Data Mining and Text Analytics

Lecture_06: Text Classification and Insights into Machine Learning and Architectures

Postgraduate Programme in AI for Business and Society

A.Y. 2025/2026

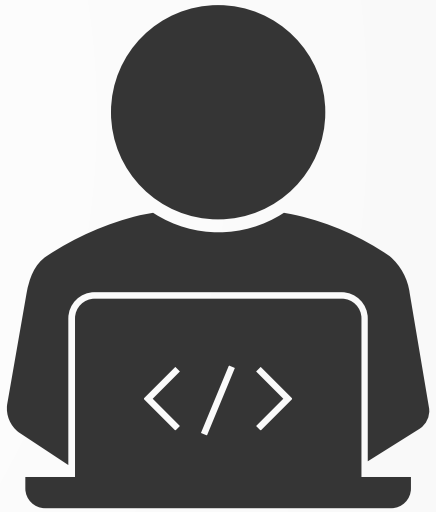
Prof Alessandro Bruno

Traditional vs **Newer** Machine Learning approaches

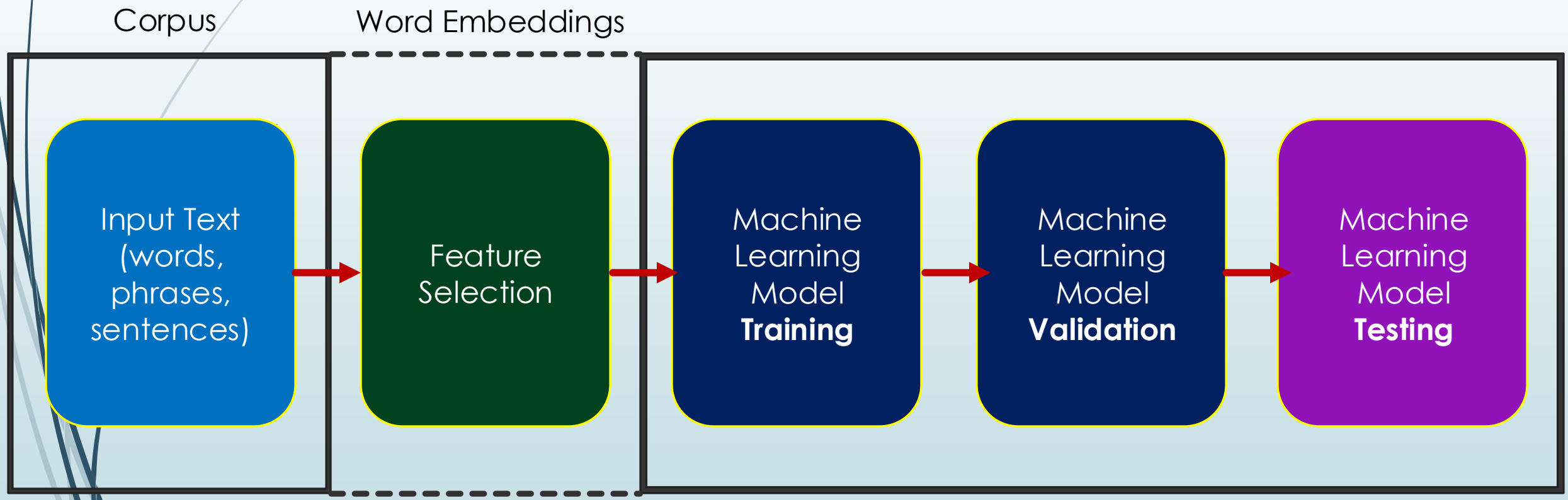
- There are plenty of machine learning methods applied on so many scenarios, which can be grouped into two families: **Traditional** and **Newer Machine Learning** approaches.
- The major different lies within a specific step: feature extraction
- In **traditional Machine Learning approaches** feature extraction is detached from all steps involved in training, validation and testing.
- You need to use feature extraction methods to set up arrays and vectors that better fit your experiments or your application.
- In some cases, you may use statistical descriptors as features representative of a given input data.
- The features are then ingested by the "traditional" machine learning architecture for training purposes.
- The punch line is as follows: you need to be good at extracting features that help a model discriminating categories of interest for your application.

Traditional vs **Newer** Machine Learning Approaches

- "Newer" machine learning approaches mostly refers to Deep Learning, which relies on DNNs (Deep Neural Networks).
- Deep Learning approaches learn hierarchical representations automatically (no manual feature engineering needed)
- They Improve their performance with more data and computational power ("scale up" well)
- They get through huge data volume and extract complex patterns (in unstructured data, such as images, text, speech).



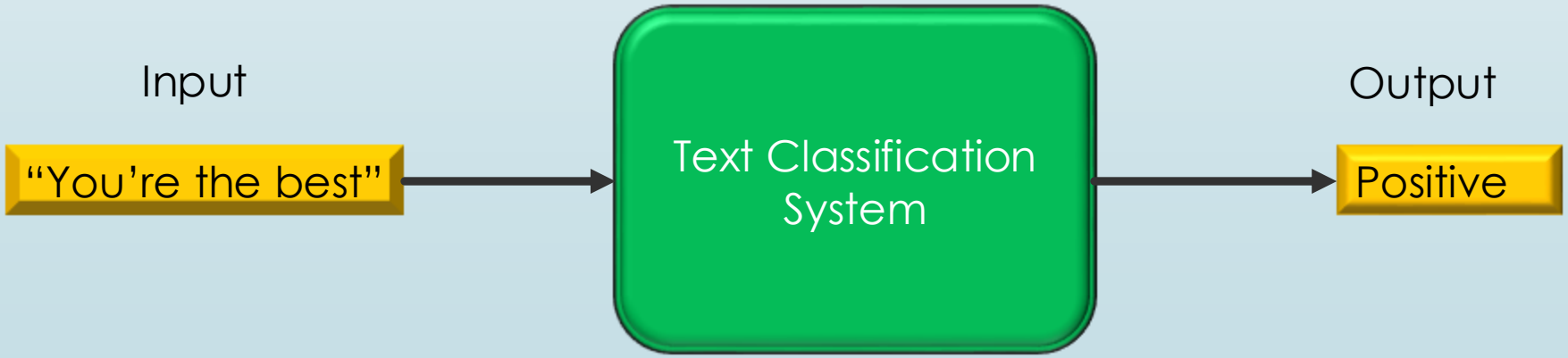
Traditional Machine Learning methods for Text Classification



Please note: a **word embedding** is a numerical array to represent a word (statements, documents, etc.)

Text Classification

- Text classification is also known as text tagging or text categorization
- Text classification is the process of categorizing text into organised groups.
- Text classifiers can automatically analyse text and then assign a set of pre-defined tags or categories based on its content.
- For instance, one might be interested in classifying "good" and "bad" statements.



Traditional Framework of **Text Classification**

The **traditional** framework of text classification consists of three separate components:

- Text Representation
- Feature Selection
- Classification

Text representation plays a critical role.

- it must reflect the content of the text and have sufficient ability to distinguish different types of text.

The choice of text representation

- depends on the classification algorithm.

SVM (Support Vector Machine) is a popular classification method

- It uses the vector space model (VSM) as text representation method.

Feature Selection

Feature selection is the process of selecting a subset of features for text representation and classification.

Feature selection methods normally include supervised and unsupervised approaches.

Unsupervised approaches are applied to a corpus without category annotation. They generally rely on TF (Term Frequency) and DF (Document Frequency).

Supervised approaches rely on category annotation, which can more effectively select a better subset of features for text classification.

Traditional Machine Learning algorithms for Text Classification

Straight after text representation and feature selection, follows a classification algorithm.

The most widely used text classification algorithms used in traditional machine learning are naïve *Bayes, Maximum Entropy, and Support Vector Machines (SVM)*

Naïve Bayes is a probabilistic model for binary and multi-class classification.

Maximum Entropy is used for NLP classification tasks as it assigns P(x,y).

- P(x,y) indicates the the joint probability of data observed (x) and the corresponding label (y).

SVM (Support Vector Machine) is a supervised discriminative learning algorithm used for binary classification. The main goal is to identify an **hyperplane** in the data space where data samples can be separated into.

Running Text Classification using SVM and Naïve Bayes

- Try and configure the off-the-shelf solution at the link below:
  - https://github.com/Gunjitbedi/Text-Classification

- You are given raw data (for instance, text from a webpage)
- You will use SVM and Naïve Bayes to classify text. Some steps are to be accomplished before getting to classification)
- Installing required libraries
- Set random seed
- Add the corpus

- Data Pre-processing
- Prepare Train and Test Datasets
- Encoding
- Word Vectorisation
- Using ML methods to run predictions

Multilayer Feed-Forward Neural Network

- Forward-structure neural network that maps a set of input vectors to a set of output vectors in a multilayer fully connected manner.

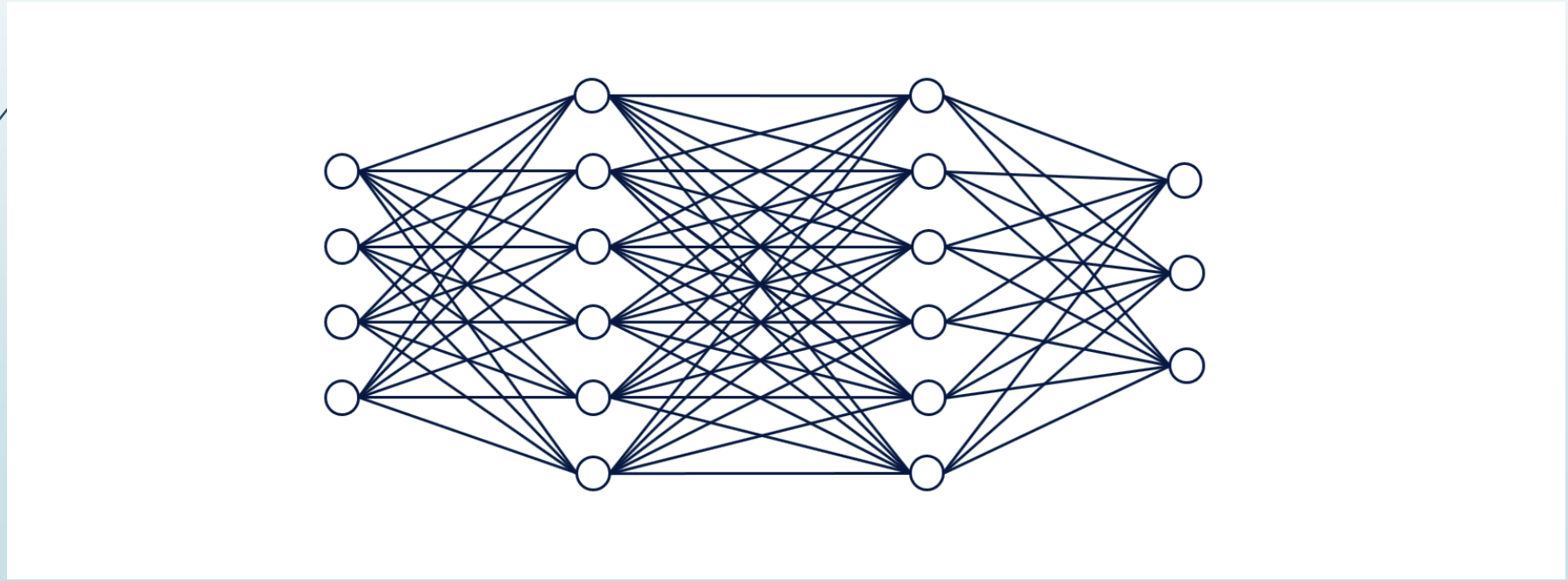
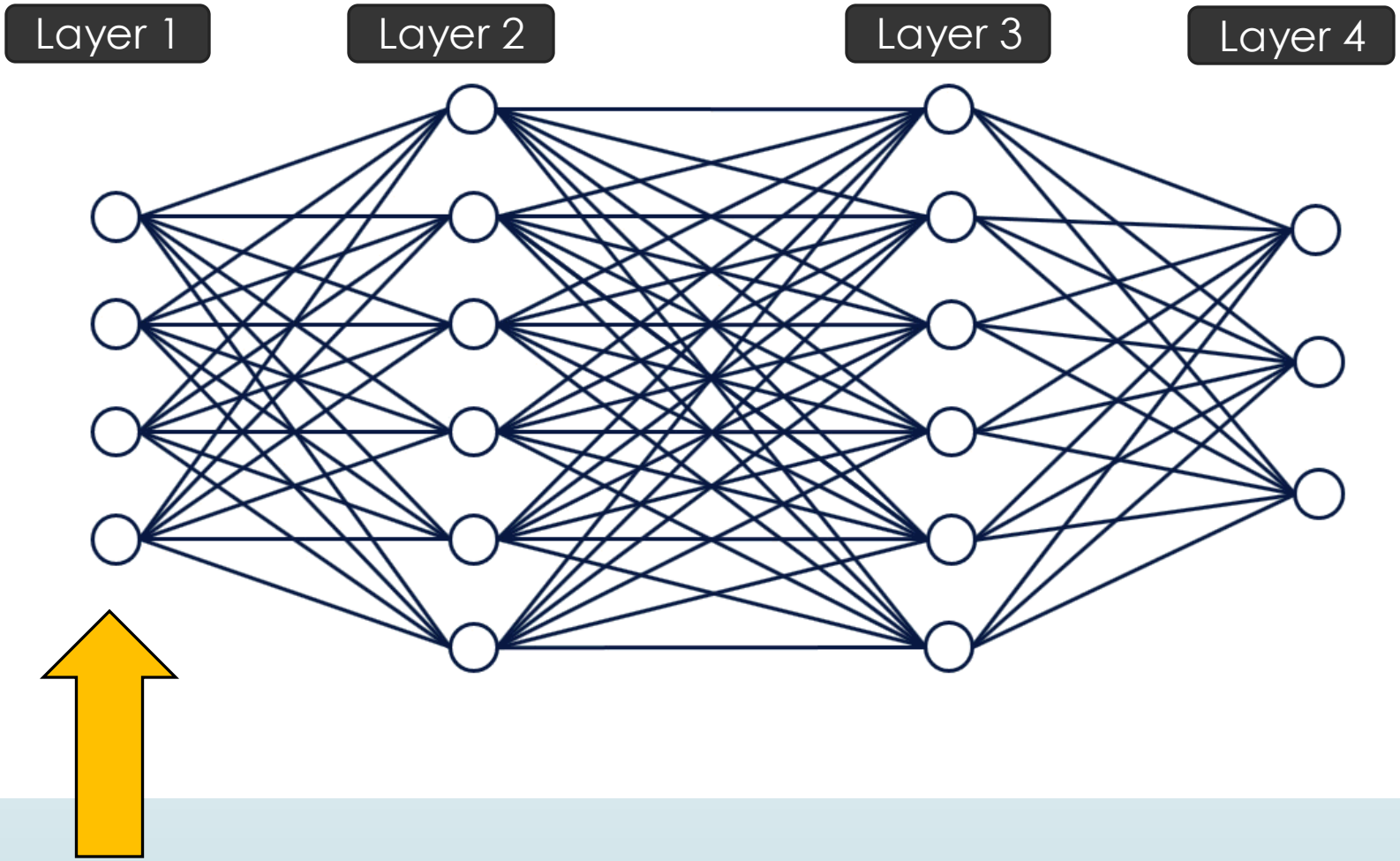
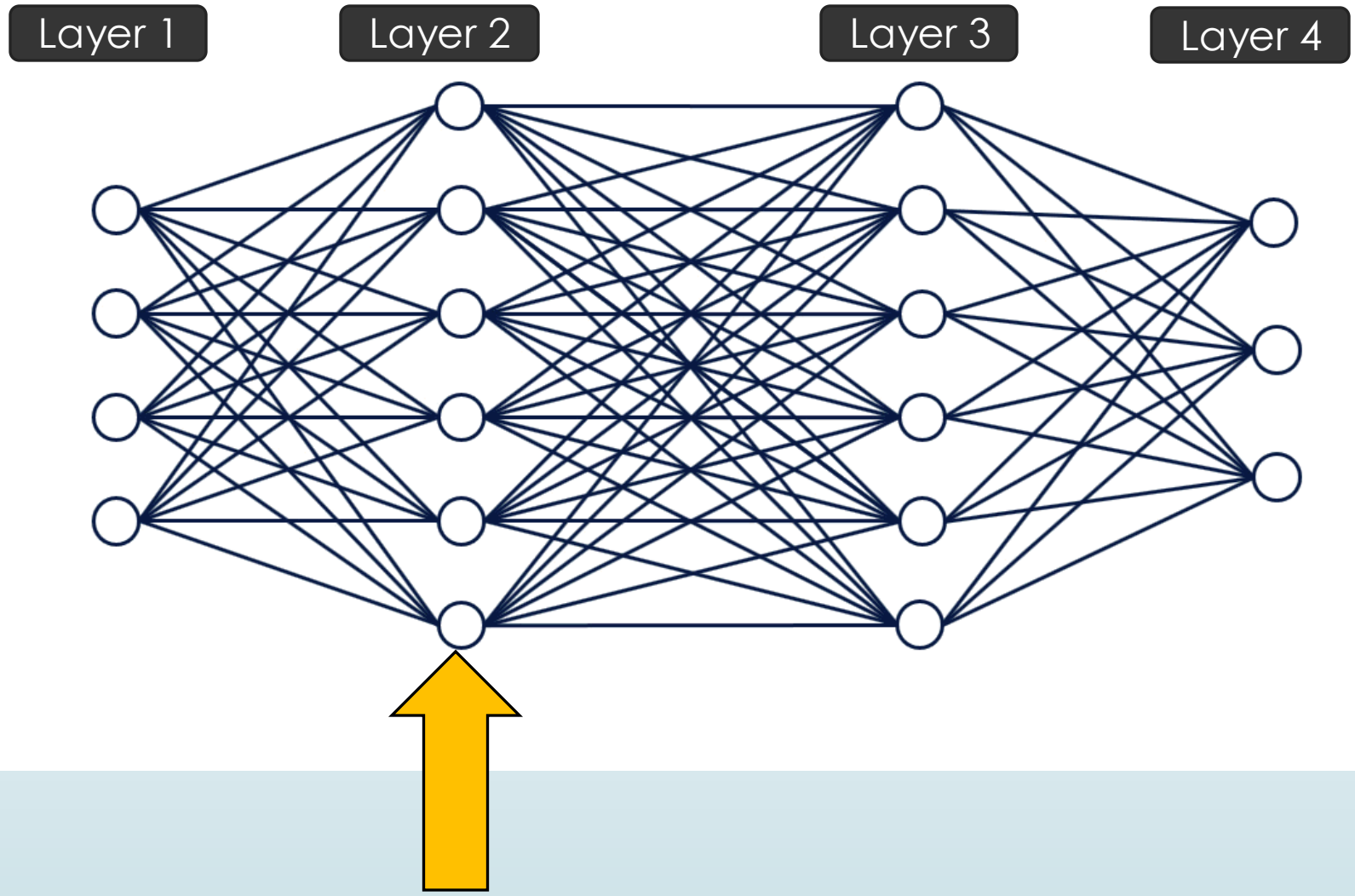


Figure. Feedforward neural network for classification task (Courtesy: Alteryx.com)

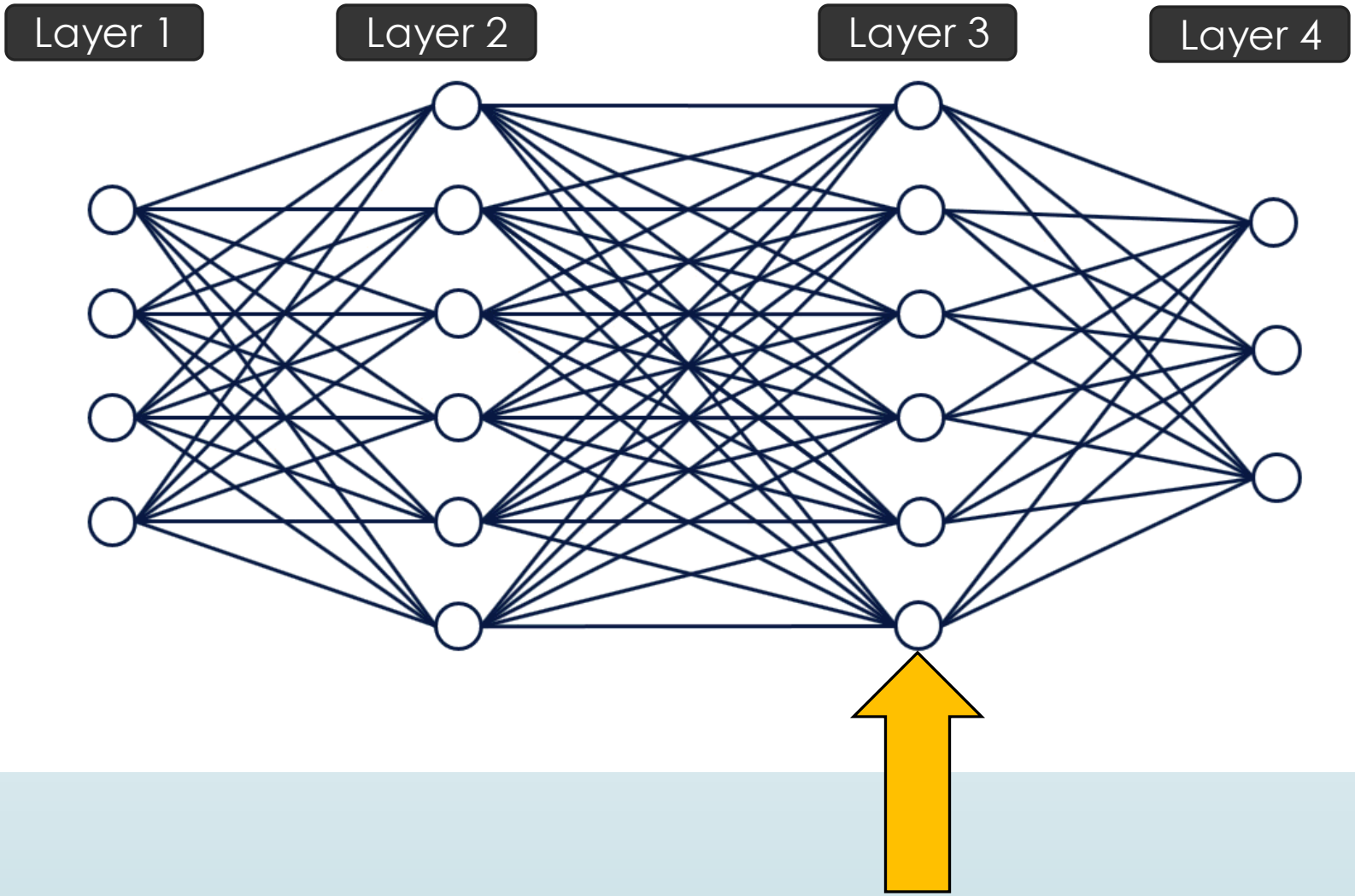


The network consists of 4 different layers:
one input layer (layer 1), two hidden layers (layer 2 and 3) and one output layer (layer 4)

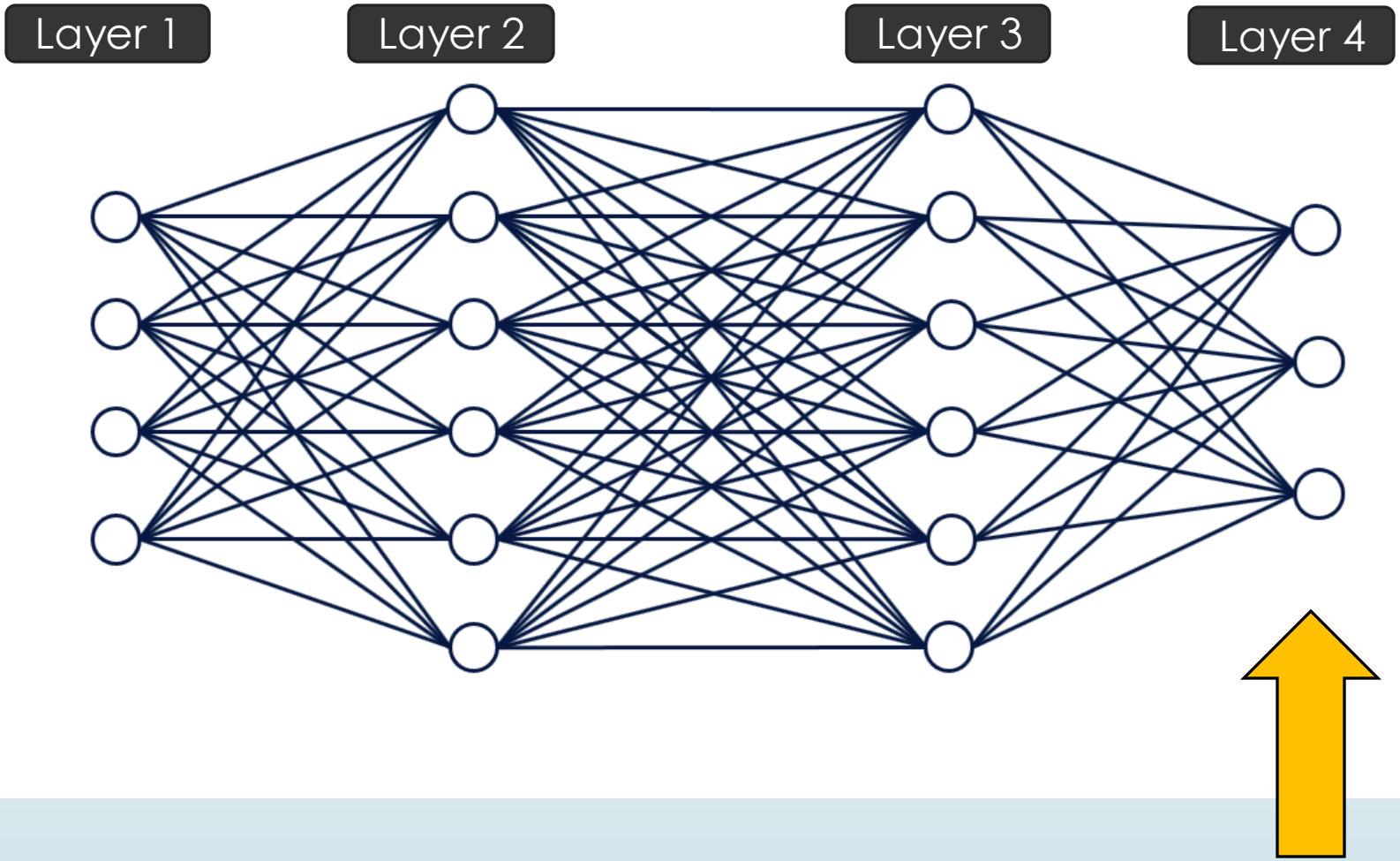
There are four input variables which are fed into different nodes in the the neural network through input layer (1st layer). No computations happen in the input layer.



Layer 2: At each neuron, all incoming values are added together (weighted sum of input signals) and then fed into an activation function.



Layer 3: At each neuron in layer three, all incoming values are added together and then processed with an activation function same as that used in layer 2



Output Layer: At each node in the output layer, all incoming values are added together in different nodes and then processed with a function such as **softmax** function to output the probabilities (in case of classification).

What happens in training steps?

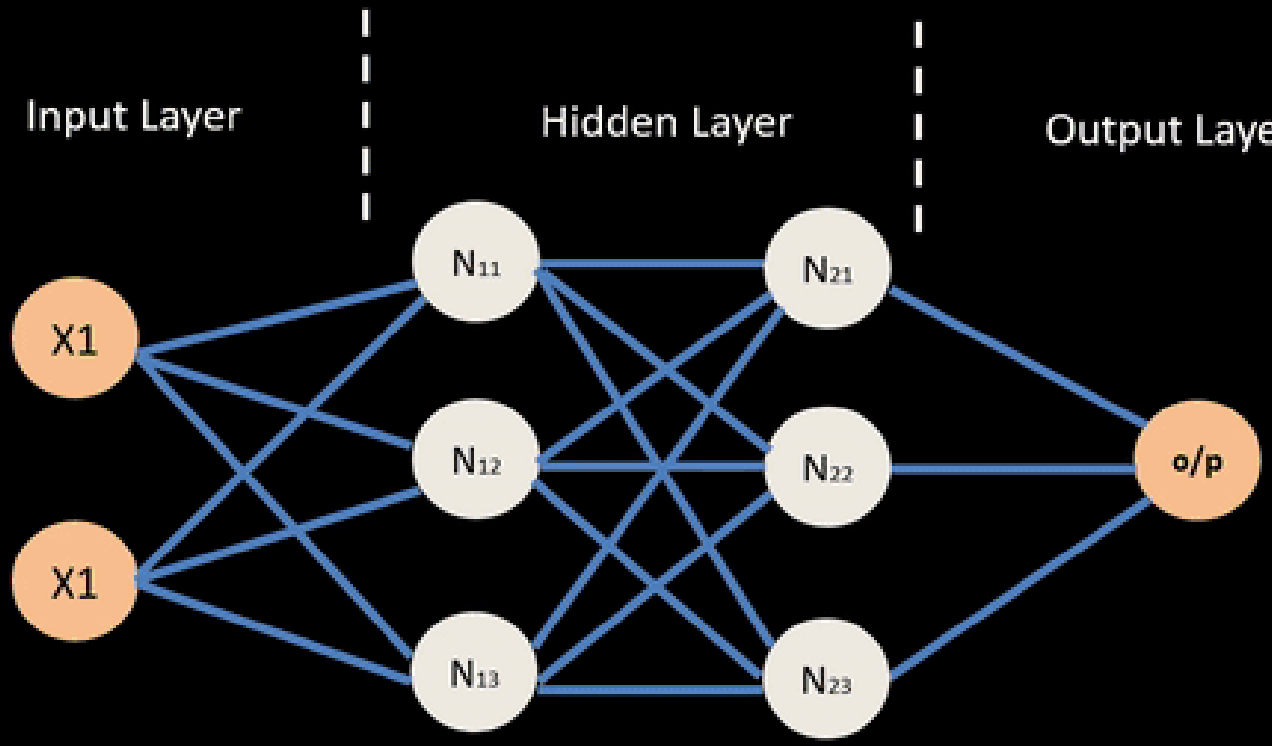
Learning or training can be viewed as the process of optimizing the loss function, that is, determining the optimal parameters of the model that best fits the training data according to the loss function

What is the loss function for?

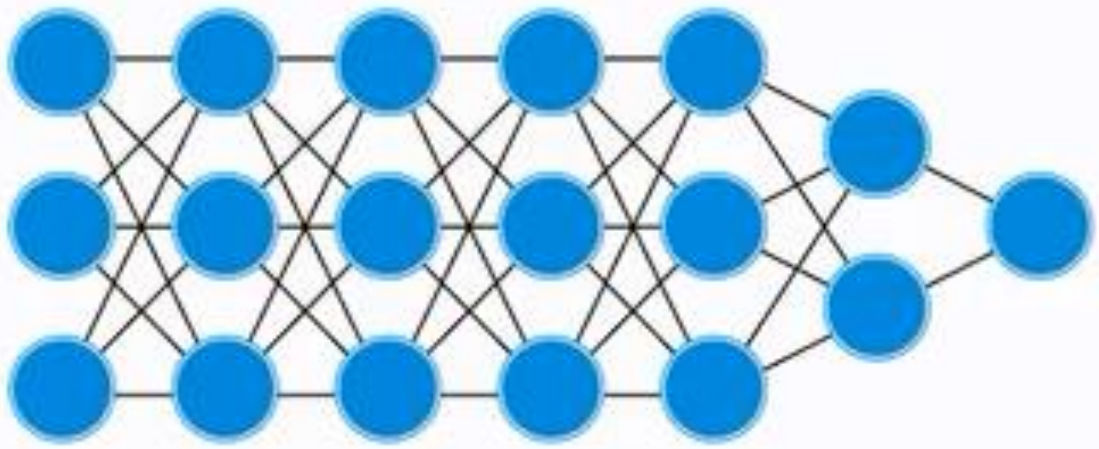
Loss function determines the distance between model's predictions and desired outputs.

Backpropagation works to lower loss function value by updating the network parameters (or weights).

Neural Network – Backpropagation



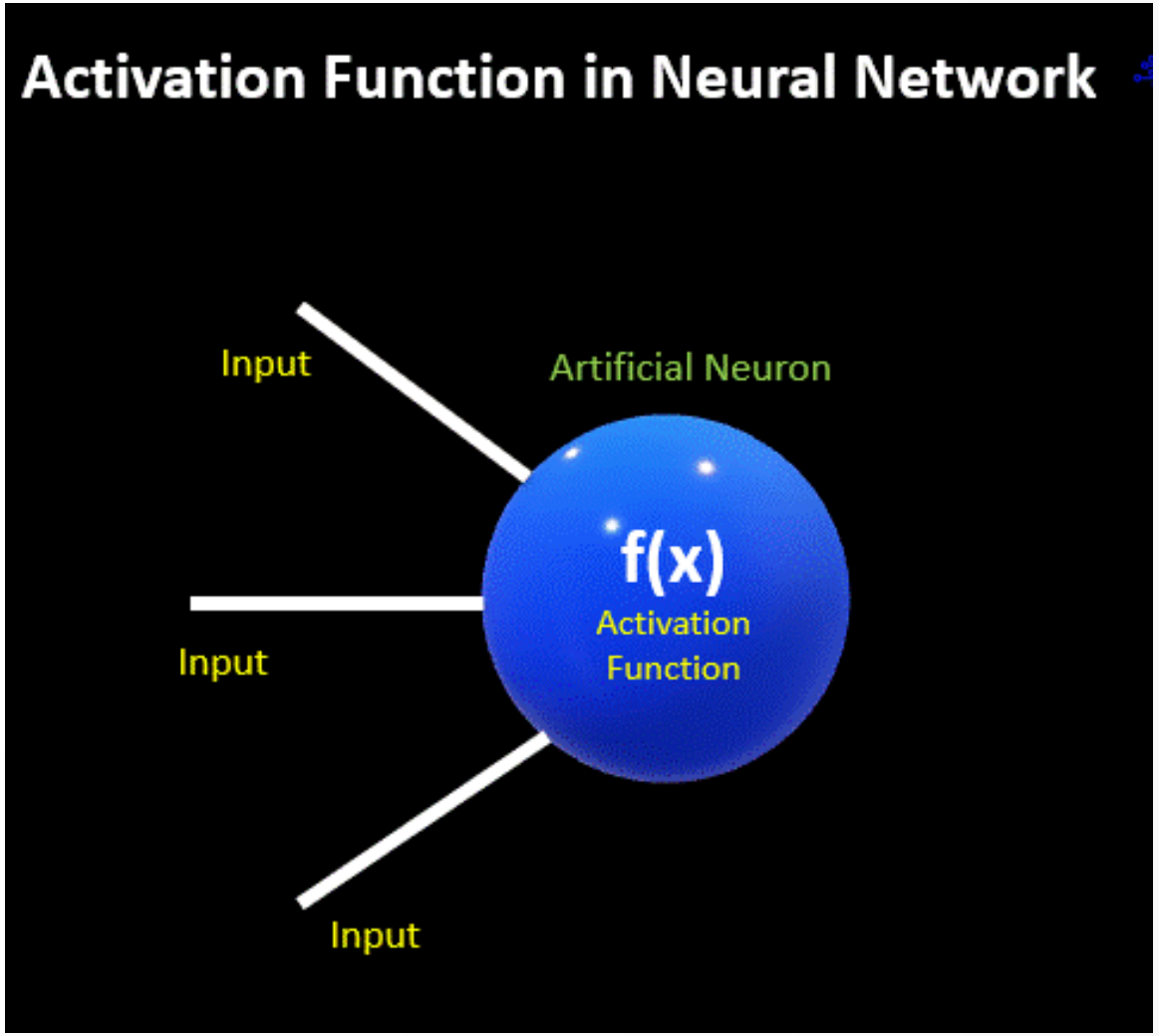
Backpropagation



Taking the stock of backpropagation

- Forward Pass (Starting Point):
  - The network makes a prediction with current weights
  - Compares prediction to actual answer
  - Calculates the error/loss
- **Backward Pass** (**The Core of Backpropagation**):
  - Starts at output layer
  - Calculates how much each weight contributed to the error
  - Moves backward layer by layer
  - Compute gradients (Direction and steepness of error change)
- Weight Update:
  - Adjusts each weight based on its contribution to the error
  - Uses learning rate to control size of adjustments
  - Aims to reduce error in next forward pass

Activation Function



Convolutional Neural Networks (CNNs)

- Convolutional Neural Network is a special kind of feed-forward neural network.
- The CNN's hidden layers consists of a series of convolutional and pooling filters (convolutional layers select features out of the text unput).
- Pooling layers **downsample** the feature vectors and obtain a smaller representation of data.
- Topic and Sentiment Classification tasks started being accomplished with CNNs in 2014. Results showed substantial improvements compared to traditional machine learning algorithms.
- CNNs turned out very accurate in classification tasks (Computer Vision and NLP)

Convolutional Neural Network

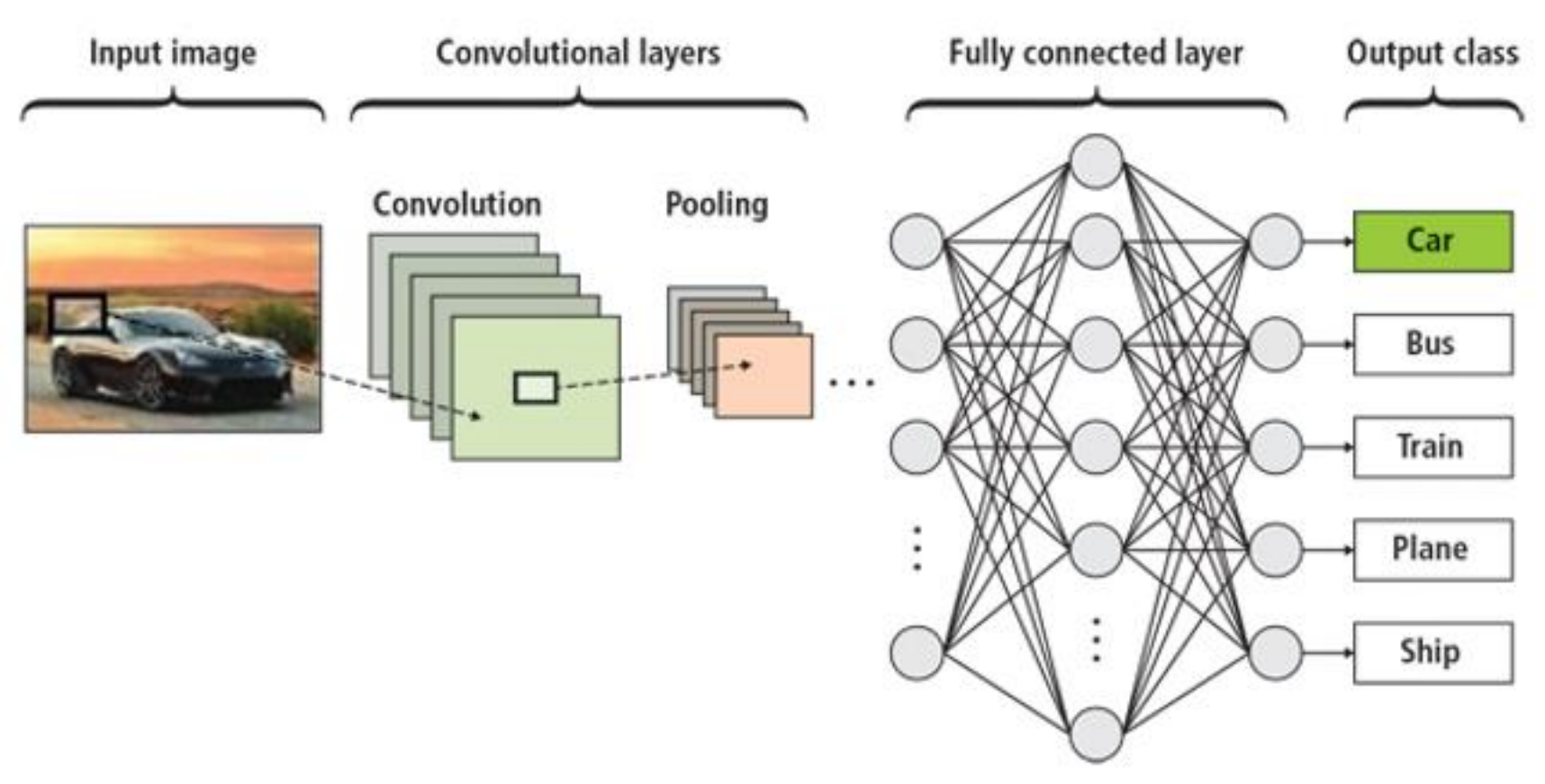


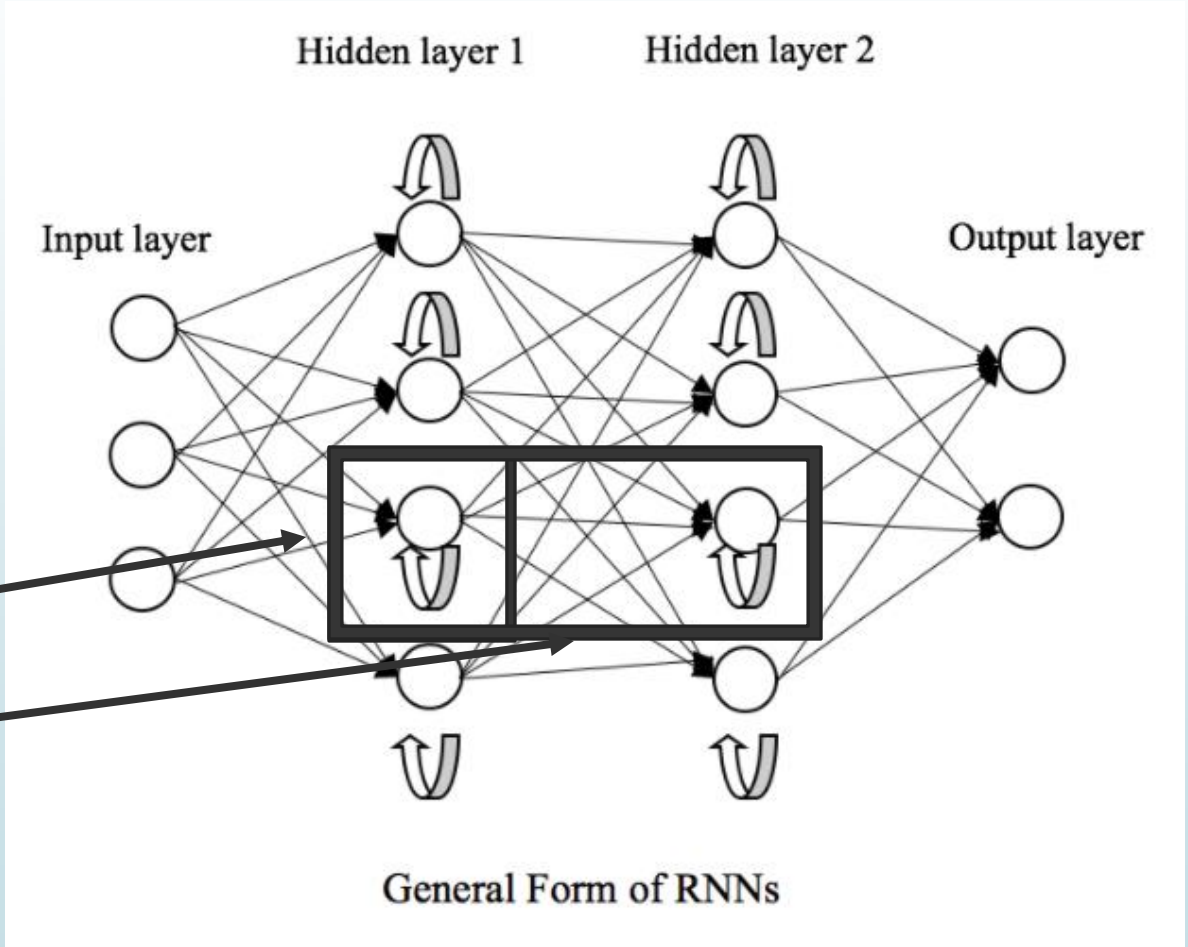
Figure: Courtesy https://www.nvidia.com/en-us/glossary/data-science/convolutional-neural-network/

Assignment (2): Text Classification via CNN

- Try and configure the following GitHub Project:
- https://github.com/cezannec/CNN_Text_Classification

Recurrent Neural Networks (RNNs)

- Recurrent (or Recursive) Neural Networks are a kind of deep neural network that applies the same set of weights recursively over an input (be it a textual input or a visual input).
- Widely adopted in learning sequences in NLP (Natural Language Processing)
- **A structure runs recursively on the network nodes**
- **A structure is expanded to a sequence.**



General Form of RNNs

Deep Learning Hyperparameters

Dataset

- Train-test split ratio (rule-of-thumb: 80/20)

Functions

- Choice of optimization algorithm
- Choice of activation function in a neural network layer
- The choice of loss function the model will use

Layers

- Number of hidden layers in a neural network
- Number of activation units in each layer
- Convolutional Layers size
- Pooling size

Iterations

- Number of iterations (epochs) in training a neural network
- Batch size (The batch size is a number of samples processed before the model is updated)

Text Classification performances

It is helpful to rate performances of classification systems.

- Concepts such as False Positive (FP), True Positive (True Positive), False Negative (FN), and True Negative come at handy and concur in determining performances metrics values.

Here is a short list summarising the meaning of all the above-mentioned concepts:

True Positive (TP) - The model **correctly** identifies a class ("good")

True Negative (TN) - The model **correctly** identifies a negative class ("bad").

False Positive (FP) - The model **incorrectly** identifies a negative class (a "bad" statement is misclassified as "good").

False Negative (FN) - The model **incorrectly** identifies a positive class (a "good" statement is misclassified as "bad").

Confusion Matrix

- Confusion Matrix helps to catch the overall performance of a given system.
- Premise: A given dataset is provided with sample and corresponding labels.

A classification system runs predictions to be compared with actual labels.

For instance, a given dataset consists of "bad" and "good" statements.

The classification system makes predictions on input statements.

"good" represents the positive class while "bad" plays as the negative class.

		Prediction	
		Positive	**Negative**
Actual	**Positive**	**TP**	**FN**
	Negative	**FP**	**TN**

Precision, Recall, F-Measure (F1-score)

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

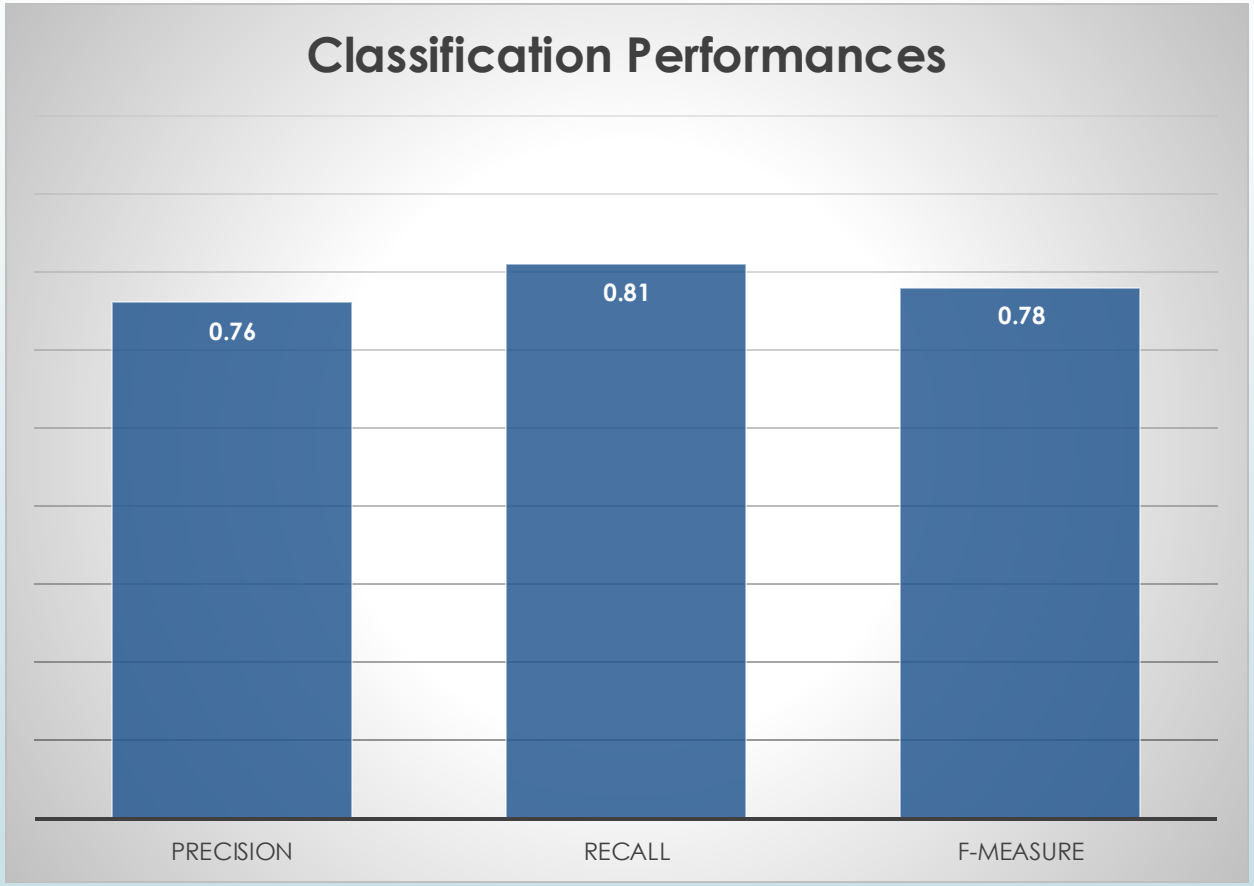
Text Classification Performances

- F1-score (or F-measure) somewhat accounts for Precision and Recall
- However, you may want to diagram all three metrics values to better depict a classification system performances.

- Given a statement classification problem with 100 labelled statements.
- Your model runs through them with the following results:
- TP = 65
- FP = 20
- FN = 15
- Calculate Precision, Recall and F-measure and lay out your considerations.

Solution

- Precision = 65/85 = **0.76**
- Recall = 65/80 = **0.81**
- F-measure = 1.231/1.57 = **0.78**



Comparing two text classification systems

- Given two Text Classification systems, S1 and S2. You are asked to benchmark them on top the following results:

- S1
  - TP = 65
  - FN = 5
  - FP = 30
- S2
  - TP = 62
  - FN = 12
  - FP = 26


That's all