Artificial Models for Music Creativity

Lesson 1 - Myths and Core Concepts of Artificial Intelligence

Alessandro Anatrini - 14.10.2023

Four Myths on Artificial Intelligence

1. Machine Learning (ML) Art is new
2. ML can create Art without the Artist
3. AI, ML and Deep Learning (DL) are all the same
4. ML will give rise to superhuman Intelligence

Sofian Audry, *Art in the Age of Machine Learning*, 2021

ML Art is new

2

ML can be traced back to the early days of cybernetics in the 40s. The expression ML first appeared in the 50s around the same time as AI. Artists have been using adaptive or learning computational systems since then, through various artistic movements such as systems art, algorithmic art, robotic art, and evolutionary art. Yet, the presence of such approaches in artistic works is often hard to trace because they are frequently used more as metaphors than as actual techniques

ML Art is new

1. A minimal definition of feedback takes into account the configuration of a system, provided with input and output, in which some kind of transformation occurs, where the output is connected to the input after a delay. From a theoretical point of view, we can think of a zero-delay feedback loop as a system whose fundamental frequency is infinity
2. Feedback can be negative or positive, leading to an equilibrium or an exponential scenario
3. DSP domain applications: filters, synthesis algorithms, delays

ML Art is new

The use of feedback systems gained popular momentum in the 1960s in relation to the success of cybernetics (Wiener 1948) and system theory (von Bertalanffy 1950), two trans-disciplinary epistemological approaches that, starting from the 40s, strongly emphasized the relevance of closed information loops in organized structures. A big role in bridging and questioning cybernitics and system theory is the work of G. Bateson (1972)

ML Art is new

1. 1948 - 1960s – N. Wiener, W. Ross Ashby, Pitts & McCulloch...
2. 70s - 80s – G. Bateson, M. Mead, Maturana & Varela...
3. from the 90s – N. Luhmann, E. Morin, I. Prigogine, F. Capra...

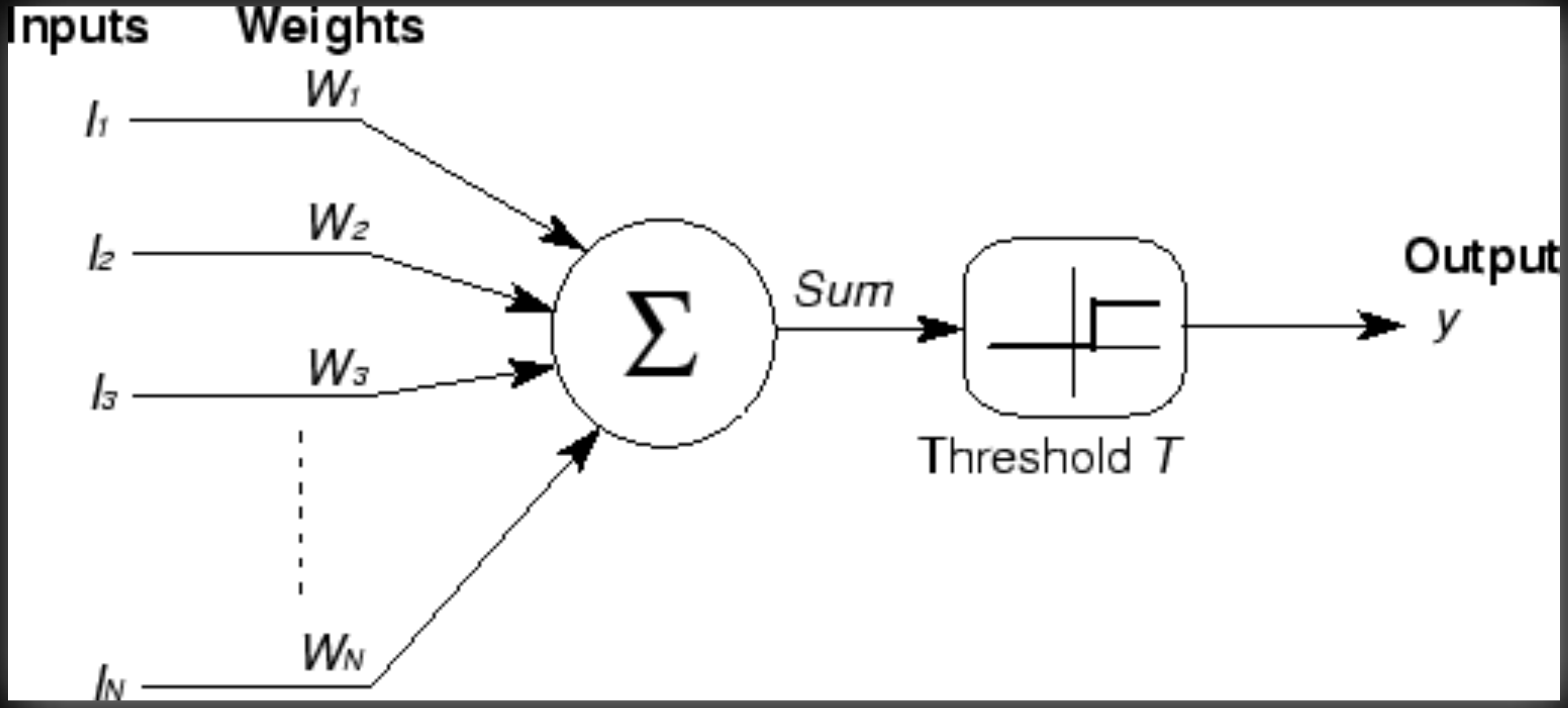


Fig.1 The Pitts-McCulloch model of neuron. (Maren et al., 1990)

ML Art is new

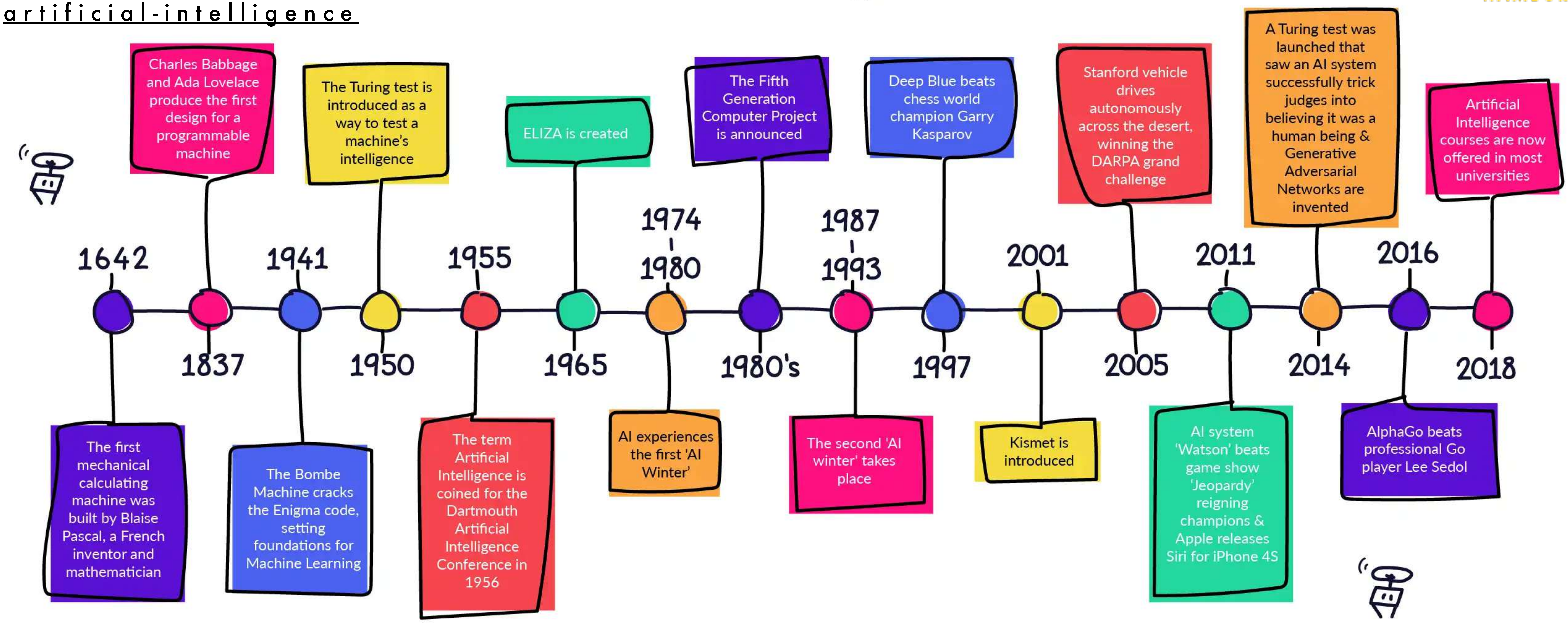
1. Cybernetics has vanished, but its significance in the development of computer technologies (AI, ML) cannot be ignored. Influence on the society and culture of the 60s, particularly in the field of contemporary art, in which cybernetics was closely related to movements such as conceptual, performance and kinetic art
2. Computationalist theories of mind (cognitivism)
3. Symbolic AI vs. ALife vs. Machine Learning

ML Art is new

Although its story is deeply rooted in cybernetics (Goodfellow, Bengio, & Courville, 2016), current-day ML has not embraced the cyberneticians's utopian dream of self-regulating technologies, relying instead on a relatively traditional engineering culture that attempts to efficiently solve concrete, measurable problems such as recognizing patterns or predicting future quantifiable events; in other words, attempting to take part in an ecosystem

ML ART IS NEW

Fig.2 Brief history of AI. Source: https://codebots.com/artificial-intelligence/history-of-artificial-intelligence



ML can create Art without the Artist

1. Composition
2. Performance
3. Music Theory
4. Digital Signal Processing

Curtis Roads, *Research in Music and Artificial Intelligence*, 1982

ML can create Art without the Artist

1. Pre-history: A. Turing, N. Schöffer, J. Cage, L. Hiller
2. "Cybernetics": R. Kayn, Cybernetic Serendipity
3. Early practices: N. Collins, A. Lucier, S. Reich, D. Tudor, EMS
4. "Systemists": N. Baginsky, A. Di Scipio

ML can create Art without the Artist

1. R. Zaripov (1965), R. Kurzweil (1965)
2. D. Cope - "Experiments in Musical Intelligence" (1980-1997)
3. D. Cope - "Emily Howell" (2009-2012)
4. Sony CSL - "Daddy's car" (2016)
5. T. Southern - "I am AI" (2017), Actress - "Young Paint" (2018)
6. H. Herndon ft. Jlin - "Godmother" (2019)
7. Dadabots - "Bot prownies" (2019) - Miscellanea

ML can create Art without the Artist

1. MusicLM
2. soundraw.io, Amper Music, AIVA, Amadeus Code
3. Humtap
4. generative.fm, Endel
5. ann, ml-lib (Pd), ml.star, MuBu, FluCoMa, rapidlib, orchidea
6. Magenta, RAVE, DWS, DDSP, WaveNet, Faust2JAX

AI, ML AND DL ARE ALL THE SAME

4

1. AI is the study of training your machine to mimic a human brain and its thinking capabilities

2. AI focuses on 3 major skills: learning, reasoning, and self-correction to obtain the maximum efficiency possible

3. Speech Recognition, Personalised Recommendations, Predictive Maintenance, Medical Diagnosis, Autonomous Vehicles, VPA, Fraud Detection, NLP, Predictive Analytics

4 AI, ML and DL are all the same

1. ML is a subset of AI that involves the use of algorithms and statistical models to allow a computer system to "learn" from data and improve its performance over time
2. The major aim of ML is to allow the systems to learn by themselves through experience without being explicitly programmed to do so
3. Image Recognition, Speech Recognition, NLP, Sentiment Analysis, Spam Filters, Credit Risk Assessment, Customer Segmentation, Predictive Maintenance, Fraud Detection

AI, ML AND DL ARE ALL THE SAME

1. DL is a subset of ML which makes use of Neural Networks (NNs) - similar to the neurons working in our brain - to mimic human brain-like behaviour
2. DL algorithms focus on information processing patterns mechanism to possibly identify the patterns just like our human brain does and classifies the information accordingly
3. Generative Models, Image and Video Recognition, Time Series Forecasting, Game-playing AI, Speech Recognition, NLP, Fraud Detection, Image Classification

AI, ML AND DL ARE ALL THE SAME

Artificial Intelligence	study which enables machines to mimic human behaviour through particular algorithm	Search Trees and much complex math is involved in AI	mimic human intelligence to increase the success rate of the task	ANI, AGI, ASI	The efficiency Of AI is basically the efficiency provided by ML and DL
Machine Learning	study that uses statistical methods enabling machines to improve with experience	If you have a clear idea about the logic involved in behind and you can visualize the complex functionalities like K-Mean, Support Vector Machines…	improve performance of specific task, thus increase accuracy	Supervised/ Unsupervised Learning, Reinforcement Learning	Less efficient than DL as it cannot work for longer dimensions or higher amount of data
Deep Learning	study that makes use of Neural Networks to imitate functionality just like a human brain	If you don't have idea about the features, so you break the complex functionalities into linear/ lower dimension features by adding more layers	focus on handling complex data to achieve highly accurate results in specific tasks	Deep Unsupervised pre-trained networks, CNN, GAN, RNN, VAE	More powerful than ML as it can easily work for larger sets of data

Core Concepts 1

Regression: the goal is to predict a numerical (or continuous) value as the output. For example, predicting the price of a house based on its features is a regression problem as well as to predict the next MIDI note. In this case, the model aims to fit a function to the data in order to predict a numerical value

Classification: the goal is to assign a category or class to an input. For example, object recognition in an image is a classification problem where the model seeks to assign a label to each detected object. The labels are discrete

Core Concepts 1

Data: Information or **Observations** that after being collected or recorded are used as input. It can include text, numbers, images or other types of information

Learning Process: It involves **Training** a ML model by exposing it to data. The model adjusts its internal parameters to make predictions or decisions based on the data it has seen

Model: It is a mathematical or computational representation of a real-world process or system. In ML, a model is designed to make predictions, classifications, or decisions based on input data

Core Concepts 1

5

Perceptron: It is a fundamental type of NN used for binary classification tasks. It consists of a single layer of artificial neurons (aka perceptrons) that take input values, apply weights, and generate an output. The perceptron is typically used for linearly separable data, where it learns to classify inputs into two categories based on a decision boundary. It finds applications in pattern recognition, image classification, and linear regression

Core Concepts 1

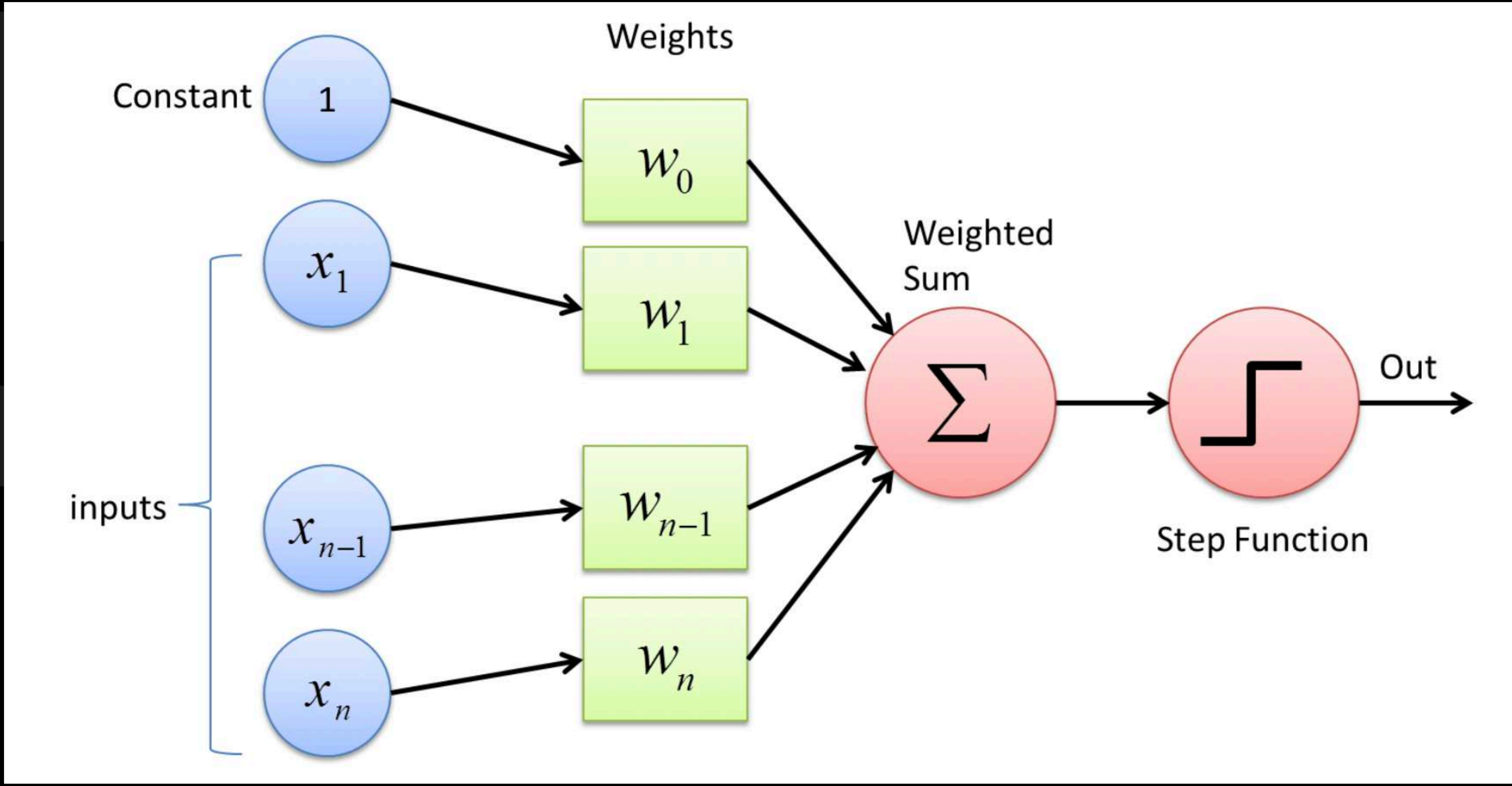


Fig.5 Scheme of a perceptron

Core Concepts 1

Linear Regression: It can be employed for solving linear regression problems, where the goal is to predict a continuous output based on input features

Image Classification: It can be used for binary image classification tasks, such as identifying whether an image contains a specific object

Limited to linear separability: It can only learn linear decision boundary

Lack of depth: It cannot learn complex hierarchical representations

Core Concepts 1

Features: the variables or attributes used to represent data. For example, when analysing image data, features might be pixels or image descriptors

Target: the value being sought to predict or classify in the data. For instance, in a classification problem, the target is the category to which an observation belongs

Weight: the value assigned to each feature or neuron in the NN during the training. Weights influence the importance of features or the strength of connections between neurons and are adjusted to achieve accurate predictions or to learn from input data

Core Concepts 1

Outliers: the data points that significantly deviate from the rest of the dataset. They are often considered as data that could negatively impact the model if not handled properly

Hyperparameters: the settings or configurations that are set prior to training a ML model and are not learned from the data. They control aspects of the model's behaviour, such as learning rate, batch size, or the number of hidden layers, and can significantly impact the model's performance

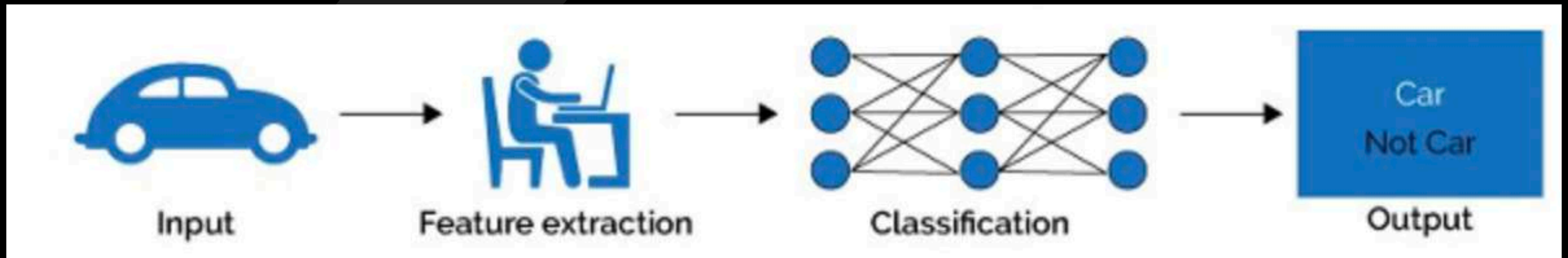
DO NOT mix up “hyperparameters” with “parameters” (weights)

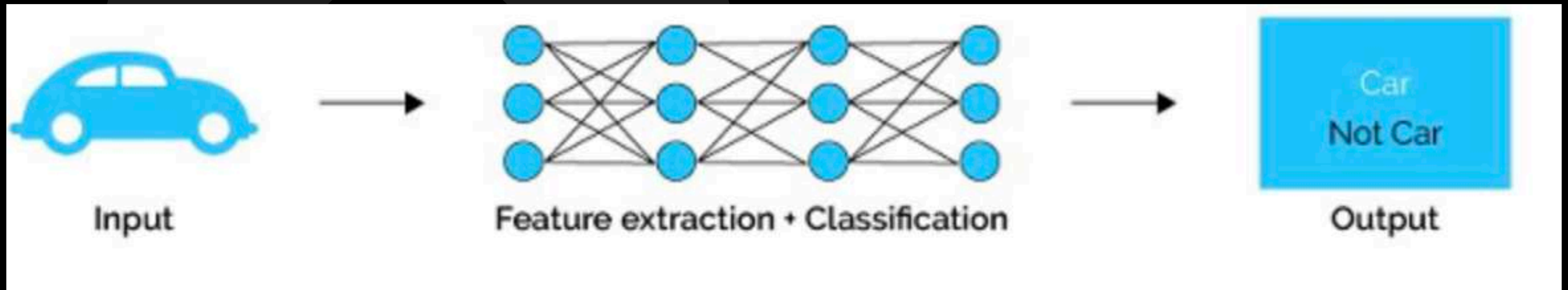
Types of NNs in Deep Learning 1

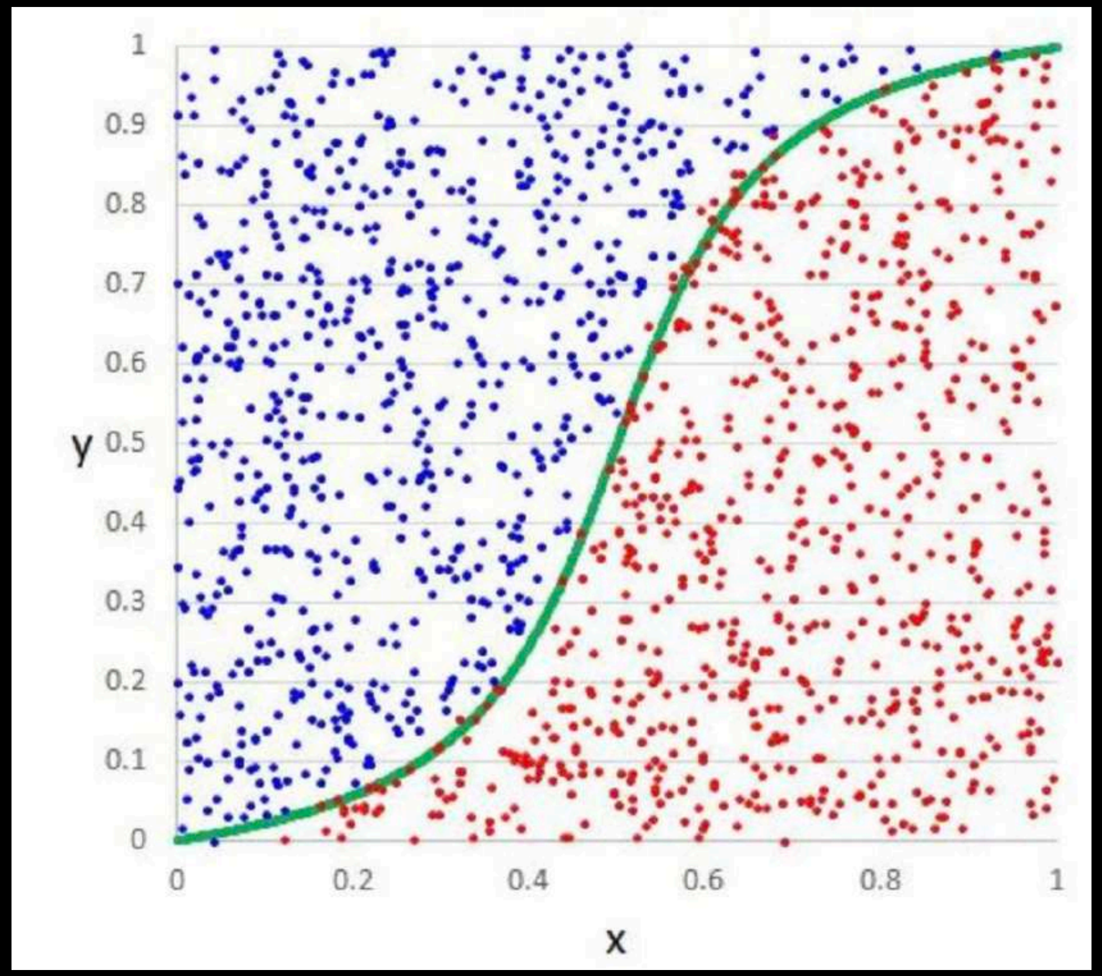
Feature Engineering: NNs can automatically learn the most relevant representations directly from the input data. This means that it's not always necessary to manually extract features or engineer data in a complex way, as required by traditional ML

Decision Boundary: More capable to efficiently capture complex, non-linear decision boundaries, making it well-suited for high-dimensional data and intricate problem-solving

Types of NNs in Deep Learning 1







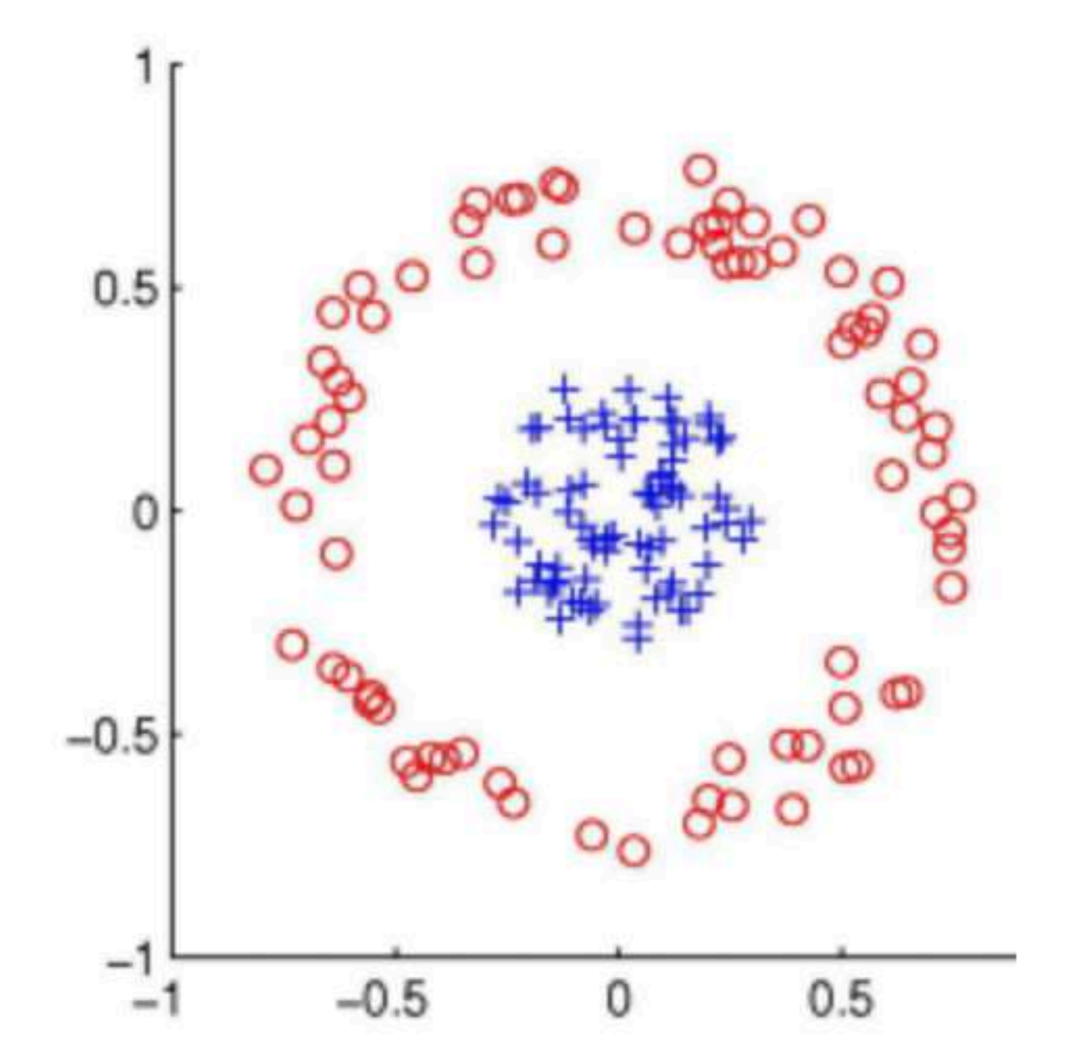


Fig.3 On top: ML vs DL workflow for image recognition

Fig.4 On the right: sigmoid vs. complex decision boundary

Types of NNs in Deep Learning 1

Artificial Neural Network (ANN): It is a group of multiple perceptrons/neurons at each layer. ANN is also known as a **Feed-Forward Neural network** because inputs are processed only in the forward direction

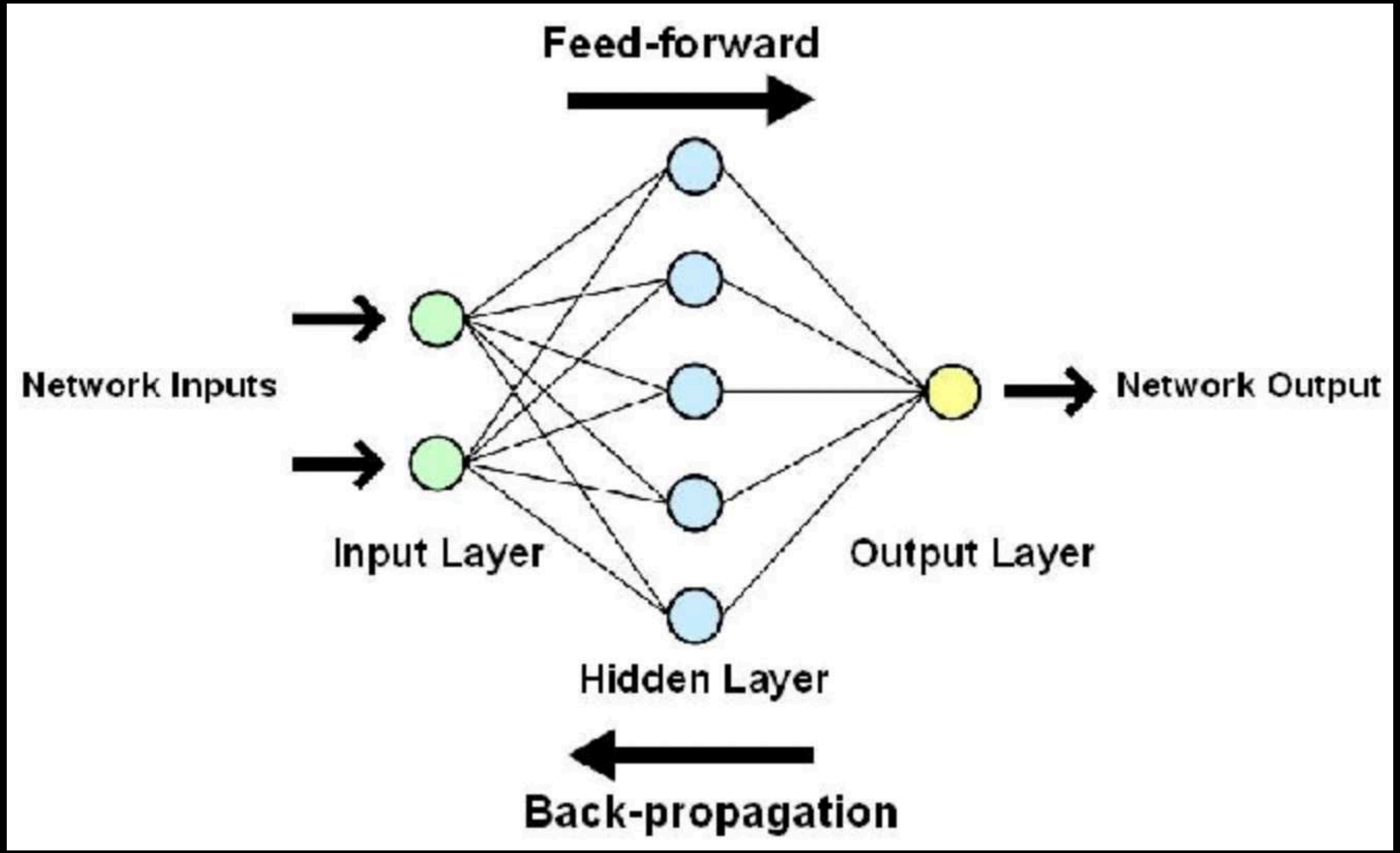


Fig.6 Scheme of a fully connected ANN with one hidden layer and back-propagation

Types of NNs in Deep Learning 1

1. In ANN each connection between nodes is associated with a weight, and each node typically applies an activation function to the weighted sum of its inputs
2. Back-propagation: calculates the gradients of the loss function with respect to the network's parameters, enabling the network to adjust its weights and biases to minimize prediction errors during training. This iterative process moves backward through the network, making it learn and improve its performance

Types of NNs in Deep Learning 1

ANN is capable of learning any nonlinear function. Hence, these networks are popularly known as Universal Function Approximators

Activation Functions are the main reason behind this capability. Introducing nonlinear properties to the network helps it to learn complex relationship between input and output

No sequential information, No spatial features of an image

Exploding and/or Vanishing Gradients

Types of NNs in Deep Learning 1

6

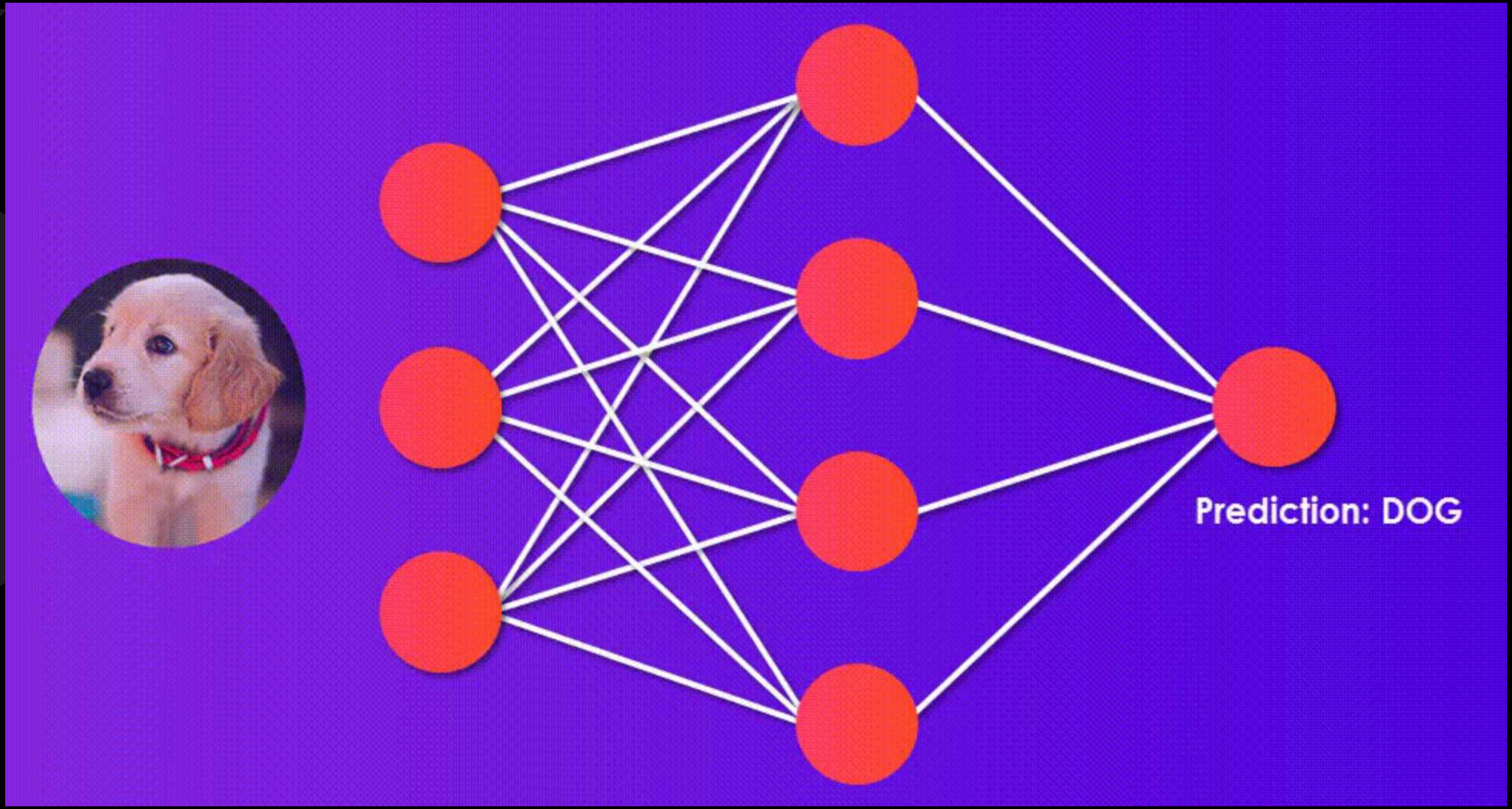


Fig.7 Considering an input image of 224*224 pixels, how many trainable parameters do we have in this ANN?

Types of NNs in Deep Learning 1

Recurrent Neural Network (RNN): what differentiate a RNN from an ANN is that RNN has a recurrent connection on the hidden layer(s). This looping constraint ensures that sequential information is captured in the input data

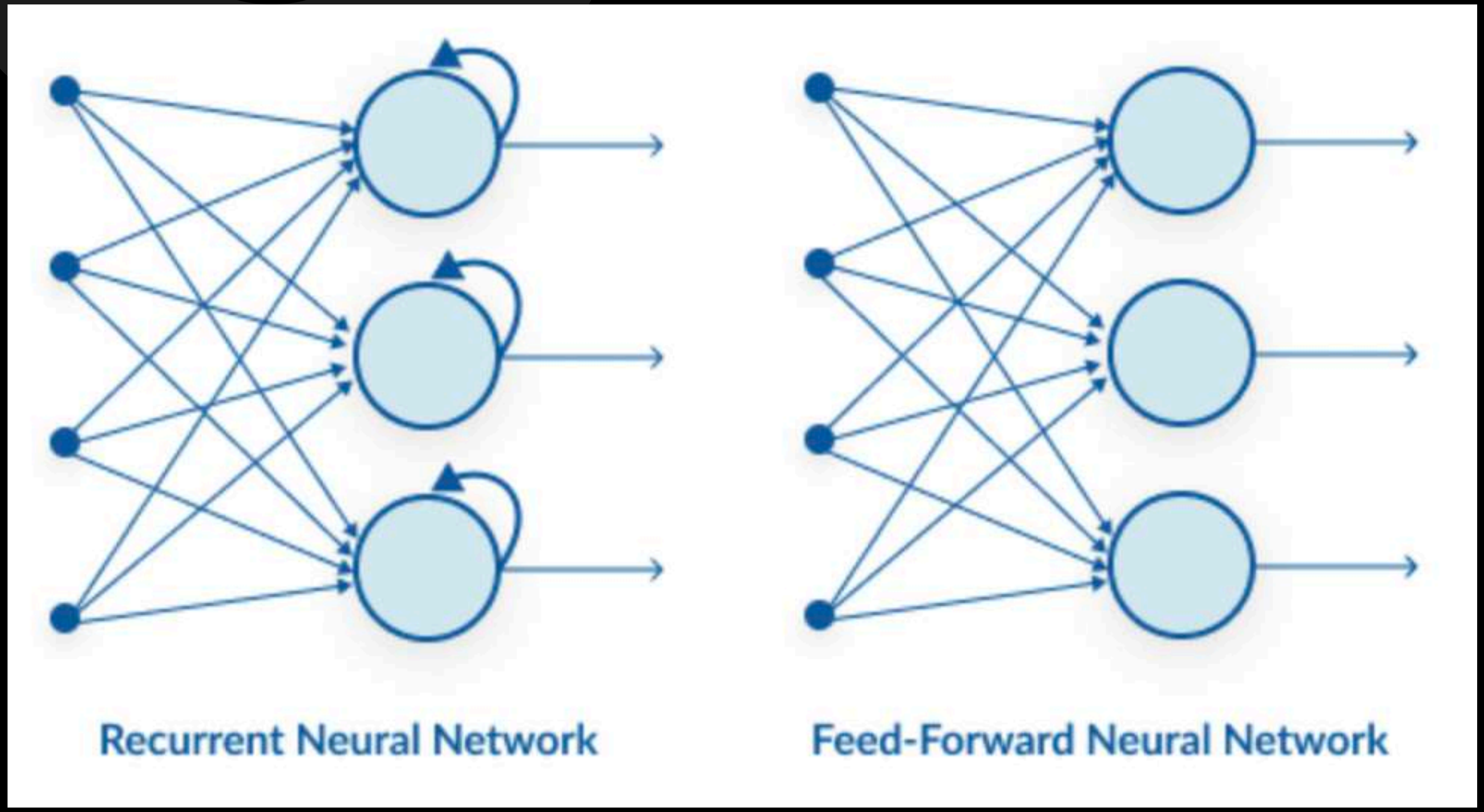


Fig.8 A looping constraint on the hidden layer of ANN turns to a RNN

Types of NNs in Deep Learning 1

RNN are good to solve problem related to: Time Series, Text, Audio Data. RNN captures the sequential information present in the input data i.e. dependency between the words in the text while making predictions

RNNs share the parameters across different time steps. This is popularly known as Parameter Sharing. This results in fewer parameters to train and decreases the computational cost

Types of NNs in Deep Learning 1

Parameter sharing is a key aspect that allows RNNs to handle variable-length data, as they can process sequences of different lengths without needing to train a new set of weights for each length

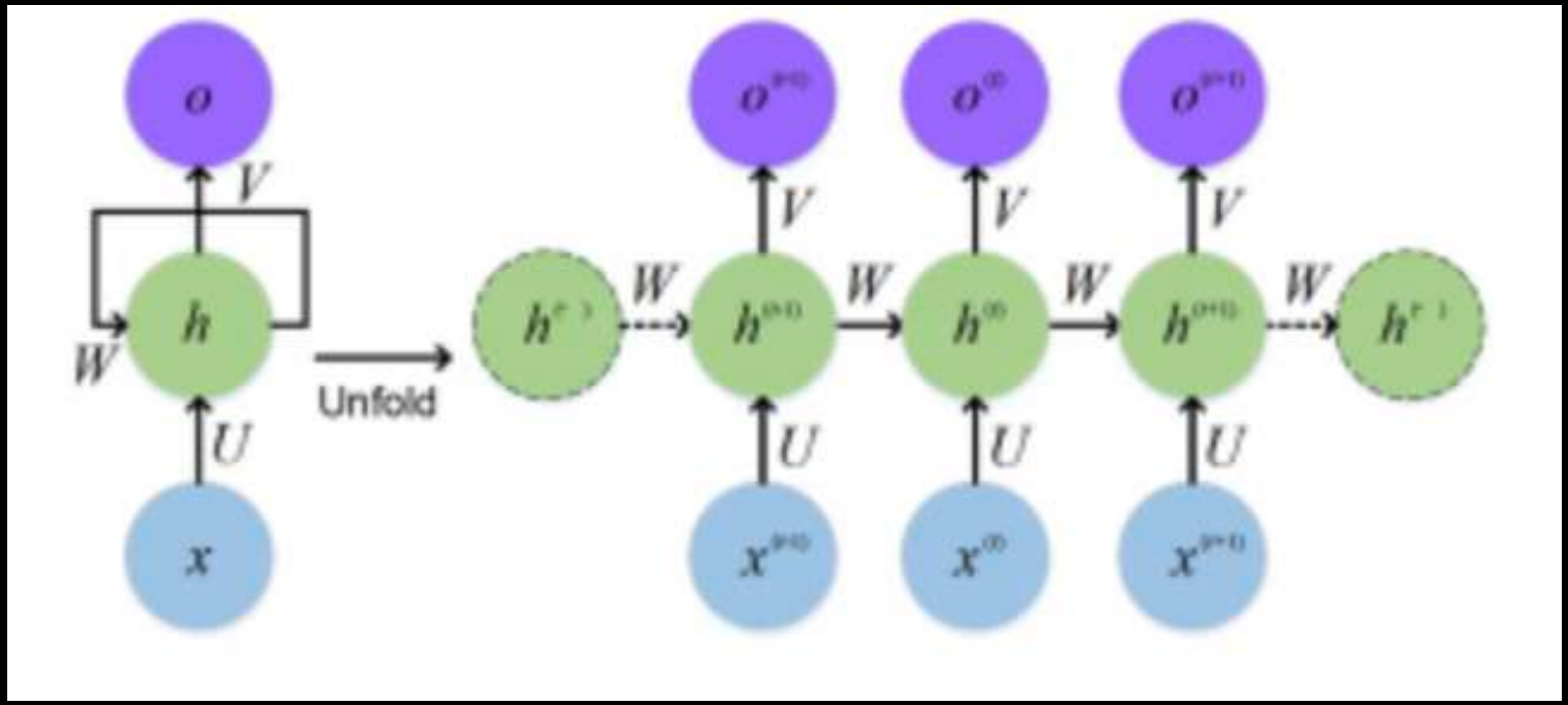


Fig.8 Parameter sharing representation: here 3 weight matrices U, V, W are shared across all time steps

Subsets of RNN architecture like **LSTM** and **GRU** have been developed to address the issue of exploding/vanishing gradients

Core Concepts 2

VARIANCE: refers to the model's sensitivity to small fluctuations or noise in the training data. High variance can lead to overfitting, where the model fits the training data too closely and doesn't generalize well to new data

BIAS: refers to the error introduced by approximating a real-world problem, which may be complex, by a simplified model. It can lead to underfitting and poor model performance

Core Concepts 2

Loss: is a measure of the error or the difference between the predicted values and the actual values in a machine learning model. It quantifies how well or poorly the model is performing

Accuracy: is a metric used to evaluate the performance of a classification model. It represents the proportion of correctly predicted instances out of the total instances in the dataset

Train-Test Split: training data is used to train the model, while the test data is used to evaluate its performance on unseen data.

Core Concepts 2

Overfitting: occurs when a model is excessively complex and learns to fit the training data too closely. As a result, it may perform poorly on new, unseen data

Underfitting: happens when a model is too simple to capture the underlying patterns in the data. It performs poorly on both the training data and new data