

Machine Learning Engineer Nanodegree

Capstone Proposal

Param Aggarwal January 24th, 2019

Proposal

Domain Background

Myntra is a fashion e-commerce company in India that has a very large catalog of products to choose from. One can use their website or app to browse through products and buy them. This catalog is manually photographed and uploaded into the Content Management System (CMS). This presents a challenge regarding the validity of this data. Often incorrect images are uploaded and have to be manually verified and removed from the catalog before publishing to the main website.

I have been an employee of Myntra and have seen these problems first hand. My motivation to resolve this issue comes from the understanding of how much such a system could be useful to the company.

Research Paper for Image Classification using Convolutional Neural Networks:

<https://arxiv.org/abs/1812.01187> (<https://arxiv.org/abs/1812.01187>)

Problem Statement

As of today this approach involves a lot of stages of manual review and approval. This increases the time from when an item is photographed to when it goes live on the website and can contribute to lost revenue. Fashion is a fast moving industry and new trends have to hit the market as soon as possible. Any attempts at speeding up the review process can help directly contribute to an increase in revenue.

This is a classification problem and not regression. We predict the classes of the uploaded images and the training is done in a supervised manner using Convolutional Neural Networks.

Datasets and Inputs

The dataset is scraped from the Myntra website for existing products that are listed. This script was written by me and will be part of the final project submission. Hence we also have the labels for these images. For example we know whether the item is a 'topwear' or 'bottomwear'. The dataset consists of 2000 images with half and half split between topwear and bottomwear. The images are in the jpeg format and are colour images. While scraping,

the images are fetched in a specific resolution of 60x80px.

The **inputs** will be images in jpeg format in a suitable resolution with the product identifier as the filename. We will read these image files and convert them into numpy arrays to give as inputs to our network.

The **outputs** will be prediction of labels associated with these images. For training we will use the separate JSON file which contains the attributes of the image, for example, the category of the clothing. This will be used as labels for our supervised machine learning.

Solution Statement

We will be training a Convolutional Neural Network supervised using the category labels in our dataset. We will be using Keras for this and will be using the accuracy score as the metric or optimizing our network.

Benchmark Model

As of today this verification is a manual approach. Hence the accuracy of a random classifier will be 0.5 or 50%. Our aim in building our neural network classifier is to beat this benchmark.

Evaluation Metrics

We will be evaluating our network performance by using the accuracy score obtained on this dataset by Keras. As mentioned above, our goal is to beat the 50% accuracy that a random classifier might have.

Project Design

We need to go through our data once and check that the downloaded dataset images are not corrupted. We will then build our Keras model layers step by step and come up with a Convolutional Neural Network that is trained with the category labels as the supervised learning parameters.

The preprocessing step will involve reading the jpeg files and decompressing them and converting them into 60x80x3 numpy array with three colour channels. We do not need to resize the images as they are already fetched at this resolution itself. While the dataset is small, the network architecture has also been kept small to make sure training happens quickly. The size of the dataset can be increased by scraping more images from the website.

With the current set of data, we will be using a standard convolutional network of two or three layers and small filter sizes. We will not be using transfer learning unless we are able to increase the size of our dataset by scraping even more images from the catalog.

Once we are done with this, we will use some random images from the website to simulate how the system would work for verification of newly uploaded images and point out errors that would be caught before the product goes live on the website.