Computational Intelligence

Subject5: Deep Neural Networks

Instructor: Ali Tourani

A.Tourani1991@gmail.Com

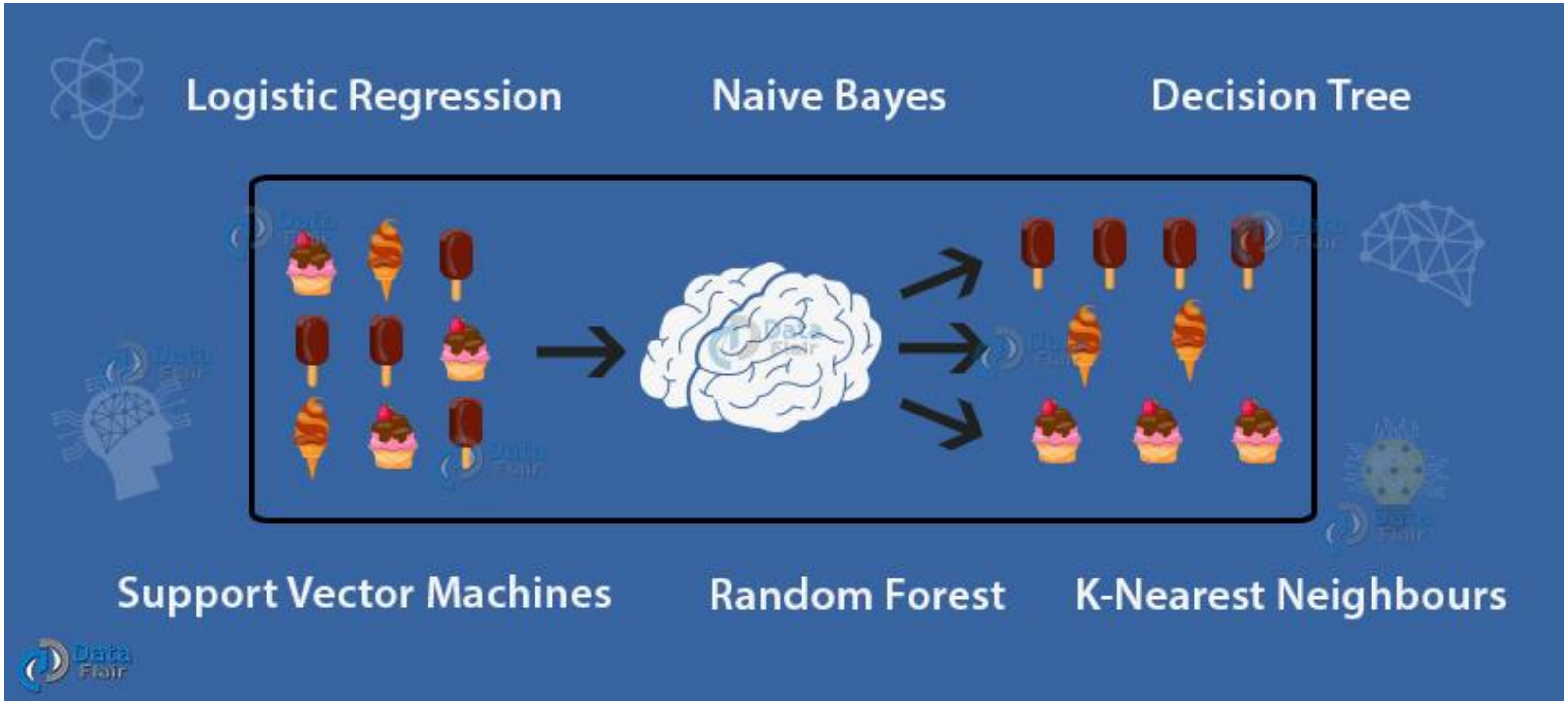
University of Guilan

Agenda

- Machine Learning Classification Algorithms
- Deep Neural Networks
- DNNs applications
- Advanced architectures
- Fine-tuning Deep Learning Models

Machine Learning Classification Algorithms

Image from data-flair.training website

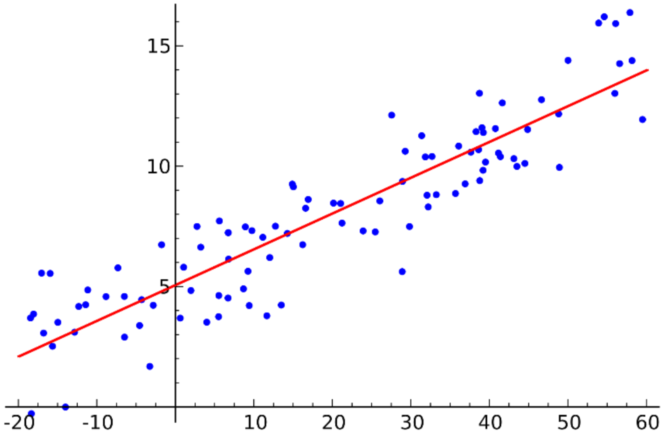


Machine Learning Classification Algorithms

University of Guilan

Common classification algorithms

- Logistic Regression
  - A binary classifier with simple implementation and usage
  - Effective when the set of input variables is well-known



Machine Learning Classification Algorithms

Common classification algorithms

- K-Nearest Neighbor (KNN)
  - A Multiclass classifier dependent to all training instances for classification
  - Classifies each data point by analyzing its nearest neighbors from the training set

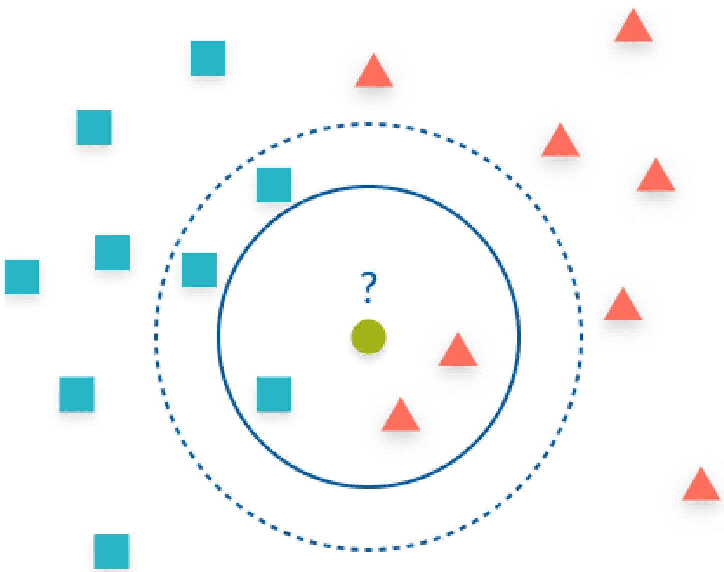


Image from missinglink.ai website

Machine Learning Classification Algorithms

Common classification algorithms

- Decision Tree Algorithm
  - A Multiclass classifier with the ability to model complex decision processes
  - A tree structure with a set of "if-then" rules learned in training process

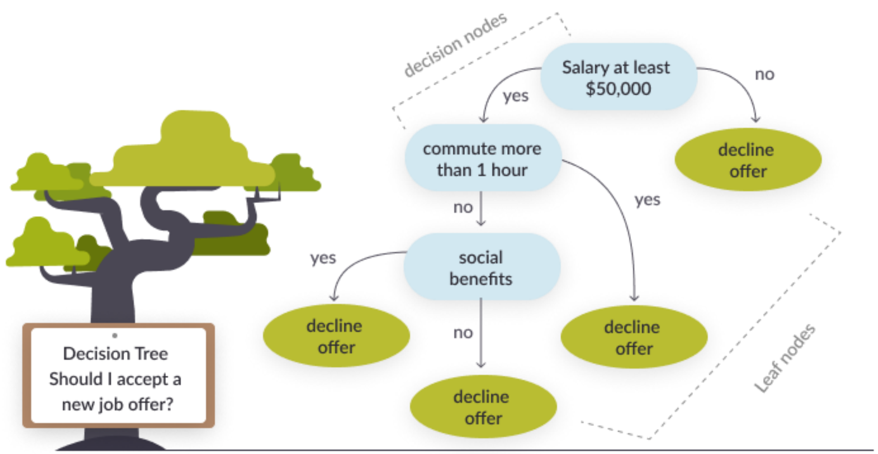


Image from missinglink.ai website

Machine Learning Classification Algorithms

Common classification algorithms

- Random Forest Algorithm
  - A Multiclass classifier with more advanced features than Decision Trees
  - Selects and aggregates the best-performing decision trees

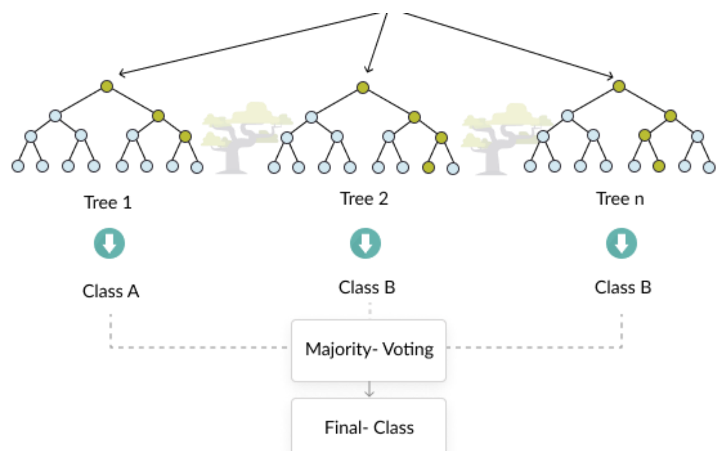


Image from missinglink.ai website

Machine Learning Classification Algorithms

Classification with Neural Networks

- Classifier can be either binary or multiclass
- How do they classify data?
  - Each neuron receives part of the input variables
  - Then, passes on the results to the next layers
  - Now, and after learning process, it can classify almost any functions

Machine Learning Classification Algorithms

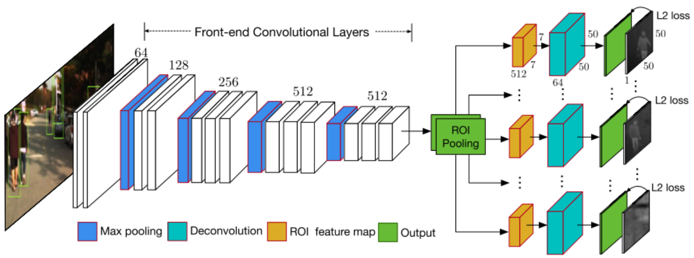
Classification with Neural Networks

- Why?
  - Very effective for high dimensionality problems
  - The ability to dynamically create prediction functions
  - The ability to solve classification problems in an optimized way
- Why not?
  - Computationally intensive
  - Difficult to implement
  - Requiring careful fine-tuning

Deep Neural Networks

Deep Learning

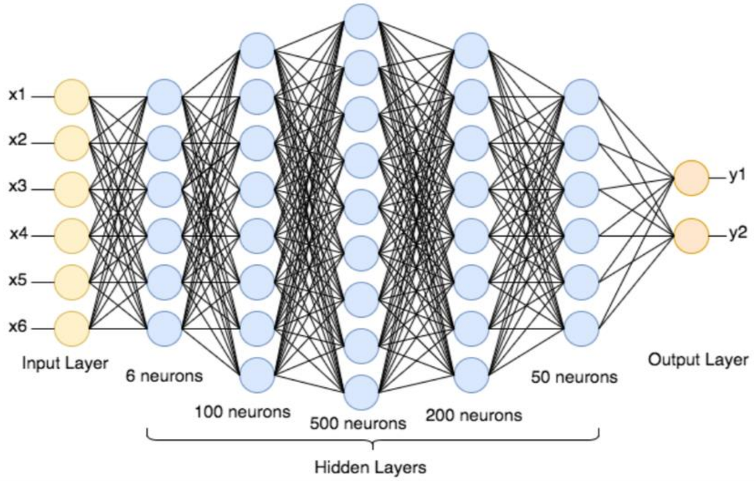
- What is it?!
  - A powerful set of techniques for learning in neural networks
  - Processes data and creates patterns for decision making
  - Able to learn without human supervision
    - How? By **drawing from data** that is both unstructured and unlabeled



Deep Neural Networks

- DNNs can be used in
  - Supervised Learning
  - Semi-supervised Learning
  - Unsupervised Learning

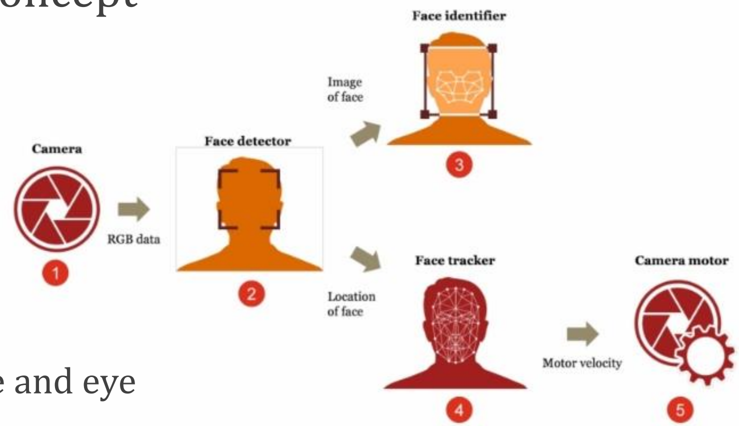
- Some of the most common types:
  - Recurrent Neural Networks
  - Convolutional Neural Networks
  - Autoencoders
  - Deep Belief Net



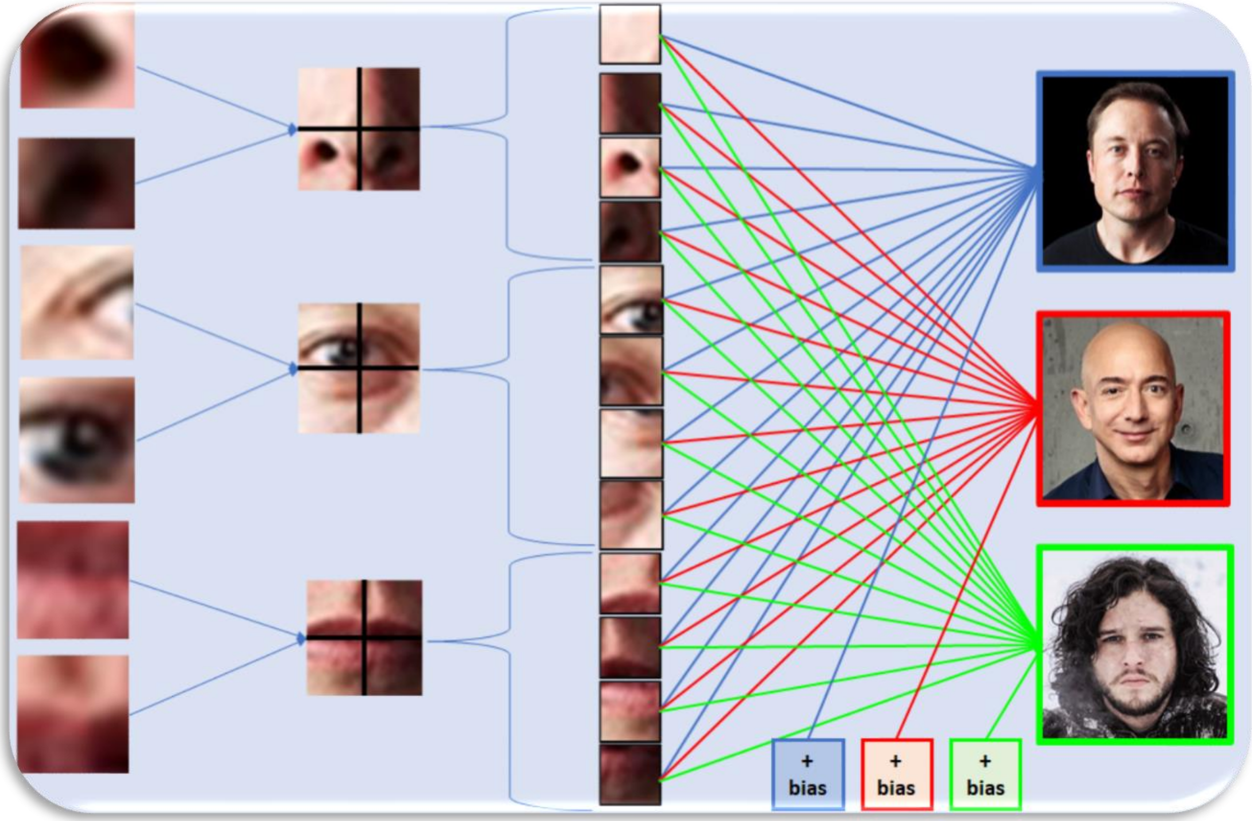
Deep Neural Networks

- Based on many (tens or hundreds) hidden layers
- Each layer extracts high-level features from the previous layer
- Each layer converts data to a more abstract concept

- **Sample application:** face detection
  - Layers a…b (a<b): Analyzing pixels of the image
  - Layers b…c (b<c): Extracting edges and lines
  - Layers c…d (c<d): Extracting visual features, e.g. nose and eye
  - Layers d…e (d<e): Extracting human face



Deep Neural Networks



Deep Neural Networks

- Needs powerful hardware
  - GPU
- Training plays a key role
- Needs a huge amount of data for training
  - Totally based on Data
    - Training-set
    - Validation-set
    - Test-set

Deep Neural Networks

University of Guilan

Machine Learning

- Using algorithms to parse data, learn from that data, and make informed decisions based on what it has learned

Deep Learning

- Structuring algorithms in layers to create an ANN to learn and make intelligent decisions on its own

Note: $DL \subset ML \subset AI$

Deep Neural Networks

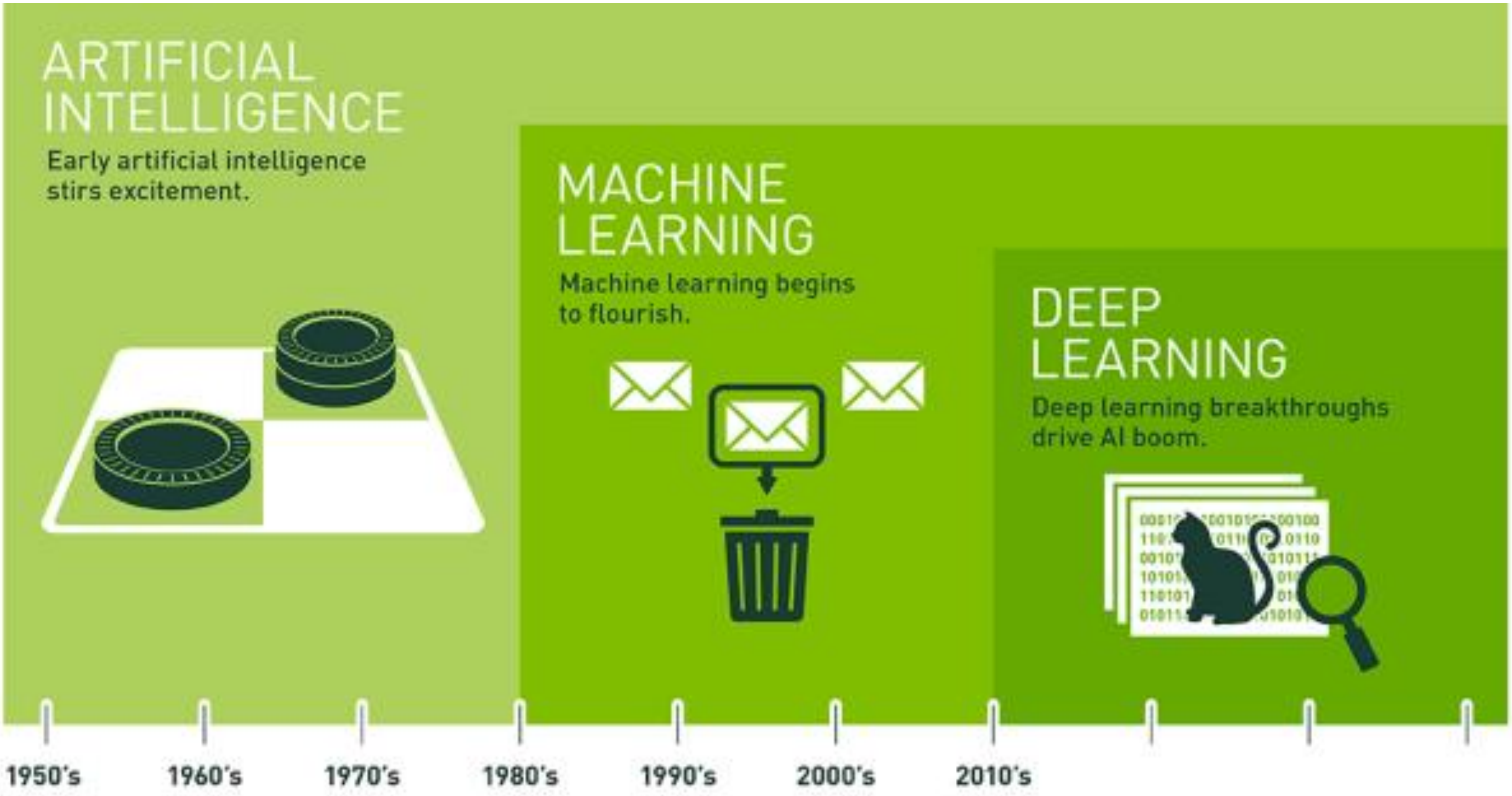


Image from whatsthebigdata.com

Deep Neural Networks

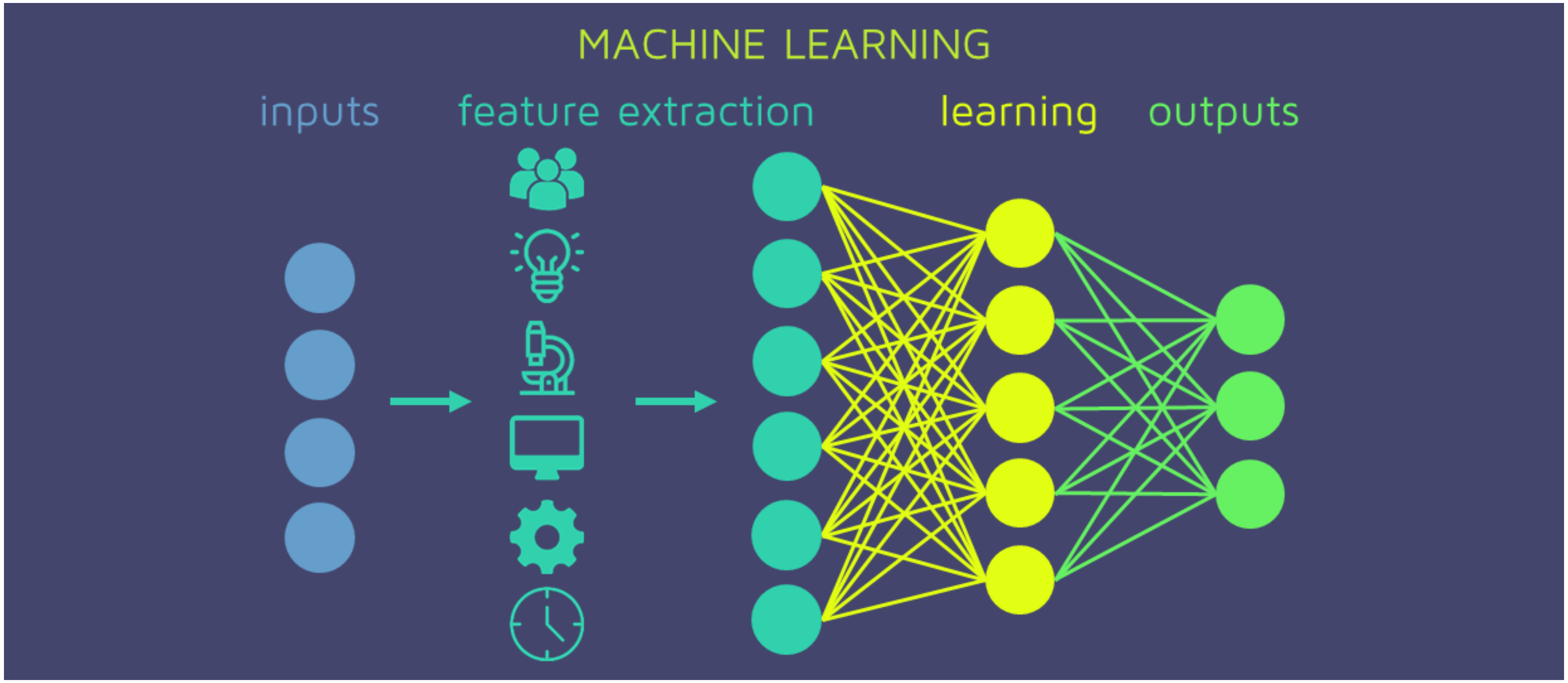


Image from quantdare.com

Deep Neural Networks

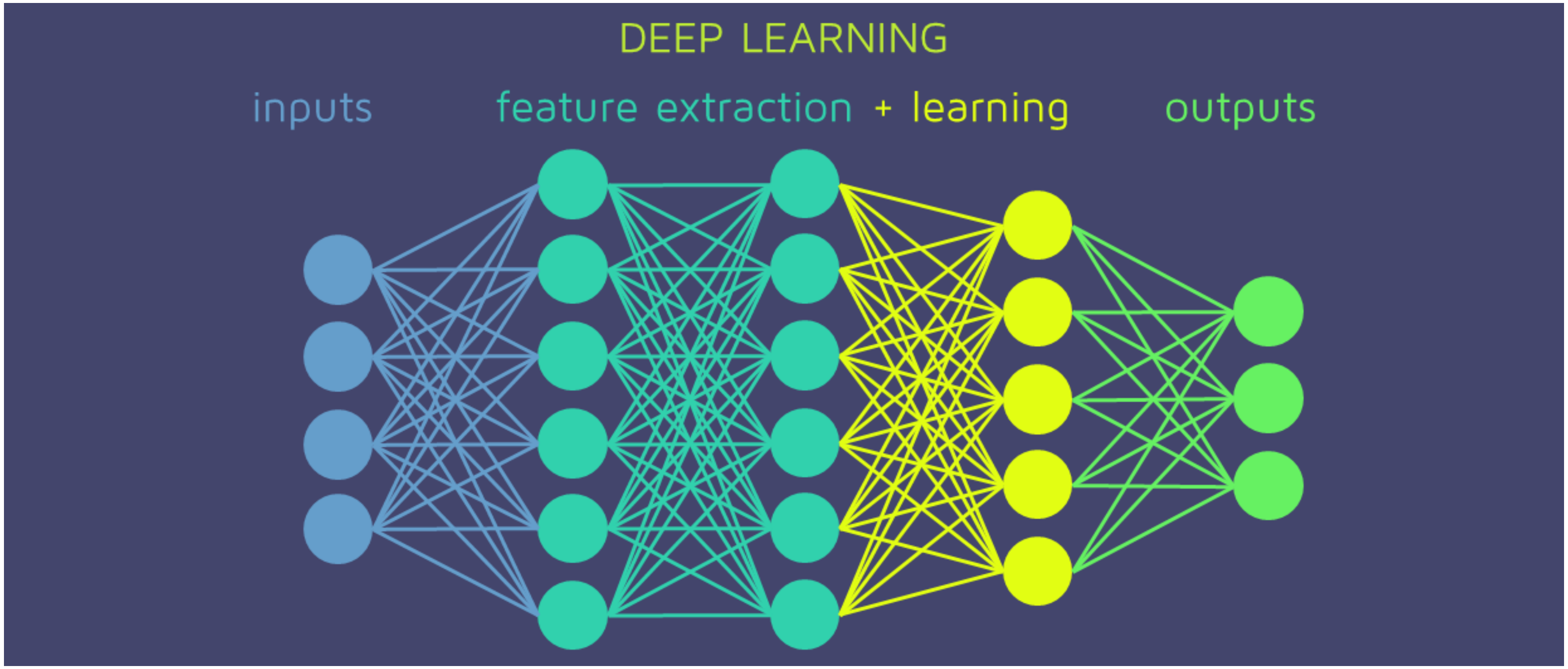


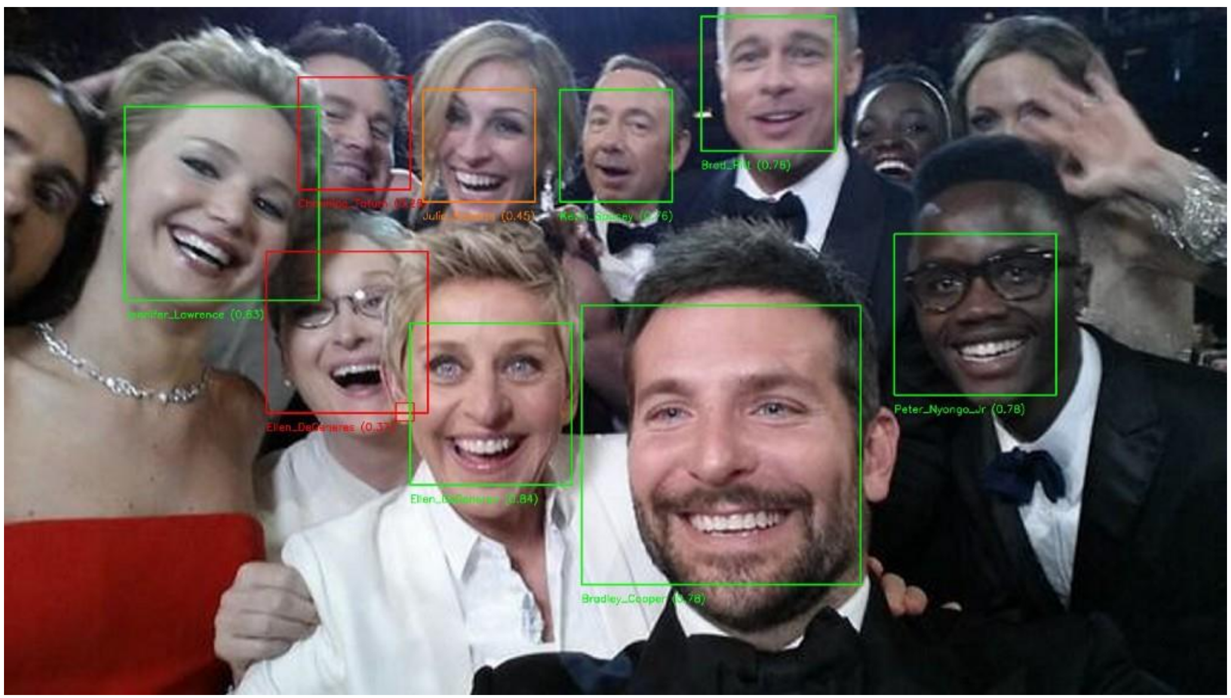
Image from quantdare.com

DNNs Applications

- Most common applications
  - Google Voice Search
  - Google Image Search
  - Face detection in smartphones
  - Handwriting and Signature recognition
  - Image generation using trained samples
  - Automatic Speech Recognition
  - Natural Language Processing

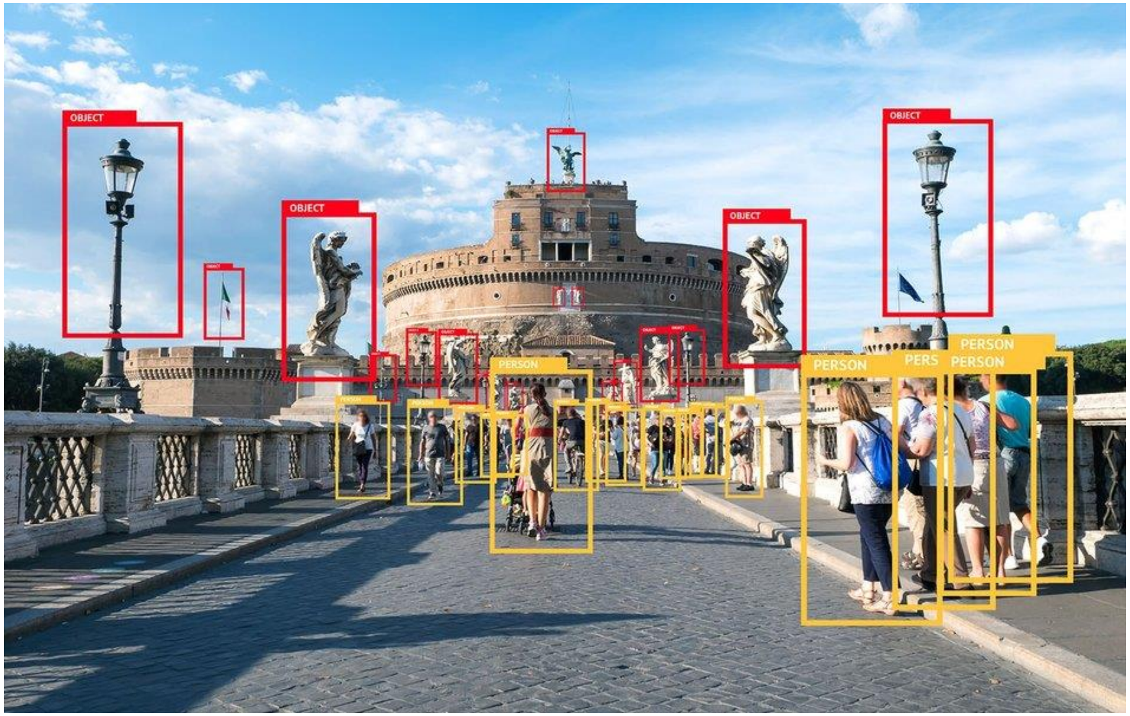
DNNs Applications

- Face Detection/Recognition



DNNs Applications

- Object Detection



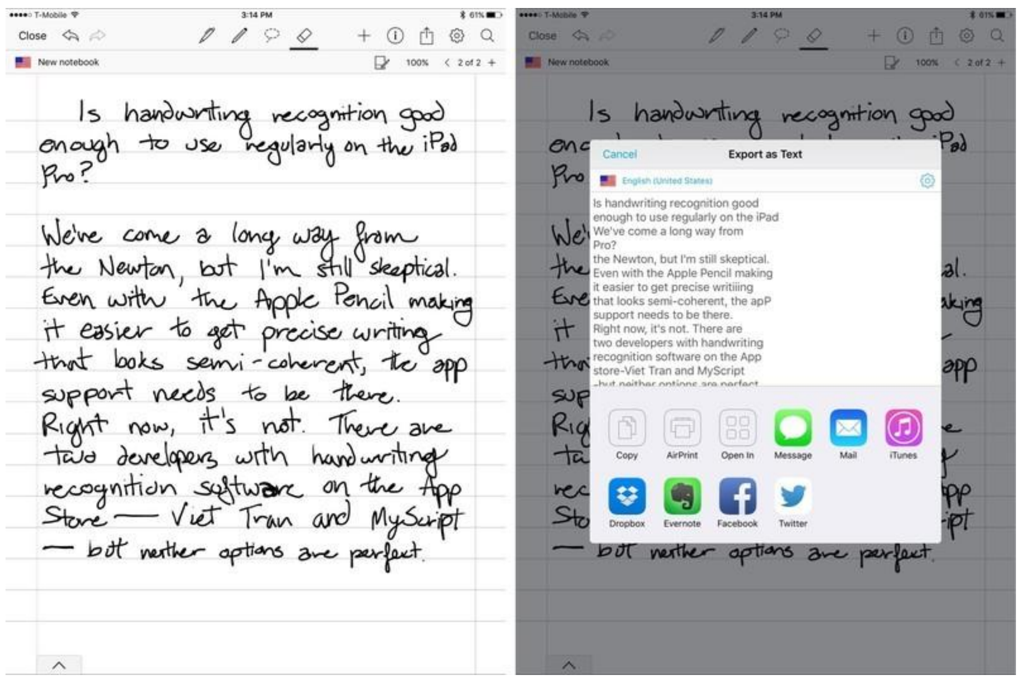
DNNs Applications

- Traffic Sign Detection

DNNs Applications

University of Guilan

- Handwriting Detection/Recognition



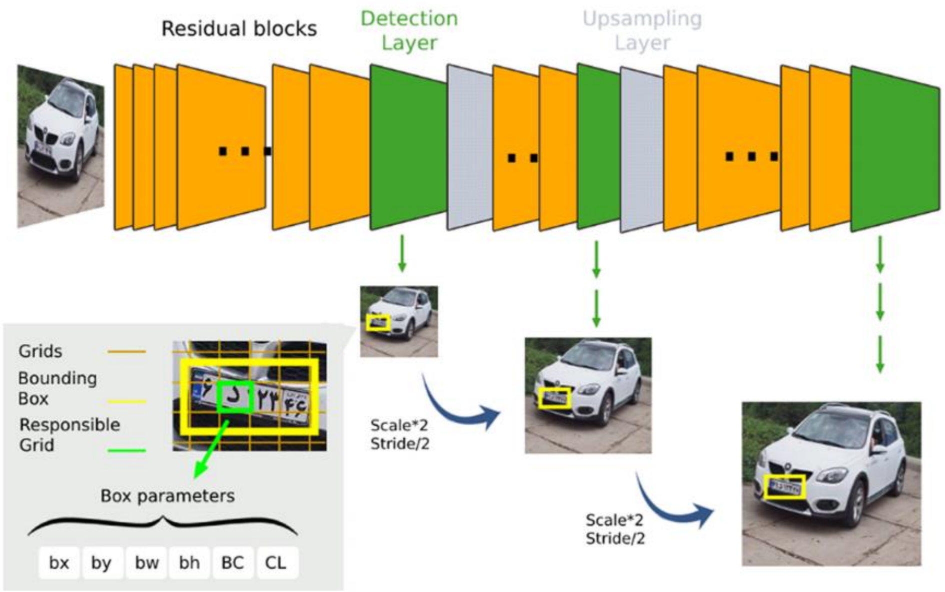
DNNs Applications

University of Guilan

- Iranian Car License-plate Recognition
  - A project supported by Guilan Science and Technology Park (link)

DNNs Applications

- Iranian Car License-plate Recognition
  - A project supported by Guilan Science and Technology Park ([link]())



Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Primarily used in the field of Computer Vision
  - **Input:** images
  - **Outputs**: a single vector of probability scores
  - Not fully-connected structure
  - Very effective at tasks involving data that is closely knitted together
  - A 3D structure with 3 sets of neurons analyzing Red, Green, and Blue layers of a color image
  - Two phases: Convolution and Pooling

Advanced Architectures

University of Guilan

- **Convolutional Neural Networks (CNN)**
  - Phase 1: Convolution
    - Scanning the image
    - Analyzing a small part of it each time
    - Creating a feature map with probabilities that each feature belongs to the required class
  - Phase 2: Pooling
    - Reducing the dimensionality of each feature
    - Maintaining its most important information

Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Convolution

$$= (7*1) + (2*0) + (3*(-1)) \\ + (4*1) + (5*0) + (3*(-1)) \\ + (3*1) + (3*0) + (2*(-1))$$

$$\underbrace{\begin{bmatrix} 7 & 2 & 3 & 3 & 8 \\ 4 & 5 & 3 & 8 & 4 \\ 3 & 3 & 2 & 8 & 4 \\ 2 & 8 & 7 & 2 & 7 \\ 5 & 4 & 4 & 5 & 4 \end{bmatrix}}_{\text{input}} * \underbrace{\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}}_{\text{mask}} = \underbrace{\begin{bmatrix} 6 & & \\ & & \\ & & \end{bmatrix}}_{\text{output}}$$

Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Convolution

$= (2*1) + (3*0) + (3*(-1))$
$+ (5*1) + (3*0) + (8*(-1))$
$+ (3*1) + (2*0) + (8*(-1))$

input

7	2	3	3	8
4	5	3	8	4
3	3	2	8	4
2	8	7	2	7
5	4	4	5	4

*

mask

1	0	-1
1	0	-1
1	0	-1

=

output

6	-9	

Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Convolution

$$= (3*1) + (3*0) + (8*(-1)) + (3*1) + (8*0) + (4*(-1)) + (2*1) + (8*0) + (4*(-1))$$

7	2	3	3	8
4	5	3	8	4
3	3	2	8	4
2	8	7	2	7
5	4	4	5	4

input

*

1	0	-1
1	0	-1
1	0	-1

mask

=

6	-9	-8

output

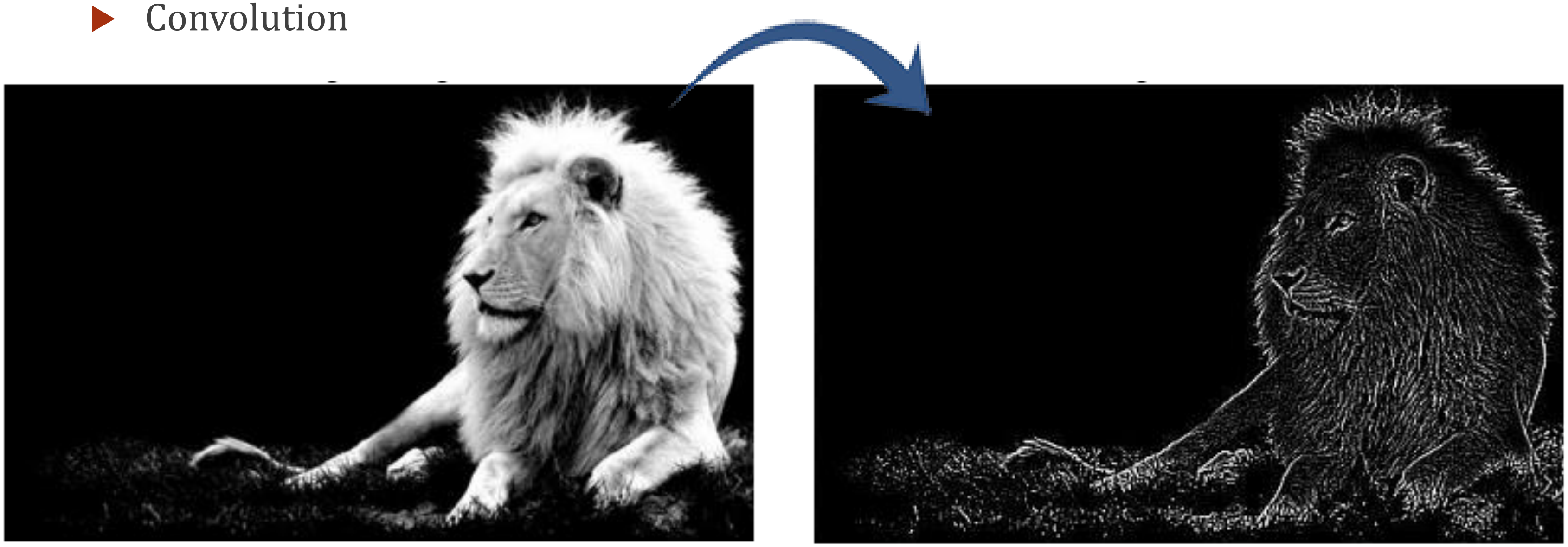
Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Convolution

$$\underset{\text{input}}{\begin{bmatrix} 7 & 2 & 3 & 3 & 8 \\ 4 & 5 & 3 & 8 & 4 \\ 3 & 3 & 2 & 8 & 4 \\ 2 & 8 & 7 & 2 & 7 \\ 5 & 4 & 4 & 5 & 4 \end{bmatrix}} * \underset{\text{mask}}{\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}} = \underset{\text{output}}{\begin{bmatrix} 6 & -9 & -8 \\ -3 & -2 & -3 \\ -3 & 0 & -2 \end{bmatrix}}$$

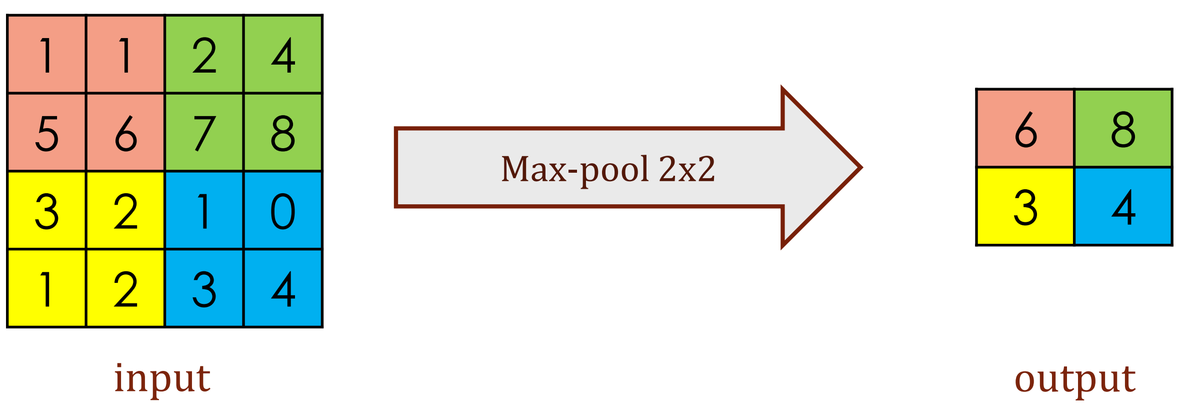
Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Convolution



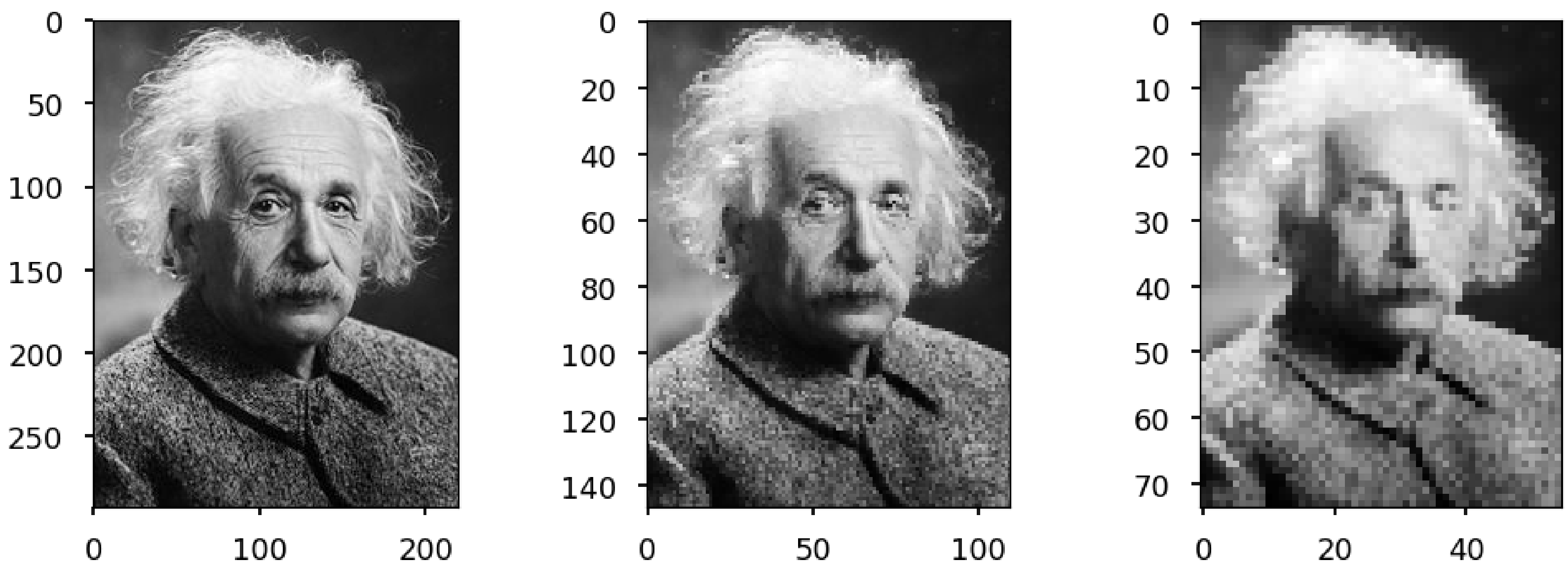
Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Pooling



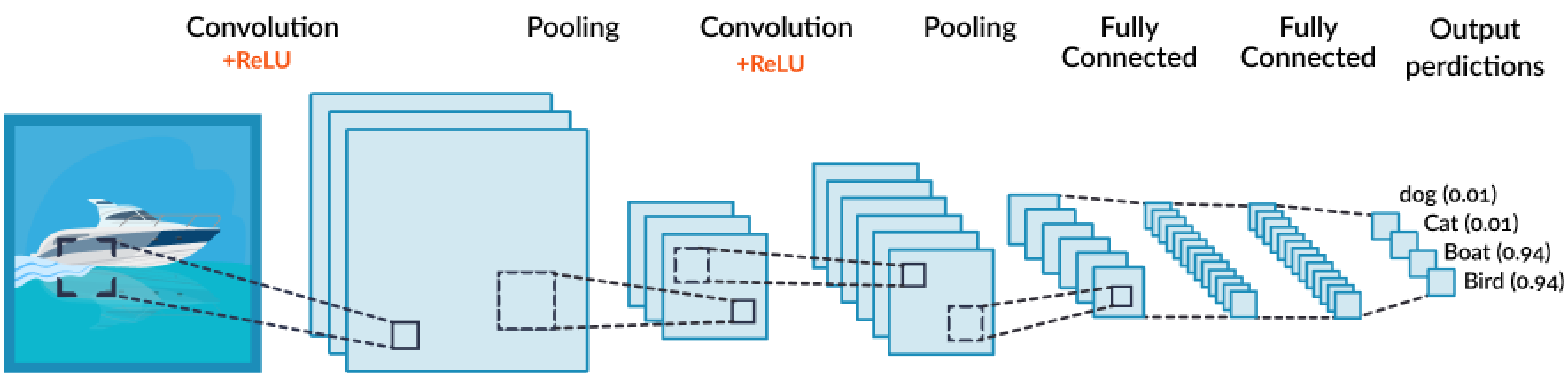
Advanced Architectures

- **Convolutional Neural Networks (CNN)**
  - Pooling

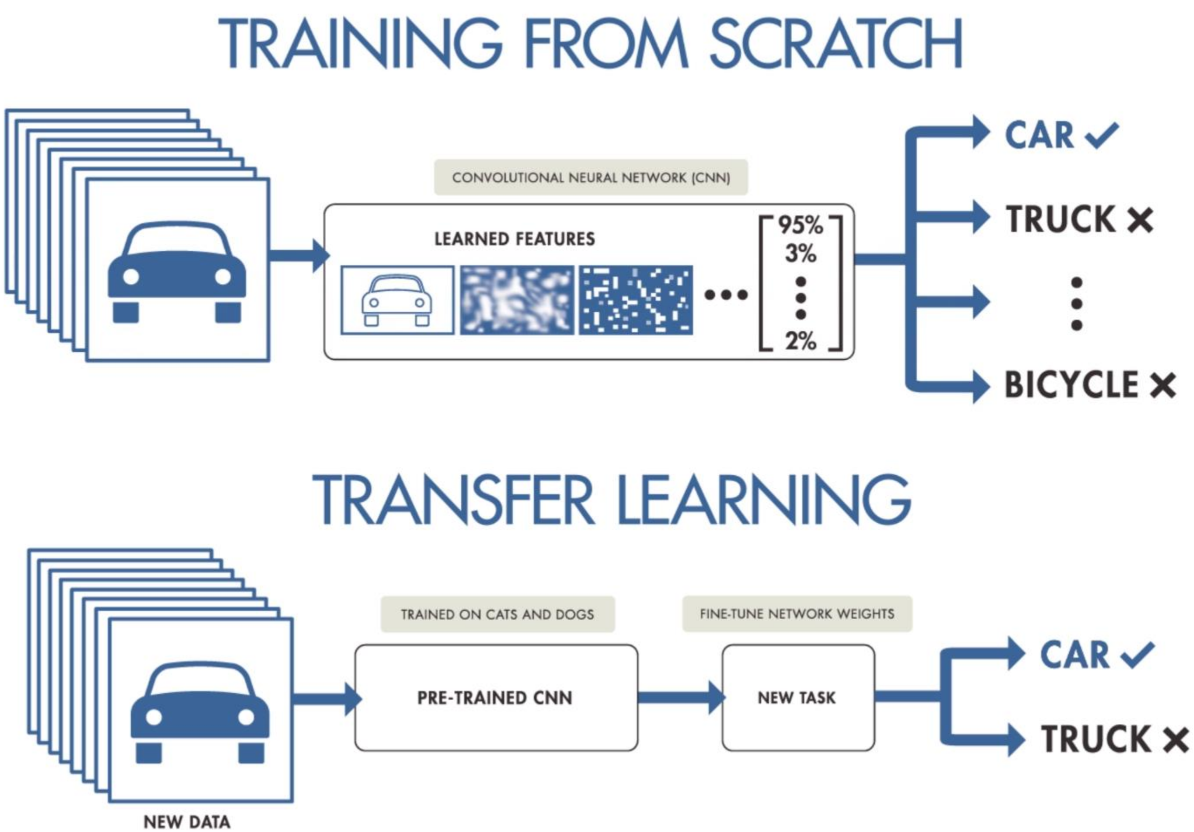


Advanced Architectures

- **Convolutional Neural Networks (CNN)**



Fine-tuning Deep Learning Models



Fine-tuning Deep Learning Models

University of Guilan

What does fine-tuning mean?

- A common practice in Deep Learning
- DNNs have a huge number of parameters
  - Often in the range of millions!
- Solutions?
  - Training a on a small dataset?
    - Greatly affects the accuracy and result in overfitting
  - Fine-tune existing networks that are trained on a large dataset
    - Training it on our small dataset

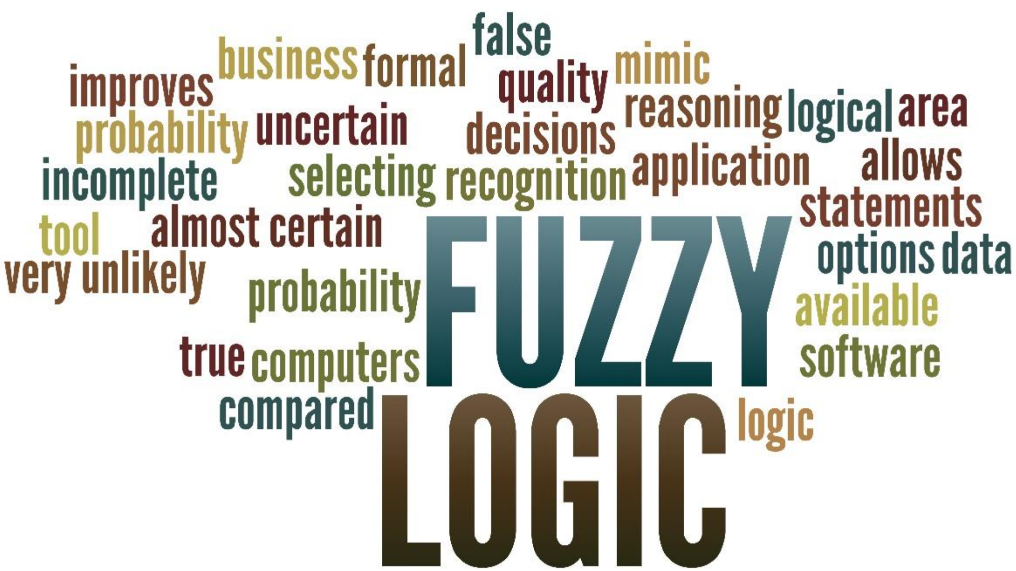
Fine-tuning Deep Learning Models

- **Conditions:**
  - Our dataset is not drastically different in context to the original dataset
    - Like fine-tuning a network that trained on ImageNet for detection of objects
  - The main dataset contains classes that we want
    - For instance, both can be used for face detection

- Note: **Fine-tuning** is one approach to **Transfer Learning**, where we take features learned on one problem, and leveraging them on a new, similar problem

What's Next?

- Fuzzy Basics



Questions?