International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

CONVOLUTIONAL NEURAL NETWORKS (CNN) AND LONG SHORT TERM-MEMORY (LSTM) BASED APPLICATION PROGRAMMING INTERFACE (API) TO PREDICT EMOTION OVER WEB.

M Priyadharshini[1], M Nisha[2], Muppala Navya Sree[3], v sathya[4]

[1]*Student, B.E Computer science and Engineering, Panimalar Engineering College, Tamilnadu, India*
[2]*Student, B.E Computer science and Engineering, Panimalar Engineering College, Tamilnadu, India*
[3]*Student, B.E Computer science and Engineering, Panimalar Engineering College, Tamilnadu, India*
[4]*Associate Professor, Dept. of Computer Science Engineering, Panimalar Engineering College, Tamilnadu, India*

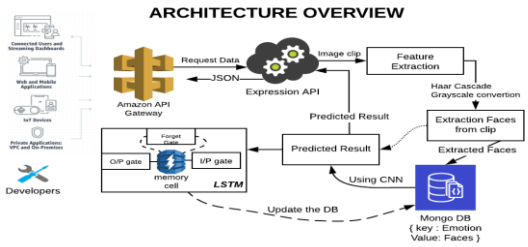
Abstract - *Human emotion plays an important role in the interpersonal relationship. Automatic recognition of emotion has been an active research topic from early years and efficiency has been a challenge. In this project we build an Application Programming Interface (API) to predict facial expression powered by Amazon EC2 Linux instance. Algorithms and models like Long Short Term-Memory (LSTM) and Convolutional Neural Networks (CNN) are used for training and performance evaluation. The potential application of this application includes Facial Expression based feedback in OTG platforms, Facial Expression detection in online interviews, Facial Expression Recognition for security, IOT devices, Connected users and streaming dashboards. Effectiveness of the API is measured through several defects and experiments by backpropagation of errors. The main goal of this project is to produce an enhanced performance level with greater accuracy.*

Key Words: Deep Learning, Cloud Computing (Flask), Web Development, Image Processing, AI, Port Sharing, SSH, Face Recognition, Emotion Detection, Application Programming Interface (API), Long Short Term-Memory (LSTM), Convolutional Neural Networks (CNN), Feedback Loop, Model Training, Data Cleaning.

1. INTRODUCTION

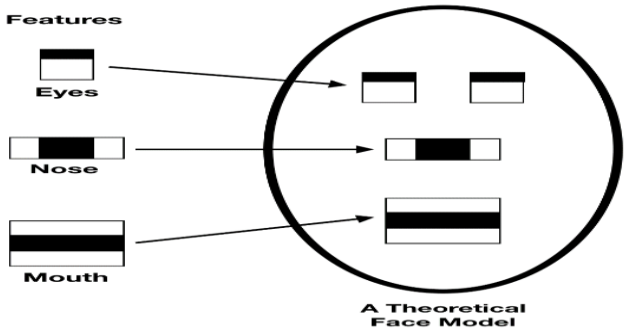
The development and usage of computer systems, software and networks are growing tremendously. These systems have an important role in our everyday life and they make human life much easier. Emotion plays a vital role in determining the thoughts, behavior and feeling of a human. An emotion recognition system can be built by utilizing the benefits of deep learning and different applications such as feedback analysis, face unlocking etc. Can be implemented with good accuracy. Facial emotion recognition system assumes a lot of importance in the era since it can capture the human behavior, feelings, intensions etc. The conventional methods have limited speed and have less accuracy while facial emotion recognition systems using deep learning has proved to be the better one. In this project, we use an Amazon EC2 Linux instance to create an Application Programming Interface (API) that predicts facial expression. Training and performance assessment are carried out using algorithms and models such as Long Short Term-Memory (LSTM) and Convolutional Neural Networks (CNN). A Fully Convolutional Neural Network based classifier is used to predict the facial emotions of the person in the image. An FCN (Fully Convolutional Networks) is a network composed of only convolutional layers, batch norms and non-linearities, i.e., it has no fully Connected (or Dense) layers. Whether s face exists in the image or not is determined by a Viola Jones Detector. Facial Expression Based Feedback in OTG Platforms, Facial Expression Detection in Online Interviews, Facial Expression Recognition for Security, IoT Devices, Connected Users, and Streaming Dashboards are some of the potential applications for this application. Backpropagation of errors is used to assess the API's effectiveness after multiple defeats and experiments.

1.1 Collecting, preparing and processing image Data



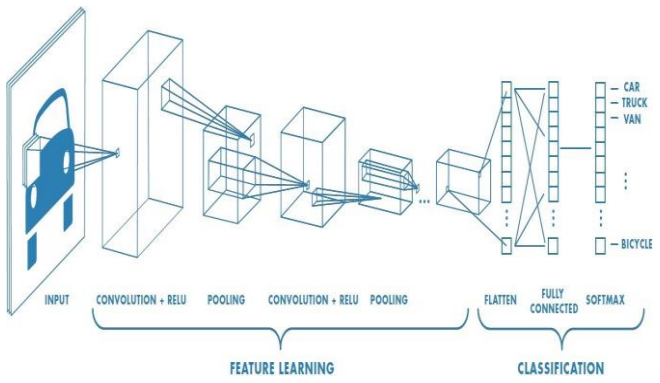
The process of gathering data depends on the type of project we desire to make. The data set can be collected from various sources such as a file, database, sensor and many other such sources but the collected data cannot be used directly for performing the analysis process as there might be a lot of missing data, extremely large values, unorganized text data or noisy data. We can also use some free data sets which are present on the internet. Kaggle and UCI machine learning repository are the repositories that are used the most for making Deep Learning models. In this project we use self-collected data to train the model, additional to that we have also used fer2013 facial dataset to improve our

accuracy. Haar Cascade classifier is an effective object detection approach. This is basically a machine learning based approach where a cascade function is trained from a lot of images both positive and negative. Based on the training it is then used to detect the objects in the other images.



1.2 CNN based Training

A Convolutional Neural Network (ConNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one form the other. The pre-processing required in a ConNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConNet have the ability to learn these filters/characteristics. The architecture of a ConNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlaps to cover the entire visual area. In this project we use CNN to train the model based on Facial datasets. The model is later exported so that Further training can be done over cloud along with LSTM.



1.3 Exporting the model to the Cloud

The model generated after CNN is transferred to cloud (AZURE) and configuration is done through SSH. SSH provides a secure channel over an unsecured network by using a client-server architecture, connecting an SSH client application with an SSH server. Th protocol, specification distinguishes between two major versions, referred to as SSH-1 and SSh-2. The standard TCP port for SSH is 22. SSH is generally used to access Unix-like operating systems, but it can also be used on Microsoft Windows 10 uses OpenSSH as its default SSH client and SSH was designed as a replacement for Telnet and for unsecured remote shell protocols such as the Berkeley rsh and the related rlogin and rexes' protocols. Those protocols send information, notably passwords, in plaintext, rendering them susceptible to interception and disclosure using packet analysis. The encryption used by SSH is intended to provide confidentiality and integrity of data over an unsecured network, such as the internet.

Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high-performance environments. Dlib's open-source licensing allows you to use it in any application, free of charge.

Flask-SocketIO gives Flask applications access to low latency bi-directional communications between the clients and the server. The client-side applications can use any of the SocketIO official client's libraries in JavaScript, C++, Java and Swift, or any compatible client to establish a permanent connection to the server.

2.1 Input and Output

Input for this application can be given in 2 ways, In the first method we should enable the image input command line and upload the required image and in the second method we should disable the image input command line and enable the video input command line.

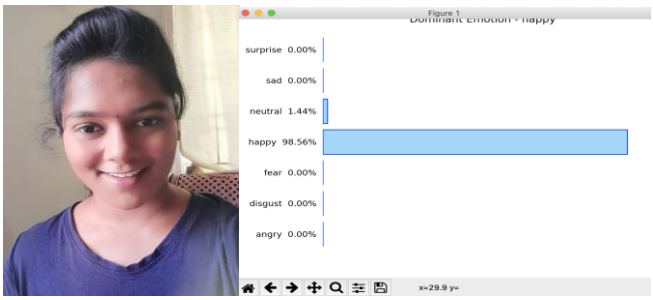


Fig 2.1 Input(uploaded Image) and Output

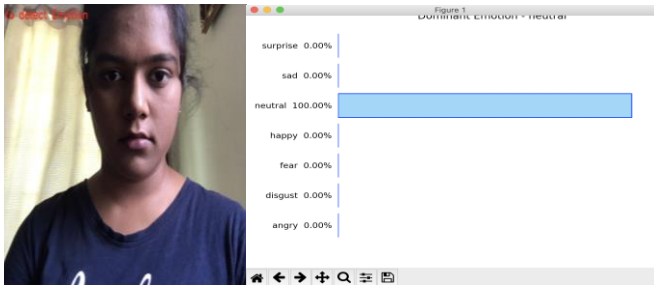
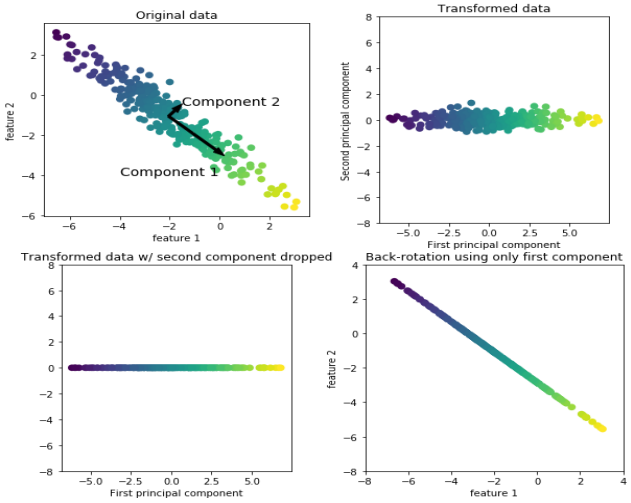


Fig 2.2 Input (Image captured from live camera) and Output

2.2 Performance Analysis

Y_hat = grid.predict(X_test)

Grid.plot_sample(target_names, y_hat)



3. CONCLUSION

By Performing CNN on model, we are able to get around ~84.2% of accuracy. To increase accuracy and capability of the project we have introduced LSTM (Long Short Term-Memory) which constantly monitors the performance of the system in prediction and performs addition and removal of dataset from training the model. This repeated process increases the efficiency and performance of the system with good results. This is similar to feedback loop which provides stability to the system.

REFERENCES

[1]. Dada EG, Bassi JS, Chiroma H, Abdulhamid SM, Adetunmbi AO, Ajibuwa OE. Machine Learning for email spam filtering: review, approaches and open research problems. Heliyon 2019; 5(6):e01802. **https://doi.org/10.1016/j.heli**

[2]. Xie M. Development of artificial intelligence and effects on the financial system. J Phys Conf 2019;1187:032084. **https://doi.org/10.1088/1742-6596/1187/3/032084.**

[3] Hegazy O, Soliman OS, Salam MA. A machine learning moel for stock market prediction. Int J Comput Sci Telecommun 2014;4(12):16-23.

[4] Beckmann JS, Lew D. Reconciling evidence-based medicine and precision medicine in the era of big data: challenges and opportunities. Genome Med 2016;8(1):134-9.

[5] Weber GM, Mandl KD, Kohane IS. Finding the missing link for big biomedical data. Jama 2014;311(24):2479-80.

[6] Loconcole C, Chiaaradia D, Bevilacqua V, Frisoli A. Real-time emotion recognition: an improved hybrid approach for classification performance. Intelligent Computing Theory 2014:320-31.

[7] Huang X, Kortelainen J, Zhao G, Li X, Moilanen A, Seppeanen T, Pietikainen M.Multimodal emotion analysis from facial expression and electroencephalogram. Comput Vis Image Understand 2016;147:114-24. **https://doi.org/10.1016/j.** cviu.2015.09.015.

[8] Raheel A, Majid M, Anwar SM. Facial Expression Recognition based on electroencephalography. In: 2019 2nd international conference on computing. Mathematics and engineering technologies (iCoMET), Sukkur, Pakisthan; 2019.

[9] Vassilis S, Herrmann Jürgen. Where do machine learning and human-computer interaction meet?. 1997.

[10] Keltner D, Ekman P. In: Lewis M, Haviland Jones JM, editors. Facial Expression of emotion, handbook of emotions. New York: Guilford Press; 2000. p. 236-49.

[11] Ekman P. Darwin and facial expression: a century ofresearch in review. United States of America: Academic Press Ishk; 2006.p.1973.

[12] Ekman P, Friesen WV. Constants across cultures in the face and emotion. J Pers Soc Psycchol 1971;17(2):124.

[13] EkmanP. Darwin and facial expresion: a century of research in review. United States of America: Academic Press Ishk; 2006.p. 1973.

[14] Ekman P, Friesen WV, Ancoli S. Facial signs of emotional experience. J Pers Soc Psychol 1980;39:1123-34.

[15] Nguyen BT, Trinh MH, Phan TV, Nguyen HD. An efficient real-time emotion detection using camera and facial landmarks. In: 2017 seventh international conference on information science and technology (ICIST); 2017. **https://doi.org/10.1109/icist.2017.7926765.**

[16]Lonconsole C, Miranda CR, Augusto G, Frisoli A, Orvalho V. Real-time emotion recognition novel method for geometrical facial features extraction. Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP) 2014:378-85.

[17] Palestra Giuseppe, Pettinicchio Adriaana, Del Coco marco, Ccarcagn Pierluigi, Leo Marco, Distante Ccosimo. Improved performance in facial expression recognition using 32 geometric features. In: Proceedings of the 18th international conference on image analysis and processing. ICIAP; 2015.p.518-28.

[18] Zhang J; Yin Z, Cheng P, Nichele S. Emotion recognition using multi-modal data and machine learning techniques: a tutorial and review. Information fusion. 2020.

[19] WilsonPI, Fernandez J. Facial feature detection using Haar classifiers. J. Comput. Small Coll., rocnik 21, c. 2006;4. s. 127-133, ISSN 1937-4771.

[20] Zhao G, Pietikainen M. Dynamic Texture Recognition Using Volume Local Binary Patterns. In: Vidal R, Heyden A, ma Y, editors. Dynamical Vision. WDV 2006, WDV 2005. Lecture Notes in Computer Science, vol. 4358. Berlin, Heidelberg: Springer; 2007