110011444

Lakshmipathi Pathagunta - AP21110011457

Guna sekhar Salapu – AP21110011476

### Undergraduate Research Opportunities Project (UROP) CSE340 – Group 2

***Under the guidance of Prof. Mr. Ravi Kant Kumar***

### AT

**SRM University AP**

**A picture containing text

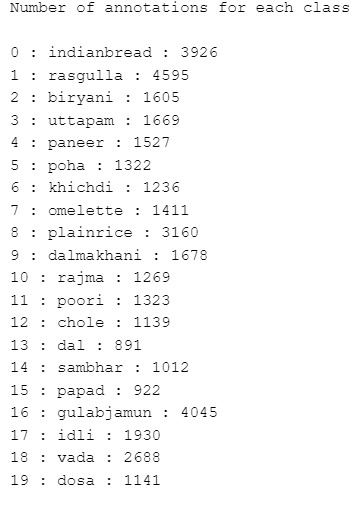
Description automatically generated**

1. **Goals and objectives** –

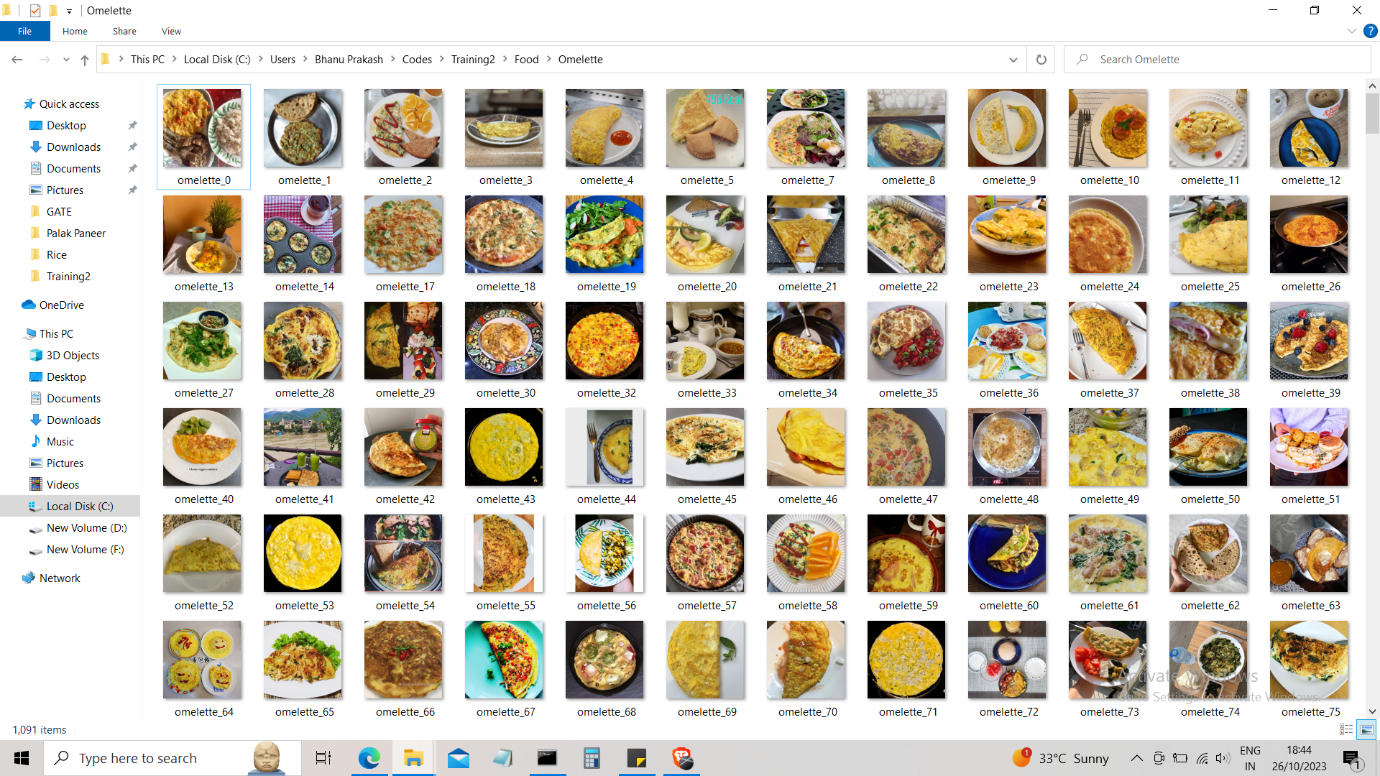
* Our primary objective is to develop a deep learning-based system for the recognition of Indian cuisine.
* Specifically focusing on enhancing the identification and classification of 15 common Indian recipes using advanced deep learning techniques.

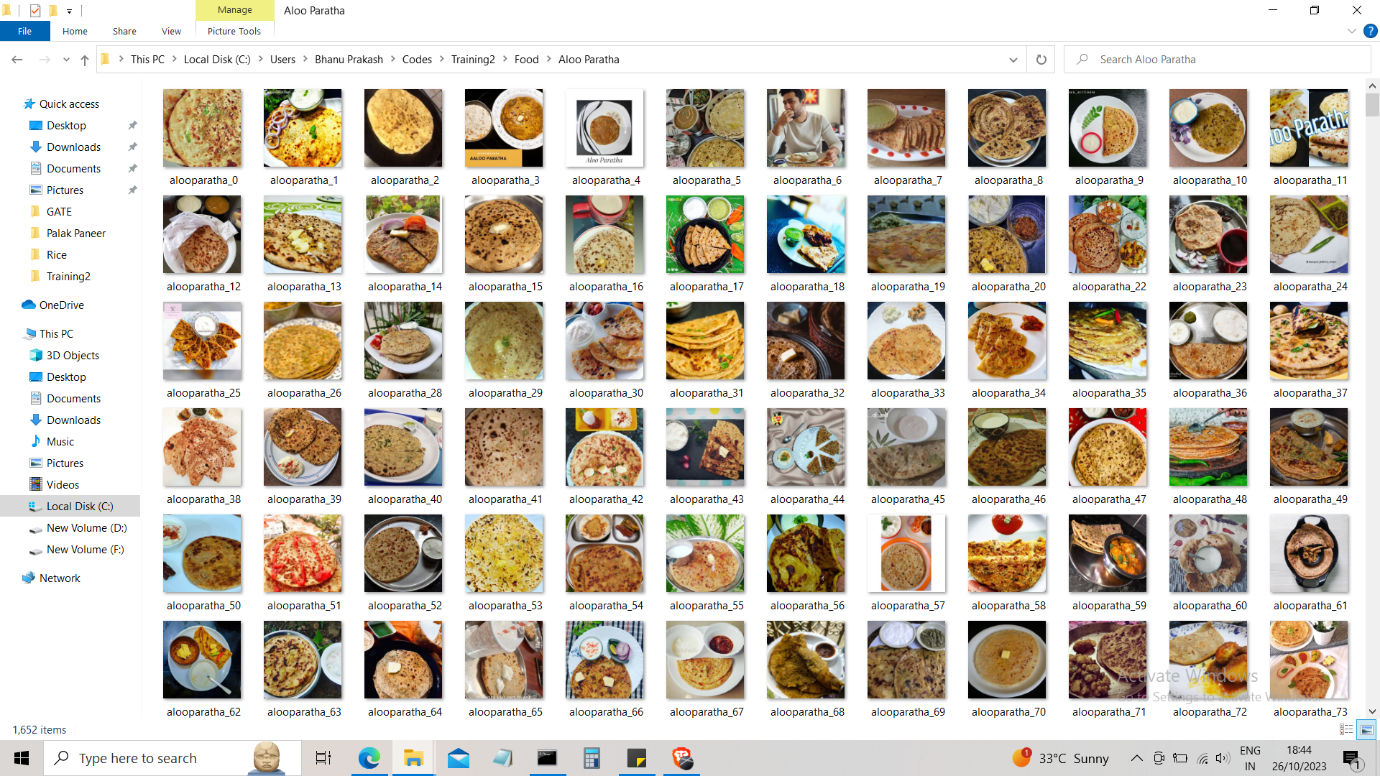
1. **Data collection and Annotations –**

* Gathered a comprehensive dataset from diverse online sources, ensuring a wide representation of Indian culinary items.
* Categorized the dataset meticulously, organizing each image into specific folders corresponding to individual food items.
* Curated an extensive collection of over 1000 high-quality images for each of the 15 selected Indian dishes, ensuring robust training and validation for the deep learning model.



Dataset sample images :





**3.MODEL**

We had used Convolutional Neural Networks for this project to detect food images.

**Convolutional Neural Network:**

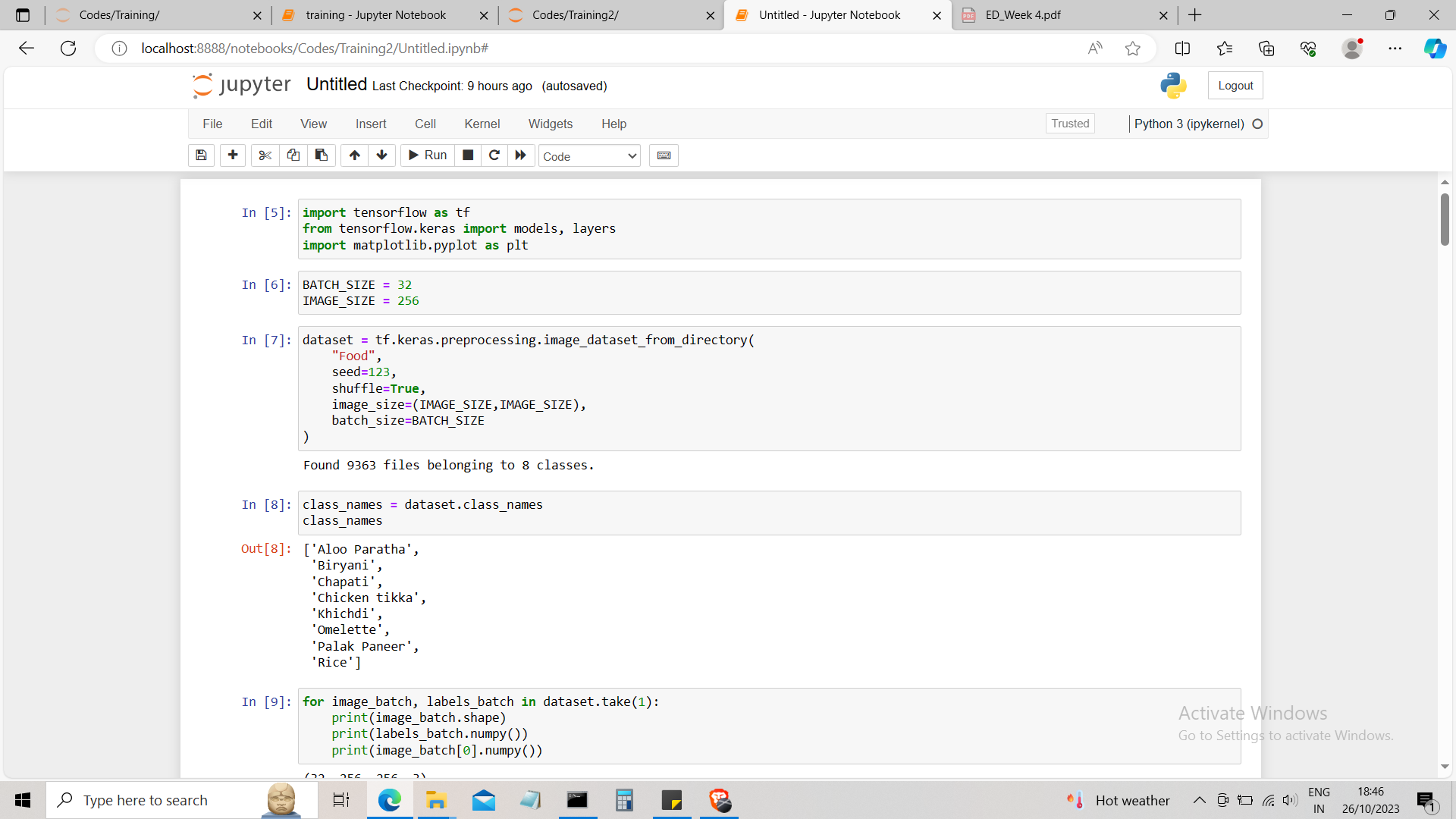
* Convolution: The CNN algorithm utilizes convolutional layers to extract features from the input image. These layers consist of a set of learnable filters that convolve across the input image, detecting various features such as edges, textures, and patterns.
* Non-linear Activation: Following the convolutional layers, non-linear activation functions like ReLU (Rectified Linear Unit) are applied to introduce non-linearity, enabling the network to learn complex relationships between the extracted features.
* Pooling: Pooling layers are employed to reduce the spatial dimensions of the feature maps, thereby decreasing the computational complexity and controlling overfitting. Common pooling operations include max pooling or average pooling, which retain the most significant features while discarding less relevant information.
* Fully Connected Layers: Towards the end of the network, fully connected layers are utilized to perform high-level reasoning and decision-making based on the extracted features. These layers connect every neuron from one layer to every neuron in the next layer, enabling the network to learn intricate relationships between the features.
* Classification: The final layer typically involves a function, which provides the probability distribution of the classes in the output layer. This enables the network to classify the input image into one of the predefined categories based on the learned features and parameters.

We used libraries like matplotlib and tensorflow-

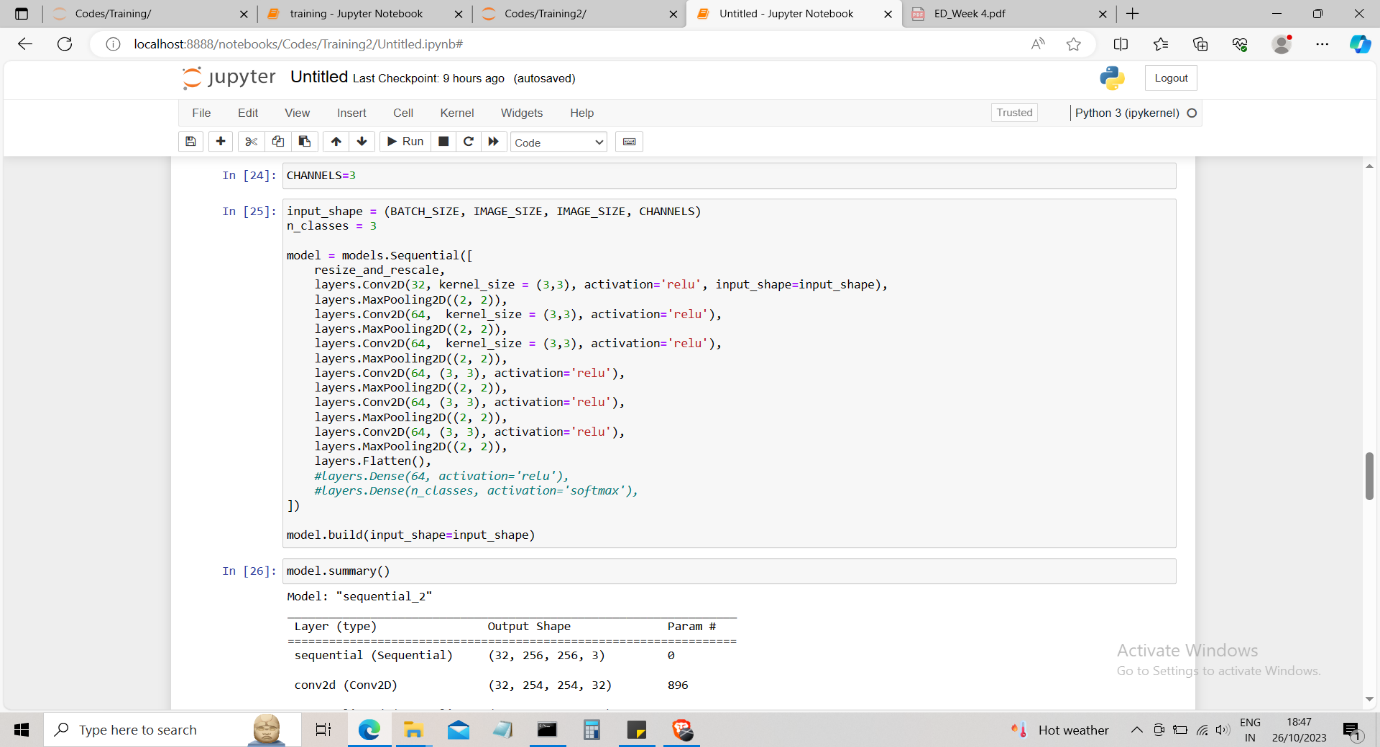
TensorFlow is an open-source machine learning framework that provides tools for building and training deep learning models. It offers a high-level API, Keras, which simplifies the process of designing neural networks, including CNNs. TensorFlow is utilized for creating the architecture of the CNN, defining the layers, compiling the model, and training the network using backpropagation and optimization algorithms.

Matplotlib is a popular plotting library in Python that enables the creation of various types of plots, charts, and visualizations. In the context of this project, Matplotlib is used for visualizing the dataset, displaying sample images, and presenting the performance metrics of the trained CNN model, such as accuracy and loss, through graphs and plots. Matplotlib helps in gaining insights into the data and evaluating the effectiveness of the CNN model during the training and testing phases.

Importing required libraries and defining batch size and image size and classifying data.

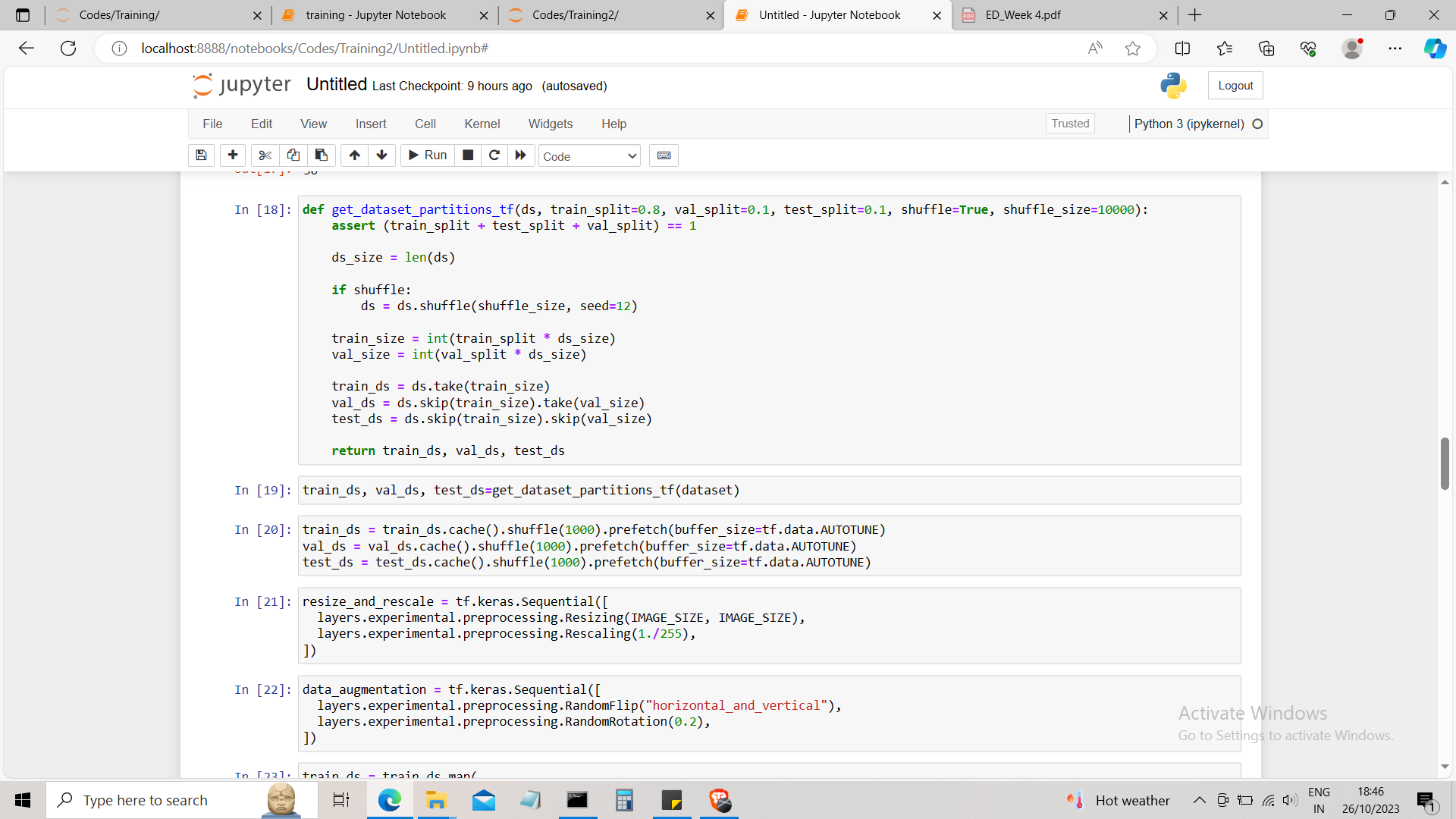


Convolutional layer



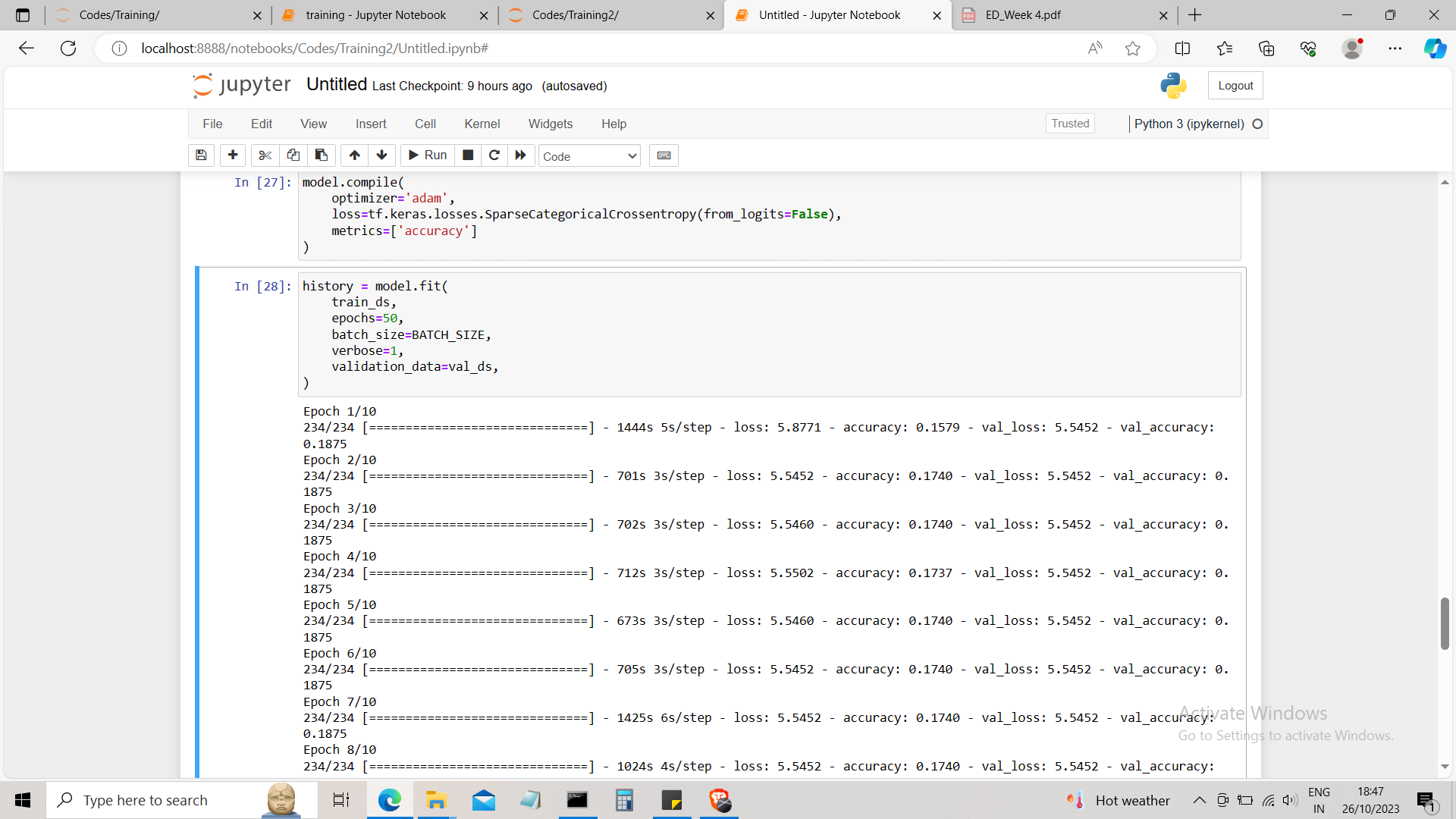
4.training and testing

We had splited the data as 80%-training 10%-testing and 10%-validation



**5.Results**

By training the model with dataset we had got 30 percent of accuracy (which we will improve further)



**6.Future work-**

* Considering the current challenges with achieving optimal prediction accuracy, our future efforts will be dedicated to enhancing the model's performance.
* We plan to enrich the training process by incorporating a more extensive and refined dataset.
* we aim to explore advanced deep learning architectures and algorithms to further refine the model's ability to find out features within Indian cuisine, thus elevating its predictive capabilities.