**Image Classification on Celebreties Dataset**

The image classification model on the Celebrities Dataset was build using Convolutional Neural Networks (CNNs), to identify and classify celebrity images. The model was trained on a dataset of cropped images, with applied data augmentation techniques. The training process includes data preparation, model architecture, compilation, training, evaluation. The trained model and a class dictionary are saved for future use. A sample image is employed to demonstrate how the model predicts.

Data Preparation

* Data Collection

Image files are gathered from the specified directory (path\_to\_data), and a DataFrame (df) is created to organize file paths and corresponding labels (name of the person).

* Data Splitting

The dataset is divided into training, validation, and test sets to facilitate model training and evaluation.

Data Augmentation

ImageDataGenerator is used to apply data augmentation specifically to the training set. Augmentation techniques include rescaling, shearing, zooming, and horizontal flipping. Validation and test sets are solely subjected to rescaling.

Model Architecture

* Neural Network Design

The CNN architecture is structured with convolutional layers for feature extraction, max-pooling layers for dimension reduction, a flatten layer for vectorization, dense layers for classification, and a softmax activation layer for multi-class classification.

Model Compilation

The model is compiled using the Adam optimizer, categorical cross entropy loss (suitable for multi-class classification), and accuracy as the evaluation metric.

Model Training

The model is trained with 10 epochs using the training generator, and the training progress is validated with the validation set.

Model Evaluation

The trained model is evaluated on the test set to determine its accuracy.

Model Saving

The trained model is saved to a file (sports\_person\_model.h5), and the class dictionary (mapping labels to indices) is saved to a JSON file (class\_dictionary.json) for future reference.

Making Predictions

A sample image path (sample\_image\_path) is provided to demonstrate the model's predictive capabilities. The image is loaded, normalized, and expanded to add a batch dimension. The model predicts the class probabilities for the input image, and the predicted class index is mapped to the name of the person using the loaded class dictionary. The predicted name of the person is then printed.