To Post or Not To Post: Using CNNs to Classify Social Media Worthy Images

Lauren Blake
Stanford University
lblake@stanford.edu

Abstract

This project considers the feasibility for CNN models to classify photos based on whether they depict the subjects best self, which is roughly equivalent to whether the subject is willing to share the photos on social media. Transfer learning based on a pre-trained ResNet-18 model is used for image classification. A new data set is created specifically for this purpose, which includes photos labeled with a social media worthiness score. Initial results are promising with accuracy over 50%. Saliency maps suggest that classification is driven by relevant facial features. To provide more robust and generalizable results, the next step would be to collect a broader data set and determine whether the initial results hold.

1. Introduction

Many people are picky about how they are depicted in photographs. They are upset if they blink, look away from the camera, are not smiling naturally, among many other concerns. This problem is exasperated by how it is difficult to review photos while posing for them and social norms to frequently post on social media.

There is an interesting potential for convolutional neural networks (CNNs) to classify images that individuals believe show their best self versus those that don't. Best self can be roughly defined as photos the individual in the photo finds flattering and would be comfortable distributing broadly through social media. Therefore, the models would automate an existing human process to determine which photos we like of ourselves and are willing to share.

For these models, the input is photos of faces and the output would be a social media worthiness score representing to what extent that photo represents the subject's best self measured on a 1-5 scale.

If successful, these models could be applied to take higher quality photographs more efficiently. For example, the models could be used to help give real-time feedback to the person using the camera, to select likely favorite photos from a photo album or from a burst of photos taken from a single shot, or to make cameras more intelligent (e.g. to tell it when to take the photo).

Model applications would build off existing technologies like face detection and smile detection which have already significantly improved photo taking. At the CS231N project fair, an Apple software engineer on the photos team was enthusiastic about combining these CNN models with their existing features. He mentioned that identifying photos of the subject's best self fits with their near-term priorities. In particular, he could see these models improving the memories feature which use machine learning to create albums around specific events, time periods, locations, or people (e.g. "best of last three months" album, "Japan" album).

2. Related Work

Based on a careful review of relevant publications, the academic literature has not covered classifying images of best self or images that are well suited for social media. However, parallels can be drawn to identifying other abstract qualities in images. There is an established literature on using machine learning to identify attractiveness in images. For example, Gray et al trained neural networks to recognize "female facial beauty" [3]. An even more active area of research is identifying the photo subject's facial expression or emotion. Many of these papers, including Lawrence et al, rely on CNNs for classification and augment CNNs with additional models for facial feature extraction [9]. Similar to this project's use of personal photos, Levi and Hassner address classifying emotions in real-world photos where lighting conditions are problematic and first require significant data processing [2]. (Although not considered to date or listed as a next step, improving lighting conditions or other parts of photo quality may be another way to improve classification results.)

As discussed in the method sections, this project relies on transfer learning based on feature extraction, which is common in facial recognition, facial detection, and facial verification [10, 11].

3. Methods

Because of the small data set size, transfer learning was used to train the CNN models. Specifically, the ResNet-18 model, which was pre-trained on ImageNet data, was used as a fixed feature extractor. Then, a separate model was trained with the features the ResNet-18 model extracted from this project's data set as the input and the social media worthiness scores as the output.

The ResNet model was developed by a research team at Microsoft in 2015 and is unique for modeling residual values in its layers [5]. The ResNet-18 includes in order: 1 convolutional layer, 16 convolutional layers separated into 4 larger block layers, and 1 fully connected layer (excluding layers that don not require weight training e.g. max-pooling layers). The final fully connected layer is removed entirely when using this model as a fixed feature extractor. The ResNet model is well suited to detect appropriate features from this project's data set because it has been trained on the extensive ImageNet data set which includes similar images (ImageNet contains 952K photos in the person subcategory).

To create the fixed feature extractor, the existing last layer of the ResNet-18 model was removed and a new separate model was used to generate the predicted social media worthiness score labels. Two different models were considered for following the ResNet-18 model and generating the score labels. These models approach social media scores either discretely or sequentially and consequently use different loss functions.

The first model uses the cross-entropy loss function.

$L_i = -\log(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}})$

$L = \frac{1}{N}\sum_{j \neq y_i} L_i$

where f_j is the output value from the linear layer and f_{y_i} is the output value for the true score label

The cross-entropy loss function treats scores discretely and labeling a photo with any incorrect score has the same impact on the loss. For example, if the photo's true score is 4, the cross-entropy loss is the same if the model predicts score 1 or score 5 for that photo. To be able to use the cross-entropy loss function with the pre-trained ResNet model, a fully connected linear layer is trained with the correct number of output classes. The model output generated by this linear layer and the actual score labels are used to calculate the loss and inform further training of the model.

The second model uses the L2 loss function (also know as the euclidean loss or mean square error loss).

$L = \sqrt{\sum_i (\hat{y_i} - y_i)^2}$

where $\hat{y_i}$ is the predicted label from the model and y_i is the true score label

The L2 loss function takes into account the sequential order of the scores and its value is impacted by how far off the predicted scores label are versus the actual scores. Returning to the prior example, the L2 loss would penalize mislabeling the score 4 photo as score 1 more than mislabeling it as score 5. Using the L2 loss function required creating output scores from the model that represented a value within a bounded range. This was accomplished by training a fully connected linear layer with 1 output class and applying a sigmoid activation function to that single score. Because the values from the sigmoid function range between 0 and 1 and the social media scores range between 1 and 5, the conversion between the output values and scores was calculated using the following functions:

$adj.\,output = \lfloor 5 \cdot output \rfloor + 1$

$adj.\,score = \frac{score-1}{5} + .1$

While training the second model, the output values and adjusted actual scores were used to calculate the loss. To be more comparable to the first model, the second model's predictions were based on the adjusted output values.

4. Dataset

A new dataset needed to be created to address this problem. Based on ease to both collect and rate the photos, 424 photos of my face were used. These photographs were taken over the last 5 years in a variety of settings (e.g. on the beach, at formal events).

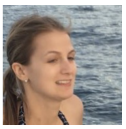
Each photos was assigned a 1 to 5 social media worthiness score, indicating to what extent I believed each photo represented my best self. High scores were assigned to photos I thought were more flattering and low scores to photos I thought were less flattering. Figure 1 explains the scoring criteria in more detail.

Across the images in the dataset, the score breakdown was 14% for score 1, 21% for score 2, 33% for score 3, 24% for score 4, and 8% for score 5.

A 5 point social media worthiness scale was chosen because it captured the right level of granularity. It would be difficult for humans to provide more info on how much they like photos because, from my experience, their opinions are generally not that nuanced. Therefore, the difference picked up with a more fine tune scale would likely be meaningless. Also, a 5 point scale captures the actions a human may take in their own process to decide what photos to share on social media. Like the 5 point scale, these actions range from will not share to will share with several different levels of uncertainty in between.

I personally scored each photo because it was not possible to use existing data (e.g. Facebook or Instagram likes) to infer these scores. For my own photo collections as well as for many others' photos, only a limited set of photos,

Score 1: Strongly dislike the photo. Has embarrassing features (e.g. eyes closed, caught completely off guard). Will absolutely not share the photo



Score 2: Dislike the photo. Has awkward features (e.g. looking away from the camera, bad hair day). Will not share the photo

Score 3: Don't strongly like or dislike the photo. Hesitant to share the photo with anyone

Score 4: Like the photo. Willing to share the photo with a few friends or in a disappearing message (e.g. Snapchat)

Score 5: Believe it is a top notch photo. Willing to broadly share the photo by posting it to a social network (e.g. Instagram, Facebook)

Figure 1. Score Criteria and Example Images

corresponding roughly to scores 4 and 5, are posted to social media. This implies that the existing data would not include photos with lower scores that are not posted. As a result, there would be no way to distinguish between these photos when trying to infer scores.

Data processing focused on cropping each photograph to only include my face. This removed extraneous parts of the image (e.g. other people, my clothing, background) and helped ensure that the model's classification was based on the features that had primarily determined the photo's score. Otherwise, it would be more difficult to determine whether CNN models could be used for this purpose. Cropping took advantage of thumbnail photos from Photos, Apple's desktop app that stores and organizes personal photos. These thumbnail images only include a square around the particular person's face in that photo.

Other data processing included resizing and normalizing. To work with both the minimum size requirements of the pre-trained ResNet model and the maximum size allowed by GPU computing constraints, the photos were reduced slightly to be 224 x 224 pixels (from 296 x 296 pixels). The images were normalized to have a mean of 0 and a standard deviation of 1.

For all of the models, the dataset was split roughly 80% training data and 20% validation data, resulting in 338 training images and 86 validation images.

5. Results and Discussion

5.1. Training and Evaluation

Stochastic gradient descent with exponential learning rate decay was used to train the models. Another consequence of the data set's small size was that it didn't seem feasible to do robust cross validation. As a result, all of the metrics discussed in this section are based on all of the validation data. To minimize the negative impact of not using cross validation, hyperparameter training was limited to several moderate changes to the learning and momentum rates and model comparison was based on one version of each model. (When the larger data set discussed in the conclusion is gathered, this analysis would be repeated with cross validation to confirm that this experimentation procedure decision has not meaningfully impacted the results.)

The models were evaluated primarily based on their classification accuracy. Afterwards, saliency maps (a visualization of which pixel values were most important to determining the predicted classification label, see Simonyan et al and Zeiler and Fergus for more details) were used to confirm that facial features had the largest impact on predicting the score label [6, 8]. Hypothetically, if the model had high classification accuracy but the saliency maps showed that labels were predicted based on other features (e.g. background, body position), the results would likely be arbitrary, not generalize to larger data sets, and not make a strong case for the feasibility to use CNNs to solve this type of problem.

5.2. Overall Results

Especially given the data set's small size, the initial results are promising, especially from the cross-entropy model. The cross-entropy model had over 50% classification accuracy and less than 0.6 mean absolute error on the

Actual Score	Classification Accuracy	Mean Absolute Error
1	67%	0.3
2	42%	0.7
3	63%	0.5
4	42%	0.8
5	0%	1.5
Overall	51%	0.6

Table 1. Cross-entropy Loss: Model Performance

Actual Score	Classification Accuracy	Mean Absolute Error
1	0%	1.8
2	16%	0.8
3	80%	0.2
4	21%	0.8
5	0%	2.3
Overall	41%	0.7

Table 2. L2 Loss: Model Performance

validation data. Classification misses were more often than not off by 1 (e.g. predicting score 2 or 4 when the actual score is 3). If these are included as accurate predictions, the accuracy rate increases to 87% (for both models).

Accuracy was highest for below average and average score values (scores 1 - 3). Photos with score 3 had the highest accuracy at 80% in the L2 loss model and 68% in the cross-entropy model. These higher accuracy rates may be caused by unbalanced data since there were significantly more score 3 photos. Nonetheless, there may be facial features in lower score photos that make them easier for the models to classify (e.g. closed eyes) or these images may be distinguishable based on their own poor quality (e.g. blurry, dark).

Supporting the reasonable accuracy rates, the saliency maps suggest that facial features have a significant impact on classification results. This is seen in the maps because the red pixels, which highlight important regions of the photo for determining classification results, are centered around the face. There is also a stark outline around the face due to red regions within the face and black regions outside. Although by no means sufficient to conclude that facial features definitively determine the classification scores, it is at least promising for the feasibility of CNN models to solve this problem to see they have a large impact. Hopefully, identifying these same facial features would be repeatable in different photos of the same subject or potentially in photos of other people and lead to similarly positive results.

5.3. Loss Comparison

Based both on classification accuracy and mean average error, the model with cross-entropy loss outperforms the model with L2 loss. Even on mean square error, the cross-entropy loss model has slightly better results (0.110 for cross-entropy model vs. 0.107 for L2 loss model).

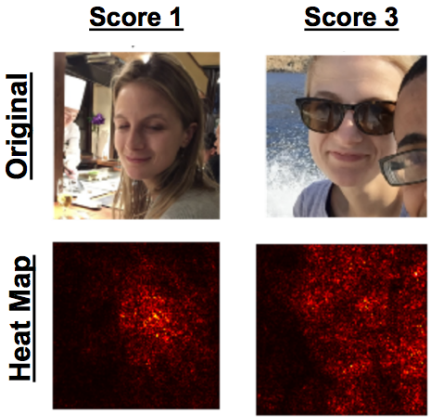


Figure 2. Cross-entropy Loss: Saliency Maps

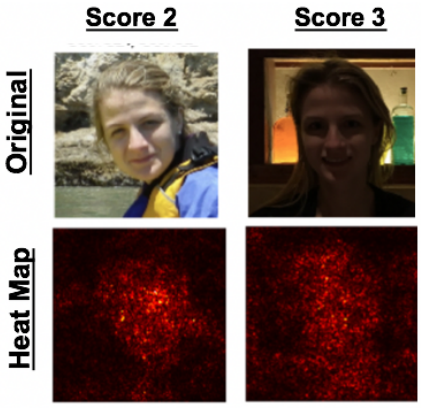


Figure 3. L2 Loss: Saliency Maps

(brighter red indicates more important for classification results)

As seen in the confusion matrices, the two models are prone to different types of errors. The L2 loss model has higher accuracy rates for average score photos (scores 2-4) and does not correctly classify any score 1 or score 5 photos. This is likely because the L2 loss function penalizes large differences between the predicted and actual scores. These large differences are more likely to occur when predicting extreme scores (score 1 or 5) and thus the model avoids predicting extreme scores entirely. In contrast, the cross-entropy model has higher accuracy rates for score 1 or score 5 photos and lower accuracy rates for average score photos. This makes sense given that the cross-entropy loss function does not penalize predicting extreme scores incorrectly more than predicting other scores incorrectly.

Number of Images							
	Predicted Scores						
		1	2	3	4	5	Total
Actual Scores	1	6	3	0	0	0	9
	2	2	8	6	3	0	19
	3	3	4	22	6	0	35
	4	2	1	6	8	2	19
	5	0	0	2	2	0	4
	Total	13	16	36	19	2	86

Figure 4. Cross-entropy Loss: Confusion Matrix

Number of Images							
	Predicted Score						
		1	2	3	4	5	Total
Actual Score	1	0	2	7	0	0	9
	2	0	3	16	0	0	19
	3	0	2	28	5	0	35
	4	0	0	15	4	0	19
	5	0	1	3	0	0	4
	Total	0	8	69	9	0	86

Figure 5. L2 Loss: Confusion Matrix

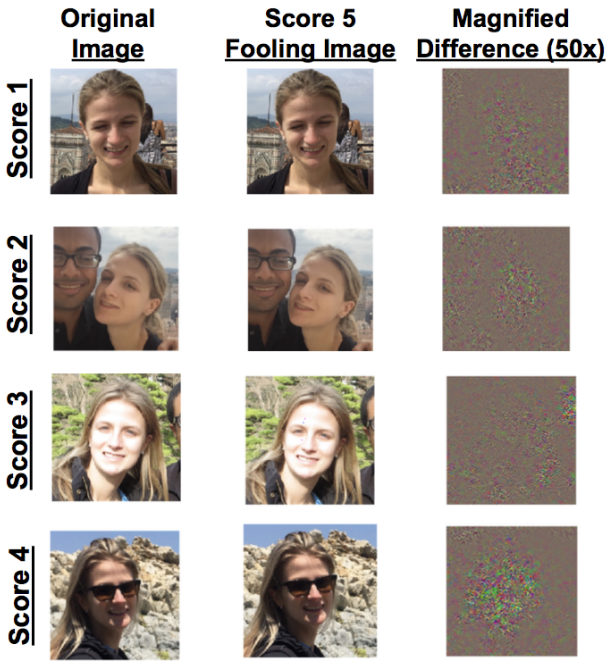


Figure 6. Cross-entropy Loss: Fooling Images

Based on my experience, people are most sensitive to have photos with extreme scores inappropriately classified. They would be particularly upset if a photo that portrayed their worst self was mislabeled and as a result shared on social media. This suggests that it is worthwhile to use the cross-entropy model and trade higher accuracy on extreme scores for lower accuracy on average scores.

However, these results are not definitive. Other variations of L2 loss that take advantage of the sequential score order may still be able to provide better classification results (including increasing classification accuracy for score 1 or score 5 photos).

5.4. Adversarial Training

Adversarial training with fooling images was used to gain further insight into the CNN models and what features drove the classification results. Fooling images result from slightly altering an image's pixel values until the model is "tricked into changing the image's score label. (Please see Goodfellow et al and Nguyen et al for more details on fooling images [4, 1].) In particular, this project created new fooling images from photos that originally had score 1 - 4 labels and these new images were changed enough to receive score 5 labels from the cross-entropy model. This image transformation is equivalent to turning a mediocre photo into one of the subject's favorite photos.

Unfortunately, the models did not complete that transformation successfully. The fooling images were nearly identical to the original images and the changes were largely invisible to human eyes. This highlights that despite reasonable accuracy rates, the CNNs models have not captured the human process to determine whether to share these photos.

However, when the changes are viewed with 50x magnification, they seem to be clustered near the face. Therefore, very small changes to the face pixels in the fooling images were able to change the model's predicted label. This reinforces the saliency map result that facial features have a large impact on classification results.

5.5. Generalizing the Results

These models and their classification results are severely harmed by using a data set that only contains my photos. Even with promising saliency maps that emphasize facial features, it is difficult to argue that these results are generalizable and not idiosyncratic to the data set. To be more specific, it is quite possible that the model has been trained to identify my unique preferences on what photos represent my best self (e.g. an open mouth smile is better than a closed mouth smile) instead of universally held preferences (e.g. avoid closed eyes). The model could also be picking up idiosyncrasies about my face or the particular photos that are not generalizable.

With the current data set, it is nearly impossible to determine the extent of this idiosyncratic results problem. Merely to start off, a separate cross-entropy model was

Actual Score	Classification Accuracy
Closed	47%
Open	99%
Overall	90%

Table 3. Closed vs. Open Eyes: Model Performance

Number of Images

		Predicted		
		Closed	Open	Total
Actual	Closed	7	8	15
	Open	1	70	71
	Total	**8**	**78**	**86**

Figure 7. Open vs. Closed Eyes: Confusion Matrix

trained to classify whether photos in the data set had open vs. closed eyes. With 90% overall accuracy and 47% accuracy on closed eye photos, the model was reasonably successful. In other contexts, CNNs have been shown to identify specific facial features. Matsuga et al use CNNs to identify smiles with 98% accuracy [7].

This suggests that CNN models can potentially identify features that are generally disliked. Furthermore, identifying closed eyes may have helped increase the classification accuracy for low score photos. If this is indeed the case, the model may be more generalizable than was originally feared.

The focus for additional work on this project would be to address this particular issue and determine whether the initial results hold when models are trained on broader data sets.

6. Conclusion

The results so far have proven out that CNN models have the potential to identify our best selves and classify social media worthy images. Along with moderately high accuracy rates, the results are promising because facial features have been shown in the saliency maps to be a likely determinant of the models' predicted social media worthiness scores.

Regardless of the positive results, a significant amount of work is still outstanding in order to more conclusively show that CNN models can feasibly address this type of problem. The next steps for this project focus on creating a new data set and identifying new features.

Taking advantage of additional time, manpower, or other resources, the immediate next step would be to develop a larger and broader data set. As mentioned above, these results are limited by the current dataset, which only included less than 500 photos and 1 person. An ideal dataset would include thousands photos of many different people. These images would be from uncurated photo collections which contain photos representing all 5 scores. Each image would be assigned a social media worthiness score by the person in the photo. It is also important that this dataset cover a wide range of races and ethnicities for photo subjects. These differences in appearance may impact what makes a person believe a photo represents her best self, which features are relevant for classification, and the ultimate classification results.

After collecting more data, there are many opportunities to explore improved or additional image features. For example, the ResNet model's ability to extract relevant human features may be enhanced by pre-training it only on human photos (instead of the entire ImageNet dataset). Or, face masks or other techniques from facial recognition could be used to generate features based on only the face portion of the image or specific facial features (e.g. eyes, mouth).

Despite the choice to crop the photos in this data set to only include the face, there also may be relevant features from other portions of photos. This reflects that people most likely post the entire photo (instead of a tight cropped version around her face) to social media and therefore evaluate all aspects of the photo when deciding to share it. Other people in the photo, how they are positioned, and the social media worthiness score for their faces could also create helpful features.

It may be many years until an iPhone decides for itself what photos to share with our friends. However, based on these initial results, the powerful combination of CNN models and more training data may make this happen much sooner than we all think.

References

[1] J. C. Anh Nguyen, Jason Yosinski. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5

[2] T. H. Gil Levi. Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, 2015. 1

[3] X. W. G. Y. Gray D., Yu K. Predicting facial beauty without landmarks. *Computer Vision ECCV 2010*, 2010. 1

[4] C. S. Ian J. Goodfellow, Jonathon Shlens. Explaining and harnessing adversarial examples. *ICLR*. 5

[5] S. R. J. S. Kaiming He, Xiangyu Zhang. Deep residual learning for image recognition. 2015. 2

[6] A. Z. Karen Simonyan, Andrea Vedaldi. Deep inside convolutional networks: Visualising image classification models and saliency maps. 2013. 3

[7] Y. M. Y. K. Masakazu Matsugu, Katsuhiko Mori. Subject independent facial expression recognition with robust face

detection using a convolutional neural network. *Advances in Neural Networks Research: IJCNN '03*, 2003. 6

[8] R. F. Matthew D Zeiler. Visualizing and understanding convolutional networks. 2013. 3

[9] A. C. T. A. B. S. Lawrence, C.L. Giles. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 1997. 1

[10] F. W. G. D. X. Cao, D. Wipf and J. Sun. A practical transfer learning algorithm for face verification. *ICCV*, 2013. 1

[11] M. R. L. W. Yaniv Taigman, Ming Yang. Deepface: Closing the gap to human-level performance in face verification, 2014. 1

7. Code References

- CS231N Assignment 2: http://cs231n.github.io/assignments2017/assignment3/
- PyTorch's transfer learning tutorial: