Learning Relation-Aware Facial Expression Representations with Transformers:

A Performance Comparison with CNNs

A20521362

Young Sun Lee

Abstract

Facial Emotion Recognition (FER) refers to the extraction of facial expression features present in images and classifying them into emotional classes. FER is considered a challenging classification problem not only because it involves extracting commonalities from different faces but also because of the variations in the images themselves, such as facial poses and lighting. With the advancement of deep learning technologies, the accuracy and efficiency of FER systems have significantly improved, leading to attempts to apply these systems in various applications. However, on the other hand, there are criticisms regarding the lack of scientific consensus on the definition of emotions and issues of ethics. Currently, it is expected that the technology will start to be used in a way that detects clear emotional expressions for the public good, and it is anticipated that it will become an essential feature when androids that interact with humans are developed in the future. This project measures the performance of various deep learning architectures applied to the FER-2013 dataset. The main deep learning architectures to be applied include AlexNet, VGGNet, ResNet-50, ResNet-18 with Transformer and MAD (Multi-Attention Dropping), ResNet-18 with Attention, etc. The performance will be explored and compared by applying various optimizers, learning rates, and epochs to these architectures.

1. Introduction

Facial Emotion Recognition refers to the identification of human expressions, such as happiness, sadness, surprise, etc., as they appear on the face. The initial impression of such technology is that it will play a significant role in human-computer interaction. Furthermore, it is thought that this capability will be possessed by robots that will eventually empathize with humans. This technology can also be applied to digital advertising, online gaming, customer feedback assessment, and healthcare, and is already being used to some extent today. In the United Arab Emirates (UAE), the technology is being used to detect people's facial expressions through public cameras and then gauge the overall mood of the citizens. Skyscanner is anonymously detecting and measuring various expressions of users in some countries. It is known that this has significantly increased customer satisfaction with Skyscanner. However, there are also concerns. There is no academic foundation for recognizing emotions from expressions. When someone smiles, is that smile a result of happiness? Or a result of embarrassment? Is a furrowed brow a result of inner emotions? Or an expression of irony? Or something created as a joke? Not only is there a lack of rigorous research backing the connection between emotions and expressions, but there are also ethical issues. Analyzing and revealing a person's emotional state that they may wish to hide can be a privacy issue, and even a slight misanalysis can lead to prejudice and disadvantage. In some cases, actors could use this in reverse to commit fraud. As a result, there are recent efforts to move towards ethical emotion recognition within certain regulations. For example, it could be used for support for the visually impaired or autistic individuals, medical diagnosis, and patient care.

Regardless of the right or wrong of emotion recognition, visual data such as photographs are necessary for computers to perform this task. If the visual data is taken under controlled conditions and in a consistent environment, the accuracy will be high. However, when applying emotion recognition in reality, factors such as varying conditions and changes in facial poses must also be considered. In this case, there can be high intra-class variability and low inter-class variability. For example, if there are data of 100 people expressing 7 different emotions, the different emotions of the same person might have more similar factors.

The dataset used in this project is the FER-2013 dataset. This dataset consists of 48*48 grayscale images of faces classified into seven emotions: angry, disgust, fear, happy, neutral, sad, and surprise, comprising 28,709 training images and 3,589 test images. The FER-2013 dataset is an open-source dataset originally created by Pierre-Luc Carrier and Aaron Courville for their ongoing project and was publicly shared for a Kaggle competition just before ICML-2013. Moreover, these images are taken under natural conditions and are not easy to classify. Human performance on this dataset is known to be 65.5%.

Facial Emotion Recognition (FER) has been of long-standing interest, and initial methods relied on handcrafted features and classical machine learning algorithms. However, the advent of deep learning has revolutionized this field. In particular, CNNs have shown outstanding performance in various computer vision tasks, including FER, due to their ability to learn hierarchical features from data. CNNs were first introduced in (LeCun et al., 1989) by applying filtering techniques to artificial neural networks to process images more effectively, and the form of CNNs currently used in deep learning was proposed in (LeCun et al., 1998).

Recently, the Transformer architecture, which has been successful in the field of natural language processing, is being applied to computer vision and achieving results. Transformers capture global information and interacting elements in images, demonstrating high performance in various computer vision tasks.

This project aims primarily to evaluate the performance of various deep learning architectures using the FER-2013 dataset and to understand their complexity. Initially taking AlexNet, which showed early success in deep learning, as a benchmark, we will measure how much more superior performance can be achieved with architectures like VGGNet, ResNet, a combination of ResNet and Transformer, and a combination of ResNet and Attention.

There are some constraints in this experiment. This project was executed on Google Colab using an A100 GPU. Preliminary trials showed that while ResNet-50 alone could run, the combination of ResNet-50 and Transformer could not due to memory limitations, so the experiment was conducted with a combination of ResNet-18 and Transformer. Also, attempts to implement a more complex vision Transformer were not successful due to memory limitations. Moreover, most models have been modified at the input to fit a certain size and channel. The FER-2013 images are black and white, so they have one channel, and the input size was set to 224x224 after augmentation.

2. Data Loading and Preprocessing

2.1 Dataset Overview

In this study, the FER-2013 dataset was utilized, recognized as a benchmark in the FER community. The FER-2013 dataset is an open-source collection originally created by Pierre-Luc Carrier and Aaron Courville for their ongoing project and was publicly shared in preparation for a competition on Kaggle just before ICML-2013. This dataset categorizes emotions into seven types: Angry, Disgust, Fear, Happy, Neutral, Sad, and Surprise. Images are black and white, measuring 48x48 pixels. It comprises 28,709 training images and 3,589 test images, with uneven distribution across different emotions. The dataset includes off-frontal poses and diverse lighting conditions, offering relatively natural pictures, which present more challenges in recognition. Human performance on this dataset is known to be around 65.5%.

2.2 Data Loading Mechanism

To efficiently load data, a custom FERDataset class, inheriting from PyTorch's Dataset class, was implemented. This class acts as a wrapper around the raw image files, simplifying access to images and their corresponding labels.

The init method of the FERDataset class initializes the dataset via directory structure, categorizing images based on emotion labels and creating a comprehensive list of image paths along with associated emotion indices.

In contrast, the getitem method assists in real-time loading of images during training. This method ensures each image is correctly opened, converted to black and white, and returned with its label. A retry mechanism is included to handle file access issues, ensuring the stability of data loading.

2.3 Preprocessing Pipeline

Once data is loaded, it undergoes a series of preprocessing steps to enhance quality and prepare it in the right format for model training. The preprocessing pipeline is constructed using PyTorch's transforms module, linking multiple image transformations:

- Black and White Conversion: While FER2013 images are inherently black and white, explicit conversion ensures consistency and removes potential color channels.
- Resizing: All images are uniformly resized to 230x230 pixels to maintain consistent input dimensions across the dataset.
- Random Rotation: To augment the dataset and introduce rotational variance, images are randomly rotated within a ±15-degree range.
- Random Cropping: Images are randomly cropped to a size of 224x224 pixels with padding, maintaining original dimensions. This step further expands the dataset and introduces transformational variance.
- Random Horizontal Flipping: Introducing horizontal flipping serves as another augmentation technique, ensuring the model's invariance to facial orientation.
- Tensor Transformation: Finally, the processed images are converted to PyTorch tensors, compatible with deep learning frameworks.

2.4 DataLoader Integration

To effectively feed the preprocessed data to the deep learning model during training, PyTorch's DataLoader class was used. This class batch processes data, randomly shuffles it for training, and parallelizes the loading process to optimize GPU utilization during training.

3. Model architecture

3.1 AlexNet

AlexNet is a monumental architecture in the deep learning community, which burst onto the scene in the 2012 ILSVRC competition and won first place with a performance difference of more than 10% over the second place. It consists of five convolutional layers followed by three fully-connected layers. Its features include the use of the ReLU activation function, overlapping pooling, parallel use of two GPUs, and a normalization called local response. At that time, the dataset used was a 3-channel color photo of 256x256x3.

The implementation of AlexNet in this project has been modified to accept an input of 224x224x1 for the base model.

3.2 VGGNet

VGGNet is a CNN Network developed by the Visual Geometry Group at Oxford University, and is famous for its simplicity and depth. Although it finished second in the 2014 ILSVRC, it is more widely used than GoogleNet, which won first place in the same competition, because of its simple structure that is easy to understand and modify. The most basic way to improve the performance of a CNN Network is to increase the depth of the network. VGGNet is a network developed to check the performance changes of other networks with different network depths.

The basic structure of VGGNet is simple, with a max-pooling following the Conv layer to reduce resolution. Like traditional CNN Networks, it connects three FC layers and then calculates the probability value of each class through a soft-max layer.

The convolutional layer is designed with a stride=1 and padding=1 to maintain the original resolution, and the resolution is gradually reduced through max-pooling with a 2x2 window size and stride=2 after the Conv layer. The number of channels starts at 64 and doubles each time the resolution is reduced by max-pooling. The Fully-Connected layer consists of a total of three layers. The first and second layers have 4,096 channels, and the last layer has 1,000 channels (the number of classes in the ImageNet Dataset).

The implementation of VGGNet in this project is an adjustment of VGGNet-16 to fit the current input. The model consists of four convolutional stages, each stage comprising two convolutional layers followed by maximum pooling. Batch normalization and ReLU activation are used after each convolution for stable and efficient training. The convolutional layers are followed by three fully connected layers leading up to the final classification layer.

3.3 ResNet

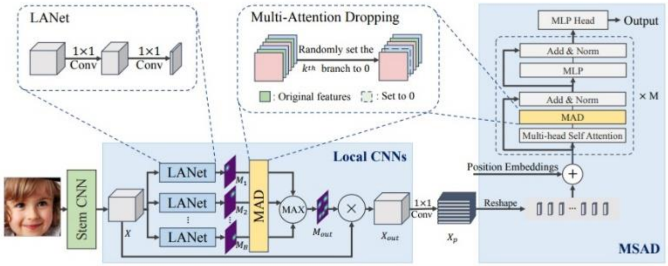
ResNet (Residual Network) is a neural network developed by Microsoft's Chinese researchers and won the 2015 ILSVRC competition. The remarkable thing is that the Top-5 error rate in the competition was 3.57%, which is recorded as the first neural network that surpassed humans since the error rate of humans is about 5%. ResNet is a deep neural network with 152 layers, which is incomparable to existing neural networks. Generally, the more complex the problem that needs to be solved, the deeper the neural network should be, but this is followed by the vanishing gradient or exploding gradient problem, and as the number of layers deepens, the number of parameters increases, leading to an increase in error. In other words, with the existing structure, deeper is not always better. The ResNet researchers realized that a change in structure is necessary to deepen the neural network. This is where the core of ResNet, residual learning, comes into play. Traditional neural networks are composed of connections between the i-th and (i+1)-th layers, but ResNet allowed connections between the i-th and (i+r)-th layers. This type of connection is called a 'shortcut connection.' By using a shortcut to connect two distant layers, there is the advantage that the gradient can be easily propagated to the previous layer during backpropagation.

The traditional method only has connections between neighboring layers and takes x as input to output H(x). It is common to aim to find the appropriate weight value through learning to find the optimal H(x). Let's say that the new goal is to obtain H(x)-x, the difference between output and input. If we call H(x)-x as F(x), then the resulting output H(x) becomes H(x)=F(x)+x. This method is the basic approach of Residual Learning. Through a shortcut, the input x of layer i is simply added to the output F(x) of layer (i+r), so only the addition operation is added without the need for parameters, and this is called identity mapping. ResNet basically follows the structure of VGG19. If you add convolution layers to VGG19 to increase the depth and add shortcuts, it becomes the simplest structure of ResNet.

The implementation of ResNet in this project has been modified to accept an input of 224x224x1 for the base model.

3.4 TransFER (Transformer Feature Extraction and Recognition) (Stem CNN+MAD+Transformer)

This model is a neural network called TransFER (Transformer Feature Extraction and Recognition), which combines the latest technologies in computer vision and natural language processing, proposed in the paper "TransFER: Learning Relation-aware Facial Expression Representations with Transformers". The following is a description of each component of the model and how it works.



3.4.1 Stem CNN (modified ResNet)

The paper used a CNN called IR-50, which has a structure similar to ResNet-50, but due to execution issues, the ResNet-18 model was used in this project. In ResNet-18, the first convolutional layer, conv1, has been modified to accept a 1-channel grayscale input.

The Stem CNN serves as a feature extractor and is composed of the layers of ResNet excluding the last two layers, which are the average pooling layer and the fully connected layer.

3.4.2 MAD (Multi-Attention Dropping)

This layer is used to prevent overfitting and improve the model's generalization ability, activated only during the training phase, and randomly sets some features of the input to '0' according to the dropout probability. This forces the model to learn less dependent feature representations.

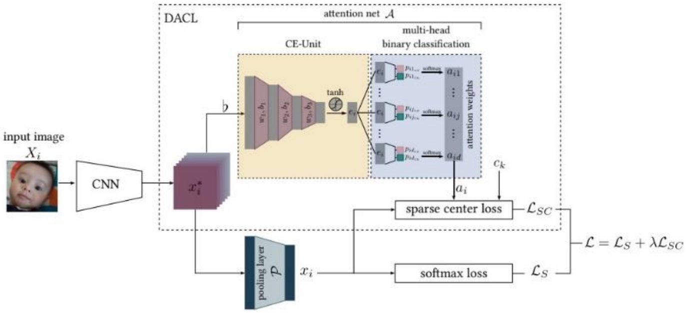
The Multi-Attention Dropping (MAD) described in the paper adopts a different approach from the standard dropout. While standard dropout randomly sets individual elements of an input feature vector or map to zero, MAD takes a group of input feature maps and entirely sets one selected feature map to zero. This process treats each feature map as a single unit, and the selected feature map is not activated during the training process.

3.4.3 TransFER (Transformer Feature Extraction and Recognition)

This layer is used to extract high-dimensional features from the input image and learn features for classification. The working principle is as follows: the local_cnn1 and local_cnn2 convolutional layers transform the feature maps of modified ResNet into a higher dimension. These two feature maps, with MAD applied, are expanded in dimension and concatenated. To adjust the feature maps for the transformer encoder, the dimensions are rearranged and flattened into a 1D sequence. The transformer_encoder models the global dependencies of each element in the sequence to learn more sophisticated features. This is achieved by using multiple encoder layers to learn the interactions of each element within the sequence. Finally, the classifier fully connected layer uses the output of the encoder to predict class probabilities.

3.5 DACL Model

The DACL Model is a deep learning architecture proposed in the paper "Facial Expression Recognition in the Wild via Deep Attentive Center Loss," which combines the advantages of a pre-trained network with the Attention mechanism. This model uses a pre-trained ResNet18 as the base model and includes an additional Attention Module for image classification.



3.5.1 Pre-trained ResNet18

In the paper, a pre-trained ResNet18 was used as the base CNN. While ResNet18 is originally designed to process color images (3 channels), the model for this project has been modified to accept grayscale images (1 channel) by altering the first convolution layer.

Removing the last FC (Fully Connected) layer: The traditional ResNet18 model has an FC layer at the end, but here nn.Identity() is used to maintain the output as is. This is to preserve the feature map for the Attention mechanism module.

3.5.2 Attention Mechanism Module

Internal convolution layers: This module has two convolution layers depending on the number of input channels. The first conv1 layer reduces the number of channels (for the query and key), and the second conv2 layer maintains the original number of channels (for the value).

Attention weight calculation using softmax: Attention weights are calculated through the matrix multiplication of the query and key, and the softmax function is applied to normalize these weights.

Matrix multiplication of the value and attention weights: The calculated attention weights are applied to the value to generate a weighted feature map.

Sum with the original input: The feature map created through the attention mechanism is combined with the original input to adjust the importance of each pixel.

3.5.3 Final Classification Layer

Flattening: The feature map that has gone through the attention mechanism is flattened into a one-dimensional vector.

Linear layer (FC): The flattened feature vector is passed to a linear layer for final classification. This layer is set according to the number of classes the model aims to predict.

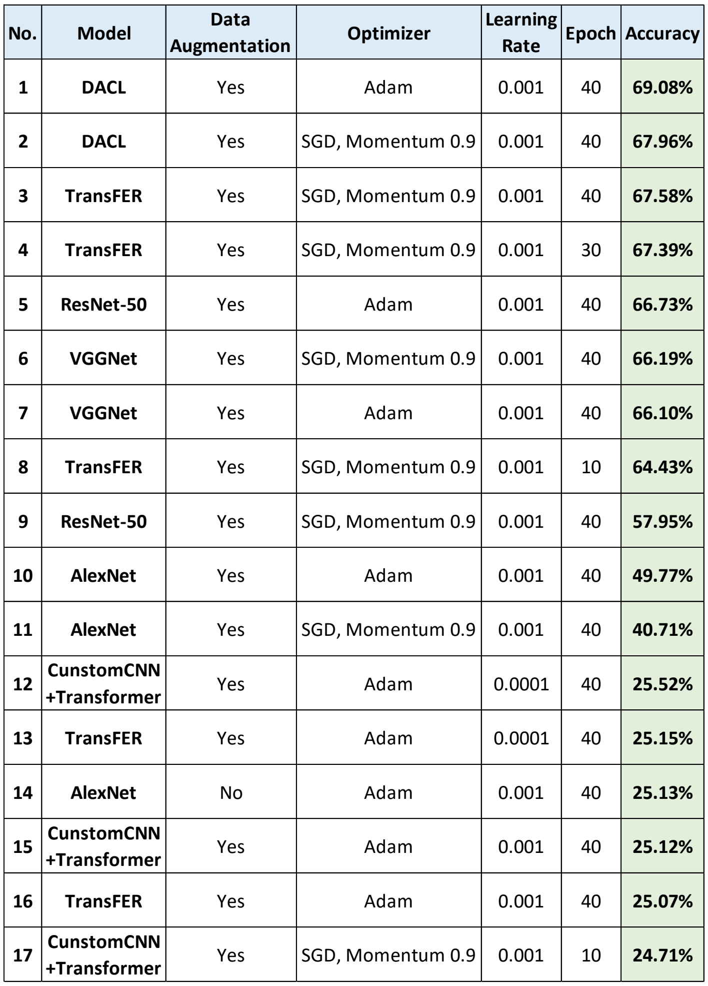
4. Results

Performance is as follows, presented in a table arranged from highest to lowest. The order of performance was DACL, TransFER, ResNet, VGGNet, AlexNet. When TransFER's stem CNN was replaced with a custom CNN, learning did not occur. The sensitivity of TransFER's structure is suggested by its failure to learn even when only the optimizer was switched to Adam.

Here's a closer look at the performance of each model:

- AlexNet's accuracy was about 50% with Adam and around 40% with SGD. Several other learning rates were tested, but 0.001 proved to be the best. Without data augmentation, no improvement in learning was observed.
- For VGGNet, an accuracy of 66% was observed with both SGD and Adam.
- ResNet showed 67% accuracy with Adam, but this dropped to 58% when using SGD.

- TransFER's performance was 68% with SGD. Notably, learning did not occur when using Adam.
- DACL, on the other hand, performed better with Adam than with the SGD used in the original paper. The performance was 68% with SGD and 69% with Adam.



No.	Model	Data Augmentation	Optimizer	Learning Rate	Epoch	Accuracy
1	**DACL**	Yes	Adam	0.001	40	**69.08%**
2	**DACL**	Yes	SGD, Momentum 0.9	0.001	40	**67.96%**
3	**TransFER**	Yes	SGD, Momentum 0.9	0.001	40	**67.58%**
4	**TransFER**	Yes	SGD, Momentum 0.9	0.001	30	**67.39%**
5	**ResNet-50**	Yes	Adam	0.001	40	**66.73%**
6	**VGGNet**	Yes	SGD, Momentum 0.9	0.001	40	**66.19%**
7	**VGGNet**	Yes	Adam	0.001	40	**66.10%**
8	**TransFER**	Yes	SGD, Momentum 0.9	0.001	10	**64.43%**
9	**ResNet-50**	Yes	SGD, Momentum 0.9	0.001	40	**57.95%**
10	**AlexNet**	Yes	Adam	0.001	40	**49.77%**
11	**AlexNet**	Yes	SGD, Momentum 0.9	0.001	40	**40.71%**
12	**CunstomCNN +Transformer**	Yes	Adam	0.0001	40	**25.52%**
13	**TransFER**	Yes	Adam	0.0001	40	**25.15%**
14	**AlexNet**	No	Adam	0.001	40	**25.13%**
15	**CunstomCNN +Transformer**	Yes	Adam	0.001	40	**25.12%**
16	**TransFER**	Yes	Adam	0.001	40	**25.07%**
17	**CunstomCNN +Transformer**	Yes	SGD, Momentum 0.9	0.001	10	**24.71%**

5. Conclusion

Facial Emotion Recognition (FER) is a significant challenge in the field of computer vision, profoundly impacting various applications from human-computer interaction to mental health monitoring. This research explored five different deep learning architectures, including AlexNet, VGGNet, ResNet, CNN with Transformer (TransFER), and CNN with Attention Mechanism (DACL), to address this challenge.

Several key points were discerned from the results:

- Depth Matters: The performance improvement from AlexNet to VGGNet and ResNet emphasizes the importance of depth in neural architectures. Deeper networks tend to exhibit superior performance in complex tasks.
- Innovation Leads to Improvement: The introduction of the Transformer and Attention mechanisms in the TransFER and DACL models marks a departure from traditional CNN architectures, enabling the capture of global dependencies in data and thereby enhancing model performance.
- Importance of Regularization: Techniques like Masked Activation Dropping (MAD) not only prevent overfitting but also introduce beneficial stochasticity in the training process, leading to robust and generalized models.
- Comprehensive Training Methodology: Our training strategy, including learning rate and number of epochs, underscores the importance of a well-designed training methodology.
- Future Directions: The outcomes of this project indicate substantial room for improvement. Future research may explore directions such as integrating Attention mechanisms, exploring more advanced regularization techniques, or investigating ensemble methods that combine the strengths of various architectures.

In conclusion, this project was a modest experiment in facial emotion recognition. Through experimentation and analysis, I sought to reveal the strengths and limitations of the architectures. This lays the groundwork for future explorations. Step by step, we move closer to the dream of machines understanding human emotions in the same way we do.

Code in GitHub:
https://github.com/sun2423/CS577DLProject

Reference:

1. Fanglei Xue et al., TransFER: Learning Relation-aware Facial Expression Representations with Transformers, ICCV 2021
2. Amir Hossein Farzaneh et al., Facial Expression Recognition in the Wild via Deep Attentive Center Loss, WACV 2021