Deep Learning

Lesson 9—Other Forms of Deep Learning

© Simplilearn. All rights reserved.

Learning Objectives

- Elaborate on the functionality of an autoencoder and its various types.
- Discuss the working and uses of reinforcement learning.
- Describe the working of Generative-Adversarial Networks.

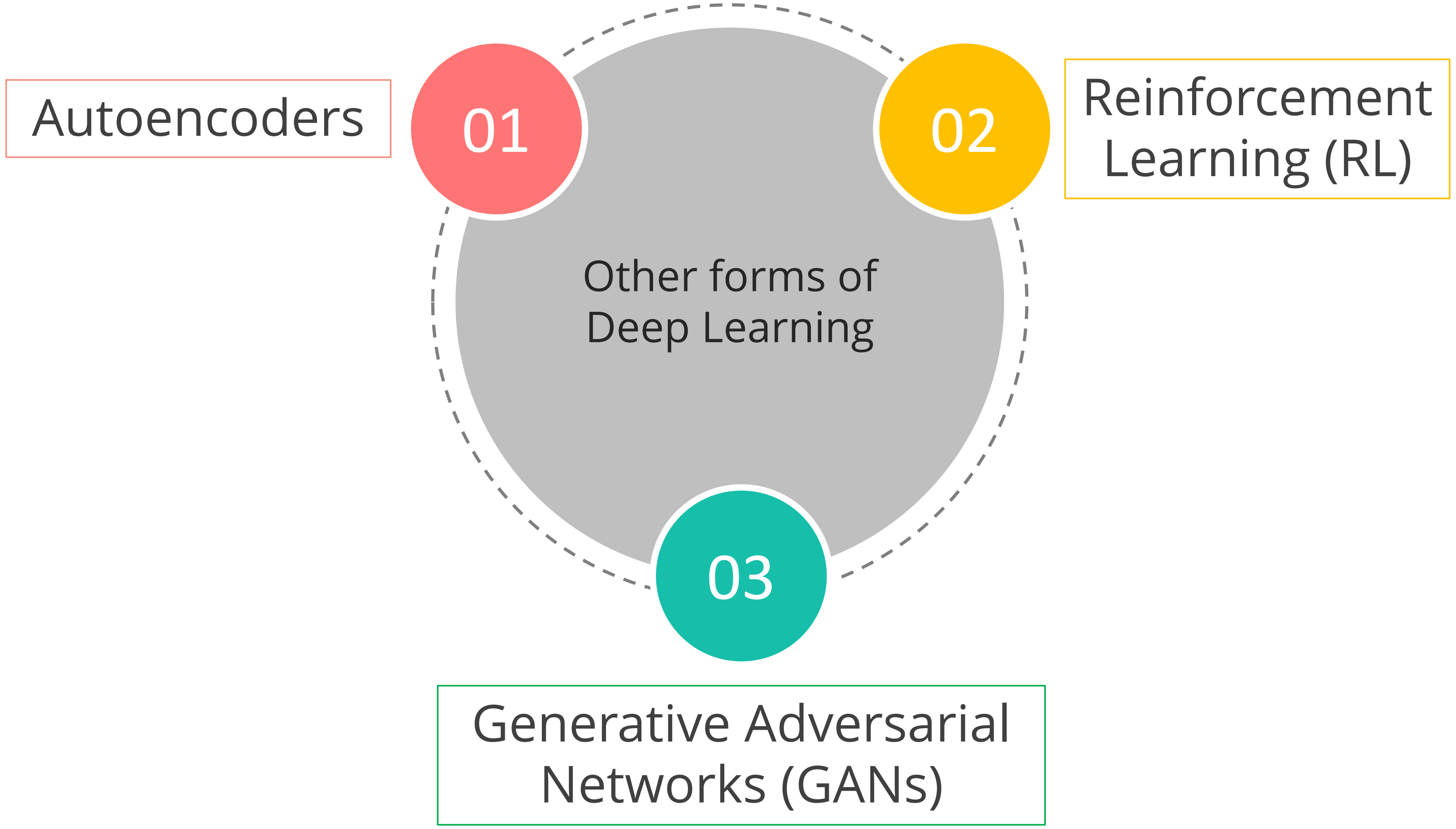
©Simplilearn. All rights reserved.

Other Forms of Deep Learning

Topic 1—Autoencoders

©Simplilearn. All rights reserved.

Other Forms of Deep Learning



©Simplilearn. All rights reserved.

Autoencoders

Autoencoders

An autoencoder is a special form of neural network which just copies its input to the output. It may also be called an identity function.

The idea is that in copying input to output, the neural net learns useful properties of the input.

The hidden layer *h* between input and output layer is called the **code**. This represents the information one is trying to learn about the input.

©Simplilearn. All rights reserved.

Autoencoders

LAYERS

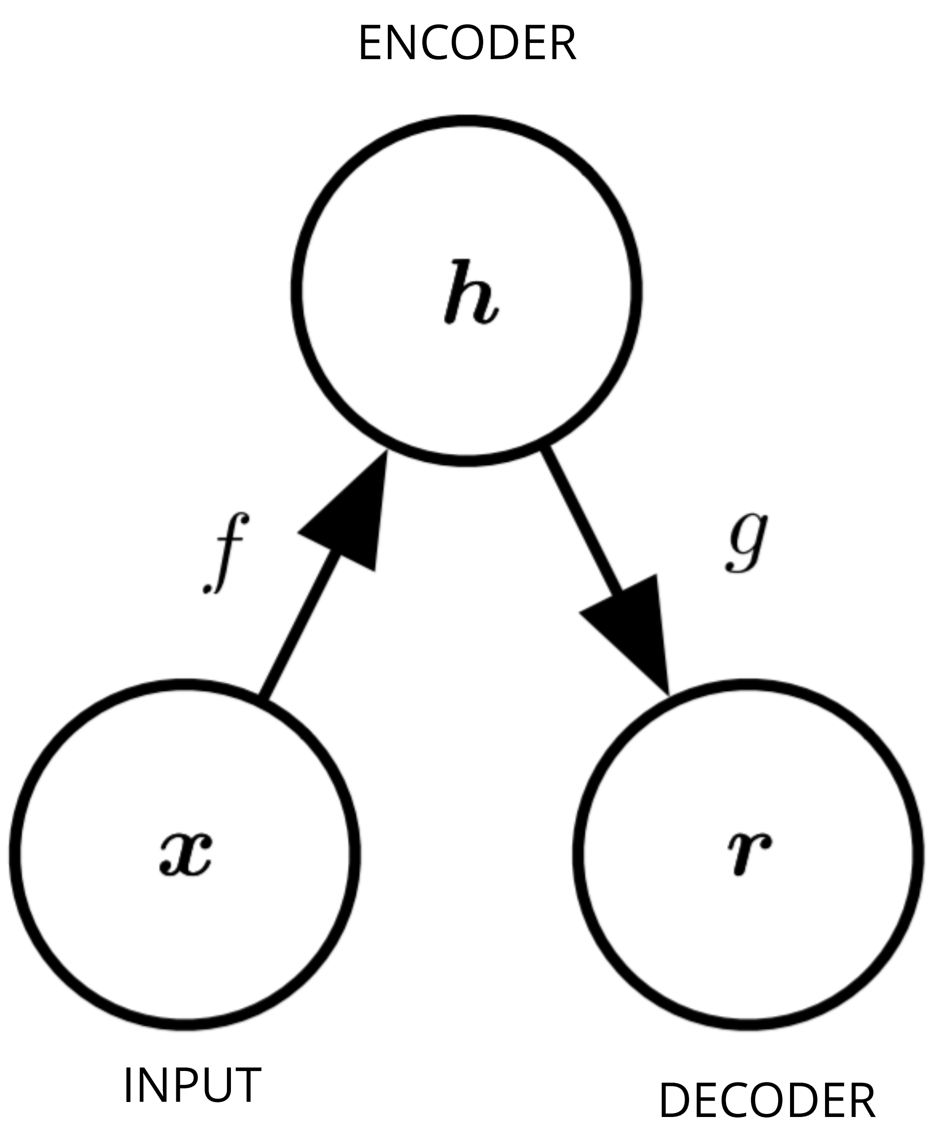
The network has two layers:

Encoder $h = f(x)$

- Encodes the data to a hidden layer
- Learns salient features of input

Decoder $r = g(h) = g(f(x))$

- Decodes the data back from hidden layer to reconstruction of the input
- "r" is the reconstruction of the input



©Simplilearn. All rights reserved.

Autoencoders

USES

- Traditionally autoencoders are used for :

Dimensionality reduction

To reduce the complexity of multi-dimensional data by transforming it to lesser dimensions

Feature learning

Learning important properties of the data

Generative modelling

Generating new data samples by having learnt the features of existing data samples via an autoencoder

©Simplilearn. All rights reserved.

Autoencoders

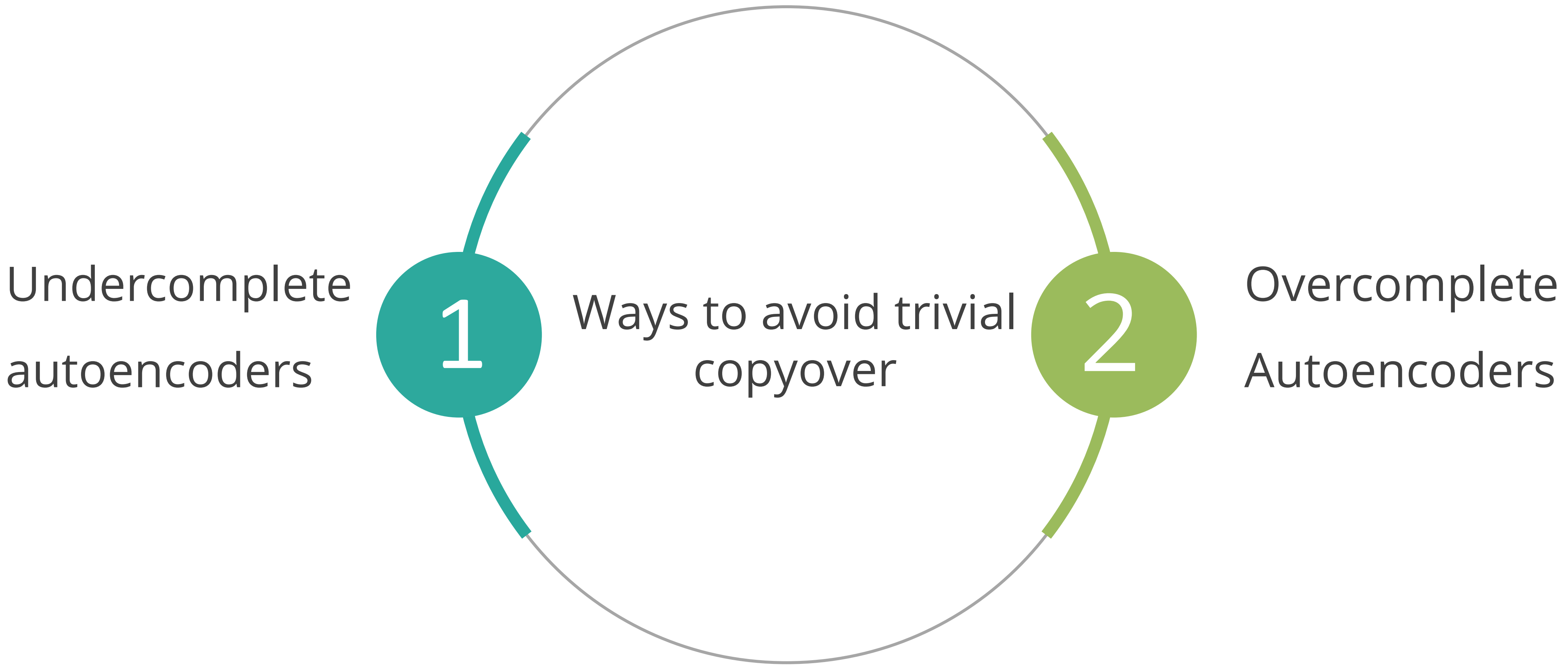
- Autoencoders are similar to feed-forward networks trained with gradient descent and backpropagation.
- It is important to avoid a trivial copyover from input to output. In case of an exact copy over, no useful information about the input will be obtained.

©Simplilearn. All rights reserved.

Autoencoders

AVOIDING TRIVIAL IDENTITY

- There are two ways to avoid trivial copyover :



©Simplilearn. All rights reserved.

Undercomplete Autoencoders

h has lower dimension than *x*. This means that h loses some dimensions compared to *x*, implying reduction in data intelligence going from *x* to *h*.

f or *g* has low capacity (e.g., linear *g*). This means that *f* or *g* is using a simplistic function and not a more complex function for less overfitted or less accurate transformation.

Forces latent layer *h* to learn the most salient features of the input.

Learning is achieved simply by minimizing the loss function: $L(x,g(f(x)))$

©Simplilearn. All rights reserved.

Undercomplete Autoencoders

This loss function represents the reconstruction error, and may be learned via mean squared error (MSE).

If the encoder and decoders are allowed too much capacity, then autoencoder might just learn to copy the input to output without deriving useful properties.

Applications include dimensionality reduction. It is a known fact that the lower dimensional data performs better in classification tasks, and consumes less memory and runtime.

©Simplilearn. All rights reserved.

Overcomplete Autoencoders

h has higher dimension than *x.*

Normally, this may cause it to trivially learn to copy input to output. This is solved by regularization: $L(x, g(h)) + \Omega(h)$, where $\Omega(h)$ is the regularizer.

Quick Recap: Regularization is a way to introduce some kind of small error in neural network to prevent overfit to input.

©Simplilearn. All rights reserved.

Overcomplete Autoencoders

TYPES

Sparse Autoencoders

Denoising Autoencoders

Types of Autoencoders

Contractive Autoencoders

©Simplilearn. All rights reserved.

Overcomplete Autoencoders

TYPES

- Sparse Autoencoders
- Denoising Autoencoders
- Contractive Autoencoders

- Helps learn latent or hidden aspects of the data.
- They are typically used to learn features for another task, such as classification.
- A type of sparse autoencoder called variational autoencoder can be used for generative modelling e.g. produce new art or new text.

©Simplilearn. All rights reserved.

Overcomplete Autoencoders

TYPES

- Sparse Autoencoders
- Denoising Autoencoders
- Contractive Autoencoders

- This kind of autoencoder receives a corrupted data point as input and is trained to produce the original uncorrupted data point as output.
- Useful for noise removal in data, e.g. in images or language text.

©Simplilearn. All rights reserved.

Overcomplete Autoencoders

TYPES

- Sparse Autoencoders
- Denoising Autoencoders
- **Contractive Autoencoders**

- Adds a certain sparsity penalty (regularizer) to the loss function, which has the effect of changing the neighbourhood of the input data to a smaller neighbourhood in the output.
- This results in the output becoming invariant to slight changes in the input.
- Such functionality is useful for dimensionality reduction, generalizing the output, information retrieval eg: search queries etc.

©Simplilearn. All rights reserved.

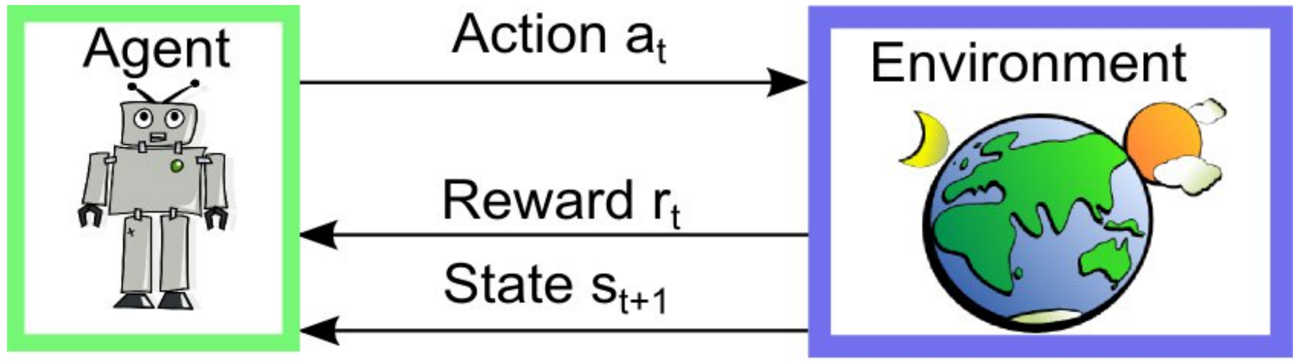
Other Forms of Deep Learning

Topic 2—Reinforcement Learning

©Simplilearn. All rights reserved.

Reinforcement Learning

- Reinforcement Learning teaches desirable behaviour by assigning rewards for the right set of actions.
- It trains a model to take decisions about future actions.
  - Eg: Robotics, gaming or driverless cars



Reinforcement Learning Setup

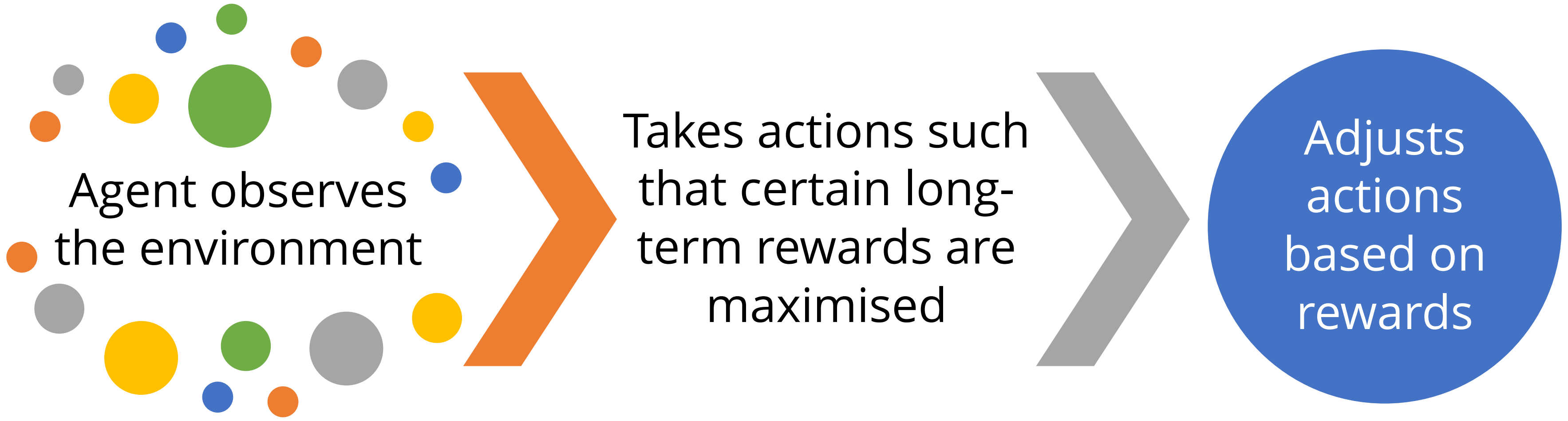
Image Source : http://www.ausy.tu-darmstadt.de

©Simplilearn. All rights reserved.

Reinforcement Learning

WORKING PRINCIPLE

- The neural network is trained by awarding a positive incentive for achieving desired behaviour. The agent in this case learns to do actions which maximise the cumulative rewards earned over a set of such action sequences.

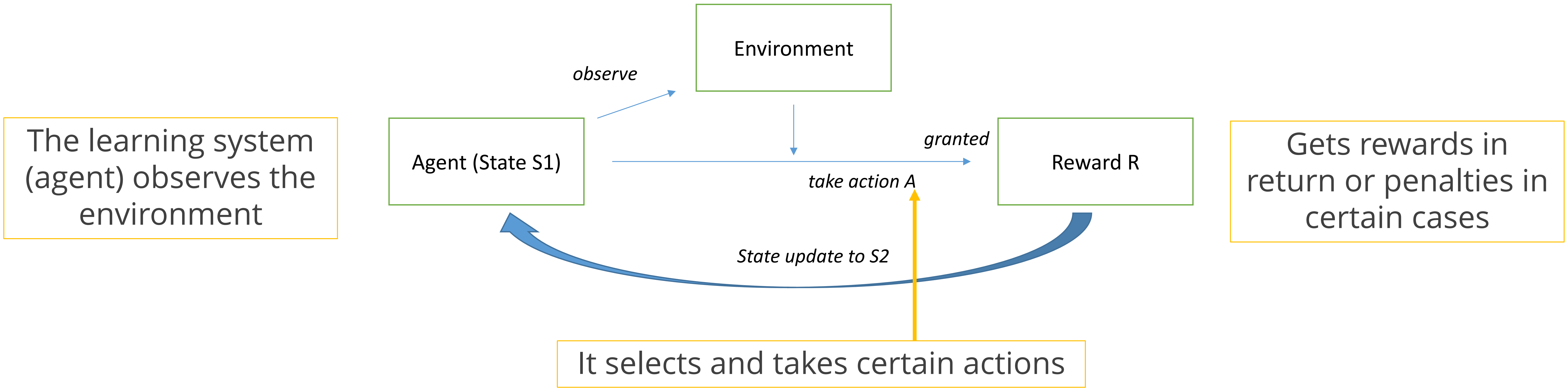


- Eg. Training a dog to fetch a ball. You cannot control the behaviour of dog but you can reward him with food every time he fetches the ball. Over time, the dog masters what behaviour will get him rewards.
- Used extensively in gaming applications as well as driverless cars.

©Simplilearn. All rights reserved.

Reinforcement Learning (Contd.)

WORKING PRINCIPLE



- The agent learns the strategy, or policy (choice of actions), which maximizes its rewards over time.
- Simple reward feedback is provided to the agent to learn its behaviour. This is called a reinforcement signal.

©Simplilearn. All rights reserved.

Reinforcement Learning (Contd.)

WORKING PRINCIPLE

- The agent interacts with the environment.

S = set of states the environment is in

A = set of actions that the agent can take

- Each time the agent performs action “a” in state “s”, it receives a real-valued reward “r”.

- The agent’s task is to learn a control policy π : S => A, that maximizes the expected sum of these rewards, with future rewards discounted exponentially by their delay.

- The immediate rewards have a higher weightage compared to the future rewards.

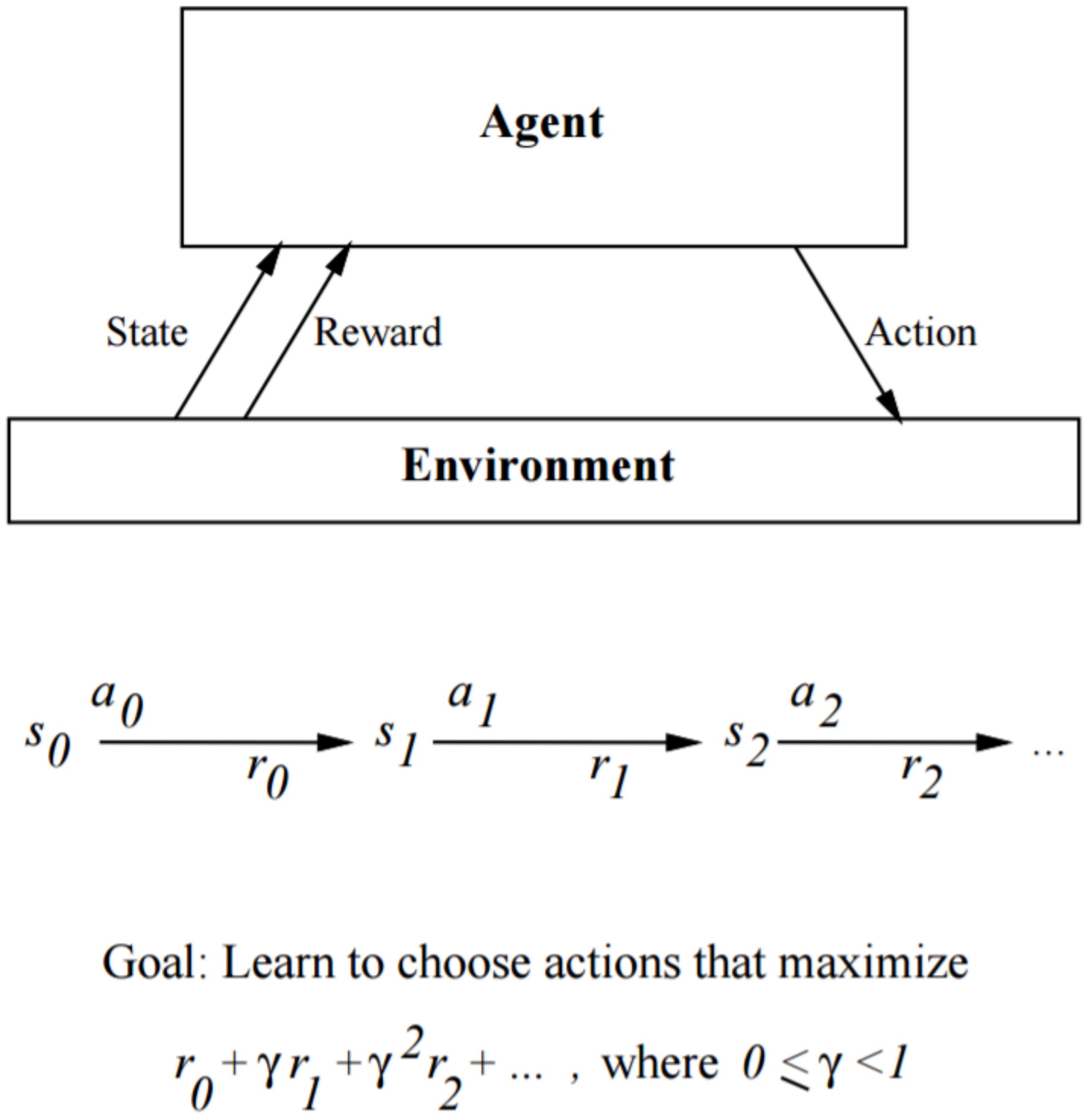


Image Source : “Deep Learning” book by Ian Goodfellow

©Simplilearn. All rights reserved.

Reinforcement Learning

USE CASES

- Typical RL environments like gaming have an advantage that huge amount of data can be generated easily unlike in real world cases of supervised learning, where a lot of data is needed from the real world.
- Google used RL systems to achieve 40% reduction in the amount for electricity for cooling in their data centres.

©Simplilearn. All rights reserved.

Other Forms of Deep Learning

Topic 3—Generative Adversarial Networks (GANs)

©Simplilearn. All rights reserved.

Generative Adversarial Networks

GANs are generative models used to learn new generation capabilities.

They are based on ‘Game Theory Scenario’ in which the generator network must compete against an adversary.

©Simplilearn. All rights reserved.

Generative Adversarial Networks (Contd.)

Generative Adversarial Networks (GANs)

- The idea of GANs is to produce output that contains new data similar to training data
- They do so by learning the natural features of the training data set
- A gradual learning enables this network to produce output which looks similar to training data
- GANs represent unsupervised learning as there does not exist an explicit desired output. It deals with unlabelled data

Example: Millions of real images are taken as training data in a GAN to produce images similar to training data, in other words realistic images

©Simplilearn. All rights reserved.

Generative Adversarial Networks

ADVANTAGES

Images produced by GAN could be sharper than the training images

The neural network is learning how the real world looks like

Used to produce samples of photorealistic images for the purposes of visualizing new interior/industrial design, shoes, bags and clothing items or items for computer games' scenes

©Simplilearn. All rights reserved.

How Do GANs Work

1

- In addition to a generative layer in the network (G), there is an additional neural network, a discriminator network D that tries to classify if an input image is real or generated.
- For example, around 200 generated images and 200 real images are fed into the discriminator and trained to classify them as “real images” or “generated images”.
- The discriminator network is usually a standard convolutional neural network.

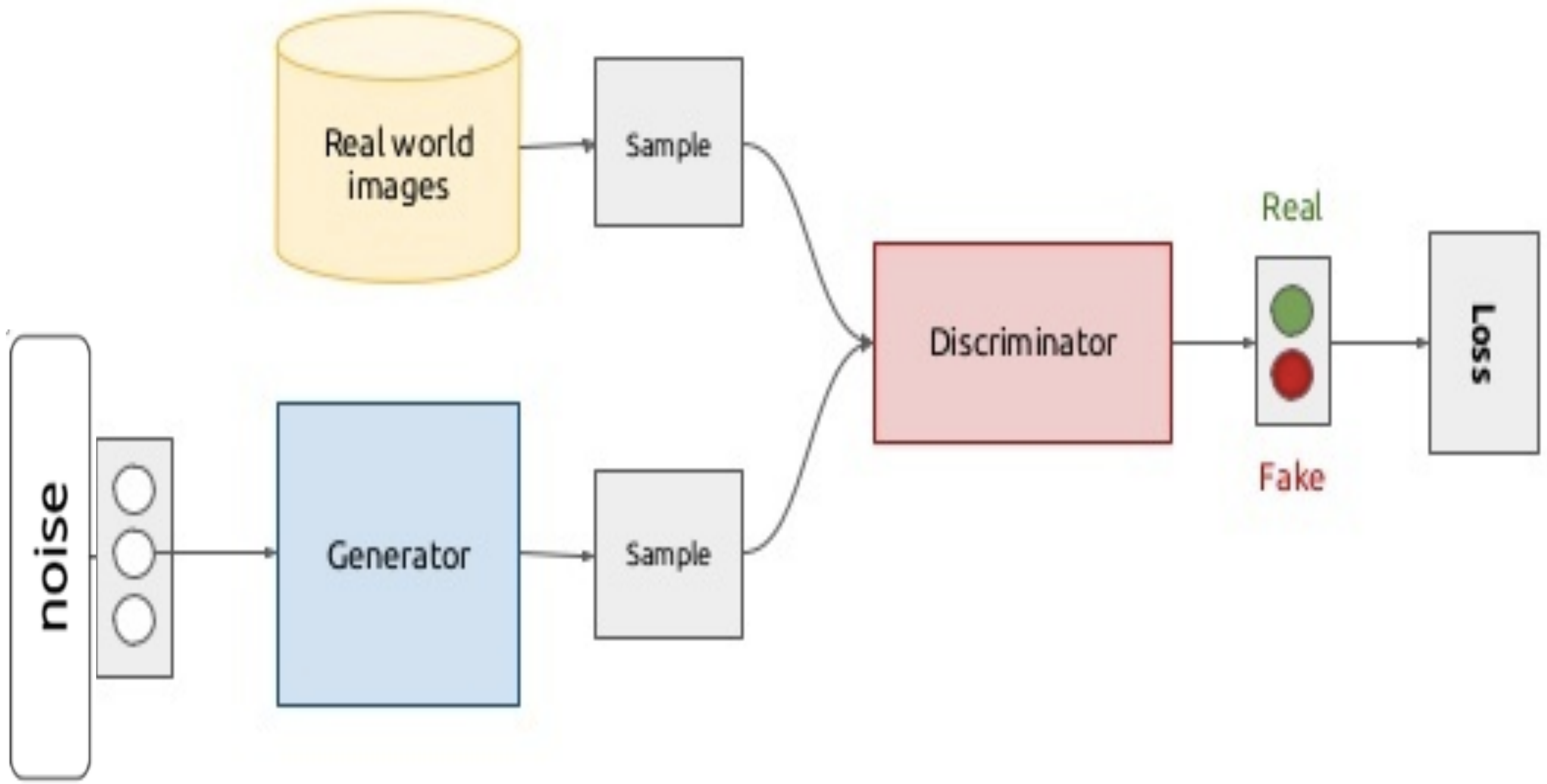


Image Source : https://wiki.tum.de

©Simplilearn. All rights reserved.

How Do GANs Work (Contd.)

2

- Backpropagation is done both through discriminator and generator to find how the generator parameters can be changed to make its 200 images slight more confusing for the discriminator.

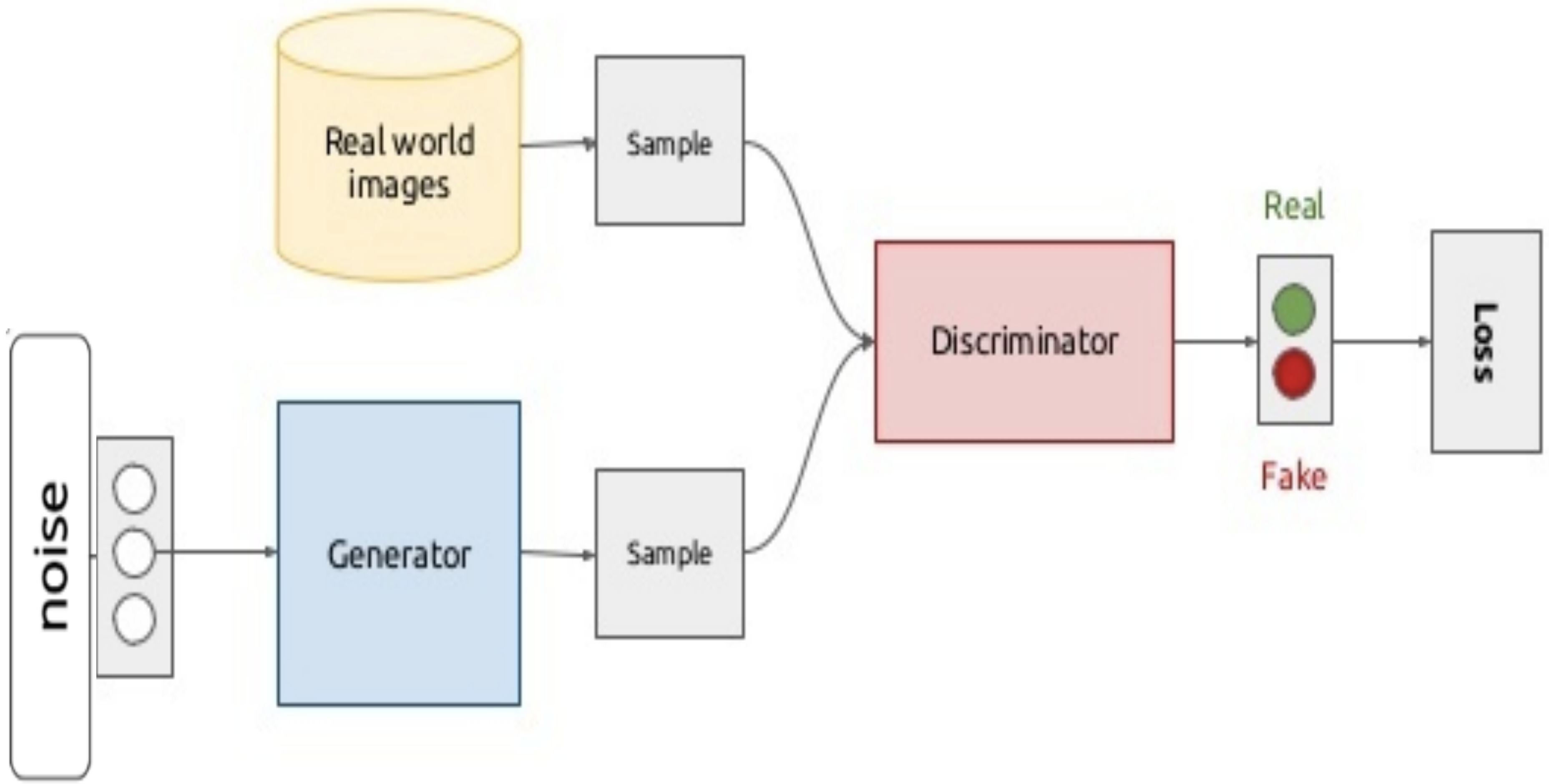


Image Source : https://wiki.tum.de

©Simplilearn. All rights reserved.

How Do GANs Work (Contd.)

3

- In effect, the generator G is trying to fool the discriminator D by generating fake images which are closer to real images, and discriminator is always trying to distinguish them.

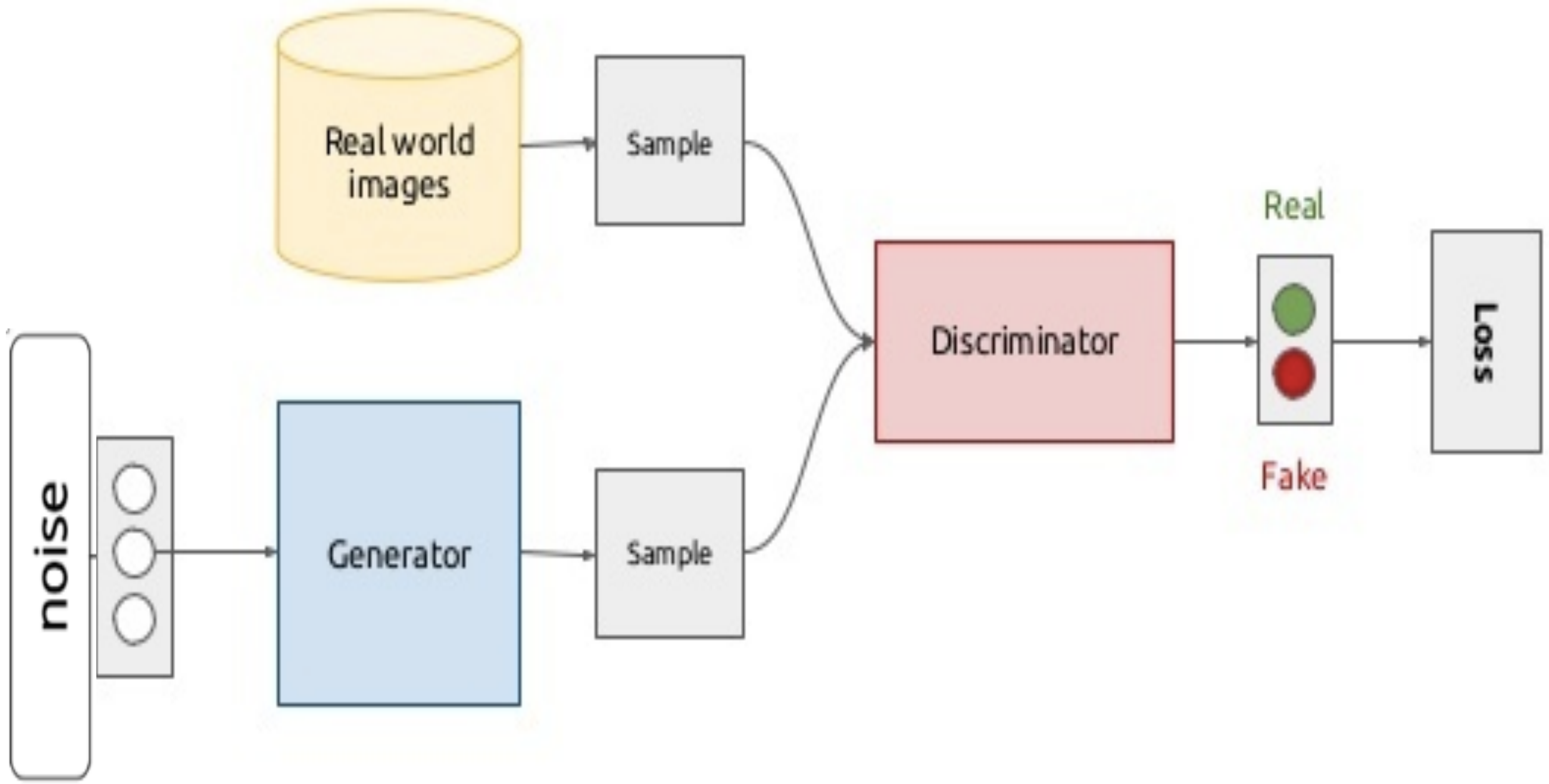


Image Source : https://wiki.tum.de

©Simplilearn. All rights reserved.

How Do GANs Work (Contd.)

4

- The training procedure for G is to maximize the probability of D making a mistake.

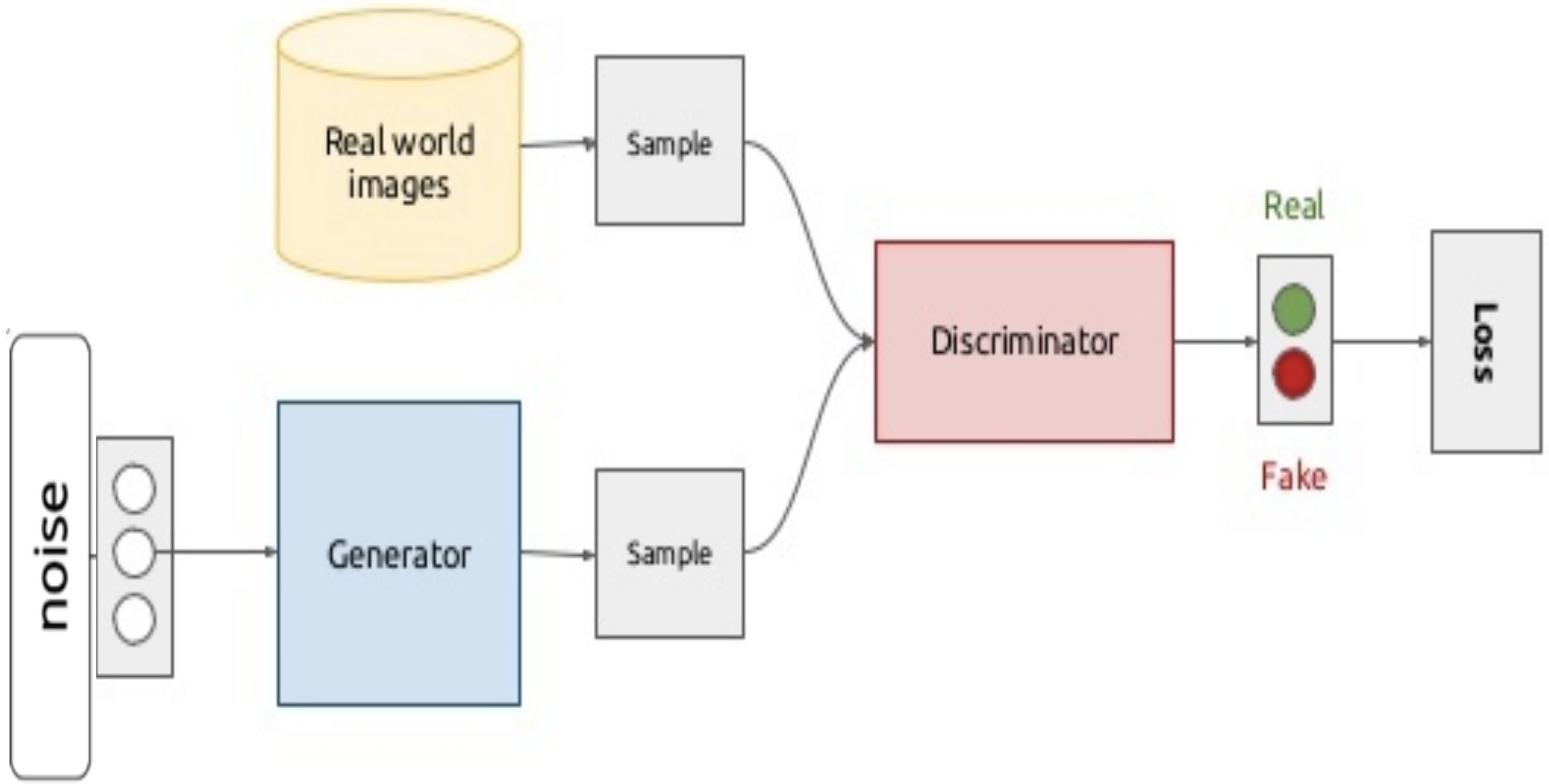


Image Source : https://wiki.tum.de

©Simplilearn. All rights reserved.

simplilearn

How Do GANs Work (Contd.)

5

- In the end, the generator will output images that are indistinguishable from real images for the discriminator.

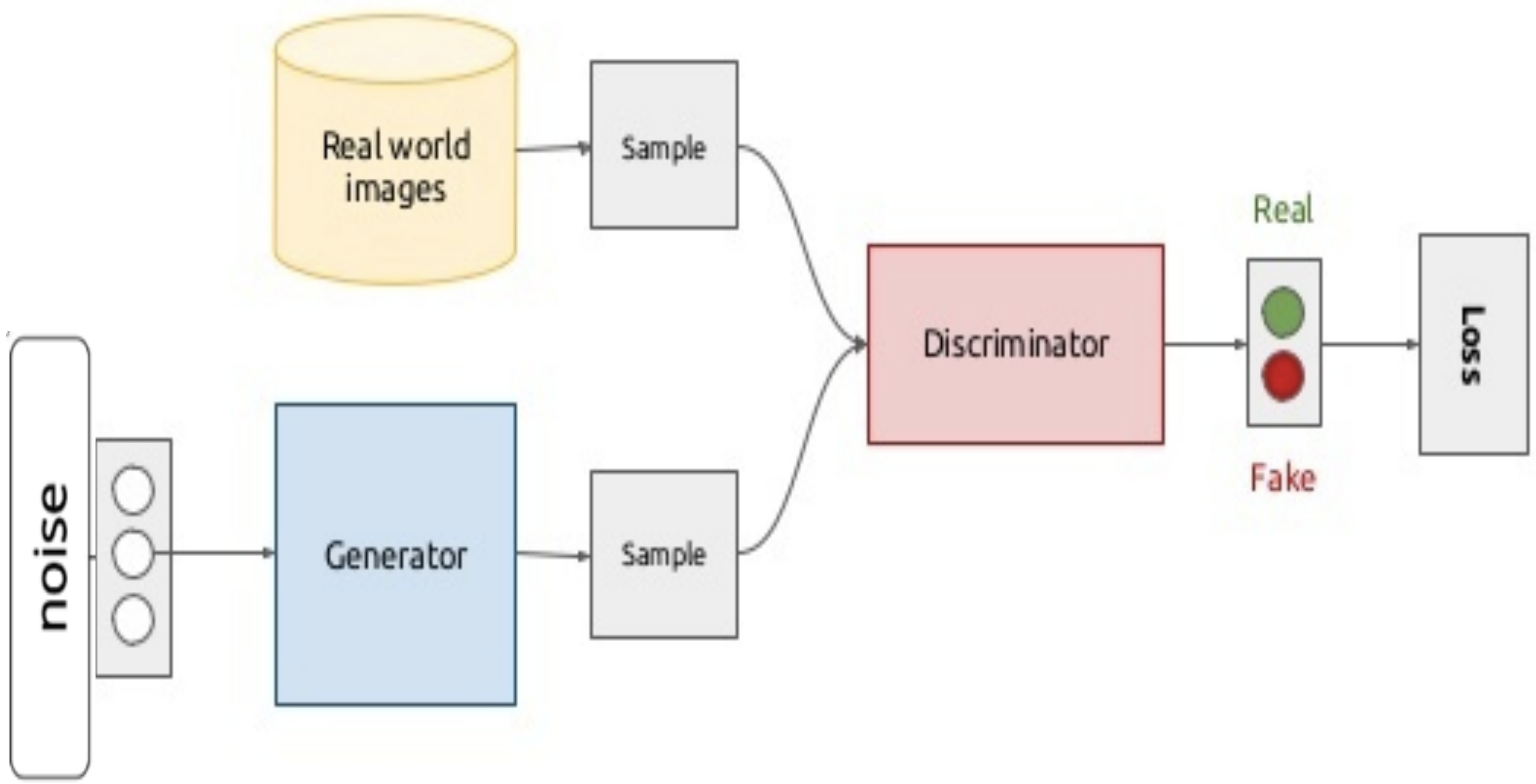


Image Source : https://wiki.tum.de

©Simplilearn. All rights reserved.

Key Takeaways

- Deep Learning includes other advanced techniques like Autoencoders, Reinforcement Learning and Generative Adversarial Networks.
- An Autoencoder represents an identity function where the input is copied to the output, but in such a manner that the copyover is approximate and not precise. This makes the model learn the most important features of the input.
- Reinforcement Learning teaches agents what kind of actions are more desirable. This is achieved by incentivising the agents with positive rewards for performing those actions.
- The generator produces samples which are close to real samples, and the discriminator tries to distinguish real samples from generated samples.
- Eventually, the generator learns to produce samples which look real to the discriminator. This helps in generating new art or new text.

©Simplilearn. All rights reserved.

This concludes “Other Forms of Deep Learning .”

©Simplilearn. All rights reserved