Functions

Activation Functions:

Activation functions helps to determine the output of a neural network. These type of functions are attached to each neuron in the network, and determines whether it should be activated or not, based on whether each neuron's input is relevant for the model's prediction.

They introduce non-linear properties to our network which have a degree more than one, which can help the network learn complex data, compute and learn almost any function representing a question, and provide accurate predictions.

Sigmod function: It is a activation function of form f(x) = 1 / 1 + exp(-x) . Its Range is between 0 and

1. It is an S-shaped curve. It is easy to understand.

Adv: Smooth gradient, Output values bound b/w 0 and 1, clear predictions, i.e very close to 1 or 0.

Dis Adv: Prone to gradient vanishing, Function output is not zero-centered, Power operations are relatively time consuming

tanh function: hyperbolic tangent function

mathematical formula is f(x) = 1— exp(-2x) / 1 + exp(-2x). Now it's the output is zero centred because its range in between -1 to 1 i.e. -1 < output < 1 . Hence optimisation is easier in this method; Hence in practice, it is always preferred over Sigmoid function.

RELU Function: It has become more popular in the past couple of years. It was recently proved that it has six times improvement in convergence from Tanh function. It's R(x) = max (0,x) i.e. if x < 0 , R(x) = 0 and if x >= 0 , R(x) = x.

Adv: When the input is positive, there is no gradient saturation problem. calculation speed faster.

Functions (cont)

Dis adv: When the input is negative, ReLU is completely inactive, which means that once a negative number is entered, ReLU will die. In this way, in the forward propagation process, it is not a problem. Some areas are sensitive and some are insensitive. But in the backpropagation process, if you enter a negative number, the gradient will be completely zero, which has the same problem as the sigmod function and tanh function.

2) We find that the output of the ReLU function is either 0 or a positive number, which means that the ReLU function is not a 0-centric function.

Leaky RELU Function: In order to solve the Dead ReLU Problem, people proposed to set the first half of ReLU 0.01x instead of 0.

ELU (Exponential Linear Units) function: adv same as relu and no dead cell issue.

Softmax:

PRelu

Swish

Maxout

Soft Plus

Optimiziers

Optimizers are algorithms or methods used to change the attributes of the neural network such as weights and learning rate to reduce the losses. Optimizers are used to solve optimization problems by minimizing the function.

What is GD: it is an iterative machine learning optimisation algorithm to reduce the cost function, and help models to make accurate predic-tions.

Batch gradient descent: In the batch gradient, we use the entire dataset to compute the gradient of the cost function for each iteration for gradient descent and then update the weights.

Optimiziers (cont)

SGD (Stochastic gradient descent): Stochastic gradient descent, we use a single data point or example to calculate the gradient and update the weights with every iteration.
Mini-batch gradient descent: Mini-batch gradients is a variation of stochastic gradient descent where
instead of a single training example, a mini-batch of samples are used. Mini -batch gradient descent is widely used and converges faster and is more stable.
As we take batches with different samples, it reduces the noise which is a variance of the weight updates.
Momentum: One disadvantage of the SGD method is that its update direction depends entirely on the current batch, so its update is very unstable. A simple way to solve this problem is to introduce momentum.
Momentum is momentum, which simulates the inertia of an object when it is moving, that is, the direction of the previous update is retained to a certain extent during the update, while the current update gradient is used to fine-tune the final update direction. In this way, you can increase the stability to a certain extent, so that you can learn faster, and also have the ability to get rid of local optimization.
Adagrad: Adagrad is an algorithm for gradient-–based optimization which adapts the learning rate to the parameters, using low learning rates for parameters associated with frequently occurring features, and using high learning rates for parameters associated with infrequent features.
Adadelta: Adadelta is an extension of Adagrad and it also tries to reduce Adagrad’s aggressive, monotonically reducing the learning rate.**
RMSProp:
Adam:Adaptive Moment Estimation (Adam).

Optimiziers (cont)

Adam can be viewed as a combination of Adagrad and RMSprop,(Adagrad) which works well on sparse gradients and (RMSProp) which works well in online and nonstationary settings repectively.
Adam implements the exponential moving average of the gradients to scale the learning rate instead of a simple average as in Adagrad. It keeps an exponentially decaying average of past gradients.
Adam is computationally efficient and has very less memory requirement.
Adam optimizer is one of the most popular and famous gradient descent optimization algorithms.

Loss Functions

Loss functions are mainly used to minimize the error
L1 Loss function: It is used to minimize the error which is the sum of all the absolute differences in between the true value and the predicted value.

    L1 = sum of i=1, n |ytrue - ypredicted|

L2 Loss Function: It is used to minimize the error which is the sum of all the squared differences in between the true value and the pedicted value.

    L1 = sum of i=1, n (ytrue - ypredicted)2

Huber Loss:Huber Loss is often used in regression problems. Compared with L2 loss, Huber Loss is less sensitive to outliers(because if the residual is too large, it is a piecewise function, loss is a linear function of the residual).
Hinge Loss: Hinge loss is often used for binary classification problems, such as ground true: t = 1 or -1, predicted value y = wx + b
Cross-entropy loss: It is used to define a loss function in machine learning and optimization. Also called the log loss, measures the performance of the classification model whose output is a probability value between 0 and 1.
Sigmoid-Cross-entropy loss
Softmax-Cross-entropy loss

CNN

Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification.

Different Types of Layers in CNN:

1. Input Layer: Holds the raw input of image with width(32), height(32) and depth(3)

2. Convolution Layer: It computes the output volume by computing dot products between all filters and image patches

3. Activation Function Layer: This layer will apply the element-wise activation function to the output of the convolution layer.

4.Pool Layer: This layer is periodically inserted within the covnets, and its main function is to reduce the size of volume which makes thecomputation fast reduces memory and also prevents overfitting. Two common types of pooling layers are max pooling and average pooling.

5. Fully Connected Layer: This layer is a regular neural network layer that takes input from the previous layer and computes the class scores and outputs the 1-D array of size equal to the number of classes.

Pooling, padding, filtering operations on CNN

Pooling: It is a down-sampling operation that is typically applied after a convolutional layer, which does some sort of spatial invariance, to reduce the spatial dimensions of the CNN.

It creates a pooled feature map sliding a filter matrix over the input matrix. In particular, max and average pooling are special kinds of pooling where max and average values are taken, respectively.

Pooling layers are 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.

CNN (cont)

The pooling layer summarises the features present in a region of the feature map generated by a convolution layer. So, further operations are performed on summarised features instead of precisely positioned features generated by the convolution layer. This makes the model more robust to variations in the position of the features in the input image.

Padding: Padding is simply a process of adding layers of zeros to our input images so as to avoid the problems like image shrinking every time a convolution operation is performed and also the pixels on the corners and the edges are used much less than those in the middle

Valid Padding : It implies no padding at all. The input image is left in its valid/unaltered shape.

Same Padding : In this case, we add 'p' padding layers such that the output image has the same dimensions as the input image.

n+2p-f/s+1 (n=size, p=padding, f=filter size, s=stride) (6, 1, 3, 1) = 6

n-f/s+1 (5, 3, 1) = 3 - with 1 stride

(n-f)/s+1 (5, 3, 2) = 2 - 2 stride

CNN Architectures

LeNet: It is a very efficient 7-level convolutional neural network for handwritten character recognition. (32*32 pixel grayscale image, tanh activation function and softmax at last FC layer)

Image -> Convolution(5*5) -> *Average Pooling(2*2)* -> Conv(5*5*) -> *Average P(2*2) -> Conv(5*5) -> FC -> FC

AlexNet: winner of the 2012 ImageNet competition

Inception: Also known as GoogLeNet , it is a 22-layer network. There are four parallel channels in each inception module , and concat is performed at the end of the channel .

Imagenet: It has 1,000 image categories represent object classes that we encounter in our day-to-day lives, such as species of dogs, cats, various household objects, vehicle types, and much more.

CNN Architectures (cont)

ResNet: It also called as Residual Neural Network (ResNet). This architecture introduced a concept called “skip connections”. Typically, the input matrix calculates in two linear transformations with ReLU activation function. In Residual network, it directly copies the input matrix to the second transformation output and sums the output in final ReLU function.

VGG: VGG-16 is a simpler architecture model since it’s not using many hyperparameters. It always uses 3 x 3 filters with the stride of 1 in convolution layer and uses SAME padding in pooling layers 2 x 2 with a stride of 2.

Three fully connected layers follow the VGG convolutional layers. The width of the networks starts at the small value of 64 and increases by a factor of 2 after every sub-sampling/pooling layer. It achieves the top-5 accuracy of 92.3 % on ImageNet.

Object Detections

HAAR Cascade: It is the machine learning object detections algorithm used to identify the objects in
an image or the video and based on the concept of features.

It has 4 stages: Haar Feature Selection, Creating Integral Images, Adaboost Training, Cascading Classifiers

It is well known for being able to detect faces and body parts in an image but can be trained to identify almost any object.

RCNN: To bypass the problem of selecting the huge number of regions, Ross Girshick et al. proposed a method where we use the selective search to extract just 2000 regions from the image, and he called them as region proposals. Therefore, instead of trying to classify the huge number of regions, you can work with 2000 regions.

Problems with R-CNN:

It still takes the huge amount of time to train the network as we would have to classify 2000 region proposals per image.

It cannot be implemented real-time as it takes around 47 seconds for each test image.

The selective search algorithm is the fixed algorithm. Therefore, no learning is happening at that stage. This leads to the generation of the bad candidate region proposals.

Faster RCNN: It has two networks: region proposal network (RPN) for generating region proposals and a network using these proposals to detect objects. The main difference here with the Fast R-CNN is that the later uses selective search to generate the region proposals. The time cost of generating the region proposals is much smaller in the RPN than selective search, when RPN shares the most computation with object detection network. In brief, RPN ranks region boxes (called anchors) and proposes the ones most likely containing objects.

Anchors play an very important role in Faster R-CNN. An anchor is the box. In default configuration of Faster R-CNN, there are nine anchors at the position of an image.

The output of the region proposal network is the bunch of boxes/proposals that will be examined by a classifier and regressor to check the occurrence of objects eventually. To be more precise, RPN predicts the possibility of an anchor being background or foreground, and refine the anchor

DarkNet: DarkNet is a framework used to train neural networks; it is open source and written in C/CUDA and serves as the basis for YOLO. Darknet is also used as the framework for training YOLO, meaning it sets the architecture of the network. Clone the repo locally, and you have it. To compile it, run a make. But first, if you intend to use the GPU capability, you need to edit the Makefile in the first two lines, where you tell it to compile for GPU usage with CUDA drivers.

YOLO: You look only once

Object Detections (cont)

YOLO is a network “inspired by” Google Net. It has 24 convolutional layers working as the feature extractors and two dense layers for making the predictions. The architecture works upon is called Darknet, a neural network framework created by the first author of the YOLO paper.

Core Concept for YOLO: The algorithm works off by dividing the image into the grid of the cells, for each cell bounding boxes and their scores are predicted, alongside class probabilities. The confidence is given in terms of IOU (intersection over union), metric, which is measuring how much the detected object overlaps with the ground truth as a fraction of the total area spanned by the two together (the union).

Yolo V3: 53 Convolutional Layers

Mask R-CNN architecture:Mask R-CNN was proposed by Kaiming He et al. in 2017. It is very similar to Faster R-CNN except there is another layer to predict segmented. The stage of region proposal generation is same in both the architecture the second stage which works in parallel predict class, generate bounding box as well as outputs a binary mask for each RoI.

Applications :

Due to its additional capability to generate segmented mask, it is used in many computer vision applications such as: Human Pose Estimation Self Driving Car Drone Image Mapping etc.