Emotion Detection with Convolutional Neural Network

Arnie Seong[1], **Jaylen Lee**[1], **and Simon Hua**[1]

[1]Statistics Graduate Student

ABSTRACT

In this report we build an convolutional neural network that labels pictures of human faces with the displayed emotion. Using data augmentation techniques to bolster our set of training images, we obtain a classifier that is able to obtain 80% accuracy on a testing image set.

Introduction

In this project, we attempted to build a classifier that could differentiate between eight facial expressions - sadness, happiness, surprise, anger, fear, contempt, disgust, and neutral - given a dataset of approximately 13,700 labelled images of faces. Convolutional neural networks that are able to classify emotions with quite high accuracy already exist; our goal was not to invent something new but instead to explore this area of machine learning which none of the members in this group had much exposure to. Thus our primary objectives in this project were to gain intuition on working with convolutional neural networks and learn to use TensorFlow.

The ability to interpret facial expressions seems like a very complex, particularly human behavior, something we would expect to be very difficult for a computer. Thus we used a complex architecture, modeled after an existing emotion classifier, as a starting point for our own convolutional neural network, and used the validation set accuracy to guide our modifications to the architecture as we continued training and testing different network models.

Data

The source of the data was an open repository (available at *github.com/muxspace/facial_expression*) of labelled images of faces and a legend (as a .csv file), created by Brain Lee Yung Rowe and his graduate students for the purpose of training machine learning algorithms. Code to load the dataset and split it into training/validation/test sets was provided by Yoshitomo Matsubara, a graduate student in Computer Science at UCI.

The dataset contains approximately 13,700 images of faces with each face labeled as sadness, happiness, surprise, anger, fear, contempt, disgust, and neutral. To build our model, we split our data into training, testing, and validation sets. We set aside about 10 percent of the images for the validation set. Most of the images are greyscale, with centered faces facing the camera, and have a dimension of 350 by 350, but a non-negligible portion of the data have non-centered faces, faces in profile, and/or non-standard image dimensions.

Table 1. Data broken down by emotion label

emotion	number	percentage
anger	252	1.84%
contempt	9	.066%
disgust	208	1.52%
fear	21	.15%
happiness	5696	41.61%
neutral	6868	50.17%
sadness	268	1.96%
surprise	368	2.69%

Data Exploration

Our labeled data was quite imbalanced with respect to the emotions represented. Of the images in our dataset, 12,594 images, or 92.00%, were labeled either as "happiness" or "neutral"; in contrast, "fear" had only 21 images (0.15%), and a mere 9 images

(0.066%) were of "contempt." This posed substantial challenges for training, as our training-validation-test split often left no images of contempt, and very few for fear, to train on.

Fig 1 displays the "mean" image for each emotion from 1 training set which was randomly sampled from the whole data using a 70-20-10 split. Difficulties arising from the imbalance of emotions are apparent: there were no "contempt" faces in this training set, and the average of the "fear" faces contains considerable fragmentation / artifacts because there were so few in the training set. The mean image for "disgust" suffers from a slightly different, but related problem: though there are 208 "disgust" images in the full dataset, many of them were non-standard image dimensions and so were stretched during image import. They were also often images of faces that were turned away from the camera, leading to similar fragmentation and image artifacts.

We might also anticipate some difficulties for the deep learning agent from the images here - for example, the differing widths and orientations of the faces may pose to be "false features" that the agent will need to learn to ignore. We might also expect markers of gender to confound classification. The average images seem to suggest that faces of males and females fall disproportionately into one group or another: the average face for "anger," for example, seems distinctly male, whereas the average face for "happiness" appears more female.

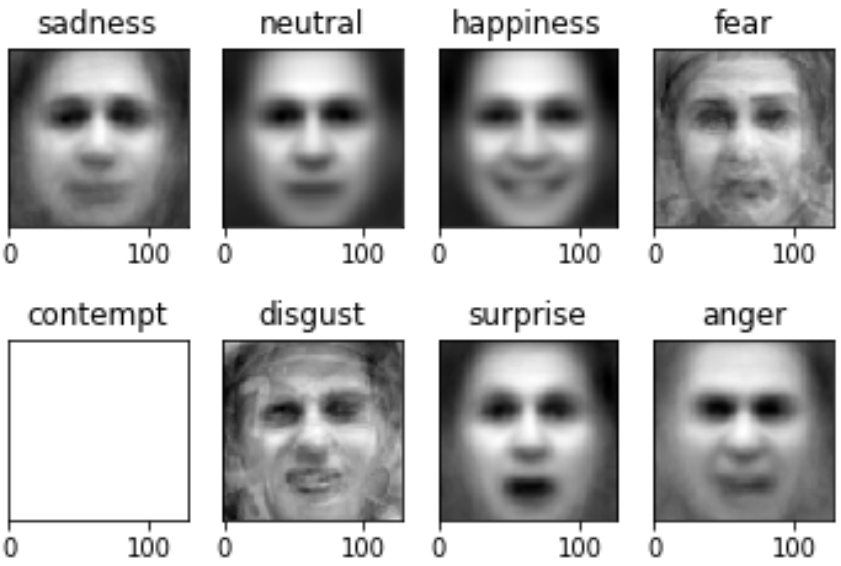

Figure 1. mean faces from 1 sampled training set, non-augmented data

Nevertheless, potential difficulties aside, it is possible to see some clear differences emerging between the average faces for each emotion. "Sadness" appears to be characterized by furrowed eyebrows, "neutral" by a somewhat stiff, expressionless face, "happiness" by a wide smile and bulging cheeks, "surprise" by a mouth agape and raised brows, and "anger" by someone yelling. And despite the visual artifacts in "fear" and "disgust," we can still make out two fairly distinct flavors of grimace.

Data Processing

Preliminary experiments on the non-augmented dataset seemed to indicate poor performance in classifying faces of contempt and fear. This is due to the relatively small percentages of faces with contempt and fear as indicated in Table 1. We implemented some image processing and manipulation techniques in order to increase the number of faces with contempt and fear in the dataset so that we can achieve better performance in these two categories.

For the 30 faces of contempt and fear in the dataset, we created an additional 11 images for each face, resulting in 360 images of fear and contempt in the augmented dataset. The 11 additional images are 11 different filters and manipulations of the original face. In particular, we implemented these techniques on the original faces: sharpen, rotate -5 degrees, rotate 5 degrees, smooth, increased smooth, transpose, greyscale, blur, Gaussian blur with a radius of 4, increased brightness, and increased contrast.

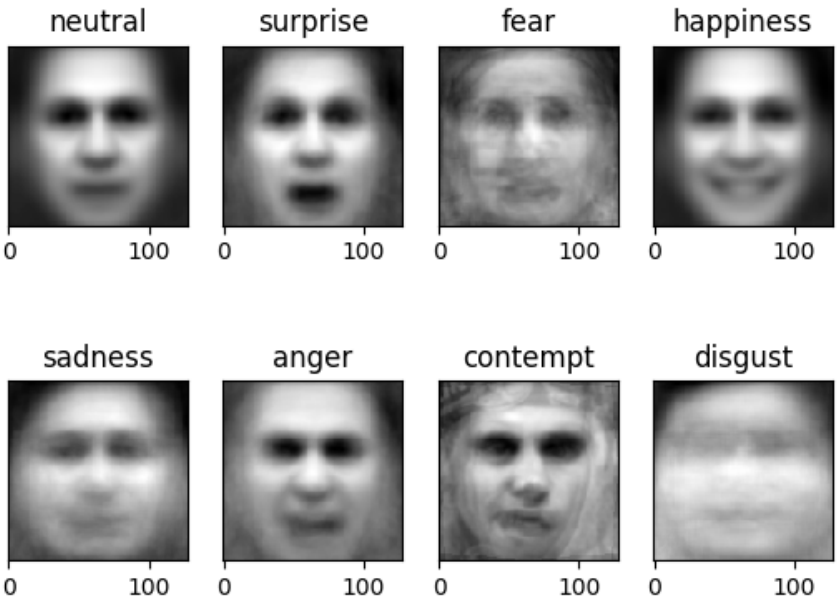

Figure 2. mean faces from 1 sampled training set from augmented data

As we can see in the figure above, the mean faces from 1 sampled training set from the augmented data now has a mean face of contempt due to our image processing and an increase in the number of images for that category. However, we do see a less definitive face for fear and disgust this time around, but this may just be due to the training, testing, and validation split; choosing another split that leads to a training set with more images of disgust and fear would probably give us a more structured mean face for those two emotions. All in all, the mean faces seem to be what we would expect to see based on the respective emotion.

Modeling

Performance Metric

We compared model performance using accuracy: defined as $\frac{\#correct}{\#total}$. This was chosen for quick model comparison when testing multiple architectures.

Convolutional Neural Network Architecture

Convolution 16 Channels. 3x3 Filter
Pooling
Flattening
FC- 128 Nodes
FC- 8 Nodes (Classification)

Table 2. Final Convolutional Neural Network Architecture

We use the architecture outlined by Simonyan and Zisserman as a starting point.[1]. This architecture is supported by 11 layers: 8 convolution layers and 3 fully connected layers with a flattening layer between them. Each fully connected layer held 4096 nodes. The convolution layers begin with a 64 channel and increasing by a factor of 2 until the final convolution layer. We

attempted to implement this exact architecture however, our dataset proved to be too small and imbalanced to train the number of parameters required for this architecture. Instead we used a stripped back version of the previous architecture that is shown in table 2.

We chose to train our model using categorical cross entropy as our loss function. We consider both kl-divergence as well as categorical cross entropy but our chosen loss function resulted in higher testing accuracy.

Results

Table 3. Final classifier results: 80.10% Test Accuracy

Emotion	N	# Predicted	True Positive	False Positive
happiness	569	615	486	129
anger	25	1	0	1
sadness	26	15	7	8
neutral	686	703	578	125
fear	25	23	23	0
surprise	36	21	14	7
disgust	20	16	8	8
contempt	10	3	3	0

Our final classifier used the architecture described in the previous section and was trained for only 3 epochs, as the plot in Figure 3 indicated that the neural network began overfitting after 3 epochs. Our test set accuracy rate was $\frac{correct}{total} = 80.10\%$, with fairly strong performance on most categories except for sadness, anger, and disgust (Table 3).

The network appears to have highest accuracy on the emotions that were most highly represented in the dataset (happiness and neutral) and the emotions for which we augmented the data (fear and contempt). The classifier had low accuracy, however, on emotions which were sparse in the data. Higher accuracy on the emotions which make up the vast majority of the dataset would be expected; the reasons for higher accuracy on the augmented emotions, which make up such a small portion of the data, is discussed below.

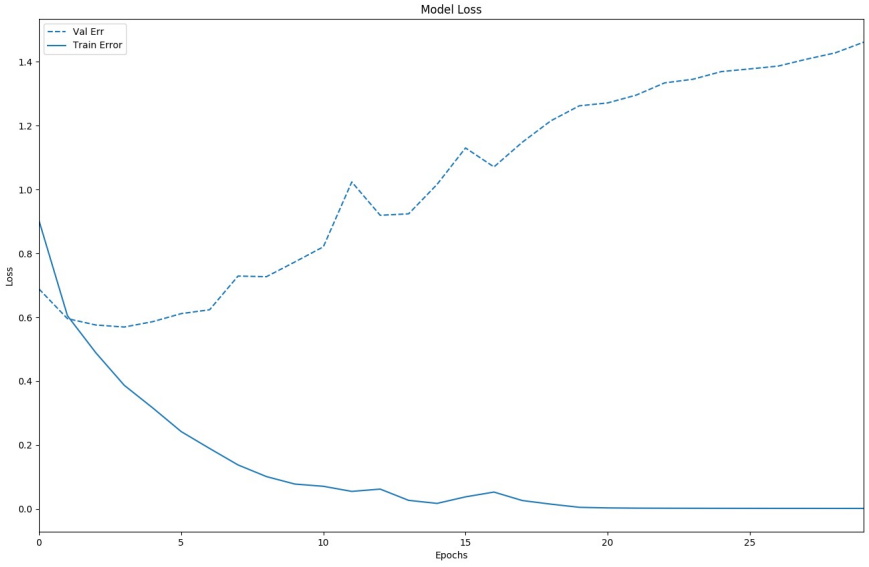

Figure 3. plot of validation and training accuracy against training epochs

Discussion

One of the findings that surprised us the most was that, among the neural network architectures we tried (and could afford to try given our resources), simpler architectures actually performed much better at this task than more complex ones. Though we initially used a significantly more complex model (2 convolution layers with 128 filters, 2 pooling layers, 1 flattening, and 3 dense layers each with 1024 nodes), it performed quite poorly (validation accuracy $< 1\%$) while also taking a very long time to train. It appears we vastly overestimated the complexity and size of the neural network necessary for this classification task, leading to extreme overfitting.

Scaling the architecture down heavily to the model displayed in Table 2 led to a classifier that, though far simpler and smaller, performed much better (accuracy $> 80\%$).

We also vastly overestimated the number of iterations/epochs required to reach a reasonable accuracy. Figure 3 clearly indicates that the neural network begins overfitting quite early in the training iterations; in subsequent model training, we found there to be negligible difference in accuracy between training for 30 epochs as opposed to just 3. It seems odd that the classifier begins overfitting so quickly, and we are unsure what to make of this result; it does not seem like it can be wholly attributable to the fact that we augmented the data, as these images represent such a small portion of the data as a whole.

Last, we attempted to make the most of the data that we were given, but supplementing the dataset with more (externally-sourced) examples of fear and contempt, not to mention of anger, sadness, surprise, and disgust, would surely improve the classifier's accuracy. Gathering additional images would also stave off a major doubt about the use of data augmentation: while it is apparently a common technique, we feel *rather* uneasy about using transformations of images in both the validation and training set as it likely leads to the classifier being both trained and validated on what is essentially the same image. This is likely why the "test" accuracy is so high on the disgust and fear images even though the number of training examples is so small: the training/validation/test split was done on the augmented dataset (as opposed to only augmenting the training data). In a future attempt at this type of classifier, we would like to extend the dataset in order to have a more equal representation of images in the training set.

A different approach to this imbalance might be to implement some sort of weighted loss function. By up-weighting the loss of "rarer" emotions, such as contempt and disgust in our dataset, we could encourage our classifier to classify more images with those labels correctly, even at the slight expense of accuracy for other labels. Implementation of this type of loss function affected the convergence of our weights and this idea was discarded in favor of data augmentation. It is of interest for future work to investigate these weighted loss functions and their ability to improve model accuracy.

References

1. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *ICLR* DOI: arXiv: 1409.1556 (2015).