한국기술교육대학교 능력개발교육원

4차 산업혁명 핵심 기술 컨퍼런스

2020년 1월 21일

4차 산업혁명 인재양성을 위한 인공지능 (AI) 교수 방법

임성규
(조지아 공대)

KOREATECH 한국기술교육대학교

KOREATECH HRDI 능력개발교육원

Agenda

1. 인공지능 (AI) 기계학습이란?

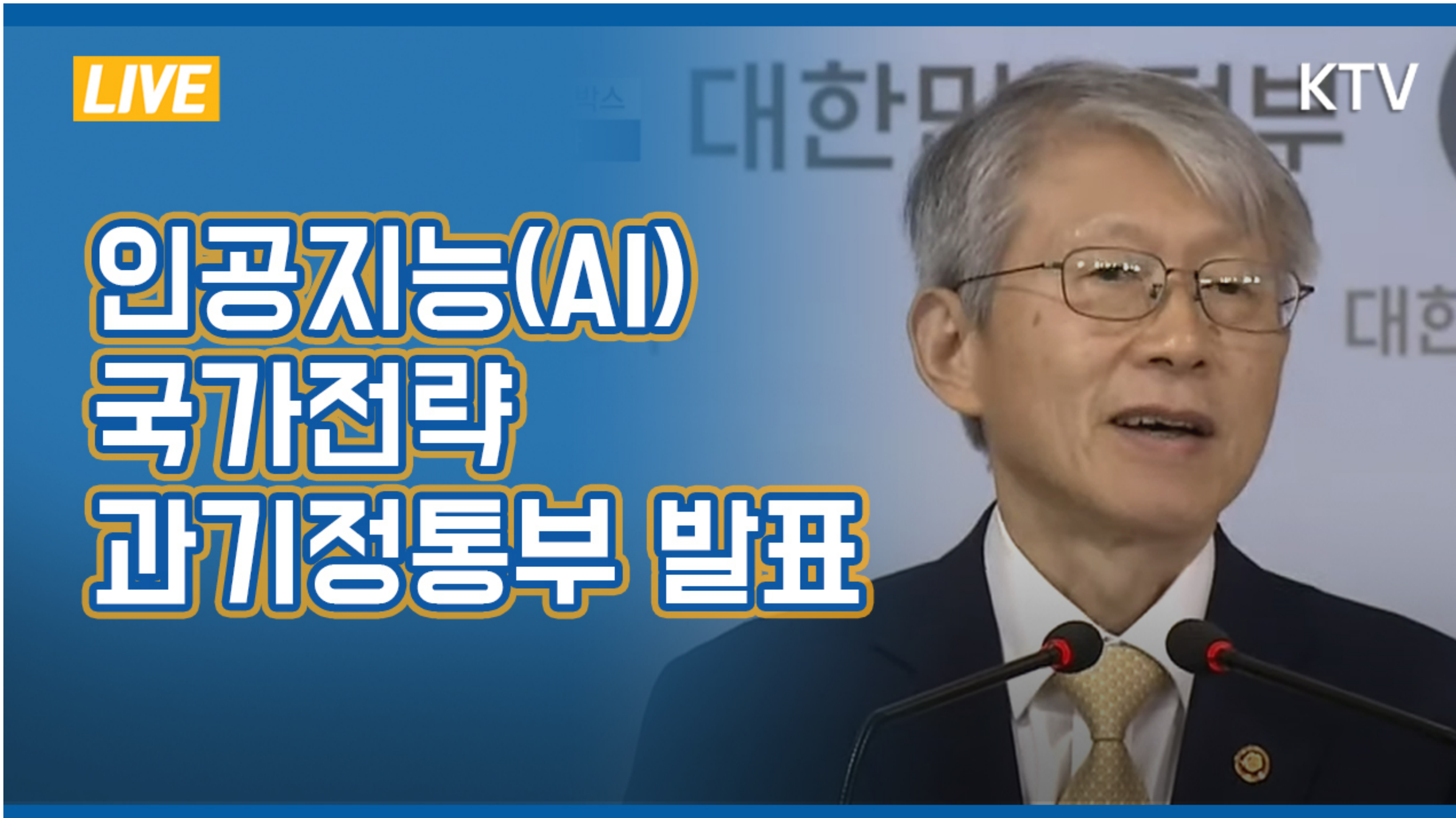
2. 기계학습의 종류
   - CNN, RNN GAN, Autoencoder, Unet

3. 기계학습 교수법 및 코딩 실습의 예
   - RNN으로 주가 변동 예측하기
   - Azure로 붓꽃 판별하기

4. 기계학습 관련 진행 연구
   - 회로설계에 적용된 기계학습

인공지능 (AI) 국가전략

인공지능 (AI) 국가전략



제 2의 석유

- Who has the world's largest photo DB?
  - How do we make $$$ out of this?
- Who has the world's largest video DB?
  - How do we make $$$ out of this?
- Who has the world's largest shopping DB?
  - How do we make $$$ out of this?
- Who has the world's personal DB?
  - How do we make $$$ out of this?

글로벌 시가총액 5대 기업

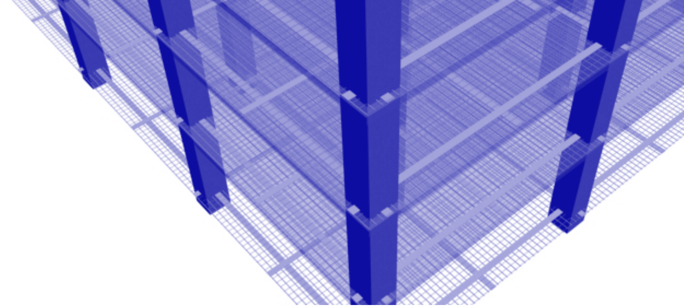
	2007년	2017년
1위	페트로 차이나	애플
2위	엑손 모빌	구글
3위	제너럴 일렉트릭	마이크로 소프트
4위	차이나 모바일	아마존
5위	중국 공상 은행	페이스북

https://en.wikipedia.org/wiki/list_of_public_corporations_by_market_capitalization

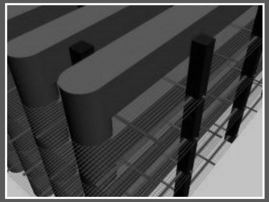
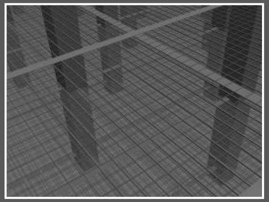
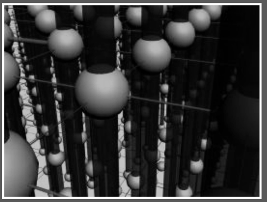
2019년 12월 30일 뉴스

日 GDP 급 시총 5조달러 테크기업 빅5, '70년 美 패권' 이어나가는 힘이다

ㄴ 인터넷→모바일→ 이젠 플랫폼으로 21세기도 움켜쥐는 美



1. 인공지능 (AI) 기계학습이란?

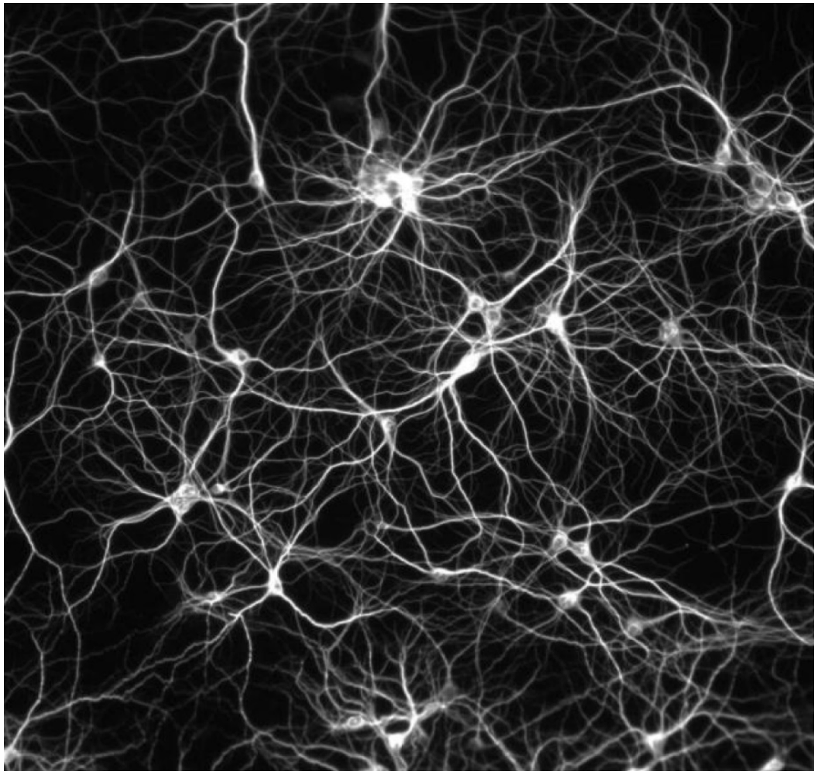





딥러닝이란? [네이버 지식백과]

- 사물이나 데이터를 군집화하거나 분류하는 데 사용하는 기술
  - 컴퓨터는 사진만으로 개와 고양이를 구분하지 못함
  - 하지만 사람은 아주 쉽게 구분할 수 있음
  - 이를 위해 기계학습 (Machine Learning) 이라는 방법이 고안됨

Human Brain

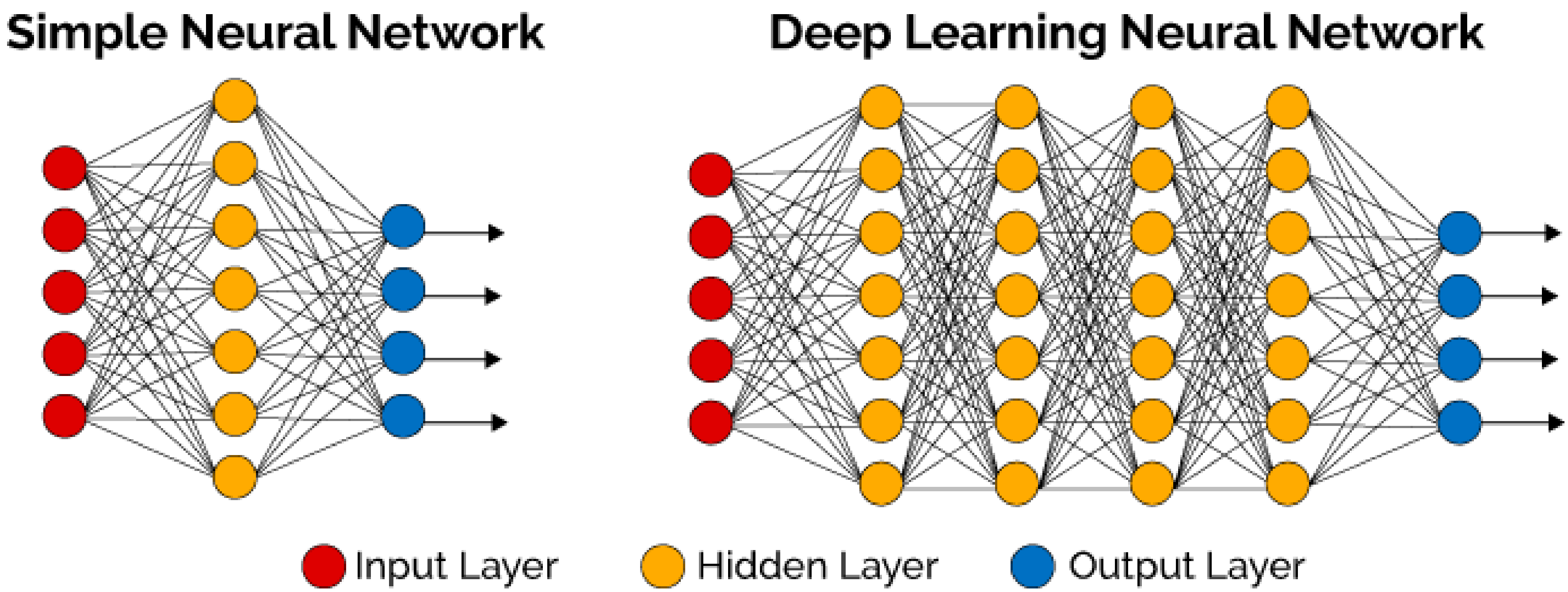
- **Amazing structure**



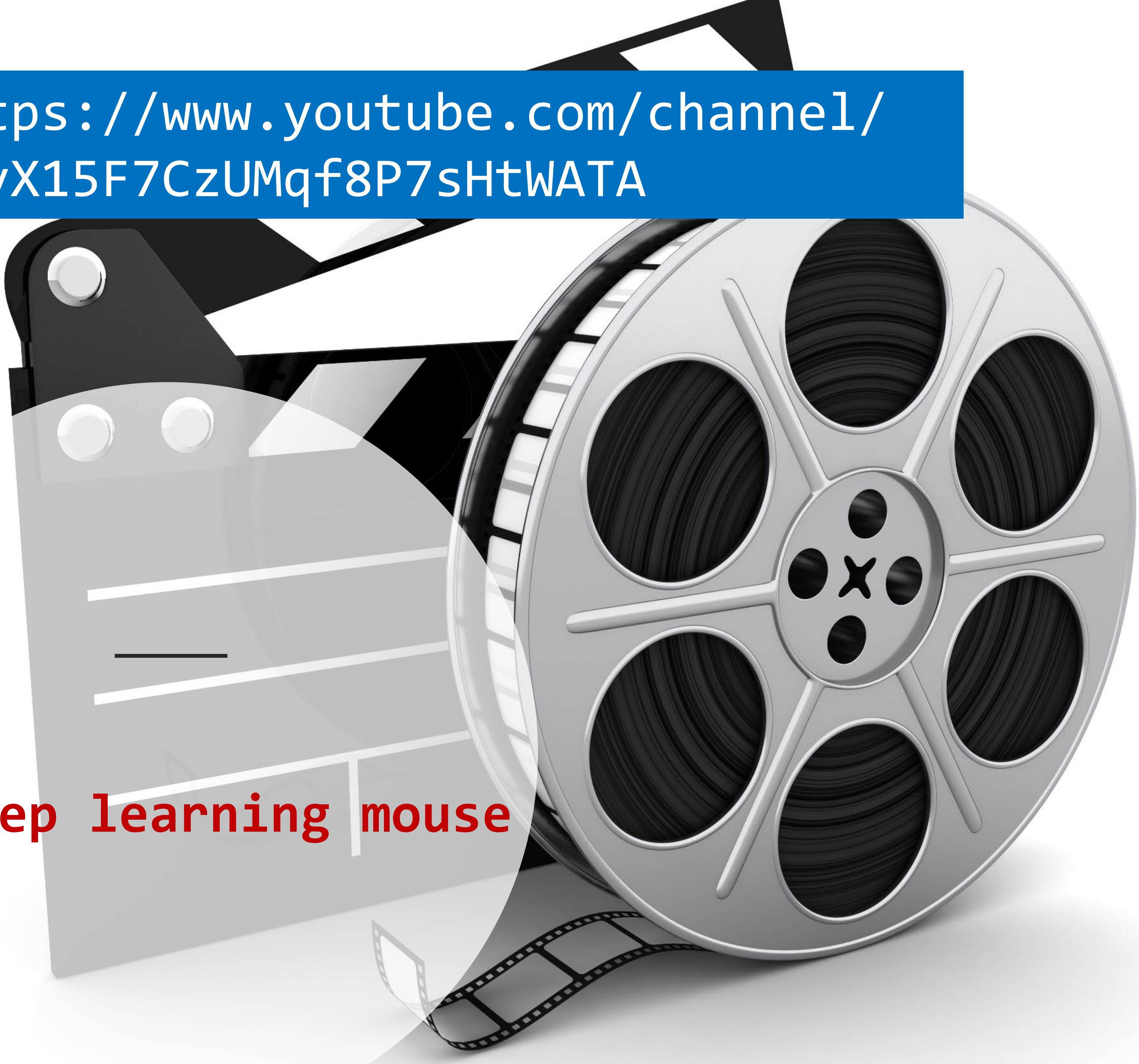
천억개의 뉴런들이 백조개의 시냅스로 연결

Deep Neural Network

- **DNN simply has more hidden layers than ANN**
  - **More weights to tune**
  - **Becomes more powerful**

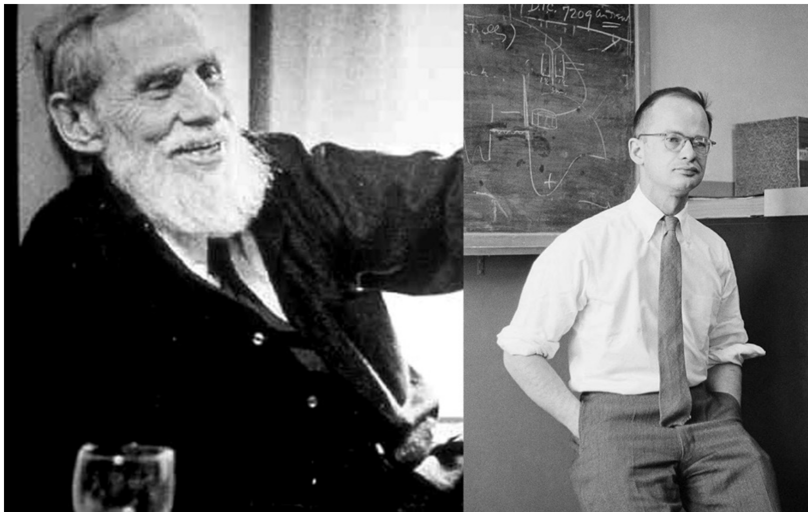


https://www.youtube.com/channel/UCvX15F7CzUMqf8P7sHtWATA



deep learning mouse

딥러닝의 계보

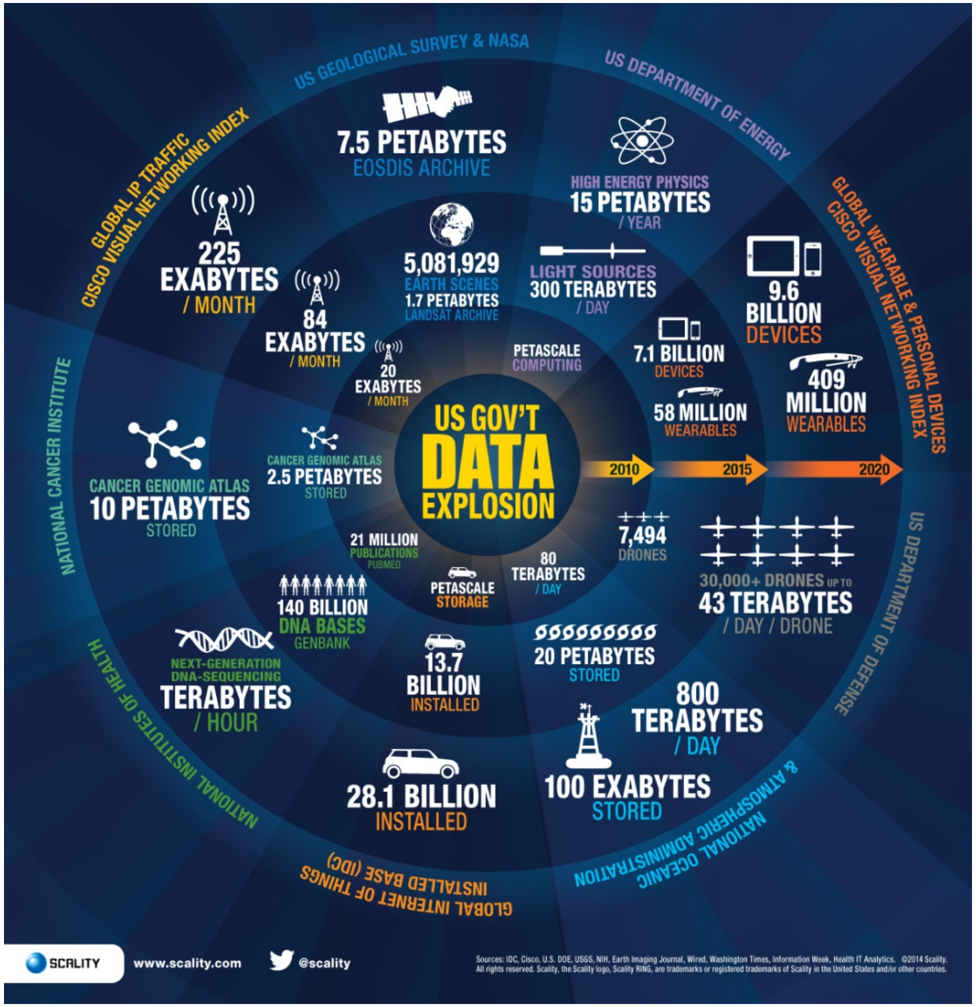


McCulloch & Pitts
math model

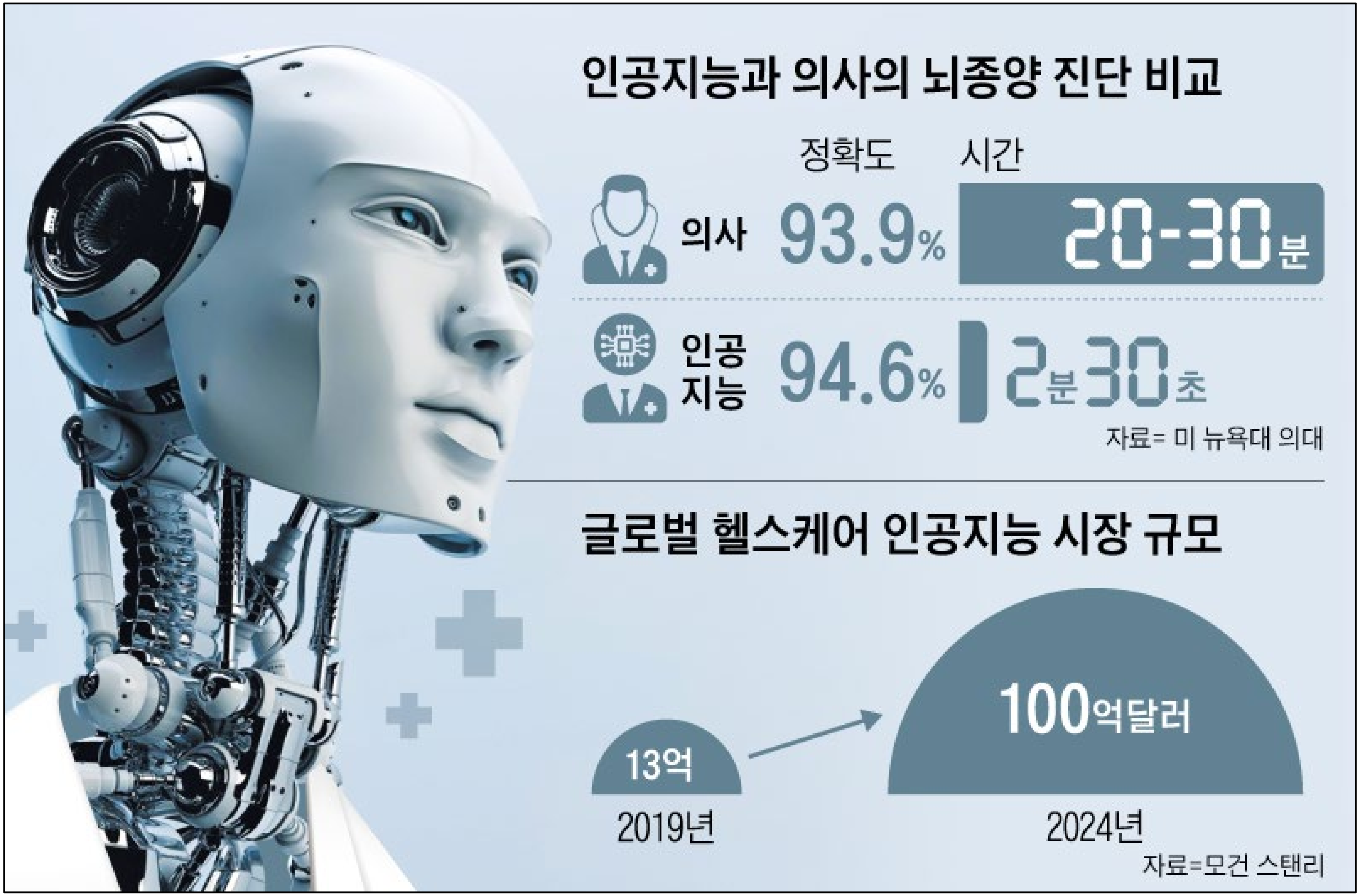
Hinton
DNN, back propagation

Rosenblatt
computer simulation

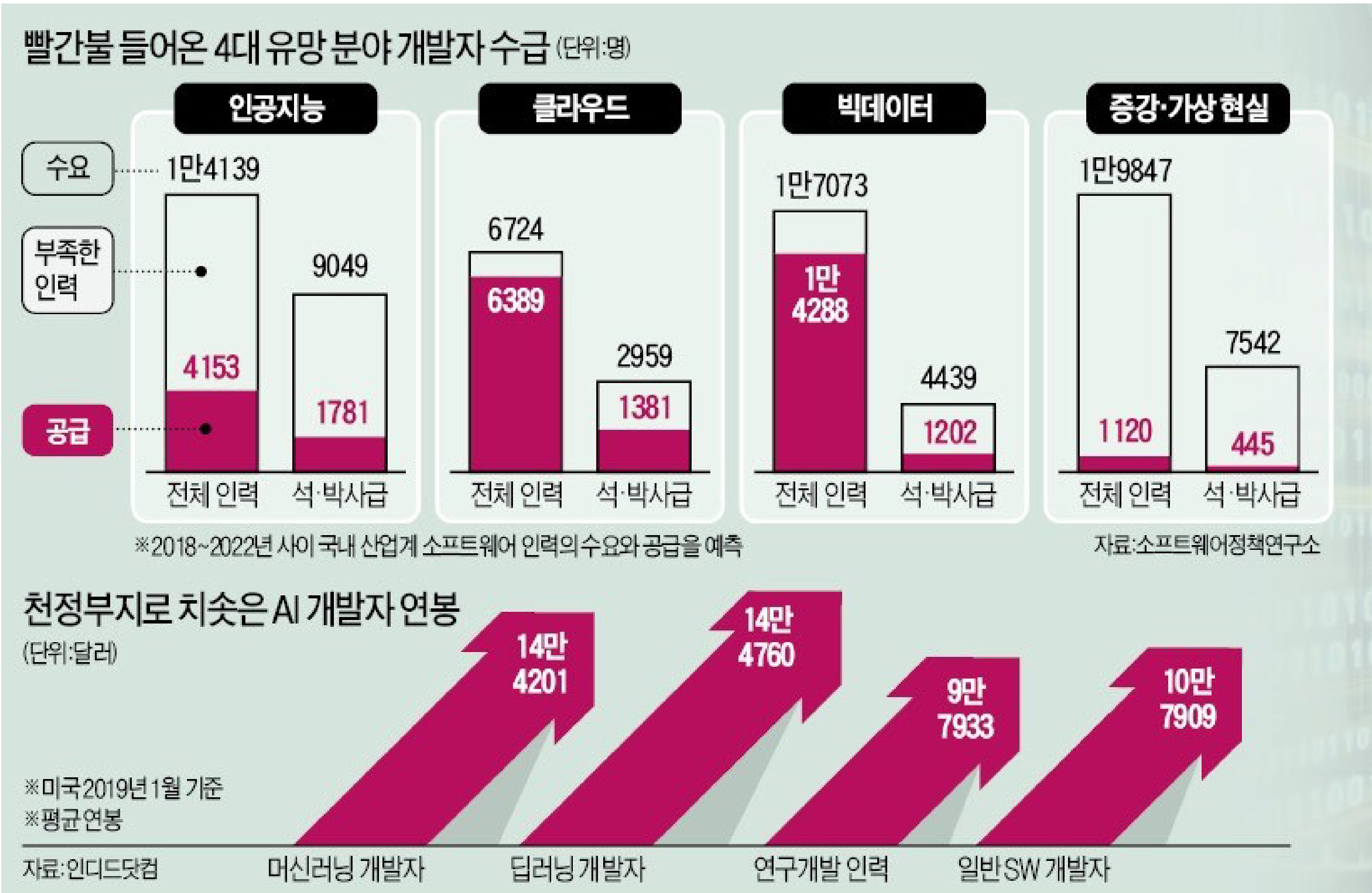
인공지능/딥러닝 부활의 이유



ML Doctor


인공지능과 의사의 뇌종양 진단 비교
정확도
시간
의사
93.9%
20-30분
인공
지능
94.6%
2분30초
자료= 미 뉴욕대 의대
글로벌 헬스케어 인공지능 시장 규모
13억
2019년
100억달러
2024년
자료=모건 스탠리

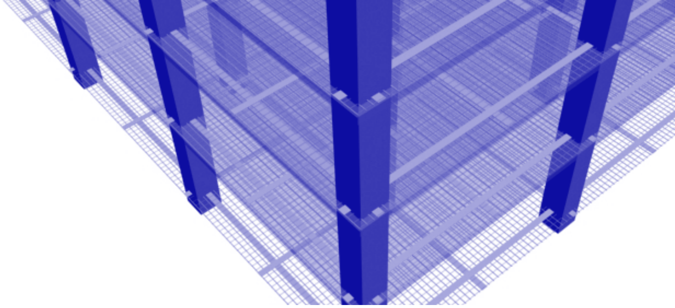
AI Human Resources In Great Need



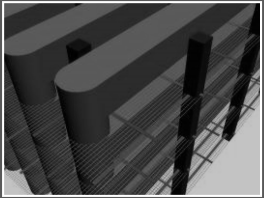
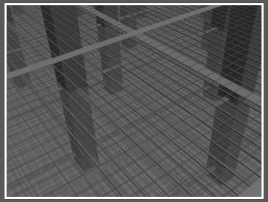
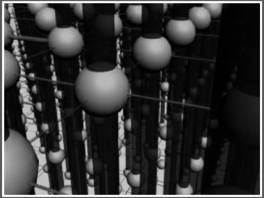
한국 AI 인재 구축의 문제점

- AI 인력을 확보하는 데 가장 큰 애로 요인
  - **실무형 기술인력 부족 (36.7%)**
  - 선진국 수준의 연봉 지급이 어렵다 (25.5%)
  - 대학원 등 전문 교육기관 및 교수 부족 (22.2%)
  - 근로시간 등 경직된 근무환경 및 조직문화 (6.7%)
  - 예산지원, 규제 완화 등 정부 지원 부족 (6.7%)

- 국내 AI 인재 육성을 위해 가장 필요한 것
  - **교육 인프라 확대 (37.8%)**
  - 기술혁신 및 신산업 창출을 저해하는 규제 완화'(21.1%)
  - AI 관련 스타트업 창업 및 기업의 AI 인재 육성 제도적 지원'(13.3%)
  - AI 인재 유치를 위한 근로환경 및 기업문화 조성'(12.2%)



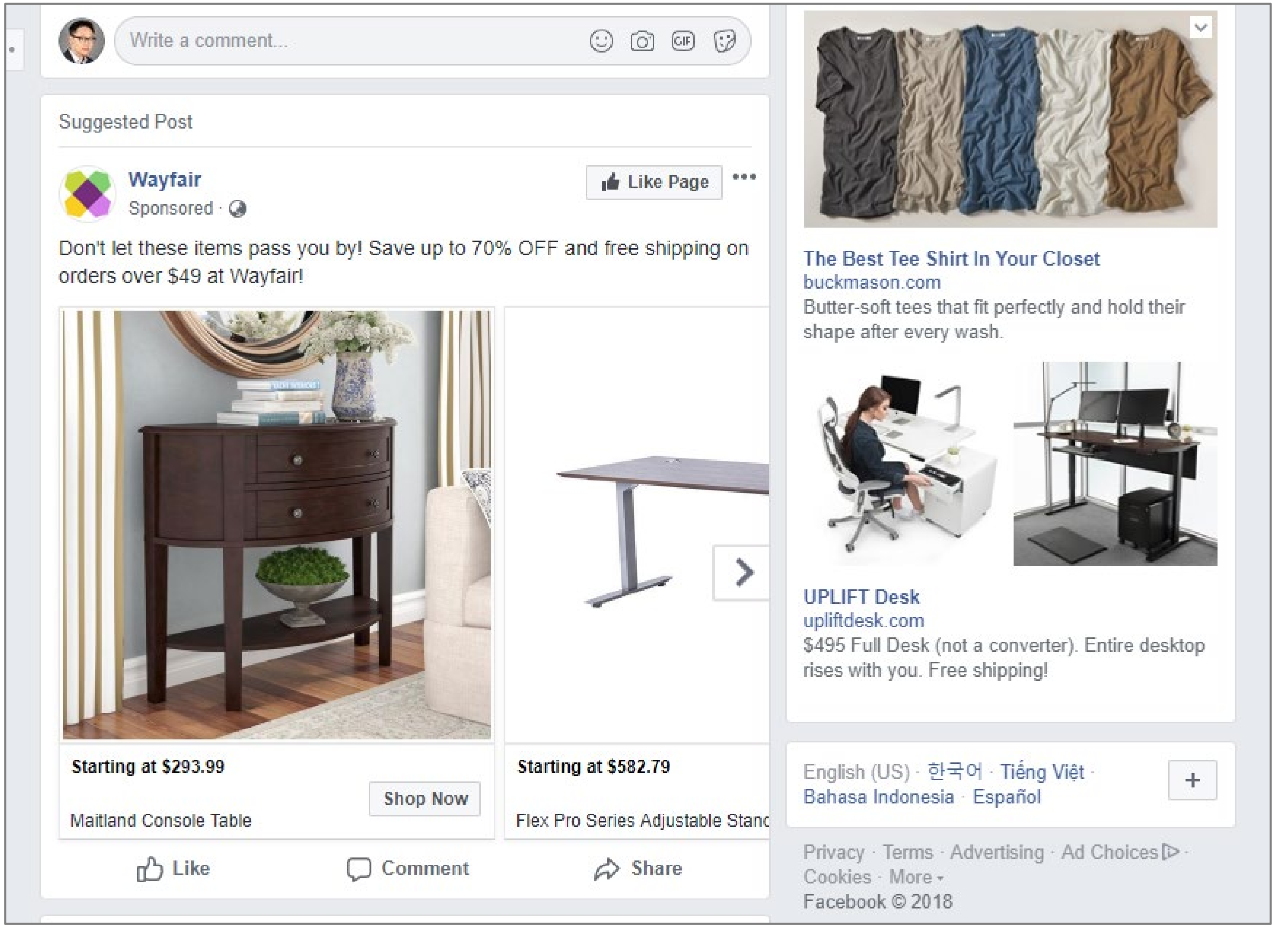
2. 기계학습의 종류

CNN, RNN GAN, Autoencoder, Unet

Convolutional Neural Network (CNN)

Yann LeCun's Job @ Facebook



Suggested Post, Banner Ads

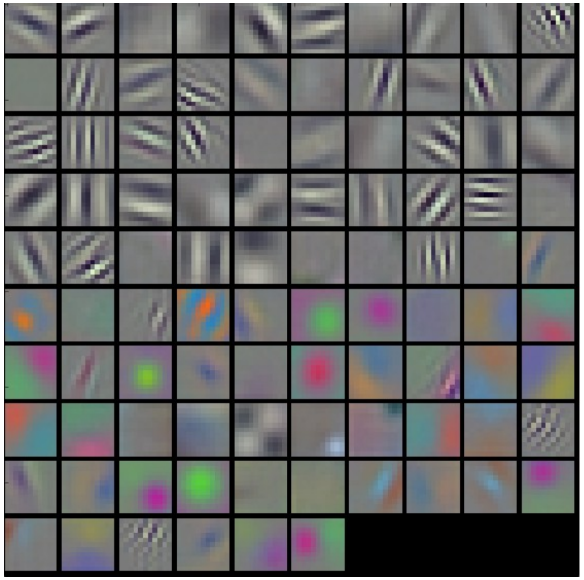
Image Recognition

- **Crucial in many ML applications**

- **Main challenge: # of pixels!!!!!!**

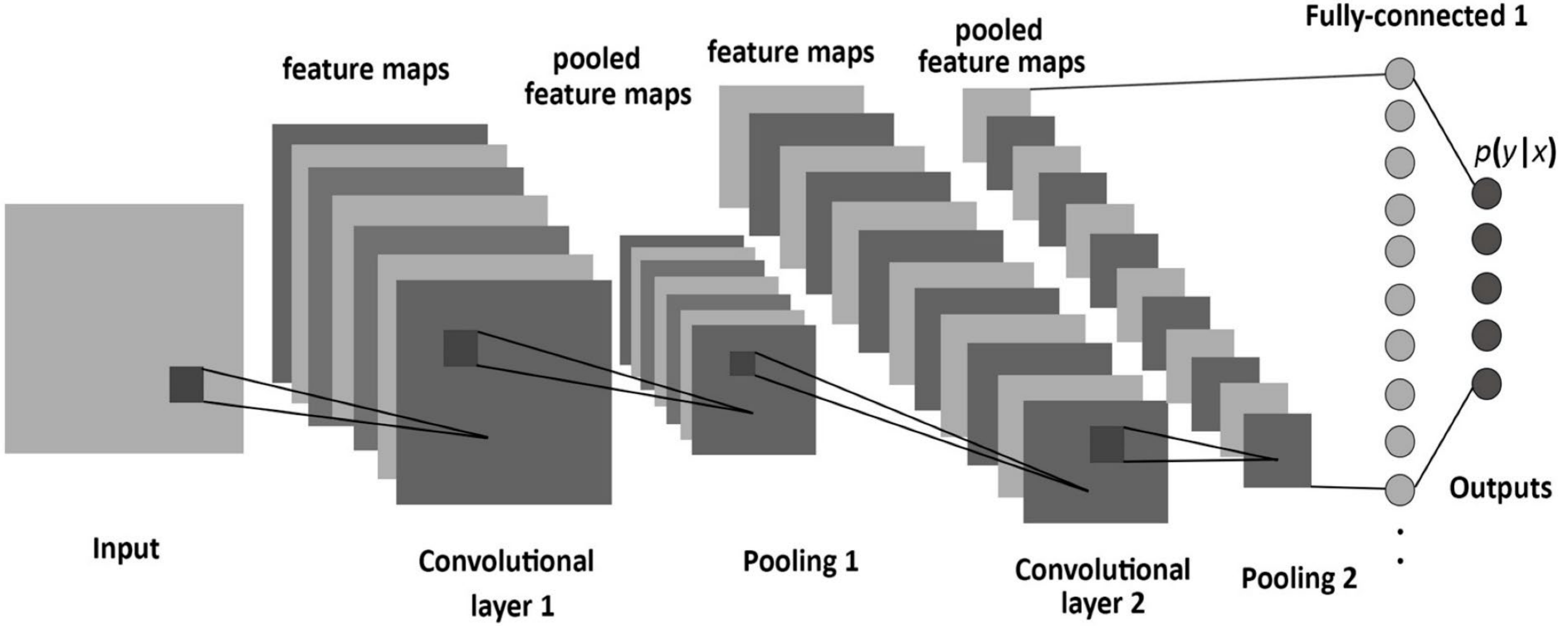
Convolutional Neural Network (CNN)

- **What is Convolution?**
  - 영상으로부터 특정 feature들을 추출하기 위한 필터를 구현할 때 convolution을 사용
- **Why CNN?**
  - Yann LeCun (1998)
  - 이미지 인식에서 MLP (Multi-Layer Perceptron)를 사용하게 되면, MLP는 모든 입력이 위치와 상관없이 동일한 수준의 중요도를 갖음
  - 이를 이용해 fully-connected neural network를 구성하게 되면 model 크기가 엄청나게 커지는 문제가 생김
  - 이에 대한 해결책으로 탄생한 것이 CNN



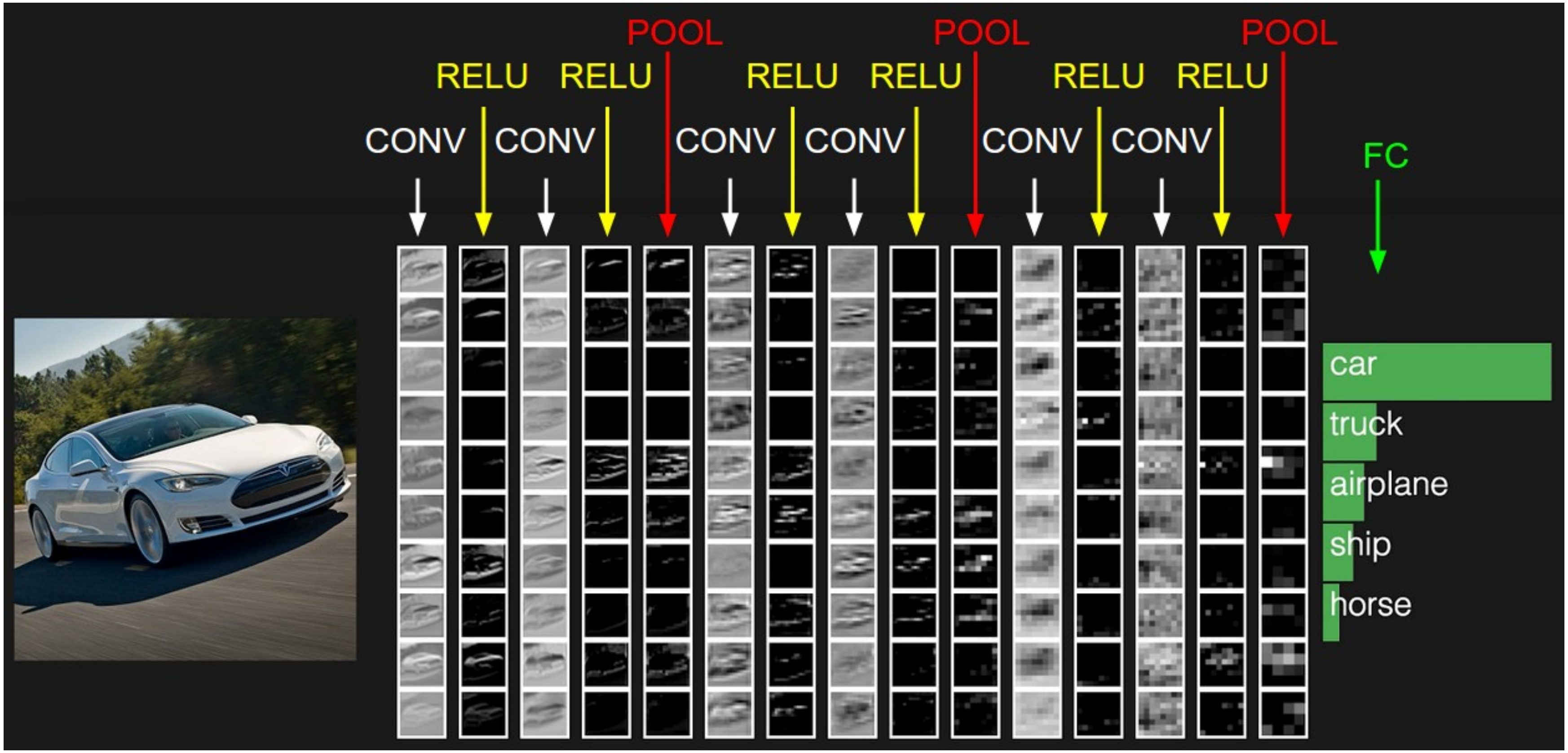
CNN Deep Learning

- **Feature Extraction**
  - **Key idea** to manage complexity while maintaining accuracy
  - Features are extracted via CONV:RELU:POOL



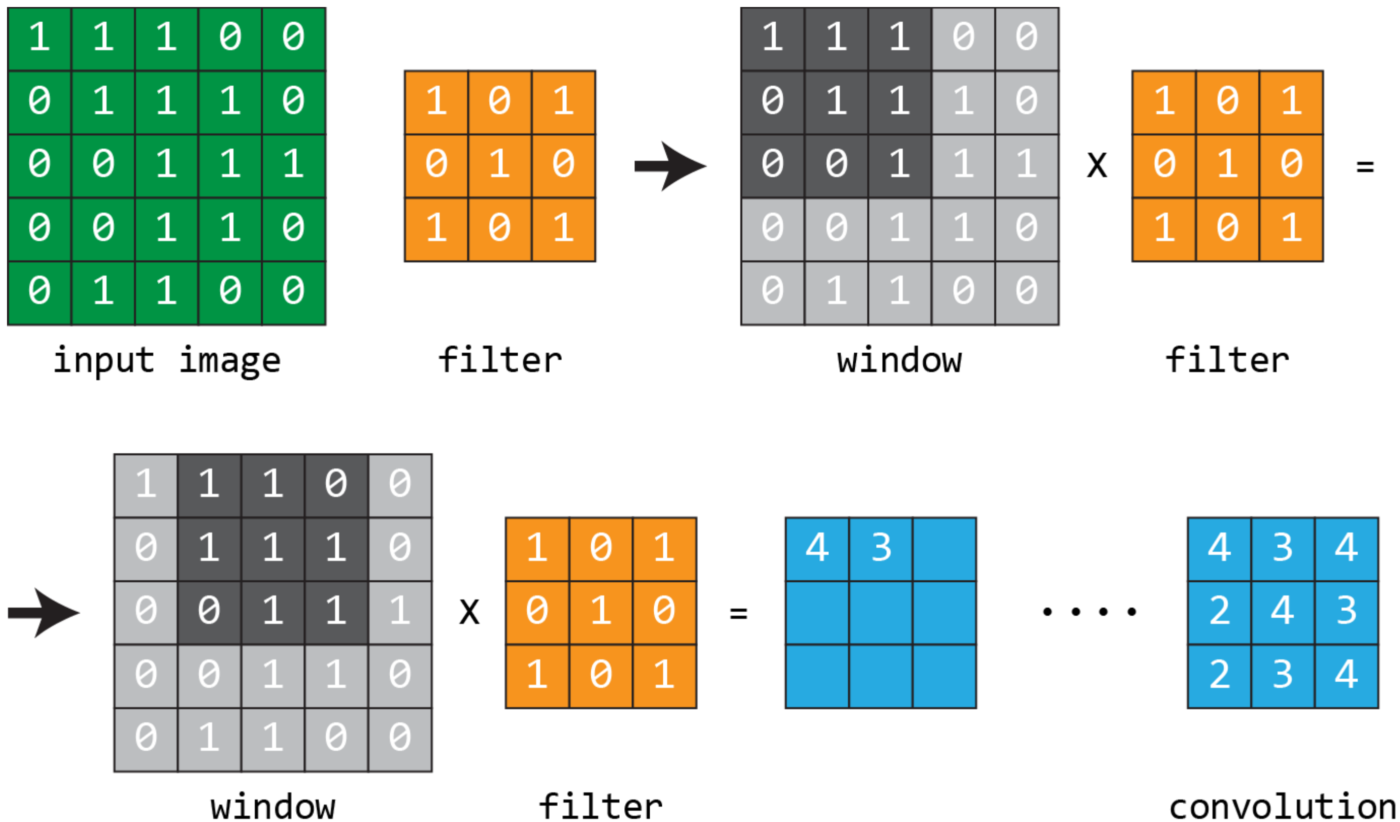
CNN Deep Learning Example

- **Hugely popular in image recognition**



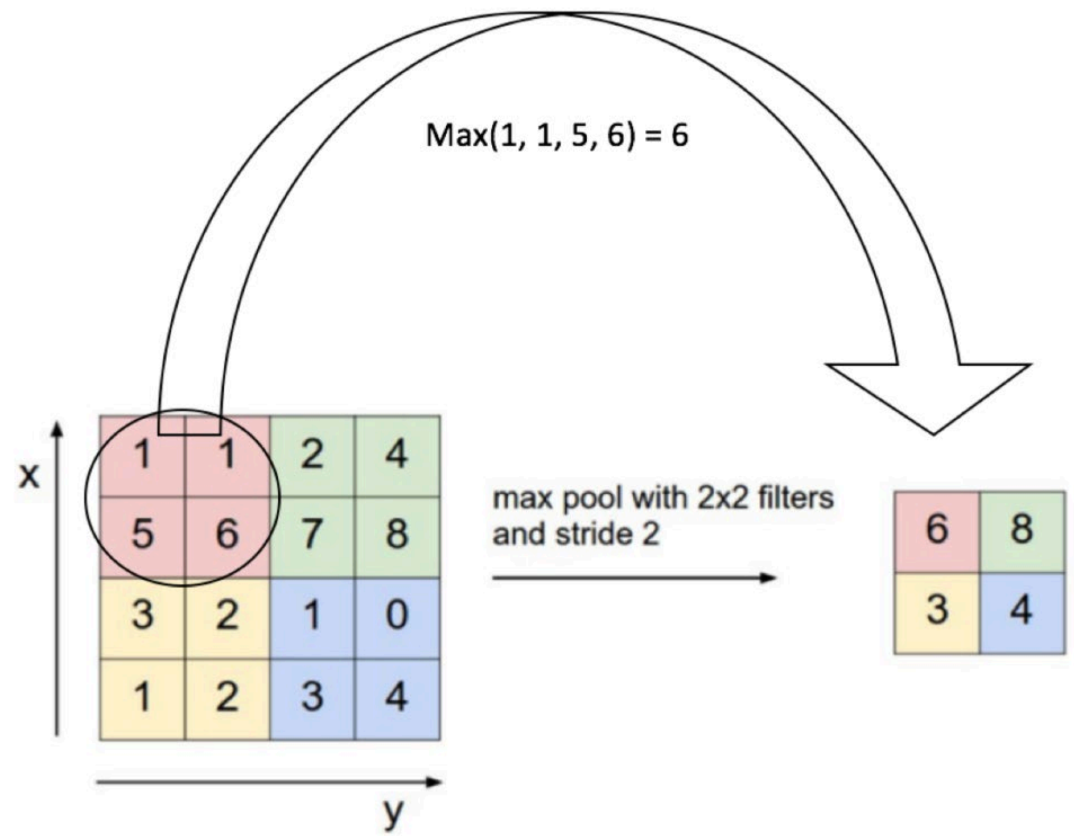
Convolution Example

- **Objective: to extract features from the input data**



Pooling

- **objective**
  - **reduce the feature map dimension and keep the important information**
  - **types: max, average, sum, etc.**



Rectified Feature Map

Recurrent Neural Network (RNN)

Image Description (not Recognition)

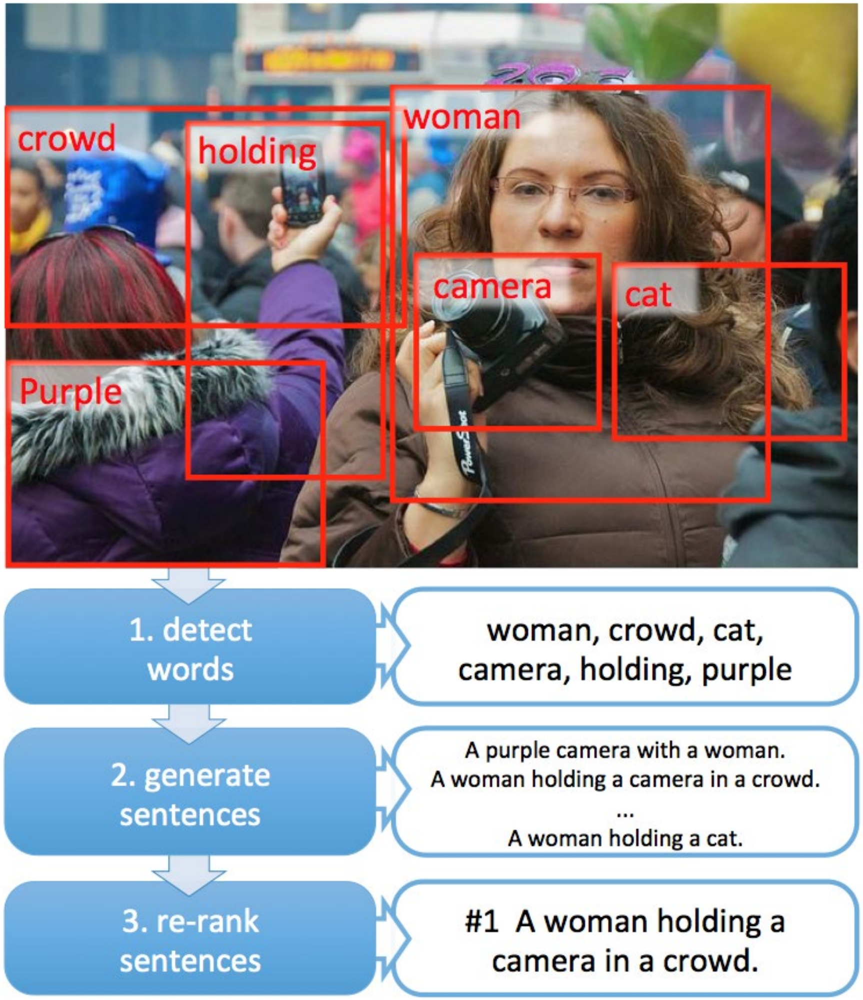
- Describe a given image using a sentence
  - Harder than image recognition/classification
  - Competition is ongoing and fierce among researchers

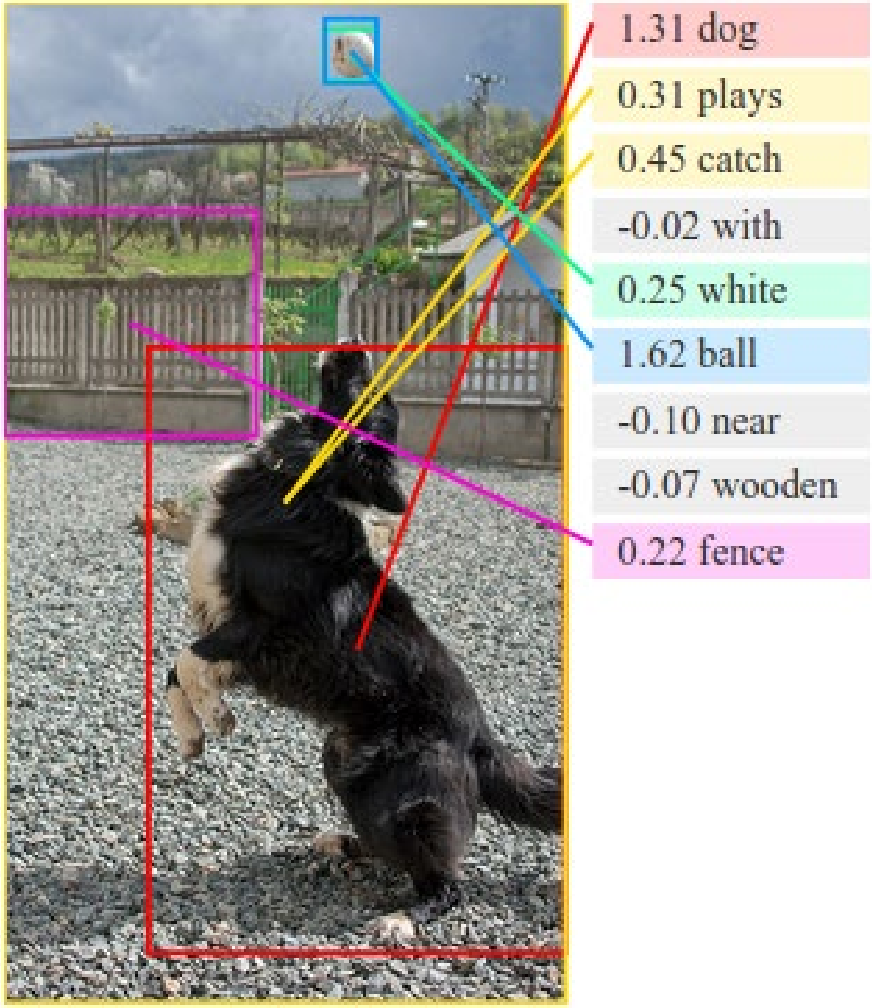
man in black shirt is playing guitar.

a young boy is holding a baseball bat.

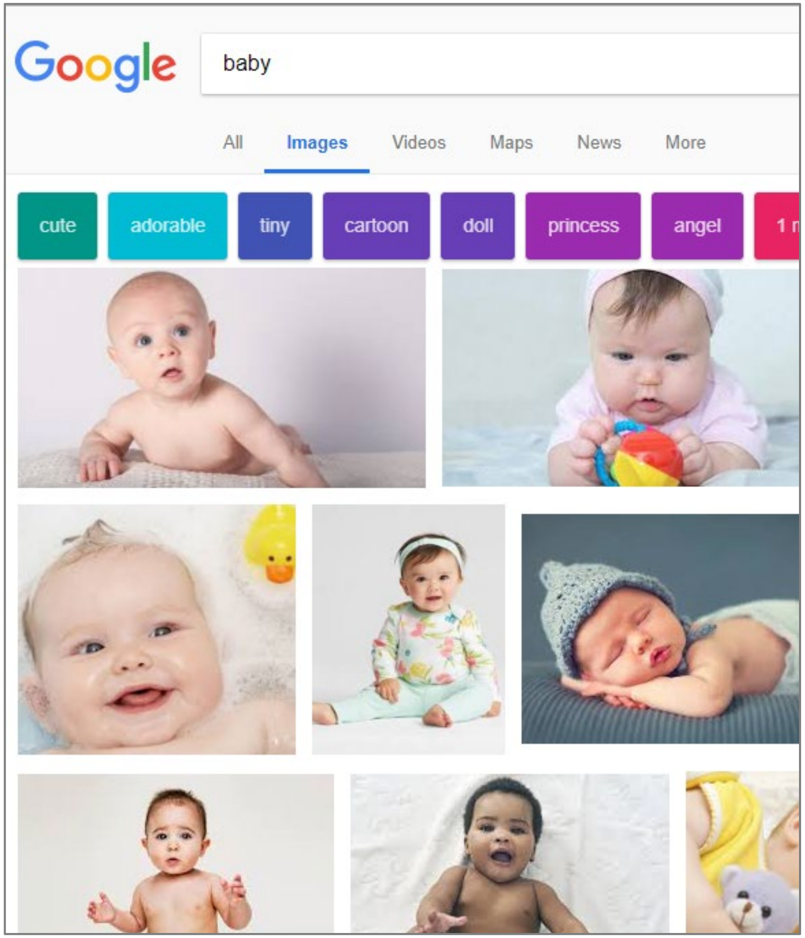
two young girls are playing with lego toy.

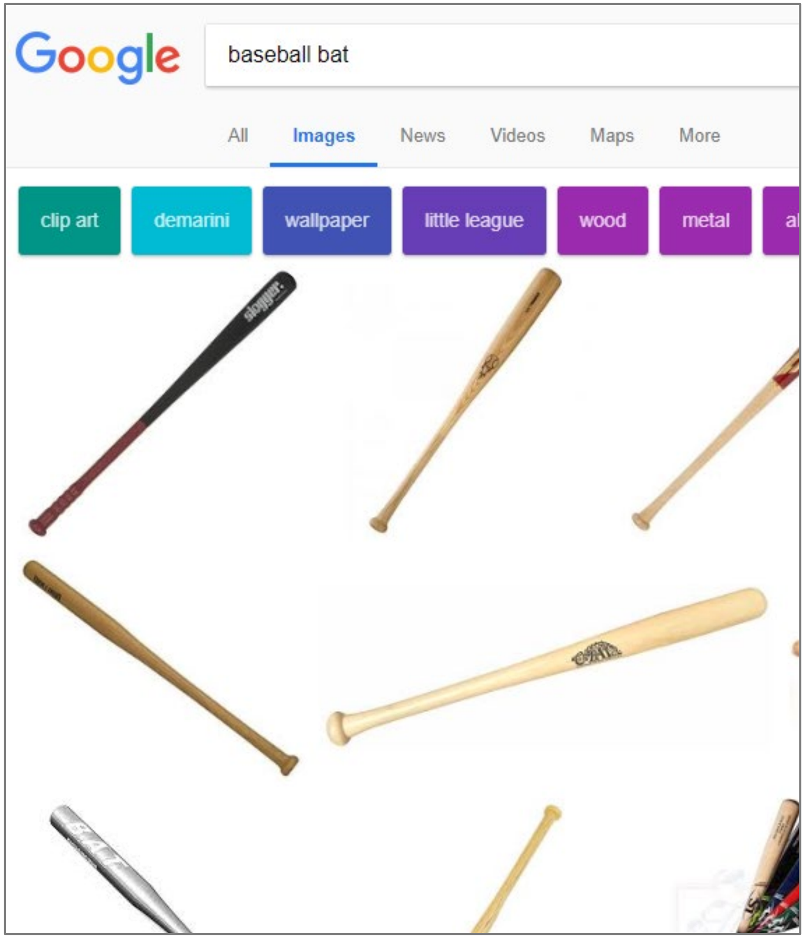
RNN Applications



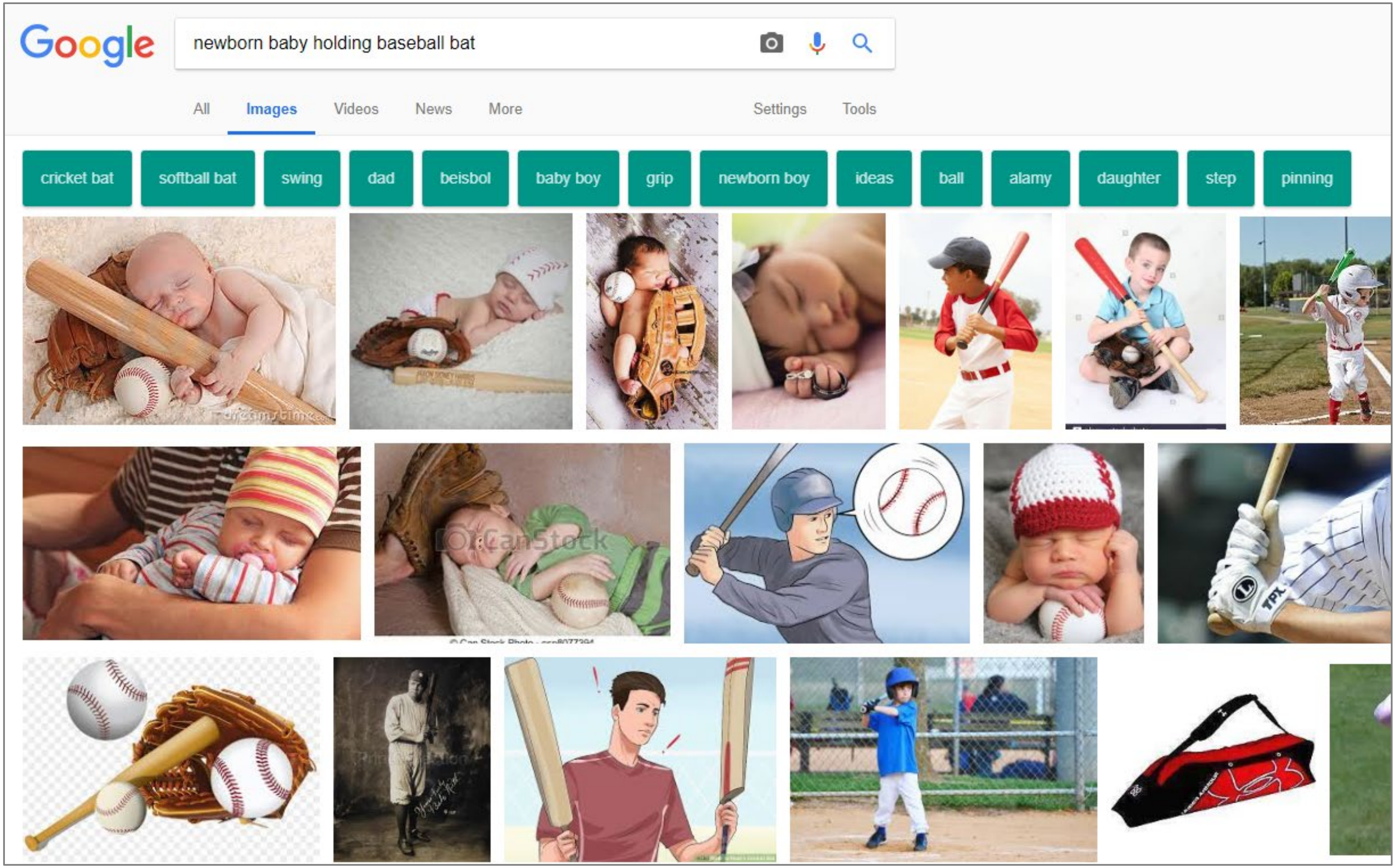


Simple Image Search





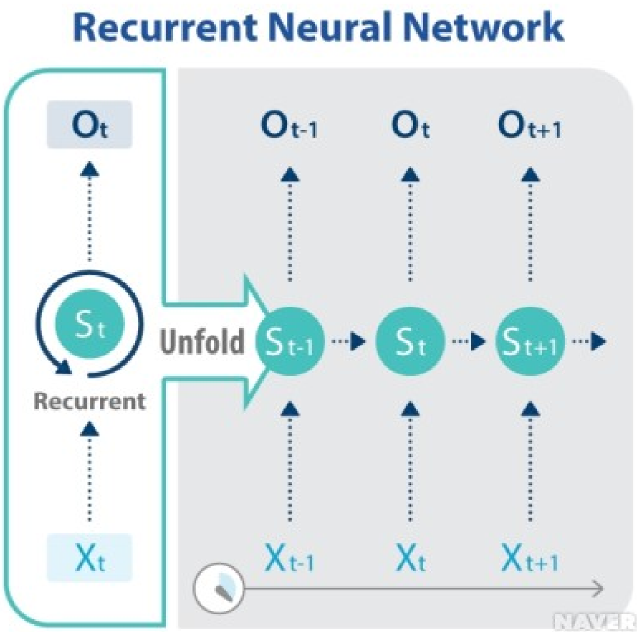
Complex Image Search

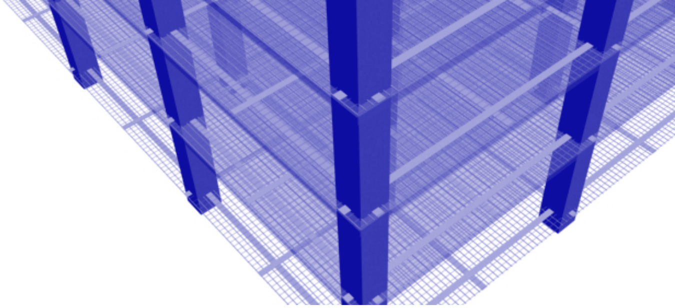


Recurrent Neural Network (RNN)

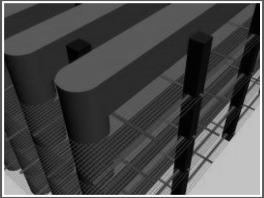
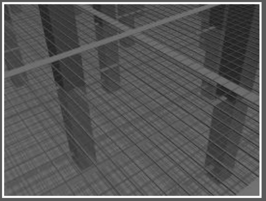
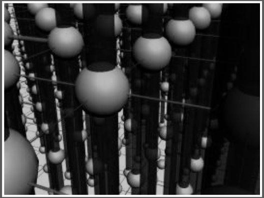
- **What is RNN?**
  - Hochreiter & Schmidhuber (1997)
  - A class of ANN where connections between units form a directed cycle.
  - This allows it to exhibit dynamic temporal behavior

- **Why RNN?**
  - Unlike feedforward neural networks, RNNs can use their internal memory to process arbitrary sequences of inputs
  - Applicable to tasks such as translation, speech recognition, etc

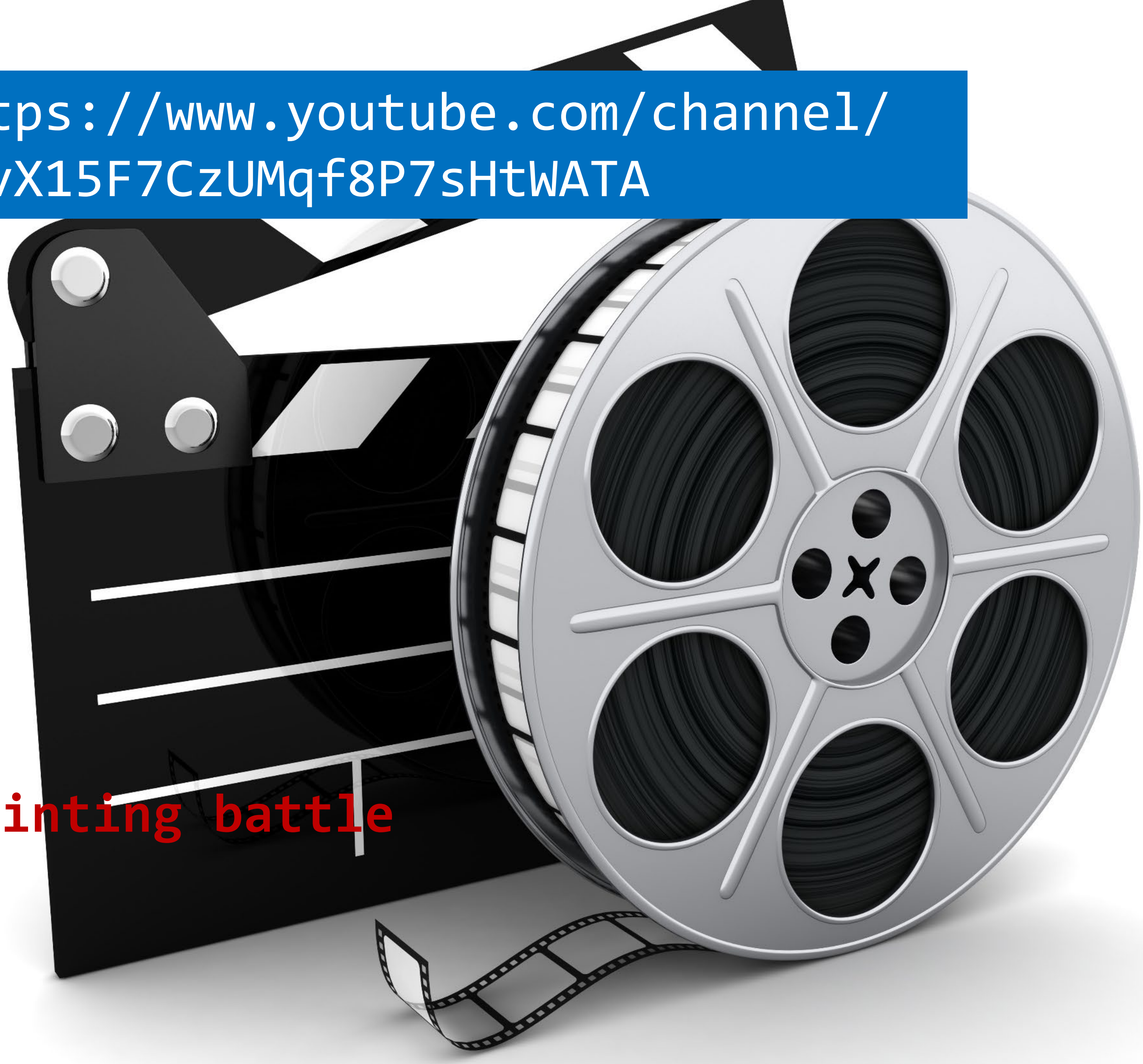




Generative Adversarial Network (GAN)

https://www.youtube.com/channel/UCvX15F7CzUMqf8P7sHtWATA



painting battle

Ian Goodfellow: Inventor of GAN

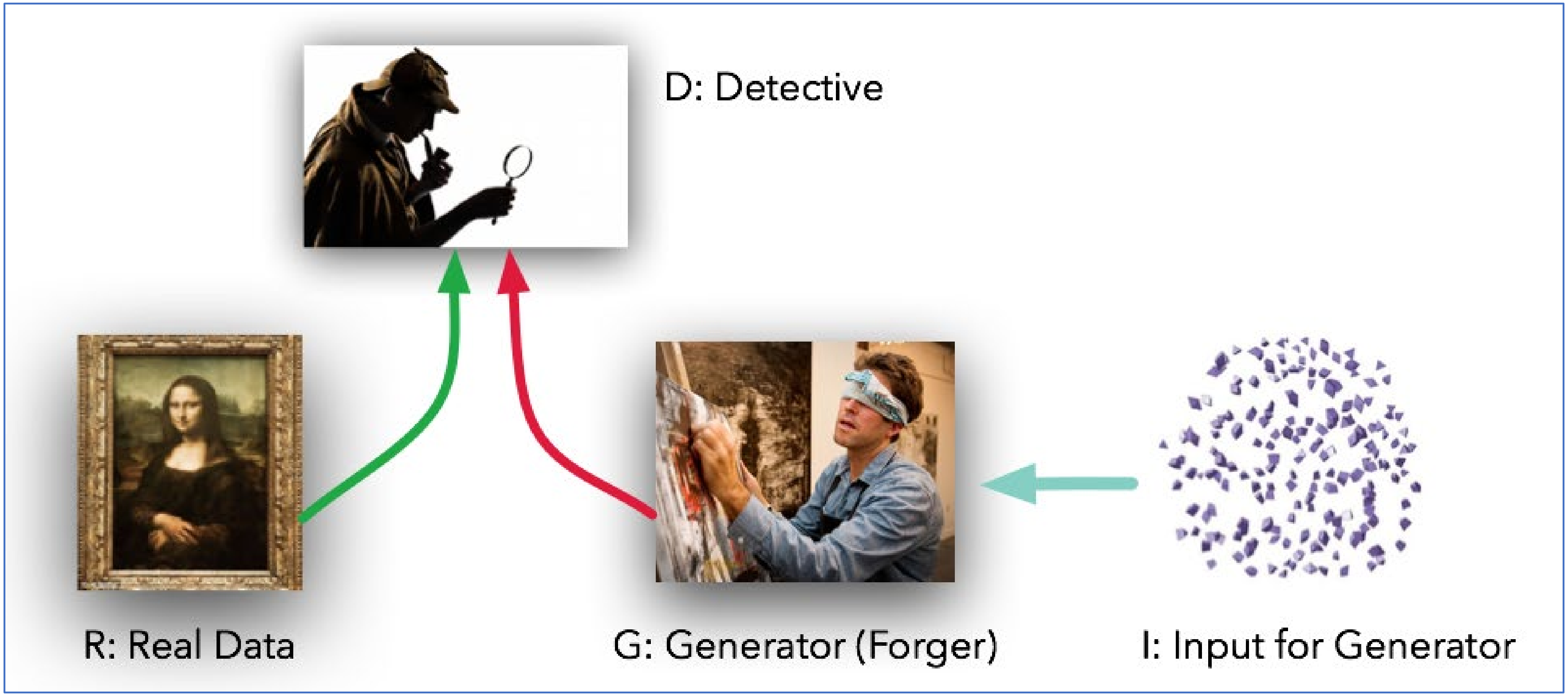
Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio

"Generative Adversarial Nets"

Int. Conf. on Neural Information Processing Systems, 2014.

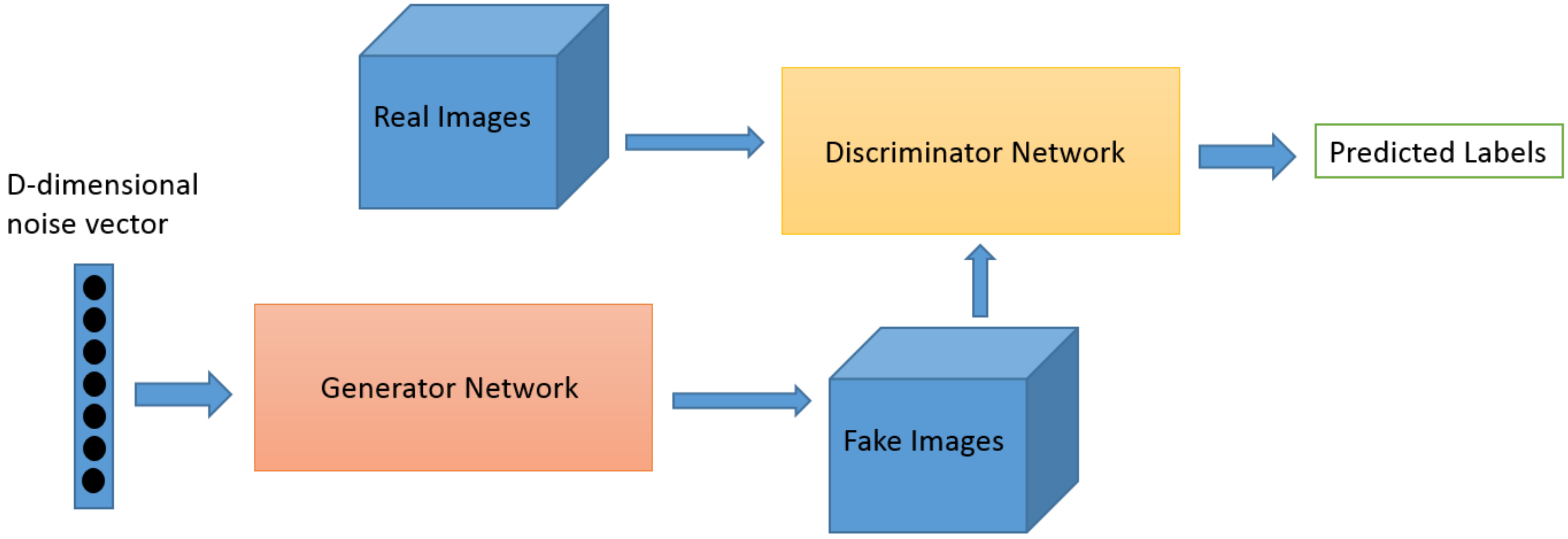
Main Idea

- Two neural networks contest with each other in a game
  - Generator generates fake data, hoping to fool discriminator
  - Discriminator evaluates them for authenticity



GAN Training

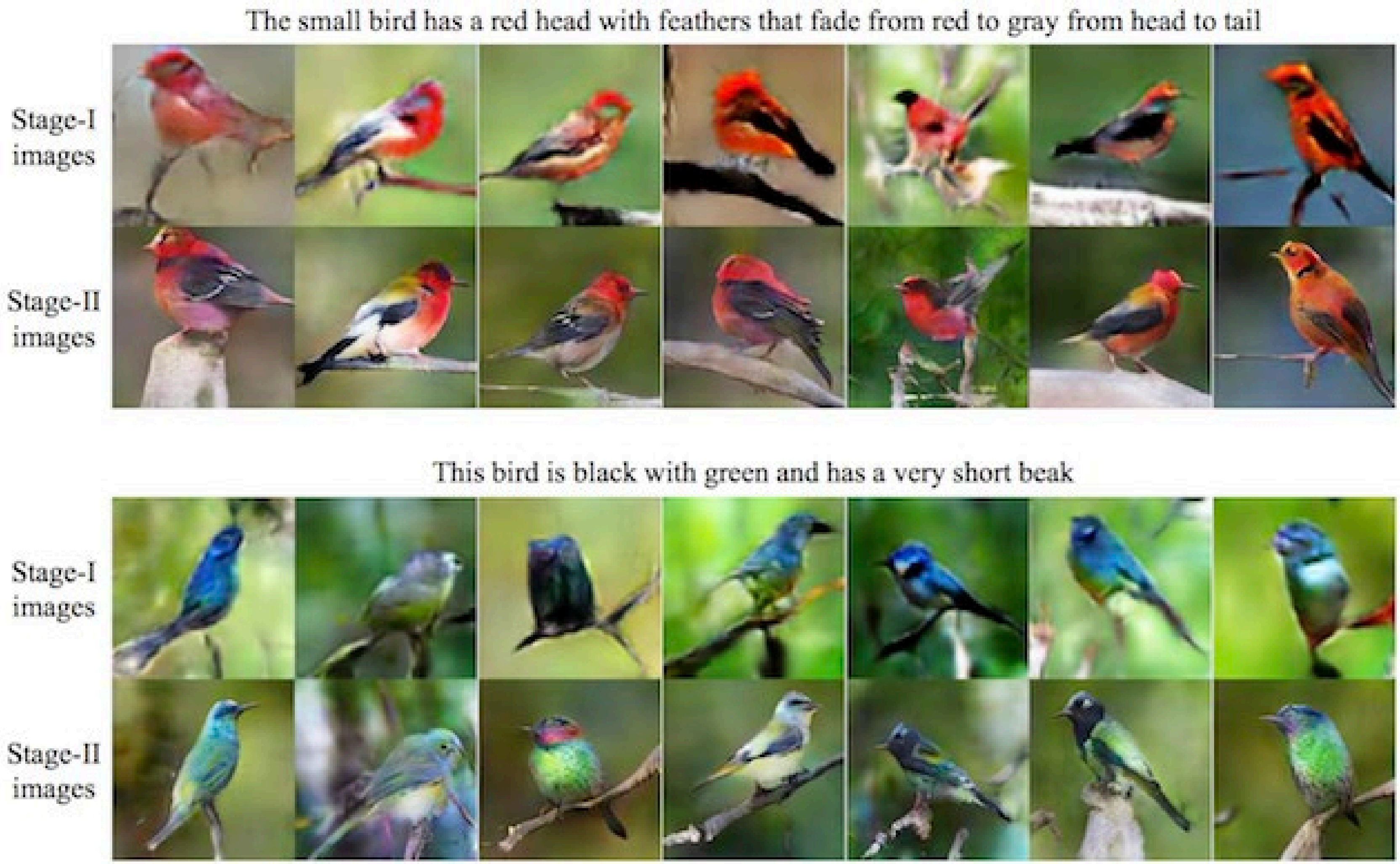
- **In case we deal with image data:**
  - **The generator takes in a random vector and returns a fake image.**
  - **This fake image and the real dataset are fed to the discriminator.**
  - **The discriminator picks one real image and the fake image and returns probabilities, a number between 0 (fake) and 1 (real).**



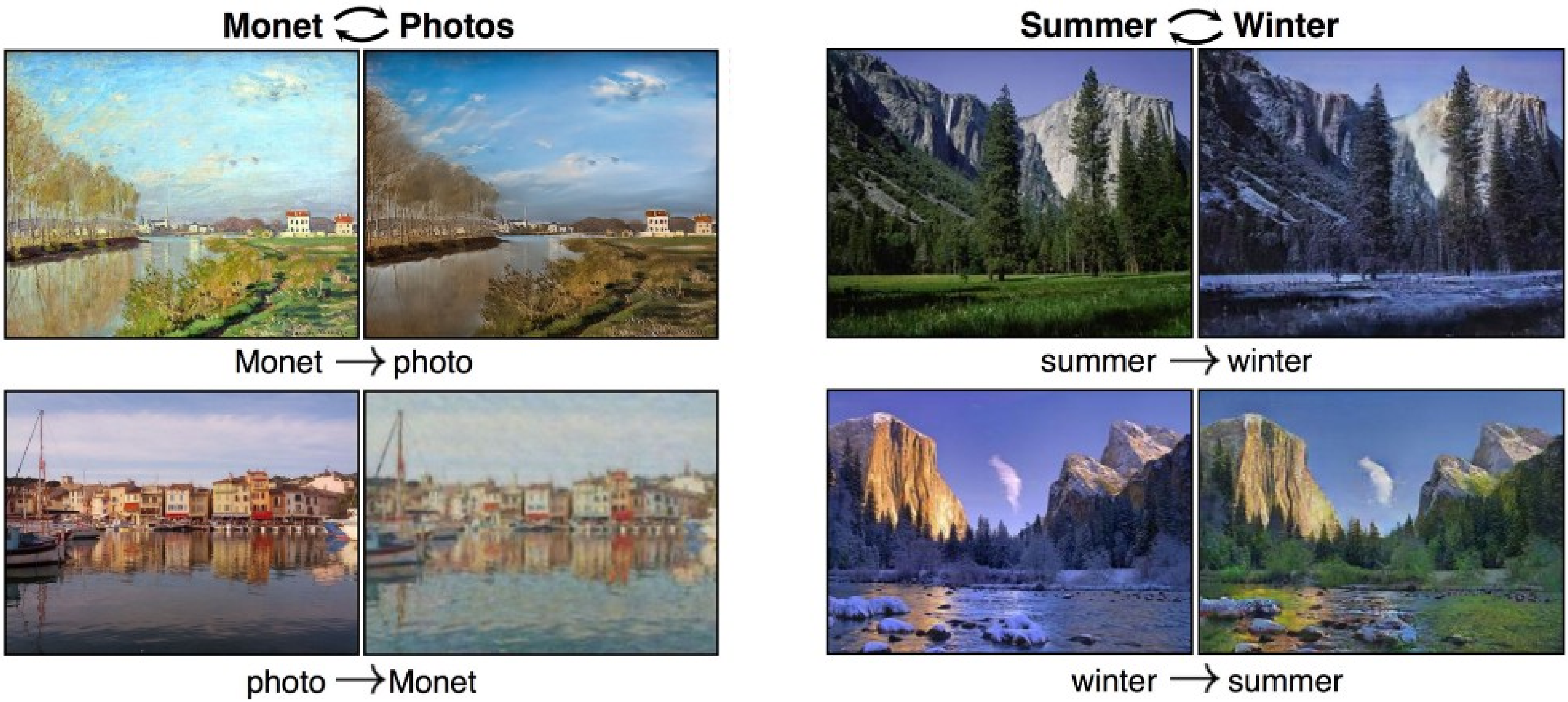
These Are Fake Photos by GAN

GAN Can Do Others, Too

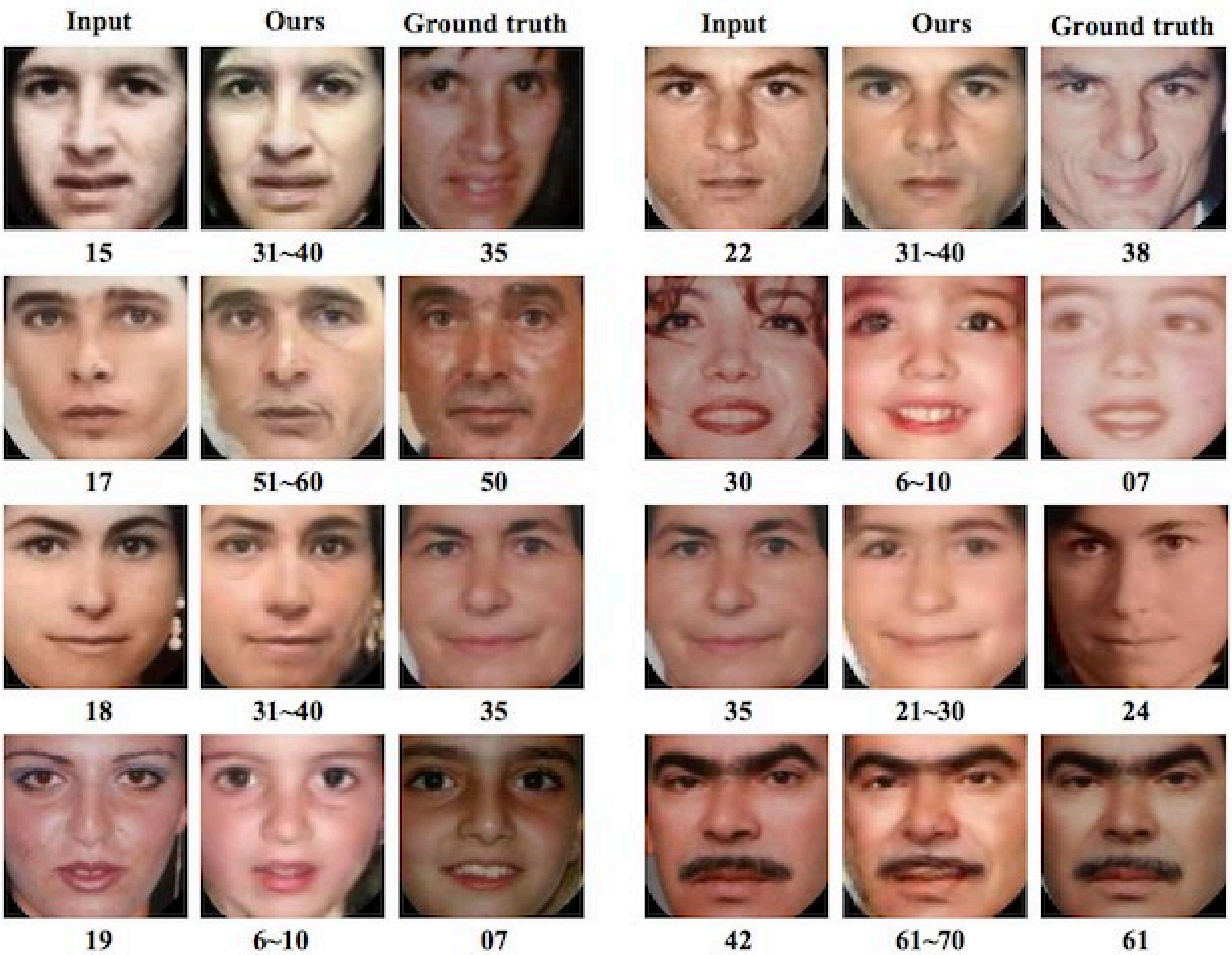
Text-to-Image Translation with GAN



GAN Can Cross Domains

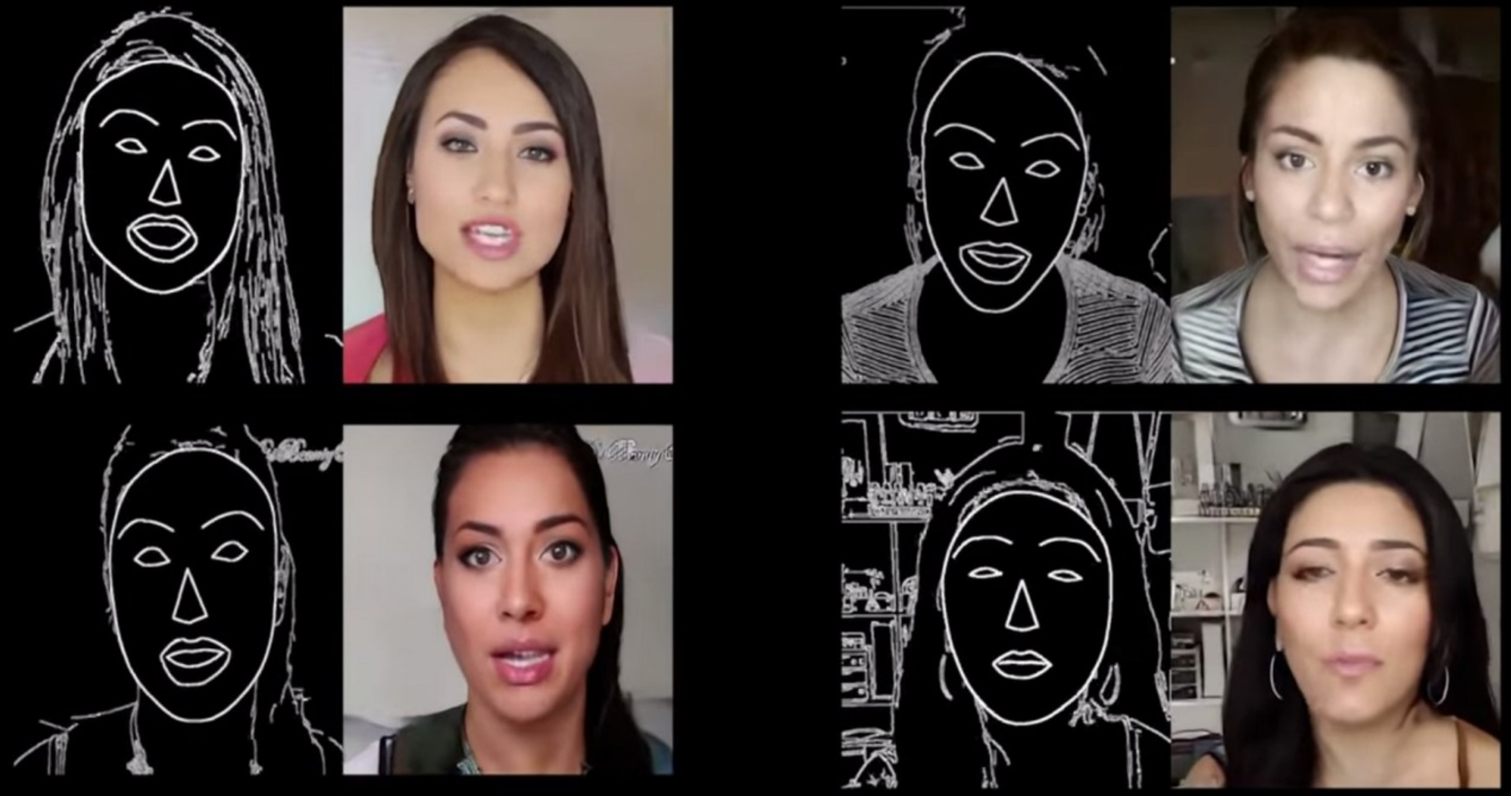


Face Aging Prediction



Next? Yup, Videos

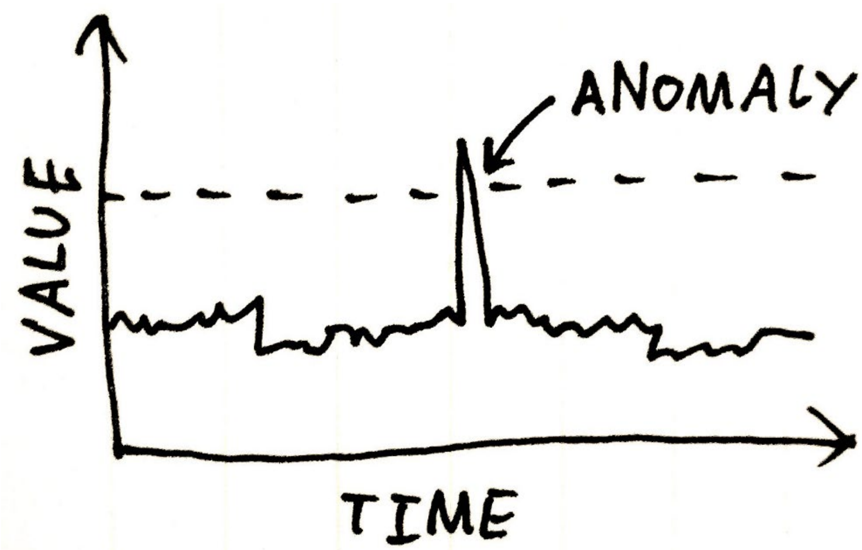
https://www.youtube.com/watch?v=8AZBuyEuDqc



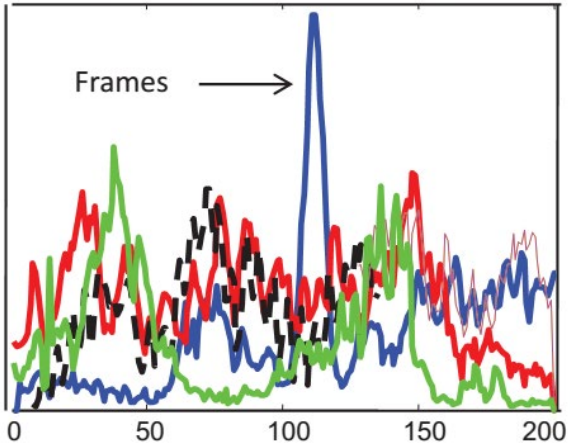
Autoencoder

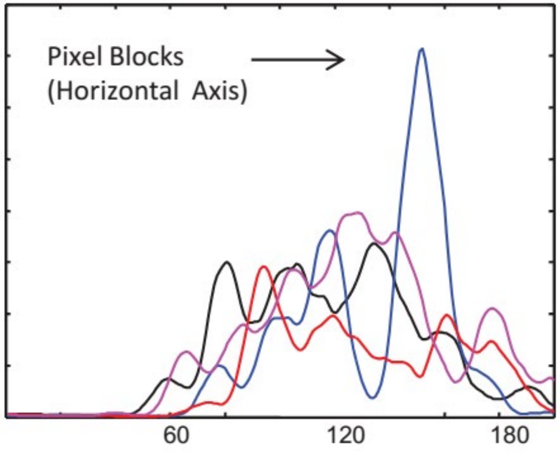
Anomaly Detection

- **Definition**
  - Identification of rare items, events or observations
  - They raise suspicions by differing significantly from the majority of the data
  - Typically the anomalous items will translate to some kind of problem
  - Can you think of killer apps?

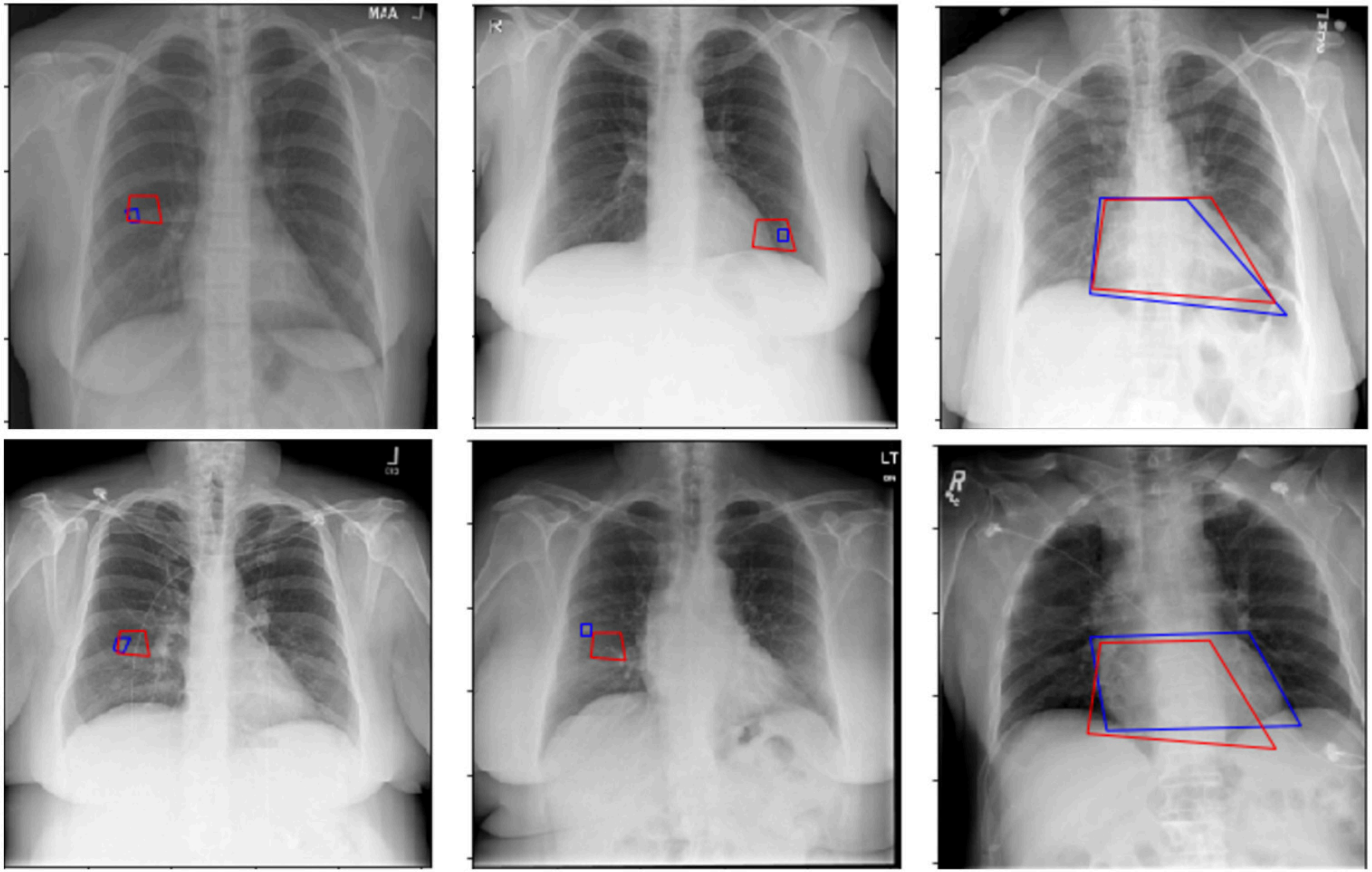


Video Analytics with ML

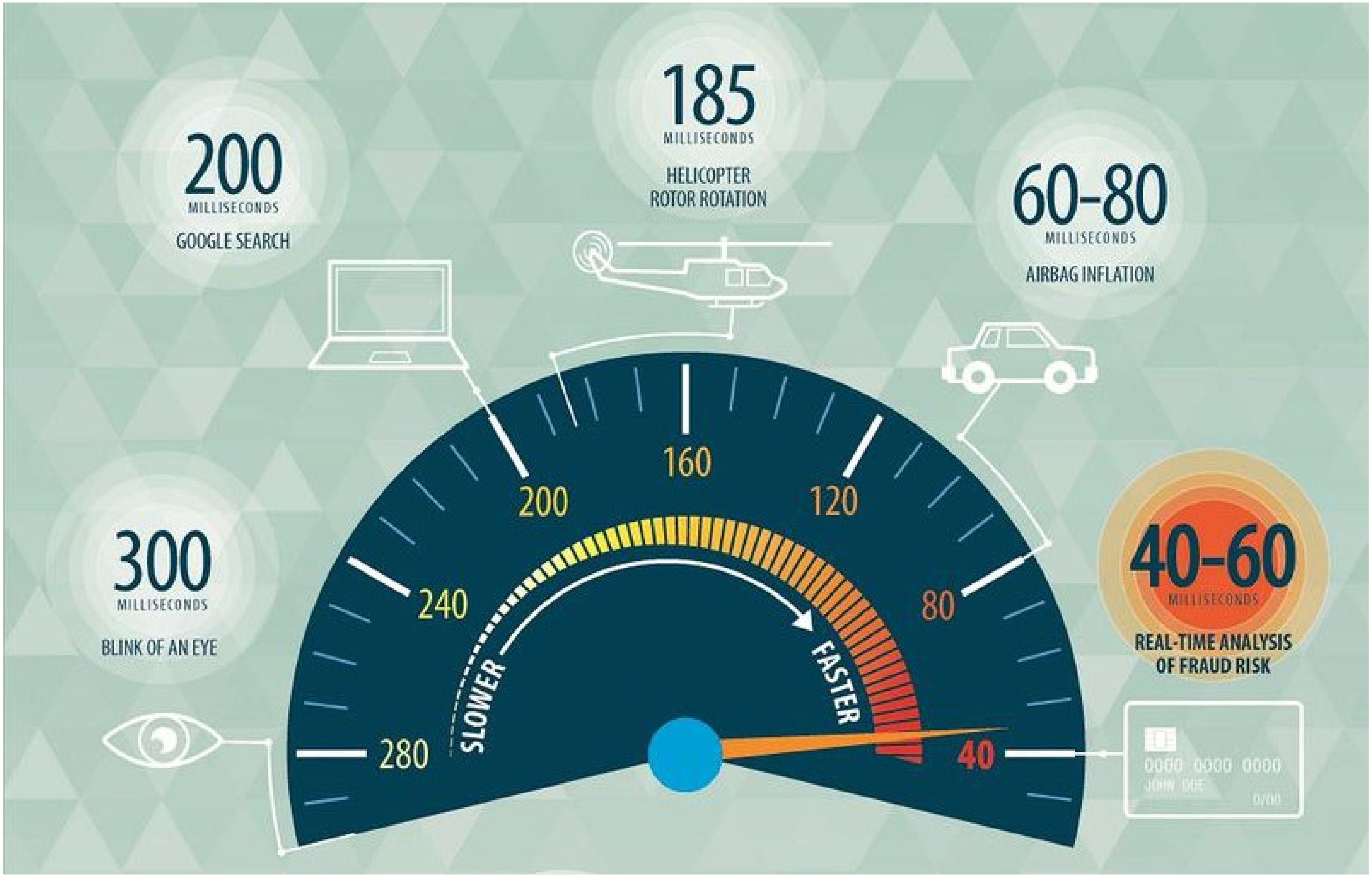




Applications in Medical Imaging



How Fast Is Credit Card Fraud Detection?

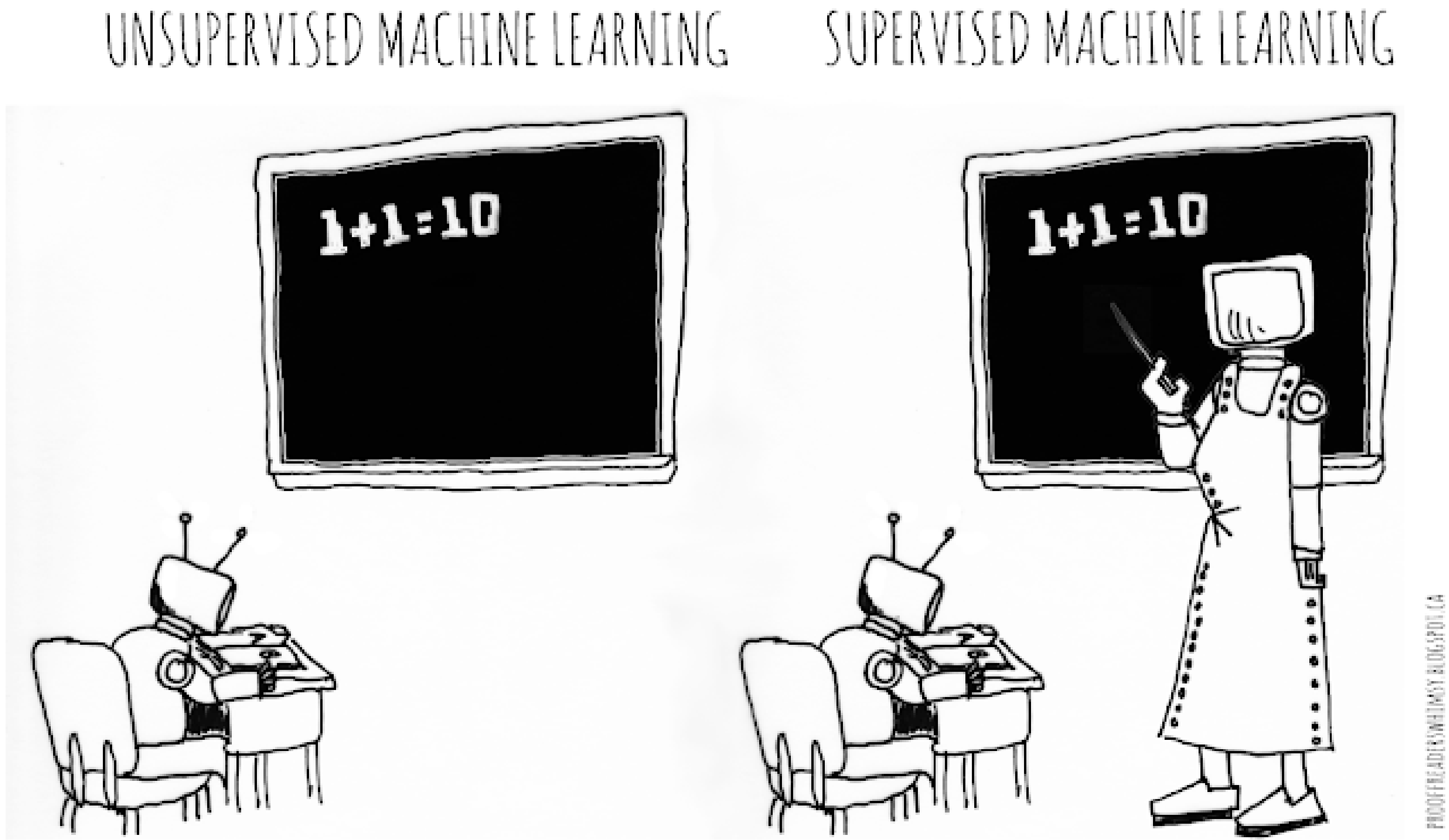


Credit Card Fraud Detection DB

- **Offered by Kaggle**
  - **September 2013 by European cardholders**
  - **Transactions occurred in two days**
  - **492 frauds out of 284,807 transactions**
  - **Highly unbalanced, the positive class (frauds) account for 0.172%**
  - **Due to confidentiality issues, details for 28 features are hidden**

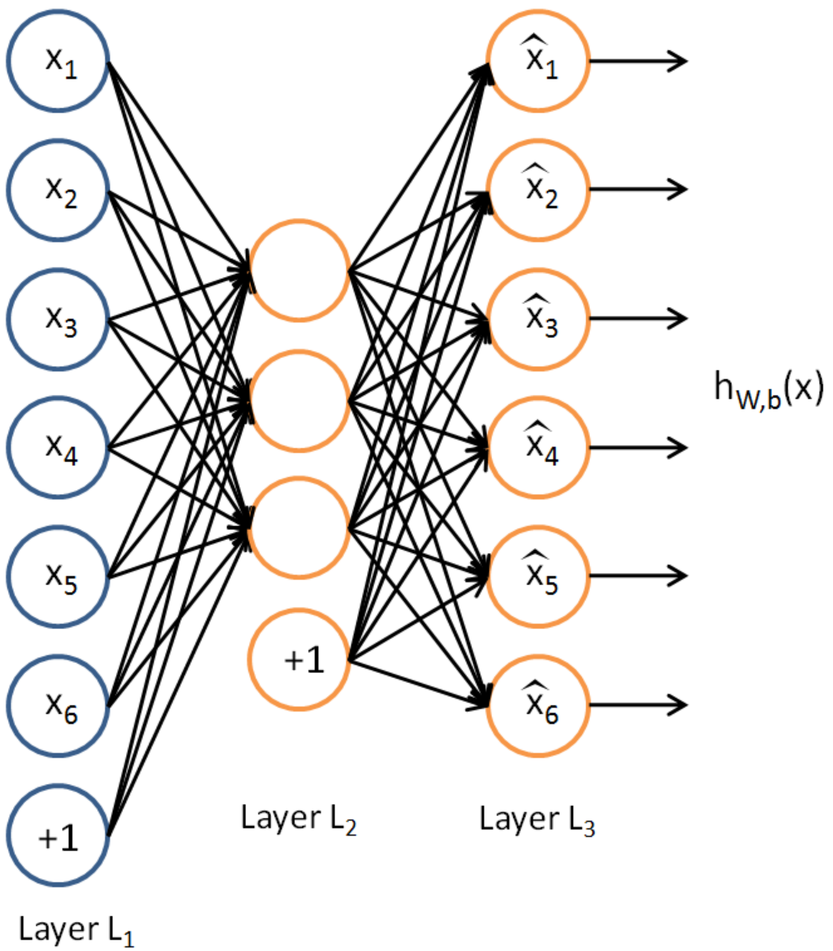
Unsupervised Learning

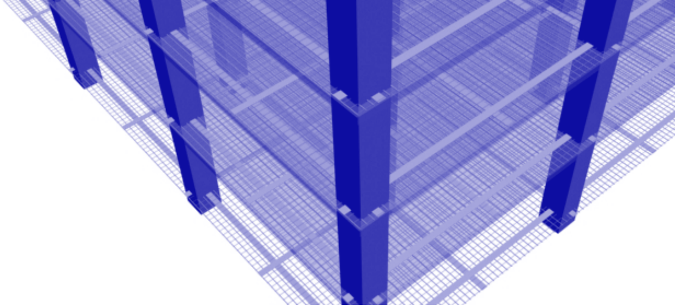
- No labels in the database, but it learns by itself!



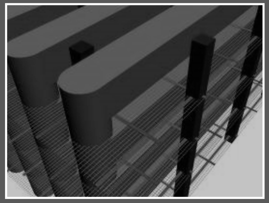
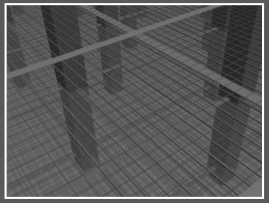
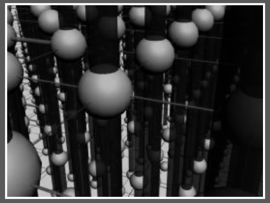
Autoencoder

- **Hidden layer is smaller**
  - **For dimensionality reduction**
  - **Divided in to encoder (first half) and decoder (second half)**
  - **Useful in feature extraction (encoder)**
  - **Useful in data reconstruction (decoder)**
  - **Killer apps in image processing**
  - **Word imbedding uses AE!**



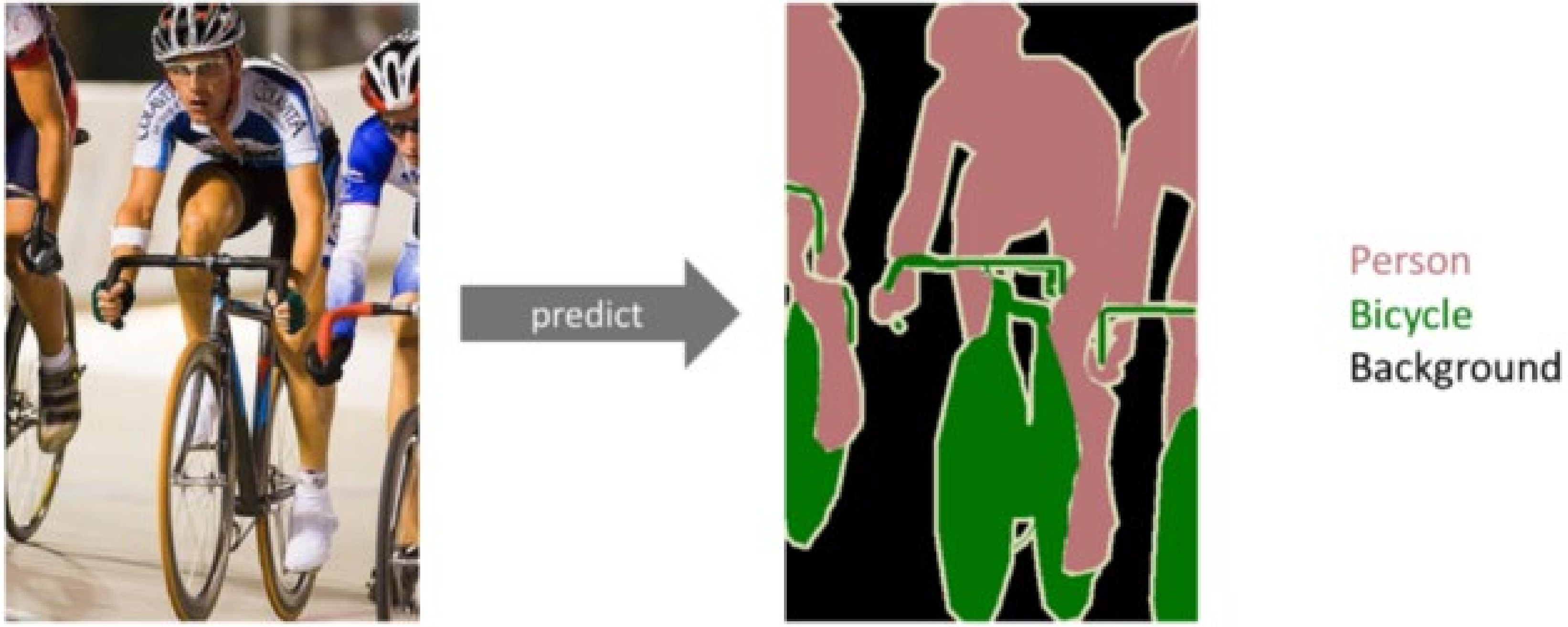


Unet

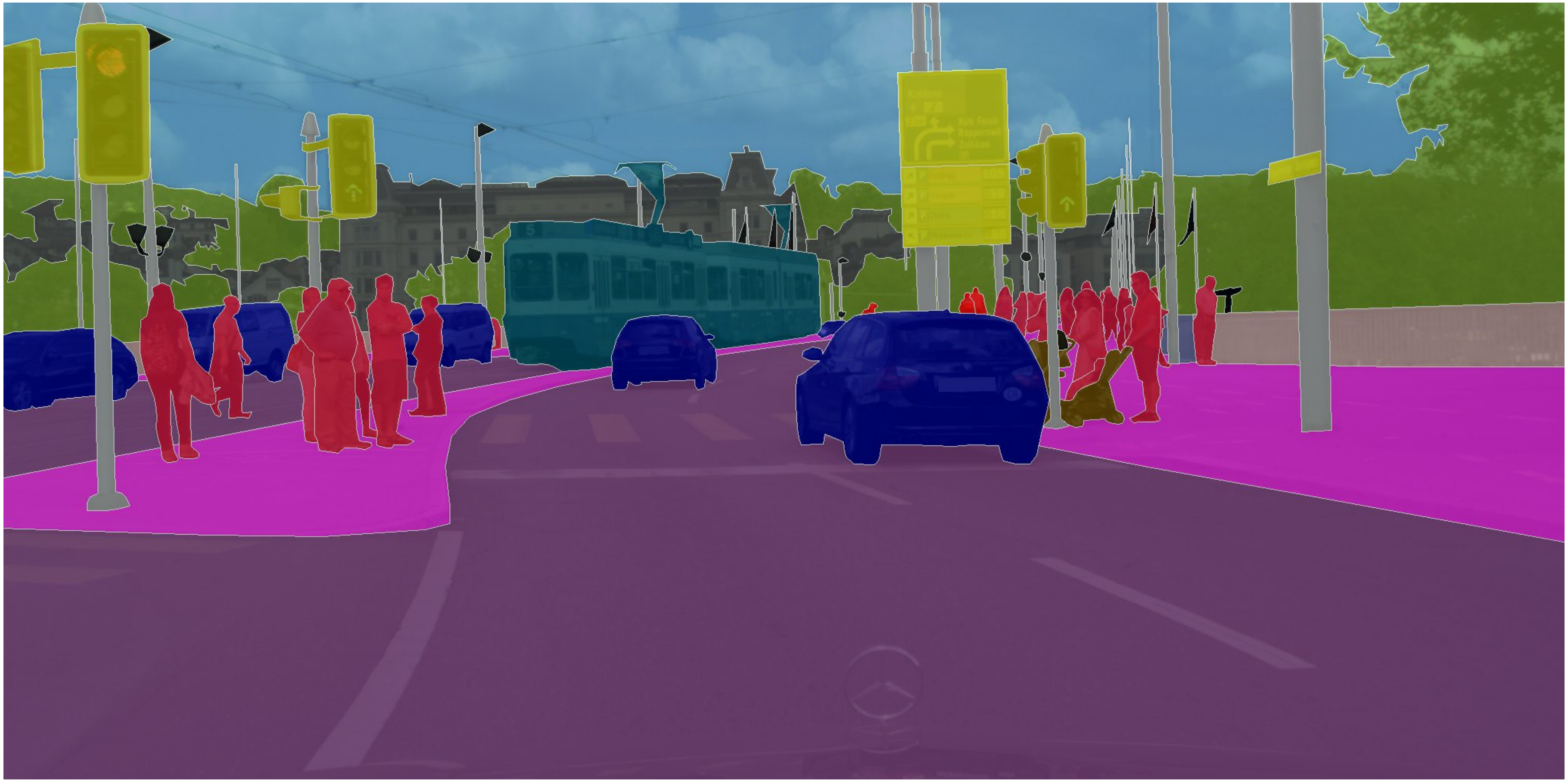
Semantic Segmentation

- **Smart compute vision application**
  - **Label each pixel with a corresponding class of what is being represented**
  - **Commonly referred to as dense prediction.**

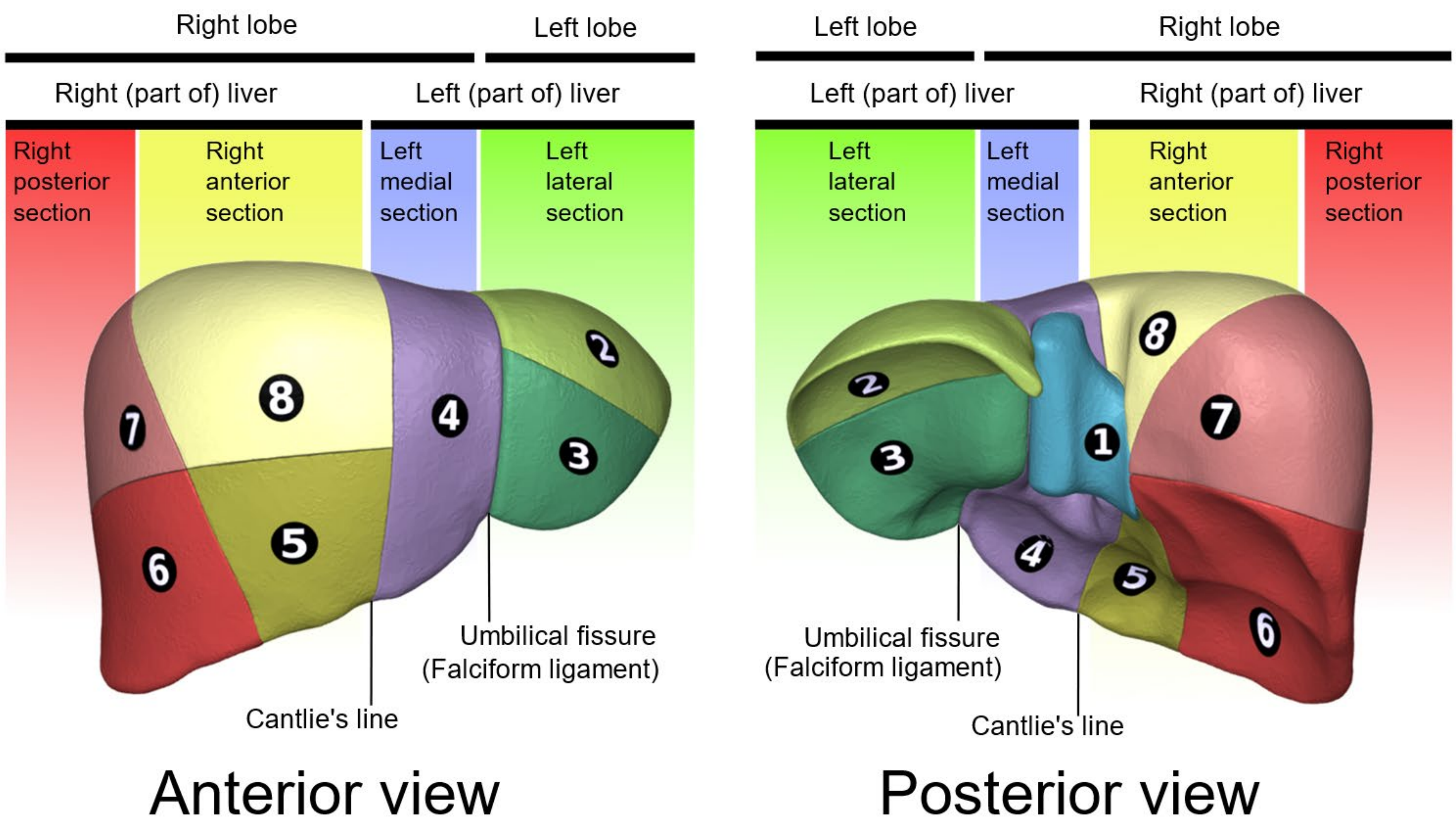


Application in Autonomous Vehicle

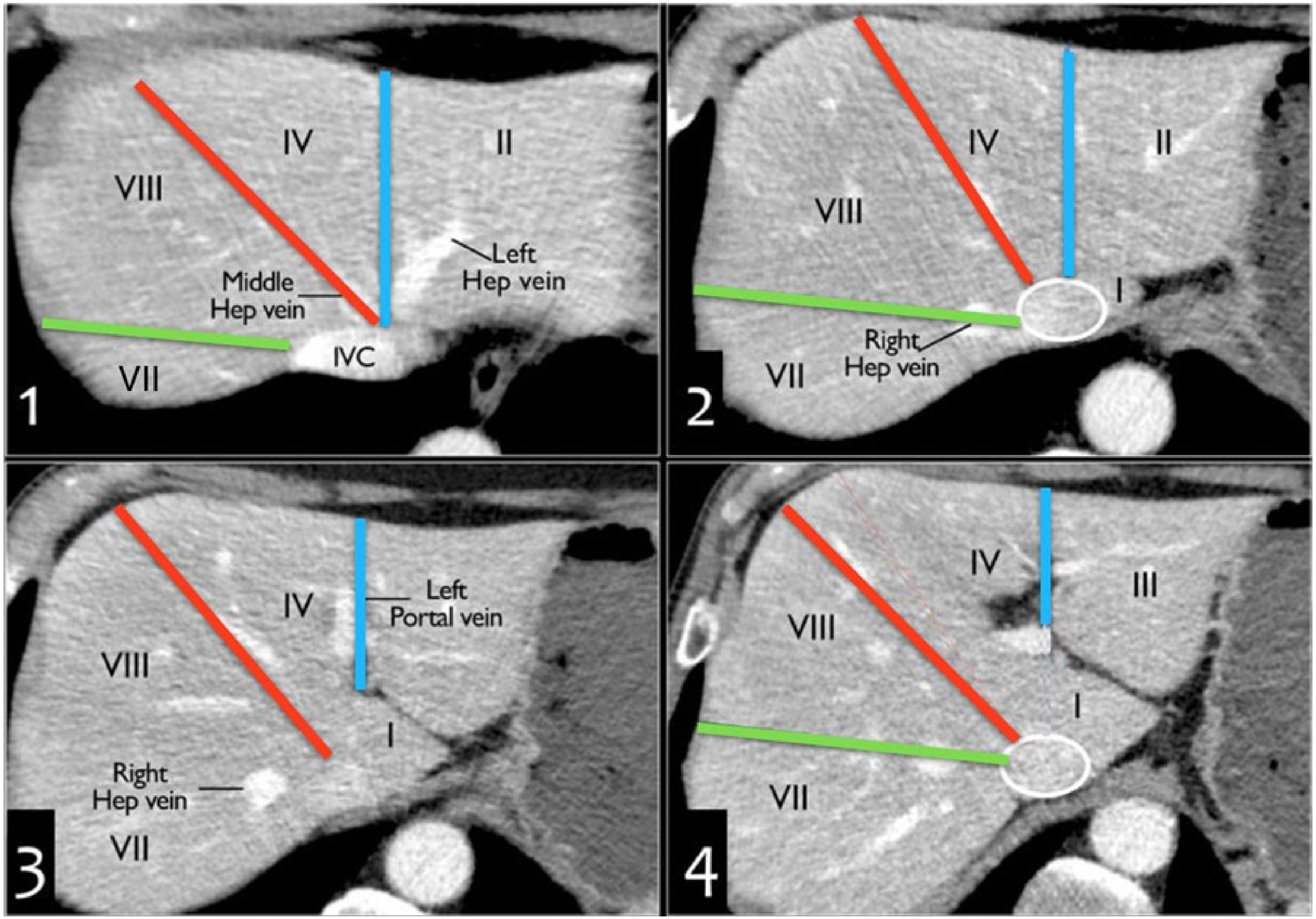
- Needs to be done really **fast and accurately**!



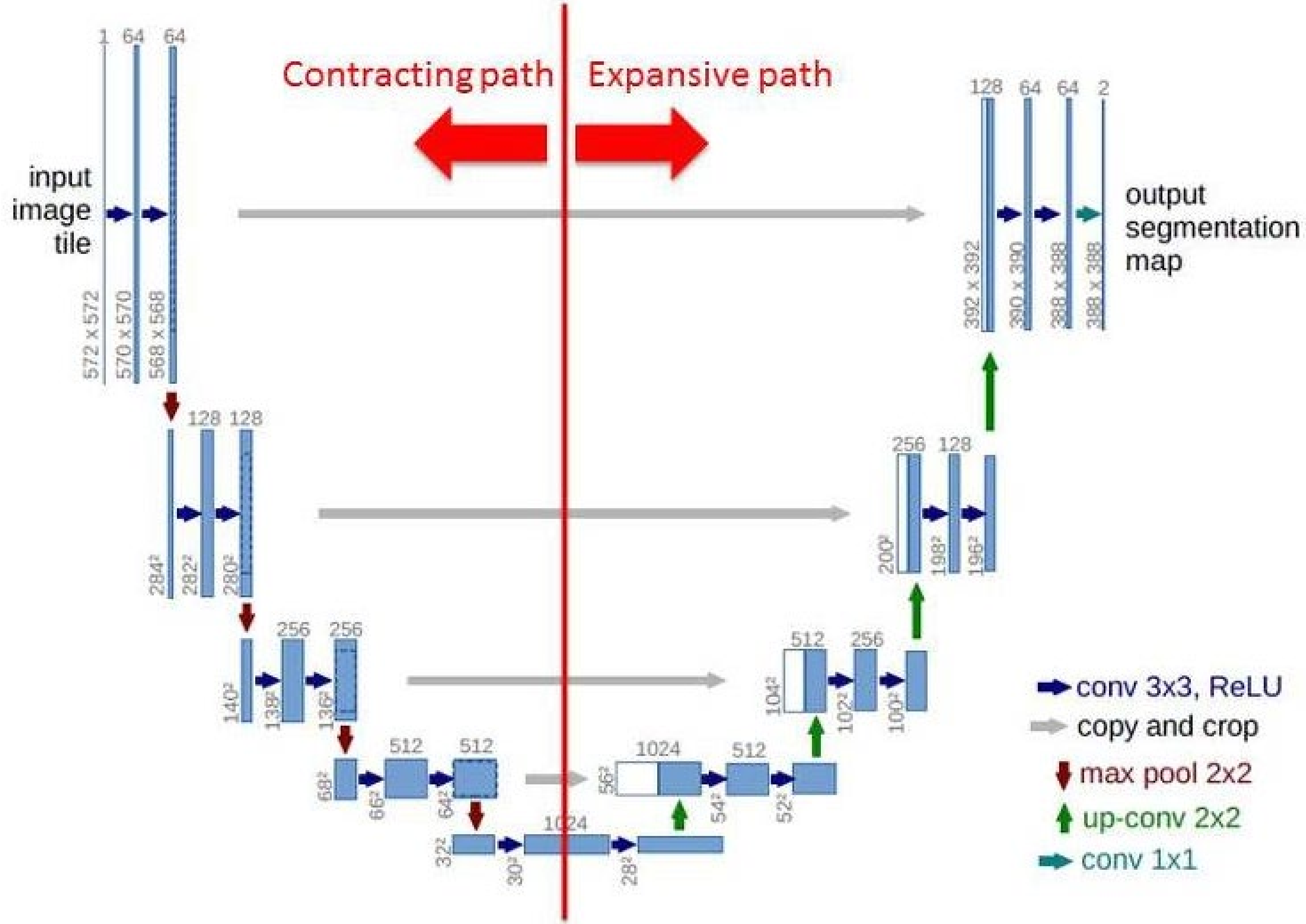
Liver Segmentation

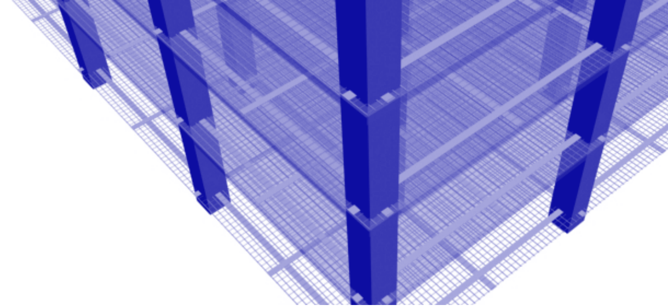


CT and MRI-based Imaging

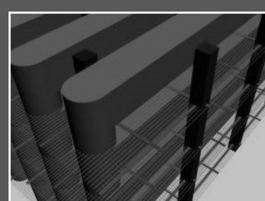
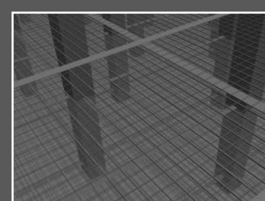
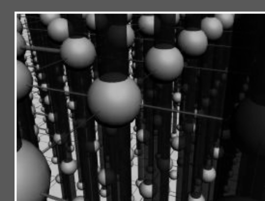


Unet: U-shaped CNN Autoencoder





3. 기계학습 교수법 및 코딩 실습의 예

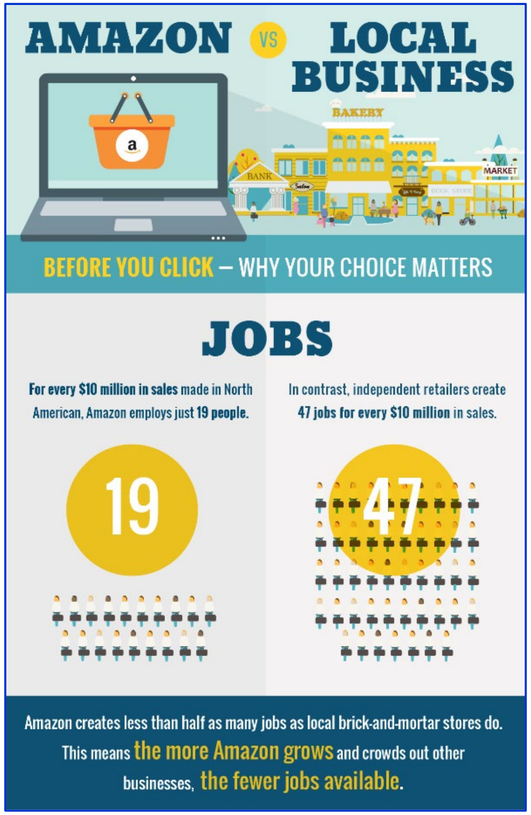
   

- RNN으로 주가 변동 예측하기
- Azure로 붓꽃 판별하기

RNN으로 주가 변동 예측하기

Amazon

Some Hate Amazon…



Amazon’s New Cash Cow

Competitive Positioning of Players in the Cloud

Amazon Stock

500X in 22 years!!!

Amazon Stock Price Database

- 4,529 daily stock price info
  - From 11/21/2000 to 11/21/2018

- Columns
  - Date: Daily Date
  - Open: Opening price of day
  - High: High price of day
  - Low: Low price of day
  - Close: Closing price of day
  - Adj Close: Adjusted close

1. stock-price.py (1/13)


```
# import packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import LSTM, Activation, Dense


# global constants and hyper-parameters
TIME_STEP = 3
MY_NUM_NEURON = 256
MY_CUTOFF = 0.7
MY_EPOCH = 10
MY_BATCH = 64
```

1. stock-price.py (4/13)


```
# display the logarithm of returns
change = raw_DB.pct_change()
log_return = np.log(1 + change)
print('\n== LOG-RETURN OF THE LAST 10 DAYS ==')
print(log_return.tail(10))

plt.show()


# scaling with z-score: z = (x - u) / s
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
scaled_DB = scaler.fit_transform(raw_DB)
```

fit_transform()

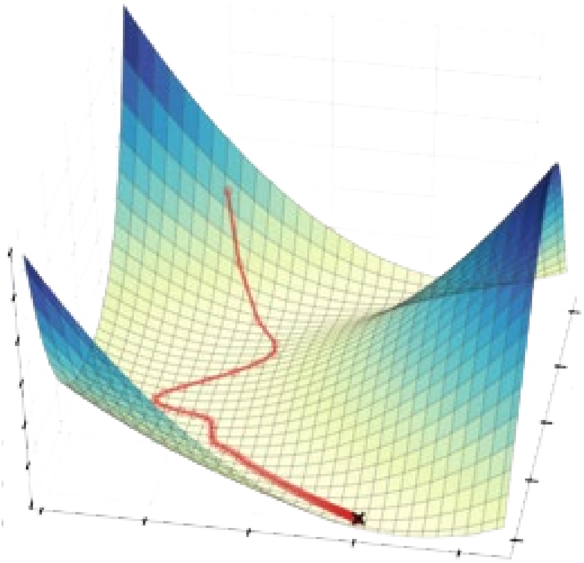

```
# conducts data centering
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()

# we must use 2D array
my_array = [[1], [2], [3], [6]]
x = scaler.fit_transform(my_array)
print(x)

# converts scaled data back to original
y = scaler.inverse_transform(x)
print(y)
```

1. stock-price.py (10/13)


```
# model training and saving

model.compile(loss = 'mse', optimizer = 'adam',
        metrics = ['accuracy’])

model.fit(X_train, Y_train, epochs = MY_EPOCH,
        batch_size = MY_BATCH, verbose = 1)

model.save('chap1.h5')
```

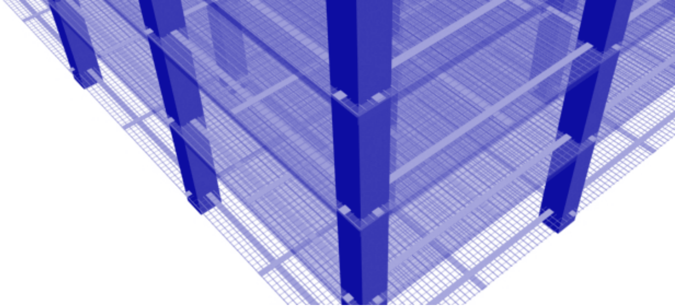
Adam Optimizer

- **Adam: Adaptive moment estimation**
  - **Keeps separate learning rates for each weight**
  - **Exponentially decay average of previous gradients**
  - **Fairly memory efficient since it doesn't keep a history of anything**
  - **Work well for both sparse matrices and noisy data**
  - **Seems promising for the stock market data.**
  - **Works well with little tuning of hyperparameters**

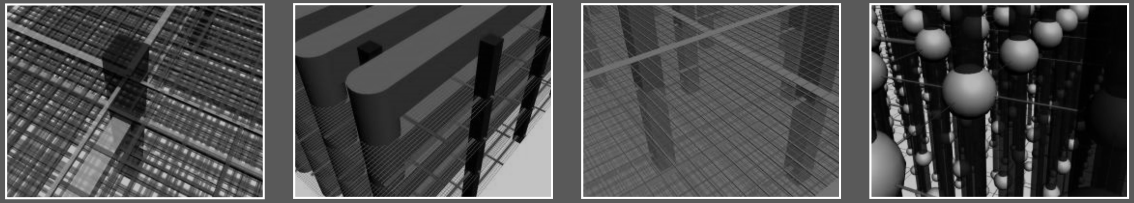


Exercise Questions

- **Q1: Examine the chap1.h5 file.**
- **Q2: Try other company stock data (GE, Microsoft, Apple, Google).**
- **Q3: Try sigmoid activation. What is the impact?**
- **Q4: Tune the MY_NUM_NEURON parameter. What is the impact?**
- **Q5: Tune the TIME_STEP parameter. What is the impact?**

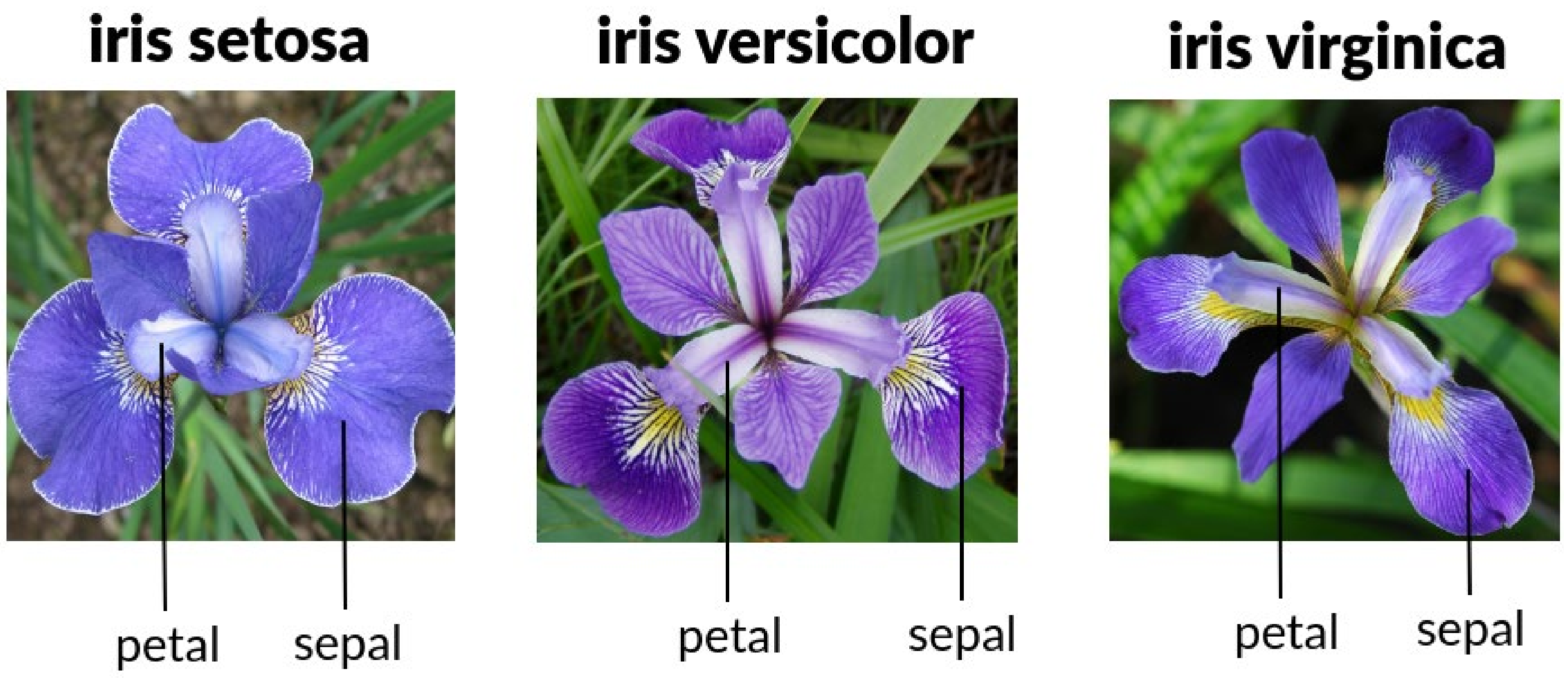


Azure로 붓꽃 판별하기

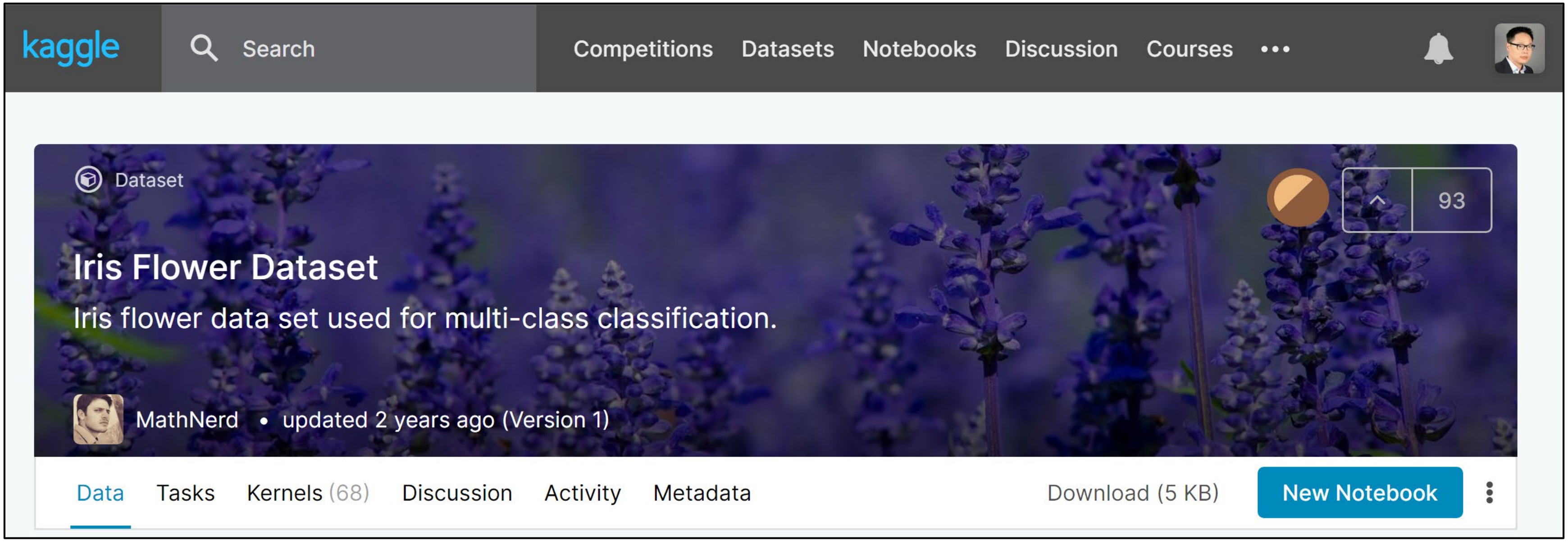


Iris Classification

- **Perhaps the best-known example in the field of machine learning**
  - The aim is to classify iris flowers among three species
  - From measurements of length and width of sepals and petals

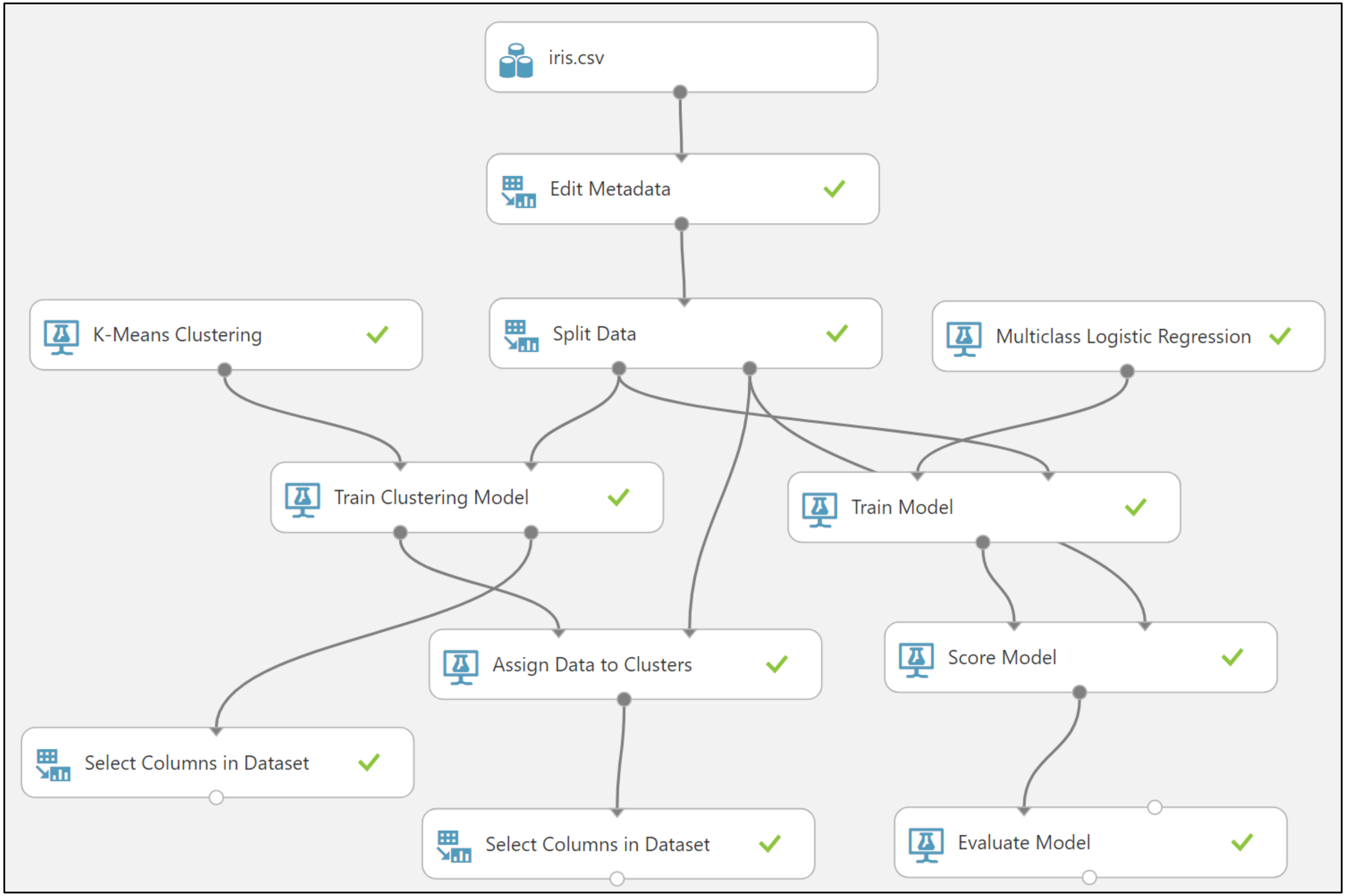


UCI Iris Flower Dataset



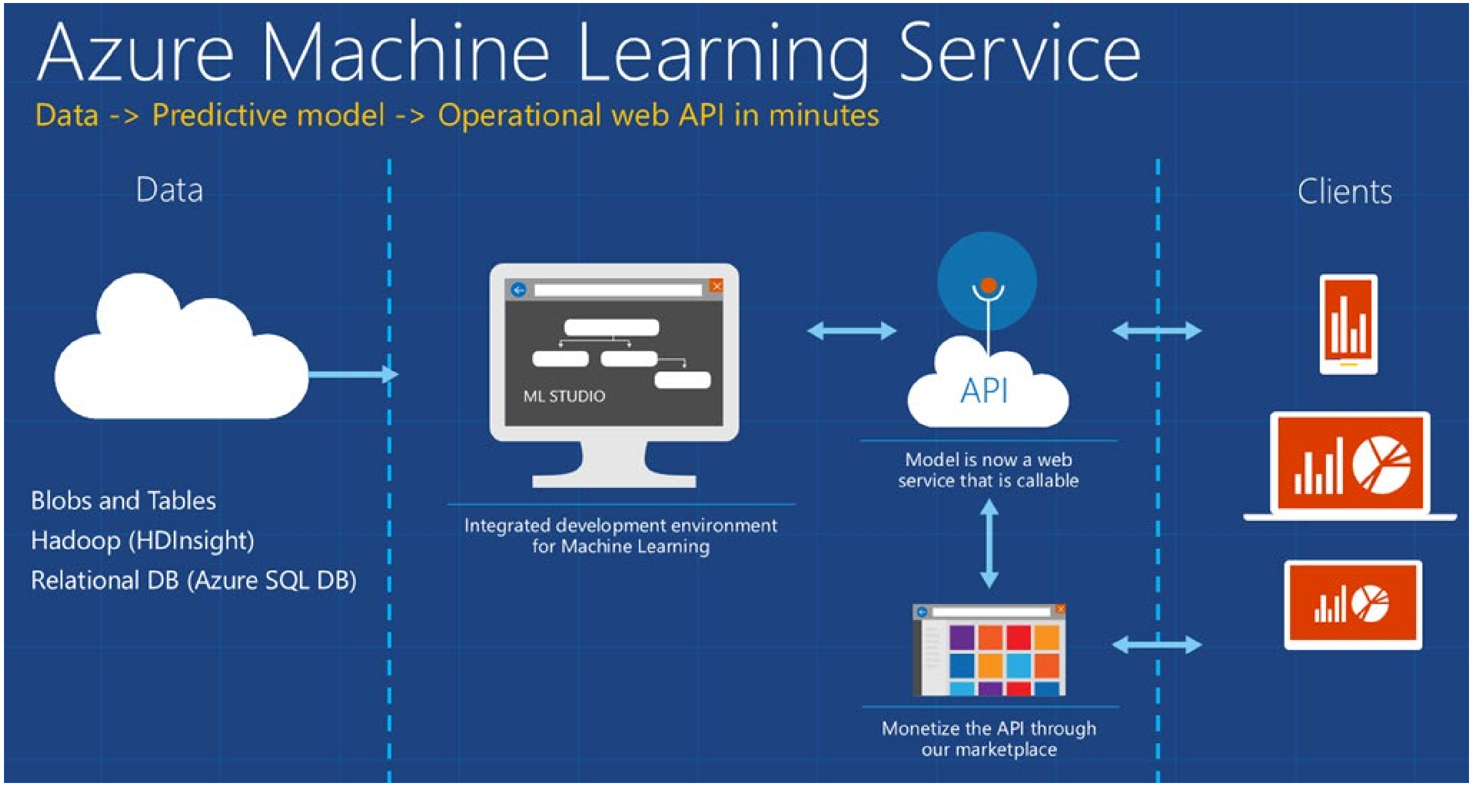
- **150 data with 5 features**
  - **Petal Length, Petal Width, Sepal Length, Sepal width and Class (Species)**
- **Goal is to predict class given the 4 dimensions**
  - **Classical classification problem**

Overview of the Model



Azure ML Design Flow

- **4 major steps**
  - **Data acquisition > Data preparation > Model training and evaluation > web deployment**

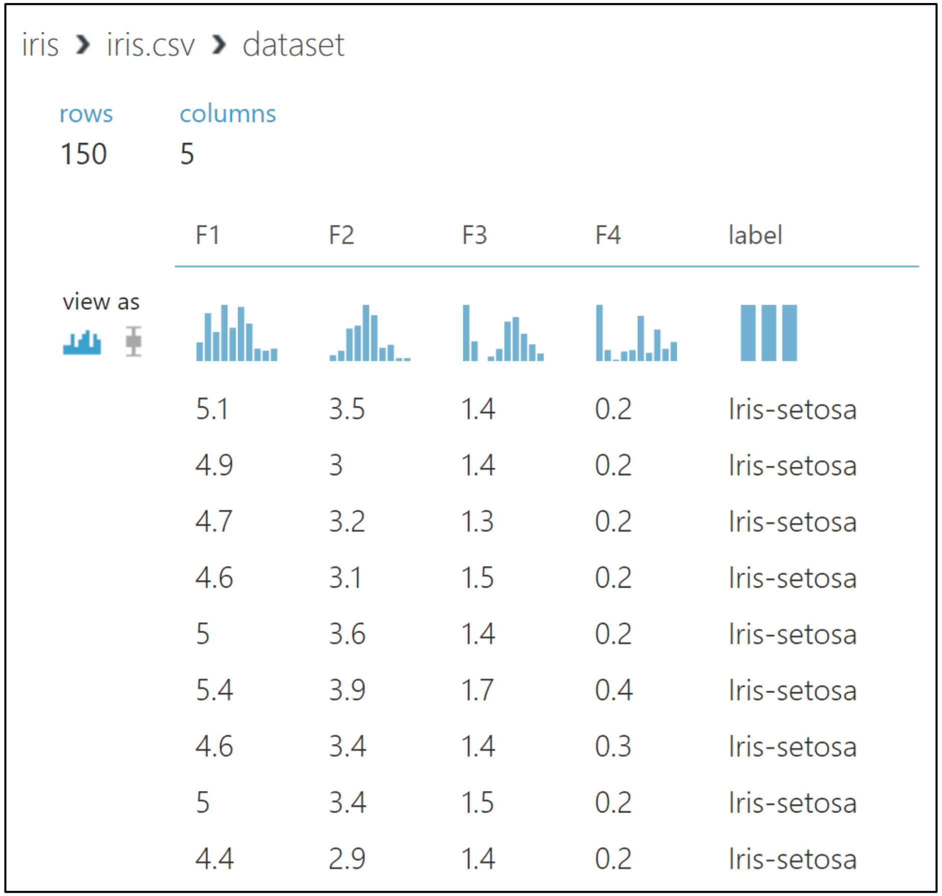


Data Acquisition

- **Search for My Datasets menu in the search window.**

- **Drag and drop iris.csv module onto the canvas on the right.**

- **Right click iris.csv module and select dataset > visualize. Examine the dataset stats.**



iris › iris.csv › dataset

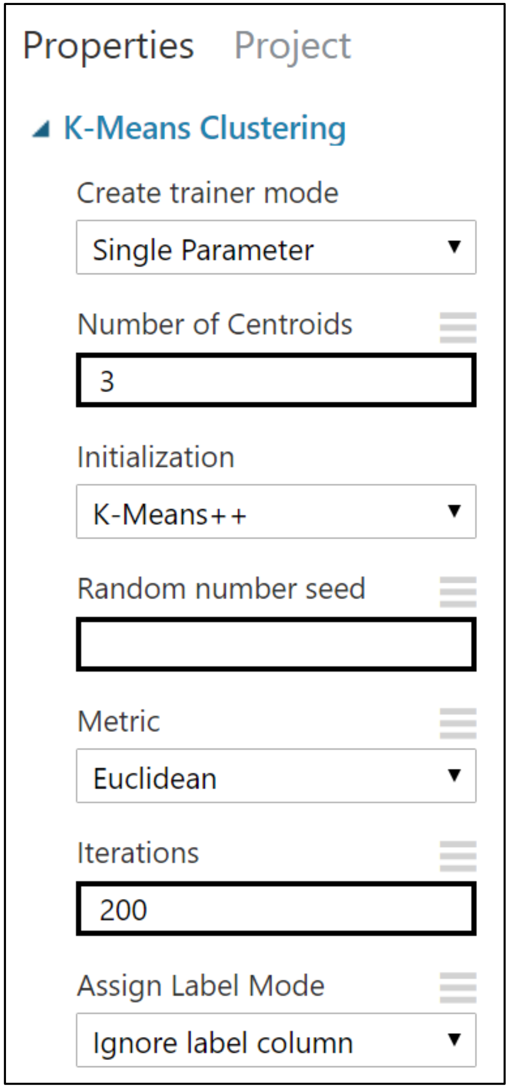
rows	columns
150	5

view as

F1	F2	F3	F4	label
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa

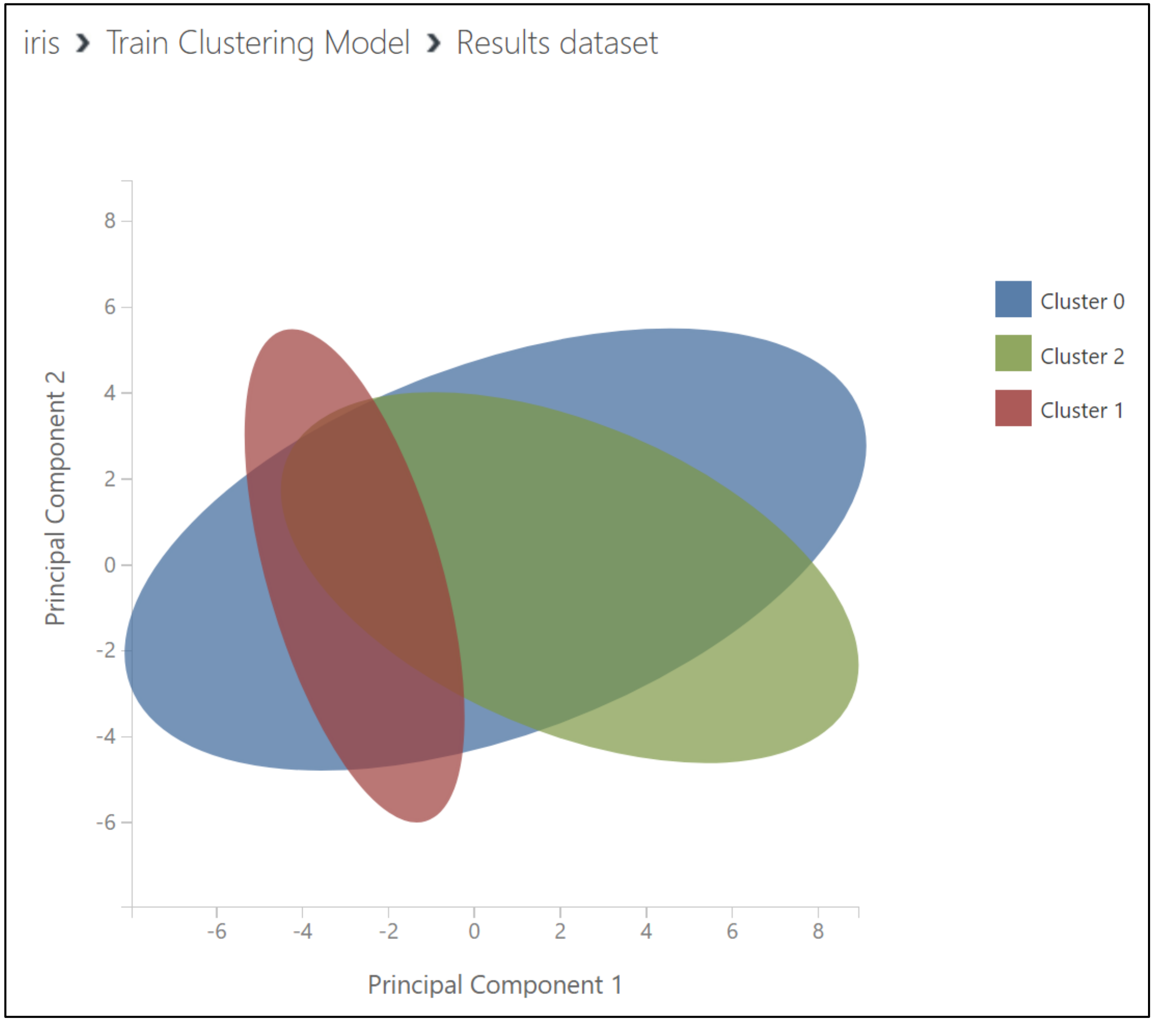
Model Training and Evaluation

- Click **K-Means Clustering** module.
- **Change the number of centroids to 3.**
- **Leave other options as default.**



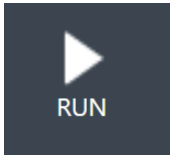
Model Training and Evaluation

- Right click **Train Clustering Model** module and visualize the result.

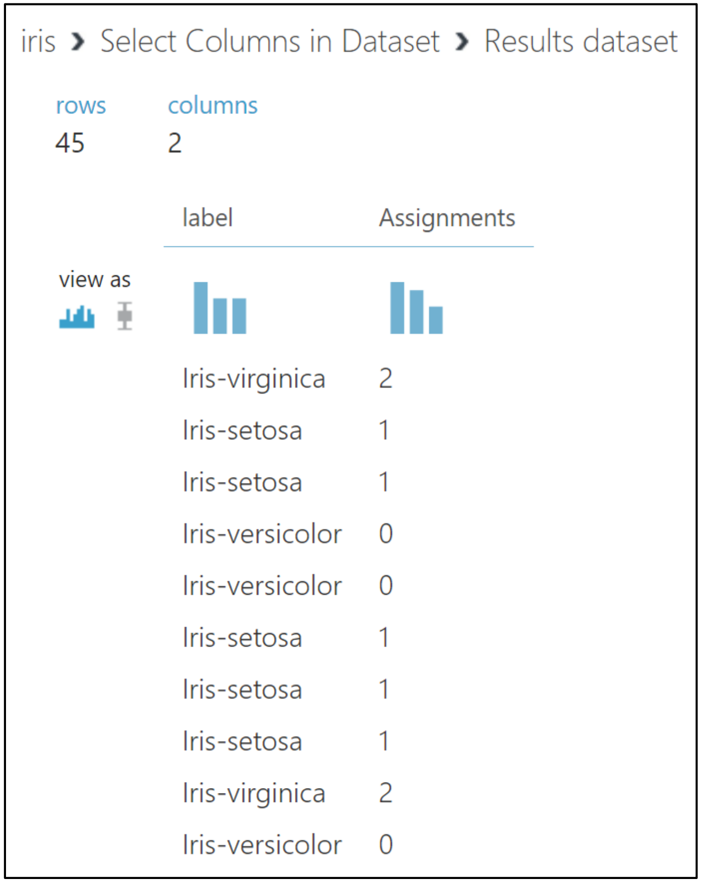


Model Training and Evaluation

- **Run the experiment.**



- **Right click <u>Select Columns in Dataset</u> module and visualize the result.**


iris › Select Columns in Dataset › Results dataset

rows 45 columns 2

view as

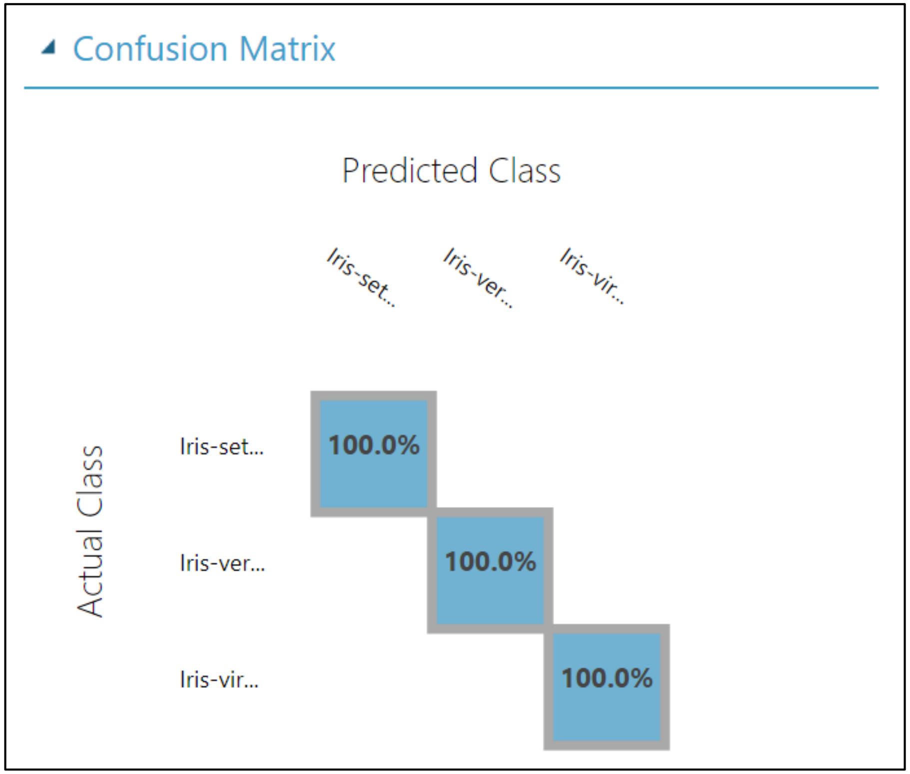
label	Assignments
Iris-virginica	2
Iris-setosa	1
Iris-setosa	1
Iris-versicolor	0
Iris-versicolor	0
Iris-setosa	1
Iris-setosa	1
Iris-setosa	1
Iris-virginica	2
Iris-versicolor	0

Model Training and Evaluation

- Right click **Evaluate Model** module to visualize the result.

Metrics

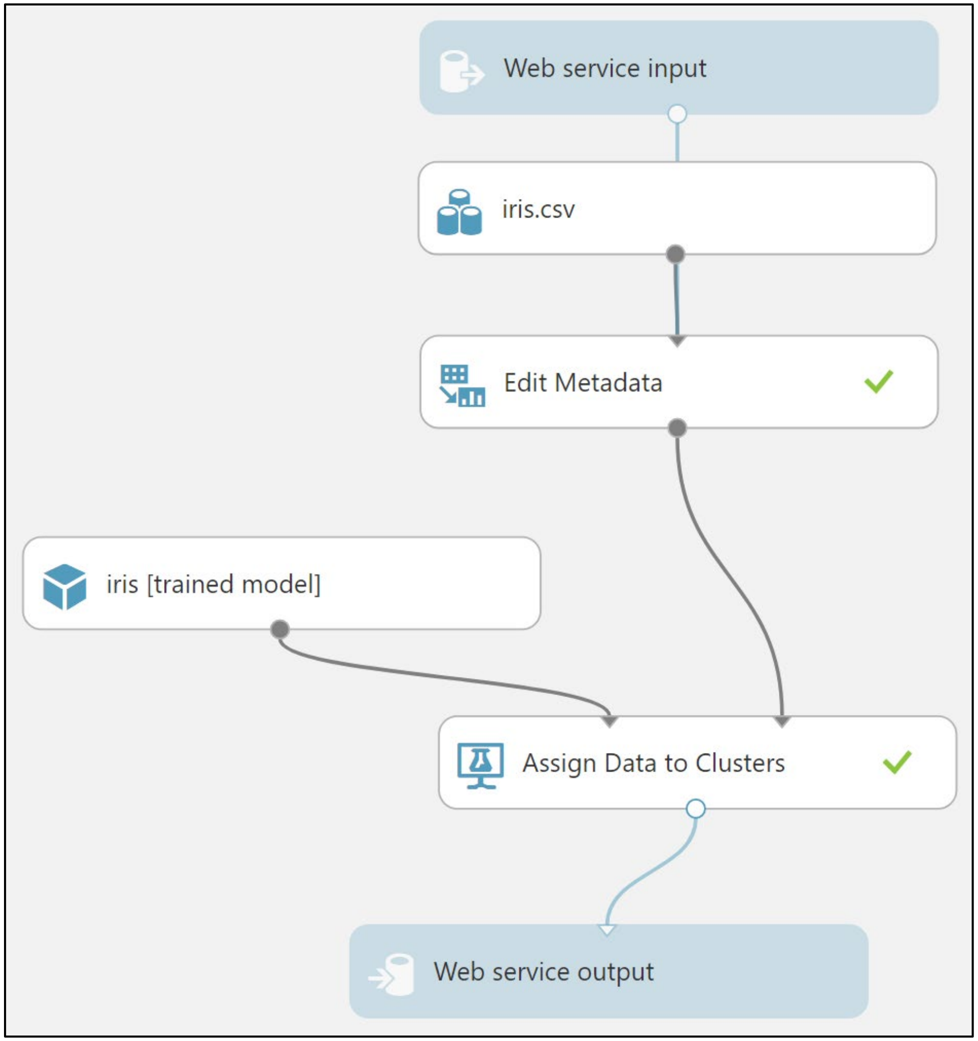
Overall accuracy	1
Average accuracy	1
Micro-averaged precision	1
Macro-averaged precision	1
Micro-averaged recall	1
Macro-averaged recall	1



Web Deployment

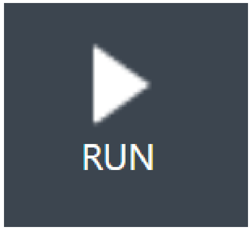
- Click **SET UP WEB SERVICE** button on the bottom and select Predictive Web Service menu.

- Newly built model is shown.



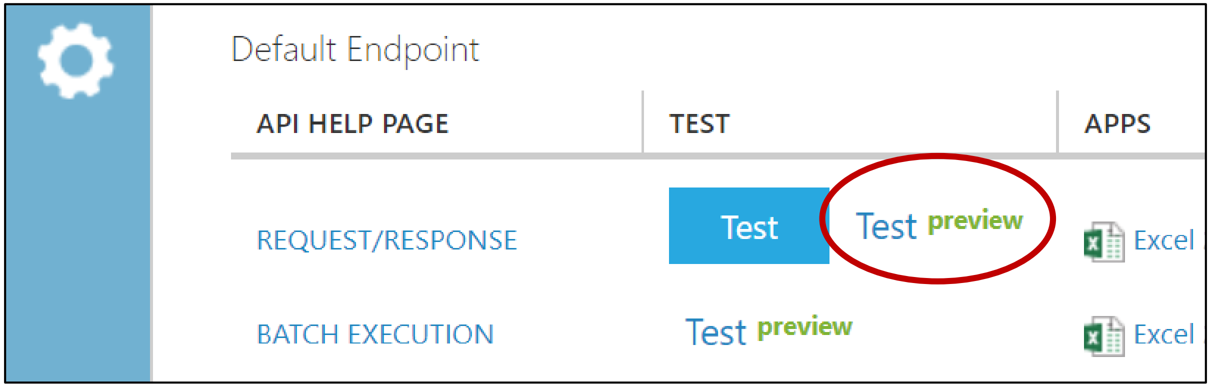
Web Deployment

- **Run the experiment.**



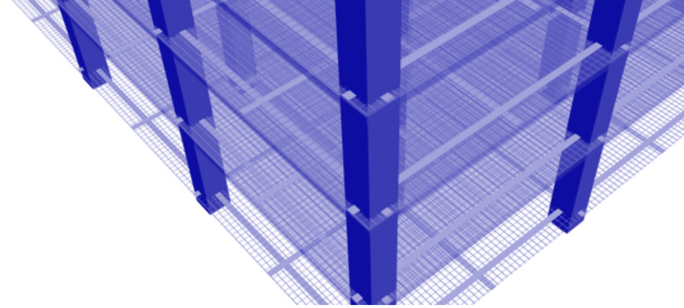
- **Click DEPLOY WEB SERVICE bottom on the bottom.**

- **Click Test-preview button.**

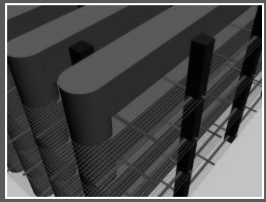
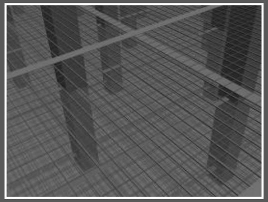
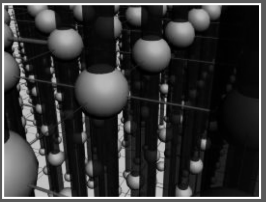


Homework

1. Use Edit Metadata module to drop F4 column while running K-means clustering. What is the impact?
2. Replace Multiclass Logistic Regression module with Multiclass Neural Network module. Which one performs better?
3. Replace Multiclass Logistic Regression module with Multiclass Decision Forest module. Which one performs better?
4. Use Select Columns in Dataset module to drop F4 column while running multi-class logistic regression. What is the impact?
5. Try changing the split ratio and see the impact.
6. What happens if we do not randomize the split process?



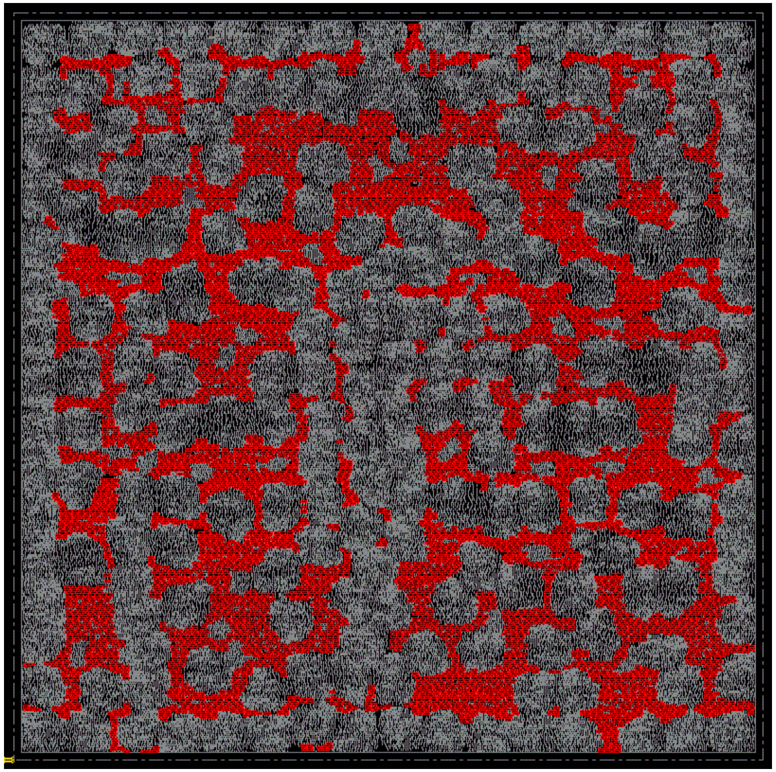
4. 기계학습 관련 진행 연구

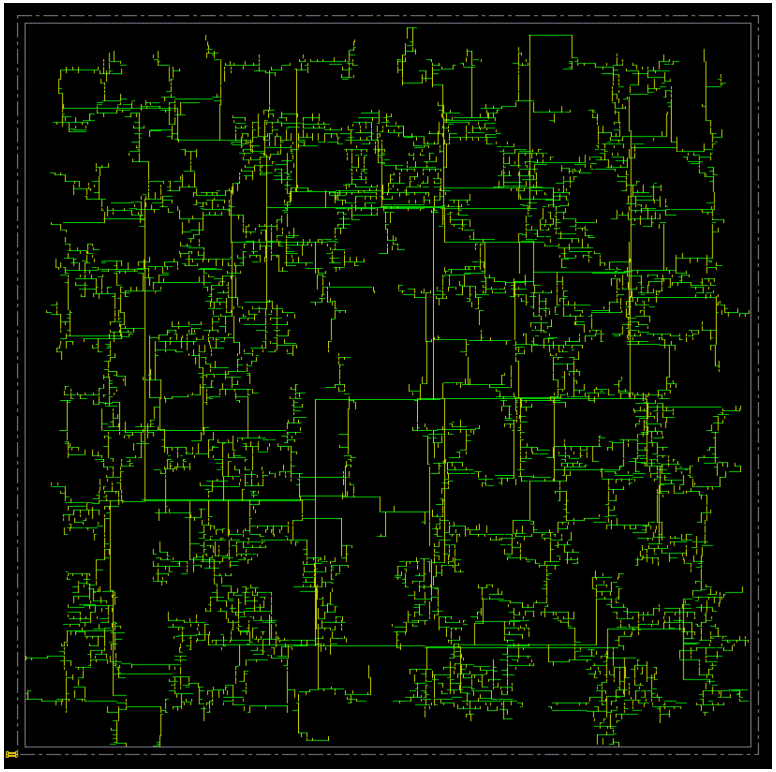
- 회로설계에 적용된 기계학습

Clock Routing In Digital Circuit

- Make electrical connects to all sequential elements in the circuit



sequential elements (red)



clock tree (green)

Two Clock Trees

- **Setting is VERY important and requires years of experience**

feature	value
Target skew	0.13ns
Max fanout	195
Max cap (trunk)	0.04pF
Max cap (leaf)	0.10pF
Target slew (trunk)	0.23ns
Target slew (leaf)	0.26ns
Target latency	0.4ns
eGR metal usage	1, 2, 3, 4
Cell density	0.6

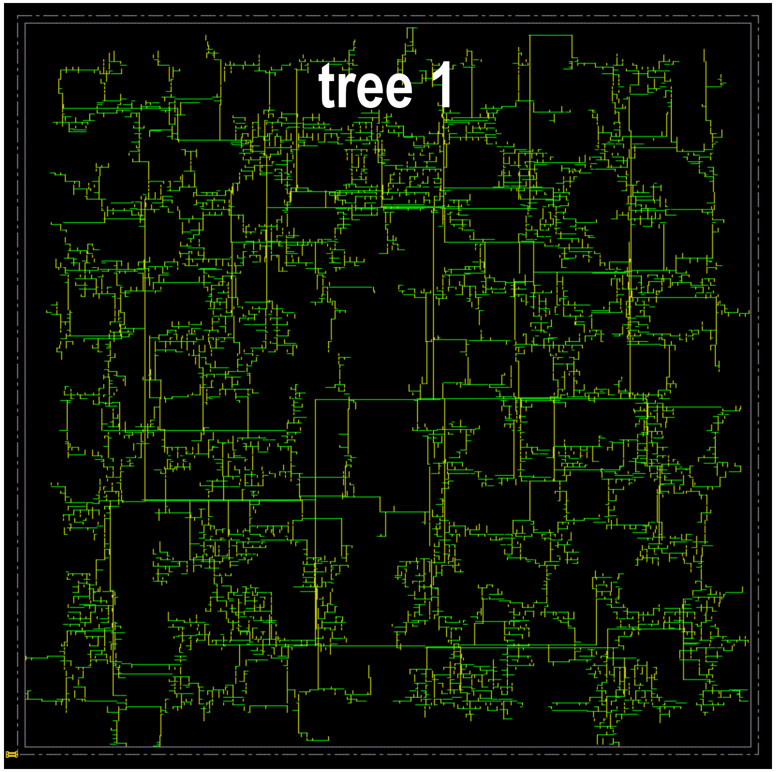
tree 1 settings

feature	value
Target skew	0.08ns
Max fanout	175
Max cap (trunk)	0.03pF
Max cap (leaf)	0.07pF
Target slew (trunk)	0.21ns
Target slew (leaf)	0.03ns
Target latency	0.2ns
eGR metal usage	1, 2, 3
Cell density	0.7

tree 2 settings

Very Difference Outcome

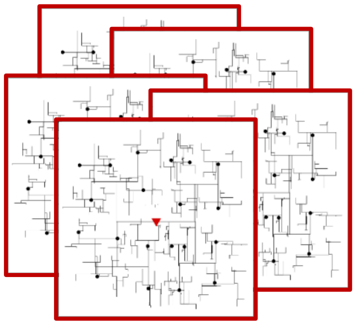
- **Quality is very different**



Power: 21.8mW	WL: 37.5mm
Skew: 0.15ps	Latency: 0.55ps

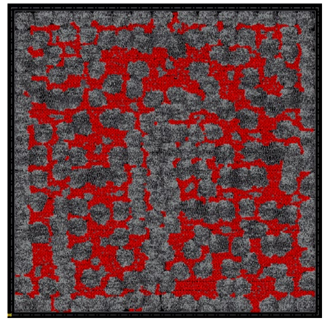
Power: 72.3mW	WL: 76.4mm
Skew: 0.13ps	Latency: 0.87ps

ML-based CTS Prediction



clock tree DB

+



placement (FF in red)

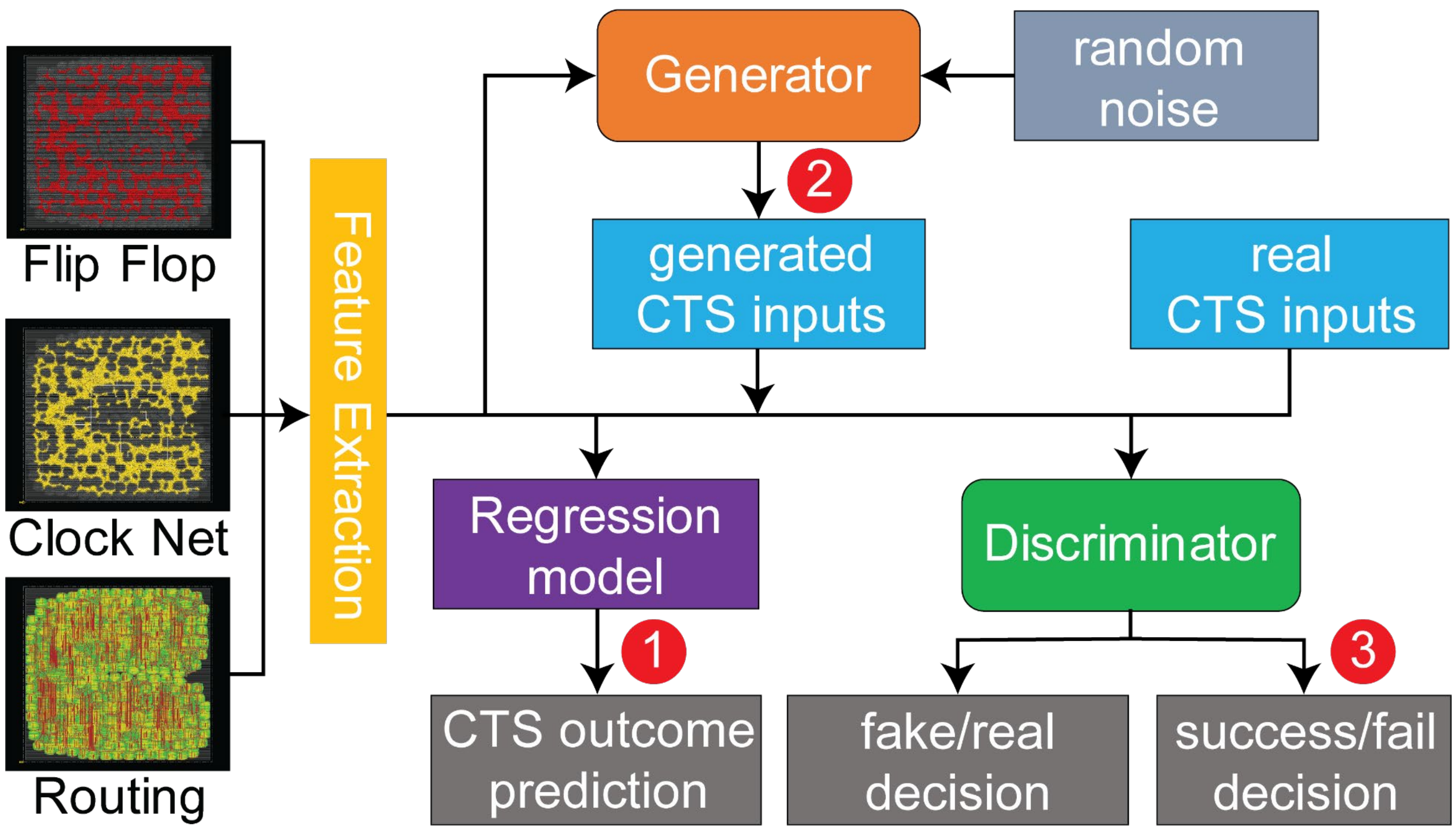
prediction:
how do I build
the best CLK?

power,
skew,
WL…

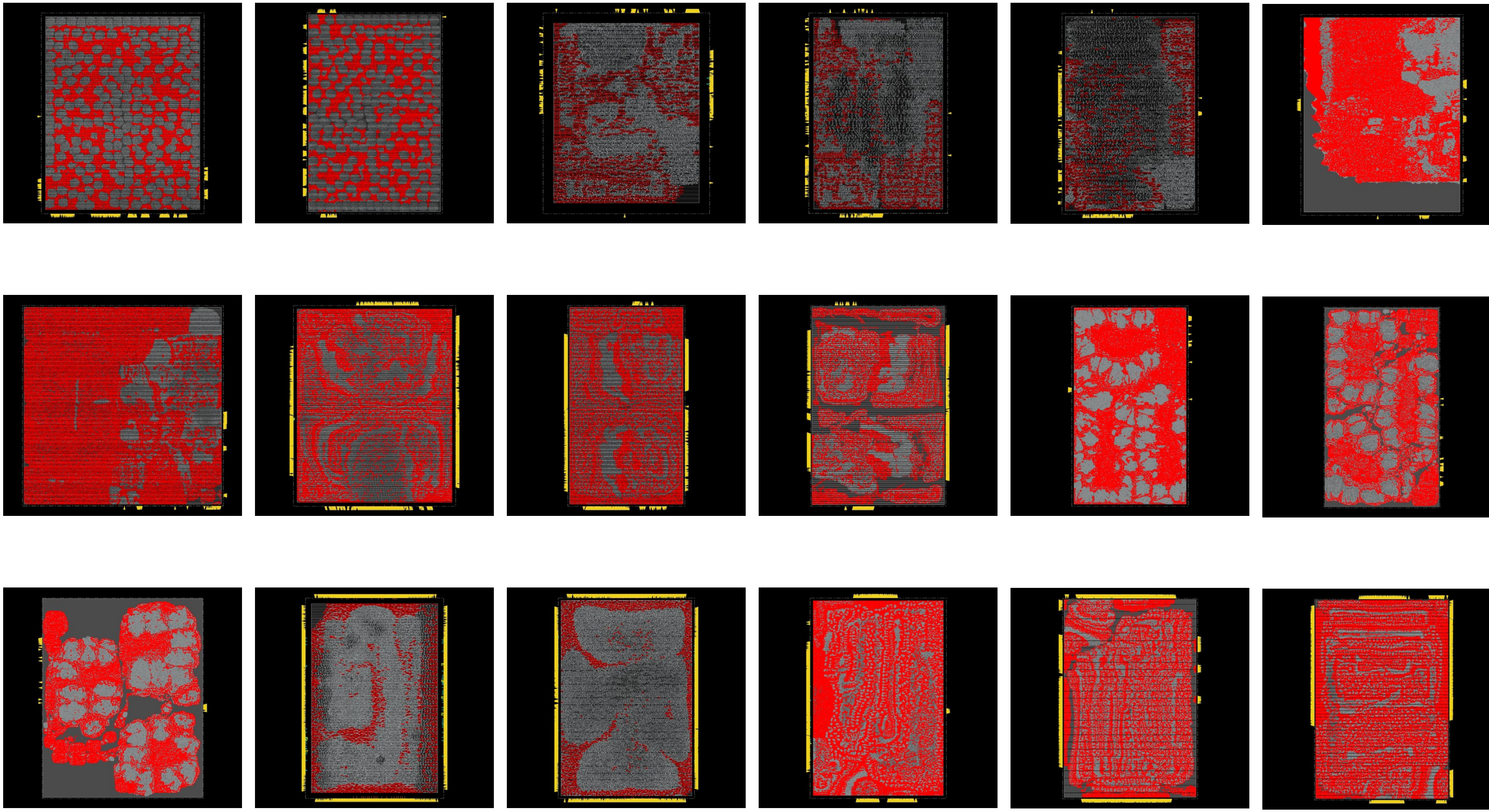
CTS
parameters
used

better CTS than human &
commercial auto-setting

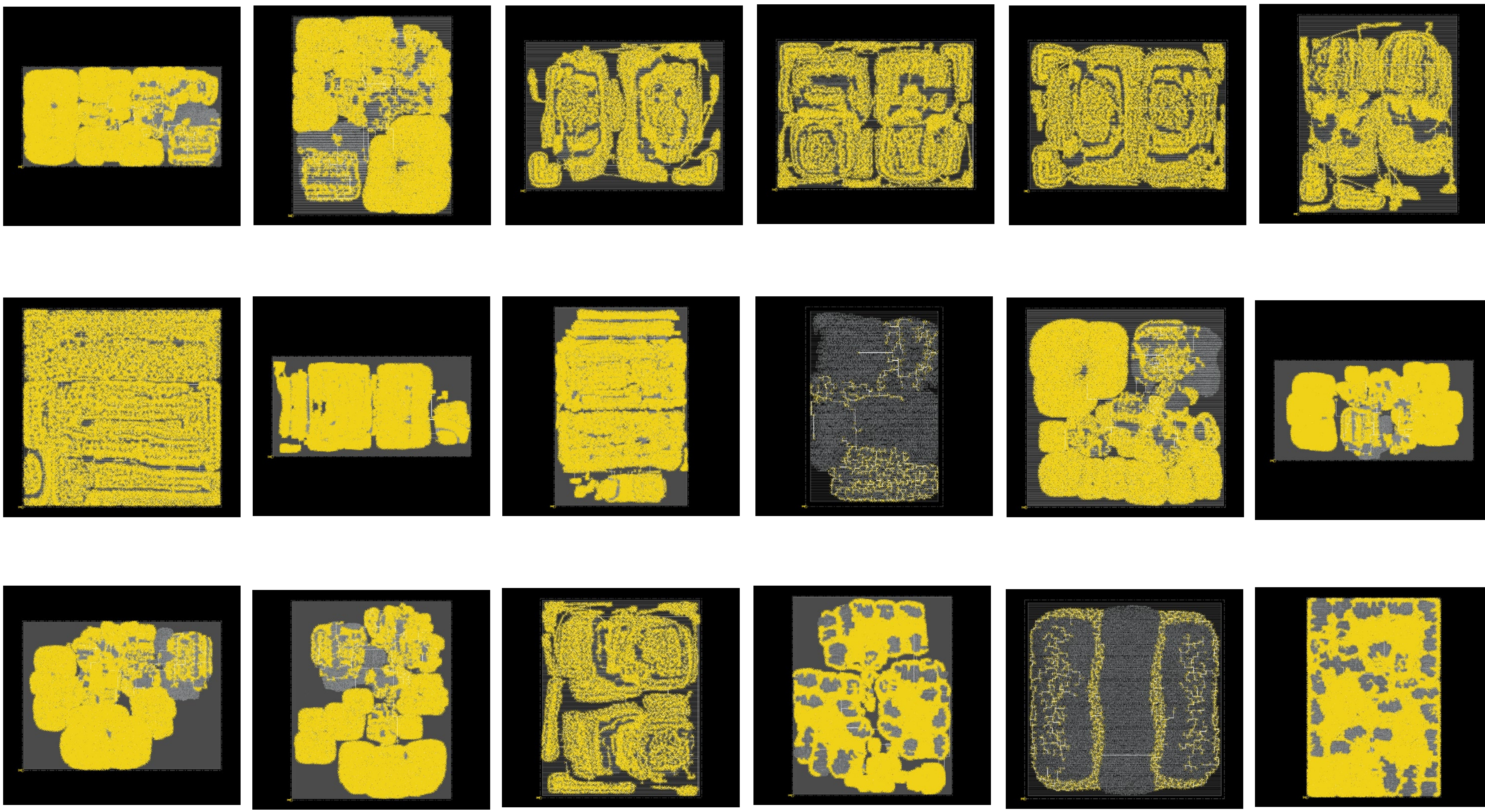
Our GAN-based Flow



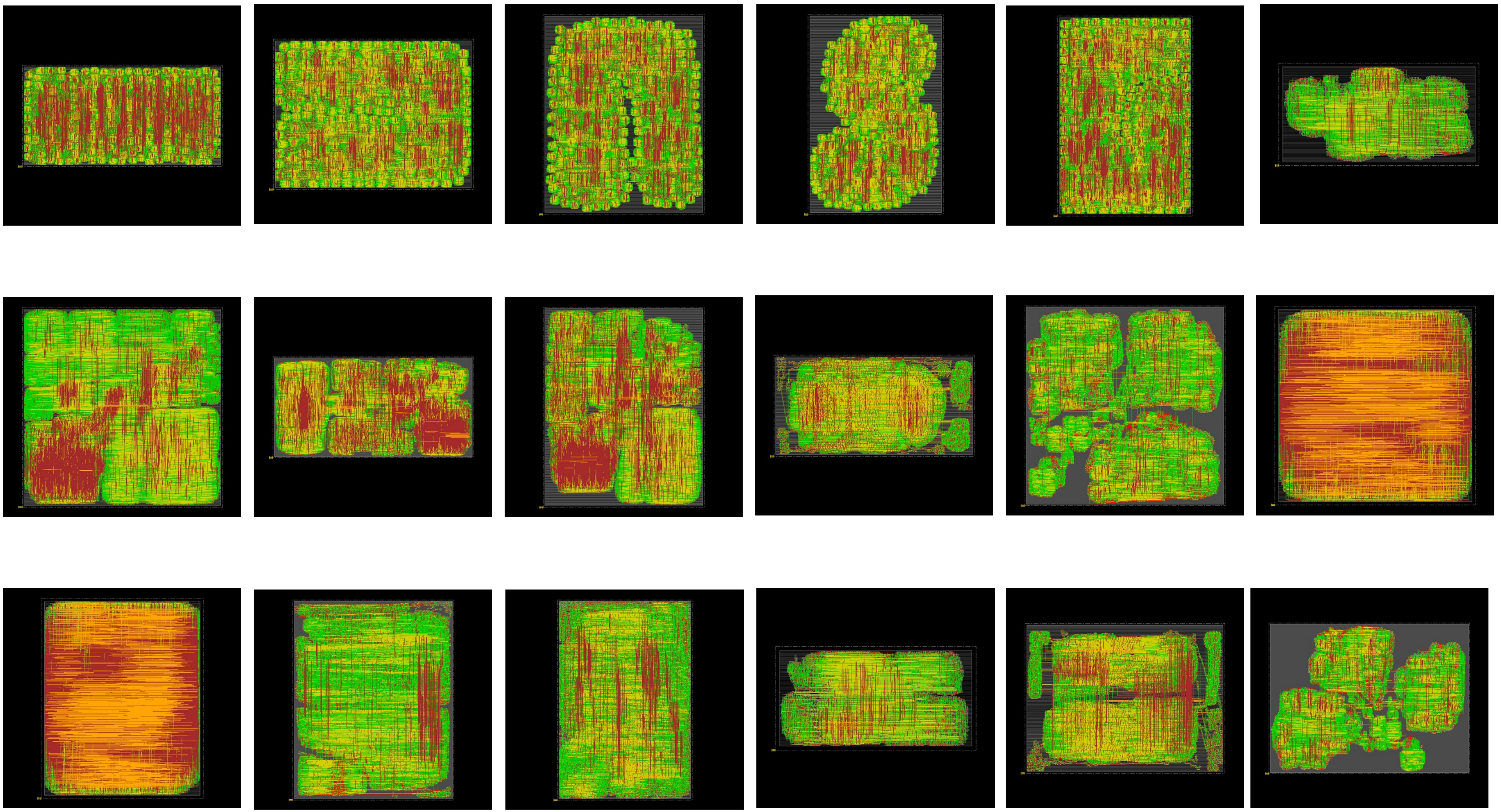
Our DB: 3000 Placements from 7 Ckts



Our DB: 3000 Clock Net from 7 Ckts

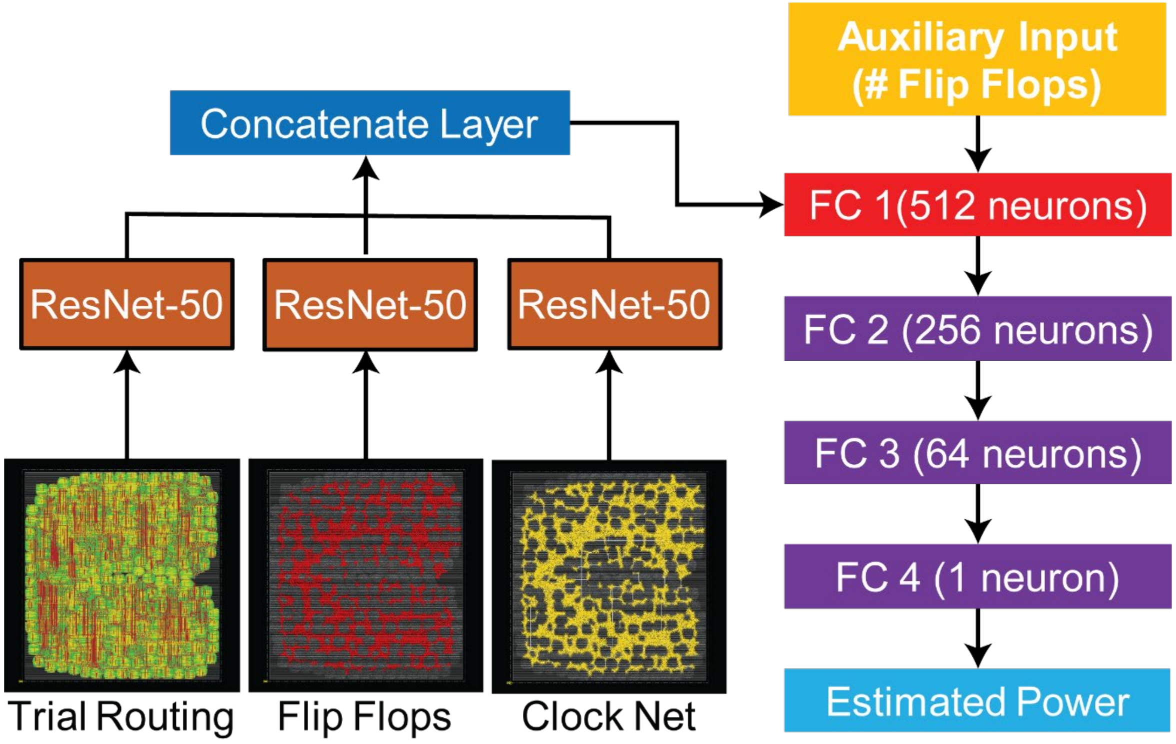


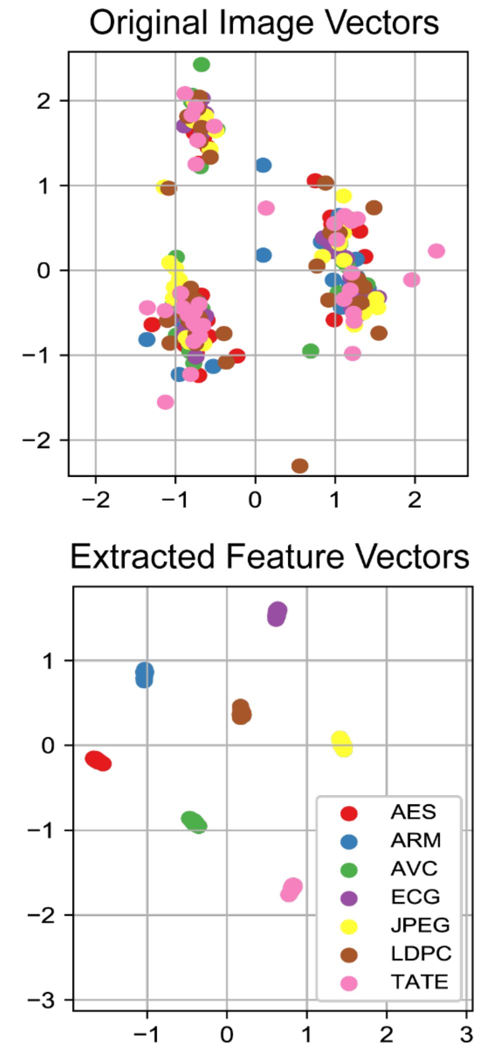
Our DB: 3000 Global Routing from 7 Ckts



Why Images?

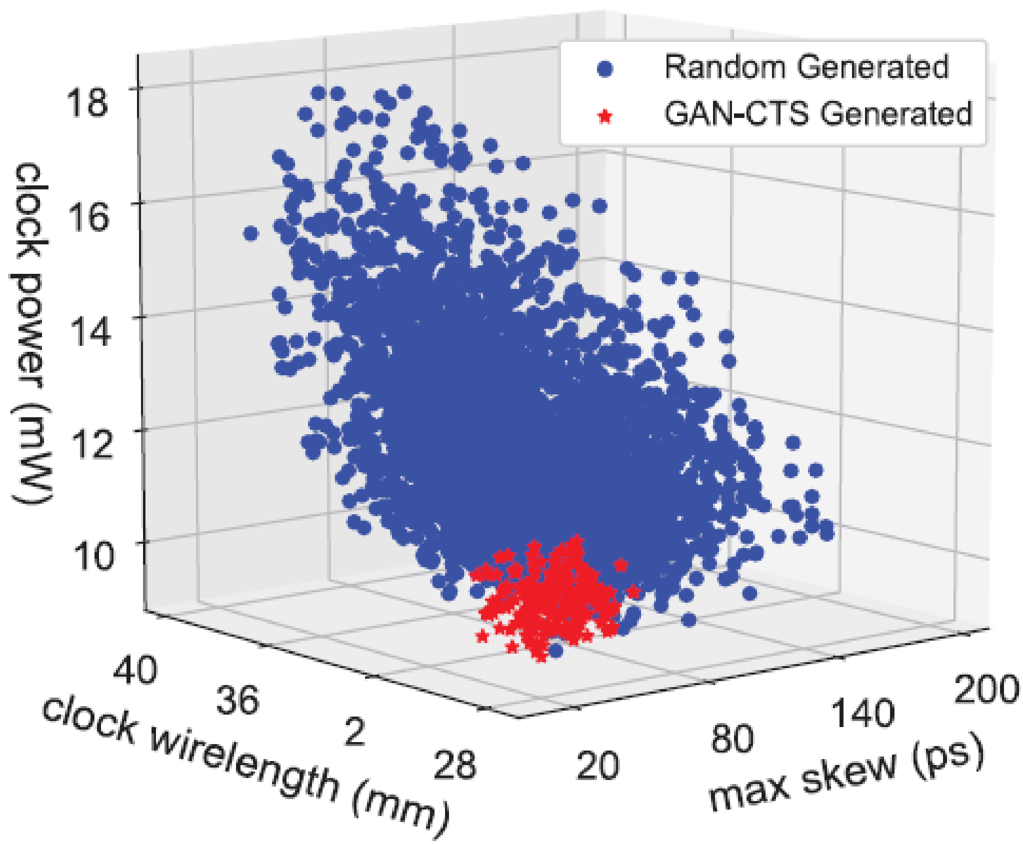
- **Extremely useful**
  - **In handling unseen netlist**



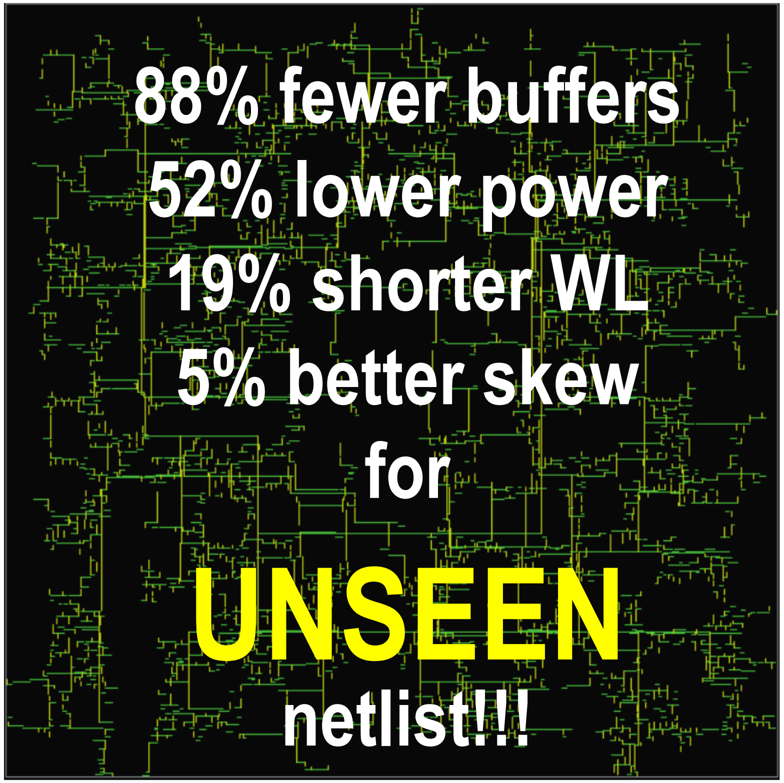


GAN-Generated Data

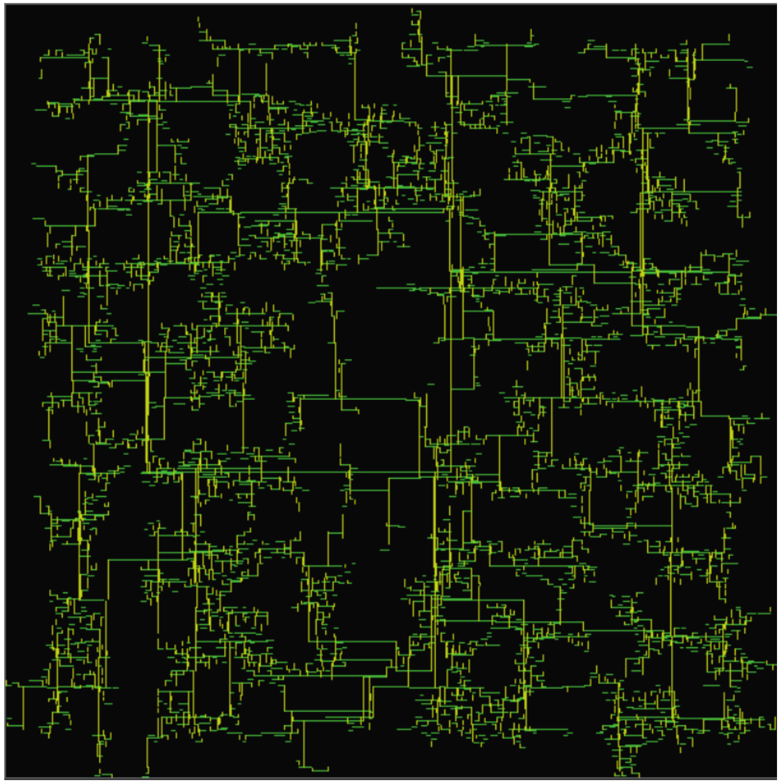
- Our generator became very smart



Gan-Optimized Clock Tree



(a) GAN-CTS optimized



(b) commercial auto-setting

Conclusions

1. 인공지능 (AI) 기계학습이란?

2. 기계학습의 종류
   - CNN, RNN GAN, Autoencoder, Unet

3. 기계학습 교수법 및 코딩 실습의 예
   - RNN으로 주가 변동 예측하기
   - Azure로 붓꽃 판별하기

4. 기계학습 관련 진행 연구
   - 회로설계에 적용된 기계학습