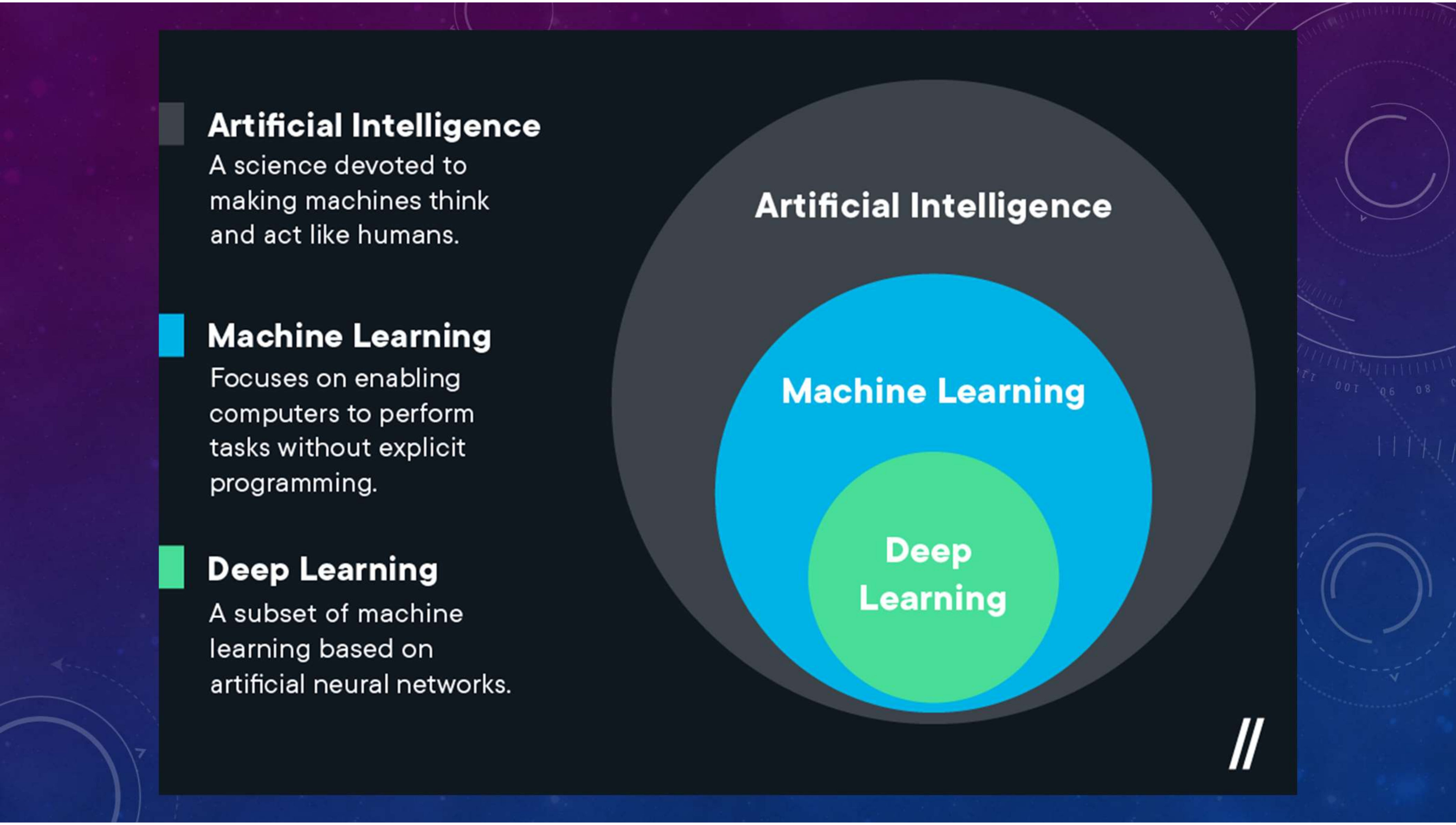
INTRODUCTION TO DEEP LEARNING

WHAT IS DEEP LEARNING?

- Deep learning is a branch of machine learning that uses artificial neural networks to help the machine to think about and respond to a particular problem.


Artificial Intelligence
A science devoted to making machines think and act like humans.
Machine Learning
Focuses on enabling computers to perform tasks without explicit programming.
Deep Learning
A subset of machine learning based on artificial neural networks.
Artificial Intelligence
Machine Learning
Deep Learning

RELATIONSHIP BETWEEN DEEP LEARNING AND MACHINE LEARNING

- The important thing to remember here is that, although it is very tempting to think of deep learning as an independent area under artificial intelligence, it is not.
- Deep learning is very much a part of artificial intelligence and is a subset of machine learning.

HISTORY OF DEEP LEARNING

- Walter Pitts was a logician in computational neuroscience, while Warren McCulloch was a neurophysiologist and cybernetician.
- In the year 1943, they created a computer model that was inspired by the neural networks present in the human brain.
- They developed something called threshold logic, which was a combination of mathematics and algorithms that compared the total input with a certain threshold.
- This enabled them to recreate the process of thinking, just as it happens in the brain.
- This was a breakthrough that led to many more deep learning innovations.

DEEP LEARNING USE CASES

OIL & GAS INDUSTRY

Baker Hughes, a GE company (BHGE), is using AI to help the oil and gas industry distill data in real time in order to significantly reduce the cost of locating, extracting, processing, and delivering oil. Deep learning neural networks are used to unseal insights from data that were previously hidden in order to achieve important goals such as seismic modeling, automated well planning, predicting machinery failure, and optimizing supply chains.

Deep learning algorithms allow oil and gas companies to determine the best way to optimize their operations as conditions continue to change. For instance, they can turn large volumes of seismic data images into 3-dimensional maps designed to improve the accuracy of reservoir predictions. Specifically, they can use deep learning to train models to predict and improve the efficiency, reliability, and safety of expensive drilling and production operations.

Source: medium.com

CONSTRUCTION INDUSTRY

Construction company Bechtel Corp. has a deep learning use case which is aimed at optimizing construction planning. The company is using reinforcement learning models similar to those used by AlphaGo (developed by Alphabet's Google DeepMind), the software that defeated elite human players of the game Go, to find the fastest route to build projects. The model runs step-by-step simulations of projects, testing out sequences of installing pipe laying concrete to find the optimal sequence.

Bechtel is just starting to explore the huge potential for bringing deep learning use cases to the construction industry. As with other industries, the goal is to take the company's industry knowledge and align it with deep learning to advance the industry forward.

Source: medium.com

FINANCIAL SERVICES INDUSTRY

There are many opportunities for applying deep learning technology in the financial services industry. One important task that deep learning can perform is e-discovery.

For example, large investment houses like JPMorgan Chase are using deep learning-based text analytics for insider trading detection and government regulatory compliance. Hedge funds use text analytics to drill down into massive document repositories for obtaining insights into future investment performance and market sentiment. The use case for deep learning-based text analytics revolves around its ability to parse massive amounts of text data to perform analytics or yield aggregations.

Source: medium.com

CYBERSECURITY

Deep learning also has several use cases in the cybersecurity space. One of the advantages that deep learning has over other approaches is accuracy. In many cases, the improvement approaches a 99.9% detection rate. The high risk and cost associated with not detecting a security threat make the expense related with deep learning justified.

Deep learning can play a number of important roles within a cybersecurity strategy. Use cases include automating intrusion detection with an exceptional discovery rate. Deep learning also performs well with malware, as well as malicious URL and code detection. Deep learning for cybersecurity is a motivating blend of practical applications along with untapped potential. With proper vetting, it’s well worth the effort to ensure the time and investment required for implementing a solution that yields the anticipated gains.

Source: medium.com

SOCIAL MEDIA

Deep learning's power can also be seen with how it's being used in social media technology.

Let's take Pinterest for example, which includes a visual search tool that lets you zoom in on a specific object in a "Pin" (or pinned image) and discover visually similar objects, colors, patterns and more. The company's engineering team used deep learning to teach their system how to recognize image features using a richly annotated data set of billions of Pins curated by Pinterest users. The features can then be used to compute a similarity score between any two images and identify the best matches.

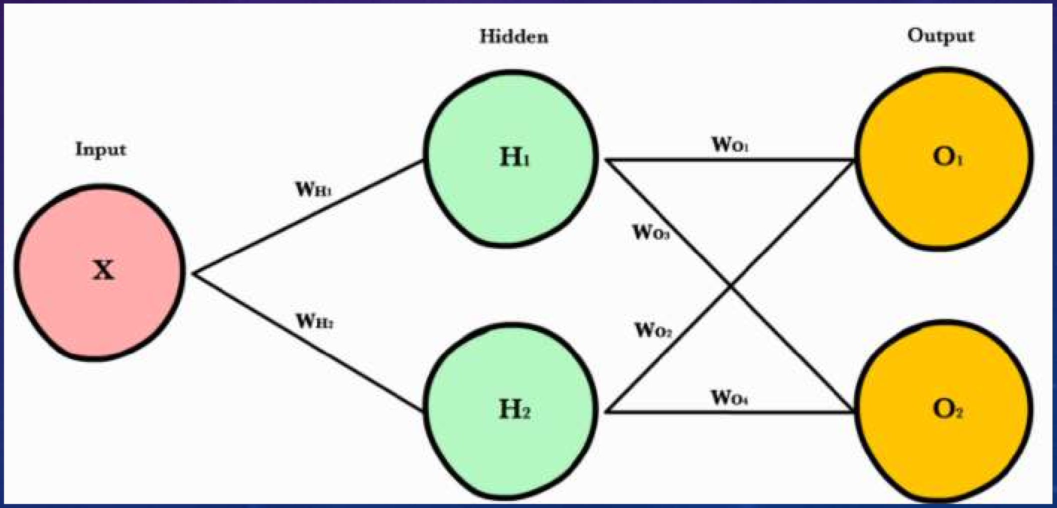
Source: medium.com

CONCEPTS AND TERMS

HTTPS://ML-CHEATSHEET.READTHEDOCS.IO/EN/LATEST/NN_CONCEPTS.HTML

NEURAL NETWORK

Neural networks are a class of machine learning algorithms used to model complex patterns in datasets using multiple hidden layers and non-linear activation functions. A neural network takes an input, passes it through multiple layers of hidden neurons (mini-functions with unique coefficients that must be learned), and outputs a prediction representing the combined input of all the neurons.

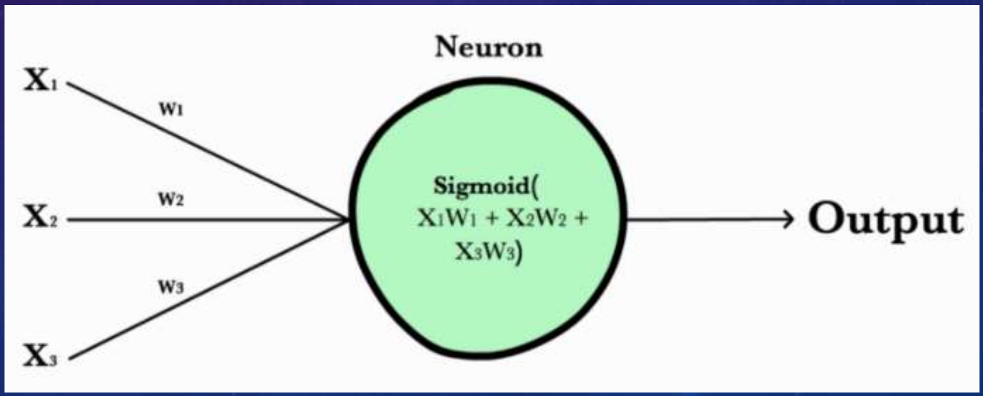


NEURAL NETWORK

- Neural networks are trained iteratively using optimization techniques like gradient descent.
- After each cycle of training, an error metric is calculated based on the difference between prediction and target.
- The derivatives of this error metric are calculated and propagated back through the network using a technique called backpropagation.
- Each neuron's coefficients (weights) are then adjusted relative to how much they contributed to the total error.
- This process is repeated iteratively until the network error drops below an acceptable threshold.

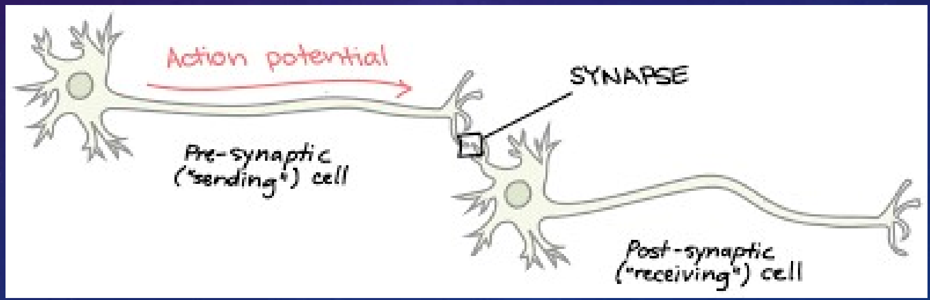
NEURON

- A neuron takes a group of weighted inputs, applies an activation function, and returns an output.
- Inputs to a neuron can either be features from a training set or outputs from a previous layer's neurons. Weights are applied to the inputs as they travel along synapses to reach the neuron. The neuron then applies an activation function to the "sum of weighted inputs" from each incoming synapse and passes the result on to all the neurons in the next layer.



SYNAPSE

- Synapses are like roads in a neural network. They connect inputs to neurons, neurons to neurons, and neurons to outputs. In order to get from one neuron to another, you have to travel along the synapse paying the “toll” (weight) along the way. Each connection between two neurons has a unique synapse with a unique weight attached to it. When we talk about updating weights in a network, we’re really talking about adjusting the weights on these synapses.



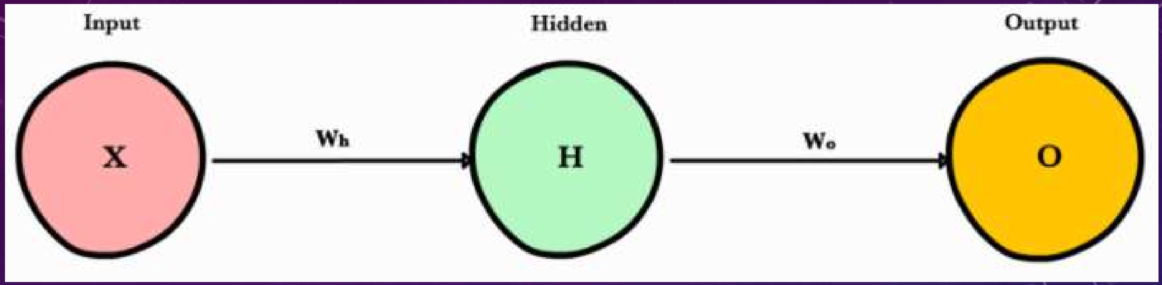
WEIGHTS

- Weights are values that control the strength of the connection between two neurons.
- That is, inputs are typically multiplied by weights, and that defines how much influence the input will have on the output.
- In other words: when the inputs are transmitted between neurons, the weights are applied to the inputs along with an additional value (the bias).

BIAS

- Bias terms are additional constants attached to neurons and added to the weighted input before the activation function is applied.
- Bias terms help models represent patterns that do not necessarily pass through the origin.
- For example, if all your features were 0, would your output also be zero? Is it possible there is some base value upon which your features have an effect?
- Bias terms typically accompany weights and must also be learned by your model.

LAYERS



- Input Layer
  - Holds the data your model will train on. Each neuron in the input layer represents a unique attribute in your dataset (e.g., height, hair color, etc.).
- Hidden Layer
  - Sits between the input and output layers and applies an activation function before passing on the results. There are often multiple hidden layers in a network. In traditional networks, hidden layers are typically fully-connected layers—each neuron receives input from all the previous layer's neurons and sends its output to every neuron in the next layer. This contrasts with how convolutional layers work where the neurons send their output to only some of the neurons in the next layer.
- Output Layer
  - The final layer in a network. It receives input from the previous hidden layer, optionally applies an activation function, and returns an output representing your model's prediction.

WEIGHTED INPUT

A neuron's input equals the sum of weighted outputs from all neurons in the previous layer. Each input is multiplied by the weight associated with the synapse connecting the input to the current neuron. If there are 3 inputs or neurons in the previous layer, each neuron in the current layer will have 3 distinct weights—one for each each synapse.

Single Input

$$\begin{aligned} Z &= Input \cdot Weight \\ &= XW \end{aligned}$$

Multiple Inputs

$$\begin{aligned} Z &= \sum_{i=1}^{n} x_i w_i \\ &= x_1 w_1 + x_2 w_2 + x_3 w_3 \end{aligned}$$

ACTIVATION FUNCTIONS

- Activation functions live inside neural network layers and modify the data they receive before passing it to the next layer.
- Activation functions give neural networks their power—allowing them to model complex non-linear relationships.
- By modifying inputs with non-linear functions neural networks can model highly complex relationships between features.
- Popular activation functions include relu and sigmoid.

LOSS FUNCTIONS

- A loss function, or cost function, is a wrapper around our model's predict function that tells us "how good" the model is at making predictions for a given set of parameters.
- The loss function has its own curve and its own derivatives.
- The slope of this curve tells us how to change our parameters to make the model more accurate!
- We use the model to make predictions. We use the cost function to update our parameters. Our cost function can take a variety of forms as there are many different cost functions available.
- Popular loss functions include: MSE (L2) and Cross-entropy Loss.

GRADIENT ACCUMULATION

- Gradient accumulation is a mechanism to split the batch of samples—used for training a neural network—into several mini-batches of samples that will be run sequentially.
- This is used to enable using large batch sizes that require more GPU memory than available. Gradient accumulation helps in doing so by using mini-batches that require an amount of GPU memory that can be satisfied.

HOW TO IMPLEMENT DEEP LEARNING?

1. Identify Your Problems
    - Before delving straight into applying deep learning, pinpoint the problems you're trying to solve. Creating a model with vague goals will not produce the outcomes you desire, and furthermore, not every business or organization is ready for deep learning.
2. Pick a tool & build a strategy
3. Assemble Your Data Sets
4. Build Your Model
5. Optimise, Test & Deploy Your Models

THE END