International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

Mood Analysis using Neural Networks

Debabrata Datta, Anubhav Kumar Roy, Suchandra Datta, Upasana Roy

Abstract*: *Identifying mood of a person using the facial expression has been an area of research interest. The research work described in this paper aims to propose a methodology to do the same. In this work, the input from a webcam is fed to the system which is captured frame by frame. The images are preprocessed and a model based on Convolutional Neural Network (CNN) has been used to predict the emotion of a person. The CNN model is trained using the FER2013 dataset to extract features for identification of emotions. The work flow consists of procuring model for training, cleaning the data, building the model and fine-tuning hyperparameters for an optimal level of accuracy. Parallel to this, the functions have been created to take input from webcam and to process the input to feed the model. The final step has been to integrate all the units. The testing of the application has been done for each function separately, then for the entire work as a whole.

Index Terms*: *Artificial Neural Network, Classification, Convolutional Neural Network, Facial Expression, Hyperparameter.

I. INTRODUCTION

The rapid growth of technology has brought with it interesting applications of the same, ranging from training networks to automatically generating captions from images to development of sophisticated robots. With the rising popularity of Convolutional Neural Networks as the network of choice for working with images, an attempt has been made to create a system which predicts the emotion of a person based on his facial expression.

Methods employed for face detection have drastically improved from usage of manual measurements to application of linear algebra (Eigen face approach) to employment of a cascade of weak classifiers [1]. The latter uses HAAR based feature selection, image representation as integral images such that computations maybe done in real-time and fed to the weighted simple weak classifiers [1]. The latest approach is deep learning based, involving multi-layered neural network and Histogram of Gradients (HOG) technique. Formerly for image representation, computation of features by hand was used, followed by bag-of-words coding scheme, then feature pooling and traditional classification algorithms like Support Vector Machines [4] and random forests [5]. It would be of interest if the features could be learnt by the computational model itself with minimum human intervention. This was realized with development of convolutional neural networks (CNN) which perform extensively well for large-scale classification problems [6]. CNN is a further improvement on artificial neural networks (ANN). Artificial neural networks, as the name suggests, are used to implement the biological counterparts using electrical signals, which serve as an underlying framework for machine learning algorithms to execute and process complex data that are affected and dependant on a host of different parameters [2,3]. This is employed for tasks like mood analysis, gender and age detection. The huge potential of CNN is only recently brought into the fore with the rise of Graphical Processing Units, the plethora of training data available and development of more effective methods to train the complex models. The paper has been organized in the following way:

Section II of this paper contains an overview of similar research work in this field, section III outlines the technology used for implementation along with the pseudocode used for the development of the application, section IV discusses the results obtained with focus on limitations and applications of the system. Section V concludes the discussion with the future scope of work.

II. RELATED WORK

Yan, Kriegman, and Ahuja formulated a classification for face detection methodologies which are of four types [7]:

1. The knowledge based method is dependent on a set of rules like relative distance between eyes, nose, mouth, expected positions of the same. There is a danger of obtaining false positives if the rules are too general or detailed.
2. The feature based method extracts features from images then using a classifier it differentiates between facial and non-facial regions.
3. In the appearance based method, a training set is used to build a face detection model using one of the following approaches. To implement this, many algorithms like Principal Component Analysis, Support Vector Machine, and Neural Networks etc. have been used.

A lot of work has been done in connection with sentiment analysis from text, images or both as explained in [12] where tweets were analyzed; the textual data were fed to one CNN model and the visual data were fed to another CNN model and the outcome was then fed to another CNN model to understand the relationship between the two.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Debabrata Datta*, Department of Computer Science, St. Xavier's College(Autonomous), Kolkata, India.
Anubhav Kumar Roy, Department of Computer Science, St. Xavier's College(Autonomous), Kolkata, India.
Suchandra Datta, Department of Computer Science, St. Xavier's College(Autonomous), Kolkata, India.
Upasana Roy, Department of Computer Science, St. Xavier's College(Autonomous), Kolkata, India.

A similar approach has been followed in [13] where a CNN was trained for textual sentiment analysis by employing transfer learning and a Deep Neural Network was trained with generalized dropout for sentiment analysis from images. An impressive set of results was obtained also for visual mood analysis by applying a CNN model [13]. As discussed in [6], a deep convolutional neural network was capable of achieving significant outcome for image classification as well. Object detection using Haar feature-based cascade classifiers has been an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001[1]. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images.

III. METHODOLOGY

The problem of identifying an emotion on a face is a classification problem with seven classes (happy, sad, fear, angry, disgust, surprise and neutral). A convolutional neural network has been constructed to solve the problem as it is adept at pattern recognition due to its capability to generalize feature extraction from images. The following steps based on CNN were considered to design the final model:

The first step was feature extraction which was achieved by a convolution operation between the input image matrix and the feature detector known as a kernel or a filter. The feature detector has helped to identify whether the image had a smile or not, the curvature of the eyebrows, which were typically raised to register surprise and such features of a face which would help to identify the emotion on the face. The outcome of the convolution operation was another matrix of same dimensions as the feature detector. For each image, a large number of filters were applied and for the proposed model, it was 64 filters in the first three convolution layers and then 128 filters for the next two convolutional layers, each of which has produced a corresponding feature map.

In the next step, the feature maps were fed as input to the activation function of CNN which was selected to be the Rectified Linear Unit activation function that has replaced all the negative values with a zero value. This has introduced a non-linearity within the network.

After this, a method called Pooling or Downsampling was performed to reduce the redundant features in the feature detector. Max pooling in which the maximum of the values were chosen is performed in the early layers to extract the major important features. Then average pooling was performed in the later layers since it gave a more continuous output by taking the average of the pixel values. Pooling was performed to reduce the number of features.

The last stage of the methodology was to convert all the pooled feature maps into a 1D array, a process known as flattening. This array was fed as input to the final part of the CNN which was an artificial neural network. The input nodes however were not all equally important since there were some features which were more important for mood detection than others. Since the model had no idea beforehand about the importance of the features, it has randomly assigned zero to a percentage of the input nodes known as the dropout value which prevented the model from overfitting [11].

To design the methodology, two main steps have been incorporated and they were data preprocessing and data modeling.

In the data preprocessing stage, the dataset that was considered was taken from [14] and was known as FER2013 dataset. This dataset was prepared by Pierre-Luc Carrier and Aaron Courville [14], as part of an ongoing research project and this dataset contains two columns, "emotion" and "pixels". The "emotion" column contains a numeric code ranging from 0 to 6 (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral), for the emotion that was present in the image. The "pixels" column contained a string that was converted to a 48*48 matrix which was then converted to the grayscale format before being fed to the model. The numeric values for the emotions have been fed to the model in a binary string as mentioned below:

Since there were 7 emotions a 1D array of size 7 was considered where, to represent the angry emotion, only the bit in 0th position was set to 1, then to represent disgust, only the bit at the 1st location was set to 1 and so on. So, the prediction for the surprise emotion was represented as [0 0 0 0 0 1 0]. Since the implementation was done in Python, the dataset was read using ‘pandas’ tool which returned a dataframe which was converted to a list. 80% of the images were used for training and the rest of the images were used for the testing purpose. The seed value that was considered was 42 for both the training and the testing.

For the data preprocessing, the batch size was an important parameter which referred to the number of samples of the training data the algorithm must iterate over before the internal parameters were updated. The batch was initially chosen to be of size 64 but the accuracy of the model did not increase with each iteration as expected. When the batches of sizes 512 upwards were selected, the loss value continued to decrease with each iteration. Another important parameter regarding data preprocessing was Epoch which was the number of times the algorithm worked over the dataset. After hyperparameter tuning, the selected batch size was 1024 and the selected epoch was 10.

After data preprocessing was over, data model was trained. To train the model, Google Colab was used which was a free platform provided by Google to train intensive machine learning models without the need for developers to buy expensive hardware. Using Google Colab, the model was trained on Tesla K80 GPU in a short interval of time. Another tool that was used in this stage was Keras [10] which was a high-level neural networks API, capable of running on top of existing libraries like CTNK or Theano or more importantly TensorFlow.

The first major algorithm in the proposed work was face detection and mood analysis whose steps are stated below:

Step 1: Start.
Step 2: Read the required XML files for face detection in OpenCV.
Step 3: Start the capture from the webcam.
Step 4: Load the model for emotion detection stored in JSON format.
Step 5: Load the model weights stored in h5 format.
Step 6: Capture a frame and convert that into the grayscale format.

Step 7: Detect the face present in the image using OpenCV function detectMultiScale().
Step 8: For each coordinate value (x,y,w,h) in faces go to step 9 else go to step 16.
Step 9: Crop face from original image. Convert the cropped image to the grayscale format and resize that to 48*48.
Step 10: Convert the pixel matrix to an array from a list. Change shape to (48,48,1) as this is the expected input to the model.
Step 11: Divide each pixel by 255 to normalize pixels to a range of 0 to 1 from 0 to 255.
Step 12: Use predict function to predict the emotion from image.
Step 13: From the linear array returned in step 12, find the item with largest value.
Step 14: Find the index of the largest term in the array. Print the emotion at this index in emotion array by printing a rectangle around the face.
Step 15: If all the rectangles in the faces have been processed go to step 16 else go to step 8.
Step 16: Print the final image with the emotion labels.
Step 17: If 'q' is pressed, close all the windows and stop the capture from the webcam else go to step 6.
Step 18: Stop.

The next major algorithm in the proposed work was to create and to train the convolutional neural networks whose steps are stated next:
Step 1: Start.
Step 2: Read the .csv file using pandas tool function read_csv ( ).
Step 3: Convert each row under the column 'pixels' into a list from a vector.
Step 4: For each item in the list created in step 3, go to step 5 else go to step 11.
Step 5: Create a list of size 48*48.
Step 6: Initialize a variable i by 0.
Step 7: if i<48 go to step 8 else go to step 10.
Step 8: Read pixel values from i *48 till i*48+48 and store as a list at index i of another list (list of lists).
Step 9: Increment i by 1 and go to step 7.
Step 10: Convert the list into an array and append to another list. Go to step 4.
Step 11: Convert the final list into an array. Divide by 255 for normalization.
Step 12: Convert each entry under the 'emotion' column to a list.
Step 13: Convert the list to an array.
Step 14: Convert the list in step 13 to categorical using the function available.
Step 15: Split the datasets into training and test sets.
Step 16: Create a model with the 1st convolution layer having 64 output filters, kernel of size 5*5, and an activation function that accepts inputs of shape (48*48*1). For Max pooling, pool size (factors by which to downscale) is chosen to be 5*5, number of pixels to shift over is a window of size 2*2.
Step 17: Create the 2nd convolution layer with 64 output filters as well; two convolution layers of 64 output filters of size 3*3. For average pooling, pool size is 3*3, strides is 2*2.
Step 18: 3rd layer has 2 convolution layers of 128 filters of size 3*3, pooling same as in step 17.
Step 19: Flatten.
Step 20: Connect the fully connected layers with appropriate dropout.
Step 21: Reshape the input as (number of samples, 48,48,1) as 4D tensor is expected as an input.
Step 22: Create batches of 1024 size, epochs 10. Using inbuilt functions, training is done.
Step 23: Calculate the training loss, the training accuracy, the test loss and the test accuracy.
Step 24: Create the confusion matrix.
Step 25: Save model to drive written in JASON format.
Step 26: Save model weights in h5 format.
Step 27: Stop.

IV. RESULTS AND DISCUSSION

For the development of the model described in the previous section, the following tools were used:
Python was used as a programming tool. For data manipulation, extraction and preparation from CSV files, JSON and SQL databases, pandas tool was used. Specifically, for image processing, OpenCV was used. The algorithms were implemented on the FER2013 dataset as mentioned earlier.
The final confusion matrix that was obtained on applying the methodology is shown in table 1.

Table 1. Confusion Matrix For The Final Model

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	517	11	99	57	137	26	138
Disgust	24	58	11	3	2	1	3
Fear	128	4	488	59	178	73	113
Happy	82	5	63	1355	75	43	142
Sad	171	9	155	100	515	18	242
Surprise	45	1	92	36	20	561	40
Neutral	161	3	82	140	159	25	708

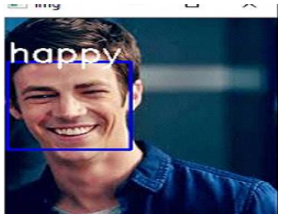

Figure 1. A Sample Image Of A Happy Face

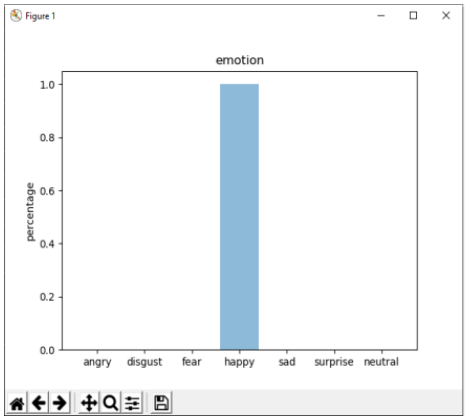

Figure 2. Prediction For The Image Given In Figure 1

'Figure 3. Another Sample Image

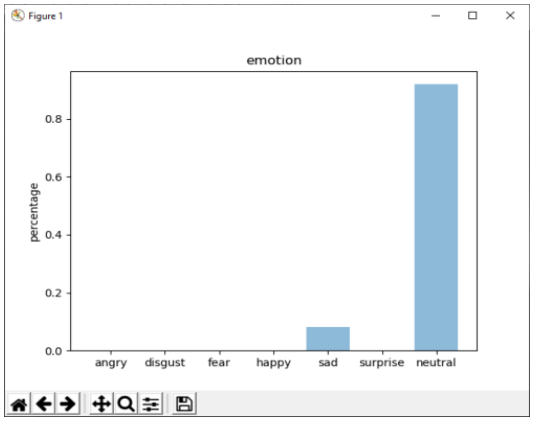

Figure 4. Prediction For The Image Given In Figure 3

The results obtained are based on some constraints like the faces must not be too tilted; faces must be highlighted with sufficient light etc. But if these constraints are satisfied, the prediction of the results has been more than 90% accurate.

V. CONCLUSION AND FUTURE SCOPE

The results demonstrated in the previous section may be applied in different areas like shopping malls to identify the level of satisfactions for the customers, in educational institutions specifically schools to detect the likings of the students etc.

From the confusion matrix of mood detection, it has been noticed that anger was misclassified as sad and neutral. Surprise was misclassified as fear. For better performance, the model may be trained on an even larger dataset. If that is not available, then data augmentation may be another option. Augmentation may be random rotations, flipping, ZCA whitening etc.

Since the artificial intelligence is regularly expanding by newer dimensions of technology, a number of learning algorithms may be used to produce a conclusive result better than that obtained using a single algorithm like CNN. With more advanced algorithms and faster hardware being extensively researched and manufactured very frequently, allocations like mood prediction through image analysis may be implemented in a more comprehensive way.

As far as the scope of improvement is concerned, the present algorithm may be fed with more number of emotions by training on a new dataset with more emotion labels. This will increase the reach of the application. Transfer learning may also increase the performance of the algorithm as this learning is a concept in which a pre-trained model is selected and incorporated with a new model. The new model maybe trained on a relatively small dataset. Training networks from scratch requires huge amounts of data and time so transfer learning is a good option. For mood analysis, VGG 16 network architecture could be used with a few layers added on top. By fine-tuning a model, feature extraction will take place more effectively.

REFERENCES

1. Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features", In the Proceedings of the IEEE Computer Science Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, USA.
2. Marcel van Gerven and Sander Bohte, "Artificial Neural Networks as Models of Neural Information Processing", Frontiers in Computational Neuroscience, December, 2017.
3. Ms. Sonali. B. Maind and Ms. Priyanka Wankar "Research Paper on Basic of Artificial Neural Network", International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 2 Issue 1, ISSN: 2321-8169, pp. 96 – 100.
4. C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," ACM Transactions on Inteligent Systems and Technology, Vol. 2, Issue 3, April, 2011.
5. L. Breiman, "Random forests," Macine Learning, Vol. 45, Issue 1, pp. 5 – 32, October, 2001.
6. A. Krizhevsky, I. Sutskever and G. Hinton, "Imagenet classification with deep convolutional neural networks," In Proceedings of the 25th International Conference on Neural Information Processing Systems, pp. 1097 – 1105, December, 2012.
7. Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja, "Detecting Faces in Images: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 1, January 2002.
8. Alberto Albiol, David Monzo, Antoine Martin, Jorge Sastre, and Antonio Albiol, "Face recognition using HOG-EBGM", Pattern Recognition Letters, Vol. 29, Issue 10, pp. 1537 – 1543, July, 2008.
9. Yi Qing Wang, "An Analysis of the Viola-Jones Face Detection Algorithm", Image Processing On Line, pp. 128 – 148, June, 2014.
10. Karan Chauhan and Shrwan Ram, "Image Classification with Deep Learning and Comparison between Different Convolutional Neural Network Structures using Tensorflow and Keras", International Journal of Advance Engineering and Research Development, Vol. 5, Issue 2, February, 2018.
11. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever and Ruslan Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", Journal of Machine Learning Research, pp. 1929 – 1958, June, 2014.
12. Guoyong Cai and Binbin Xia, "Convolutional Neural Networks for Multimedia Sentiment Analysis", In Proceedings of the 4th CCF Conference on Natural Language Processing and Chinese Computing, pp. 159 – 167, October, 2015.
13. Q. You, J. Luo, H. Jin, and J. Yang, "Joint Visual-Textual Sentiment Analysis with Deep Neural Networks", In Proceedings of the 23rd Annual ACM Conference on Multimedia, pp. 1071 – 1074, Octorber 2015.
14. https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data, last accessed on 23rd June, 2019 at 11:45 am.

AUTHORS PROFILE

Mr. Debabrata Datta pursued Master of Technology from University of Calcutta, India and he is currently pursuing his Ph.D. in Technology from the same university. He is an Assistant Professor in the department of Computer Science, St. Xavier's College (Autonomous), Kolkata, India He is a life member of IETE. He has published 25 research papers in different reputed international journals and conferences. His main research interest is in the field of Data Analysis. He has more than 11 years of teaching experience and has more than 5 years of research experience.

Mr. Anubhav Kumar Roy has completed his B.Sc. with honours in Computer Science from St. Xavier's College (Autonomous), Kolkata, India. He is currently pursuing post graduation in Computer Science from the same institute.

Miss Suchandra Datta pursued her B.Sc. with honours in Computer Science from St. Xavier's College (Autonomous), Kolkata, India. She has already published two research papers in two reputed journals during her undergraduate study. She is currently pursuing post graduation in Computer Science from the same institute.

Miss Upasana Roy pursued her B.Sc. with honours in Computer Science from St. Xavier's College (Autonomous), Kolkata, India. She is currently pursuing post graduation in Computer Science from the same institute.