LadenDirekt furniture detection pipeline proposal

November 11, 2018

1 LadenDirekt furniture detection pipeline

1.1 Domain Background

The project will be performed for LadenDirekt - website aggregator of furniture, men, and women fashion. The company collects the information from partners and pushes it through their interface to customers willing to shop for advertised products on the German market.

1.2 Problem statement

The goal of Data science project is to collect the data acquired from vendors(scrapped from their DB), download pictures of furniture and try to build the CNN that detects different kind of furniture, communicate detection with the DB and push the proposed items from DB to the client. This pipeline will be applied in the web app that can be distributed by company to clients.

Use case. Client is willing to shop for "table" and is browsing around shops, trying to find desired product, color, price etc. Then the client opens up the app on his/her phone, makes a picture of the item through this app. The CNN identifies the item and pushes the most alike item from the listings to the customer.

The ultimate goal is to build the pipeline that will figure out each of 877 categories. Although, it is almost impossible to create such a model the approach will include working step by step. So the main idea is to build algorithm that will pick at least couple major categories like tables, chairs, chest of drawers, etc.

1.3 Datasets and inputs

Data set contains about 313000 pictures of items currently available in DB. There are 877 categories of products identified by LadenDirekt team. Some of the categories contain up to 12000 pictures, but some will only have 1 or 2 pictures for the category. Although, it is great to have so many files available for the project time and machine power constraints won't allow to process all the files. So decision was made to download only up to 150 pictures per each category. For those categories that have only 1 or 2 files the data augmentation will be used to produce more viable data set. the resulting data set will consist of about 83000 pictures.

1.4 Solution statement

The solution would be to try several Convolutional Neural Networks to predict the specified categories . Firstly for major categories like tables, chairs,chest of drawers and then apply the knowledge on other categories. There were two Kaggle competitions performed on Fashion and Furniture CNN recognitions.

1.5 Benchmark model

Well. i think that furniture/fashion identification is a bit tough task to achieve. There were several competitions on Kaggle (iMaterialist Challenge (Furniture) at FGVC5 and iMaterialist Challenge (Fashion) at FGVC5) that tried to achieve the most efficient result in identification Furniture, Fashion. I will try to follow along with authors suggestions to work my way through.

1.6 Evaluation metrics

For this implementation each image has one ground truth label (category). An algorithm to be evaluated will produce 1 label per image. If the predicted label is the same as the ground truth label, then the error for that image is 0, otherwise it is 1. The final score is the error averaged across all images.

1.7 Project design

As stated above I will try to follow along the solution obtained by Lei & Dowakin during iMaterialist Kaggle Challenge (Furniture) at FGVC5.

The implementation will use pytorch to do all the training and testing. The pretrained models will be taken from the Cadene's github repository https://github.com/Cadene/pretrained-models.pytorch.

The guidelines on the code implementation will be taken from Roman Dowakin github repository https://github.com/skrypka/imaterialist-furniture-2018 .

The models that would be used for modeling: - inceptionv4 - densenet201 - densenet161 - dpn92 - xception - inceptionrResNetv2 - resnet152 - senet154 - renext101 - nasnet.

For training There will be used SGD in most cases. The learning rate of the fully connected layer is ten times than others. The initial lr is 0.001 and switched to 0.0001 later. Though, for SEnet, it is hard to train with SGD, so Adam would be used first.

For testing, there will be 12TTA used = (normal+horizontal flip)6 crops(full image, center, 4 corners) for each model and save the result. Then, the gmean will be used to get the final result for the training set as the result is greatly improved in this way.

Then try ensembling the trained models to achieve the best result.

1.7.1 Work process

1. Get the data - write a script to download about 150 pictures per each category
2. Perform image augmentation for the categories that have small image quantities.
3. Split the data into Train/Validation/Test sets
4. Use pretrained models to identify the best net to work with.
5. Try ensembling
6. All the testing is performed on error comparison to ground truth labels