**Real Time Facial Emotion Recognition**

**Shashank Bhatia**

**Data science trainee,**

**AlmaBetter, Bangalore**

**Abstract:**

Deep learning is a very significant subset of machine learning because of its high performance across various domains. Convolutional Neural Network (CNN), is a powerful image processing deep learning type often used in computer vision that comprises an image and video recognition.

So my project is about using CNN to recognize facial emotions in real time for the students attending the class online using their webcam.

***Keywords: machine learning, Deep learning, CNN, AI, Emotion detection***

**1.Problem Statement**

The Indian education landscape has been undergoing rapid changes for the past 10 years owing to

the advancement of web-based learning services, specifically, eLearning platforms.

Global E-learning is estimated to witness an 8X over the next 5 years to reach USD 2B in 2021. India

is expected to grow with a CAGR of 44% crossing the 10M users mark in 2021. Although the market

is growing on a rapid scale, there are major challenges associated with digital learning when compared with brick and mortar classrooms. One of many challenges is how to ensure quality learning for students. Digital platforms might overpower physical classrooms in terms of content quality but when it comes to understanding whether students are able to grasp the content in a live class scenario is yet an open-end challenge.

In a physical classroom during a lecturing teacher can see the faces and assess the emotion of the class and tune their lecture accordingly, whether he is going fast or slow. He can identify students who need special attention. Digital classrooms are conducted via video telephony software program (exZoom) where it’s not possible for medium scale class (25-50) to see all students and access the mood. Because of this drawback, students are not focusing on content due to lack of surveillance.

While digital platforms have limitations in terms of physical surveillance but it comes with the power of data and machines which can work for you. It provides data in the form of video, audio, and texts which can be analysed using deep learning algorithms. Deep learning backed system not only solves the surveillance issue, but it also removes the human bias from the system, and all information is no longer in the teacher’s brain rather translated in numbers that can be analysed and tracked.

In this project, you are required to solve the above-mentioned challenge by applying deep learning algorithms to live video data. The solution to this problem is by recognizing facial emotions.

**2. Introduction**

Emotion recognition is the process of identifying human emotion. It is widely being used in multiple industries for market research, instead of filling long surveys the organizations are simply capturing the images of participants at different time intervals.

It is also being used to make AI more emotionally intelligent, another use case is to help blind people read facial expressions or to monitoring signs of attention while driving in an effort to enhance driver safety.

Our data consists of photos for following emotion classes –

* angry
* disgust
* fear
* happy
* neutral
* sad
* surprise

Our goal here is to train a model using these photographs to be able to predict the facial expression in real time using live stream video.

**3. Steps involved:**

* **Exploratory Data Analysis**

Next, we performed EDA which consisted of seeing the distribution of the data, which is to figure out the number of pictures available for each class and in total for train and test data set.

* **Data Augmentation**

Before we can fit data in any models we have to make certain changes to our data to ensure we get the best results.

When you don't have a large image dataset, it's a good practice to artificially introduce sample diversity by applying random, yet realistic, transformations to the training images, such as rotation and horizontal flipping. This helps expose the model to different aspects of the training data and reduce overfitting.

In this case we manually edit our photos so as to account for real life scenarios. We made the following random changes to our data before to prepare our data before analysis:

* rotation
* width shift
* height shift
* horizontal flip
* rescaling
* zoom
* Brightness
* **Model definition**

We defined our CNN model with 4 convolutional layers and 2 fully connected layers ending with an output layer with SoftMax activation function for final classification.

A CNN layer definition looks something like this –

1. **model.add(Conv2D(64,(5,5),padding = 'same',input\_shape = (48,48,3)))**

Here, 64 is the number of layers and 5 is height and width of the kernel will go through.

Padding is kept same so the image of size doesn’t decrease to prevent data loss.

Input shape is the the shape of the image this model will take.

1. **model.add(BatchNormalization())**

Layer that normalizes its inputs.

Batch normalization applies a transformation that maintains the mean output close to 0 and the output standard deviation close to 1. This has the effect of stabilizing the learning process and dramatically reducing the number of training epochs required to train deep networks.

1. **model.add(Activation(‘relu’))**

An activation function in a neural network defines how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network.

1. **model.add(MaxPooling2D(pool\_size = (2,2)))**

Max pooling selects the brighter pixels from the image. It is useful when the background of the image is dark and we are interested in only the lighter pixels of the image

It is used to reduce the dimensions of the feature maps. Thus, it reduces the number of parameters to learn and the amount of computation performed in the network.

1. **model.add(Dropout(0.35))**

Dropout randomly switches off some neurons in the network which forces the data to find new paths, thus reducing overfitting.

* **Model training and evaluation**

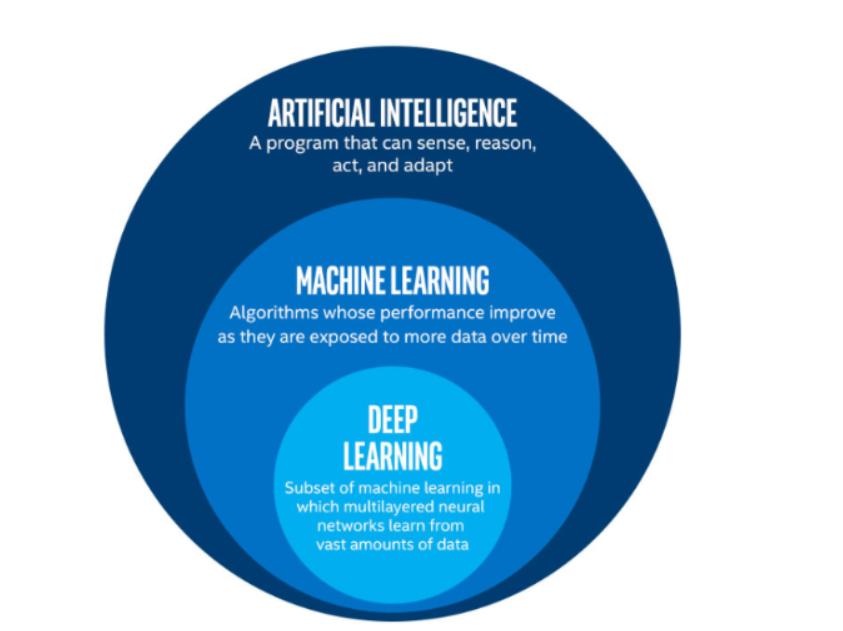
After creating our basic model, we fit the data in the models by first defining number of epochs, then we evaluate the model using evaluation set and save the best weights for our final application.

* **Packaging and deplyment**

Now we package our model according to industry standards and deploy the app on web based application using streamlit and also as end to end solution using AWS or Azure.

**4. Deep learning and CNN:**

* **Deep Learning**



Deep learning is a class of [machine learning](https://en.wikipedia.org/wiki/Machine_learning) [algorithms](https://en.wikipedia.org/wiki/Algorithm) that uses multiple layers to progressively extract higher-level features from the raw input. Most modern deep learning models are based on artificial neural networks, specifically convolutional neural networks. In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation.

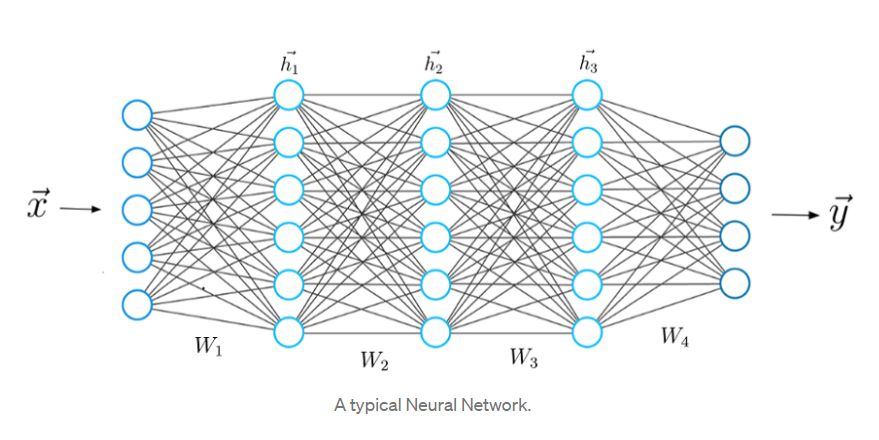
It basically that teaches computers to do what comes naturally to humans: learn by example.

Most deep learning methods use neural network architectures, which is why deep learning models are often referred to as deep neural networks.

The term “deep” usually refers to the number of hidden layers in the neural network. Traditional neural networks only contain 2-3 hidden layers, while deep networks can have as many as 150.

Deep learning models are trained by using large sets of labeled data and neural network architectures that learn features directly from the data without the need for manual feature extraction.

To achieve this, deep learning uses a multi-layered structure of algorithms called neural networks.

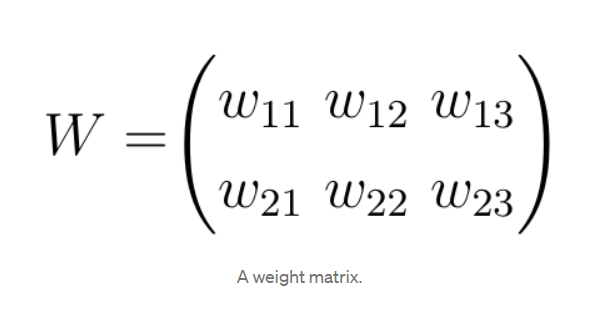


The design of the neural network is based on the structure of the human brain. Just as we use our brains to identify patterns and classify different types of information, neural networks can be taught to perform the same tasks on data.

The individual layers of neural networks can also be thought of as a sort of filter that works from gross to subtle, increasing the likelihood of detecting and outputting a correct result.

each connection between two neurons is represented by a different weight w. Each of these weight w has indices. The first value of the indices stands for the number of neurons in the layer from which the connection originates, the second value for the number of the neurons in the layer to which the connection leads.

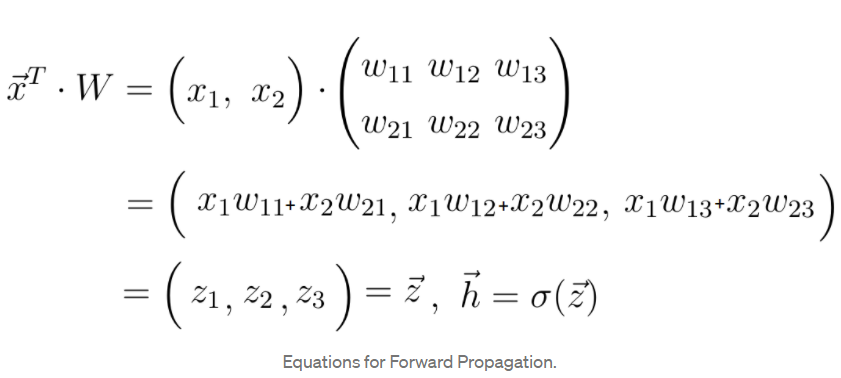
All weights between two neural network layers can be represented by a matrix called the weight matrix.



**Learning Process of a Neural Network**

For a given input feature vector **x**, the neural network calculates a prediction vector, which we call here as **h**. This step is also referred to as the **f**orward propagation. With the input vector **x** and the weight matrix **W** connecting the two neuron layers, we compute the dot product between the vector **x** and the matrix **W**.

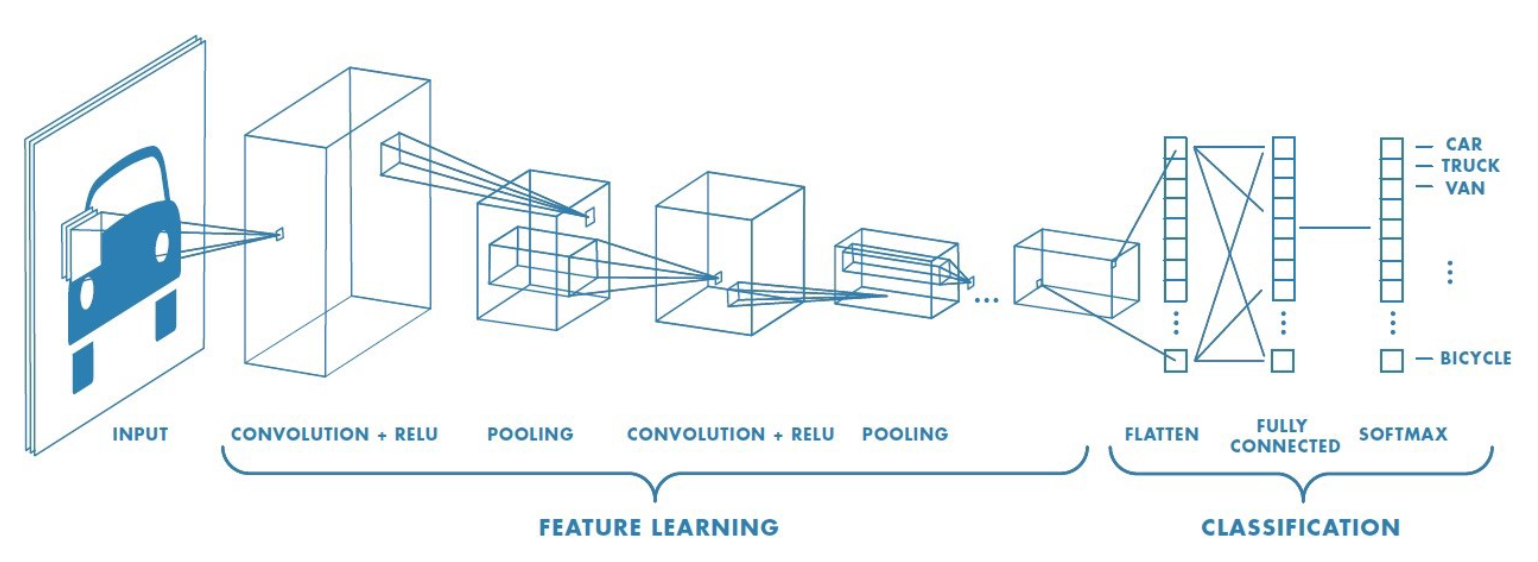
The result of this dot product is again a vector, which we call **z.** The final prediction vector **h** is obtained by applying a so-called activation function to the vector **z**. In this case, the activation function is represented by the letter **Sigma**. An activation function is only a nonlinear function that performs a nonlinear mapping from **z** to **h**.



**5. Deep learning and CNN:**

A convolutional neural network (CNN) is a type of artificial neural network used in image recognition and processing that is specifically designed to process pixel data.

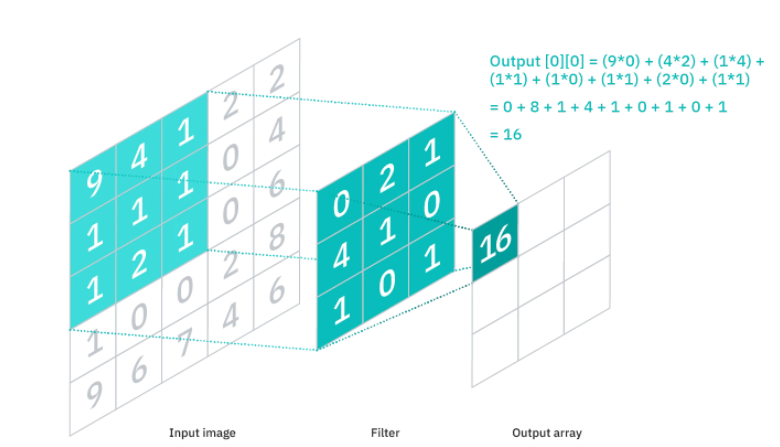
CNNs are powerful image processing, artificial intelligence (AI) that use deep learning to perform both generative and descriptive tasks, often using machine vison that includes image and video recognition, along with recommender systems and natural language processing (NLP).



CNN 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 from the other.

A CNN network consists of following components –

**The Kernel –** it is a feature detector, also known as a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.



**Pooling Layer -** also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input. Similar to the convolutional layer, the pooling operation sweeps a filter across the entire input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array.

**Fully Connected Layer –** also known as

Dense layer, refers to the layer whose inside neurons connect to every neuron in the preceding layer. The Fully-Connected layer is learning a possibly non-linear function in that space.

we flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training. Over a series of epochs, the model is able to distinguish between dominating and certain low-level features in images and classify them using the Softmax Classification technique.

**Activation Function –** it is the last component of the convolutional layer to increase the non-linearity in the output and is also known as transfer function.

t is used to determine the output of neural network like yes or no. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function).

The activation function of a node defines the output of that node given an input or set of inputs.

**6. Conclusion:**

Starting with EDA, we did image preprocessing, model training, model testing, and deployment on streamlit app as end-to-end solution.

* We achieved 67.3% accuracy on validation data.
* 5 out of 7 predictions were correct on test data.
* Model can be run on local machine and predictions can be done in real time.
* Model have been deployed on Streamlit app as a web app.

**References-**

1. towardsdatascience.com
2. machinelearningmastery.com
3. Analytics Vidhya
4. Almabetter
5. medium.com