Facial Expression Recognition using Deep Neural Network

Gulshan Madhwani ([gmadhwan@iu.edu](mailto:gmadhwan@iu.edu)), Ritesh Ta

wde ([rtawde@iu.edu](mailto:rtawde@iu.edu))

Under the guidance of Prof. David Crandall ([djcran@iu.edu](mailto:djcran@iu.edu))

1. Abstract:

We propose a Deep Neural Network (DNN) architecture for facial expression recognition. The proposed architecture is independent of any hand-crafted feature extraction and is completely automated. Network learns various features automatically to provide insights and better understanding. Automatic expression recognition has wide range of applications such as human-computer interaction, psychiatric counselling, criminal investigation, etc. The proposed architecture is the efficient solution for identifying facial expressions through techniques of computer vision and machine learning.

1. Introduction

Facial expression are one of the core aspect of human communication. They depict non-verbal cues, and play an important role in interpersonal relations. Being able to understand one's emotions and the feelings is an important factor for an appropriate and correct understanding.

Automatic recognition of human expressions can be important component of human-machine interfaces; it may also be used in behavioral sciences and in clinical practice. Although humans recognize expression with no effort, it still remains challenging for the machines to do the same.

In this work, we present an approach for human facial expressions recognition through the techniques of Deep Neural Network (DNN).

1. Dataset

For this application, we have used a labelled dataset from kaggle.com. This is an entirely new dataset for one of the challenges in representational learning for facial expression recognition. Dataset contains images over three categories, i.e., training, public test and private test. Dataset contains data for 7 expression in total viz., Angry, Disgust, Fear, Happy, Sad, Surprise and Neutral.

The data consists of 48x48 pixel grayscale images of human faces arranged in row major order. It means that 48x48 images are flattened to be fit in a row. Images are more or less centered over faces.

Dataset consists of 3 columns as follows:

* 1. Emotion : Specifies class label associated with each of the expressions. They are labelled as (0 – Anger, 1 – Disgust, 2 – Fear, 3 – Happy, 4 – Sad, 5 – Surprise, 6 – Neutral )
  2. Pixels : String of space-separated pixel values in row major order.
  3. Usage : Training, public test or private test

Though this dataset provides data with 7 types of expression, for the purpose of this project, we are focusing only on first 4 expressions.

Distribution of all the samples for these 4 expressions is given in the following table.

|  |  |
| --- | --- |
| Category | # of samples |
| Training | 15743 |
| Public Test | 1915 |
| Private Test | 1954 |

1. Software packages

To implement methodologies to achieve task of expression recognition, we've used python and its associated library to ease our task.

As we operate on matrices of pixel values at core levels, linear algebra techniques have been used to solve the problem. Libraries used are :

* 1. Numpy : for linear algebra operations
  2. Matplotlib : For plotting the graph

1. Terminologies
   * Neural networks:

Neural networks are a set of algorithms, modeled loosely after the human brain, that are designed to recognize patterns. They help in classifying different objects. They group unlabeled data according to similarities of the inputs, and they classify data when they have a labeled dataset to train on. The pattern they recognize are real-world data.

* + Feed forward networks:

Goal of neural networks is to arrive at a point of least error as fast as possible. Each step in neural network involves a guess, an error measurement and a slight update in weight parameters, an incremental adjustment to the coefficients. Input of one layer is forwarded to the next layer in feed forward fashion with adjustments as stated above until final layer is reached.

* + Deep neural networks:

Deep learning neural networks are distinguished from simple neural networks by their depth, that is, the number of node layers through which data passes in a multistep process of pattern recognition. In deep nets, each layer of nodes is connected to every other node in the previous and next layer called as fully-connected network and trains on a distinct set of features based on the previous layer's output. The more deep you go in neural net, the more complex features can be recognized by the nodes, since they aggregate and recombine features from the previous layer.

* + Activation function:

The role of activation function is to make neural networks non-linear. Consider following example to understand importance of activation function.

Let Li denotes the output of the i-th layer. Then the equation of fully-connected feed-forward nets says Li = f (Wi Li-1) for i>1, where f is the activation function. If we miss on f, then we get

Li = WT LT-1 = WT WT-1 LT-2 = …. = (WT WT-1 … W1)W0

Let W = WT WT-1 ...W1 , then LT =W L0

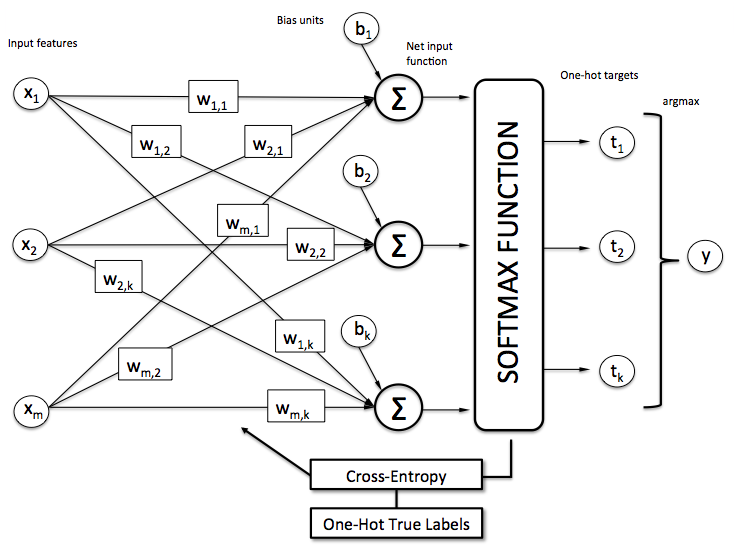
This shows that without activation function, deep neural network has degenerated to a linear transformation, which is not strong enough and makes it slow to train the network.

* + Back propagation:

Back propagation is the gradient descent step for the neural network. It helps in propagation backward in the network in order to find the possible error at each layer of neural network and to adjust the weights and biases accordingly.

* + Softmax:

Softmax function calculates the conditional probability of belonging to class y given input. It is given by the following formula.

P(y=j | z(i)) = Φ softmax(z(i)) =

* + Regularization:

Figure 1 : Softmax (Image courtesy: http://www.kdnuggets.com/2016/07/softmax-regression-related-logistic-regression.html)

Regularization is a technique to avoid overfitting problem.

1. Architecture:
2. Methodologies

To train the facial expression recognition task using DNN, we go through different steps as follows:

* + After the training data is loaded, we take initial weights for both the hidden and the output layer randomly using gaussian normal distribution with normalization.
  + Biases for hidden as well as output layer are set to zeros initially as those will be updated in later iterations during back propagation.
  + Parameters used are:
    - # of epochs : 10000
    - # of hidden layers : 200
    - Regularization : 10-7
    - Learning rate : 10-7
* We’ve used one-hot encoding scheme to transform categorical features to a format that works better with the classification algorithms.
* For activation function, hyperbolic tangent function is applied instead of sigmoid as logistic sigmoid can cause neural network to get stuck during training. Alternative to logistic sigmoid is the tangent function as shown below:

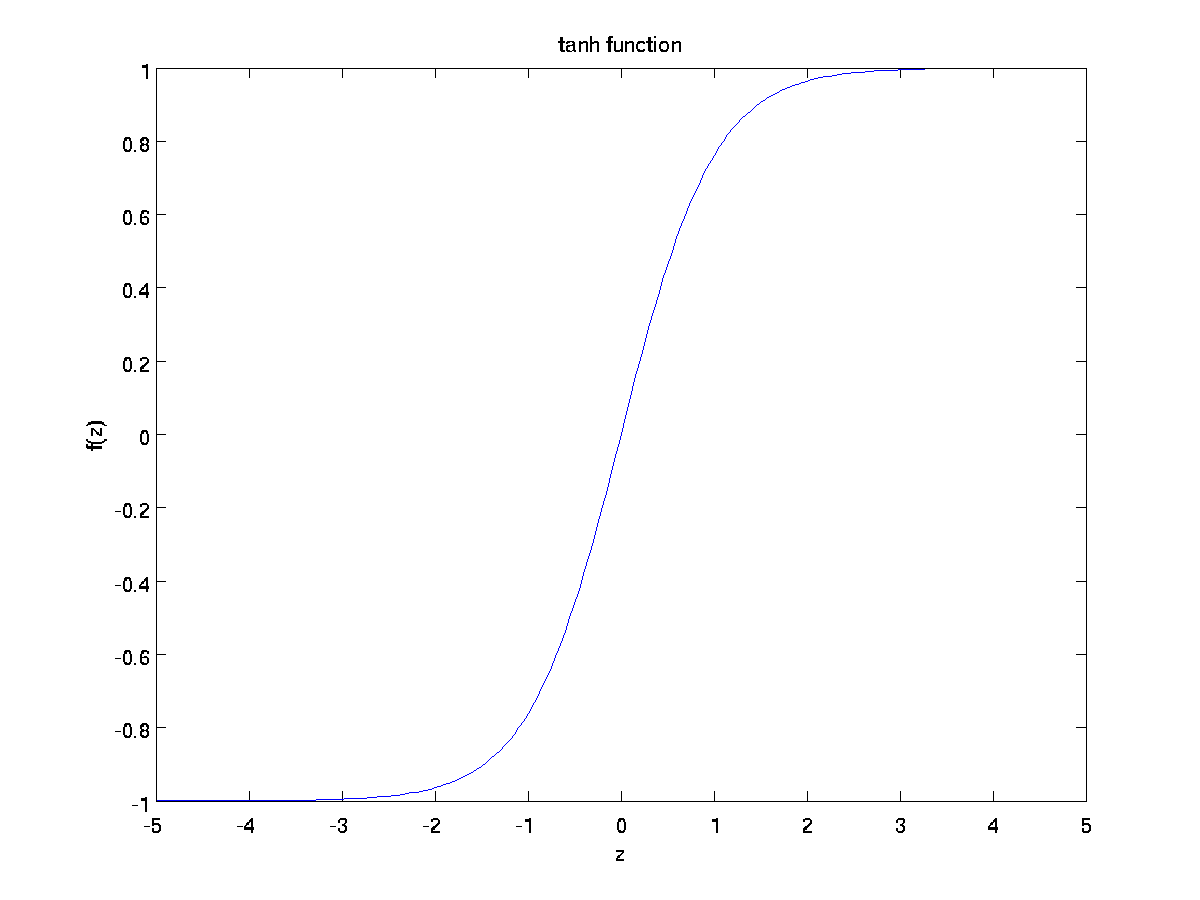


Figure 2: tanh function (Image courtesy: http://ufldl.stanford.edu/wiki/images/a/aa/Tanh\_Function.png)

* Gradient descent step is taken in each iteration to update the weights and bias.
* Softmax is applied to get the last layer results.
* For estimating the cost function in the training phase, cross entropy is used. At each 50th iteration, trained model is tested to estimate the cost and error to plot a function of cost and iterations at the end.

(***still more methodologies to be explained***)

1. Results
2. Applications
3. Conclusion

We developed deep neural network for targeting facial expression recognition problem. The results we obtained from deep neural nets are robust enough to learn facial features and help recognizing facial expressions. We did not use any feature engineering for the input data and trained the network using raw pixel data with the significant success rate as observed from the experimental results.

1. Future Work

The amount of work depicted here using Deep Neural Network is a preliminary but significant step for facial expressions recognition. There is still a long way to go to achieve near 100% accuracy. There are several ways through which we can obtain it such as:

* Images used were grayscale images; we can better train the network using color images.
* We can amalgamate face detection followed by the expression recognition in real time.
* Here, we focused on 4 facial expressions. There are still more expressions to cover for.
* More promising and contemporary approaches such as Convolutional Neural Network (CNN) can be used which have shown great impacts in the field of computer vision.

1. References

<https://www.kaggle.com/c/challenges-inrepresentation-learning-facial-expressionrecognition-challenge/data>

Shima Alizadeh, Azar Fazel (2016). Convolutional Neural Networks for Facial Expression Recognition

Arushi Raghuvanshi, Vivek Choksi (2016). Facial Expression Recognition with Convolutional Neural Networks

Christopher Pramerdorfer, Martin Kampel (2016). Facial Expression Recognition using Convolutional Neural Networks: State of the Art

Peter Burkert, Felix Trier, Muhammad Zeshan Afzal, Andreas Dengel, Marcus Liwick. DeXpression: Deep Convolutional Neural Network for Expression Recognition

<https://github.com/lazyprogrammer>

<https://www.udemy.com/data-science-logistic-regression-in-python>

<https://docs.scipy.org/doc/>

<https://matplotlib.org/contents.html>

<https://deeplearning4j.org/neuralnet-overview>

<https://www.quora.com/What-is-the-role-of-the-activation-function-in-a-neural-network>

<https://www.quora.com/How-does-back-propagation-work>

<https://www.quora.com/What-is-one-hot-encoding-and-when-is-it-used-in-data-science>