University of Pittsburgh

Introduction to Deep learning

-- Deep Convolution Neural Networks (DCNNs) & Graph Neural Networks (GNNs)

Oct. 21, 2021

Haoteng Tang, Ph.D. candidate,
Electrical and Computer Engineering,
Swanson School of Engineering,
University of Pittsburgh

Outline

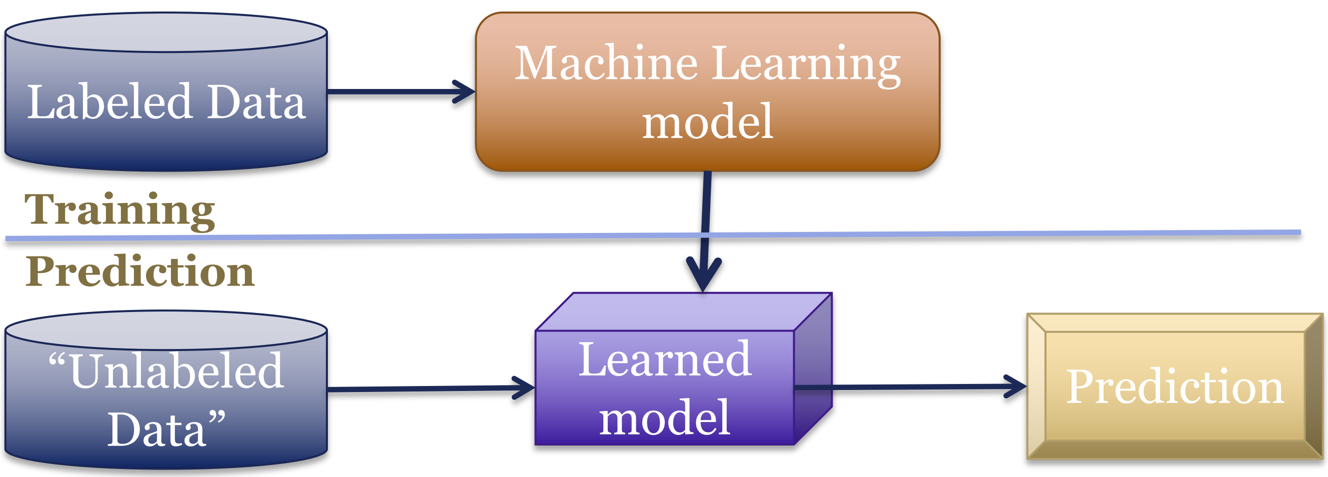
- Start from Machine Learning (ML)
- Introduction to Deep learning (DL)
- Deep Convolution Neural Networks (DCNNs)
- Graph Neural Networks (GNNs)
- Deep Learning Libraries and Tools

Machine Learning

- What is machine learning?

Machine learning is a field of computer science which gives computers the ability to learn without being explicitly programmed.

Specifically, machine learning algorithms build a model based on sample data, known as "training data" in order to make predictions or decisions without being explicitly programmed to do so



Machine Learning

- Different types of (machine) learning

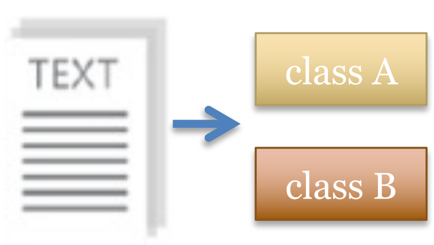
➢ Supervised learning: Learn with a labeled dataset

Example: labeled email classification based on the content; brain age regression

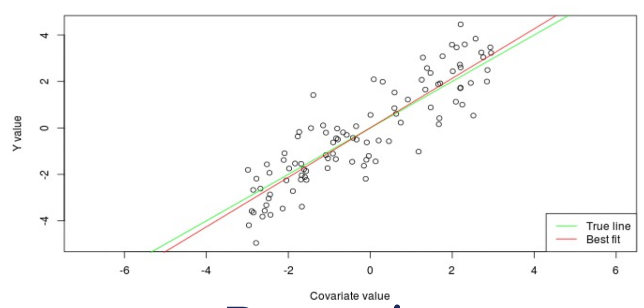
➢ Unsupervised learning: Explore the patterns of unlabeled dataset

Example: document/article clustering based on the topic

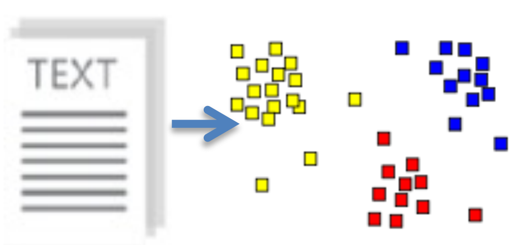
➢ Others: Semi-supervised learning, Reinforcement learning etc.



Classification



Regression



Clustering

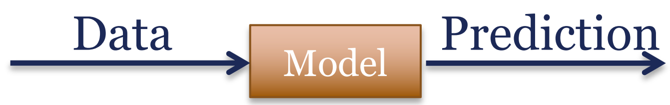
Machine Learning

- Machine learning in practice, general workflow

Step	Description
Acquire data	What kind of data? Image? Text? Video?
Clean data	Data missing? Incorrect? Ambiguity?
Present data	How you present your data to the computer? (e.g., English article, Chinese article)
Describe data	Extract features by your expertise, domain specific, may need Ph.D.
Learning model	Build up your model, feed your data features and optimize your model.

Machine Learning

- Machine learning models



➢ Classification model

Example: logistic regression, naïve Bayes, support vector machine (kernel machine)

➢ Regression model

Example: linear regression, lasso regression etc.

➢ Clustering model

Example: K-means, Expectation-Maximization based clustering (e.g., GMM)

Machine Learning

- One question in machine learning

Acquire data	What kind of data? Image? Text? Video?
Clean data	Data missing? Incorrect? Ambiguity?
Present data	How you present your data to the computer? (e.g., English article, Chinese article)
Describe data	Extract features by your expertise, domain specific, may need Ph.D.
Learning model	Build up your model, feed your data features and optimize your model.

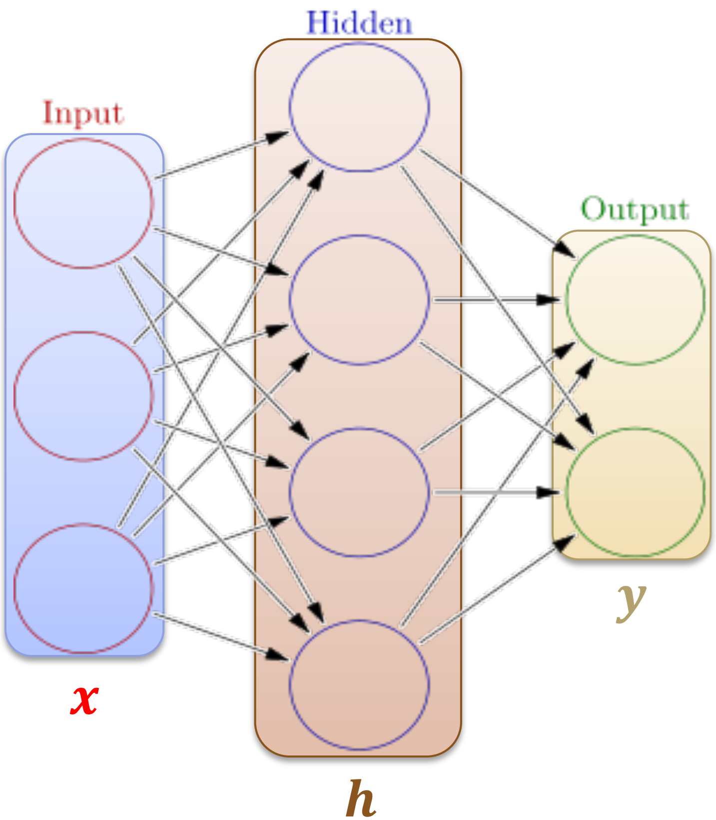
Machine Learning

- Why extract the features by the model itself?
  - Time consuming, low efficiency, expensive
  - Manually designed features are often over-specified, incomplete and take a long time to design and validate
- Toward Deep learning

The deep learning model can extract the features from the data and learn latent, solid and adaptive features (representations) by optimizing the model parameters.

Deep Learning

- How to extract features by the model? – Neural Networks



$$h = \sigma(W_1 x + b_1)$$

$$y = \sigma(W_2 h + b_2)$$

Activation functions, weight, bias

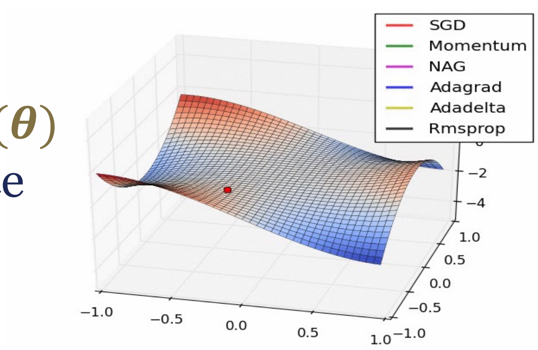
How do we train?

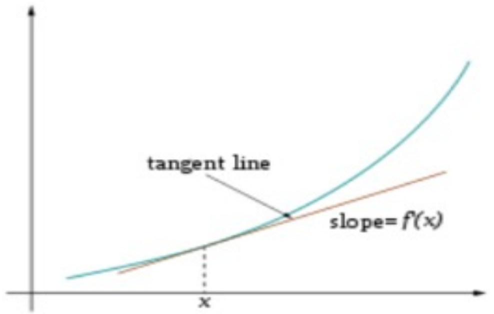
Deep Learning

- How to train this neural network

Sample labeled data (**batch**) → **Forward** it through the network, get predictions → **Back-propagate** the errors → **Update** the network weights

Optimize (min. or max.) **loss/cost function $J(\theta)$**
Generate **error signal** that measures difference between predictions and target values



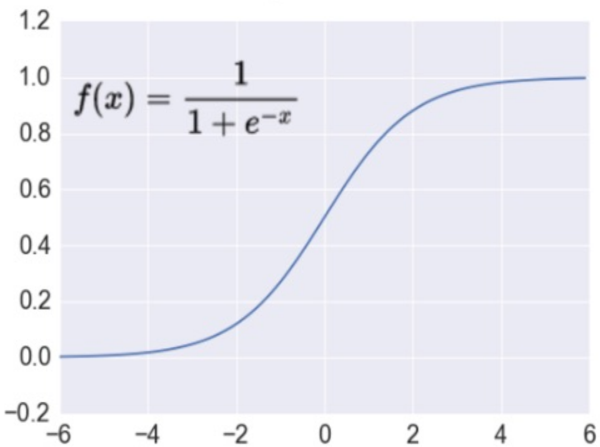


Use error signal to change the **weights** and get more accurate predictions
Subtracting a fraction of the **gradient** moves you towards the **(local) minimum of the cost function**

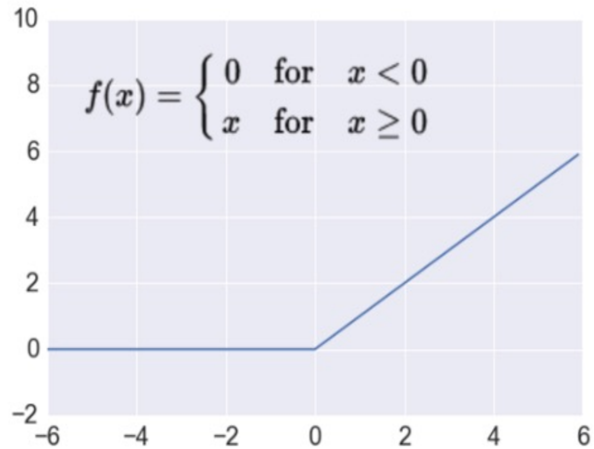
Deep Learning

- Activation function

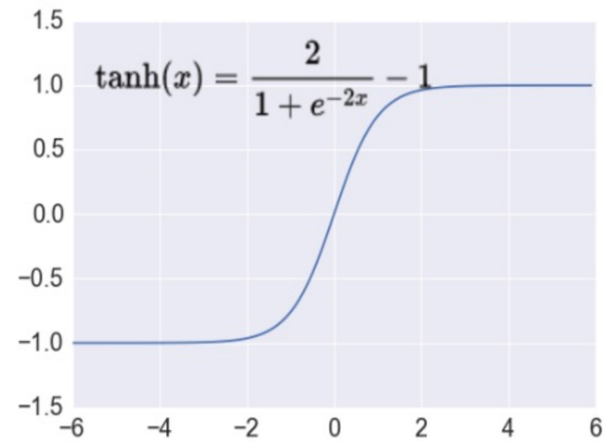
$\boldsymbol{\sigma}$ is the activation function to perform the nonlinear transform. For example, $\boldsymbol{\sigma}$ can be the following functions.



Sigmoid



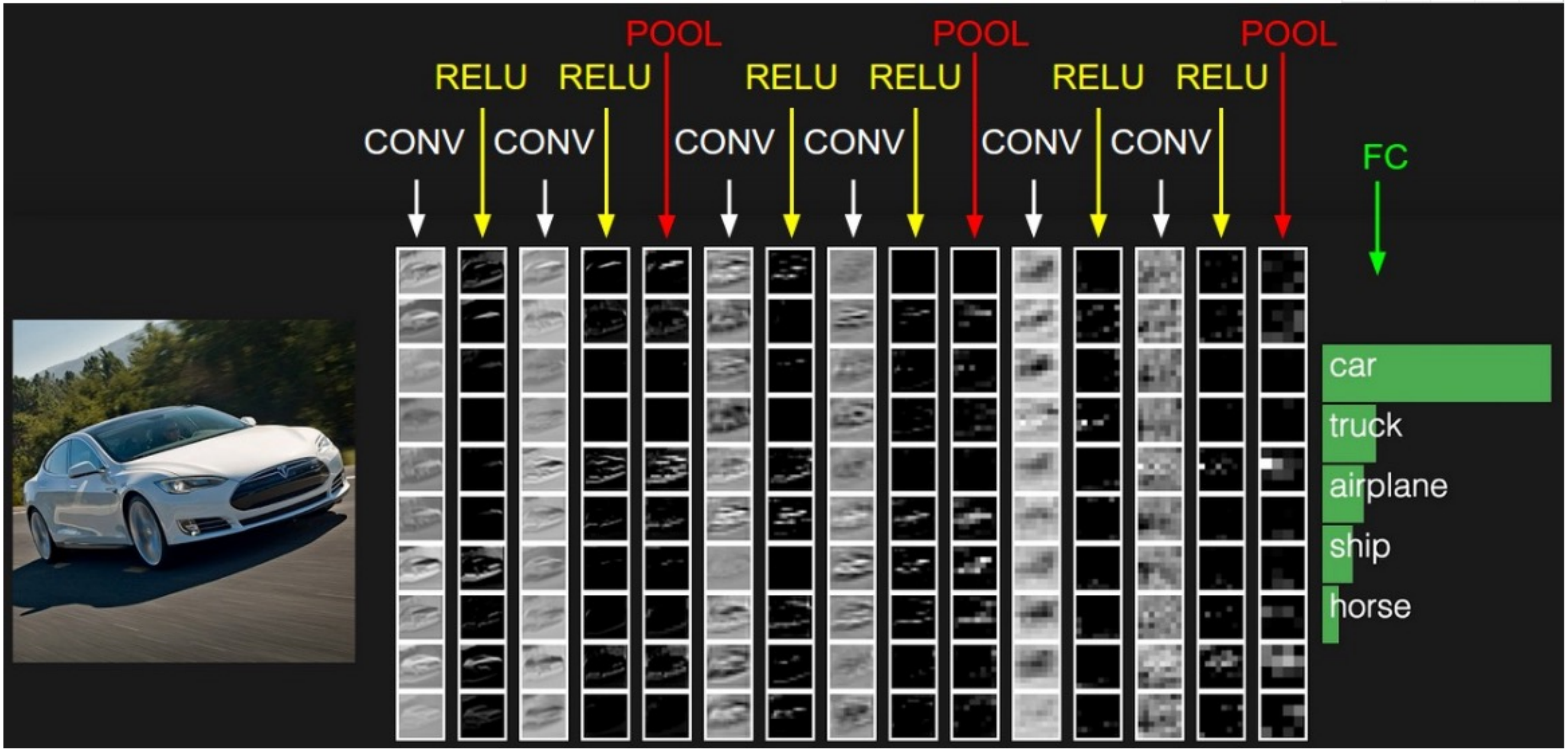
ReLU



Tanh

These functions are proposed inspired by how the human brain works.
What else?

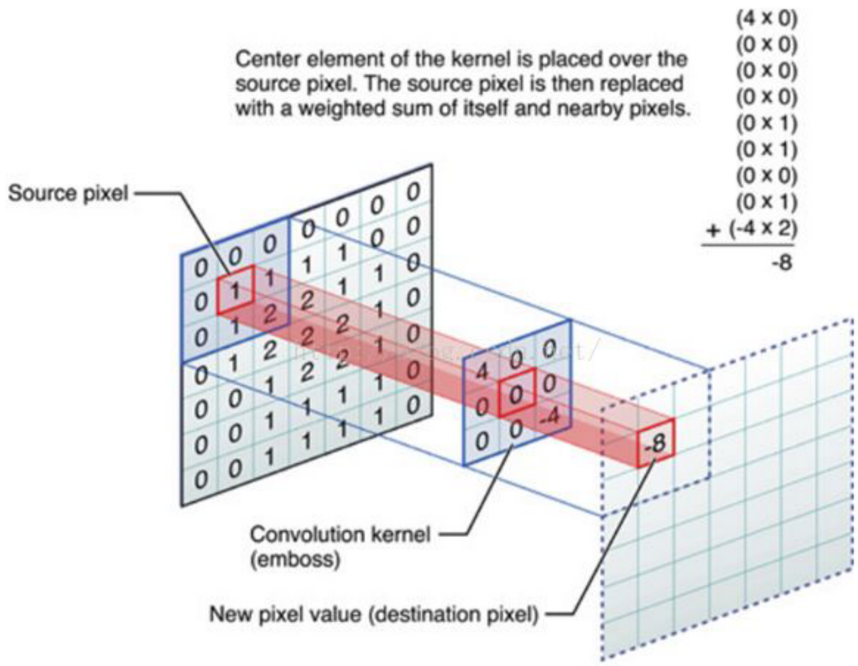
Deep Convolution Neural Networks (DCNN)

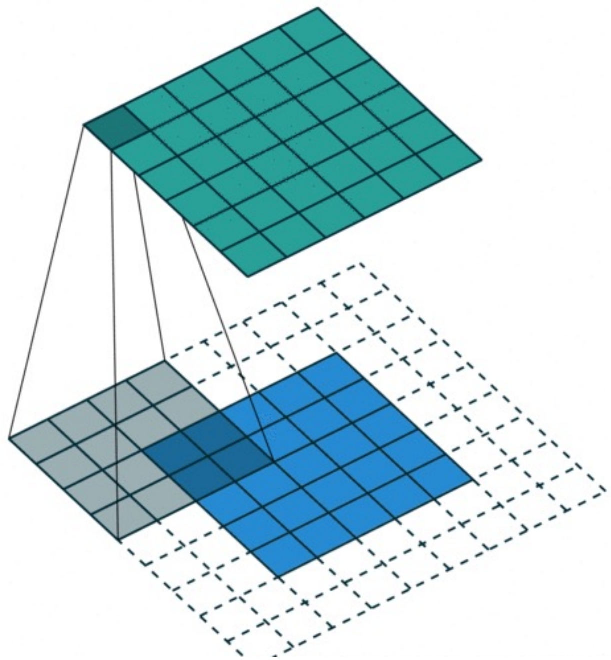


University of Pittsburgh

Deep Convolution Neural Networks (DCNN)

- Convolutional layer





Deep Convolution Neural Networks (DCNN)

- How to define a convolutional kernel?

2D (for each pixel)

Assume that the input of the current convolutional layer is $X \in \mathcal{R}^{H \times W \times c}$

Assume that we define a convolutional kernel as $k \in \mathcal{R}^{f \times f \times c}$

Assume that we define m such kernel.

The output of this convolutional layer is $\mathrm{Y} \in \mathcal{R}^{H \times W \times m}$

3D (for each Voxel)

Assume that the input of the current convolutional layer is $X \in \mathcal{R}^{H \times W \times D \times c}$

Assume that we define a convolutional kernel as $k \in \mathcal{R}^{f \times f \times f \times c}$

Assume that we define m such kernel.

The output of this convolutional layer is $\mathrm{Y} \in \mathcal{R}^{H \times W \times D \times m}$

Deep Convolution Neural Networks (DCNN)

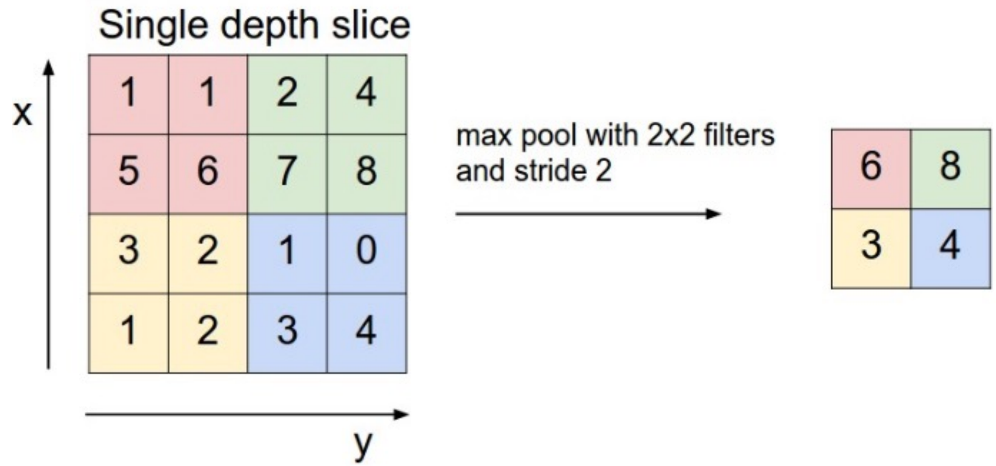
- Pooling operation (down-sample)

Why pooling?

Subsampling pixels will not change the object. Therefore, we can subsample the pixels to make the whole image smaller so that the parameters in the network will be reduced. (GPU memory, computation speed, many issues should be considered!)

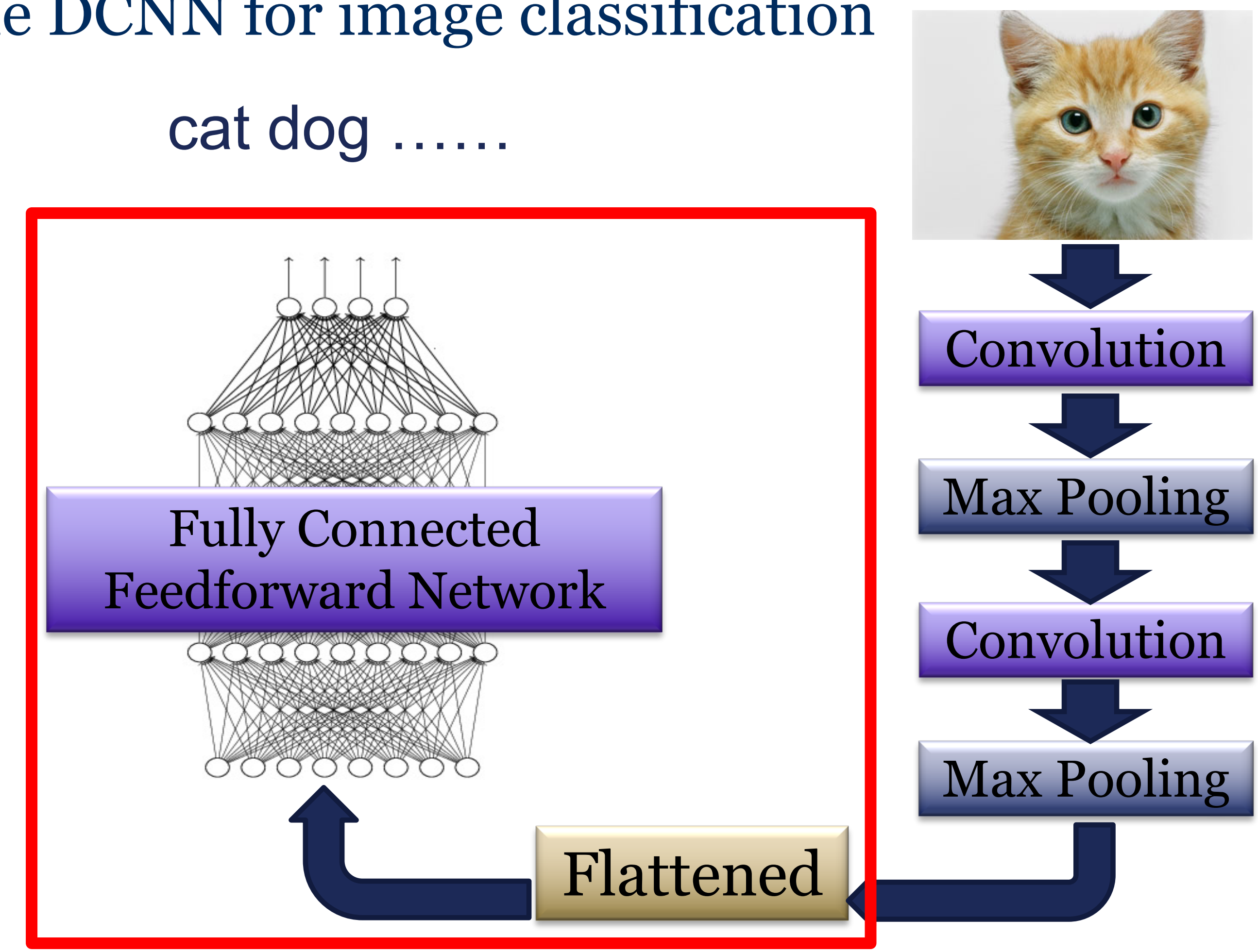
How pooling?

mean pooling, max pooling, etc.



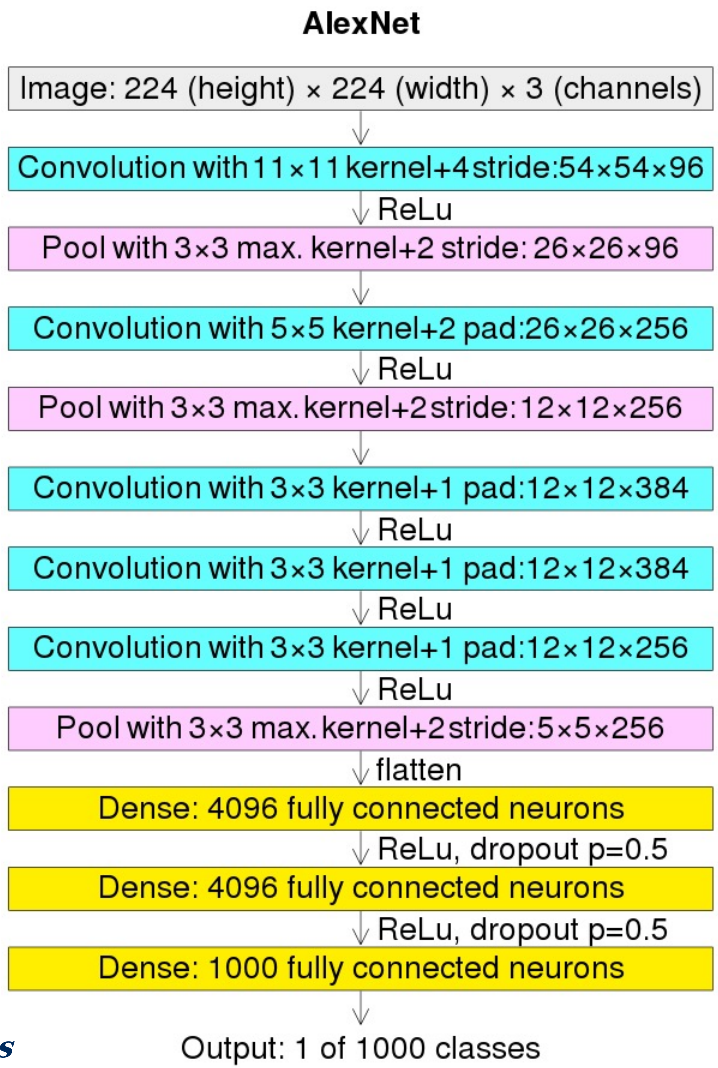
Deep Convolution Neural Networks (DCNN)

- The whole DCNN for image classification



Deep Convolution Neural Networks (DCNN)

- DCNN Generation 1 – (Alex Net)



- AlexNet manifests the effectiveness of the deep neural networks.
- It shows the importance of the depth of the network.

ImageNet Classification with Deep Convolutional Neural Networks

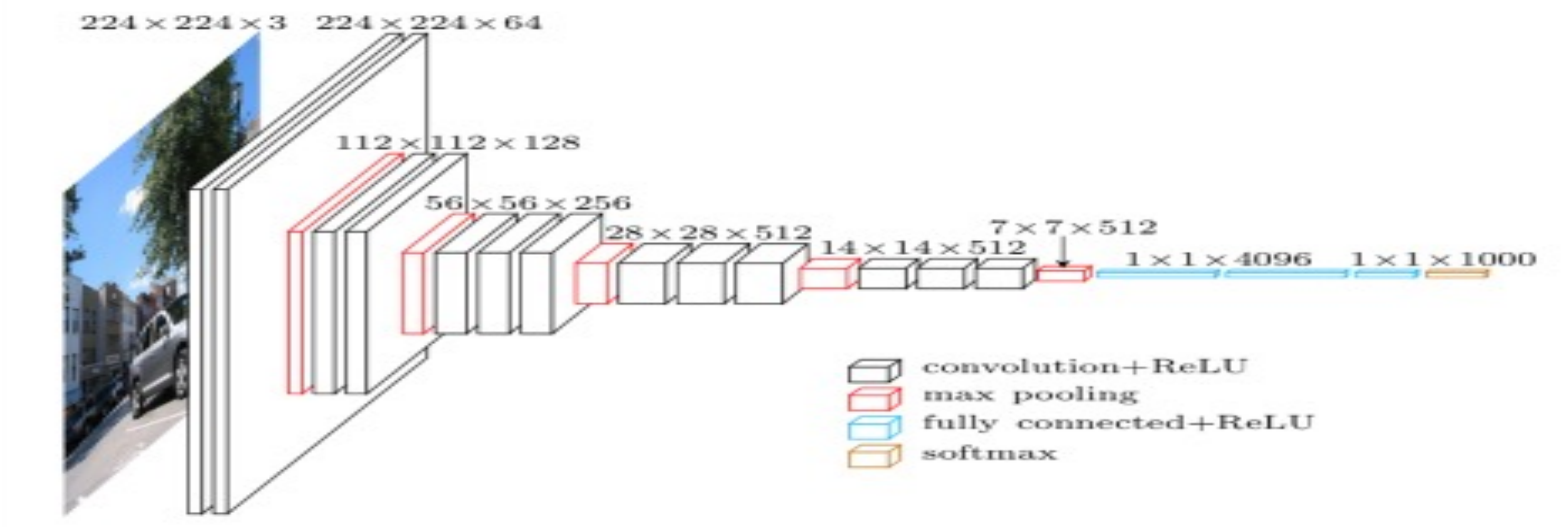
Deep Convolution Neural Networks (DCNN)

- DCNN Generation 2 – (VGG)

➢ The framework is clear

➢ Small convolutional filter (3*3) may better than the large one (e.g., 7*7)

➢ Show the probability to improve the accuracy by making the network deeper



Very Deep Convolutional Networks for Large-Scale Image Recognition

Deep Convolution Neural Networks (DCNN)

- DCNN Generation 3 – (ResNet)

When the network goes deep, we expect to obtain a better performance, but:

Neural network degradation;

The accuracy that the model obtained is even decreased;

The local correlation among gradients are very low in the deep layer;

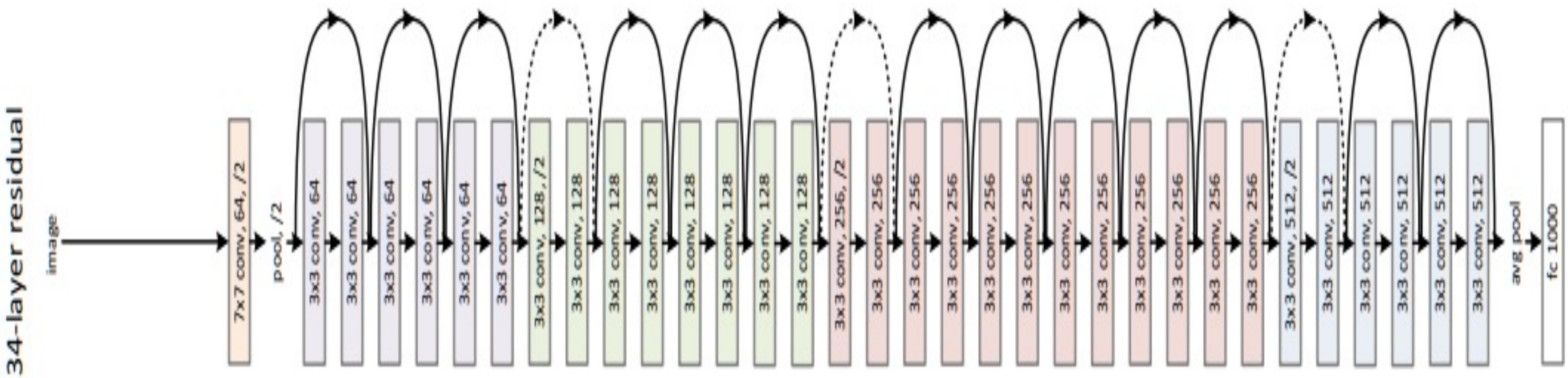
Using a lot of nonlinear activation function will lead to information missing.

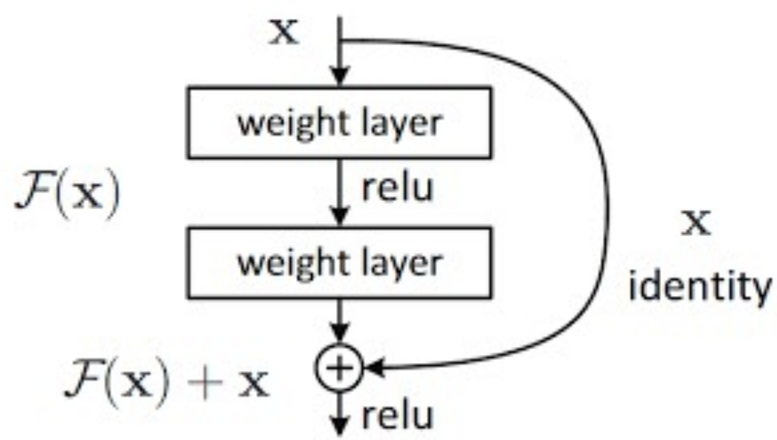
We hope the deep layers in the network to improve the performance. At least, we hope the deep layers can maintain the performance of the shallow layers (do nothing). However, they may even reduce the performance.

Deep Residual Learning for Image Recognition

Deep Convolution Neural Networks (DCNN)

- DCNN Generation 3 – (ResNet)





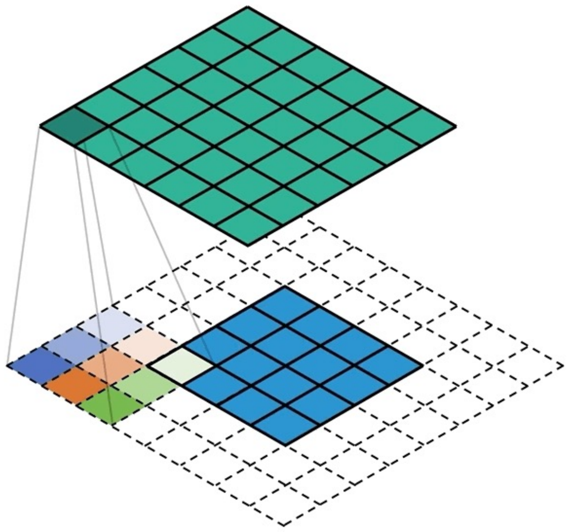
Deep Convolution Neural Networks (DCNN)

- Up-sampling

➢ Interpolation

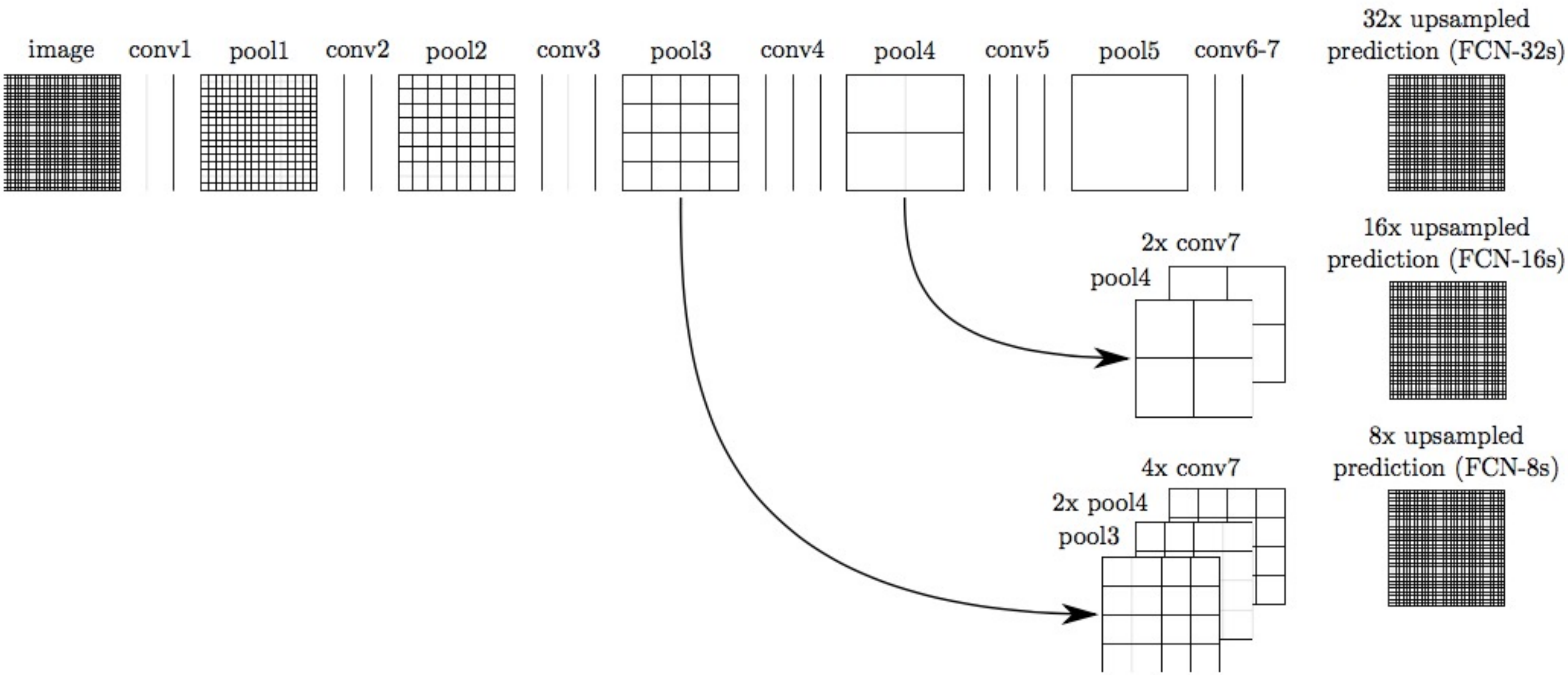
Bi-linear interpolation for 2D image, Tri-linear interpolation for 3D image, etc.

➢ Deconvolution



Deep Convolution Neural Networks (DCNN)

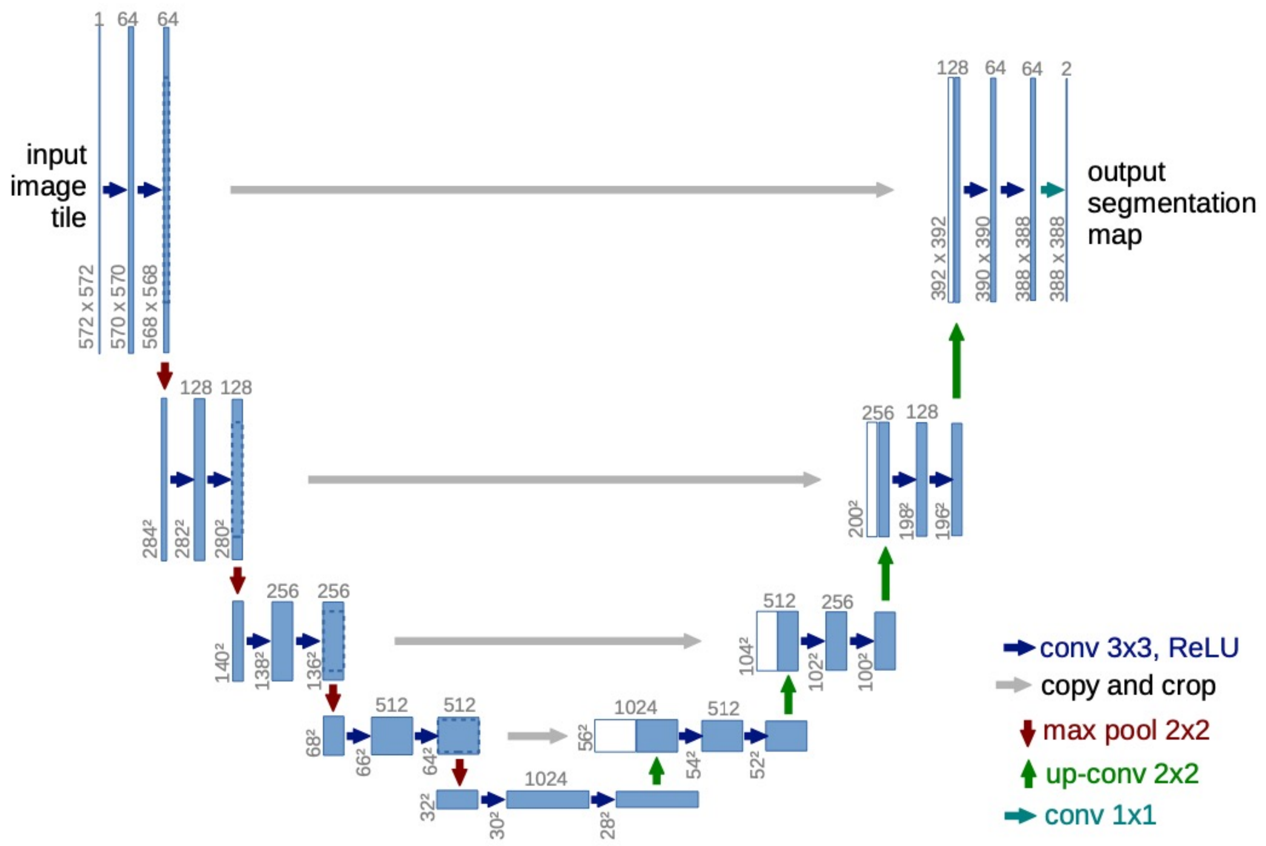
- Example: Fully Convolutional Network (FCN)



Fully Convolutional Networks for Semantic Segmentation

Deep Convolution Neural Networks (DCNN)

- Example: Deep U-Net

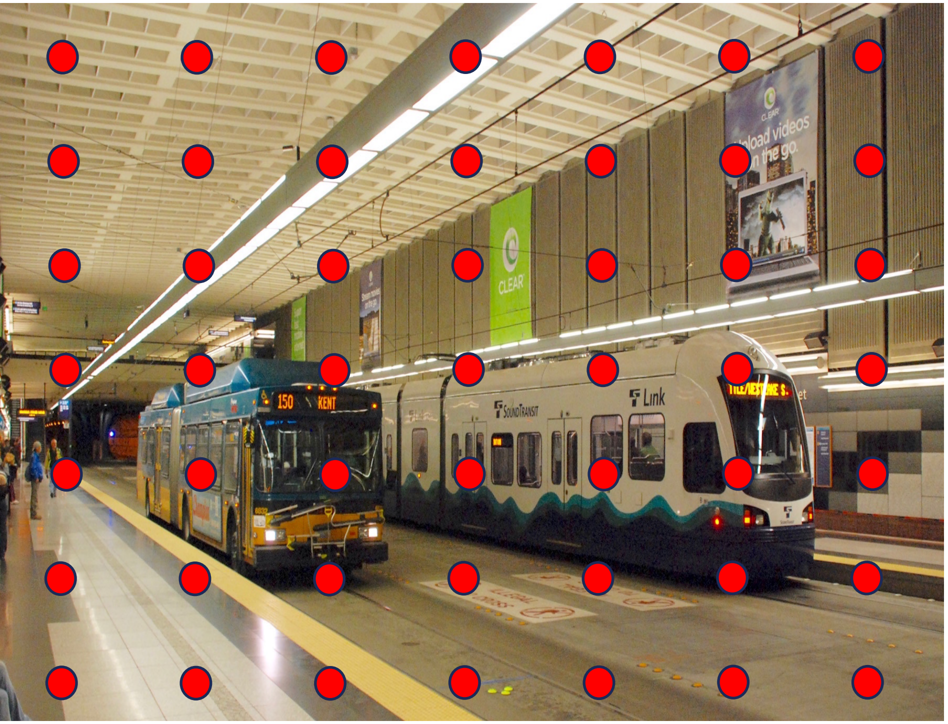


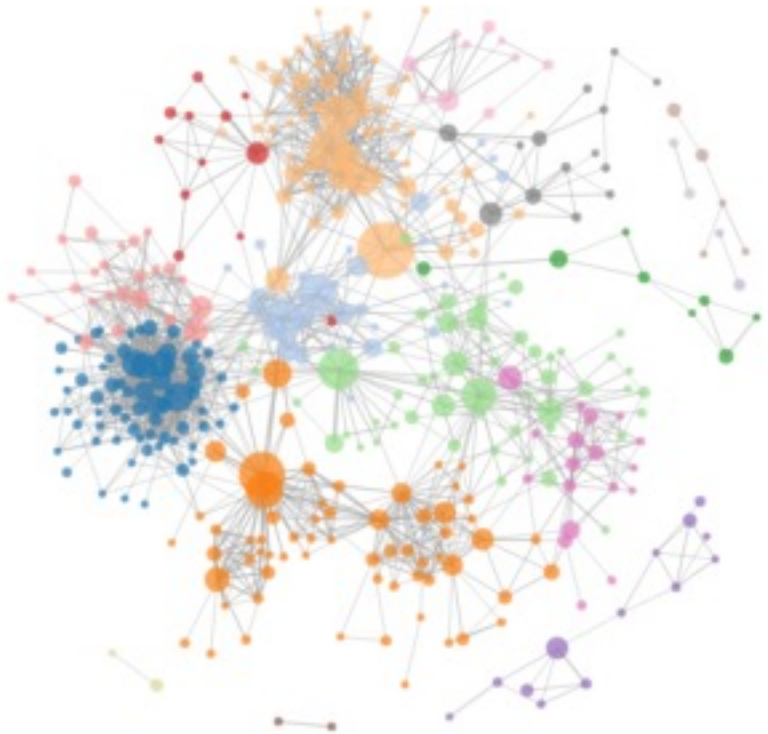
U-Net: Convolutional Networks for Biomedical Image Segmentation

Graph Neural Networks

- Beyond grid - Graph Structured Data

➢ Grid-like data and non-grid like data

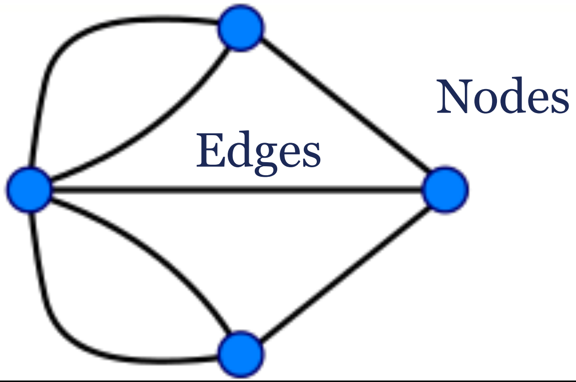


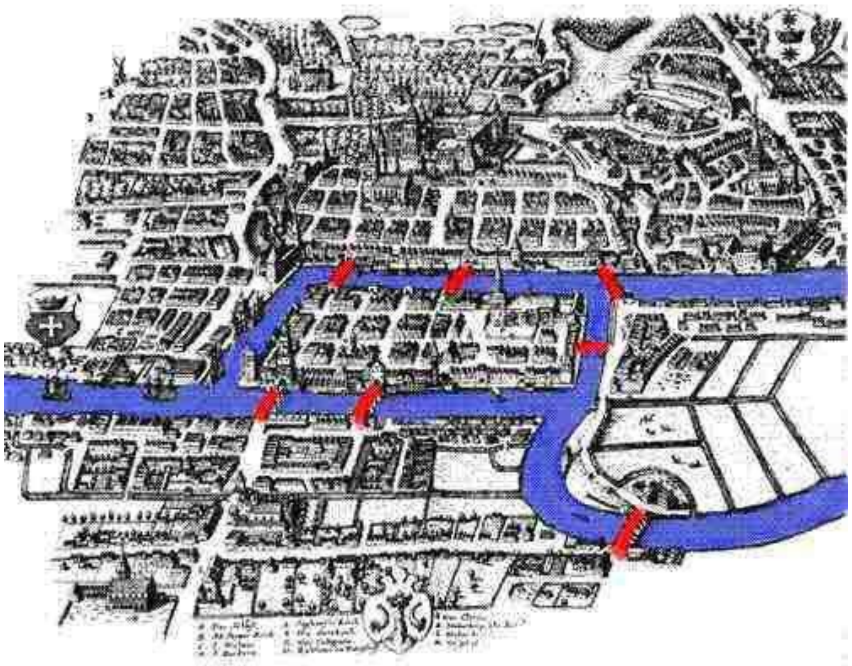


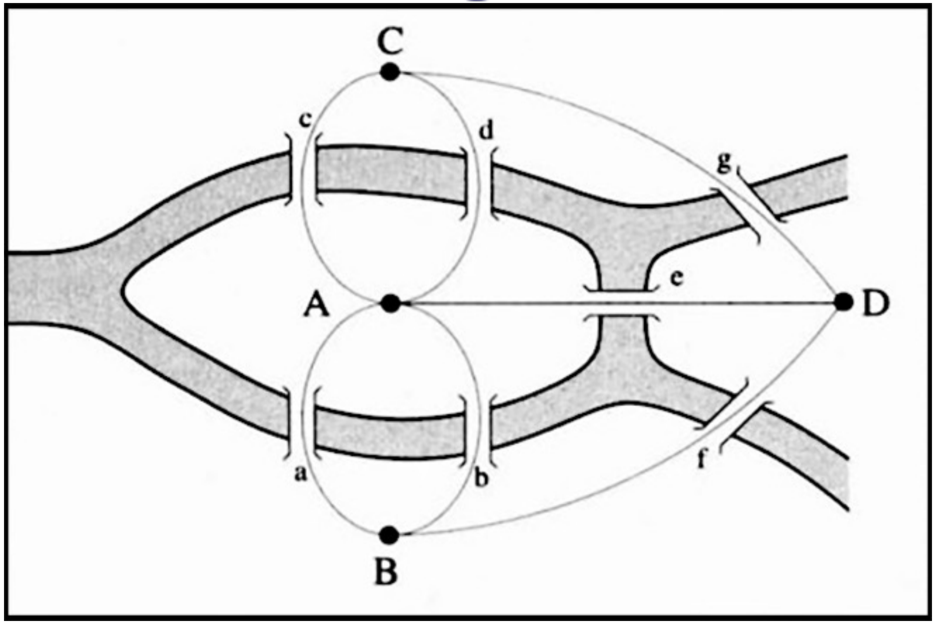
Graph Neural Networks

- Graph intro.

➢ Konigsberg 7 bridges question







Can you find a method to traverse all the 7 bridges given the condition that you only can pass through each bridge for 1 time?

Graph Neural Networks

- Graph intro.

Graph is a type of data representing the relations among objects.

- Graph categories

directed graph vs undirected graph

weighted graph vs unweighted graph

signed graph vs unsigned graph

Graph Neural Networks

- Directed graph and undirected graph

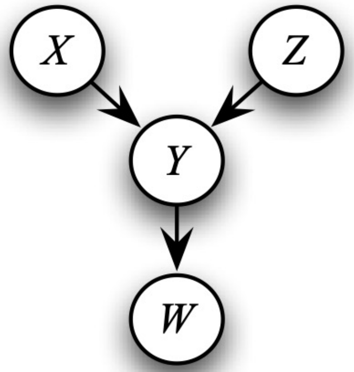
➢ directed graph (conditional random field, CRF)

Let G be a directed graph with vertices (nodes) x= $(x_1, x_2, \dots, x_d)$.

And $p(x)$ is the joint probability function on the graph nodes. We have,

$$p(x) = \prod_{j=1}^{d} p\left(x_j \middle| \pi_{x_j}\right),$$

where π_{x_j} is the set of parent nodes of x_j.



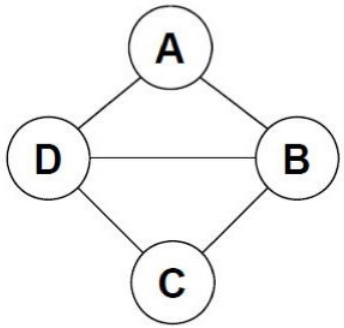
➢ undirected graph (Markov random field, MRF)

Let G be an undirected graph with vertices (nodes) x= $(x_1, x_2, \dots, x_d)$.

And $p(x)$ is the joint probability function on the graph nodes. We have,

$$p(x) = \frac{1}{Z} \prod_{c \in C} \Psi_c(x_c),$$

where $Z = \sum_x \prod_{c \in C} \Psi_c(x_c)$ is known as partition function as a normalizer.



Graph Neural Networks

- How to present a graph for computer programming?

- Present graph edges
  - Adjacency Matrix
  - Coordinate Format (COO)

- Present node features
  - Node feature matrix

Graph Neural Networks

- Graph Representation Learning

How to embed the graph into the latent space as latent features which encode the node correlation?

- Graph walk + NLP methods
- Graph Neural Networks

Graph Neural Networks

- Graph Neural Networks

Graph neural networks are designed based on the message passing mechanism to learn the graph representations (e.g., node representation). The GNNs can be formulated in terms of message propagation (MSG), message aggregation (AGG) and feature update (UPDATE):

$$output = UPDATE\ [AGG\big[MSG[G]\big]]$$

Graph Neural Networks

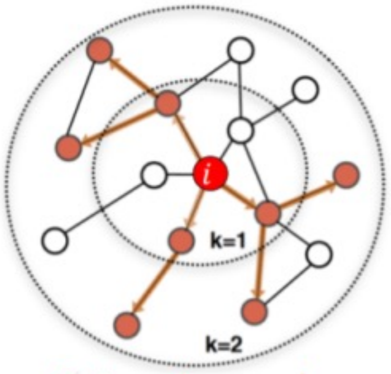
- Example of GNN, Graph Convolution Network (GCN)

➢ Propagation rule of GCN, node embedding

$$H^{(l+1)} = \sigma(\widetilde{D}^{-1/2}\tilde{A}\widetilde{D}^{-1/2}H^{(l)}W^{(l)}),$$

where $\tilde{A} = A + I$, $\widetilde{D}_{ii} = \sum_j \tilde{A}_{ij}$.

➢ Explain the how the information propagate with this propagation rule



SEMI-SUPERVISED CLASSIFICATION WITH GRAPH CONVOLUTIONAL NETWORKS

Graph Neural Networks

- Whole graph embedding

➢ What can we do now?

❖ We can obtain the node embedding via GCN.

❖ We can use the embedded node features (node latent features) to classify node.

❖ We even can use the embedded node to reconstruct the graph (e.g., VGAE).

➢ What we want to do next?

❖ How can we obtain the whole graph representation? (Embed the whole graph)?

❖ How can we do whole graph classification?

Coding, Libraries and Tools

- Python and anaconda
- Pytorch (or tensorflow)

An open-source machine learning framework exploited and provided by Facebook AI.

(An open-source machine learning framework exploited and provided by Google.)

- Torch_geometric, DGL etc for graph learning

Deep graph learning packages to fast implement the graph model.

Coding, Libraries and Tools

- Anaconda installation

https://docs.anaconda.com/anaconda/install/

- Pytorch Installation (let's install cpu version)

https://pytorch.org

- Torch_geometric (Option)

https://pytorch-geometric-temporal.readthedocs.io/en/latest/notes/installation.html

Coding, Libraries and Tools

- Demo -- GCN code download (Pytorch Version !!)

https://github.com/tkipf/pygcn

- Code Discussion

Please follow me and check the code now.

University of Pittsburgh

Thank you!