CS230

An Iterative Unfreezing Strategy for Predicting Human Emotional Response to Images

Ashwin Sreenivas
Department of Computer Science
Stanford University
ashwinsr@stanford.edu

Jessica Zhao
Department of Computer Science
Stanford University
jesszhao@stanford.edu

Abstract

Research in emotion recognition has largely focused on identifying emotions from the human face. There is less literature on identifying human emotional response to images. For example, an image of a sunset may inspire awe, while an image of a graveyard may inspire fear. We are interested in predicting these emotional responses to images. Given an input image, we are interested in predicting 1 of 8 emotion classes: Amusement, Anger, Awe, Contentment, Disgust, Excitement, Fear, or Sadness. We apply transfer learning by initializing our model's weights with VGG19 pre-trained on ImageNet and introduce an iterative unfreezing strategy to fine-tune our model. We achieve 60.1% accuracy, surpassing the 58.3% accuracy achieved by the authors for the original dataset.

Github Repository: github.com/jessicae5/cs-230-final-project

1 Introduction

Our work aims to predict human emotional response to general images that may or may not contain faces. Given an input image, we use a neural network to output 1 of 8 emotion classes: Amusement, Anger, Awe, Contentment, Disgust, Excitement, Fear, or Sadness.

We envision many applications across many industries. For the film industry, we can classify movies with their emotional trajectories: for example, 70% sad, 20% happy, 10% awe. Streaming services such as Netflix and Amazon Prime can tag their content by emotional class and offer suggestions based on mood. For the consumer industry, we can help people better organize and enjoy their photos. Many photo applications (e.g. Apple Photos, Google Photos) already have quantitative tools such as facial tagging and object recognition, but we can make albums smarter by adding qualitative information that curates moments of amusement, excitement, and awe.

This has historically been a technically challenging problem because of the sheer number of factors that can influence an emotional response. However, the introduction of deep learning frameworks combined with a proliferation of data and compute power can help make substantial progress to solving this problem.

2 Related work

Prior work in emotion recognition can be separated into two categories: classification based on manual feature selection and classification based on learned features via neural networks.

CS230: Deep Learning, Fall 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2.1 Manual Feature Selection

The space of images increases exponentially when expanding the scope of image emotion recognition beyond facial expressions. Different techniques for emotion recognition arise when considering images with faces, images with occluded faces, and images without faces at all. In their work on emotion detection, Miyakoshi and Kato consider a subset of this image space by building a model for facial emotion recognition that accounts for occlusion. As features, they detail specific facial features such as "blinking of right eye," "vertical movement of left brow," and "horizontal stretch of right brow left corner" and use a Bayesian network to infer gaps in knowledge when faces are occluded. They achieve above 50% accuracy on each category. However, though their work can be effective for faces both visible and occluded, we hope to apply emotion recognition to the larger search space of all images. Additionally, the authors used a 6-category classification (Happiness, Anger, Sadness, Surprise, Disgust, Fear) while we hope to use an 8-category classification.

In their work on image emotion recognition, Zhao, Gao, et al. propose an artistically-inclined strategy for manual feature selection. Instead of the standard low-level features of color, value (lightness or darkness), line, texture, shape and space, they use 6 high-level aesthetic features: symmetry, emphasis, movement, harmony, variety, and gradation. The authors posit that high-level features are closer to a human's emotional response and traditional low-level features cannot map to emotions as directly. For example, symmetric images tend to evoke positive emotions while images with strong color contrast are more likely to evoke negative emotions, while there is a less clear mapping of low-level features such as shape to emotion. They employ K-fold cross validation and PCA for dimensionality reduction on their feature vectors to output 1 of 8 emotion classes (the same as ours), yielding accuracies ranging from 56-69% by category on datasets composed of nineteenth-century European and abstract art. While their approach is successful for the art dataset, there may be limitations in imposing aesthetic features. Their model may be biased towards aesthetically-inclined images and may not extend as well to general images. We hope to focus on building a generalized model.

2.2 Deep Learning Strategies

Research by You et al. focuses on building a large scale dataset for image emotion recognition (which we use in our work) and establishes a baseline for the problem space. The authors build a neural network that applies transfer learning by initializing their model's weights from the original ImageNet-trained model with 5 layers and achieve a baseline of 32.1%. They also build a fine-tuned model using techniques from previous authors, including recognizing image style (Karayev et al. 2013) and semantic segmentation techniques (Long, Shelhamer, and Darrell 2014), which achieves a state-of-the-art accuracy of 58.3%. In our work, we extend their research by exploring different models pre-trained on ImageNet and applying a novel iterative unfreezing strategy.

A paper by Xu, Cetintas et al. explores visual sentiment prediction with Deep CNNs. They apply transfer learning with a CNN architecture described by Krizhevsky, Sutskever, and Hinton with 7 layers and a softmax layer (5 convolutional and 2 fully connected layers with ReLU activations), pre-trained on a subset of 1.2 million labelled ImageNet data. They achieve a 64.9% accuracy on a Twitter image dataset. However, instead of categorical classification, they use a 5-level sentiment polarity range from [-2,2] indicating strongly negative to strongly positive emotions. However, we hope to use an 8-category classification rather than numerical polarity.

3 Dataset

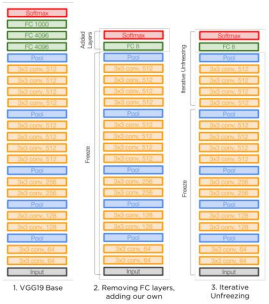
3.1 Methodology

We use a dataset curated by You et al. through Amazon Mechanical Turk. The researchers presented an image with an emotion and Mechanical Turkers were asked if they agreed or disagreed with the pairing. Each image was presented to 5 unique participants and therefore has 5 votes. We take the majority label as ground truth (e.g. if 3 agreed and 2 disagreed on an image presented as "Awe", we label the image "Awe").

(a) Sample images labelled by emotion

Emotion	Train Count	Dev Count	Test Count
Amusement	4244	251	256
Anger	1088	46	57
Awe	2657	146	143
Contentment	4645	245	269
Disgust	1398	78	94
Excitement	2474	142	119
Fear	874	46	32
Sadness	2393	144	129
Total	19773	1098	1099

(b) Dataset breakdown



(c) Model Architecture

Figure 1: Dataset and Model

3.2 Preprocessing

The dataset contains links to images on Flickr that we separately scraped. As the original dataset was curated in May 2016, some of the image links have expired. Moreover, we only use images with an affirmative classification. We also discovered duplicate image classifications, which accounted for less than 5% of the dataset. To address this, we take the most salient label (least divisive based on number of agrees and disagrees). Thus out of the 89,319 original image links, we use 21,970. The retrieved images are a mixture of color and black and white. We randomly split our resulting dataset into 90% train, 5% dev, and 5% test following the breakdown in Figure 1b.

3.3 Data Augmentation

We experimented with different forms of data augmentation. We found that shearing with any range was ineffective. However, we did find that horizontal mirroring and rotations with random degree values from [0, 10] in either direction were both effective. Finally, we resize all images to the input dimensions (224, 224, 3) and apply the standard VGG19 preprocessing using the Keras library.

4 Methods

4.1 Transfer Learning Initialization

We use Keras on a Tensorflow backend through the AWS Deep Learning AMI platform. Given the relatively small size of our dataset and the challenges in interpreting images, we decided to apply transfer learning from models pre-trained on ImageNet. You et al., the authors who curated the dataset, also applied transfer learning using the original pre-trained ImageNet model with 5 layers. We experimented with a few models including Inception, DenseNet, ResNet50, VGG16, and VGG19 before deciding on the best-performing VGG19 to initialize our weights.

4.2 Training FC Layers

We start by removing the top 4 layers (the 3 FC layers and the softmax output) of VGG19. In our architecture search, we experimented with adding different numbers of FC layers of varying sizes and observe that 1 FC layer of size 8 yielded the best accuracy [Figure 2a]. We apply a softmax activation on this final layer, where the number of classes $C = 8$:

$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}}$$

We freeze all layers excluding this and use the Adam Optimizer with α =1e-4 and Categorical Cross Entropy Loss, the general logistic loss for multi-class classification, to train, where t_i is the ground truth:

$$CE = \sum_i t_i log(f(s)_i)$$

We use Adam as it combines the advantages of Momentum, which dampens out oscillations and is helpful for sharp-peaked landscapes, and RMSProp, which divides learning rate by the average of squared gradients and helps adapt α to the parameters.

4.3 Iterative Unfreezing

Next, we unfreeze the top 4 convolutional layers one at a time and train them with our single FC layer over 4 iterations. Thus, the first training iteration unfreezes the FC layer and 1 convolutional layer, the second training iteration unfreezes the FC layer and 2 convolutional layers, and so on until all 4 convolutional layers are unfrozen. Throughout the iterative unfreezing, we gradually decrease the learning rate. The first iteration of unfreezing uses Adam with α =1e-5; subsequent iterations use Adam with α =1e-6. After we finish the iterative unfreezing process across the 4 convolutional layers, we unfreeze the entire model and train overall using Adam using α =1e-6.

4.4 Rationale

We find that this iterative strategy works more effectively than training all layers simultaneously. Because the weights of our added FC layer are randomly initialized, we discover that collectively unfreezing layers beyond this layer during training causes the large gradients caused by the randomized weights to propagate through the model, wrecking the finely tuned weights. Thus, we first focus on training the FC layer in isolation. Then, by applying iterative unfreezing across the 4 convolutional layers, we individually fine-tune each layer and avoid potential large gradients. Throughout our architecture search for the unfreezing process, we also find that decreasing learning rate for later convolutional layers is an optimal fine-tuning mechanism. To prevent against overfitting, we experimented with the number of layers to unfreeze. We decided to stop unfreezing following 4 convolutional layers when we noticed overfitting to the training set after adding additional layers. We determine mini-batch size of 32 and epoch size of 10 based on best performance.

5 Results and Discussion

The iterative unfreezing model achieves an accuracy of 60.1% on our test set, which surpasses the 58.3% accuracy in the original paper by You et al. We perform manual error analysis and examine the misclassified images by class to tabulate trends. Globally, we notice there are challenges in extracting connotation from images and that some images can have multiple correct classifications.

5.1 Anger misclassified as Sadness

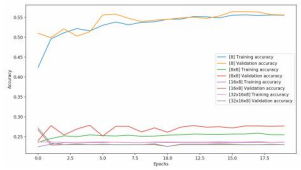
Out of the 16 misclassified, 11 have a non-unanimous vote and 7 have a strongly-divided vote. 11 are black and white, meaning our network may be associating monochromatism with sadness. Moreover, 9 could have been sadness based on human evaluation. Sadness is challenging to identify as it arrives in many forms not immediately apparent in a facial expression or. landscape A frown can evoke sadness; a pile of rubbish labelled “drugs” can also evoke sadness for the state of the world.

5.2 Sadness misclassified as Anger

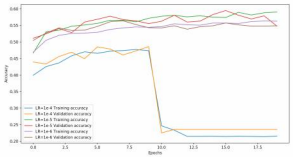
The 3 misclassified images are all close-ups of faces that contain red, warm temperature tones, meaning our network could be associating red colors with anger.

5.3 Contentment misclassified as Sadness

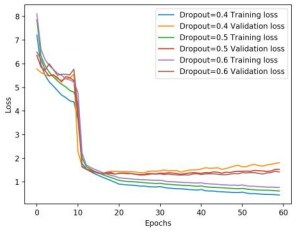
Out of the 29 misclassified, 20 have a non-unanimous vote and 11 have a strongly-divided vote. We notice that 8 images are women looking away from the camera, meaning the network could be associating indirect facial gazes with sadness. 4 are faces of animals or statues, suggesting the network is not as skilled as reading non-human faces.



(a) Searching for FC Layers



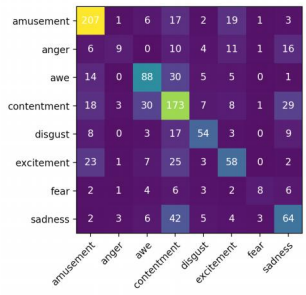
(b) Searching for Learning Rate



(c) Searching for Dropout

Model	Train Accuracy	Dev Accuracy	Parameters		
			FC Layers	Learning Rate	Dropout
Search for FC layers	**55.6%**	**55.6%**	**8**	**1e-5**	**0.5**
	25.4%	27.7%	8x8	1e-5	0.5
	23.6%	23.0%	16x8	1e-5	0.5
	23.6%	23.0%	32x16x8	1e-5	0.5
Search for Learning Rate	21.6%	23.5%	8	1e-4	0.5
	59.0%	54.8%	8	1e-5	0.5
	56.3%	**54.9%**	**8**	**1e-6**	**0.5**
Search for Dropout	84.1%	57.9%	8	variable*	0.4
	77.5%	**59.8%**	**8**	**variable***	**0.5**
	72.1%	59.1%	8	variable*	0.6

(d) Architecture Comparison



(e) Emotions Confusion Matrix

Emotion	Prec.	Recall	F1
Amusement	73.9%	80.9%	77.2%
Anger	50.0%	15.8%	24.0%
Awe	61.1%	61.5%	61.3%
Contentment	54.1%	64.3%	58.7%
Disgust	65.1%	57.4%	61.0%
Excitement	52.7%	48.7%	50.7%
Fear	57.1%	25.0%	34.8%
Sadness	49.2%	49.6%	49.4%
Overall	**60.1% test accuracy**		

(f) Emotion Statistics

Figure 2: Hyperparameter Search and Results

(a) Anger as Sadness

(b) Sadness as Anger

(c) Contentment as Sadness

(d) Sadness as Contentment

Figure 3: Sample Misclassified Images

5.4 Sadness misclassified as Contentment

Out of the 42 misclassified, 35 have a non-unanimous vote and 23 have a strongly-divided vote. 8 have nature backgrounds but feature a sad person, meaning the network could be associating outdoor landscapes with contentment but overlooking the person. Moreover, we notice 5 graveyard photos that the network misclassifies as contentment, suggesting that it has not learned what a gravestone means to humans. Yet the graveyard photos could reasonably evoke dual emotions of sadness and contentness: sadness of death, contentness of finding peace. We also notice 6 animal photos (horse, dog, and cat faces) that are misclassified, again suggesting the network's weakness in reading non-human faces. We also notice 5 images with text that clearly denote sadness, such as a story of a cat who lost a fight to cancer, suggesting our network is not learning from the text in images or there are too few images with text from which to learn.

6 Future Work

Future work can expand upon our single-category classification by taking into account the broad spectrum of human emotional response. As humans rarely respond with a sole emotion, we can allow for multiple types of emotional responses to images. We can extend our model to accommodate multi-hot encodings so images can be labelled, for example, both amusing and exciting. Additionally, our error analysis surfaced a weakness for understanding text in images. To improve upon this, we can also train on more images with text to help the network better interpret language.

7 Contributions

Ashwin worked on setting up Keras/Tensorflow on AWS, initializing the baseline model (VGG19), hyperparameter search, and generating graphs. Jessica worked on setting up AWS/EC2, scraping, cleaning, and preprocessing the dataset for use with ImageDataGenerator, literature reviews, and quantitative / qualitative error analysis.

References

[1] François Chollet et al. Keras. `https://keras.io`, 2015.

[2] JLaESaT Darrell. Fully convolutional networks for semantic segmentation'. *UC Berkeley*, 2014.

[3] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[4] Sergey Karayev, Matthew Trentacoste, Helen Han, Aseem Agarwala, Trevor Darrell, Aaron Hertzmann, and Holger Winnemoeller. Recognizing image style. *arXiv preprint arXiv:1311.3715*, 2013.

[5] Yoshihiro Miyakoshi and Shohei Kato. Facial emotion detection considering partial occlusion of face using bayesian network. In *Computers & Informatics (ISCI), 2011 IEEE Symposium on*, pages 96–101. IEEE, 2011.

[6] Hong-Wei Ng, Viet Dung Nguyen, Vassilios Vonikakis, and Stefan Winkler. Deep learning for emotion recognition on small datasets using transfer learning. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pages 443–449. ACM, 2015.

[7] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. Building a large scale dataset for image emotion recognition: The fine print and the benchmark. In *AAAI*, pages 308–314, 2016.

[8] Sicheng Zhao, Yue Gao, Xiaolei Jiang, Hongxun Yao, Tat-Seng Chua, and Xiaoshuai Sun. Exploring principles-of-art features for image emotion recognition. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 47–56. ACM, 2014.