Advanced Deep Learning

by Prof. Seungchul Lee
Industrial AI Lab
http://isystems.unist.ac.kr/
POSTECH

Table of Contents

1. Artistic Style Transfer

- Original paper (https://arxiv.org/abs/1508.06576 (https://arxiv.org/abs/1508.06576))

- https://shafeentejani.github.io/2016-12-27/style-transfer/ (https://shafeentejani.github.io/2016-12-27/style-transfer/)
- https://harishnarayanan.org/writing/artistic-style-transfer/ (https://harishnarayanan.org/writing/artistic-style-transfer/)

Artistic Style Transfer for Video

- Vincent van Gogh

In [1]:
```
%%html
<center><iframe src="https://youtube.com/embed/ckqemfh0JMM?rel=0"
width="560" height="315" frameborder="0" allowfullscreen></iframe></center>
```

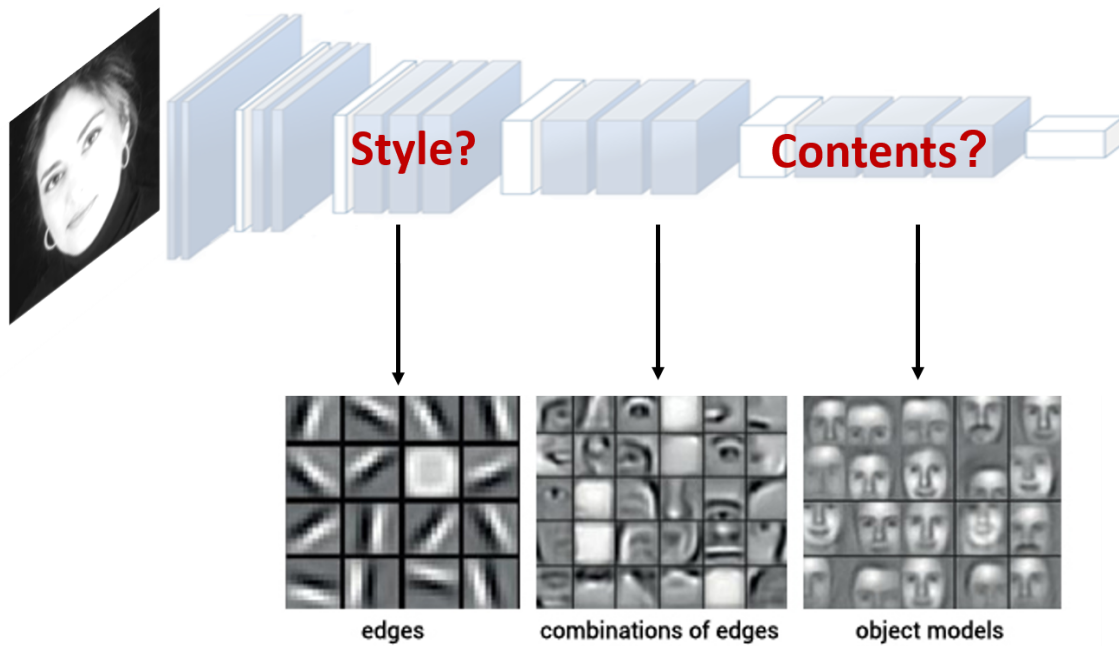[iSystems] Video Style Transfer Gogh

- Picasso

In [2]:
```
%%html
<center><iframe src="https://youtube.com//embed/nMwU4avioVo?rel=0"
width="560" height="315" frameborder="0" allowfullscreen></iframe></center>
```

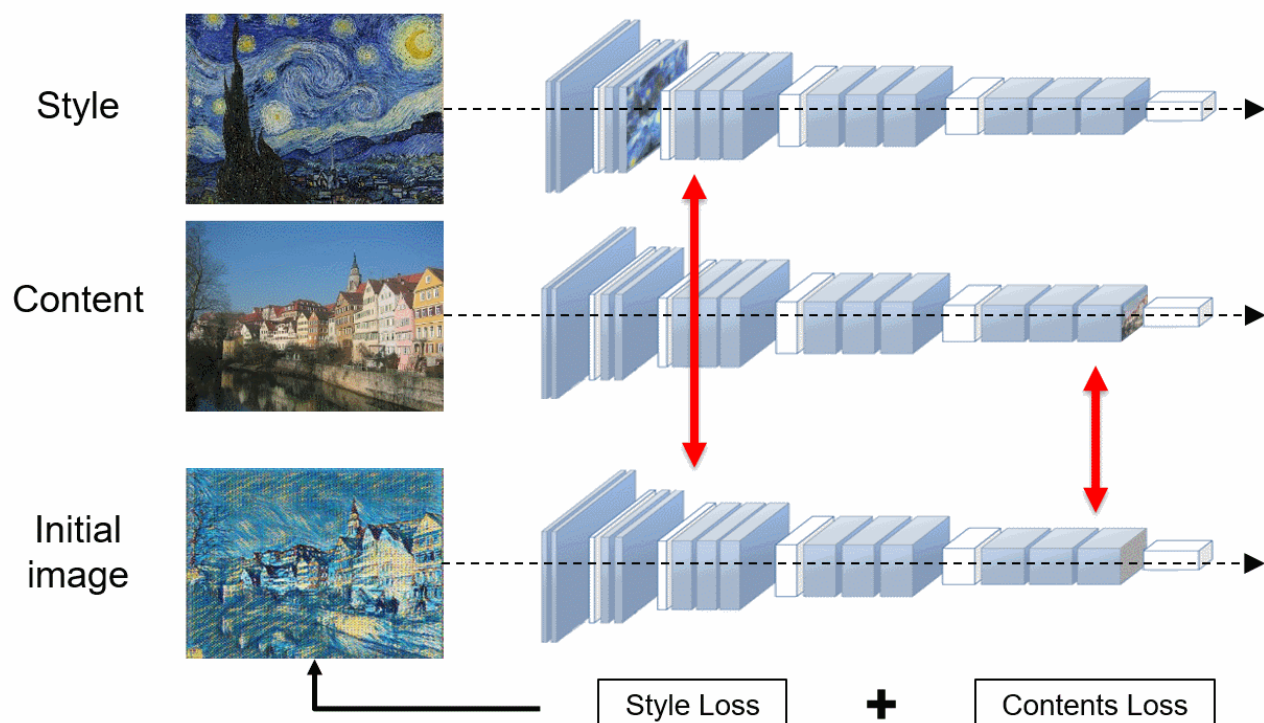[iSystems] Video Style Transfer Picasso

Revisit CNN

- Hierarchical feature representation
    - Contents representation
    - Style representation

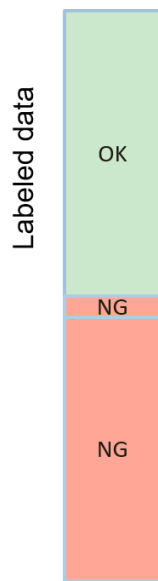|		
edges	combinations of edges	object models

Style Transfer

- Image construction

2. Discriminant Model vs. Generative Models
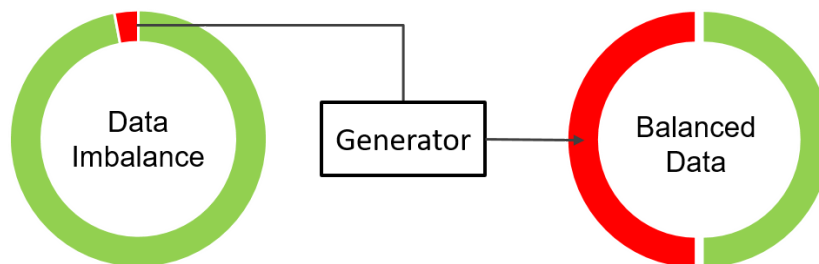
Imbalanced Data

- Not enough data from faulty status
- Data Imbalance
    - Under sampling
    - Over sampling
    - Re-weighting
    - (Ada)Boosting

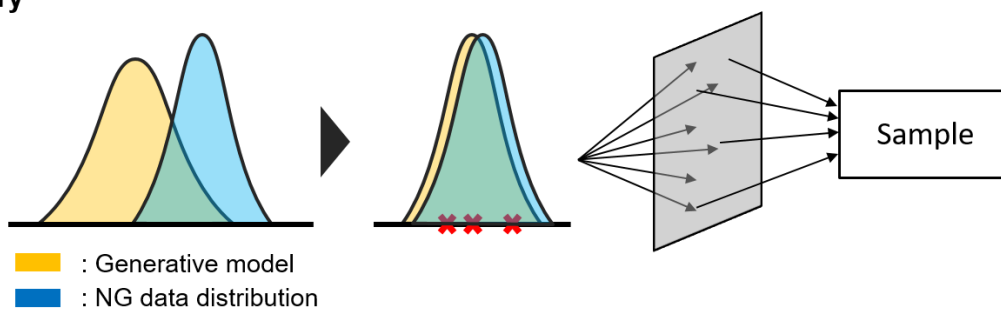$$L(x, y, \theta) = \sum_{i=1}^{N} \omega(y_i) \cdot l(y_i, \hat{y}_i)$$

Generative Model

- Data imbalance
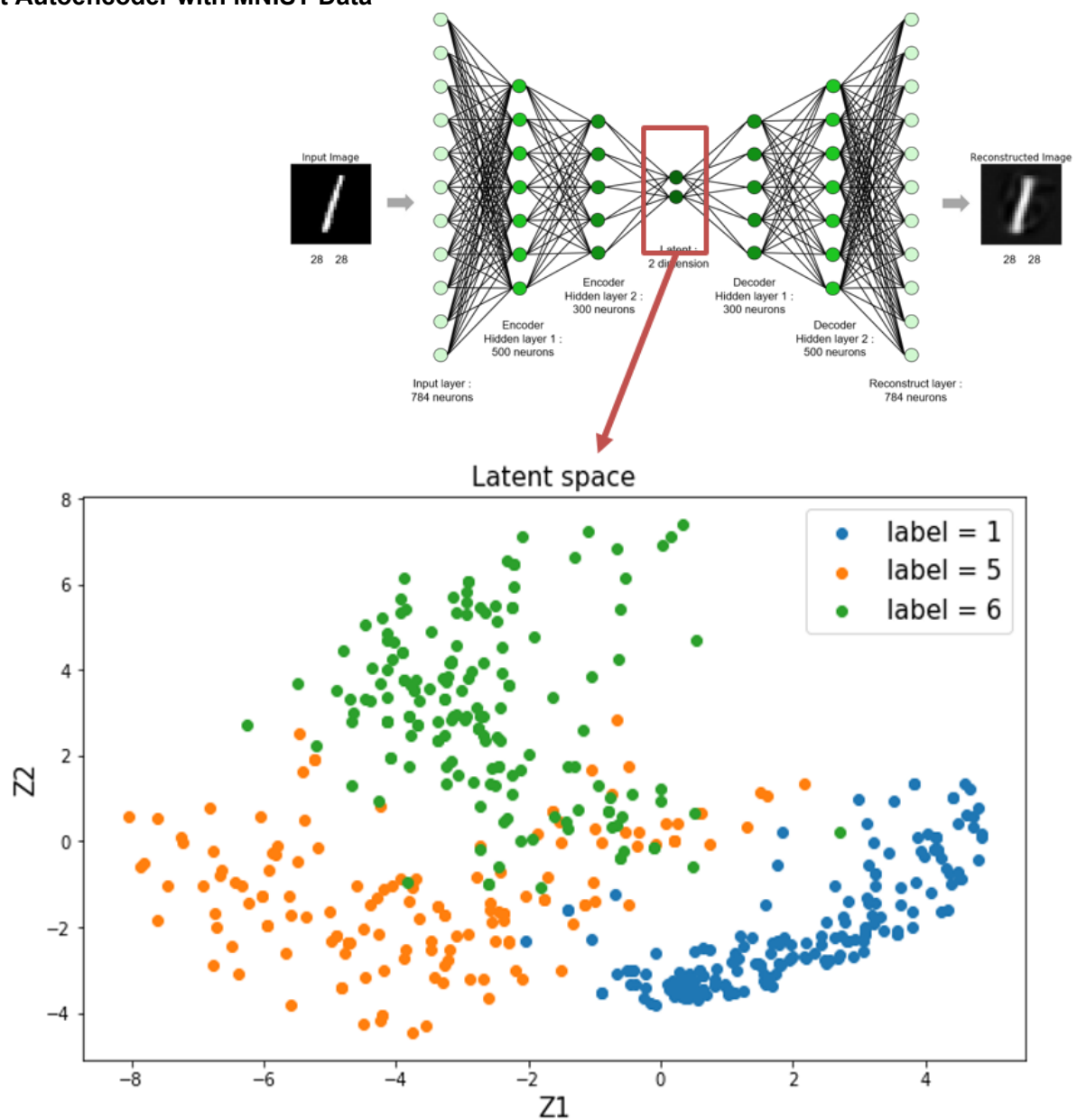 - Problematic in reality
  - For example, 98% OK, 2% NG
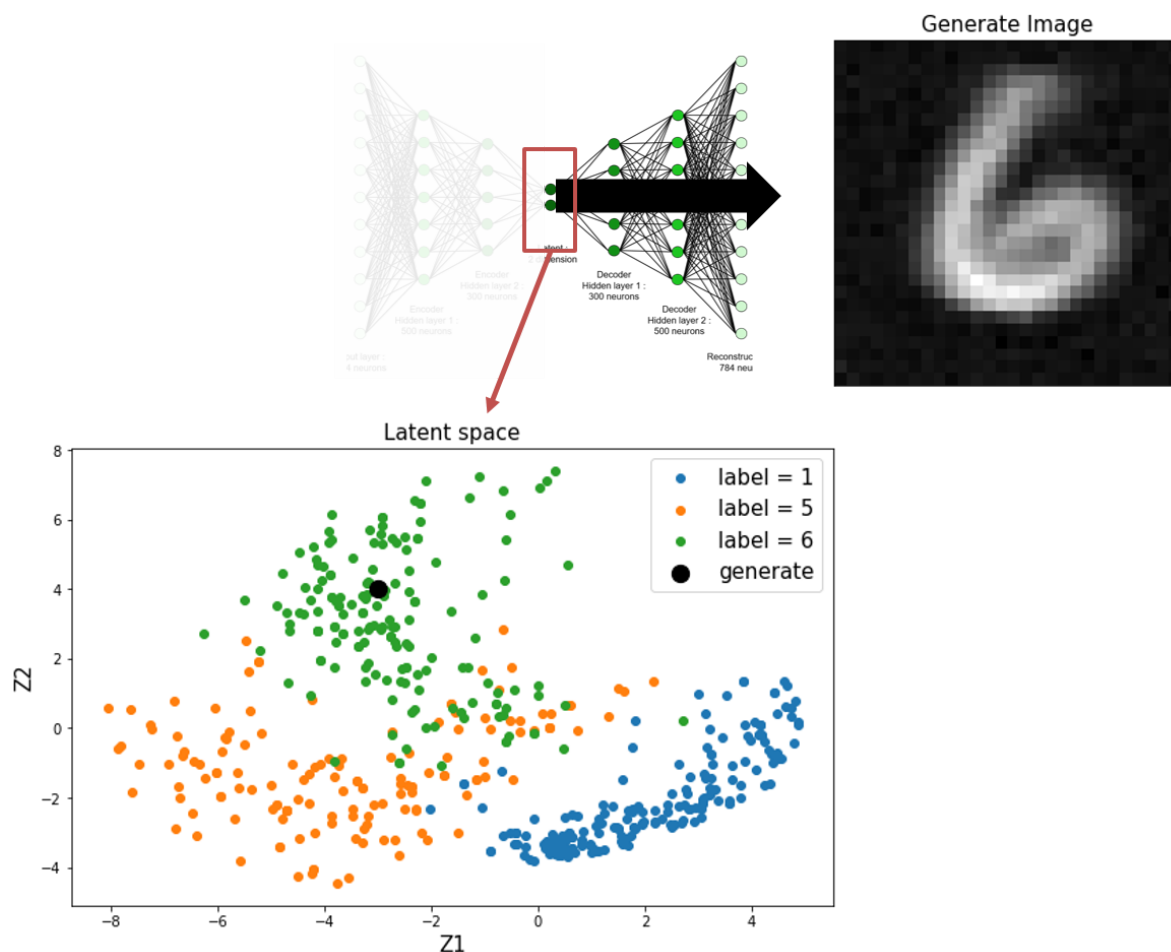

Application

- -

Theory

: Generative model
: NG data distribution

Revisit Autoencoder with MNIST Data

Latent space

Data Generation from Decoder

Generate Image

Latent space

3. Generative Adversarial Networks (GAN)

- original paper by Ian J. Goodfellow (https://arxiv.org/abs/1406.2661 (https://arxiv.org/abs/1406.2661))

- https://blog.openai.com/generative-models/ (https://blog.openai.com/generative-models/)

Turing test
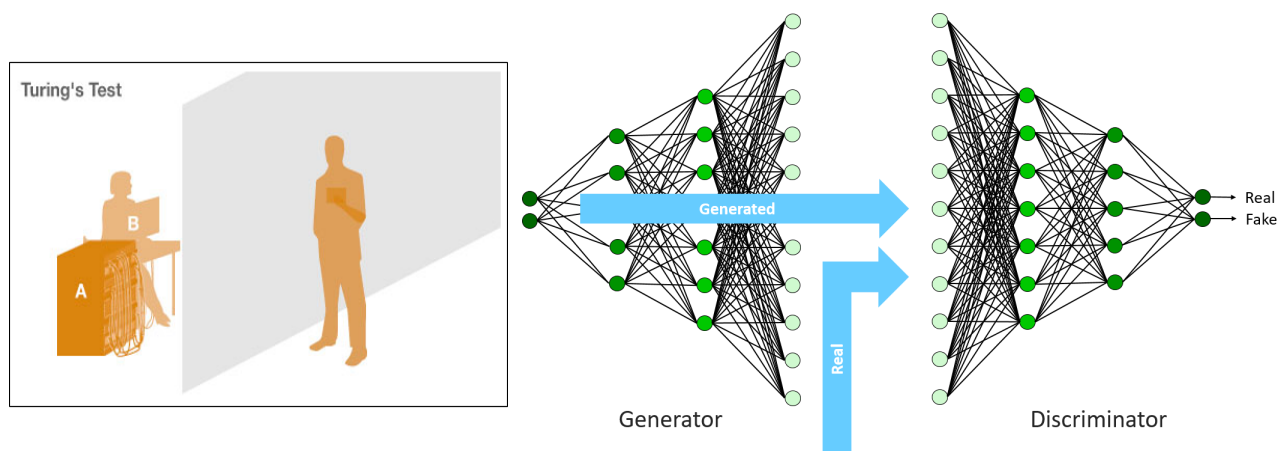
- How to generate data?
  - Train through competition
  - Generator vs. Discriminator

$$H(D(G(z)), 1) + H(D(x), 0)$$

Real Data

$$x \sim p_{data}(x)$$

$$z \sim p_z(z)$$

Prior

Generator Network

$$G(z)$$

Generated Data

Discriminator Network

$$D(x)$$

1: Fake

0: Real

$$H(D(G(z)), 0)$$
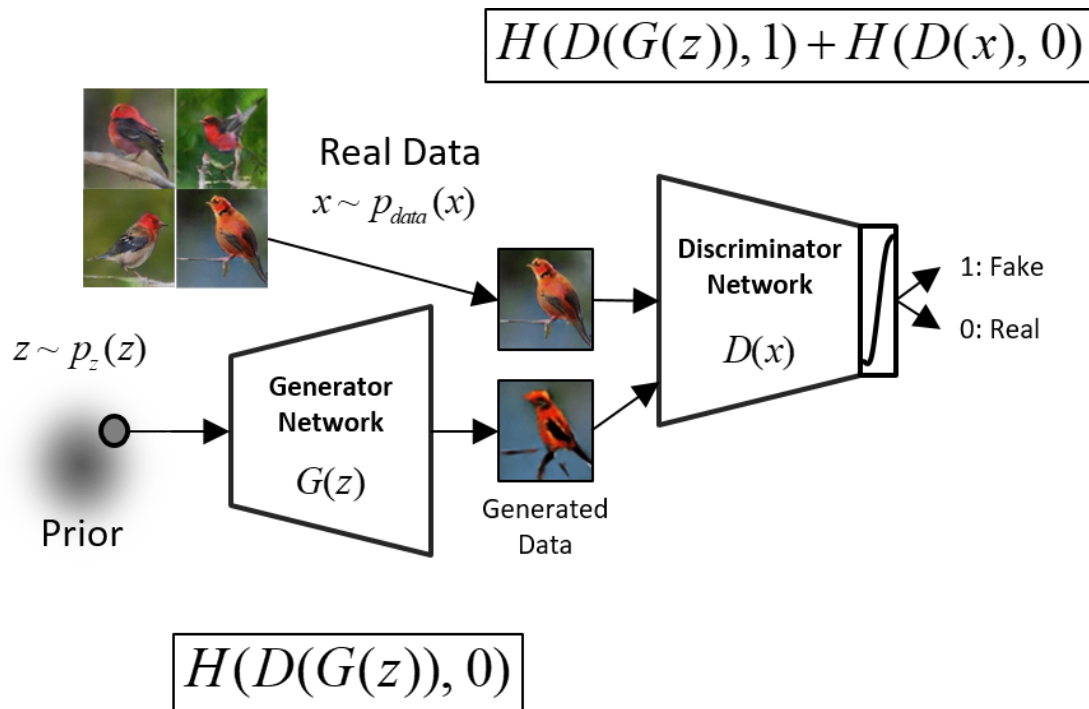
4. Deep Learning Implementation

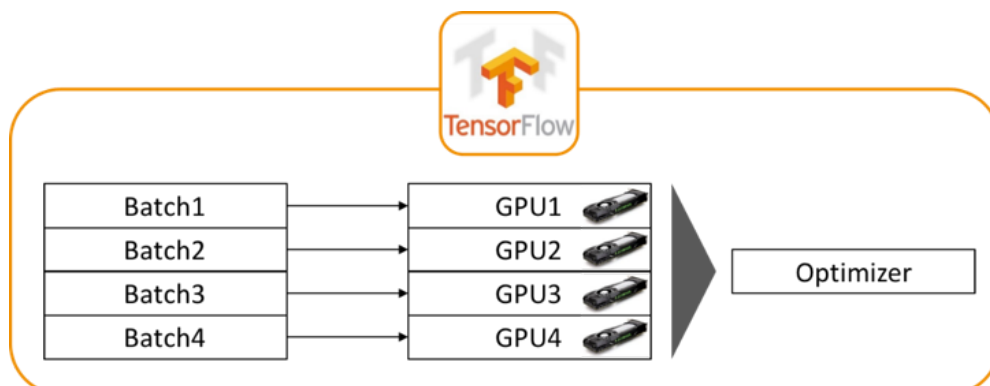**Computation Environment for Model Learning**

- Development environment (open source)
    - Ubuntu 14.04
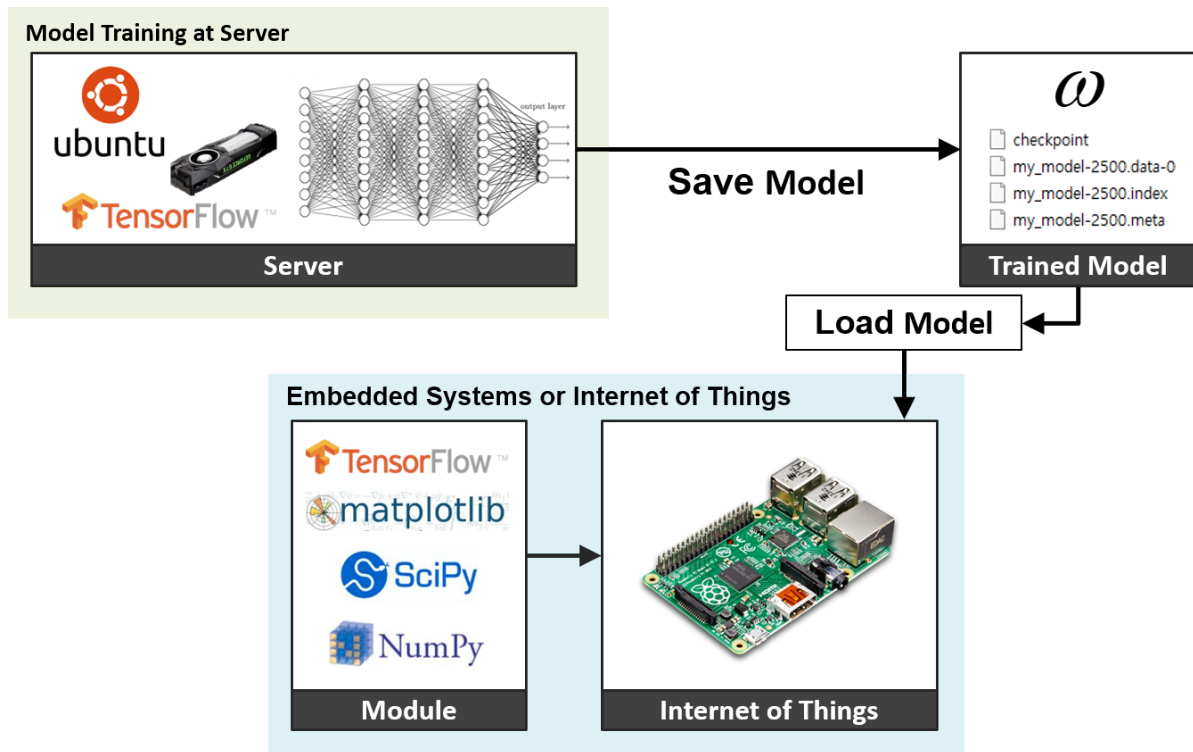   - Python3
    - TensorFlow

- Machine
    - GPU: GeForce GTX TITAN X (PASCAL)
    - CPU: Intel i7-5930k 6 Core 3.5GHz processor

- Parallel computing
    - Multi GPUs

Implementation of Deep Learning Model


```
In [3]: %%html
        <center><iframe src="https://www.youtube.com/embed/-eqVRXtX44Y?rel=0"
        width="560" height="315" frameborder="0" allowfullscreen></iframe></center>
```


[iSystems] Deep Learning on Raspberry Pi

```
In [4]: %%javascript
        $.getScript('https://kmahelona.github.io/ipython_notebook_goodies/ipython_no
        tebook_toc.js')
```