Deep Learning Model for Facial Expression Recognition

Darrel Chang

California State Polytechnic University, Pomona

Introduction / Abstract

We presented a deep learning model trained for facial recognition to classify faces as happy, neutral, or sad. This was done utilizing the PyTorch framework for deep neural networks.

Facial expression recognition has applications in human-computer interaction and emotional analysis. Previous studies have utilized various deep learning techniques to handle this complex problem, but there are many common issues that have arisen. Issues such as overfitting and overshooting are problems especially when training models with limited data.

Methods

Many different model aspects, such as neural structure, data processing, and hyper-parameters, were taken into consideration when trying to find a model that was optimal for

Model Architecture: For image feature extraction, the neural network consists a set of convolutional layers and then a set of fully connected layers. For the convolutional layers, Conv2d layers, LeakyReLU activation, and MaxPool2d layers were used for feature extraction, where kernels/filters would be developed to identify specific features.

The fully connected layers used included the use of dropout and batch normalization for regularization. LeakyReLU activation was used to mitigate the vanishing gradient problem.

Data Preprocessing: Initially attempts to improve the model included data normalization as well as augmentation techniques. The augmentation techniques included Gaussian blur, horizontal flip, and rotation on the images to reduce the model's reliance on specific features

Hyper-parameters: The model was trained with a learning rate of 0.01 and 8 epochs, with a batch size of 32.

Model Analysis/Optimization: Since testing targets were not provided, the training data provided was split into training data and testing data in order to measure model performance on training data versus testing data. Having testing data on the model revealed overfitting issues, making adjusting hyper-parameters easier.

Results

Overshooting with high learning rates: Using a learning rate that was greater than 0.001, such as 0.01 caused the model to overshoot the minima. This led to consistent unstable convergence over multiple trials as well as a degradation of performance on the data set.

Overfitting with a high number of epochs: Training this model with 100 epochs resulted in overfitting, where the model's performance on the training data was not generalizable to testing data.

Dropout and Batch normalization: The inclusion of dropout as well as batch normalization reduced overfitting issues and allowed the model to better generalize for unseen data.

LeakyReLU activation: The decision to use the LeakyReLU activation function instead of ReLU mitigated a vanishing gradient problem found during training.

During training, the convergence point was higher than ideal in terms of testing error. One possibility is having too complex of a neural network. This would hinder the training process as well as generalization abilities. With a neural network too complex for the problem may have created a higher risk of local minima. With more local minima, the optimizer has a more difficult time converging to the global minima.

The complex model also made it more challenging to find optimal hyper-parameters for this specific problem set.

Conclusions

This research demonstrates the importance of choosing the current hyper-parameters and data processing techniques when designing a conventional neural network for image classification. We found that a simpler model with regularization techniques such as dropout and batch normalization would have done better than complex models that have too many hyper-parameters to adjust. Complex models also suffered from overfitting, demonstrating the significance of understanding the significance of different model complexity as well as different data processing techniques on performance to reach optimal results with image classification problems such as facial expression recognition.

References

Research Paper: "Dropout: A Simple Way to Prevent Neural Networks from Overfitting" by Nitish Srivastava et al.

•This seminal paper introduces the Dropout regularization technique and its effectiveness in preventing overfitting in complex neural networks.

•Link: https://www.cs.toronto.edu/~hinton/absps/JMLRdropout.pdf

Research Paper: "Deep Face Recognition" by Yaniv Taigman et al. (2014)

- This paper introduced a deep convolutional neural network specifically designed for face recognition tasks.

•Link: https://www.cs.tau.ac.il/~wolf/papers/deepface-conference.pdf