Lane Segmentation Week 8

HCT CV Course

学习目标

- Segmentation Loss
- Ensemble Learning
- 面试方法
- 课程总结
- 问题讨论

Week 8 is not End!

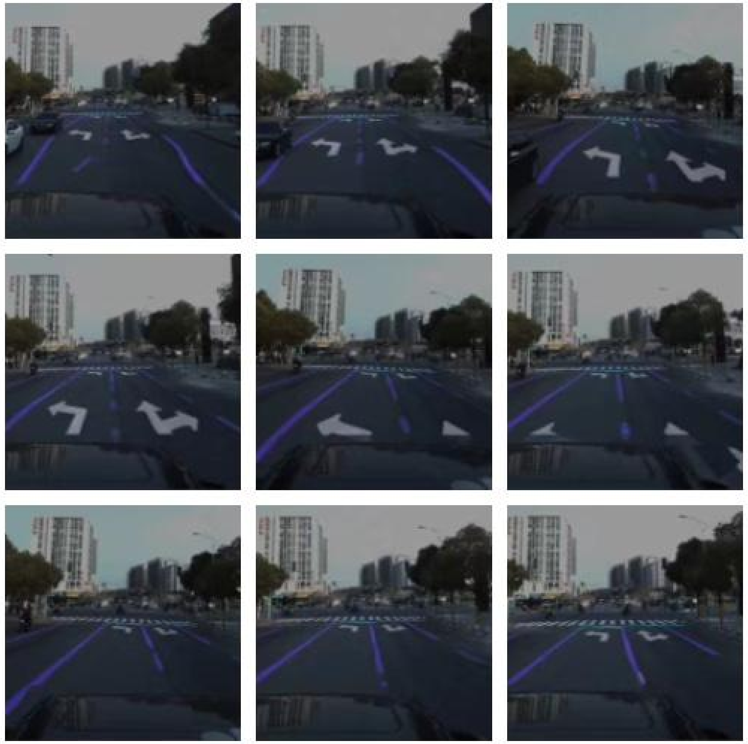
- Week09：YOLO
- Week10：Anchor Free
- Week11：Human Pose Estimation
- Week12：HRNet

后厂理工学院
houchangtech.com

Make your hand dirty

Howard Chow

✌️第一个完整手撸的工程。深度神经网络真的太神奇了，它咋知道那就是车道线呀😂

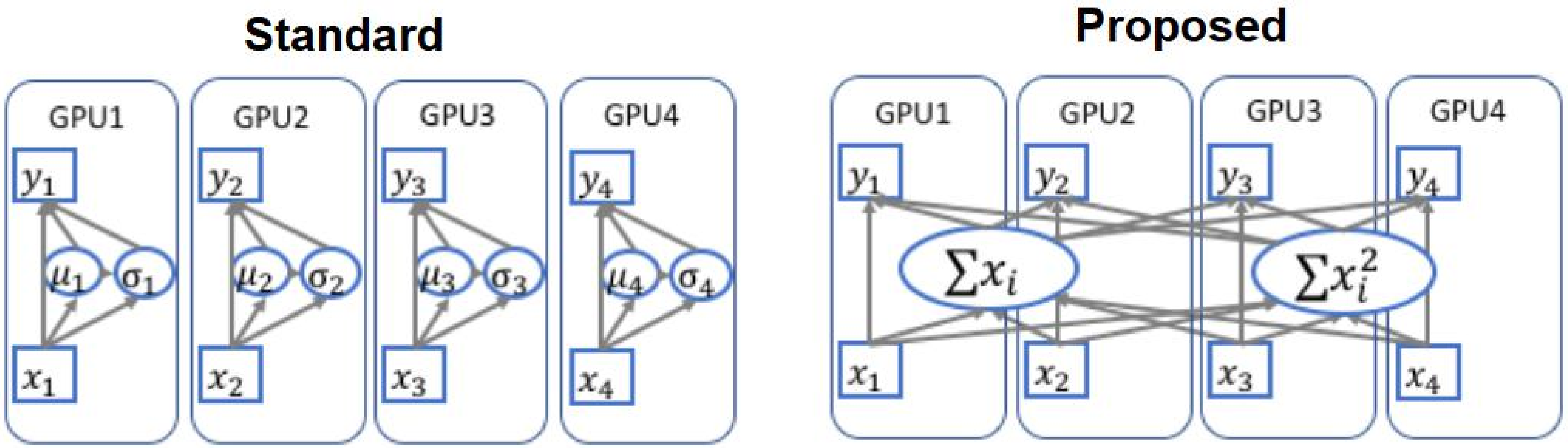


1小时前

♡ Alan Wang, AI-King

Rongfan Leo: 效果还挺好的

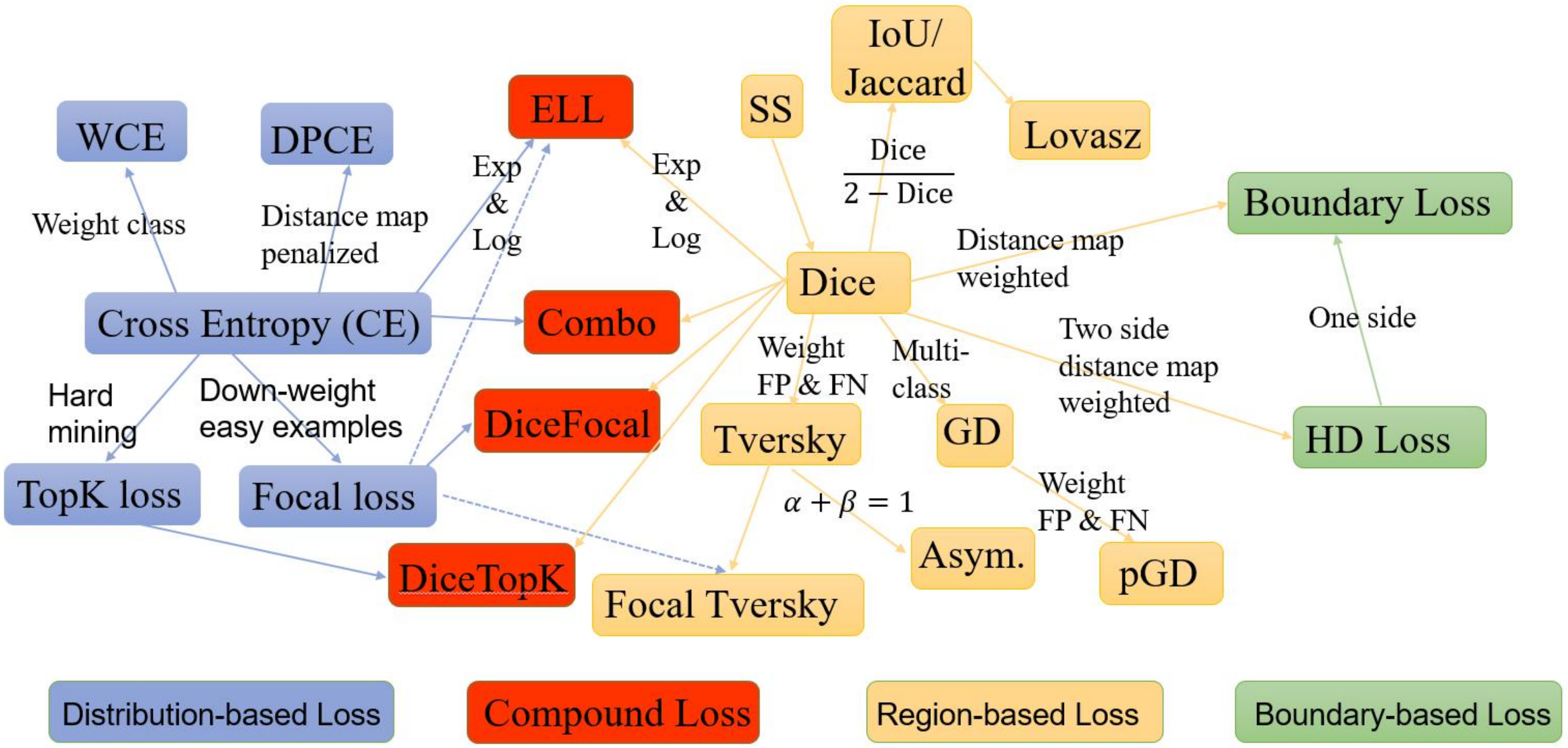
Syn BN


Standard
GPU1
GPU2
GPU3
GPU4
y_1
y_2
y_3
y_4
μ_1
σ_1
μ_2
σ_2
μ_3
σ_3
μ_4
σ_4
x_1
x_2
x_3
x_4
Proposed
GPU1
GPU2
GPU3
GPU4
y_1
y_2
y_3
y_4
$\sum x_i$
$\sum x_i^2$
x_1
x_2
x_3
x_4

Synchronized Batch Normalization

- https://github.com/vacancy/Synchronized-BatchNorm-PyTorch

Loss



WCE
DPCE
ELL
SS
IoU/ Jaccard
Lovasz
Weight class
Distance map penalized
Exp & Log
Exp & Log
$\frac{\text{Dice}}{2-\text{Dice}}$
Boundary Loss
Dice
Distance map weighted
Cross Entropy (CE)
Combo
One side
Two side distance map weighted
Weight FP & FN
Multi-class
Hard mining
Down-weight easy examples
DiceFocal
Tversky
GD
HD Loss
TopK loss
Focal loss
$\alpha + \beta = 1$
Weight FP & FN
Asym.
pGD
DiceTopK
Focal Tversky
Distribution-based Loss
Compound Loss
Region-based Loss
Boundary-based Loss

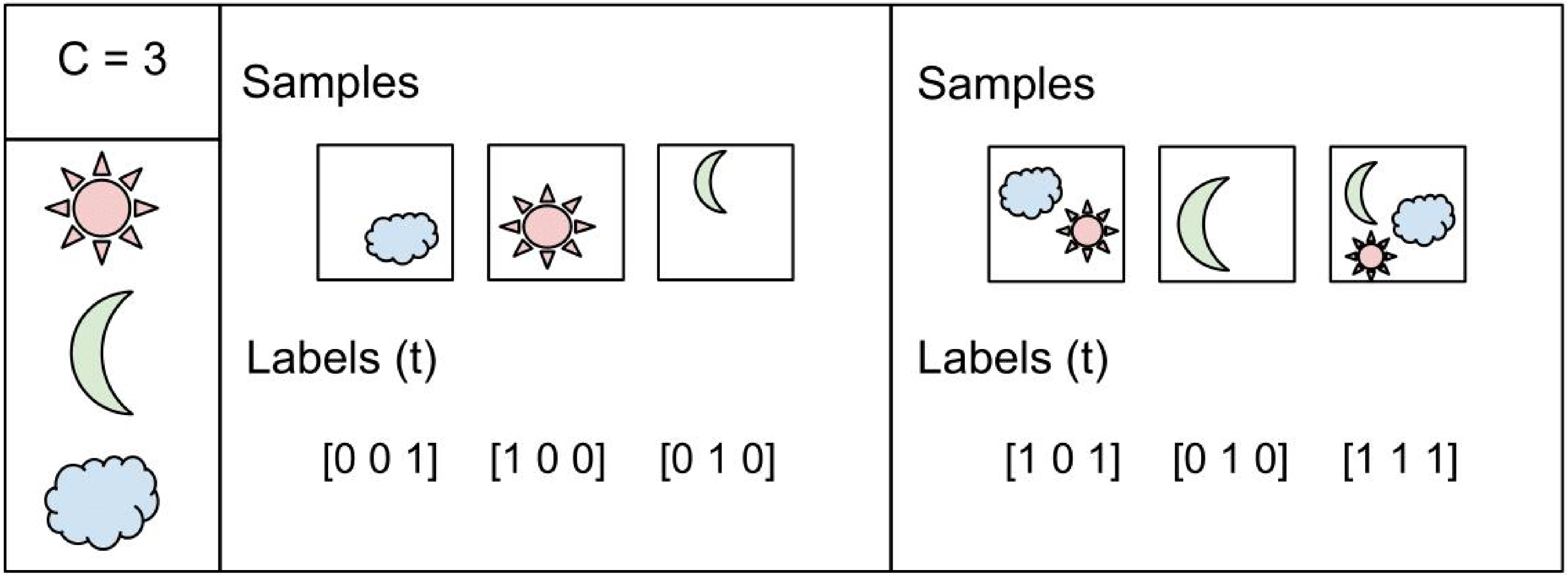
Loss

- A collection of loss functions for medical image segmentation
- https://github.com/JunMa11/SegLoss

后厂理工学院
houchangtech.com

Multi-Class

Multi-Label



Multi-Class Classification

- One-of-many classification.
- Each sample can belong to ONE of C classes.
- one-hot vector
- a positive class and C-1 negative classes

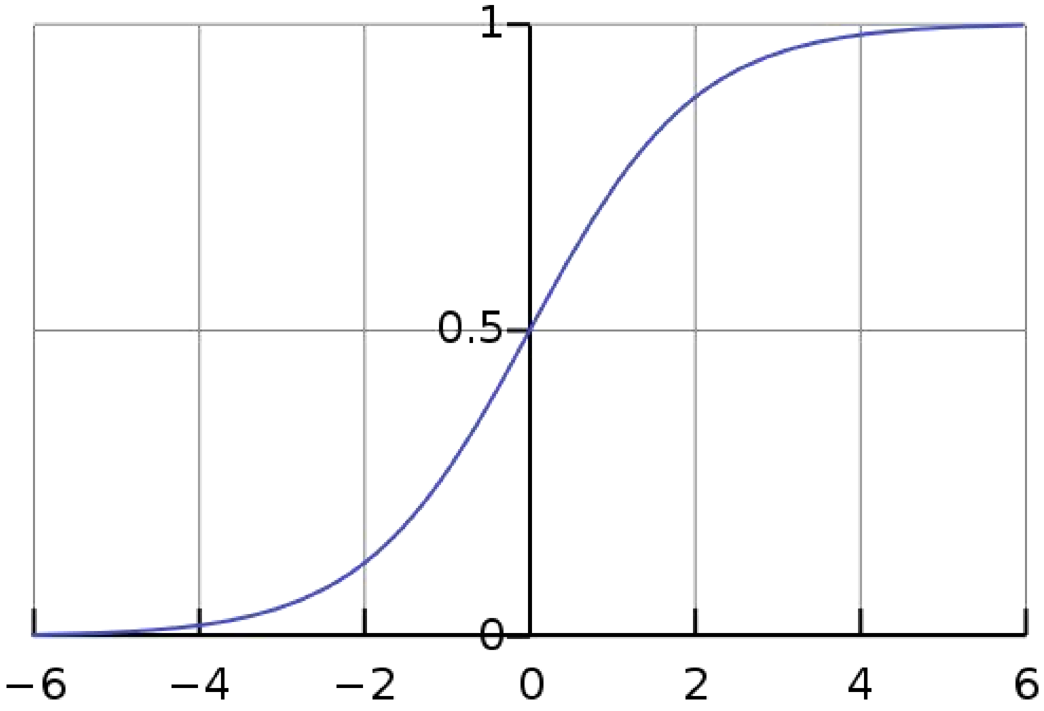
Multi-Label Classification

- Each sample can belong to more than one class.

activation function

- Activation functions are used to transform vectors before computing the loss in the training phase.

sigmoid



后厂理工学院
houchangtech.com

Softmax

Input pixels, $\mathbf{x}$

Forward propagation →

Shape: (3, 32, 32)

Feedforward output, $\mathbf{y}_i$

cat	dog	horse
5	4	2
4	2	8
4	4	1

Shape: (3,)

Softmax function →

Softmax output, $\mathbf{S}(\mathbf{y}_i)$

cat	dog	horse
0.71	0.26	0.04
0.02	0.00	0.98
0.49	0.49	0.02

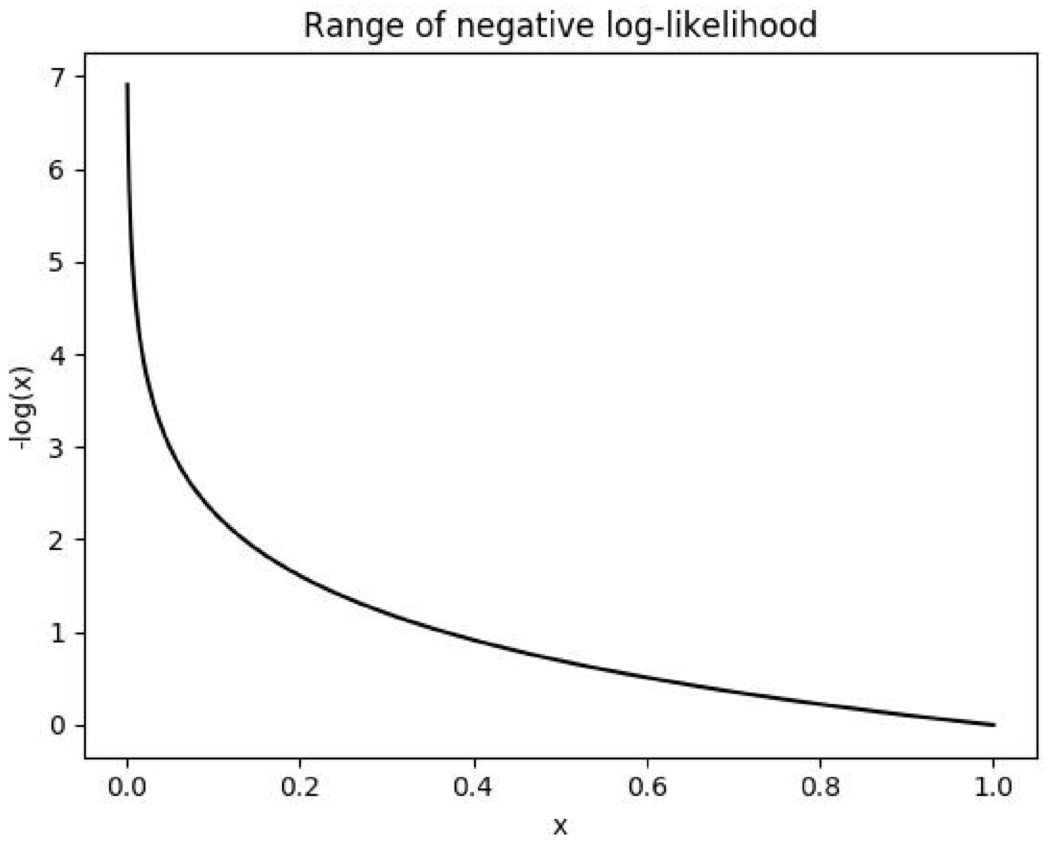
Shape: (3,)

Negative Log-Likelihood (NLL)

- loss function
- summed for all the correct classes

$$L(\mathbf{y}) = -\log(\mathbf{y})$$

Negative Log-Likelihood (NLL)



Negative Log-Likelihood (NLL)

Input pixels, $\mathbf{x}$

Softmax output, $\mathbf{S}(\mathbf{y}_i)$

cat	dog	horse
0.71	0.26	0.04
0.02	0.00	0.98
0.49	0.49	0.02

The correct class is highlighted in red

$-\log(a)$ at the correct classes $\longrightarrow$

Loss, $\mathbf{L}(a)$

NLL
0.34
0.02
0.71

Total: **1.07**

Correct classes are known because we are training

Predictor confidence of **horse** is high. *Lower unhappiness.*

Predictor confidence of **dog** is low. *Higher unhappiness.*

Derivative of the Softmax

- https://eli.thegreenplace.net/2016/the-softmax-function-and-its-derivative/

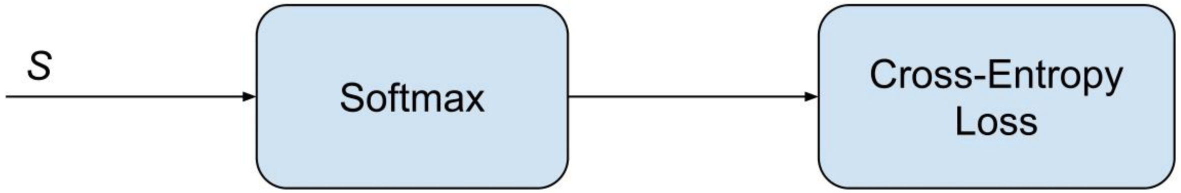
Cross-Entropy loss

- usually an activation function (Sigmoid / Softmax) is applied to the scores before the CE Loss computation

$$Cross - entropy\ loss = -\sum_{c=1}^{M} Y \log(P)$$

Categorical Cross-Entropy loss

- Also called Softmax Loss. It is a Softmax activation plus a Cross-Entropy loss.



$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \qquad CE = -\sum_i^C t_i log(f(s)_i)$$

后厂理工学院
houchangtech.com

Binary Cross-Entropy Loss

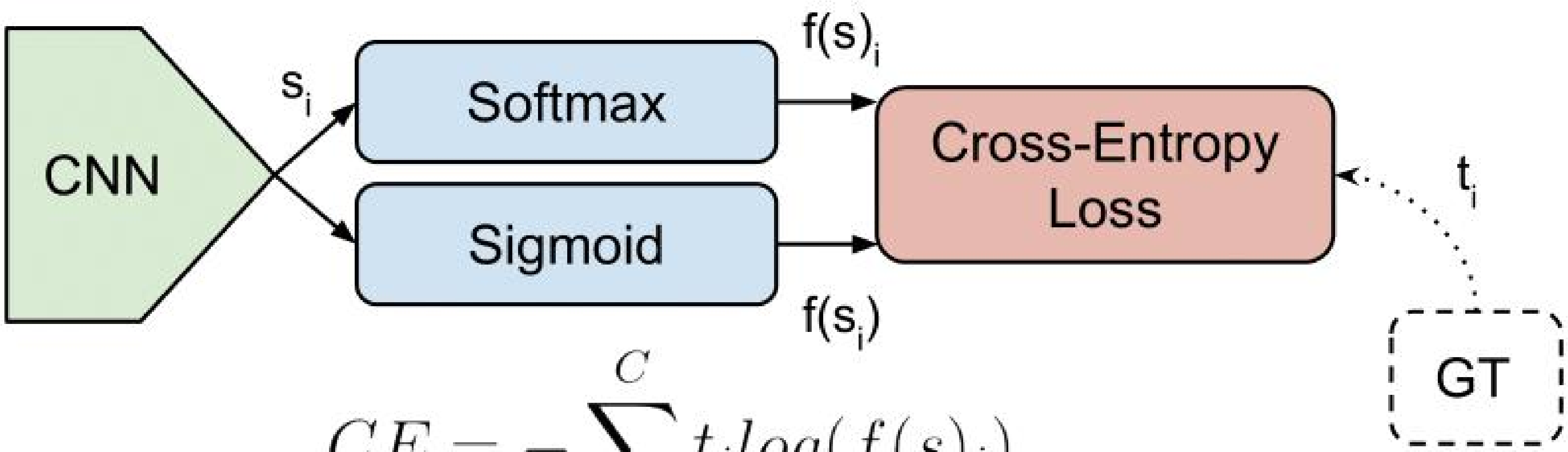
- Also called Sigmoid Cross-Entropy loss. It is a Sigmoid activation plus a Cross-Entropy loss.

$$CE = -t_1 log(f(s_1)) - (1 - t_1) log(1 - f(s_1))$$

$$f(s_i) = \frac{1}{1 + e^{-s_i}}$$

后厂理工学院
houchangtech.com

Cross-Entropy Loss



$$CE = -\sum_{i}^{C} t_i log(f(s)_i)$$

$$CE = -\sum_{i=1}^{C'=2} t_i log(f(s_i)) = -t_1 log(f(s_1)) - (1 - t_1) log(1 - f(s_1))$$

Information Theory

- Probability and Information Theory
- http://www.deeplearningbook.org/contents/prob.html

Probability

- frequentist probability
- Bayesian probability

后厂理工学院
houchangtech.com

Bayes's Rule

- we knowP(y | x) and need to knowP(x | y)

$$P(\mathrm{x} \mid \mathrm{y}) = \frac{P(\mathrm{x})P(\mathrm{y} \mid \mathrm{x})}{P(\mathrm{y})}.$$

Conditional Probability

$$P(\mathrm{y} = y \mid \mathrm{x} = x) = \frac{P(\mathrm{y} = y, \mathrm{x} = x)}{P(\mathrm{x} = x)}.$$

What is Information

- Probability
- frequentist probability

self-information

- Likely events should have low information content, and in the extreme case,events that are guaranteed to happen should have no information contentwhatsoever.
- Less likely events should have higher information content.
- Independent events should have additive information. For example, findingout that a tossed coin has come up as heads twice should convey twice asmuch information as finding out that a tossed coin has come up as headsonce.

self-information

$$I(x) = -\log P(x).$$

大道至简

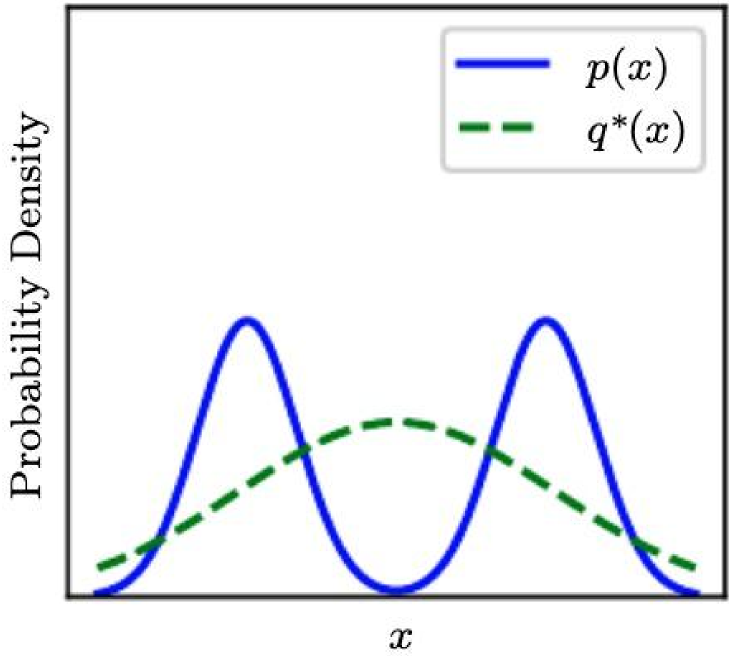
Kullback-Leibler (KL) divergence

$$D_{\mathrm{KL}}(P\|Q) = \mathbb{E}_{\mathrm{x}\sim P}\left[\log \frac{P(x)}{Q(x)}\right] = \mathbb{E}_{\mathrm{x}\sim P}\left[\log P(x) - \log Q(x)\right].$$

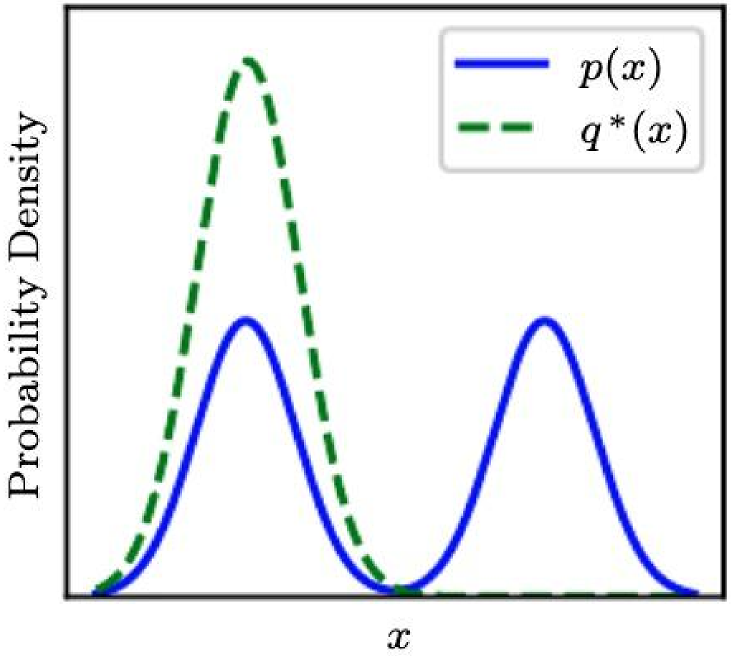
后厂理工学院
houchangtech.com

Kullback-Leibler (KL) divergence

$q^* = \text{argmin}_q D_{\text{KL}}(p||q)$



$q^* = \text{argmin}_q D_{\text{KL}}(q||p)$



后厂理工学院
houchangtech.com

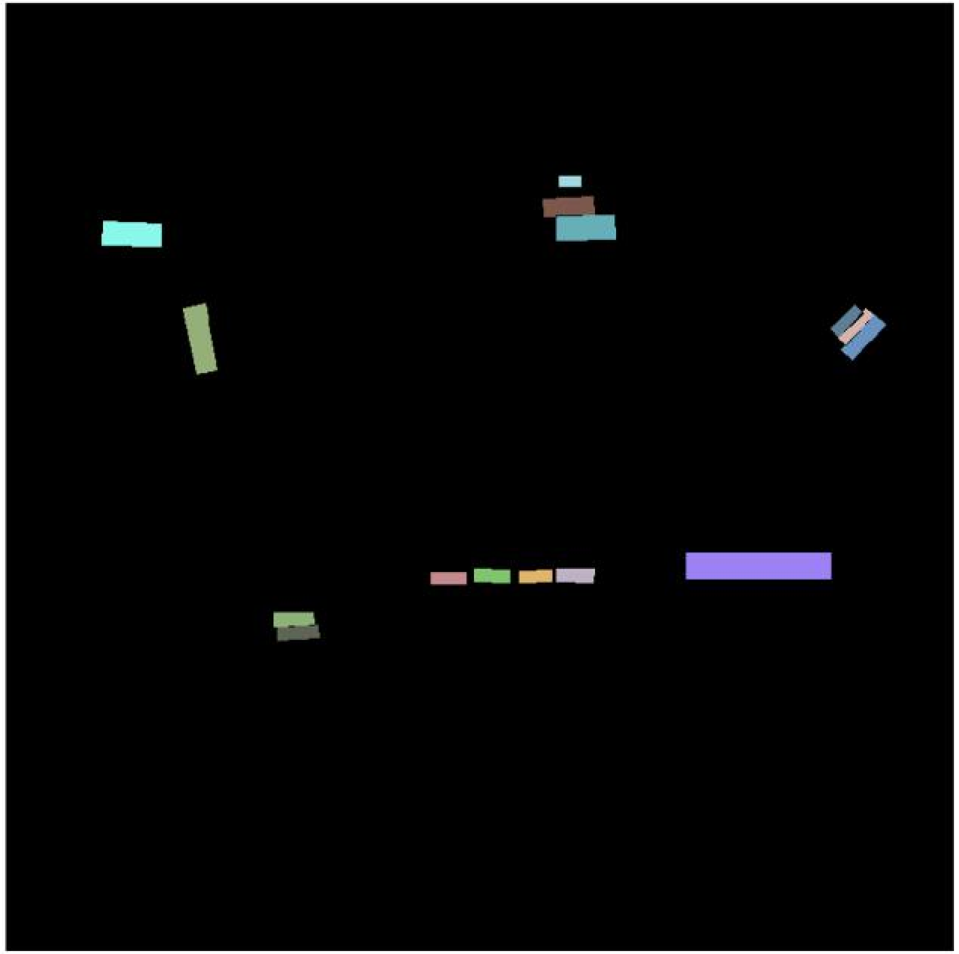
Cross-Entropy

$$D_{KL}(p \parallel q) = \underbrace{H(p,q)}_{\text{Cross entropy}} - \overbrace{H(p)}^{\text{Entropy}}$$

class imbalance

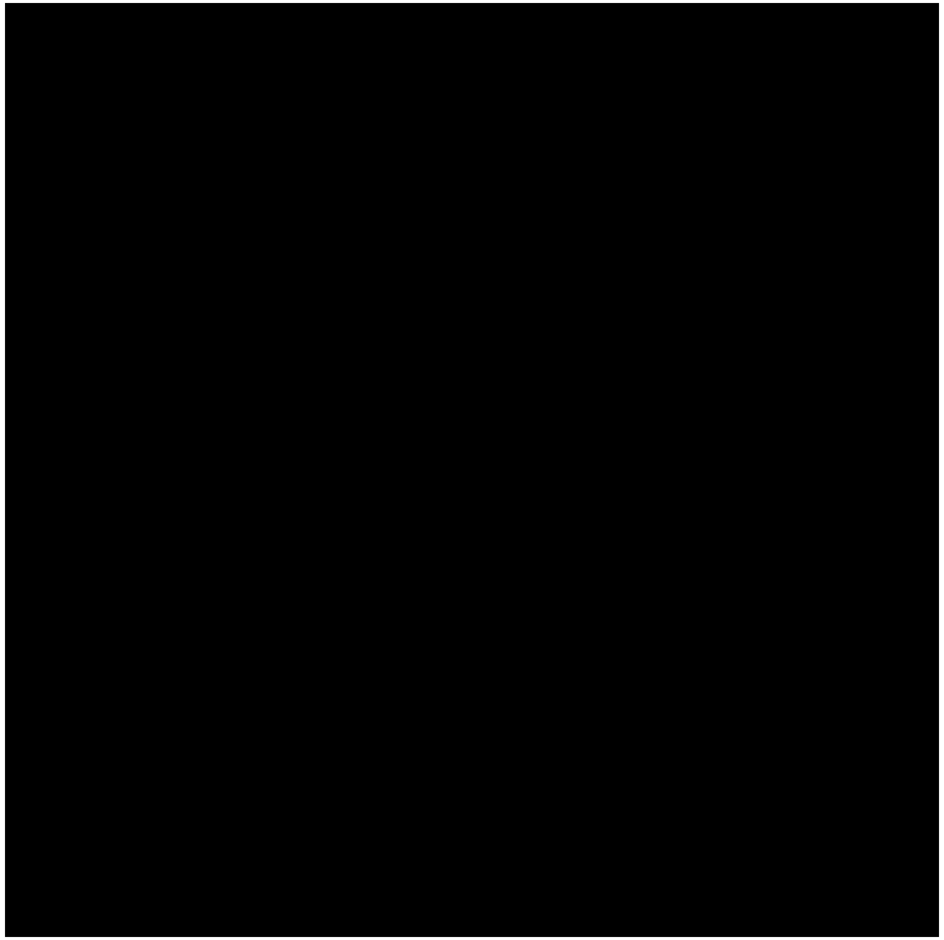
- Unfortunately, class imbalance is prevalent in many real world data sets, so it can't be ignored

class imbalance



后厂理工学院
houchangtech.com

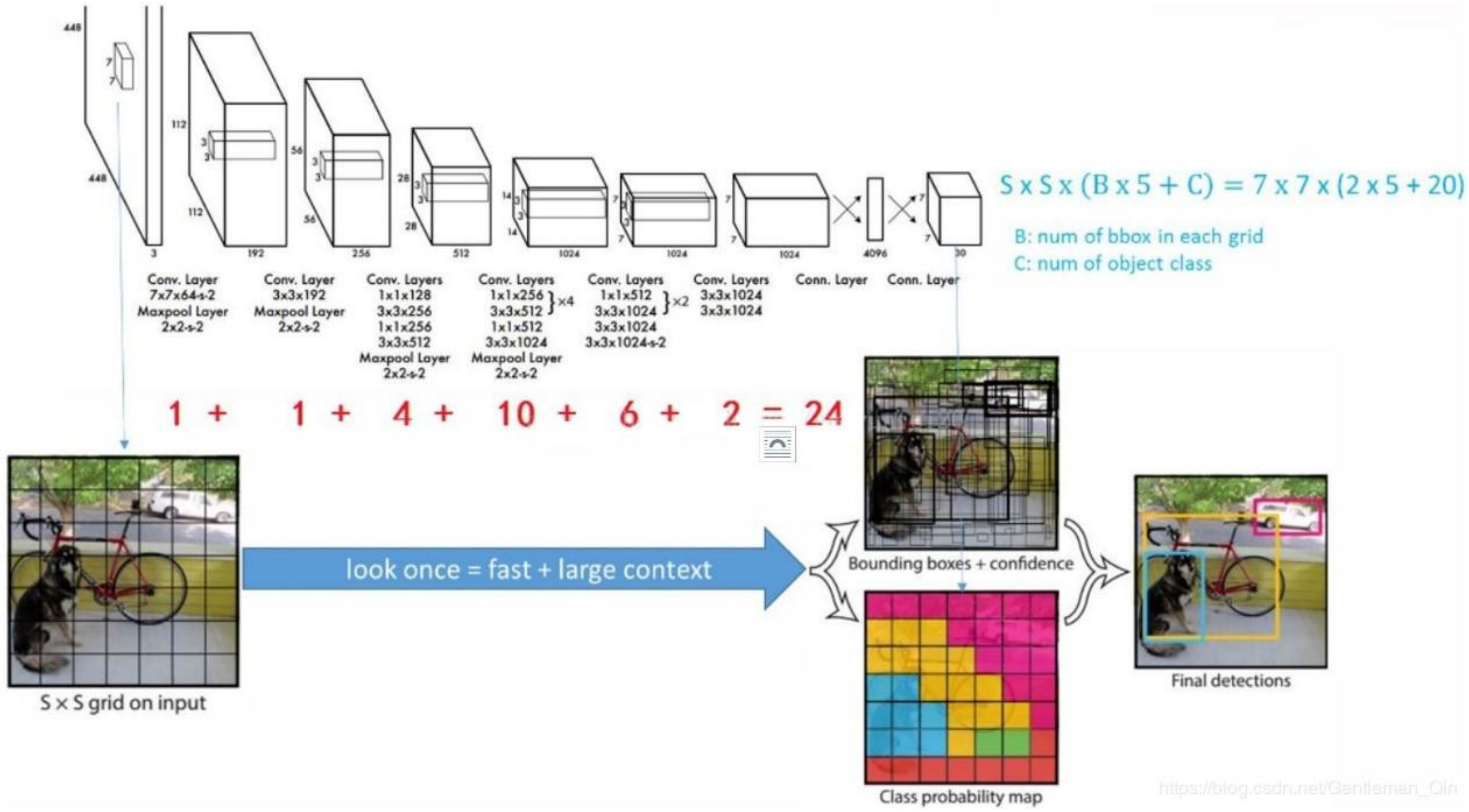
class imbalance



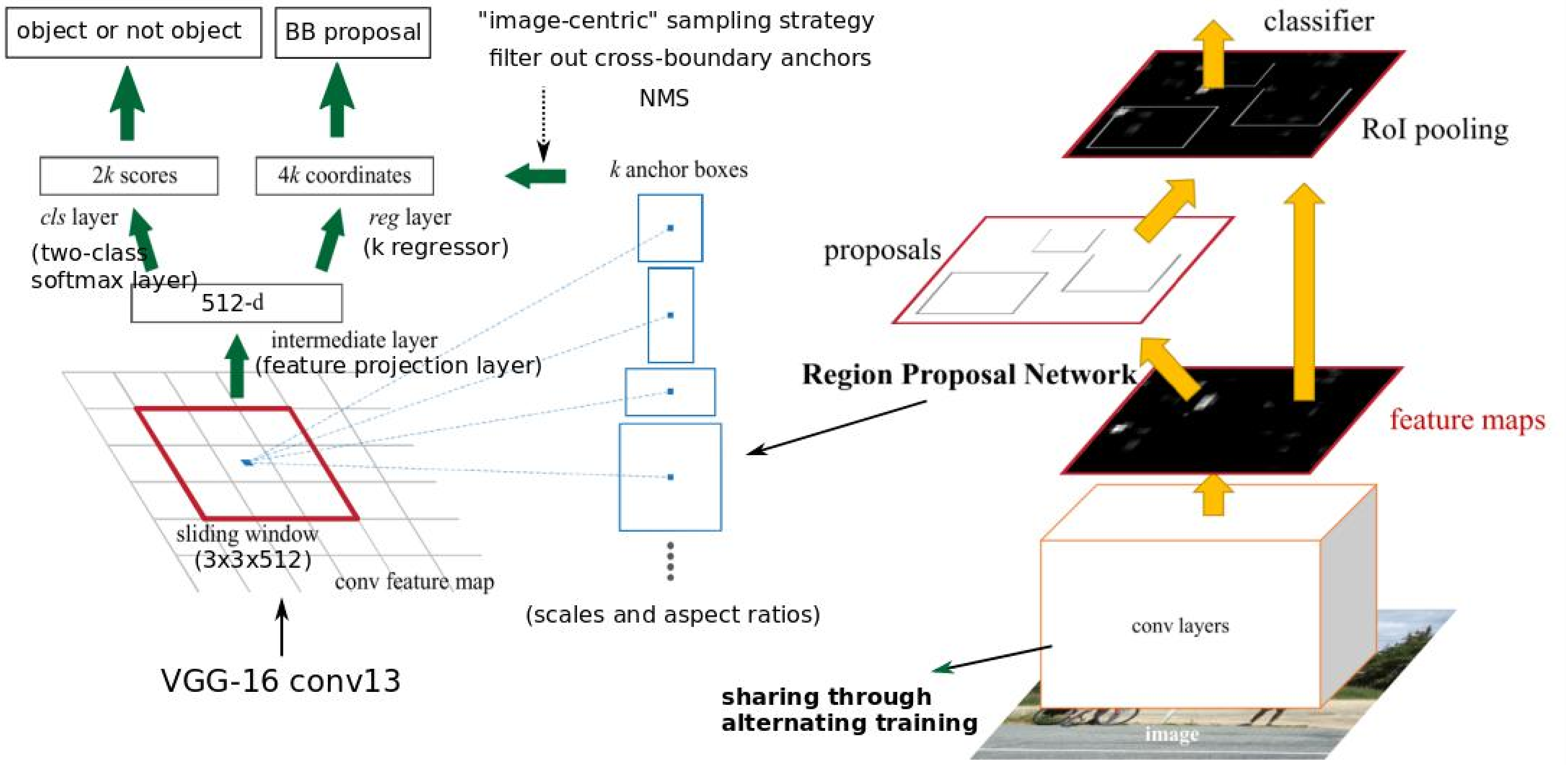
class imbalance

- An ordinary way to overcome this problem is to do sampling to balance the dataset. If you have fewer samples in some classes, you sample them, or duplicate them so that the classes are balanced.

one-stage


Conv. Layer
7x7x64-s-2
Maxpool Layer
2x2-s-2
Conv. Layer
3x3x192
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x128
3x3x256
1x1x256
3x3x512
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x256
3x3x512 }x4
1x1x512
3x3x1024
Maxpool Layer
2x2-s-2
Conv. Layers
1x1x512
3x3x1024 }x2
3x3x1024
3x3x1024-s-2
Conv. Layers
3x3x1024
3x3x1024
Conn. Layer
Conn. Layer
S x S x (B x 5 + C) = 7 x 7 x (2 x 5 + 20)
B: num of bbox in each grid
C: num of object class
1 + 1 + 4 + 10 + 6 + 2 = 24
look once = fast + large context
S × S grid on input
Bounding boxes + confidence
Class probability map
Final detections

two-stage


object or not object
BB proposal
"image-centric" sampling strategy
filter out cross-boundary anchors
NMS
2k scores
4k coordinates
k anchor boxes
cls layer
(two-class softmax layer)
reg layer
(k regressor)
512-d
intermediate layer
(feature projection layer)
sliding window
(3x3x512)
conv feature map
VGG-16 conv13
(scales and aspect ratios)
classifier
RoI pooling
proposals
Region Proposal Network
feature maps
conv layers
image
sharing through
alternating training

class imbalance

- the extreme foreground-background class imbalance encountered during training of dense detectors is the central cause

class imbalance

- In R-CNN and Fast R-CNN, because the model is not end-to-end and it consists several distinct models, the class imbalanced problem could be solved by sampling more minor class samples or removing major class samples. However, in end-to-end models, sampling to balance the classes could not be easily achieved

focal loss

- Mathematically, sampling is equivalent to adding weights to samples.

focal loss

- Focal Loss for Dense Object Detection
- https://arxiv.org/abs/1708.02002

Task

- Synthetic Financial Datasets For Fraud Detection
- https://www.kaggle.com/ntnu-testimon/paysim1

Note

dice loss

- V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation
- https://arxiv.org/abs/1606.04797

Note

Note

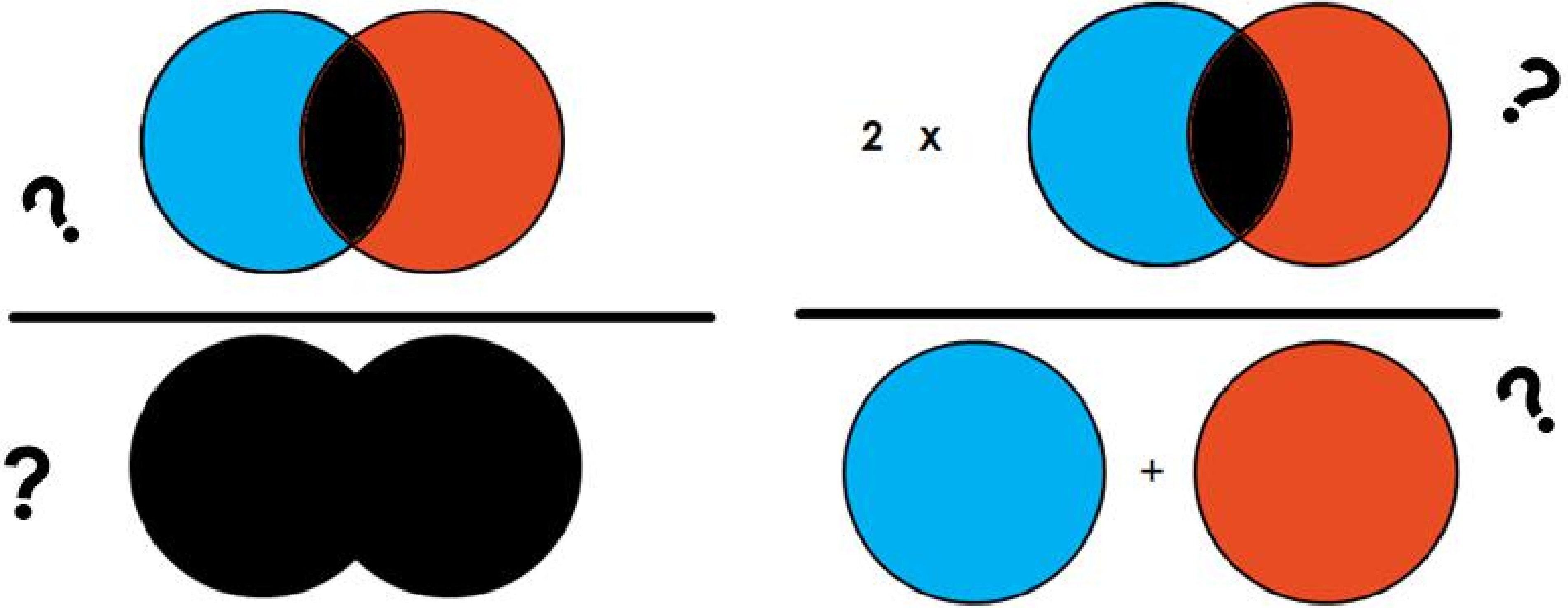
Combinations

$$\mathrm{CE}(p, \hat{p}) + \mathrm{DL}(p, \hat{p})$$

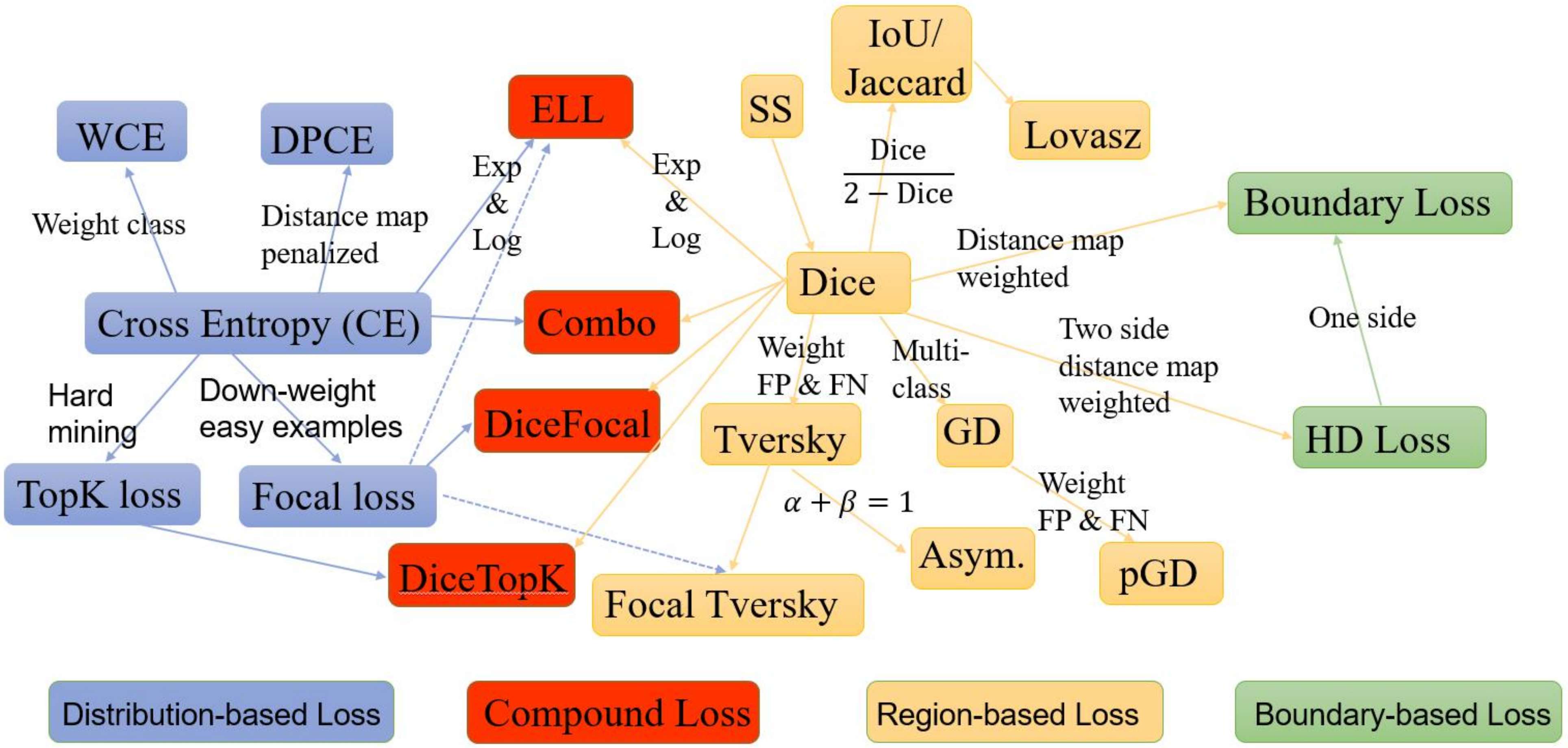
Note that CE returns a tensor, while DL returns a scalar for each image in the batch. This way we combine local (CE) with global information (DL).

后厂理工学院
houchangtech.com

IoU and Dice Coefficient



Loss


WCE
DPCE
ELL
SS
IoU/ Jaccard
Lovasz
Weight class
Distance map penalized
Exp & Log
Exp & Log
Dice / 2 − Dice
Boundary Loss
Distance map weighted
Dice
Cross Entropy (CE)
Combo
One side
Two side distance map weighted
Weight FP & FN
Multi-class
Hard mining
Down-weight easy examples
DiceFocal
Tversky
GD
HD Loss
TopK loss
Focal loss
α + β = 1
Weight FP & FN
Asym.
DiceTopK
pGD
Focal Tversky
Distribution-based Loss
Compound Loss
Region-based Loss
Boundary-based Loss

Note

面试方法

- 认真准备
- 认真准备
- 认真准备

Note

Note

Note

Note

Note

Ensemble Learning

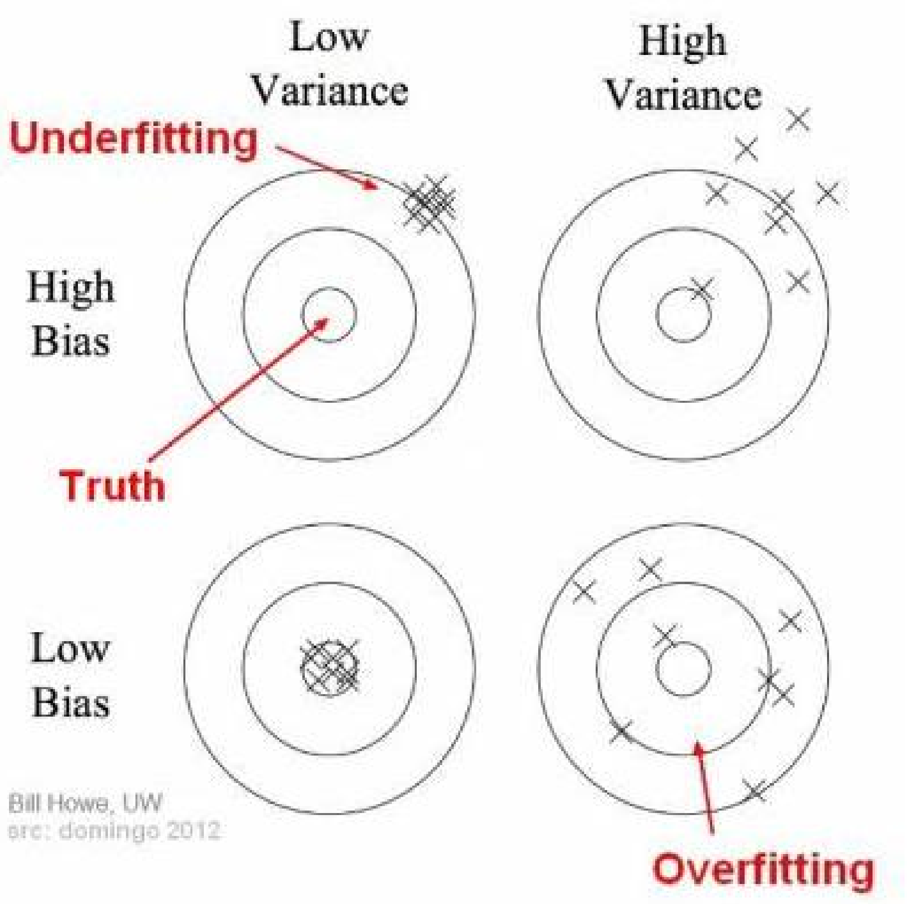
- combination between models increase accuracy and in machine learning combination is Ensembling

Errors

$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\hat{f}(x) - E[\hat{f}(x)]\right]^2 + \sigma_e^2$$

$$Err(x) = \mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

trade-off management of bias-variance errors



Ensemble Learning

- Ensemble learning is one way to execute this trade off analysis.

the wisdom of the crowd

- Suppose you ask a complex question to thousands of random people, then aggregate their answers. In many cases you will find that this aggregated answer is better than an expert's answer.

Basic Ensemble

- Max Voting
- Averaging
- Weighted Average

Advanced Ensemble

- Stacking
- Blending
- Bagging
- Boosting

Max Voting

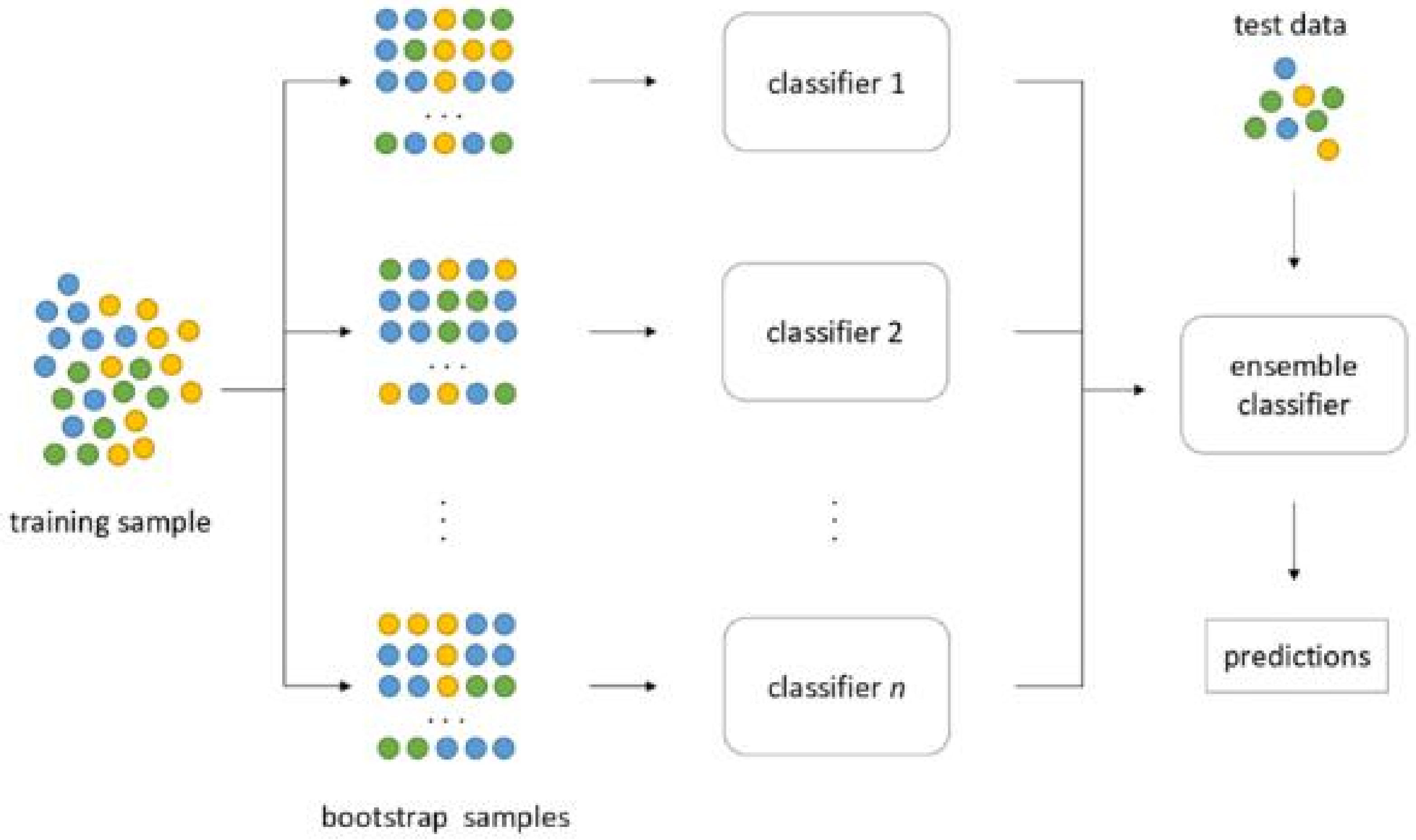
Averaging

Weighted Average

Bagging

- Bagging is very common in competitions.
- your data must have variance

Bagging


training sample
bootstrap samples
classifier 1
classifier 2
classifier n
test data
ensemble classifier
predictions

Random forest

- Random Forest is another ensemble machine learning algorithm that follows the bagging technique. It is an extension of the bagging estimator algorithm. The base estimators in random forest are decision trees. Unlike bagging meta estimator, random forest randomly selects a set of features which are used to decide the best split at each node of the decision tree.

Boosting

- The term ‘Boosting’ refers to a family of algorithms which converts weak learner to strong learners. Boosting is an ensemble method for improving the model predictions of any given learning algorithm. The idea of boosting is to train weak learners sequentially, each trying to correct its predecessor.

AdaBoost

- Adaptive boosting or AdaBoost is one of the simplest boosting algorithms. Usually, decision trees are used for modelling. Multiple sequential models are created, each correcting the errors from the last model. AdaBoost assigns weights to the observations which are incorrectly predicted and the subsequent model works to predict these values correctly.

stacking

- Stacking is a similar to boosting
- you also apply several models to your original data. The difference here is, however, that you don't have just an empirical formula for your weight function, rather you introduce a meta-level and use another model/approach to estimate the input together with outputs of every model to estimate the weights or, in other words, to determine what models perform well and what badly given these input data.

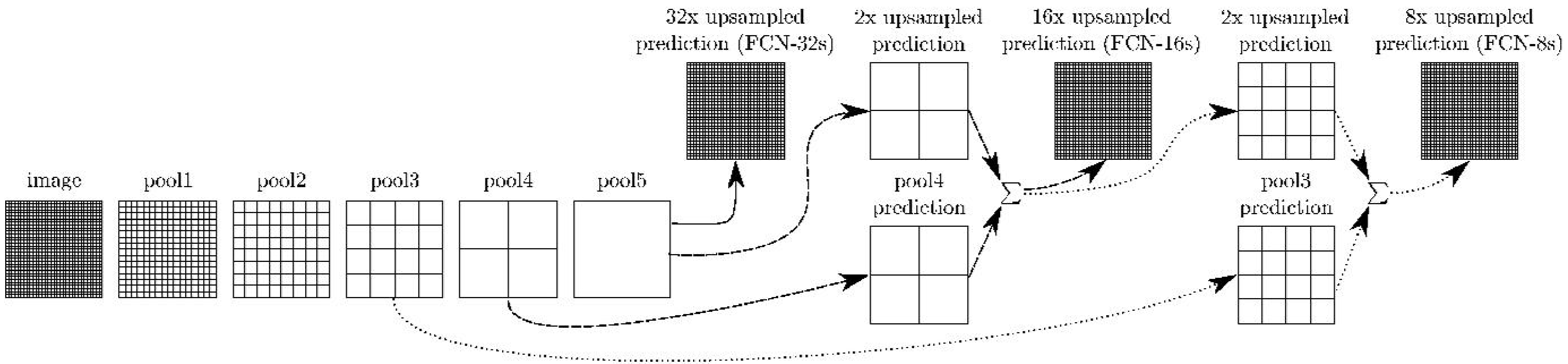
Note

Note

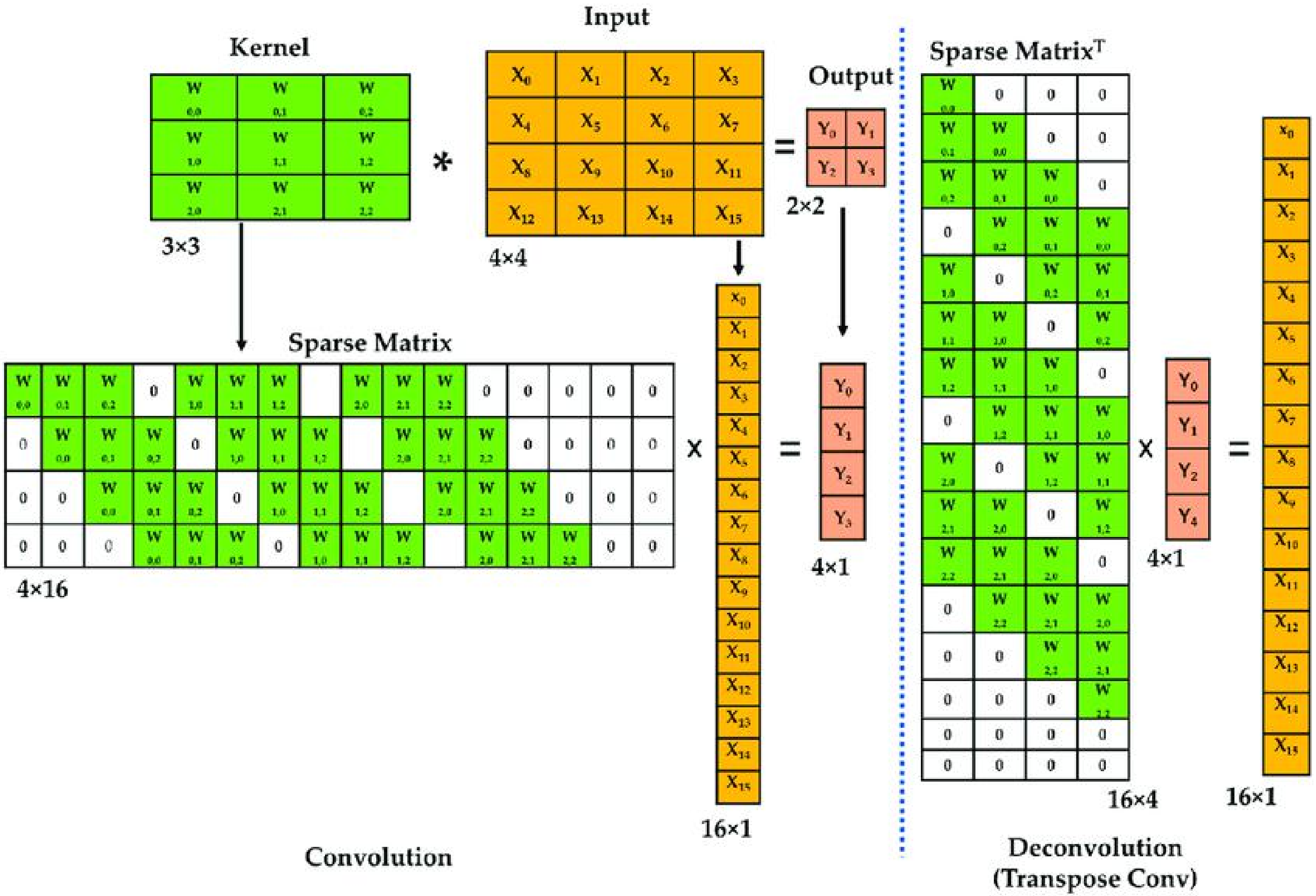
Note

后厂理工学院
houchangtech.com

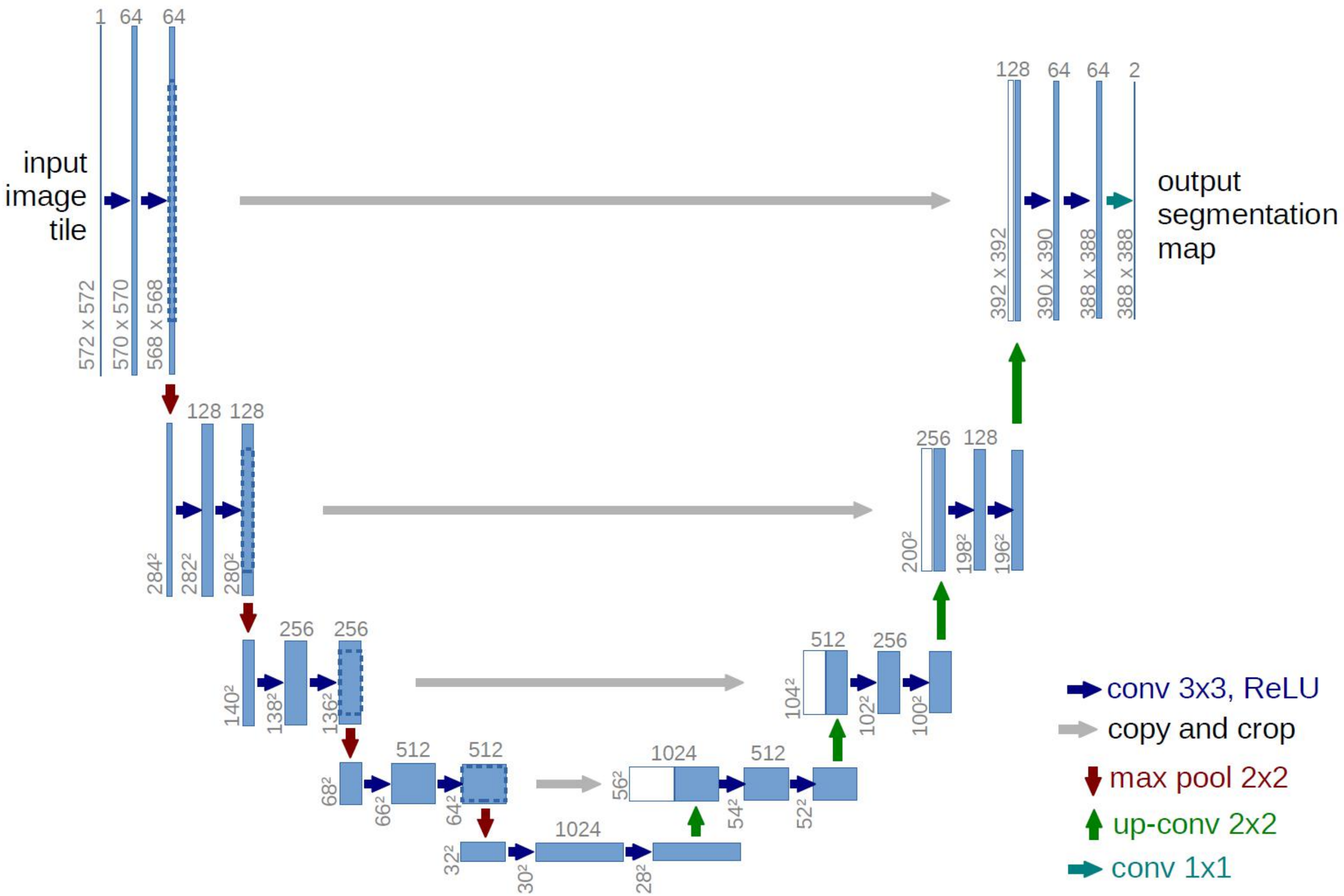
FCN



Transposed Conv

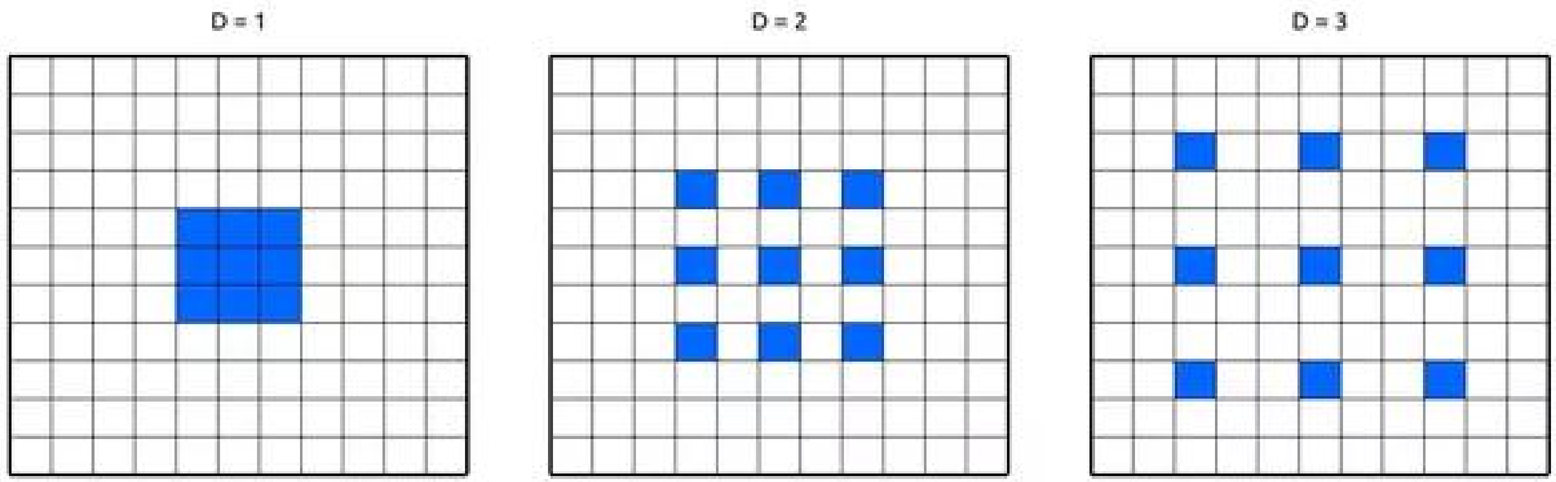


U-Net

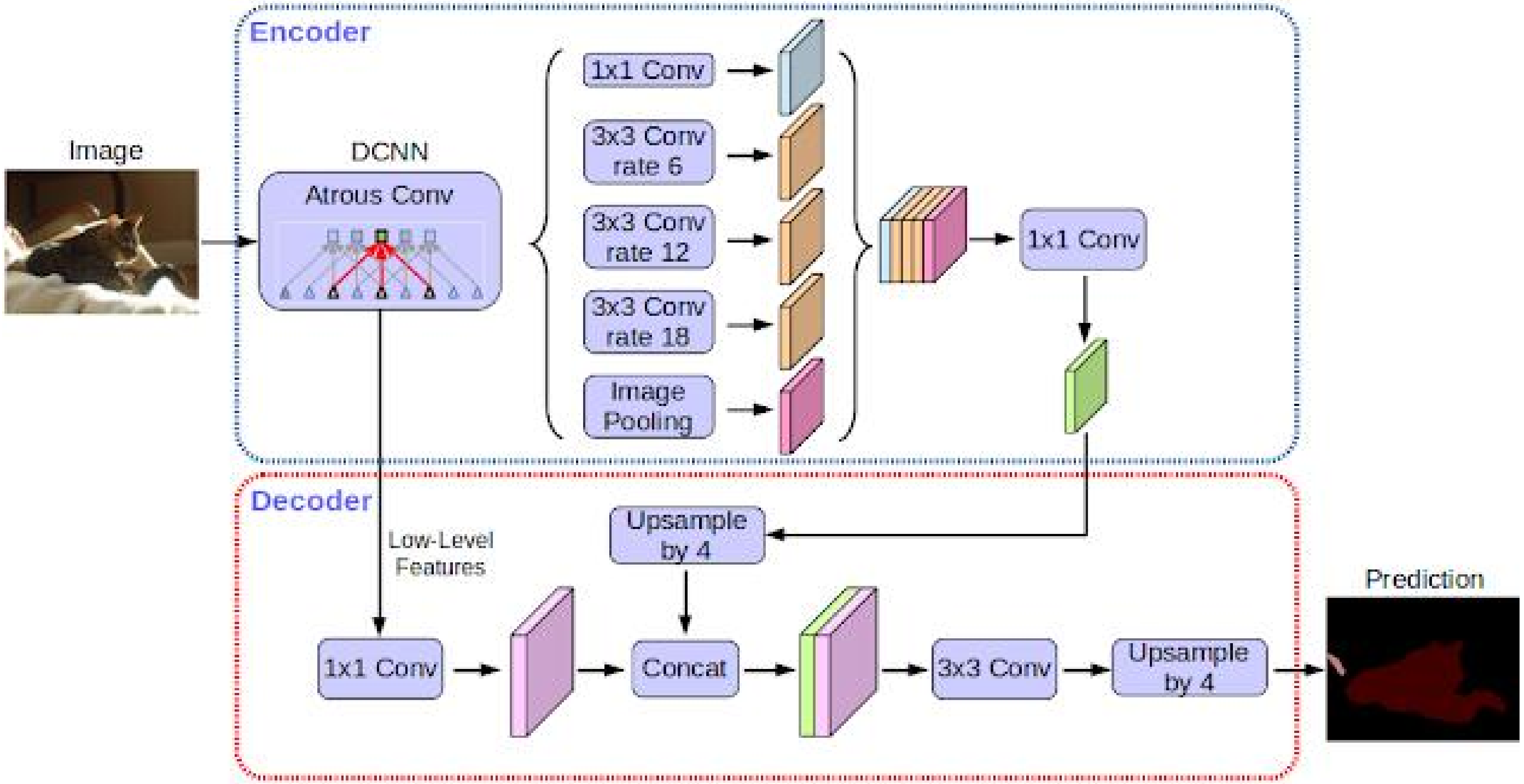


input image tile
output segmentation map
572 x 572
570 x 570
568 x 568
392 x 392
390 x 390
388 x 388
388 x 388
1
64
64
128
64
64
2
128
128
256
128
256
256
512
256
512
512
1024
512
1024
conv 3x3, ReLU
copy and crop
max pool 2x2
up-conv 2x2
conv 1x1

Dilated Conv



Deeplab V3+


Encoder
Image
DCNN
Atrous Conv
1x1 Conv
3x3 Conv rate 6
3x3 Conv rate 12
3x3 Conv rate 18
Image Pooling
1x1 Conv
Decoder
Low-Level Features
Upsample by 4
1x1 Conv
Concat
3x3 Conv
Upsample by 4
Prediction

课程总结

- 掌握Semantic Segmentation
- 掌握Lane Segmentation

重难点

- Transposed Conv
- Dilated Conv
- U-Net
- Deeplab v3+

重难点

- Data Generator
- Metrics
- Loss
- Training

课程作业

- 提交项目训练结果

后厂理工学院

houchangtech.com

一所专注前沿互联网技术领域的创新实战大学