# Deep Learning for Shallow Concerns: Predicting Photo Popularity

Ivan Bogatyy
Google
111 8th Ave, New York, NY 10011
ivanbogatyy@gmail.com

Michael Tom*
Radiate Inc.
315 Hudson St, New York, NY 10013
michael@radiatetheworld.com

## Abstract

*We explore the problem of predicting photo attractiveness (measured by ratio of "likes" to "dislikes" on a social app). Using pre-trained convolutional neural network models and a transfer learning approach, we obtain significant progress using a relatively small dataset (ten-fold MSE loss reduction over primitive baseline). We compare two base models (Inception and FaceNet), showing that a more relevant base task (facial recognition) yields significant gains over more generic one (ImageNet classification). Further, we build a classifier predicting gender from photo, with findings further confirming previous two points.*

## 1. Introduction

Inspired by Andrej Karpathy's experiments [4], we explore the problem of building a machine-learned model to predict photo attractiveness. His approach was to download publicly available selfies from a social network (likely Twitter) and use some heuristics to approximate how good or bad a selfie was given how many people have likely seen it and how many of those "liked" it. After building this dataset of photos labeled either "good" or "bad" (by bucketing the top $50\%$ and bottom $50\%$ for every person), a transfer learning setup was used to approximate the labels, with a VGG [9] network pre-trained on either ImageNet [7] or unspecified facial data. Interestingly, no benefit was found from pre-training on facial data.

Our approach uses a similar transfer learning setup, while having several important distinctions:

- We benefit from using robust labels available through proprietary data.
- Having more reliable labels, we have a more fine-grained prediction setup and loss function (MSE regression).
- We inherit from a more accurate Inception v3 rather than VGG, as well as from a SotA facial recognition model [8].

To elaborate on the first bullet point, our dataset consists of users' photos and a log of all "swiping" events (a user can "swipe right" on another user, meaning "like", or "swipe left", meaning "ignore"). The swiping events are aggregated per recipient to produce "like ratios", which serve as photo labels.

## 2. Related Work

There are many interesting computer vision problems related to human subjects in images and video. Facial recognition is the most widely studied, with Google's FaceNet [8] and Facebook's DeepFace [10] nearing perfect results on Labeled Faces in the Wild dataset [3] and YouTube Faces DB [11].

Further, there has been work on predicting various properties of the human subject on a photo. CelebFaces dataset [5] has been used to predict various facial attributes like eye color, hair color, facial hair, face shape and so forth. Another prominent application is predicting emotional expression on the picture, benchmarked on the Emotions in the Wild dataset [1].

Finally, and most relevant to this project, there has been some exploration [4] of predicting selfie popularity using mined social network data. After finishing the project, we found another academic paper exploring this subject [6][1], coincidentally also using proprietary data from a different social app.

## 3. Methodology and Data

### 3.1. Framework

As outlined in the introduction, we use the ratio of the number of "likes" (aka "right swipes") to the total num-

*Michael Tom is not enrolled in CS231N.

[1]Featured in TechCrunch https://techcrunch.com/2016/01/11/blinq-dating-app-uses-ai-to-judge-hotness/

ber of swipes ("likes" and "dislikes") as labels for recipient photos.

Most users have more than one photo, and their respective impact is hard to quantify (e.g. for learning Web searching ranking from clicks, normalizing URL click ratios to account for their position within the SERP top-10 is its own area of research, which we do not intend to replicate here). Undoubtedly, the first photo has the highest impact (since scrolling to subsequent photos requires interrupting the flow), so we limit our data to only contain the main photo of each user, and that user's "like ratio" is decided to be the photo's label.

Given a photo and no other information, our goal is to predict that label, with mean squared error (MSE) serving as training loss. We weight the loss by the total number of swipes a given user has to account for information disparity across users.

As a simpler yet interesting experiment, we also attempt to predict gender, using log-loss for training and classification accuracy for evaluation. The rest of the setup mirrors predicting like ratios.

To clarify, we note that gender information (true or predicted) is not supplied during "like ratio" training and evaluation.

### 3.2. Model

Our models are structured as follows. Raw images (all of them are square per app requirements) are re-sized to $299 \times 299$, followed by the Inception v3 architecture (note: to make models directly comparable, we chose a variant of FaceNet based on Inception v3 architecture). After that, we take either the $PreLogits$ endpoint of Inception v3, or the facial embeddings (final layer) of FaceNet. On top of that representation, either zero, one or two fully connected layers are built, followed by a single prediction neuron, either like ratio (with MSE loss), or gender probability (with log-loss).

### 3.3. Dataset

Our dataset contained roughly $85k$ users and $40m$ swiping events. After filtering out users with no photos, we were left with $76k$ users.

Swiping events had to be filtered too. Given that users change their pictures from time to time, and the swiping events do not explicitly keep the relevant picture in the log, we had to determine the last time every user updated their main photo, and then throw out any swiping decisions on them that happened before that. After this filtering, we were left with $11m$ swiping events, which were used to produce labels.

Table 1. Average like ratios by sender's and recipient's genders.

| * | $\rightarrow$ F | $\rightarrow$ M |
|---|---|---|
| F $\rightarrow$ | 0.33 | 0.24 |
| M $\rightarrow$ | 0.75 | 0.31 |

### 3.4. Dataset breakdown

The initial set of users $85k$ users partitions into $32k$ females and $53k$ males $(37 : 63)$, with $7.8m$ and $32m$ total swipes made by females and males respectively. After keeping only the users with photos, we are left with $32k$ females and $44k$ males $(42 : 58)$.

We also explore the users' swiping patterns, grouping average like ratios by the sender's and recipient's genders in Table 1.

### 3.5. FaceNet applicability

The FaceNet model relies on being supplied a tightly bounded image of a face, with no background. To that end, they preface running their model by running a facial detector and producing a bounding box to crop the input image (c.f. [8], top of Section 5).

We replicate the setup, but since some photos happen to produce zero bounding boxes, wheres others (including many group photos) produce more than one, these photos are unfit for FaceNet. We thus create a separate evaluation setup, limiting our dataset to only contain those photos that have exactly 1 bounding box. This data is independently shuffled and divided into train/dev/test.

Since a direct comparison would be unfair (the latter dataset is likely to be easier to learn on), we evaluate Inception on both setups.

## 4. Experiments

### 4.1. Setup

We store photos in an HDF5 file after decoding and resizing, requiring approximately $76k \times 300 \times 300 \times 3 = 20GB$ of space. We train zero, one or two fully-connected layers on top of the representation network. While fine-tuning by back-propagating into the network would have been interesting, we did not yet explore that, instead only enabling the added layers for training.

We use Adam training (SGD produced worse results) and use the dev-set to decide when to stop and to pick the best learning rate.

We use mini-batch of $256$, up to $20$ training epochs, and learning rate decay following [2], i.e. decaying $100$-fold by the end of training.

Table 2. MSE losses on test.

| Model | ALL | 1BBOX |
|---|---|---|
| Const baseline | 0.187 | 0.163 |
| Inception-0FC | 0.0252 | 0.0193 |
| Inception-1FC | **0.0227** | 0.0195 |
| Inception-2FC | 0.0228 | 0.0192 |
| FaceNet-0FC | - | 0.0183 |
| FaceNet-1FC | - | **0.0121** |
| FaceNet-2FC | - | 0.0129 |

Table 3. Gender classification accuracy on test.

| Model | ALL | 1BBOX |
|---|---|---|
| Const baseline | 63.3 | 58.6 |
| Inception-0FC | 86.0 | 90.4 |
| Inception-1FC | **86.9** | 91.1 |
| Inception-2FC | 86.5 | 91.1 |
| FaceNet-0FC | - | 90.4 |
| FaceNet-1FC | - | **95.9** |
| FaceNet-2FC | - | 95.4 |

### 4.2. Results

For like ratio prediction, MSE losses are show in 2. For gender prediction, accuracies are show in 3.

## 5. Subjective results

For ratio predictions, we display photos ranked by model predictions in Figure 1 (captioned by their true labels, that is, true like ratios and genders).

For gender predictions, we were interested in visualizing classification errors. We show top 8 highest log-loss test-set photos for Inception in Figure 2 and for FaceNet in Figure 3. Interestingly, there is plenty of group photos even for FaceNet, meaning the detection and bounding box algorithm has room for improvement as well.

## 6. Submission Metadata and Grading

### 6.1. Contributions

Michael Tom gracefully provided raw data for the project (user photos, gender information and individual "swipe" events log) from his social app for music festivals, Radiate Inc. He is not enrolled in CS231N. Other work on the project (processing and filtering the data, model design, training and evaluation, visualizations, writeup) was done by Ivan Bogatyy.

This was not submitted to any conferences or used as a dual project for any other class.

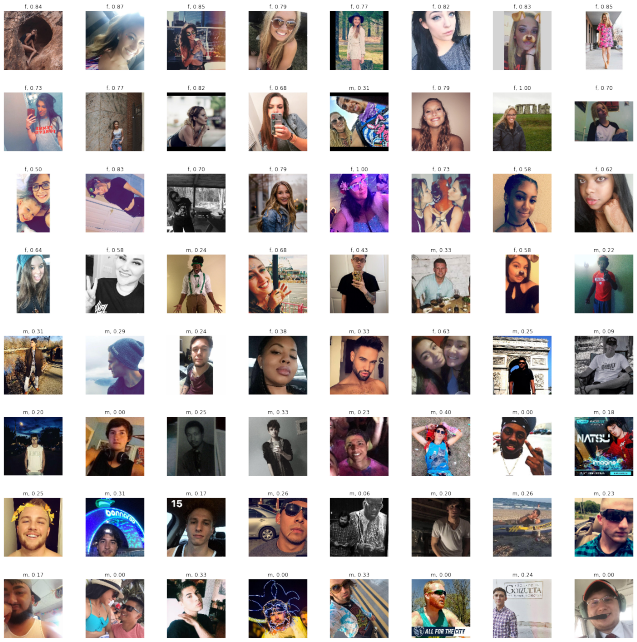

Figure 1. User photos ranked by model's predictions. Rows are quantiles, and columns are samples within (almost) same value. Down-scaled for space considerations, but quality preserved for zooming in.

Figure 2. Inception test data ranked by highest gender log-loss.

Figure 3. FaceNet test data ranked by highest gender log-loss.

### 6.2. Code used

This project used the TensorFlow code[2], including the Slim library, in particular the import that defines the Inception v3 architecture, as well as the Inception v3 checkpoint[3].

Further, it used Google's proprietary FaceNet code, similar to the one publicly available[4], except that it was built around Inception v3 architecture (the publicly available code and checkpoint use Inception ResNet v1, which reduces comparability). In particular, we used facial detection code similar to the publicly available[5] before running feature extraction code[6].

Finally, various Jupyter code snippets from the assignments, in particular

[2]https://github.com/tensorflow/tensorflow
[3]https://github.com/tensorflow/models/tree/master/inception
[4]https://github.com/davidsandberg/facenet
[5]https://github.com/davidsandberg/facenet/blob/master/src/align/detect_face.py
[6]https://github.com/davidsandberg/facenet/blob/master/src/models/inception_resnet_v1.py

`NetworkVisualization-TensorFlow.ipynb`, were used as convenient building blocks for my own visualizations.

## 7. Conclusions

Our results show that the problem of predicting photo attractiveness is clearly amenable to transfer learning computer vision methods. MSE loss is reduced roughly tenfold by training a ConvNet model (compared to a baseline predicting a single best constant), and predictions roughly match subjective judgements. Further, contrary to previous findings, we find that using a neural network pre-trained on a more relevant task yields significant improvement, even for same architecture.

Gender prediction can be performed with accuracy ranging from $86.9\%$ in the most general setup to $95.9\%$ in a somewhat constrained setup. Note that headroom is considerably below $100\%$ given group photos and mislabeled photos.

## References


[1] P. E. Griffiths and A. Scarantino. Emotions in the wild: The situated perspective on emotion, August 2005.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[3] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[4] A. Karpathy. What a deep neural network thinks about your #selfie. http://karpathy.github.io/2015/10/25/selfie/.

[5] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[6] R. Rothe, R. Timofte, and L. Van Gool. Some like it hot - visual guidance for preference prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[7] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[8] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.

[9] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[10] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[11] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.