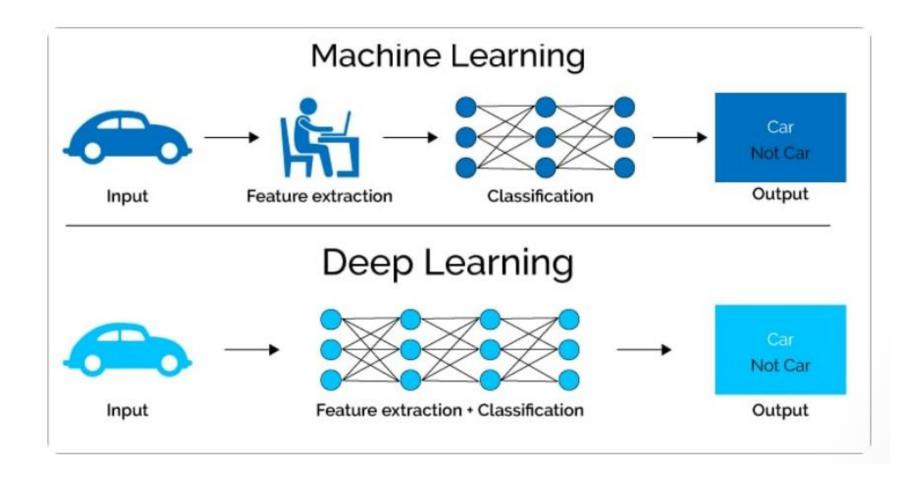


# Artificial Intelligence

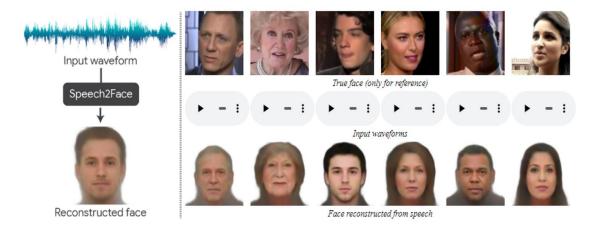
Intro. to Deep Learning

#### ML vs DL



## Applications







#### **Applications**



Language
Translation with
Deep Learning and the
Magic of Sequences

The bottleneck is no longer access to information; now it's our ability to keep up.

All can be trained on a variety of different types of texts and summary lengths.

A model that can generate long, coherent, and meaningful summaries remains an open research problem.

The last few decades have witnessed a fundamental change in the challenge of taking in new information. The bottleneck is no longer access to information now its our ability to keep up. We all have to read more and more to keep up-to-date with our jobs, the news, and social media. We've looked at how AI can improve people's work by helping with this information deluge and one potential answer is to have algorithms automatically summarize longer texts. Training a model that can generate long, coheren, and meaningful summaries remains an open research problem. In fact, generating any kind of longer text is hard for even the most advanced deep learning algorithms. In order to make summarization successful, we introduce two separate improvements: a more contextual word generation model and a new way of training summarization models via reinforment learning (RL). The combination of the two training methods enables the system to create relevant and highly readable multi-sentence summaries of long text, such as news articles, significantly improving on previous results. Our algorithm can be trained on a variety of different types of texts and summary lengths. In this blog post, we present the main contributions of our model and an overview of the natural language challenges specific to text summarization.

Salesforce created an algorithm that automatically summarizes text using machine learning

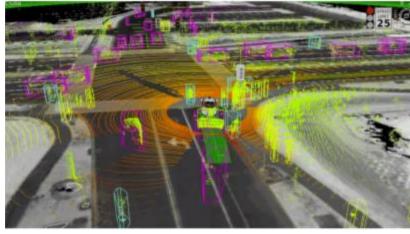
**Text Generation/Summarization (NLP)** 

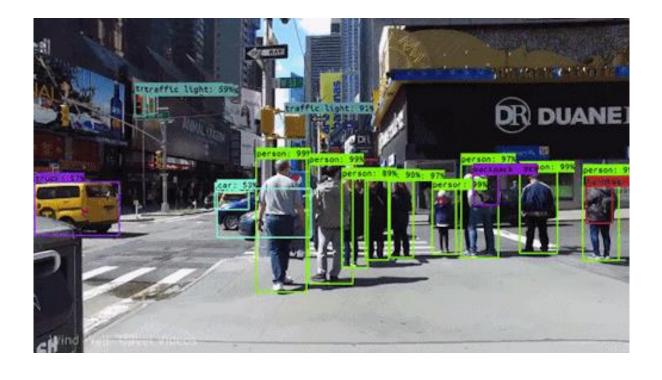


**Recommend Movies** 

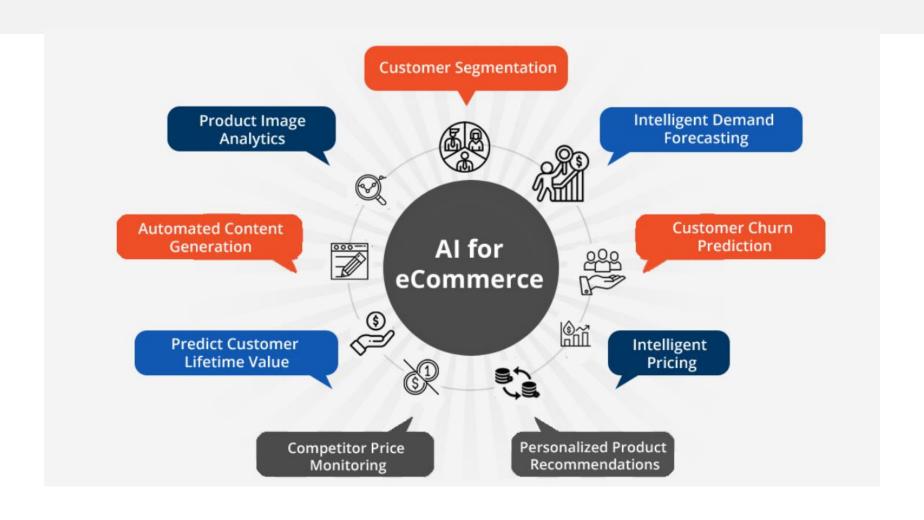
### Applications: Self Driving Cars







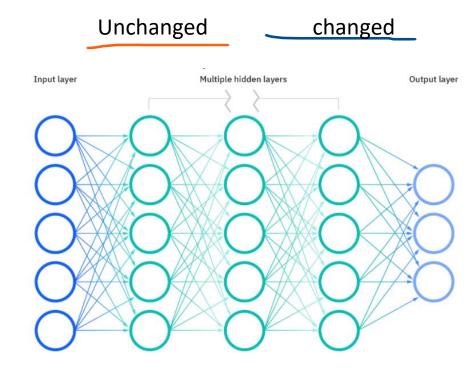
### Applications: Ecommerce Websites



#### Deep Neural Network

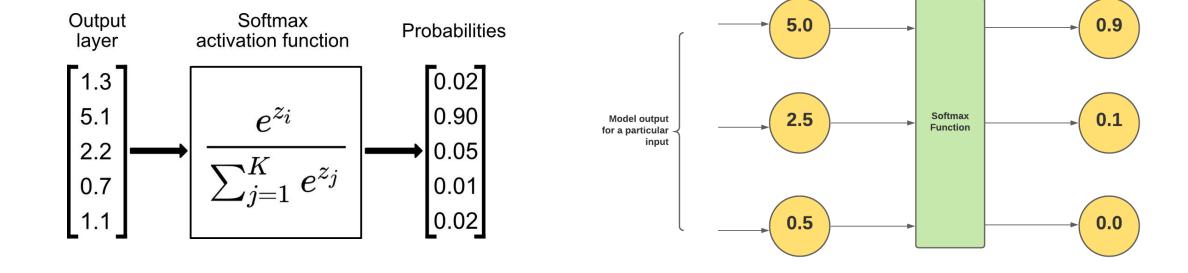
- Any network i.e. Feed Forward or recurrent network with increasing number of hidden layer (>2)
- called Deep Neural Network

- Vanishing Gradient Problem:
  - ReLU is an activation function that is known as to better transmit error than Sigmoid

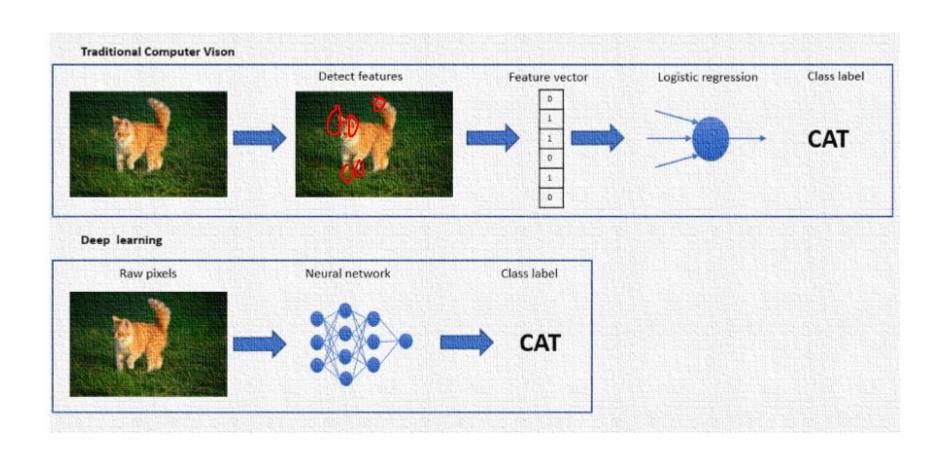


#### SoftMax

The softmax function is a function that turns a vector of K real values into a vector of K real values that sum to 1. The input values can be positive, negative, zero, or greater than one, but the softmax transforms them into values between 0 and 1, so that they can be interpreted as probabilities.

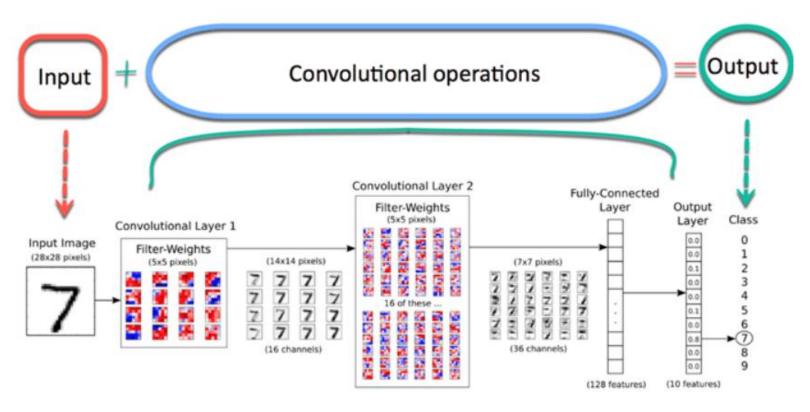


### Computer Vision



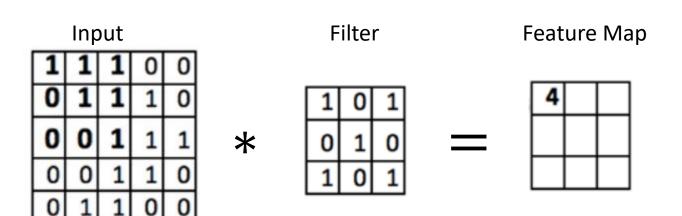
#### **CNN**

• Convolutional Neural Network(CNN), are multi-layer neural networks that assume input data to be image.

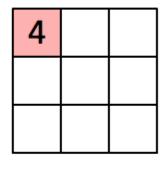


### Convolutional Layer

- The purpose of the convolution is to extract the features of the object on the image locally. It means the network will learn specific patterns within the picture and will be able to recognize it everywhere in the picture.
- Image is basically 2D Matrix:



<b>1</b> <sub>×1</sub>	1,0	1,	0	0
0,0	1,	1,0	1	0
<b>0</b> <sub>×1</sub>	0,×0	1,	1	1
0	0	1	1	0
0	1	1	0	0

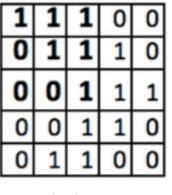


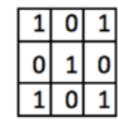
Image

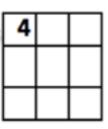
Convolved Feature

#### Convolutional Layer: Output

• Out = 
$$\frac{n - f + 2 \cdot P}{s} + 1$$
 = 2 + 1 = 3 where s = 1, p = 0







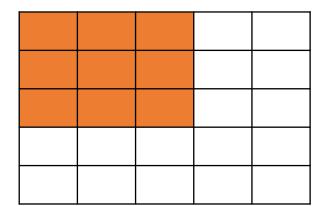
nxn

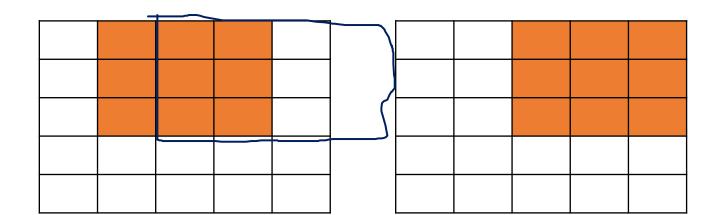
fxf

out

### Convolutional Layer: Stride (s)

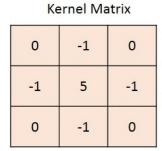
- The filter is moved across the image left to right, top to bottom, with a one-pixel column change on the horizontal movements, then a onepixel row change on the vertical movements.
- The amount of movement between applications of the filter to the input image is referred to as the stride, and it is almost always symmetrical in height and width dimensions.
- Default stride is 1





### Convolutional Layer: Stride (s)

0	0	0	0	0	0	
0	105	102	100	97	96	
0	103	99	103	101	102	P
0	101	98	104	102	100	
0	99	101	106	104	99	7
0	104	104	104	100	98	



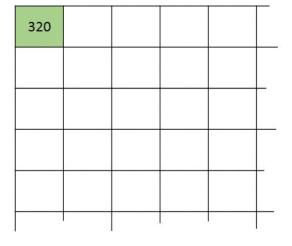


Image Matrix

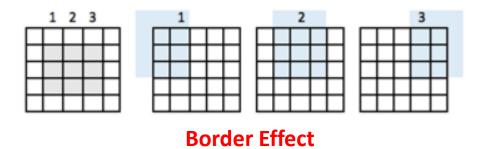
$$0*0+0*-1+0*0 +0*-1+105*5+102*-1 +0*0+103*-1+99*0 = 320$$

Output Matrix

Convolution with horizontal and vertical strides = 1

### Convolutional Layer: Padding (p)

• By default, a filter starts at the left of the image with the left-hand side of the filter sitting on the far left pixels of the image. The filter is then stepped across the image one column at a time until the right-hand side of the filter is sitting on the far right pixels of the image.



 To get the same output dimension as the input dimension, you need to add padding. Padding consists of adding the right number of rows and columns on each side of the matrix. It will allow the convolution to center fit every input tile.

#### RELU (Non-Linear Activation Function)

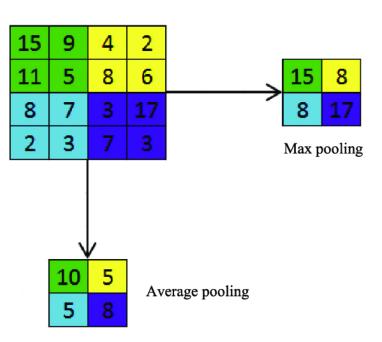
- Non Linearity (ReLU)
- At the end of the convolution operation, the output is subject to an activation function to allow non-linearity. The usual activation function for convnet is the Relu. All the pixel with a negative value will be replaced by zero.

#### Vanishing Gradient Problem:

ReLU is an activation function that is known as to better transmit error than Sigmoid

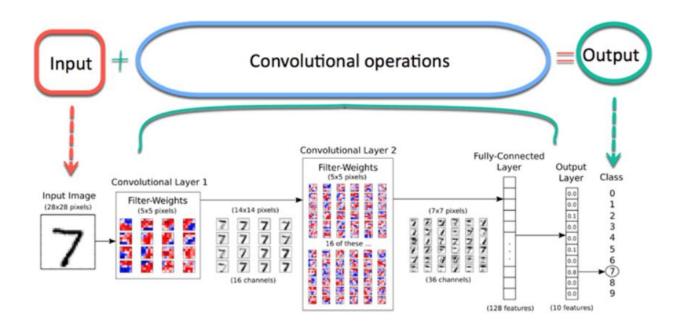
#### Pooling Layer

- The purpose of the pooling is to reduce the dimensionality of the input image. The steps are done to reduce the computational complexity of the operation. By diminishing the dimensionality, the network has lower weights to compute, so it prevents overfitting.
  - Max Pooling
  - Average Pooling



#### Fully Connected (Dense) Layers

- The feature map has to be flatten before to be connected with the dense layer.
- Dense Layers are Artificial Neural Network



#### Other Material

 https://www.deeplearningwizard.com/deep\_learning/practical\_pytor ch/pytorch\_convolutional\_neuralnetwork/