Learning Paradigms – COMP4423 Computer Vision

Xiaoyong Wei (魏驍勇)

x1wei@polyu.edu.hk

Department of Computing
電子計算學系

THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

Opening Minds • Shaping the Future
啟迪思維 • 成就未來

Review of Assignment 2, Project Ideas
3D Reconstruction,
Image Generations,
Applications

Question 5: Multiple Choice

What additional resources or support would you like to see in future lectures?

	Percent Answered
More guest speakers from the industry	26.315%
Additional tutorials or workshops	36.842%
Access to relevant research papers and articles	5.263%
Collaboration opportunities with other students	31.578%
Other (please specify below)	0.00%
Unanswered	0.00%

Group Project April 1 – April 30

Other comments or suggestions?

Unanswered Responses

11

TUT 11

Given Answers

If possible, the deadline of different assignments or project could be announced early and give more buffer time to due with the problem, it could give more flexibility and time for students to do their assignment or learning with the projects.

I would be interested in taking a deeper look into the inner workings. That is to say, things like TorchNoGrad vs TorchGrad as well as optimizations and scalability at the design phase.

I would also be interested in analyzing things like yolo v7 and other such deep frameworks, as from what I have seen the way they approach feature extraction is rather different.

The only other things that come to mind would be more Q&A opportunities, otherwise, this is legitimately one of, if not the best courses in the department of computing.

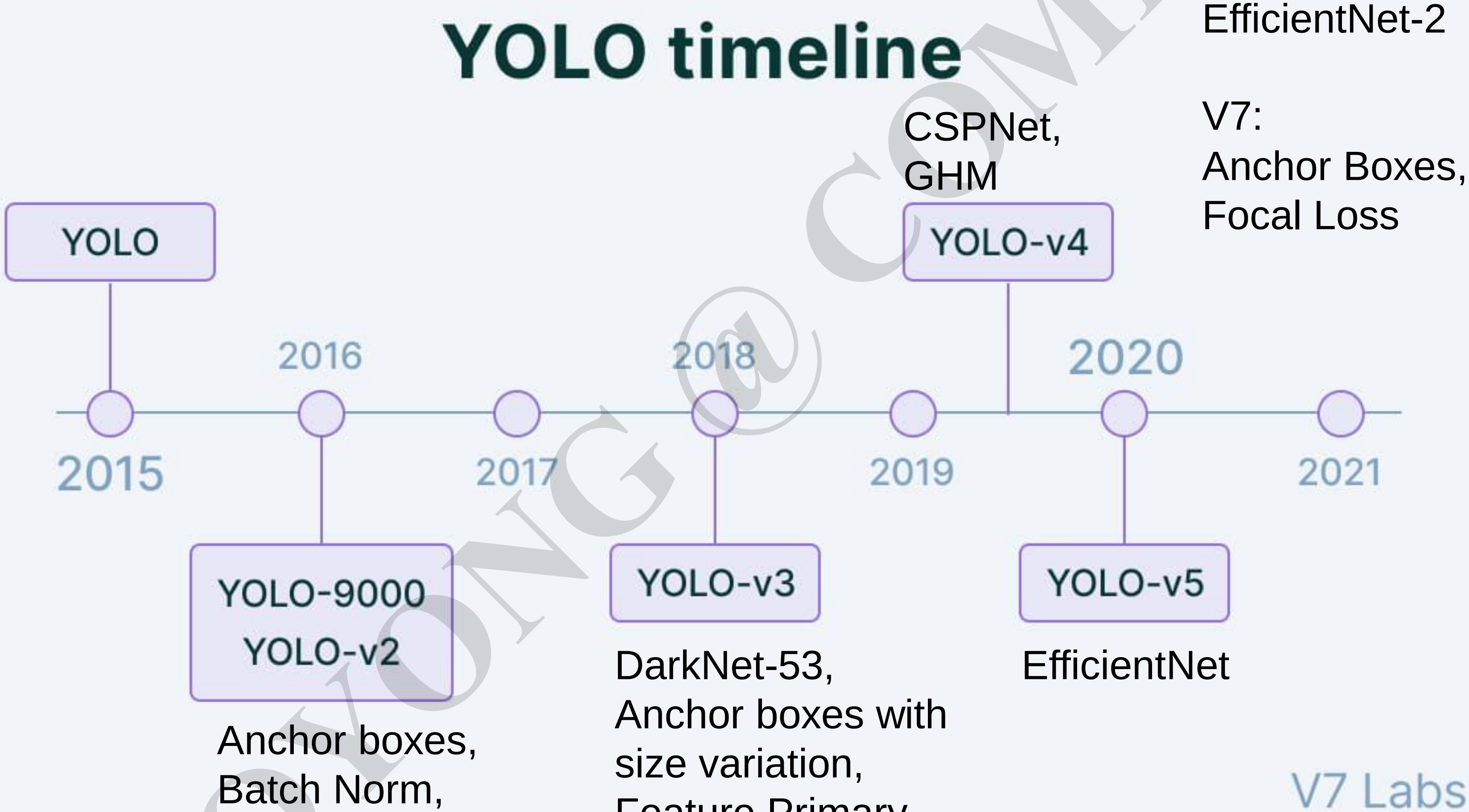
/

nil

N/A

More time for assignments will be better

Troublesome configuration in COMP lab computers for doing assignments and tutrials



CSPNet,
GHM

V6:
EfficientNet-2

V7:
Anchor Boxes,
Focal Loss

Anchor boxes,
Batch Norm,
New Loss

DarkNet-53,
Anchor boxes with size variation,
Feature Primary

EfficientNet

You may have heard the term "learning" a lot when people are talking about Computer Vision. However, the term is indeed "overused". Sometimes it stands for a specific model, while it may also refers to a learning framework

Let’s check out some examples

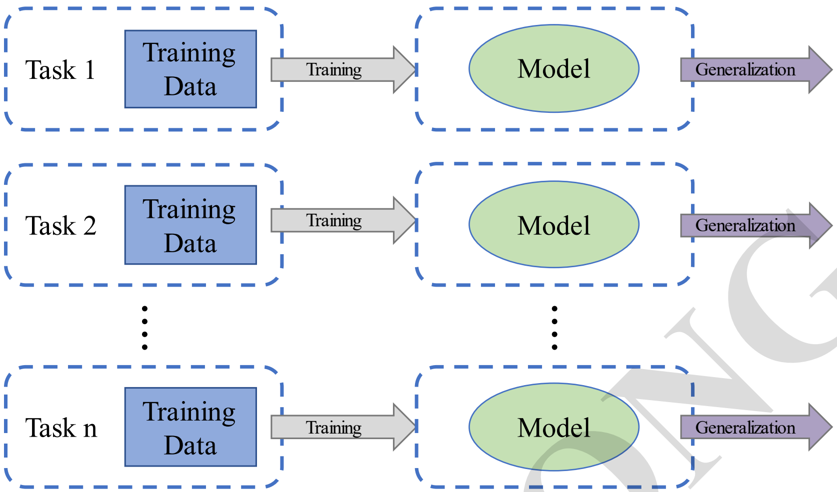
Outline

- >Multi-task learning
- >N-shot learning (Few-shot, Zero-shot)
- >Transfer learning, Metric learning, Meta-learning
- >Generative networks (VAE, GAN)
- >Reinforcement learning

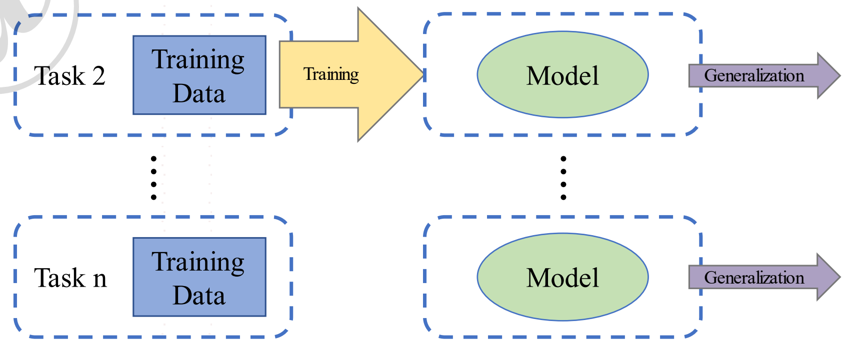
Multi-task Learning

Multitask learning (MTL) is a machine learning technique that aims at **improving the generalization performance** of a learning task by jointly learning multiple-related tasks. The key to a successful MTL is that **the tasks need to be related**. Here the "related" does not mean the tasks are similar. It means at some level of abstraction these tasks **share part of the representation**. If the tasks are indeed similar, learning them together can help transfer knowledge among tasks since it leverages training data more effectively.

Single vs. Multiple Task Learning

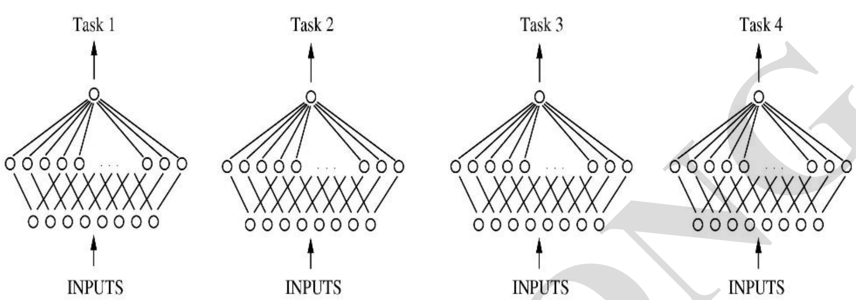


Single-task Learning

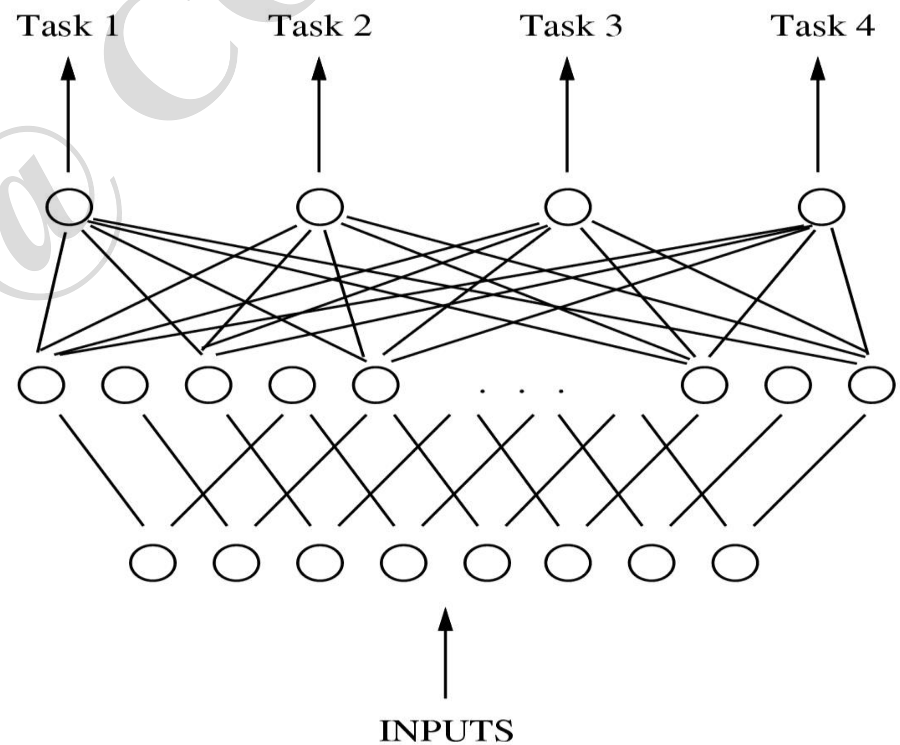


Multi-task Learning

Single vs. Multiple Task Learning

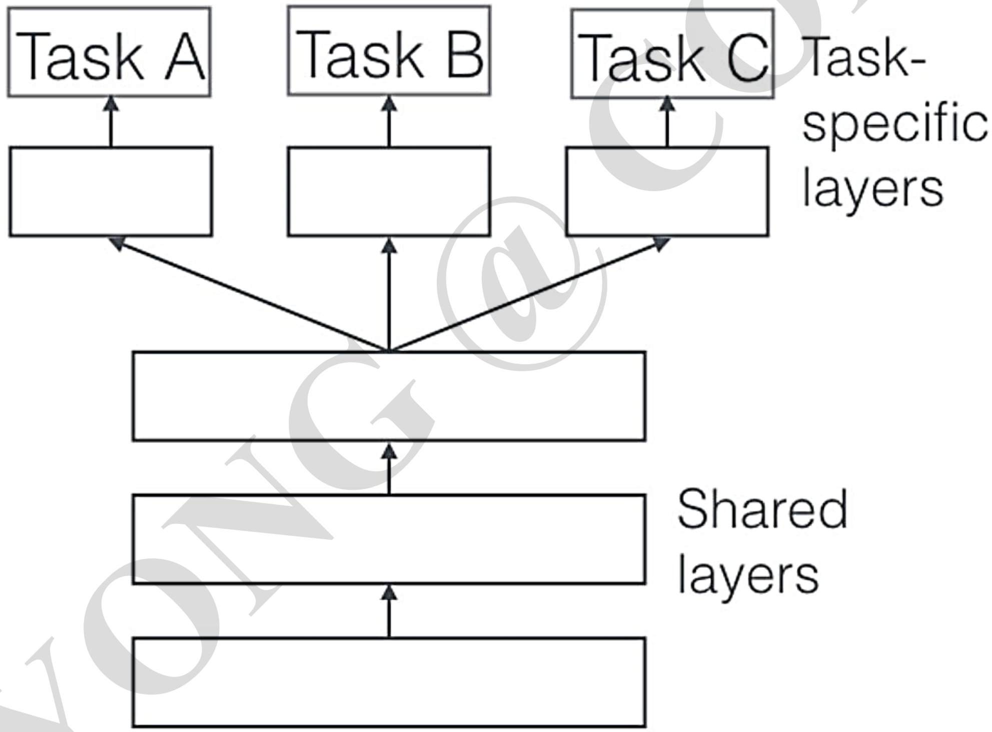


Single-task Learning



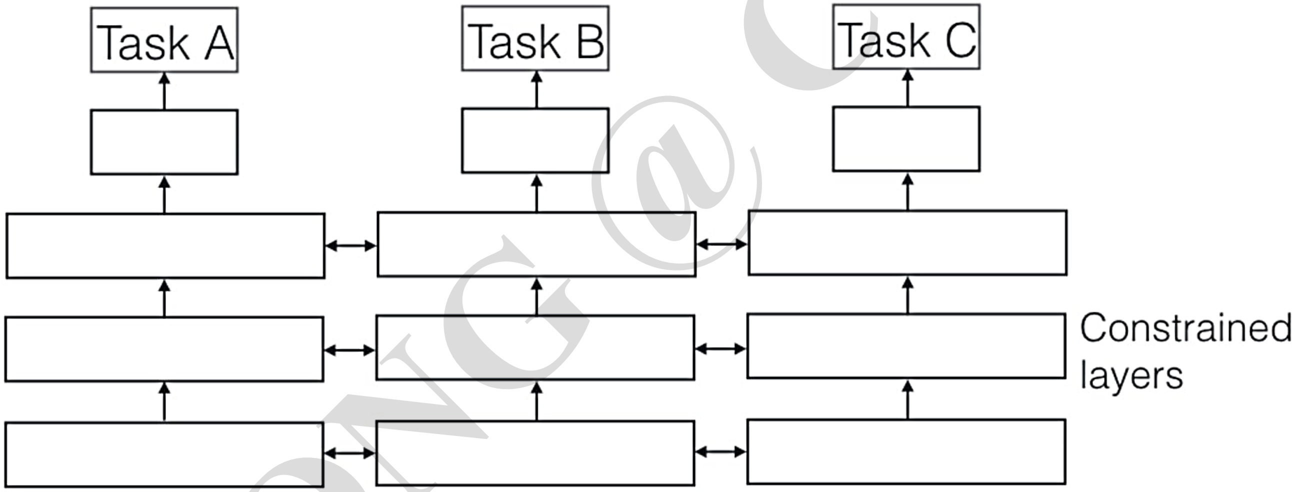
Multi-task Learning

Multiple Task Learning



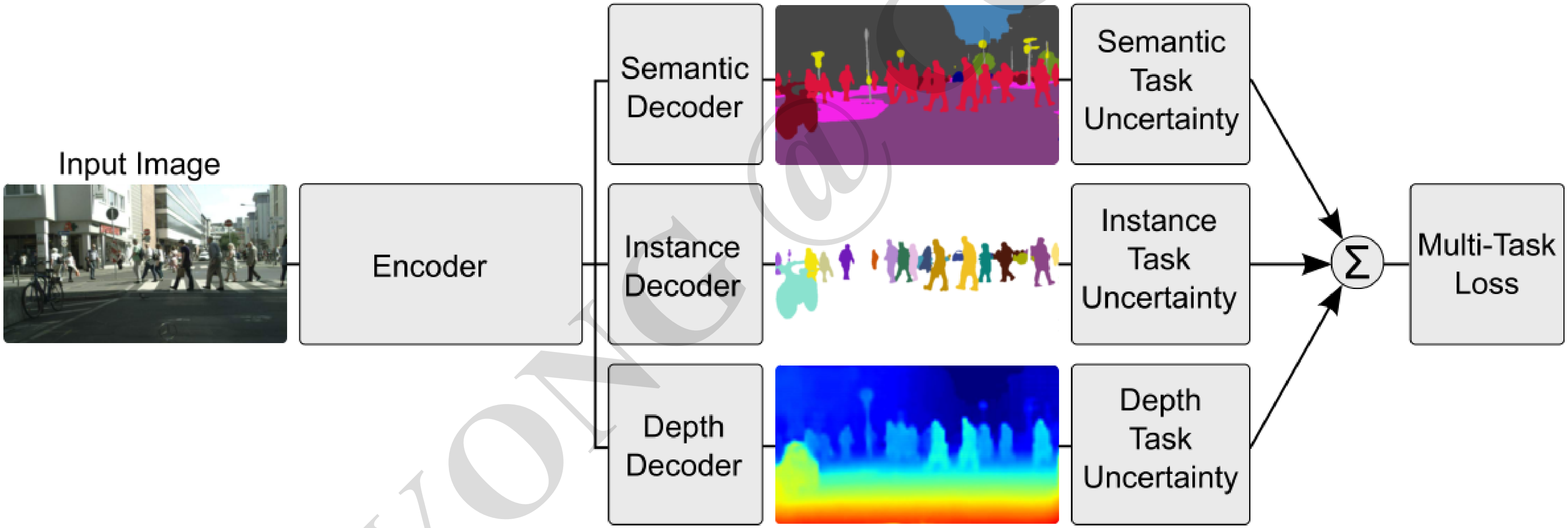
Hard Parameter Sharing

Multiple Task Learning



Soft Parameter Sharing

Multiple Task Learning



Uncertainty-based loss function weighting for multi-task learning (Kendall et al., 2017).

Deep models are hungry about data. However, in reality, we have only a limited amount of (labeled) data available

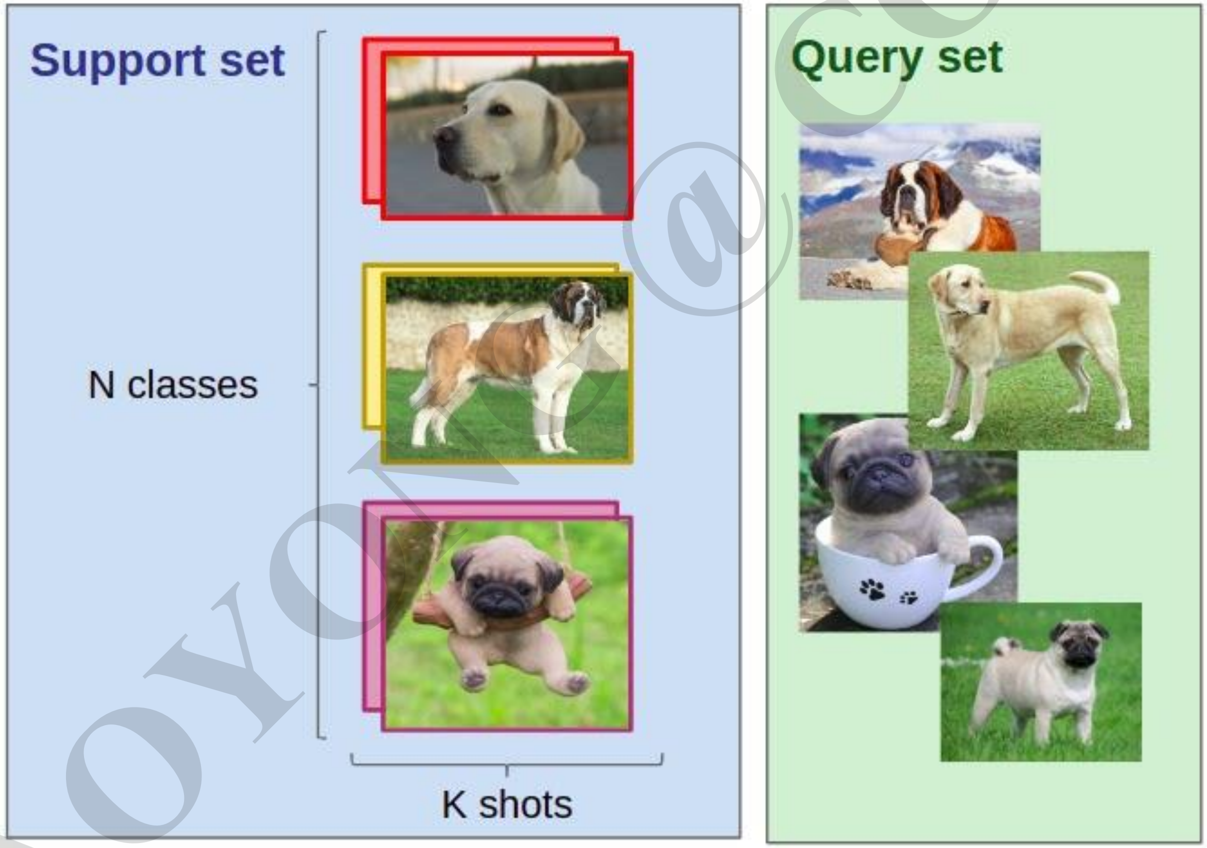
N-shot Learning

- To classify new data when you have only a few training samples with supervised information

- **Few-shot Learning** to classify classes with only a few examples
- **One-shot Learning** to classify classes with only one example
- **Zero-shot Learning**: to classify **unseen** classes without any examples

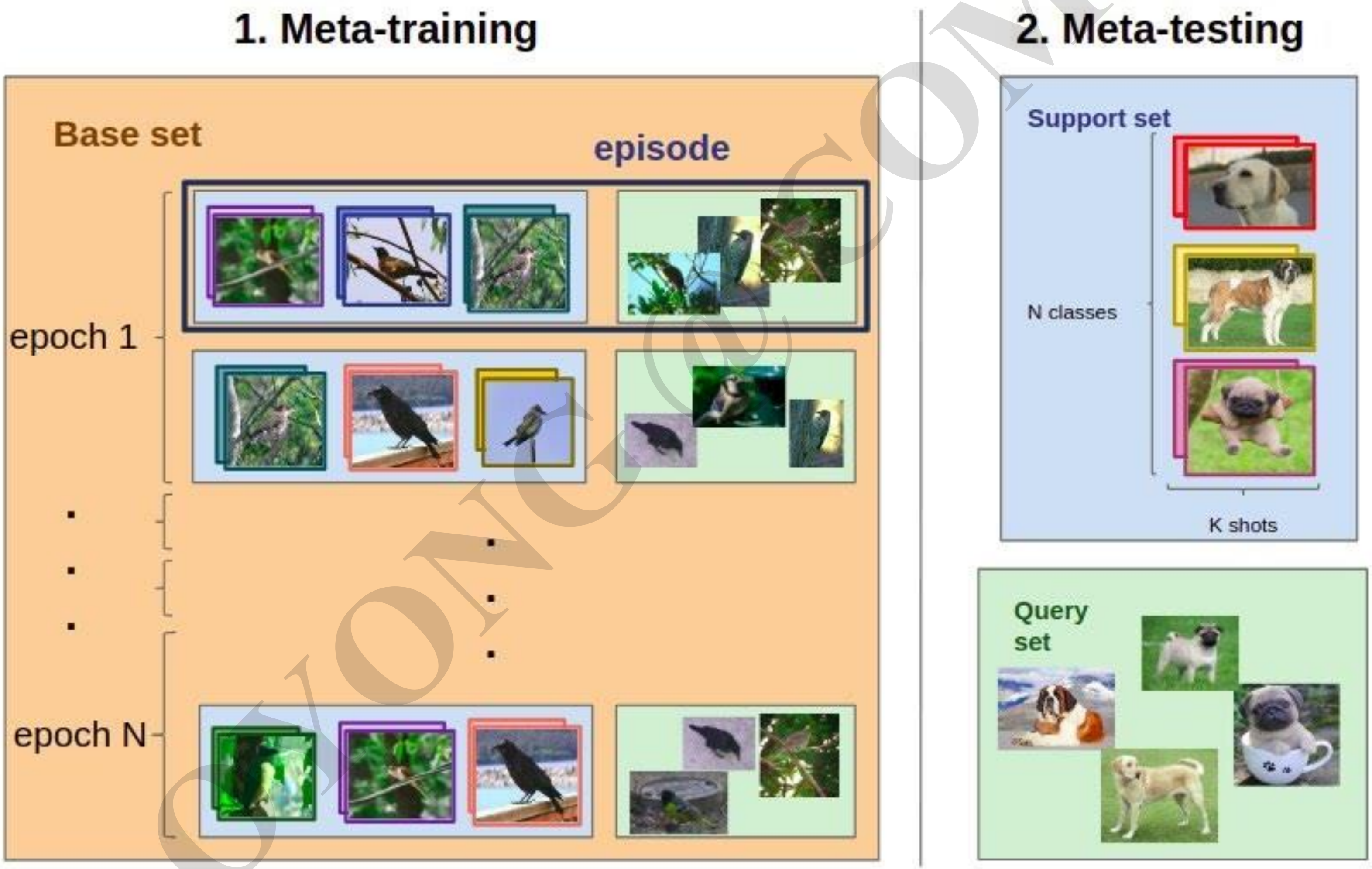
Few-Shot Learning

>**N-way-K-Shot-classification** problem to classify N classes each with K examples



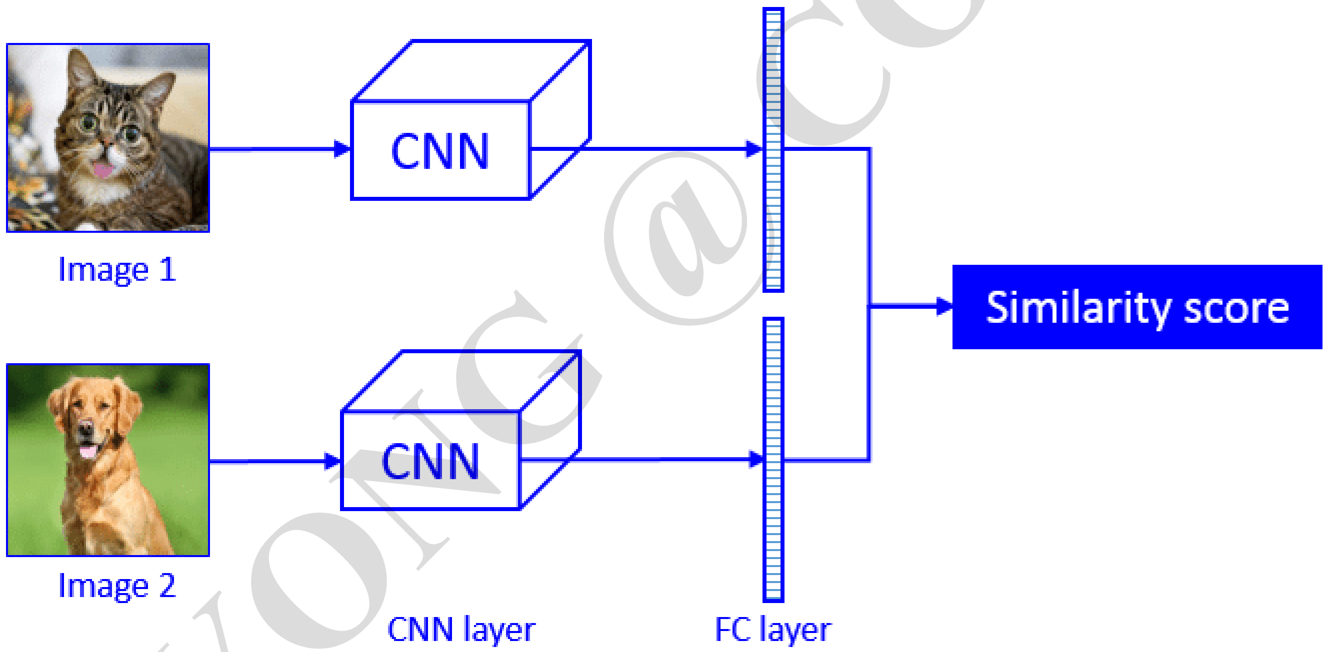
https://neptune.ai/blog/understanding-few-shot-learning-in-computer-vision

Few-Shot Learning – Meta (Transfer) Learning



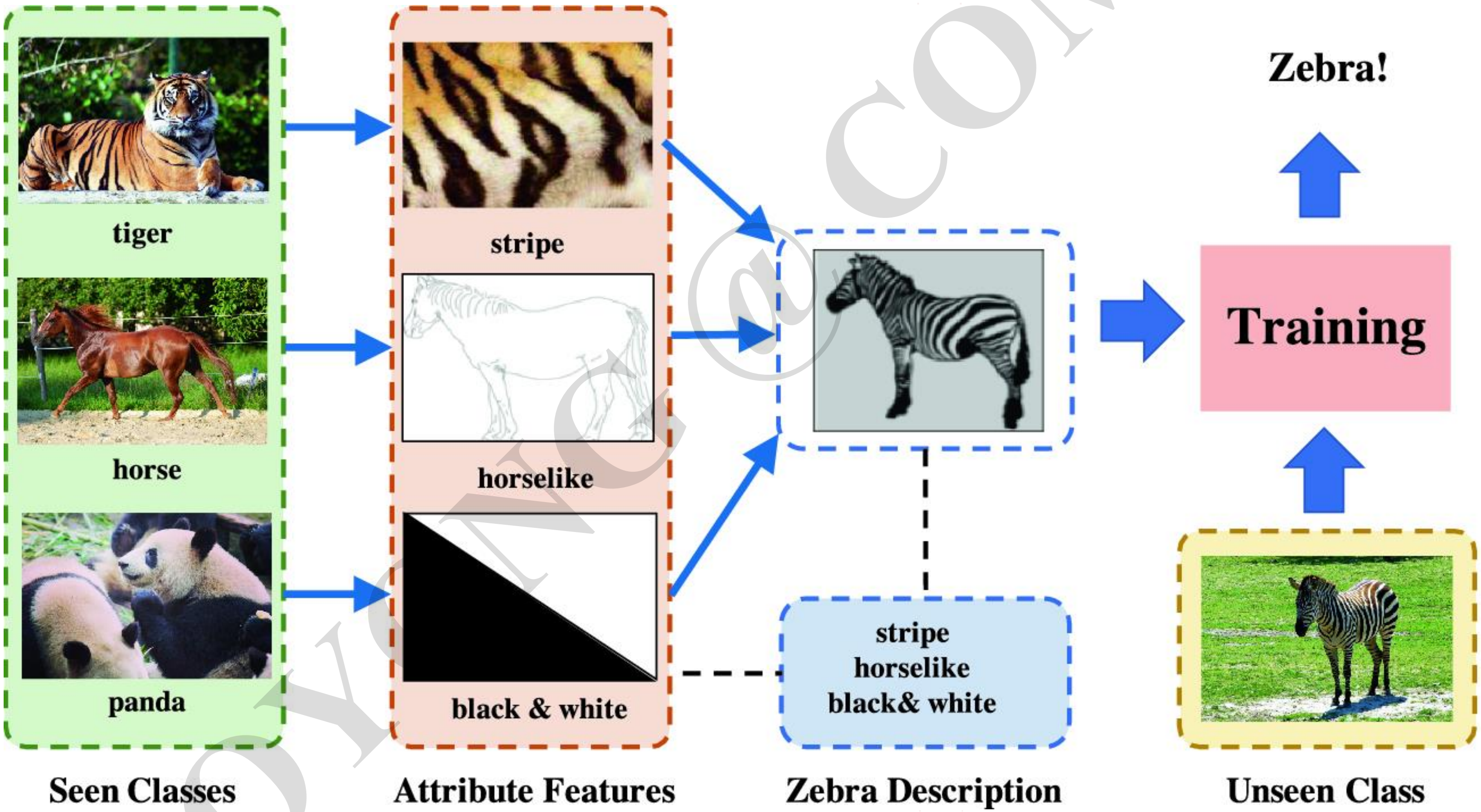
https://neptune.ai/blog/understanding-few-shot-learning-in-computer-vision

Few-Shot Learning – Metric/Contrastive Learning



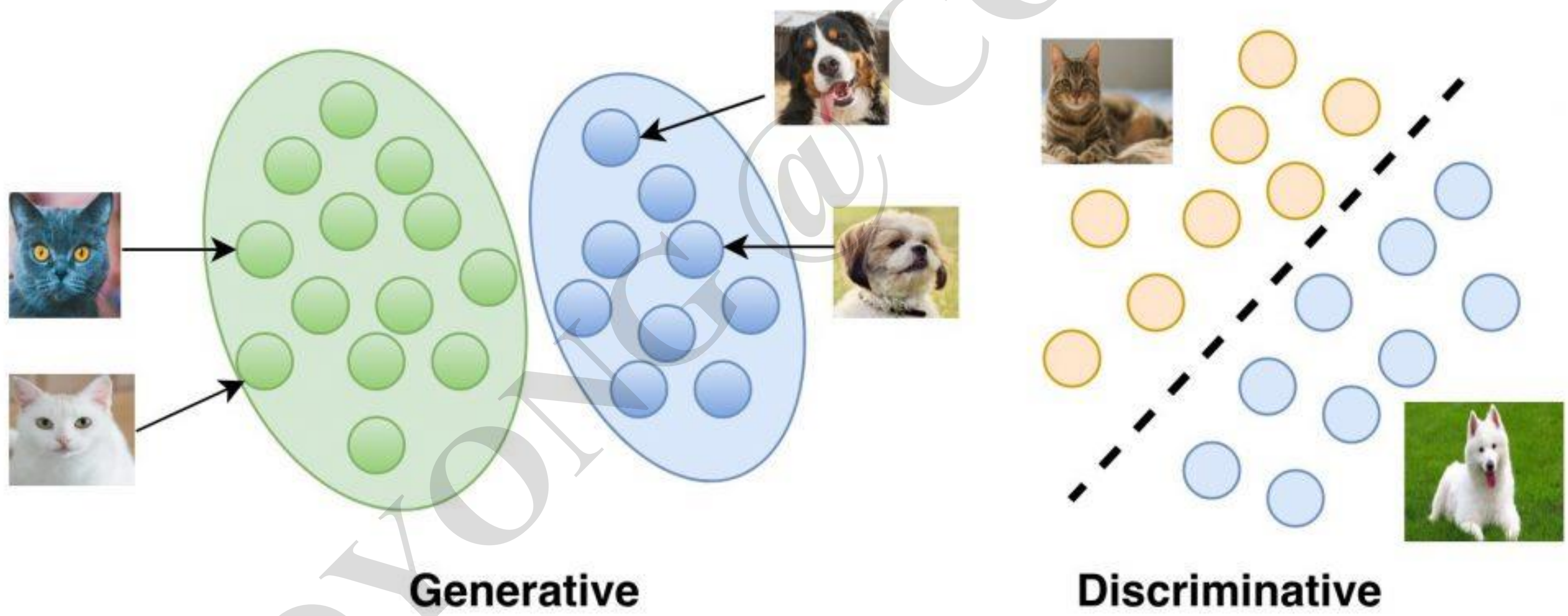
A New Deep Learning Based Multi-Spectral Image Fusion Method

Zero-shot Learning



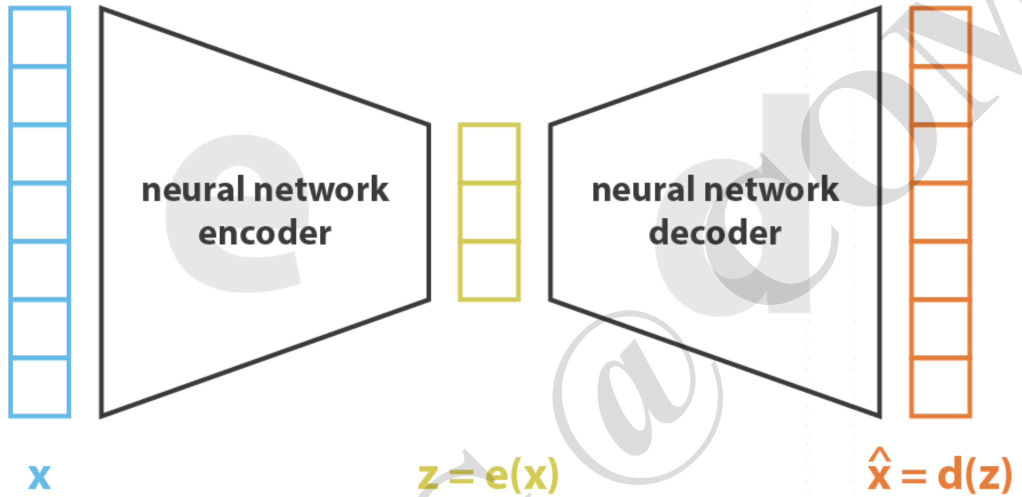
So far, we are mainly discussing how to **discriminate** things. However, in zero-shot learning, we start to **generate** things (unseen classes).

Generative vs. Discriminative



We can be more generative!

Variational Auto-Encoder(VAE)



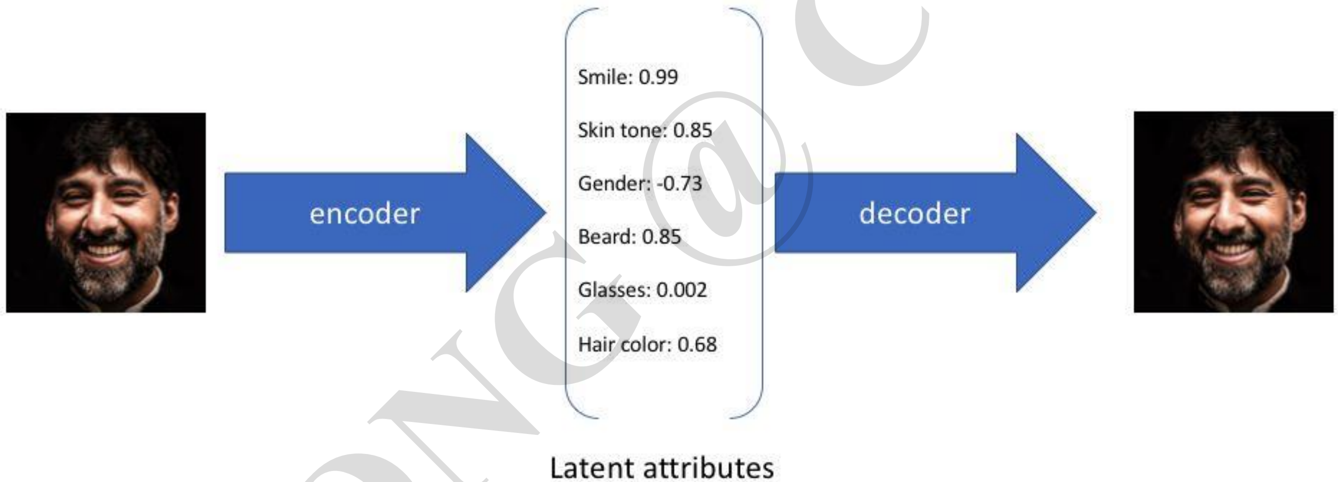
$$\text{loss} = \| x - \hat{x} \|^2 = \| x - d(z) \|^2 = \| x - d(e(x)) \|^2$$

In the training step, the input data is compressed into a vector of low dimensional normal distribution by the encoder, and the vector is reconstructed into the input data by the decoder.

In the generation step, a vector of normal distribution is randomly sampled and input into the decoder to generate data.

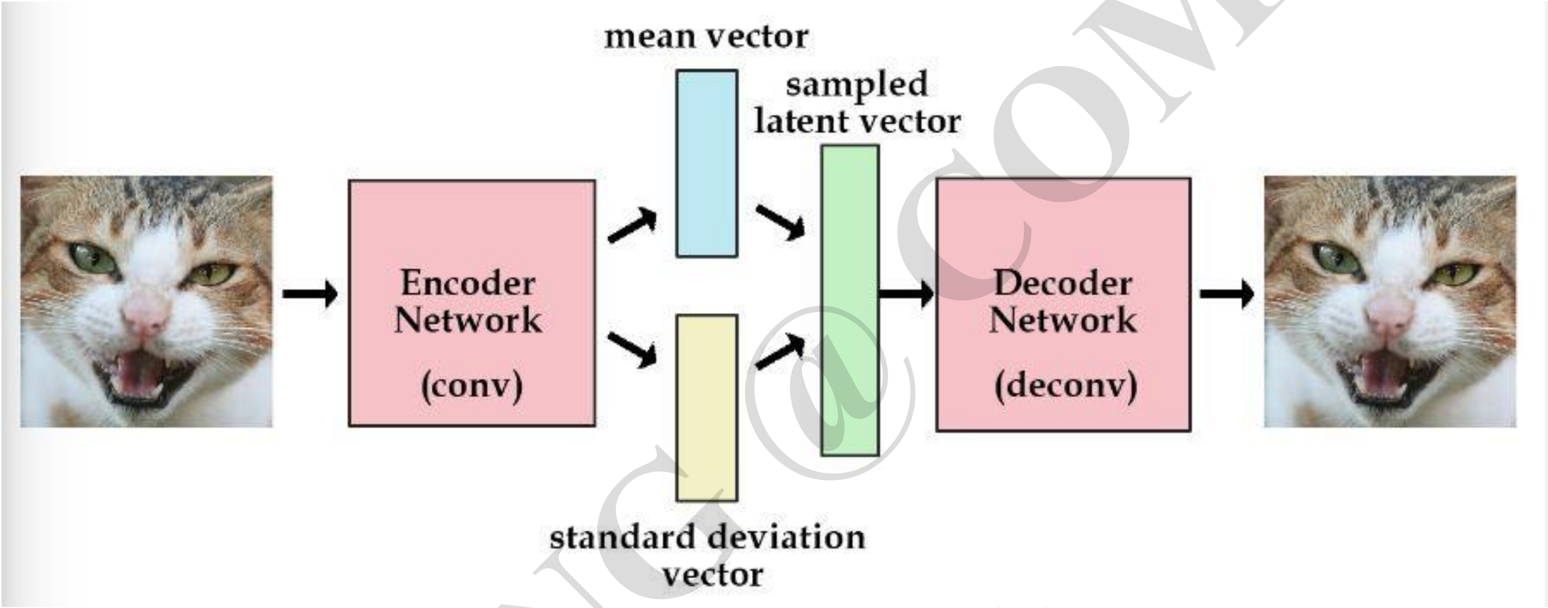
KINGMA D P, WELLING M. Auto-encoding variational bayes[Z]. 2014.
https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73

Variational Auto-Encoder(VAE)



KINGMA D P, WELLING M. Auto-encoding variational bayes[Z]. 2014.

Variational Auto-Encoder(VAE)



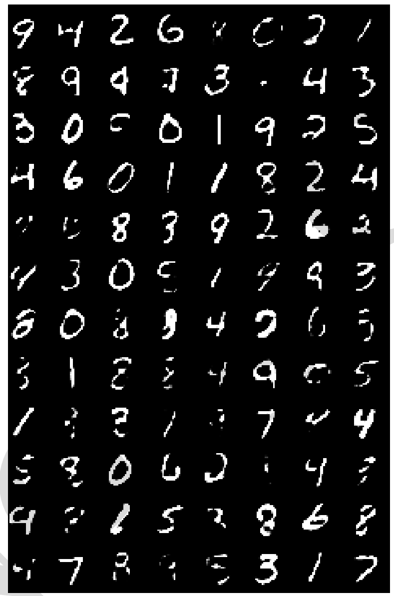
How to make the vector generated by the encoder follow the normal distribution is the core of VAE.

In fact, the output of VAE encoder is not a vector, but two values, which represent the mean and variance of a normal distribution, respectively. Once learned, the parameters can be used to modify a vector E that follows the standard normal distribution $N(0,1)$ by adding the mean values and multiplying by the variance. The modified vector is then used as the input of the decoder for generation.

This process is called reparameterization, which avoids the situation that the sampling is non differentiable (we cannot calculate the gradients in this case).

KINGMA D P, WELLING M. Auto-encoding variational bayes[Z]. 2014.

Variational Auto-Encoder(VAE)



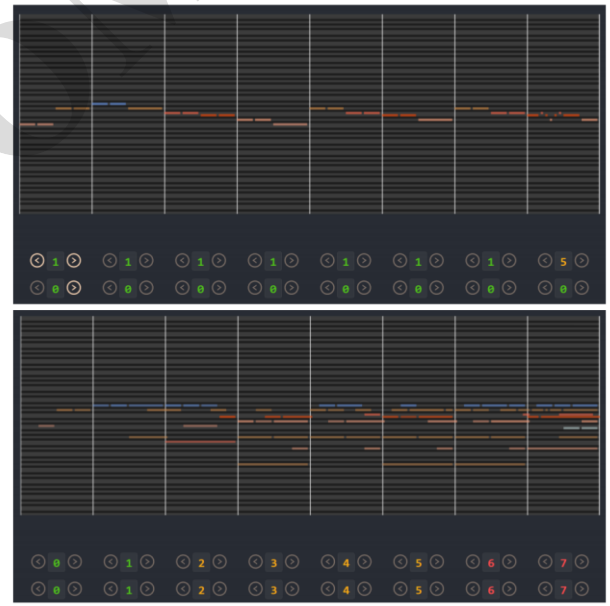


Image generation

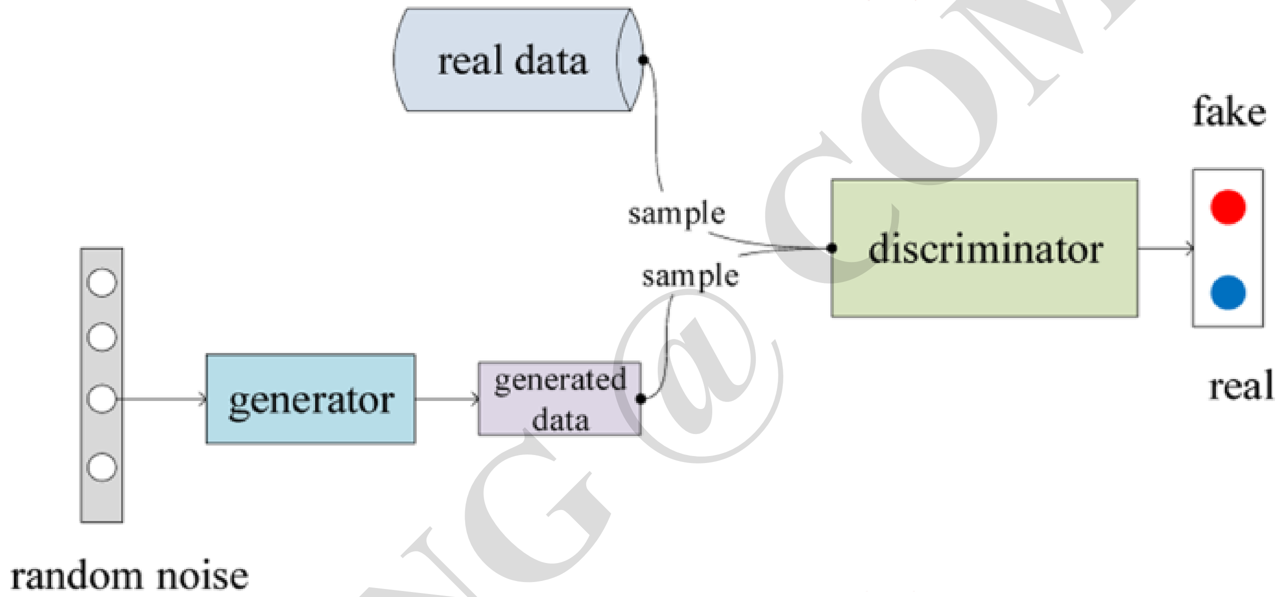
Symbolic music generation

https://blog.csdn.net/ppp8300885/article/details/80070723
Muse Morphose: Full-Song and Fine-Grained Music Style Transfer with One Transformer VAE

THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

VAE is not the only choice. Another alternative is GAN.

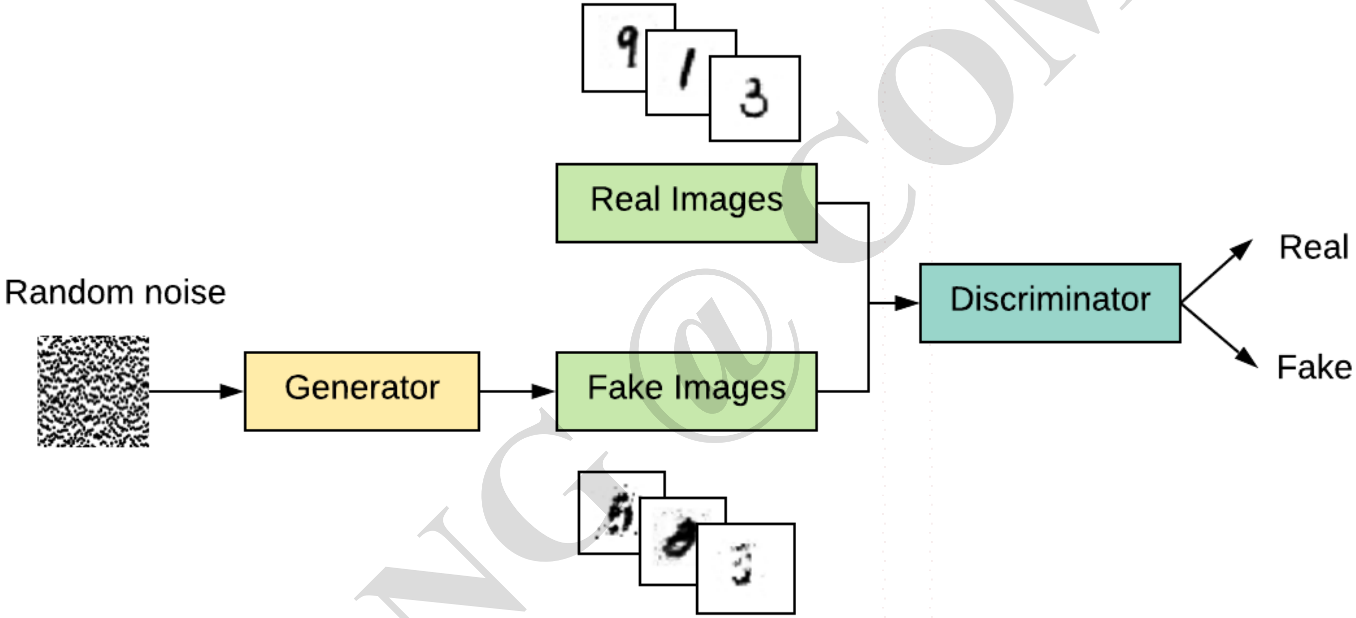
Generative Adversarial Networks(GAN)



Source: Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network

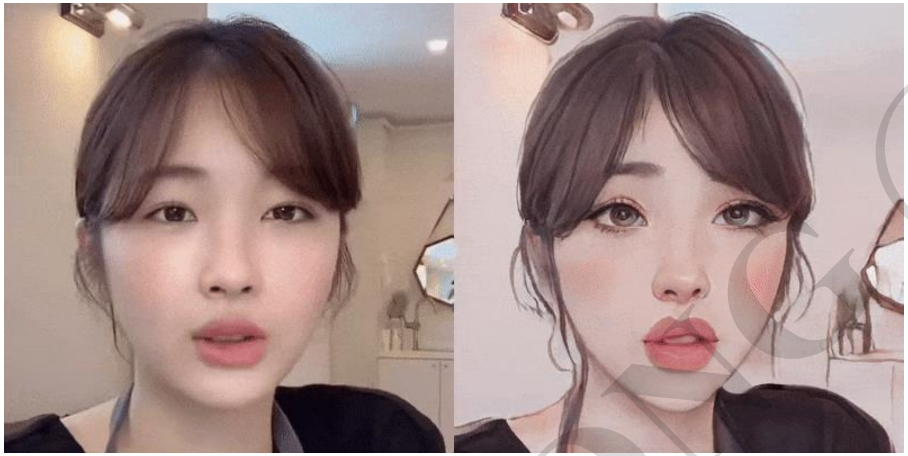
> GAN is an unsupervised generation model. It consists of two parts of generator and discriminator. The two parts are "opponents" to each other, in the way the generator tries its best to generate the "fake" data to "fool" the discriminator, while the discriminator tries its best to avoid being fooled. They make each other stronger through fighting.

Generative Adversarial Networks(GAN)



Taking MNIST dataset as an example. We can input a noise into G to generate a fake image, and then D judges whether the image is fake or not. If it’s being recognized as a fake image, the fooling failed, which means G is not good enough and its parameters need to be updated. Otherwise, D's is not smart enough and the parameters of D need to be updated.

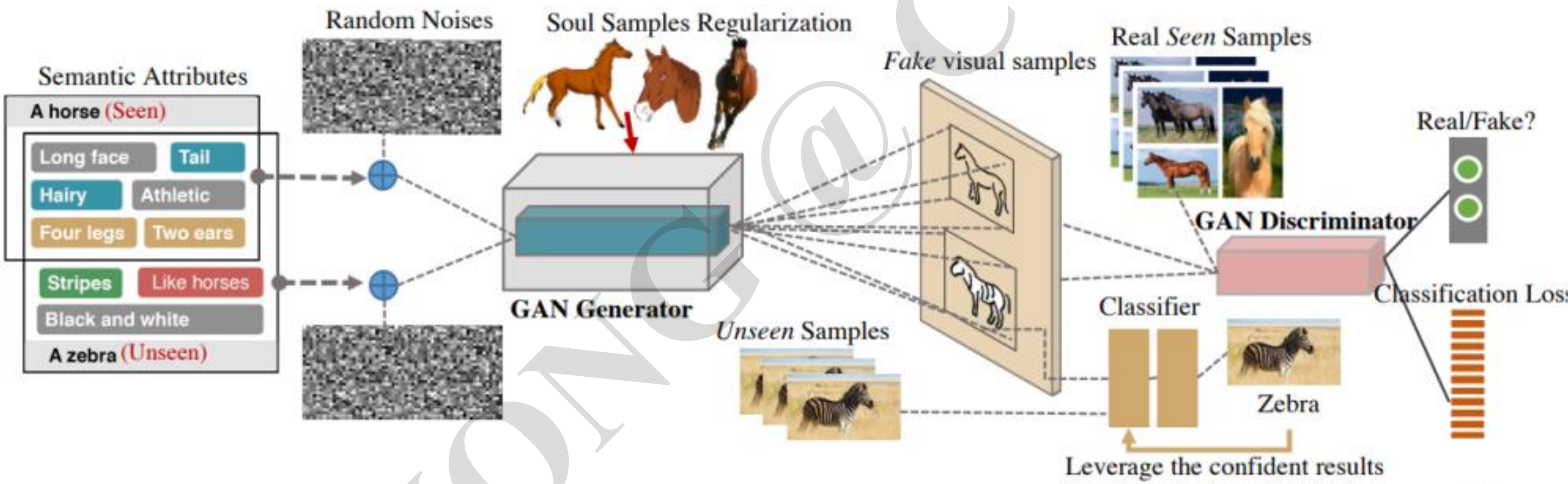
Applications



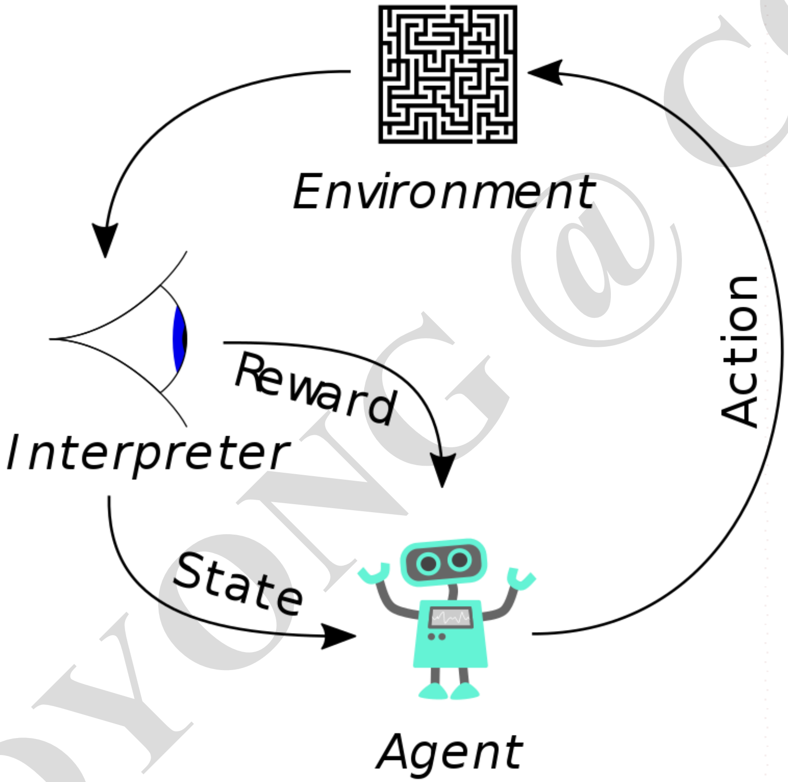
Style Transfer

Image Generation

Zero-shot Learning using GAN



Reinforcement Learning



Source: https://en.wikipedia.org/wiki/Reinforcement_learning#/media/File:Reinforcement_learning_diagram.svg

Reinforcement Learning

https://songyanho.github.io/Reinforcement-Learning-for-Self-Driving-Cars/

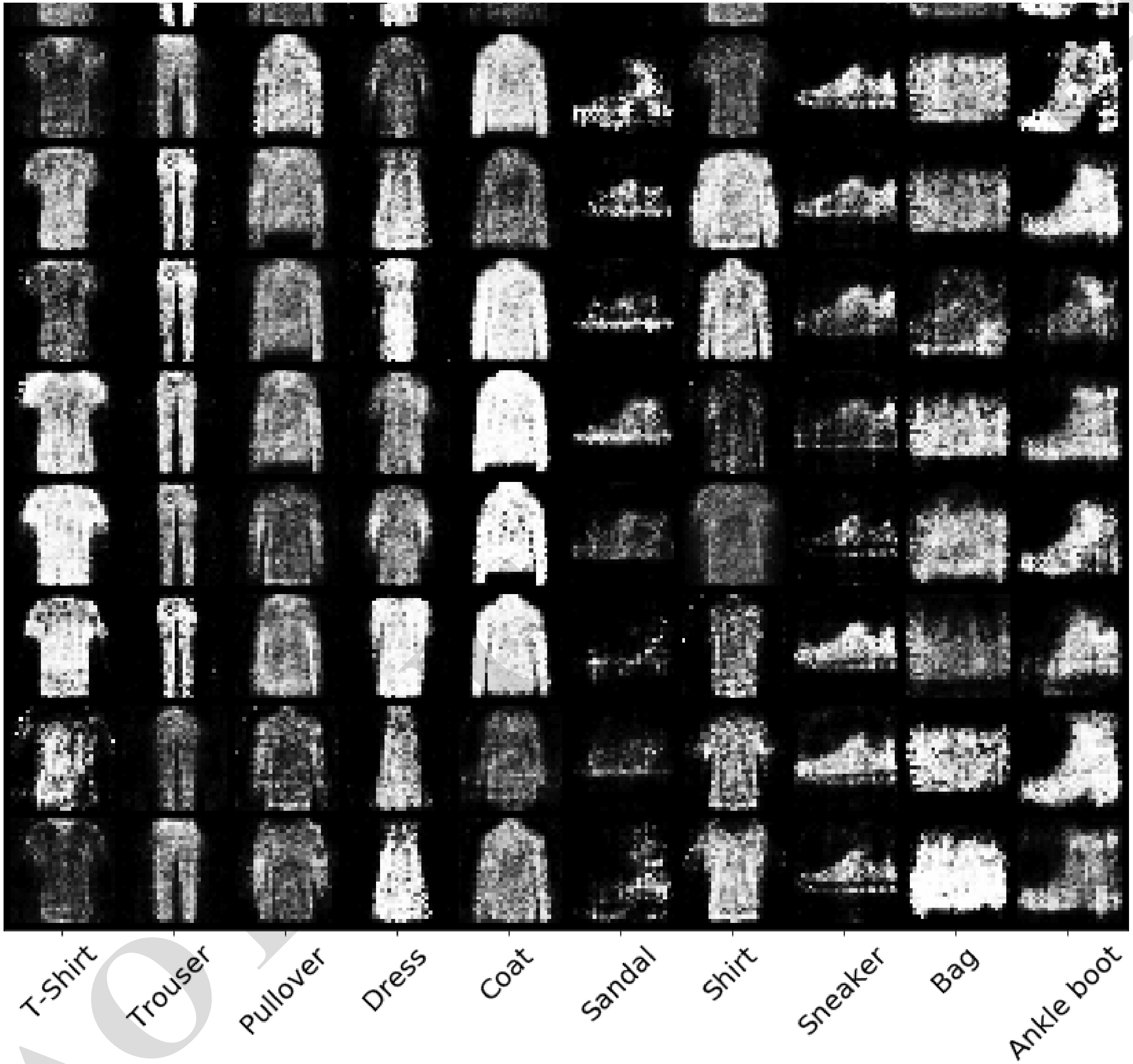
Reading List

>Multimodal learning

>Pretraining and big models

>Diffusion models

Project


T-Shirt
Trouser
Pullover
Dress
Coat
Sandal
Shirt
Sneaker
Bag
Ankle boot

THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學
Department of Computing
電子計算學系

Thank you!