

**Image Scraping and Classification Project**

Submitted by:

Pooja Mishra

**ACKNOWLEDGMENT**

I would also like to thank Flip Robo Technologies and Datatrained team who has given me such wonderful opportunity to learn more about Machine Learning and their mentors Shubham Yadav, Sajid Chaudhary for their constant encouragement, valuable suggestions, and cooperation.

This acknowledgement will remain incomplete if I fail to express my deep sense of obligation to my family members and God for their consistent blessings and believe in me.

Thank You.

**INTRODUCTION**

* Business Problem Framing

Images are one of the major sources of data in the field of data science and AI. This field is making appropriate use of information that can be gathered through images by examining its features and details.

In this project, we’ll be performing image classification in which our system will receive an image as input and now the system is aware of a set of categories and its goal is to assign a category to the image. Here we have taken image data from Google of

* Sarees (women)
* Trousers (men)
* Jeans (men)

That is the idea behind our project here, we want to build a model system that can identify the sport represented in that image. The three classification classes here are Sarees, Jeans and Trousers.

* Conceptual Background of the Domain Problem

Image Classification is the task of assigning an input image, one label from a fixed set of categories. This is one of the core problems in Computer Vision that, despite its simplicity, has a large variety of practical applications.

* Review of Literature

The idea behind this project is to build a deep learning-based Image Classification model on images that will be scraped from e-commerce portal. This is done to make the model more and more robust. The problem we’re dealing with collection of images to recognize it into categories. Here we have taken image data of Clothing wear.

* Motivation for the Problem Undertaken

The large number of images have been collected for each category. What do you think this image represents (Above image)? Hard to guess right? The image to the untrained human eye can easily be misclassified as saree, Jeans and Trousers but technically we can solve this of classification with help CNN i.e. Convolutional Neural Networks. Convolutional neural networks or CNN’s are a class of deep learning neural networks that are a  huge breakthrough in image recognition. . The question now is can we make a system that can possibly classify the image correctly. Yes with help of CNN we can categorise the images to its goal.

**Analytical Problem Framing**

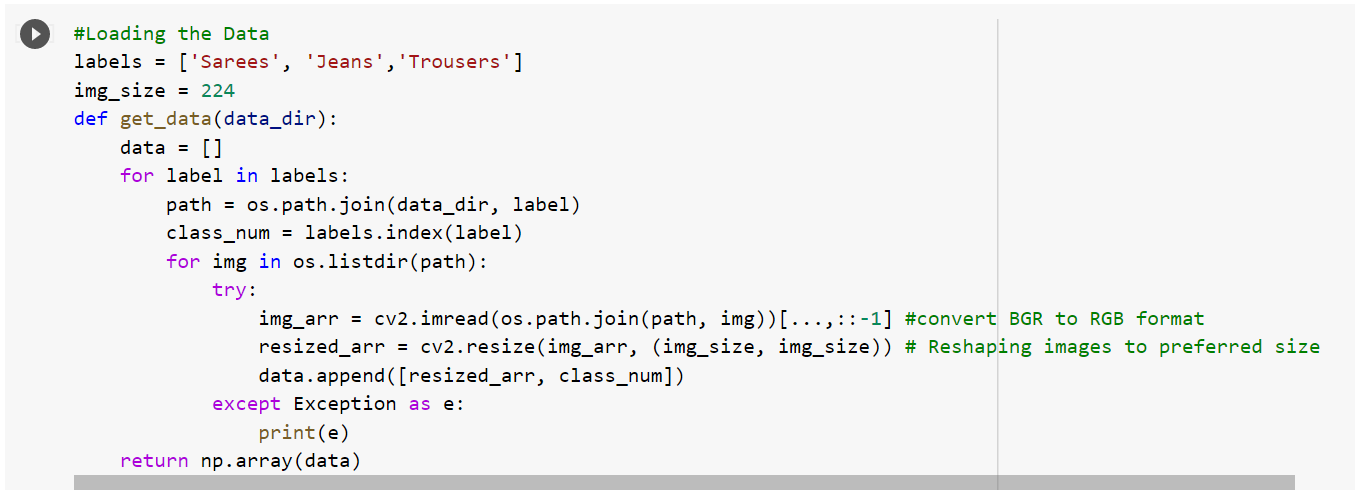
* Mathematical/ Analytical Modeling of the Problem

The mathematical approach required here is the setting up of dataset. A total of 600 images were downloaded, which was divided into train and test. I performed an 80-20 split with the train folder having 480 images and the test folder has 120. Both the classes have sarees, jeans & trousers have 160 images each for train set and 40 iamges each for test set.

* Data Sources and their formats

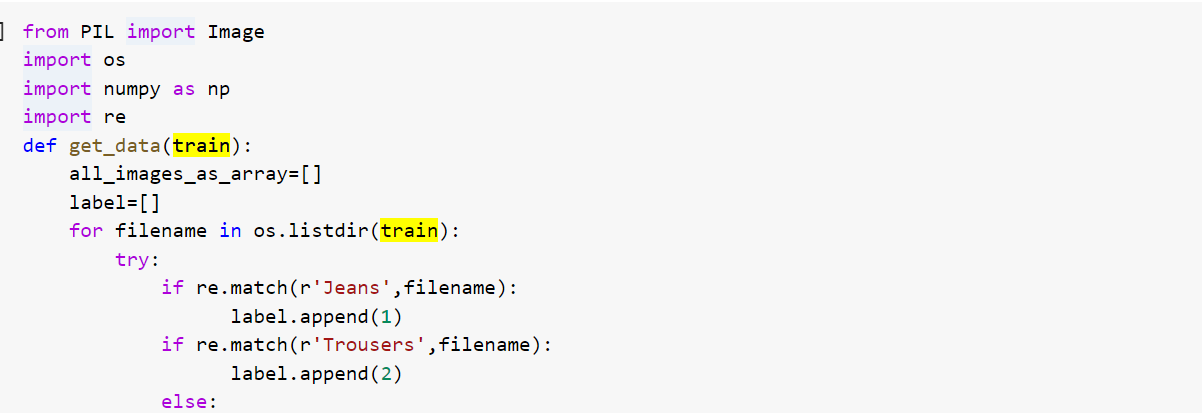
Since we are working on an image classification problem I have made use of two of the biggest sources of image data, i.e, ImageNet, and Google OpenImages. I implemented two python scripts that we’re able to download the images easily. The images have been scraped from Google. Our data structure is as follows:-

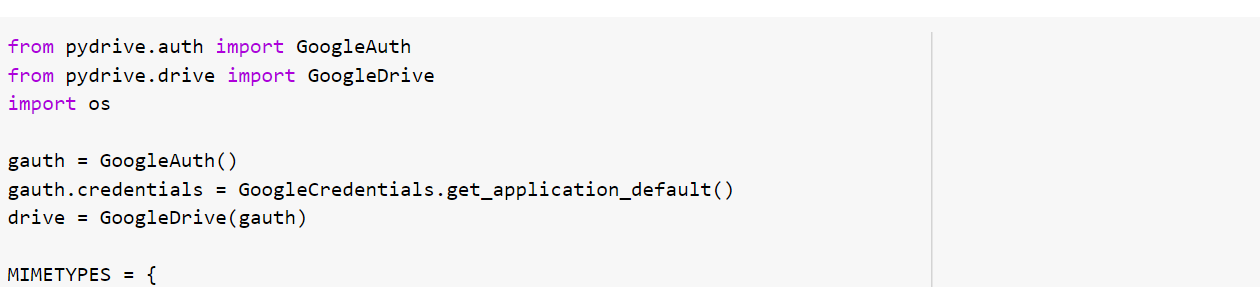
* Input – 600
  + Train – 480
    - Sarees – 160
    - Trousers – 160
    - Jeans - 160
  + Test – 120
    - Sarees – 40
    - Trousers – 40
    - Jeans – 40



* Data Preprocessing Done

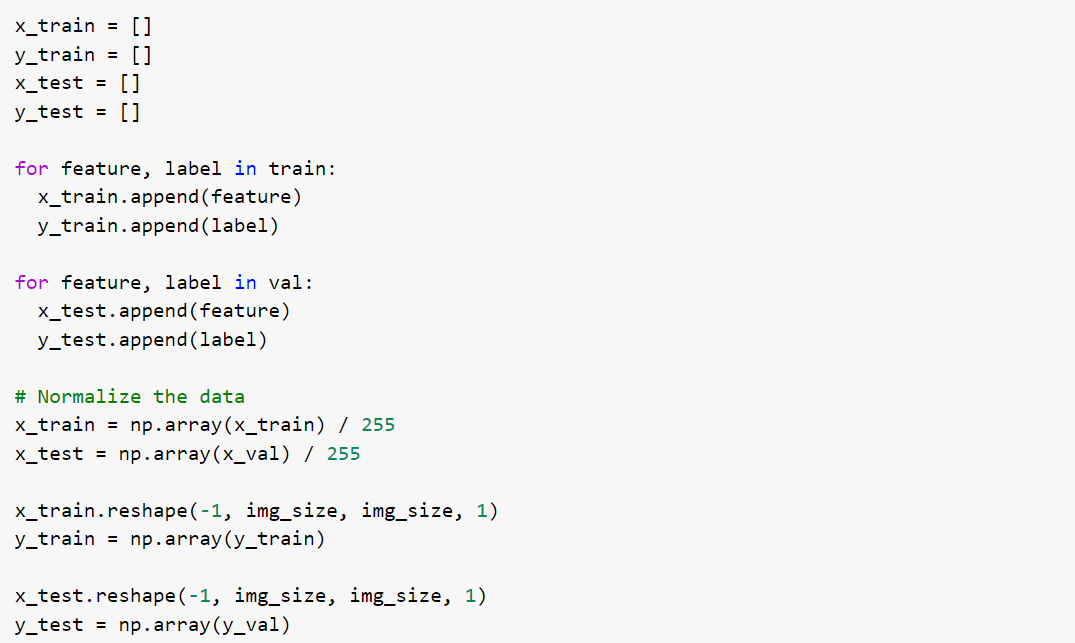
Next, we are performing some Data Preprocessing and Data Augmentation before we can proceed with building the model.





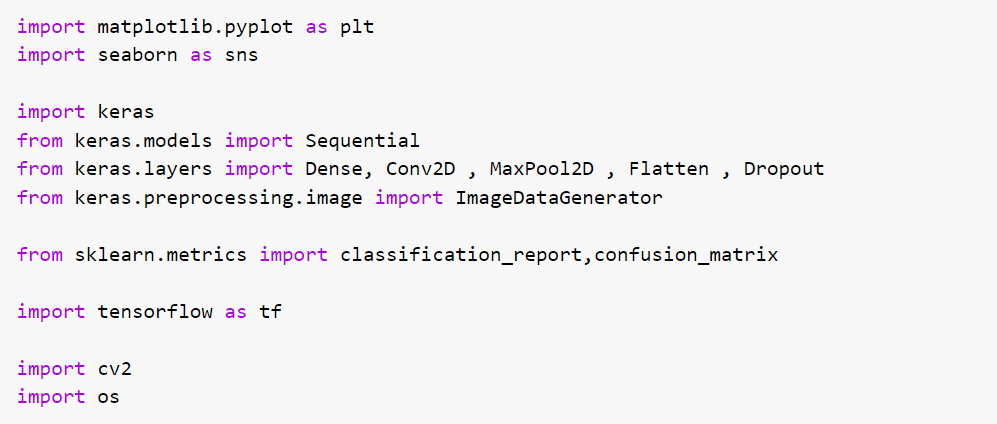
* Data Inputs- Logic- Output Relationships

In Data Input -Logic – Output relationships include the train and test split to perform the model building. With train dataset we have features and labels while with test dataset we have only features to predict the labels.



* Hardware and Software Requirements and Tools Used

Here we will be making use of the Keras library for creating our model and training it. We also use Matplotlib and Seaborn for visualizing our dataset to gain a better understanding of the images we are going to be handling. Another important library to handle image data is Opencv.



**Model/s Development and Evaluation**

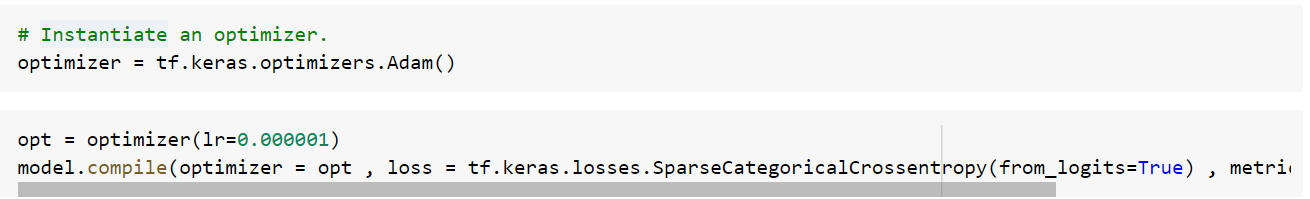
* Identification of possible problem-solving approaches (methods)

To build the model, we’re going to use standard CNN with TensorFlow. The model consists of four convolutional layers before we flatten it and use a dense layer at the very end.



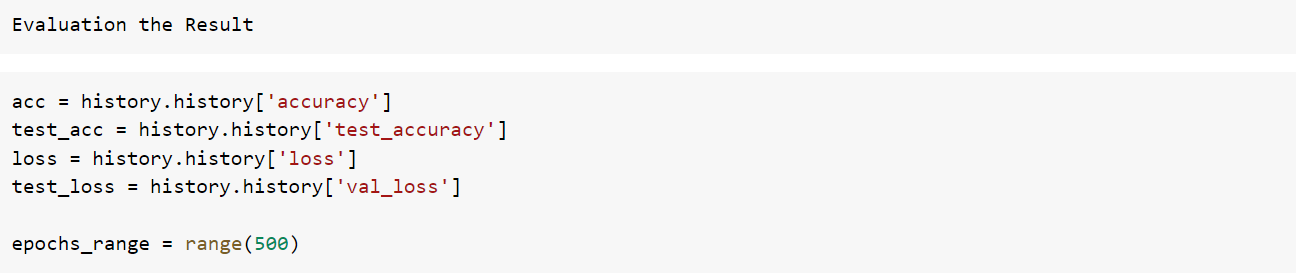
* Testing of Identified Approaches (Algorithms)

 In this article, we will use Adam optimizer for the model. As the problem is a classification problem, we need to use categorical cross-entropy as the loss function. Finally, accuracy metrics will be used.



* Run and Evaluate selected models

Finally, we can now make a prediction based on images in our test set with the trained model. Let’s create a function to preprocess the image that we want to predict and then assign the model to make a prediction based on that image.

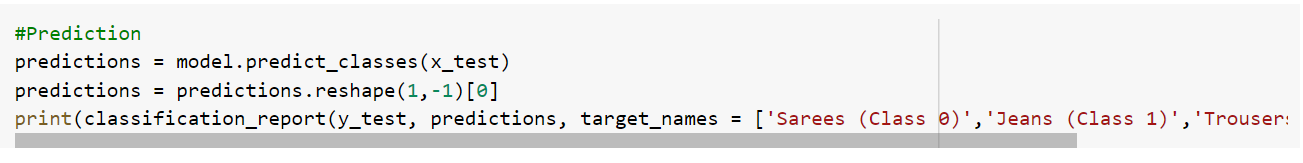


* Key Metrics for success in solving problem under consideration

As we can see with transfer learning we were able to get a much better result. All categories precision are higher than our CNN model and also the overall accuracy reached 91% which is really good for such a small dataset. With a bit of hyperparameter tuning and changing parameters, we might be able to achieve a little better performance too!

* Interpretation of the Results

That is not the end, we saw that our models were misclassifying a lot of images which means that is still room for improvement. We could begin with finding more data or even implementing better and latest architectures that might be better at identifying the features.



**CONCLUSION**

* Key Findings and Conclusions of the Study

All categories precision are higher than our CNN model and also the overall accuracy reached 91% which is really good for such a small dataset. With a bit of hyperparameter tuning and changing parameters, we might be able to achieve a little better performance too!

* Learning Outcomes of the Study in respect of Data Science

With this epistemic uncertainty, our CNN model basically knows what it doesn’t know. It demands us to supply more variations in training data in order for it to classify our rose image with better certainty, like the one with dandelion image. Overall, epistemic uncertainty can be reduced by supplying more training data to the model.

* Limitations of this work and Scope for Future Work

As suspected, the addition of regularization techniques slows the progression of the learning algorithms and reduces overfitting, resulting in improved performance on the holdout dataset. It is likely that the combination of both approaches with further increase in the number of training epochs will result in further improvements.

It may be worth exploring changes to the learning algorithm such as changes to the [learning rate](https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/), use of a learning rate schedule, or an adaptive learning rate such as [Adam](https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/) and more libraries implementation.