Machine Learning Nanodegree

Jens Laufer

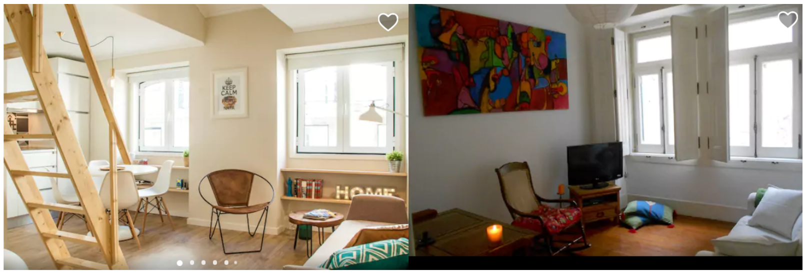
Juli 20, 2018

Capstone Proposal

Classifier for Image Aesthetics

1. Domain Background

Which place to book?



Images are a very important factor in making decisions in different domains like e.g. holiday booking, dating, ordering food or HR etc. But what aspects of an simage are influencing our decision? It's most probably a combination of low level aspects like the image quality (resolution, lightning, color etc), medium level aspects (composition of the image, objects in the image) and high level aspects (Do I like the bed?, Is this person trustful?). The high level aspects seems to be more domain specific than the low level aspects. The decision is also driven by our personal experience and weighted by each person differently. Maybe there are features of an image our brain notices, but we are unaware of them.

Several studies focused on the low level aspects of image aesthetics on the one hand (Lu et al. 2015), (Murray, Marchesotti, and Perronnin 2012a), (Kong et al. 2016), (Lu et al. 2014), (Lu et al. 2015), (Schwarz, Wieschollek, and Lensch 2018) and on the other hand there are studies which examined rather all the aspects in a certain domain. (Zhang et al. 2017), (Nguyen et al. 2017)

1.1 Personal Motivation

I started recently a business that provides data services for Airbnb hosts and investors (bnbdata.co). I want to provide a rating service for the images in an Airbnb listing, as the quality of images is very important for the booking decision. The public available datasets for image aesthetics are consisting of a wide range of images from different areas therefore I want to concentrate in a first MVP (Minimal Viable Product) on rather low level aesthetics aspects than on high level features for the hospility industry. In later MVPs (not part of this project) I want also take higher level domain specific factors into account. But for this I have to collect my own annotated image dataset with Airbnb listing images.

2. Problem Statement

Judging the aesthetics of an image is a very subjective task and instead of being clear it's rather fuzzy, with many differently weighted hidden features. There are different ways to quantify the aesthetics of an image: Images could be presented to people (e.g Amazon Machanical Turks) who rate the image and the mean/median from the ratings is taken as metric. It's important to have a sufficient number of ratings for this method. Another way is to take photos from social image platforms like Flickr. People can add a photo by a single click to theirs favorites list. A metric of the aesthetics of a photo could the ratio between favs and views.

It is difficult to tackle the problem with classical programming as so many fuzzy aspects play a role as said before. The problem seems to be suitable for the supervised machine learning space.

3. Datasets and Inputs

There are requirements for datasets for image classifications tasks in general and the image aesthetics predictions in particular:

- Sufficient Number of images in each rating class (~1000)
- Sufficient Number of people who rated the aesthetics of each individual photo to get a robust mean/median rating for each image

There are several datasets which seem appropriate for the problem:

Dataset	Reference	Number of images	Description
AVA	(Murray, Marchesotti, and Perronnin 2012b)	>250k	Aesthetic Visual Analysis (AVA) contains images along with a rich variety of meta-data including a large number of aesthetic scores for each image
AADB	(Kong et al. 2016)	~10k	Aesthetics and attributes database (AADB) contains aesthetic scores and meaningful attributes assigned to each image by multiple human raters.
AROD	("Will People Like Your Image-Arod Dataset" 2018)	~380k	Photos are scraped from Flickr with the number of views and favs

Which one of the dataset will be used needs to be evaluated in a exploratory data analysis (EDA). As the distributions of all three datasets of the ratings are normal (most ratings are in the midrange) additional creation of images through augmentation might be needed. Although not all augmentation techniques might be suitable, as they might have an effect on the aesthetics of an image. E.g the image might be flipped horizontally, but flipping vertically would not result in the same rating.

Having three datasets is interesting for doing cross-dataset evaluations with the final classification models.

4. Solution Statement

A Convolutional neural network (CNN) is a class of deep, feed-forward artifical neural networks, most commonly applied to analyzing visual imagery. It's inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the human visual cortex. (Wikipedia 2018)

A Convolutional neural network (CNN) seems to be a good candidate to solve the problem. The images are used as input for the CNN and the rating metric is used as the output variable.

The problem will be treated as a classification problem by binning the continuous rating metric into several buckets.

5. Benchmark Model

Accuracies of different models on the AVA dataset are reported in different papers. These accuracies are used for benchmarking the models which are created in this work.

Model	Reference	Accuracy
Murray	(Murray, Marchesotti, and Perronnin 2012a)	68.00 %
Reg	(Kong et al. 2016)	72.04 %
DCNN	(Lu et al. 2014)	73.25 %
DMA	(Lu et al. 2015)	74.46 %
Schwarz	(Schwarz, Wieschollek, and Lensch 2018)	75.83 %

6. Evaluation Metrics

The problem is a classification problem. The performance can be quantified with several metrics (f-beta, Recall, Precision, Accuracy). To compare the classifier against the benchmark model(s) from section 5 the **Accuracy** is used. The Accuracy is the ratio of correct predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$TP : TruePositives$$

$$TN : TrueNegatives$$

$$FN : FalseNegatives$$

$$FP : FalsePositives$$

7. Project Design

7.1 Preparation

The first step is to setup an environment for the project. There are several ways for doing this:

1. Local Anaconda environment with Juypter
2. Anaconda environment with Jupyter in the cloud on AWS, GCloud or MS Azure
3. FloydHub cloud environment with Jupyter
4. Docker container with Anaconda environment with Jupyter which can be deployed with Kubernetes locally, on AWS, GCloud or MS Azure

To get a reproducable setup for different runtime environments option 4 is preferred. A research is needed if this option is working with GPU empowered cloud environments at a reasonable price. A goal is also to have reusable modules/scripts which can be used also independently from Jupyter notebooks.

7.2 Retrieval of metadata for the datasets

First the metadata (without images) for the different datasets will be retrieved for a first analysis. This is done to reduce the overhead of downloading all the image files for datasets which are not used at the end.

7.3 Exploratory Data Analysis (EDA) on metadata

The metadata of the datasets needs to be analyzed to make sure if a dataset is a candidate for model training or not. It's important that each class have a sufficient number of images and all classes are having approximately the same number of images. At the end one dataset is selected for model training. From the literature the AROD dataset seems the most promising candidate. The AVA dataset is also needed for benchmarking the models against the models from literature.

7.4 Retrieval of images for the dataset(s)

In a second step the images needs to be retrieved. This might be challenging for the bigger datasets (AVA, AROD) which consist of a huge number of images. Another challenge is to reduce costs while downloading the images: Paying for GPU on a cloud platfom in this stage should be avoided. If the images are not prepacked they have to be downloaded (e.g. from Flickr) with scripts. This can be time consuming and there is also a risk of corrupted images or a IP might be blocked as the provider of the images might think of a DoS attack. Another problem might be that a image is not available anymore.

7.5 Data auditing and cleansing

The image data and metadata needs to be audited to ensure high data quality. Corrupted entries/images need to be removed. The images in the AVA dataset are e.g. rated by Amazon Mechanical Turk. The active involvement of people might affect data quality.

7.6 Feature engineering

New features might to be engineered. The AROD dataset e.g has for each image the views and likes from the Flickr platform. A score might be the ratio between likes and views. One problem is that there is no linear relationship between views (and likes) and the time. A way to deal with this needs to be found in case we are using this dataset.

7.7 Model design

In this step a model is created with Keras. The goal is to use transfer learning to not reinvent the wheel.

7.8 Data prepocessing

The images need to be preprocessed for "feeding" the neural network for training. The images might be resized or/and cropped without losing too much information and transformed into a tensor. Additional images might be created with augmentation. The data also needs to be split into a training and test set.

7.9 Model training

The model will be trained against the training set. The challenge is here to find a balance between training time, which can cause high costs while training the model in the cloud on the one hand and finding a robust model with high accuracy on the test set. The goal is to have a fast feedback loop/fail fast environment to be able to perform many model tuning iterations. It might be required to reduce training time with a reduction of the training data or run the training on more GPU.

7.10 Model testing

The model needs be tested for performance against the test set and checked for under-/overfitting.

7.11 Model fine tuning

In case of under-/overfitting the model needs to be finetuned and retrained (step 8.)

7.12 Model benchmarking

The model needs to be crosschecked against the AVA datset to check it's performance in comparison to the models described in different papers which are all checked gainst this dataset. (5. Benchmark model)

8. References

Kong, Shu, Xiaohui Shen, Zhe Lin, Radomir Mech, and Charless Fowlkes. 2016. "Photo Aesthetics Ranking Network with Attributes and Content Adaptation." In *European Conference on Computer Vision*, 662–79. Springer.

Lu, Xin, Zhe Lin, Hailin Jin, Jianchao Yang, and James Z Wang. 2014. "Rapid: Rating Pictorial Aesthetics Using Deep Learning." In *Proceedings of the 22nd Acm International Conference on Multimedia*, 457–66. ACM.

Lu, Xin, Zhe Lin, Xiaohui Shen, Radomir Mech, and James Z Wang. 2015. "Deep Multi-Patch Aggregation Network for Image Style, Aesthetics, and Quality Estimation." In *Proceedings of the Ieee International Conference on Computer Vision*, 990–98.

Murray, Naila, Luca Marchesotti, and Florent Perronnin. 2012a. "AVA: A Large-Scale Database for Aesthetic Visual Analysis." In *Computer Vision and Pattern Recognition (Cvpr), 2012 Ieee Conference on*, 2408–15. IEEE.

———. 2012b. "AVA: A Large-Scale Database for Aesthetic Visual Analysis." https://github.com/mtobeiyf/ava_downloader.

Nguyen, Laurent Son, Salvador Ruiz-Correa, Marianne Schmid Mast, and Daniel Gatica-Perez. 2017. "Check Out This Place: Inferring Ambiance from Airbnb Photos."

Schwarz, Katharina, Patrick Wieschollek, and Hendrik PA Lensch. 2018. "Will People Like Your Image? Learning the Aesthetic Space." In *Applications of Computer Vision (Wacv), 2018 Ieee Winter Conference on*, 2048–57. IEEE.

Wikipedia. 2018. "Convolutional neural network — Wikipedia, the Free Encyclopedia." http://en.wikipedia.org/w/index.php?title=Convolutional%20neural%20network&oldid=849614684.

"Will People Like Your Image- Arod Dataset." 2018. https://github.com/cgtuebingen/will-people-like-your-image.

Zhang, Shunyuan, Dokyun Lee, Param Vir Singh, and Kannan Srinivasan. 2017. "How Much Is an Image Worth? Airbnb Property Demand Estimation Leveraging Large Scale Image Analytics."