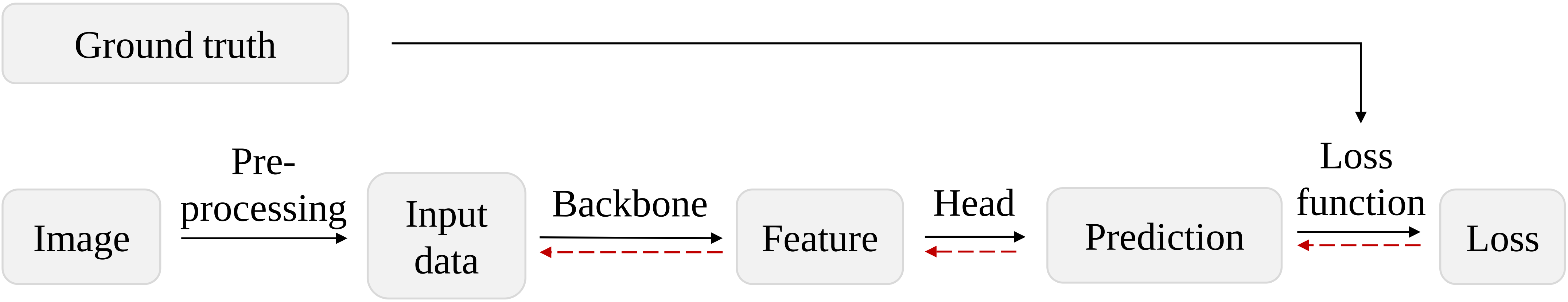
Automatic Loss Function Design for Generic Vision Tasks

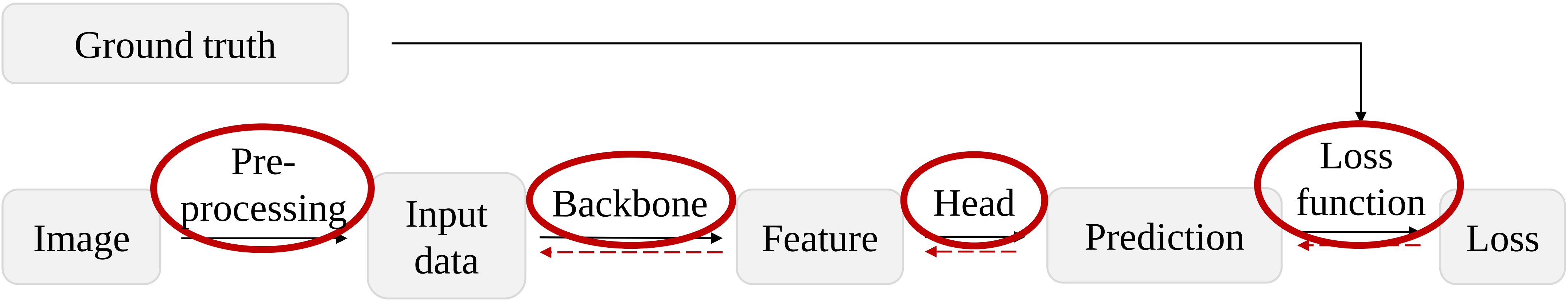
Jifeng Dai

SenseTime Research

General Deep Learning Training Framework

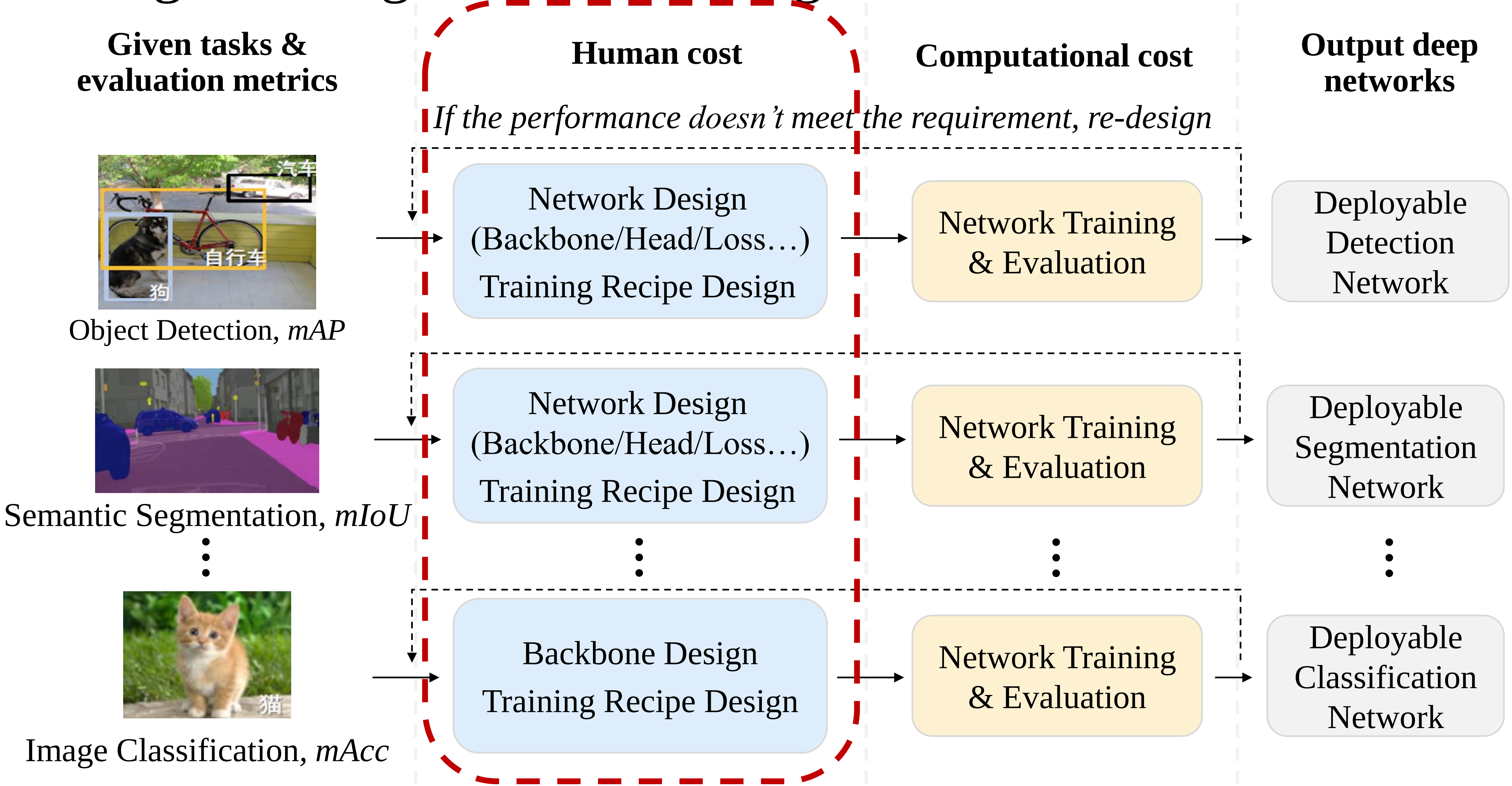


General Deep Learning Training Framework



- How to design the deep networks?
  - Handcrafting
  - AutoML

Design Paradigm - Handcrafting



Design Paradigm - AutoML

For image classification task only

Computational cost

Output deep networks

Image Classification, *mAcc*

Automated Pre-processing Design
Automated Backbone Design
Automated Training Recipe Design

Deployable Classification Network

- Current AutoML focuses on searching **for the image classification task only**.

Design Paradigm

- Handcrafting
  - Requires human expertise & insight
  - High human cost for trial and error
- AutoML
  - Current AutoML algorithms focus on automatedly searching for pre-processing strategies, backbone networks and training recipes for the image classification task.

Design Paradigm

- Handcrafting
  - Requires human expertise & insight
  - High human cost for trial and error
- AutoML
  - Current AutoML algorithms focus on automatedly searching for pre-processing strategies, backbone networks and training recipes for the image classification task.
- Fully automated design?

Design Paradigm – Fully Auto Design for Generic Tasks

Given tasks & evaluation metrics

Object Detection, *mAP*

Semantic Segmentation, *mIoU*

⋮

Image Classification, *mAcc*

Computational cost

Fully Automated Design of Deep Networks for **Generic Tasks** (Pre-processing/Backbone/Head/Loss/Training Recipe)

- Searching for **generic tasks and metrics**.

Output deep networks

Deployable Detection Network

Deployable Segmentation Network

⋮

Deployable Classification Network

Design Paradigm – Fully Auto Design for Generic Tasks

- Pre-processing
- Training Recipe
- Backbone
- Head
- Loss function

→ An unified search framework **for generic tasks**

Design Paradigm – Fully Auto Design for Generic Tasks

- Pre-processing
- Training Recipe
- Backbone
- Head
- Loss function

→ An unified search framework **for generic tasks**

Under-investigated!

Auto Seg-Loss: Searching Metric Surrogates for Semantic Segmentation

Hao Li[1,*], Chenxin Tao[2,*], Xizhou Zhu[3], Xiaogang Wang[1,3], Gao Huang[2], Jifeng Dai[3,4,‡]

[1]The Chinese University of Hong Kong, [2]Tsinghua University,
[3]Sensetime Research, [4]Qing Yuan Research Institute, Shanghai Jiao Tong University
*Equal contribution ‡Corresponding author

Published as a conference paper on ICLR 2021

Different Focuses of Different Metrics

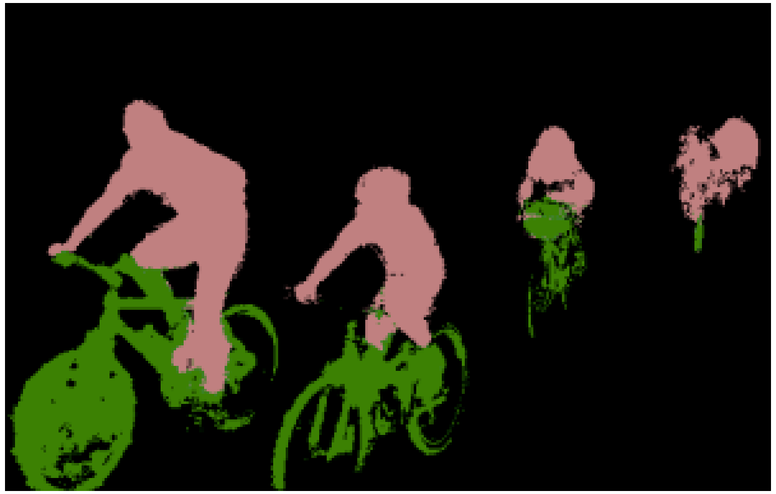
Image

Prediction #1

mIoU = 82%

BF1 = 40%

Prediction #2



mIoU = 83% **(+1%)**

BF1 = 75% **(+35%)**

mIoU: Mean Intersection over Union
BF1: Boundary F1 Score

Note: The tolerance radius of the BF1 score here is 5 pixels.

Different Focuses of Different Metrics

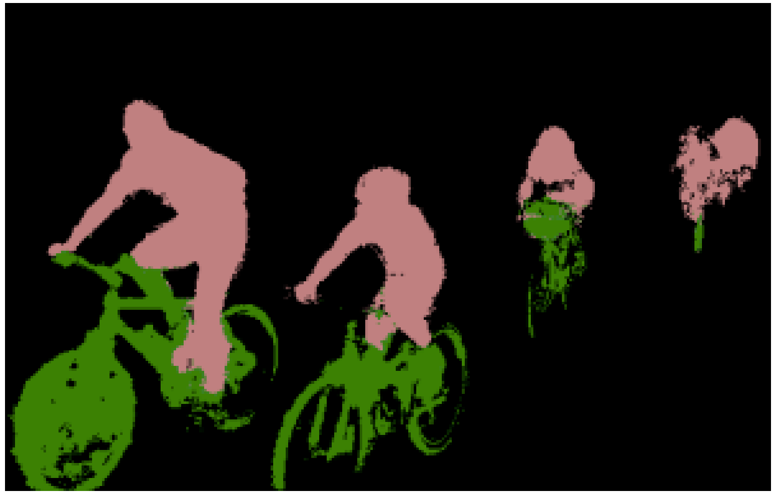
Image

Prediction #1

mIoU = 82%

BF1 = 40%

Prediction #2



mIoU = 85% **(+3%)**

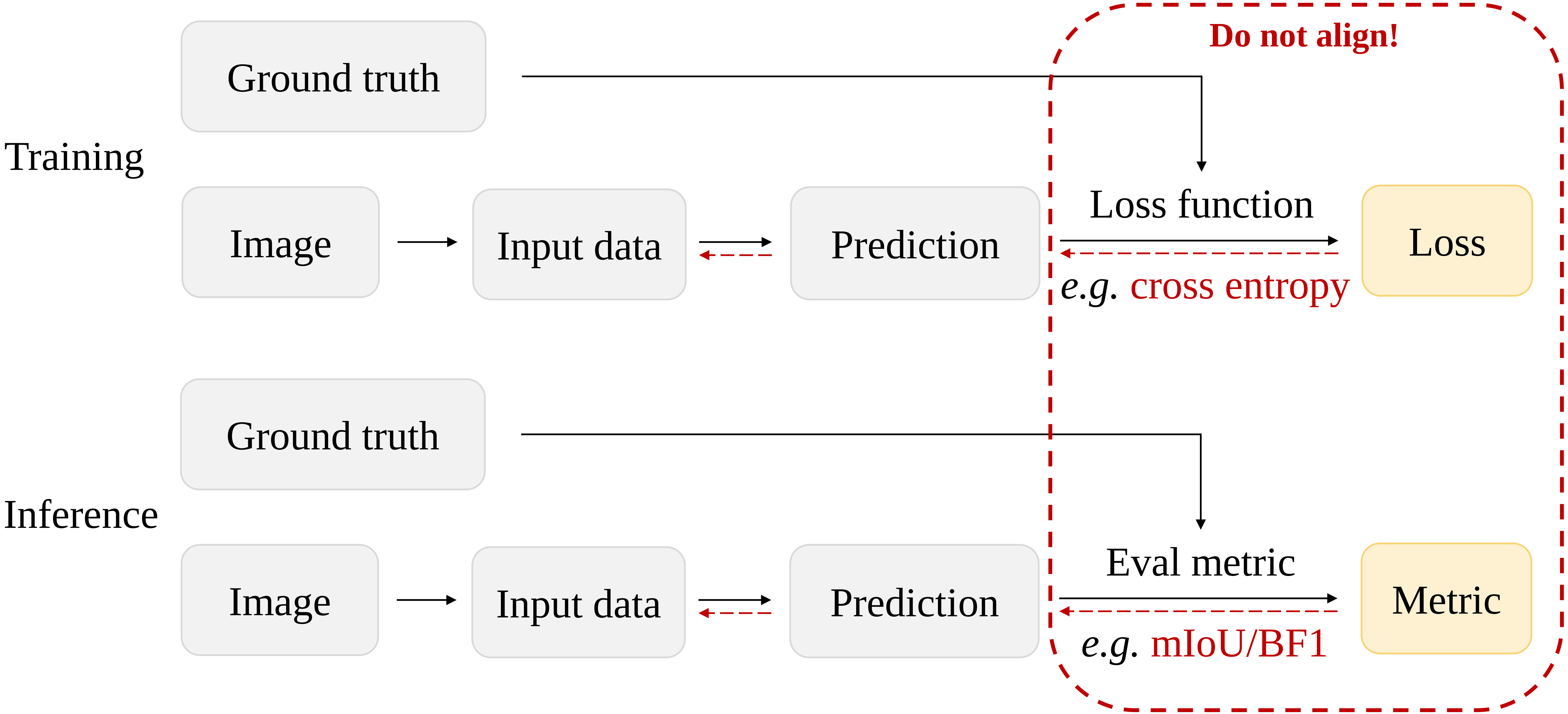
BF1 = 75% **(+35%)**

Some metrics (*e.g.* mIoU) measure the accuracy on the whole image, while others (*e.g.* BF1) focus more on the boundaries.

mIoU: Mean Intersection over Union
BF1: Boundary F1 Score

Note: The tolerance radius of the BF1 score here is 5 pixels.

Recalling General Image Recognition Framework



Designing Metric-Specific Loss Functions

Loss Functions

- Manually design
- Automatically search

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer, 2015.
[2] Francesco Caliva, Claudia Iriondo, Alejandro Morales Martinez, Sharmila Majumdar, and Valentina Pedoia. Distance map loss penalty term for semantic segmentation. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*, 2019.
[3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovasz-softmax loss: A tractable ´ surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4413– 4421, 2018.
[4] Chuming Li, Xin Yuan, Chen Lin, Minghao Guo, Wei Wu, Junjie Yan, and Wanli Ouyang. Amlfs: Automl for loss function search. In *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, pp. 8410–8419, 2019.

Designing Metric-Specific Loss Functions

Loss Functions

Manually design

High human cost

Cross entropy variants

e.g. WCE[1], DPCE[2]

Metric-based surrogates

e.g. Lovasz-Softmax[3]

Automatically search

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer, 2015.
[2] Francesco Caliva, Claudia Iriondo, Alejandro Morales Martinez, Sharmila Majumdar, and Valentina Pedoia. Distance map loss penalty term for semantic segmentation. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*, 2019.
[3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovasz-softmax loss: A tractable ´ surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4413– 4421, 2018.
[4] Chuming Li, Xin Yuan, Chen Lin, Minghao Guo, Wei Wu, Junjie Yan, and Wanli Ouyang. Amlfs: Automl for loss function search. In *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, pp. 8410–8419, 2019.

Designing Metric-Specific Loss Functions

Loss Functions

Manually design

High human cost

Cross entropy variants

e.g. WCE[1], DPCE[2]

Metric-based surrogates

e.g. Lovasz-Softmax[3]

Automatically search

Combination of losses

e.g. AM-LFS[4]

Limited to existing losses

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer, 2015.
[2] Francesco Caliva, Claudia Iriondo, Alejandro Morales Martinez, Sharmila Majumdar, and Valentina Pedoia. Distance map loss penalty term for semantic segmentation. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*, 2019.
[3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovasz-softmax loss: A tractable ´ surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4413– 4421, 2018.
[4] Chuming Li, Xin Yuan, Chen Lin, Minghao Guo, Wei Wu, Junjie Yan, and Wanli Ouyang. Amlfs: Automl for loss function search. In *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, pp. 8410–8419, 2019.

Designing Metric-Specific Loss Functions

Loss Functions

- Manually design
  - High human cost
  - Cross entropy variants
    - e.g. WCE[1], DPCE[2]
  - Metric-based surrogates
    - e.g. Lovasz-Softmax[3]
- Automatically search
  - Combination of losses
    - e.g. AM-LFS[4]
    - Limited to existing losses
  - Search surrogate losses for evaluation metrics

Idea: designing the search space based on the formulation of the target evaluation metric.

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241. Springer, 2015.
[2] Francesco Caliva, Claudia Iriondo, Alejandro Morales Martinez, Sharmila Majumdar, and Valentina Pedoia. Distance map loss penalty term for semantic segmentation. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*, 2019.
[3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovasz-softmax loss: A tractable ´ surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4413– 4421, 2018.
[4] Chuming Li, Xin Yuan, Chen Lin, Minghao Guo, Wei Wu, Junjie Yan, and Wanli Ouyang. Amlfs: Automl for loss function search. In *Proceedings of the IEEE International Conference on Computer Vision (CVPR)*, pp. 8410–8419, 2019.

Revisiting Metrics for Semantic Segmentation

- Non-differentiable

Type	Name	Formula
Acc-based	Global Accuracy†	$\mathrm{gAcc} = \frac{\sum_{n,c,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,c,h,w} y_{n,c,h,w}}$ (1)
	Mean Accuracy†	$\mathrm{mAcc} = \frac{1}{C}\sum_c \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} y_{n,c,h,w}}$ (2)
IoU-based	Mean IoU†	$\mathrm{mIoU} = \frac{1}{C}\sum_c \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}}$ (3)
	Frequency Weighted IoU†	$\mathrm{FWIoU} = \sum_c \frac{\sum_{n,h,w} y_{n,c,h,w}}{\sum_{n,c',h,w} y_{n,c',h,w}} \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}}$ (4)
	Boundary IoU*	$\mathrm{BIoU} = \frac{1}{C}\sum_c \frac{\sum_n \sum_{h,w \in \mathrm{BD}(y_n)} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_n \sum_{h,w \in \mathrm{BD}(y_n)} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}}$ where $\mathrm{BD}(y) = y \text{ XOR } \text{Min-Pooling}(y)$ (5)
F1-score-based	Boundary F1 Score*	$\text{BF1-score} = \frac{1}{C}\sum_c \frac{2 \times \mathrm{prec}_c \times \mathrm{recall}_c}{(\mathrm{prec}_c + \mathrm{recall}_c)}$ where $\mathrm{prec}_c = \frac{\sum_{n,h,w} \mathrm{BD}(\hat{y}_n)_{c,h,w} \text{ AND Max-Pooling}(\mathrm{BD}(y_n)_{c,h,w})}{\sum_{n,h,w} \mathrm{BD}(\hat{y}_n)_{c,h,w}}$, $\mathrm{recall}_c = \frac{\sum_{n,h,w} \text{Max-Pooling}(\mathrm{BD}(\hat{y}_n)_{c,h,w}) \text{ AND}(\mathrm{BD}(y_n)_{c,h,w})}{\sum_{n,h,w} \mathrm{BD}(y_n)_{c,h,w}}$ (6)

Revisiting Metrics for Semantic Segmentation

- Non-differentiable
  - One-hot operation (Argmax)
  - Logical operation (AND/OR/XOR)
- Extend these operations to the continuous domain!

Type	Name	Formula
Acc-based	Global Accuracy†	$$\text{gAcc} = \frac{\sum_{n,c,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,c,h,w} y_{n,c,h,w}} \quad (1)$$
	Mean Accuracy†	$$\text{mAcc} = \frac{1}{C}\sum_c \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} y_{n,c,h,w}} \quad (2)$$
IoU-based	Mean IoU†	$$\text{mIoU} = \frac{1}{C}\sum_c \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}} \quad (3)$$
	Frequency Weighted IoU†	$$\text{FWIoU} = \sum_c \frac{\sum_{n,h,w} y_{n,c,h,w}}{\sum_{n,c',h,w} y_{n,c',h,w}} \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}} \quad (4)$$
	Boundary IoU*	$$\text{BIoU} = \frac{1}{C}\sum_c \frac{\sum_n \sum_{h,w\in \text{BD}(y_n)} \hat{y}_{n,c,h,w} \text{ AND } y_{n,c,h,w}}{\sum_n \sum_{h,w\in \text{BD}(y_n)} \hat{y}_{n,c,h,w} \text{ OR } y_{n,c,h,w}} \quad (5)$$ where $\text{BD}(y) = y \text{ XOR Min-Pooling}(y)$
F1-score-based	Boundary F1 Score*	$$\text{BF1-score} = \frac{1}{C}\sum_c \frac{2 \times \text{prec}_c \times \text{recall}_c}{(\text{prec}_c + \text{recall}_c)}$$ where $\text{prec}_c = \frac{\sum_{n,h,w} \text{BD}(\hat{y}_n)_{c,h,w} \text{ AND Max-Pooling}(\text{BD}(y_n)_{c,h,w})}{\sum_{n,h,w} \text{BD}(\hat{y}_n)_{c,h,w}}$, $\text{recall}_c = \frac{\sum_{n,h,w} \text{Max-Pooling}(\text{BD}(\hat{y}_n)_{c,h,w}) \text{ AND}(\text{BD}(y_n)_{c,h,w})}{\sum_{n,h,w} \text{BD}(y_n)_{c,h,w}}$ (6)

Extending Metrics to Surrogates

$$\text{mIoU} = \frac{1}{C}\sum_{c} \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } \sum_{n,h,w} y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } \sum_{n,h,w} y_{n,c,h,w}}$$

- One-hot operation

Network Outputs

$$\hat{y}_{n,c,h,w} = \text{argmax}_c(z_{n,c,h,w})$$

$$\hat{y}_{n,c,h,w} \approx \tilde{y}_{n,c,h,w} = \text{softmax}_c(z_{n,c,h,w})$$

Extending Metrics to Surrogates

$$\text{mIoU} = \frac{1}{C}\sum_{c} \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } \sum_{n,h,w} y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } \sum_{n,h,w} y_{n,c,h,w}}$$

- One-hot operation
- Logical operation

AND

	$y_2 = 0$	$y_2 = 1$
$y_1 = 0$	0	0
$y_1 = 1$	0	1

$\longrightarrow$ $f_{AND}(y_1, y_2) = y_1 y_2$

OR

	$y_2 = 0$	$y_2 = 1$
$y_1 = 0$	0	1
$y_1 = 1$	1	1

$\longrightarrow$ $f_{OR}(y_1, y_2) = y_1 + y_2 - y_1 y_2$

$(y_1, y_2 \in [0, 1])$

Extending Metrics to Surrogates

$$\mathrm{mIoU} = \frac{1}{C}\sum_{c} \frac{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ AND } \sum_{n,h,w} y_{n,c,h,w}}{\sum_{n,h,w} \hat{y}_{n,c,h,w} \text{ OR } \sum_{n,h,w} y_{n,c,h,w}}$$

- One-hot operation
- Logical operation

No guarantee to drive the training well!

AND

	$y_2 = 0$	$y_2 = 1$
$y_1 = 0$	0	0
$y_1 = 1$	0	1

$\longrightarrow$ $f_{AND}(y_1, y_2) = y_1 y_2$

OR

	$y_2 = 0$	$y_2 = 1$
$y_1 = 0$	0	1
$y_1 = 1$	1	1

$\longrightarrow$ $f_{OR}(y_1, y_2) = y_1 + y_2 - y_1 y_2$

$(y_1, y_2 \in [0, 1])$

Extending Metrics to Surrogates

$$\text{mIoU} = \frac{1}{C}\sum_{c}\frac{\sum_{n,h,w}\hat{y}_{n,c,h,w}\ \text{AND}\ \sum_{n,h,w}y_{n,c,h,w}}{\sum_{n,h,w}\hat{y}_{n,c,h,w}\ \text{OR}\ \sum_{n,h,w}y_{n,c,h,w}}$$

- One-hot operation
- Logical operation

$$f_{AND}(y_1, y_2) = y_1 y_2$$

$$\Big\downarrow\ g(y;\theta): [0,1] \to \mathbb{R}$$

$$h_{AND}(y_1, y_2; \theta_{AND}) = g(y_1; \theta_{AND})g(y_2; \theta_{AND})$$

Extending Metrics to Surrogates

$$\text{mIoU} = \frac{1}{C}\sum_{c}\frac{\sum_{n,h,w}\hat{y}_{n,c,h,w}\ \text{AND}\sum_{n,h,w}y_{n,c,h,w}}{\sum_{n,h,w}\hat{y}_{n,c,h,w}\ \text{OR}\sum_{n,h,w}y_{n,c,h,w}}$$

- One-hot operation
- Logical operation

Add constraints:

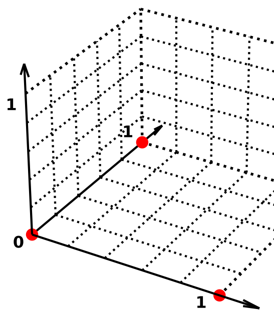
Truth table constraint: $g(0;\theta) = 0, g(1,\theta) = 1$

Monotonicity constraint: $\frac{\partial h}{\partial y_i} \geq 0$

Extending Metrics to Surrogates

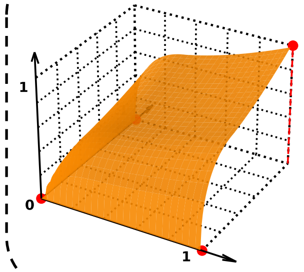
- Extend one-hot operation $\hat{y}_{n,c,h,w} \approx \tilde{y}_{n,c,h,w} = \mathrm{softmax}_c(z_{n,c,h,w})$
- Extend logical operation
$$f_{AND}(y_1, y_2) = y_1 y_2$$
$$f_{OR}(y_1, y_2) = y_1 + y_2 - y_1 y_2$$
- Parameterize logical operation
$$h_{AND}(y_1, y_2; \theta_{AND}) = g(y_1; \theta_{AND}) g(y_2; \theta_{AND})$$
$$h_{OR}(y_1, y_2; \theta_{OR}) = g(y_1; \theta_{OR}) + g(y_2; \theta_{OR}) - g(y_1; \theta_{OR}) g(y_2; \theta_{OR})$$
- Add constraints
Truth table constraint: $g(0; \theta) = 0, g(1, \theta) = 1$
Monotonicity constraint: $\frac{\partial h}{\partial y_i} \geq 0$

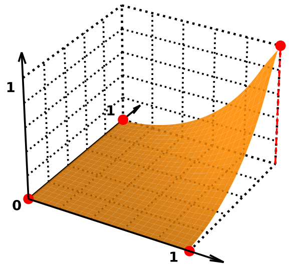
AND



$\xrightarrow{\text{interpolate}}$

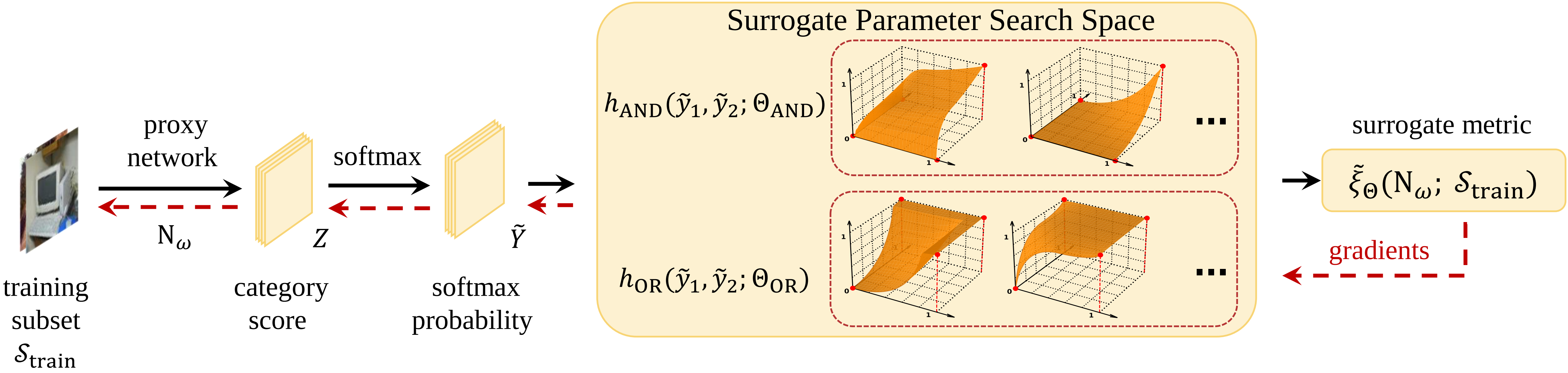
$\boldsymbol{h_{AND}}$





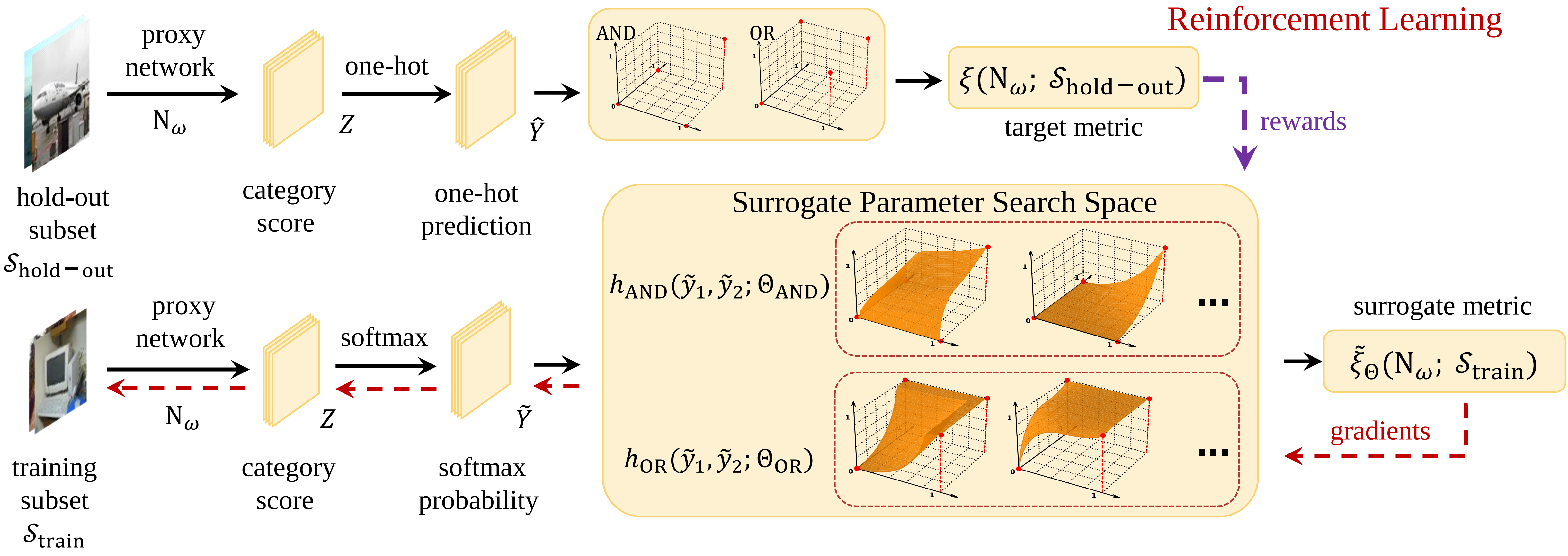
...

Auto Seg-Loss Search Process - *Sampling*

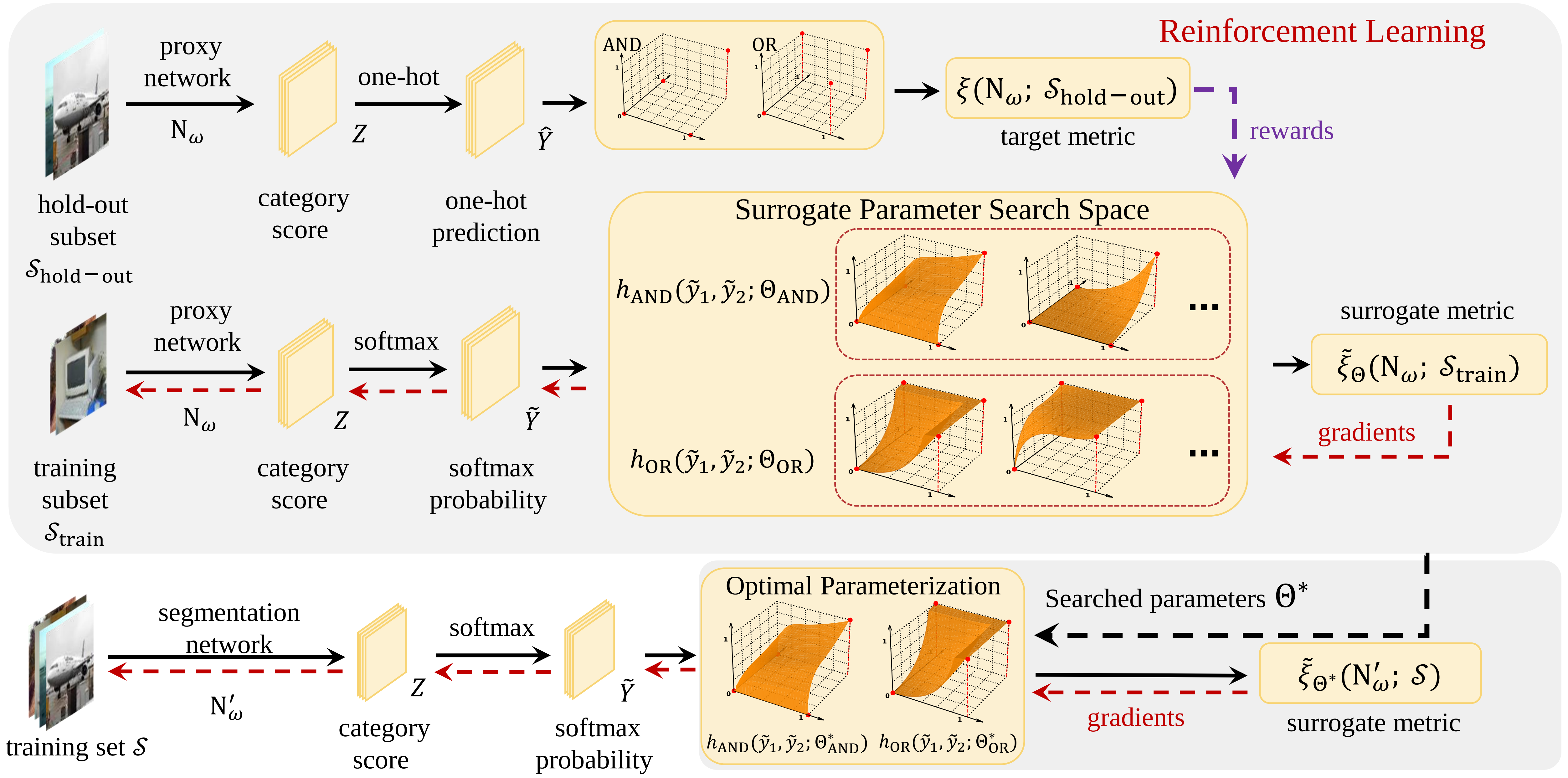


(Θ_{AND} and Θ_{OR} are sampled from truncated gaussian distributions)

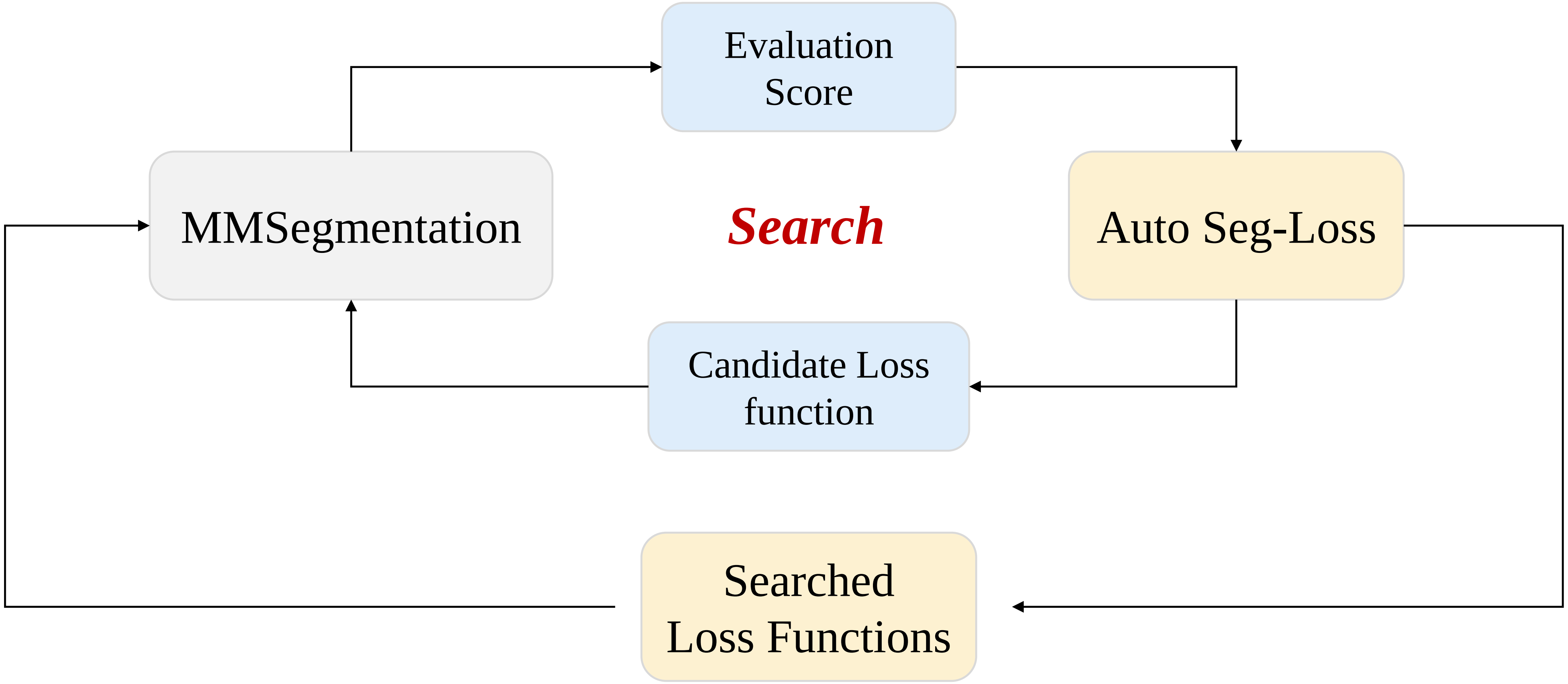
Auto Seg-Loss Search Process - *Updating*



Auto Seg-Loss Search Process – *Re-training*



MMSegmentation



Experiments on Semantic Segmentation

Dataset	PASCAL VOC						Cityscapes					
Loss Function	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**
Cross Entropy	78.69	91.31	70.61	65.30	87.31	95.17	79.97	**93.33**	62.07	62.24	87.01	**96.44**
WCE (Ronneberger et al., 2015)	69.60	85.64	61.80	37.59	**92.61**	91.11	73.01	90.51	53.07	51.19	**89.22**	94.56
DPCE (Caliva et al., 2019)	79.82	91.76	71.87	66.54	87.76	**95.45**	80.27	**93.38**	62.57	65.99	86.99	**96.46**
SSIM (Qin et al., 2019)	79.26	91.68	71.54	66.35	87.87	95.38	**80.65**	**93.22**	63.04	72.20	86.88	**96.39**
DiceLoss (Milletari et al., 2016)	77.78	91.34	69.85	64.38	87.47	95.11	79.30	**93.25**	60.93	59.94	86.38	**96.39**
Lovàsz (Berman et al., 2018)	79.72	91.78	72.47	66.65	88.64	95.42	77.67	92.51	56.71	53.48	82.05	96.03
Searched mIoU	**80.97**	**92.09**	73.44	68.86	88.23	**95.68**	**80.67**	**93.30**	63.05	67.97	87.20	**96.44**
Searched FWIoU	80.00	**91.93**	75.14	65.67	89.23	95.44	79.42	**93.33**	61.71	59.68	87.96	**96.37**
Searched BIoU	48.97	69.89	**79.27**	38.99	81.28	62.64	45.89	39.80	**63.89**	38.29	62.80	58.15
Searched BF1	1.93	0.96	7.39	**74.83**	6.51	2.66	6.78	3.19	18.37	**77.40**	12.09	8.19
Searched mAcc	69.80	85.86	72.85	35.62	**92.66**	91.28	74.10	90.79	54.62	53.45	**89.22**	94.75
Searched gAcc	79.73	91.76	74.09	64.41	88.95	**95.47**	79.41	**93.30**	61.65	62.04	87.08	**96.51**
Searched mIoU + BIoU	**81.19**	**92.19**	76.89	69.56	88.36	**95.75**	**80.43**	**93.34**	**63.88**	65.87	87.03	**96.45**
Searched mIoU + BF1	78.72	90.80	71.81	73.57	86.70	94.88	78.30	93.00	61.62	71.73	87.13	**96.23**

Our searched losses are on par or better than the previous losses on their target eval metrics!

Experiments on Semantic Segmentation

Dataset	PASCAL VOC						Cityscapes					
Loss Function	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**
Cross Entropy	78.69	91.31	70.61	65.30	87.31	95.17	79.97	**93.33**	62.07	62.24	87.01	**96.44**
WCE (Ronneberger et al., 2015)	69.60	85.64	61.80	37.59	**92.61**	91.11	73.01	90.51	53.07	51.19	**89.22**	94.56
DPCE (Caliva et al., 2019)	79.82	91.76	71.87	66.54	87.76	**95.45**	80.27	**93.38**	62.57	65.99	86.99	**96.46**
SSIM (Qin et al., 2019)	79.26	91.68	71.54	66.35	87.87	95.38	**80.65**	**93.22**	63.04	72.20	86.88	**96.39**
DiceLoss (Milletari et al., 2016)	77.78	91.34	69.85	64.38	87.47	95.11	79.30	**93.25**	60.93	59.94	86.38	**96.39**
Lovàsz (Berman et al., 2018)	79.72	91.78	72.47	66.65	88.64	95.42	77.67	92.51	56.71	53.48	82.05	96.03
Searched mIoU	**80.97**	**92.09**	73.44	68.86	88.23	**95.68**	**80.67**	**93.30**	63.05	67.97	87.20	**96.44**
Searched FWIoU	80.00	**91.93**	75.14	65.67	89.23	95.44	79.42	**93.33**	61.71	59.68	87.96	**96.37**
Searched BIoU	48.97	69.89	**79.27**	38.99	81.28	62.64	45.89	39.80	**63.89**	38.29	62.80	58.15
Searched BF1	1.93	0.96	7.39	**74.83**	6.51	2.66	6.78	3.19	18.37	**77.40**	12.09	8.19
Searched mAcc	69.80	85.86	72.85	35.62	**92.66**	91.28	74.10	90.79	54.62	53.45	**89.22**	94.75
Searched gAcc	79.73	91.76	74.09	64.41	88.95	**95.47**	79.41	**93.30**	61.65	62.04	87.08	**96.51**
Searched mIoU + BIoU	**81.19**	**92.19**	76.89	69.56	88.36	**95.75**	**80.43**	**93.34**	**63.88**	65.87	87.03	**96.45**
Searched mIoU + BF1	78.72	90.80	71.81	73.57	86.70	94.88	78.30	93.00	61.62	71.73	87.13	**96.23**

The searched losses show a huge improvement on the boundary metrics.

Generalization of the Searched Losses

Search once, use everywhere.

- Generalization among datasets
  - Searching on dataset A and training networks on dataset B

Datasets	Cityscapes ⟶ VOC						VOC ⟶ Cityscapes					
Loss Function	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**	**mIoU**	**FWIoU**	**BIoU**	**BF1**	**mAcc**	**gAcc**
Cross Entropy	78.69	91.31	70.61	65.30	87.31	**95.17**	79.97	**93.33**	62.07	62.24	87.01	**96.44**
Searched mIoU	**80.05**	**91.72**	**73.97**	67.61	88.01	**95.45**	**80.67**	**93.31**	**62.96**	66.48	87.36	**96.44**
Searched BF1	1.84	0.93	7.42	**75.85**	6.48	1.47	6.67	3.20	19.00	**77.99**	12.12	4.09
Searched mAcc	70.90	86.29	73.43	37.18	**93.19**	91.43	73.50	90.68	54.34	54.04	**88.66**	94.68

- Generalization among networks
  - Searching with a network and training networks with different structures

Network	R50-DeepLabv3+			R101-DeepLabv3+			R101-PSPNet			HRNetV2p-W48		
Loss Function	**mIoU**	**BF1**	**mAcc**	**mIoU**	**BF1**	**mAcc**	**mIoU**	**BF1**	**mAcc**	**mIoU**	**BF1**	**mAcc**
Cross Entropy	76.22	61.75	85.43	78.69	65.30	87.31	77.91	64.70	85.71	76.35	61.19	85.12
Searched mIoU	**78.35**	66.93	85.53	**80.97**	68.86	88.23	**78.93**	65.65	87.42	**77.26**	63.52	86.80
Searched BF1	1.35	**70.81**	6.05	1.43	**73.54**	6.12	1.62	**71.84**	6.33	1.34	**68.41**	5.99
Searched mAcc	69.82	36.92	**91.61**	69.80	35.62	**92.66**	71.66	39.44	**92.06**	68.22	35.90	**91.46**

Take-away

- Auto Seg-Loss is the first general framework for searching surrogate losses for mainstream semantic segmentation metrics.

- We propose an effective parameter regularization and search algorithm, which can find loss surrogates optimizing the target metric performance with mild computational overhead.

- The searched losses can generalize well to other datasets and networks. They also show superior performance over other loss functions.

Rethinking Auto Seg-Loss

- Auto Seg-Loss focuses on the task of semantic segmentation, and can be applied to more tasks by parameterizing the evaluation metrics.

- Challenge:

  Some evaluation metrics and operations are hard to parameterize.

  e.g. ranking & matching operations in mAP.

Rethinking Auto Seg-Loss

- Auto Seg-Loss focuses on the task of semantic segmentation, and can be applied to more tasks by parameterizing the evaluation metrics.

- Challenge:

  Some evaluation metrics and operations are hard to parameterize.

  e.g. ranking & matching operations in mAP.

- Can we search loss functions for more general tasks?

AutoLoss-Zero:
Searching Loss Functions from Scratch for Generic Tasks

Hao Li[1,*], Tianwen Fu[2,*], Jifeng Dai[3,5], Hongsheng Li[1], Gao Huang[4], Xizhou Zhu[3,‡]

[1]CUHK-SenseTime Joint Laboratory, The Chinese University of Hong Kong
[2]Department of Information Engineering, The Chinese University of Hong Kong
[3]Sensetime Research [4]Tsinghua University
[5]Qing Yuan Research Institute, Shanghai Jiao Tong University
*Equal contribution ‡Corresponding author

AutoLoss-Zero

- Zero – from scratch, minimal human expertise
- Target:

  A general framework for searching loss functions for any given tasks and evaluation metrics.

AutoLoss-Zero

- Zero – from scratch, minimal human expertise
- Target:

  A general framework for searching loss functions for any given tasks and evaluation metrics.
- Challenge:

  Heterogeneity of various tasks and evaluation metrics.

  → **Search space:** Basic primitive operators

  → **Search algorithm:** Efficient enough to explore the huge space; No task-specific heuristics

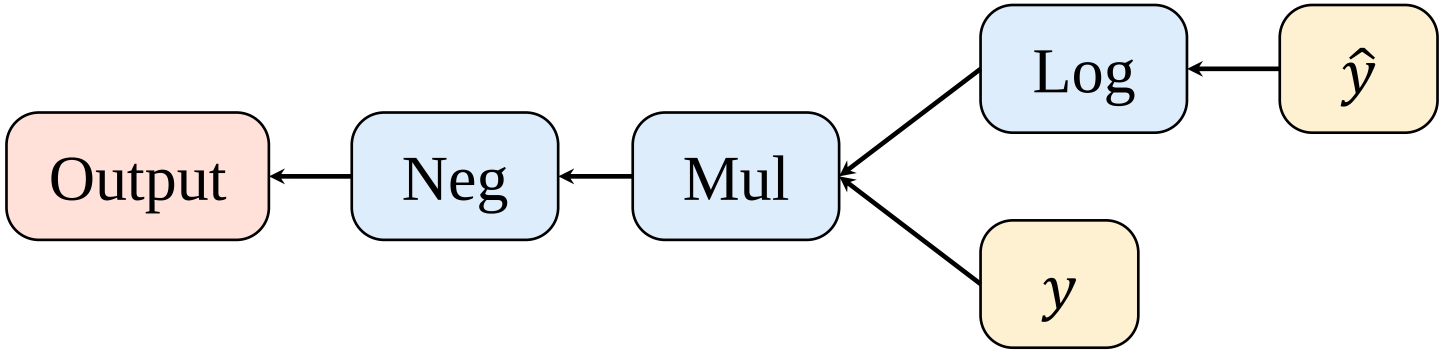
Search Space

- Primitive mathematical operators to construct the loss functions
- All the tasks share the same operator list.

Element-wise Operator	Expression	Arity
Add	$x + y$	2
Mul	$x \times y$	2
Neg	$-x$	1
Abs	$\|x\|$	1
Inv	$1/(x+\epsilon)$	1
Log	$\mathrm{sign}(x) \cdot \log(\|x\| + \epsilon)$	1
Exp	e^x	1
Tanh	$\tanh(x)$	1
Square	x^2	1
Sqrt	$\mathrm{sign}(x) \cdot \sqrt{\|x\| + \epsilon}$	1
†Aggregation Operator	**Expression**	**Arity**
Mean_{nhw}	$\frac{1}{NHW} \sum_{nhw} x_{nchw}$	1
Mean_c	$\frac{1}{C} \sum_c x_{nchw}$	1
$\text{Max-Pooling}_{3\times3}$	$\text{Max-Pooling}_{3\times3}(x)$	1
$\text{Min-Pooling}_{3\times3}$	$\text{Min-Pooling}_{3\times3}(x)$	1

Search Space

- Primitive mathematical operators to construct the loss functions
- Computational graphs to represent the loss functions

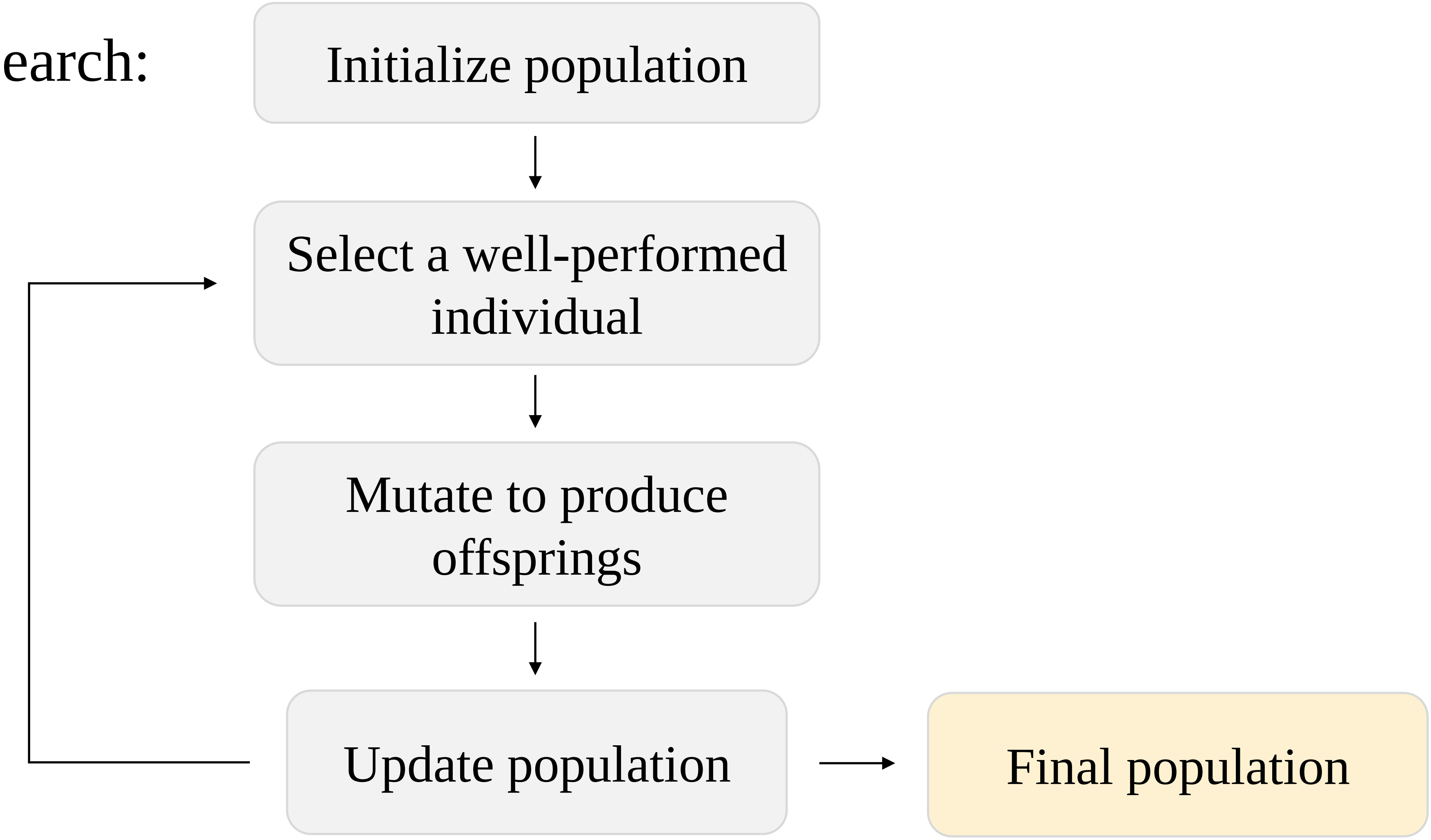


Cross Entropy (CE): $Output = -y * \log(\hat{y})$

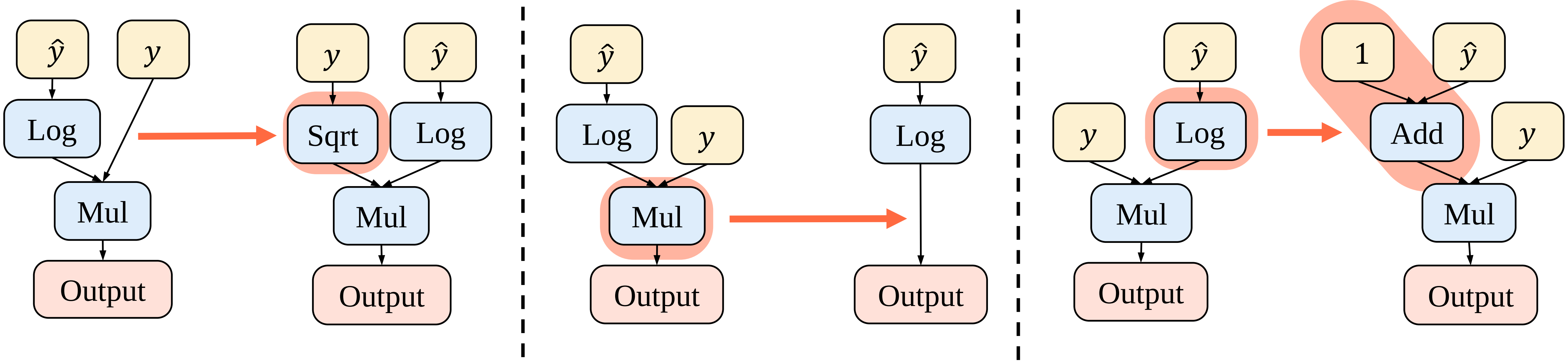
$$L = {}^{1}/_{NHW} * \sum_{n,c,h,w} Output_{n,c,h,w}$$

Search Algorithm

- Evolutionary search:



Search Algorithm - Mutation



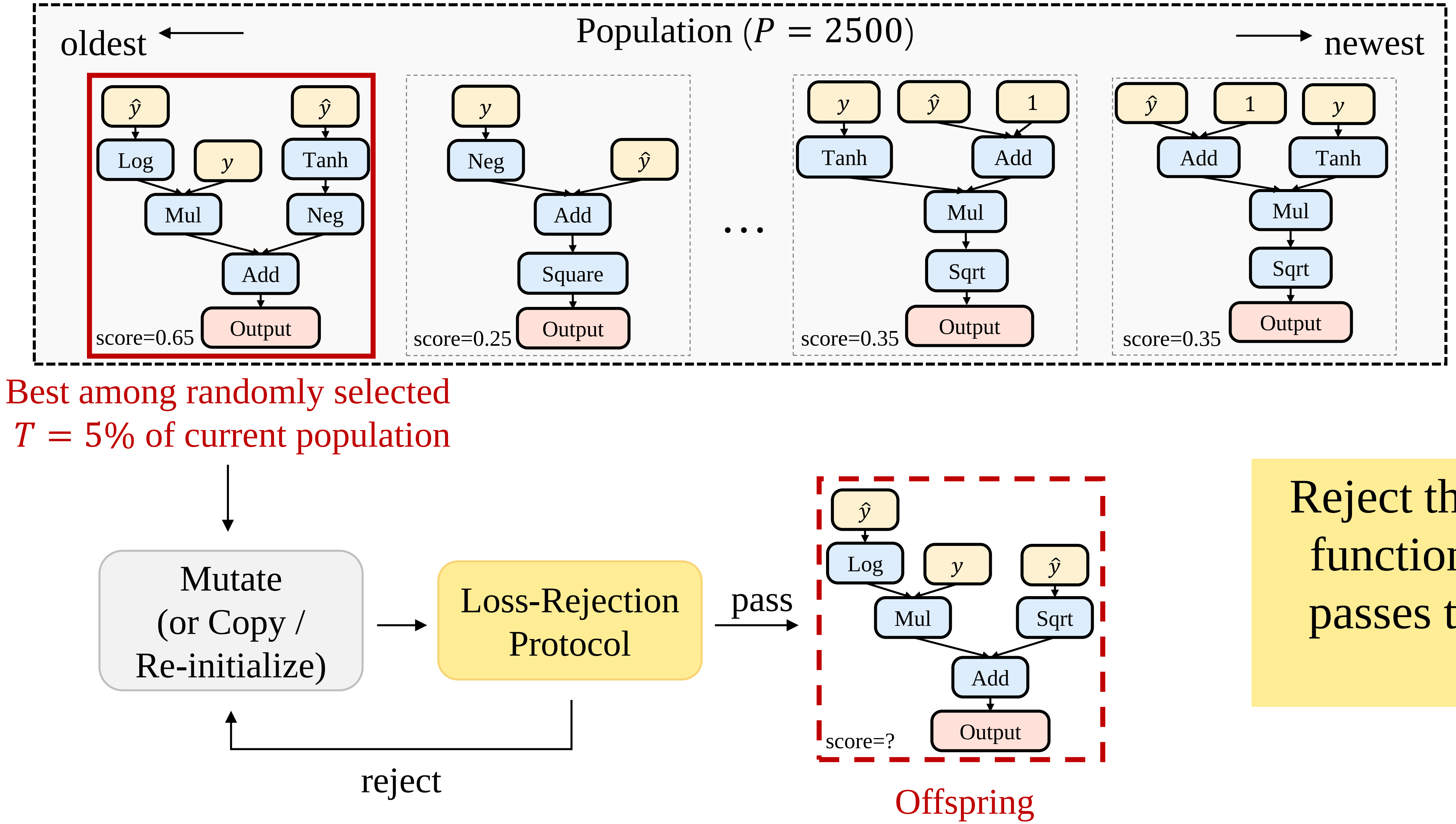
Insertion Deletion Replacement

Search Efficiency

Naïve evolution requires expensive network training
in evaluating the loss functions.

	Speed-up	# Explored Losses in 48h
Naïve Evolution	$1\times$	~ 300
+ Loss-Rejection Protocol	$\sim 700\times$	$\sim 2.1\times 10^5$
+ Gradient-Equivalence-Check Strategy	$\sim 1000\times$	$\sim 3.2\times 10^5$
+ Stop Training for Invalid Loss Values	$\sim 5000\times$	$\sim 1.5\times 10^6$

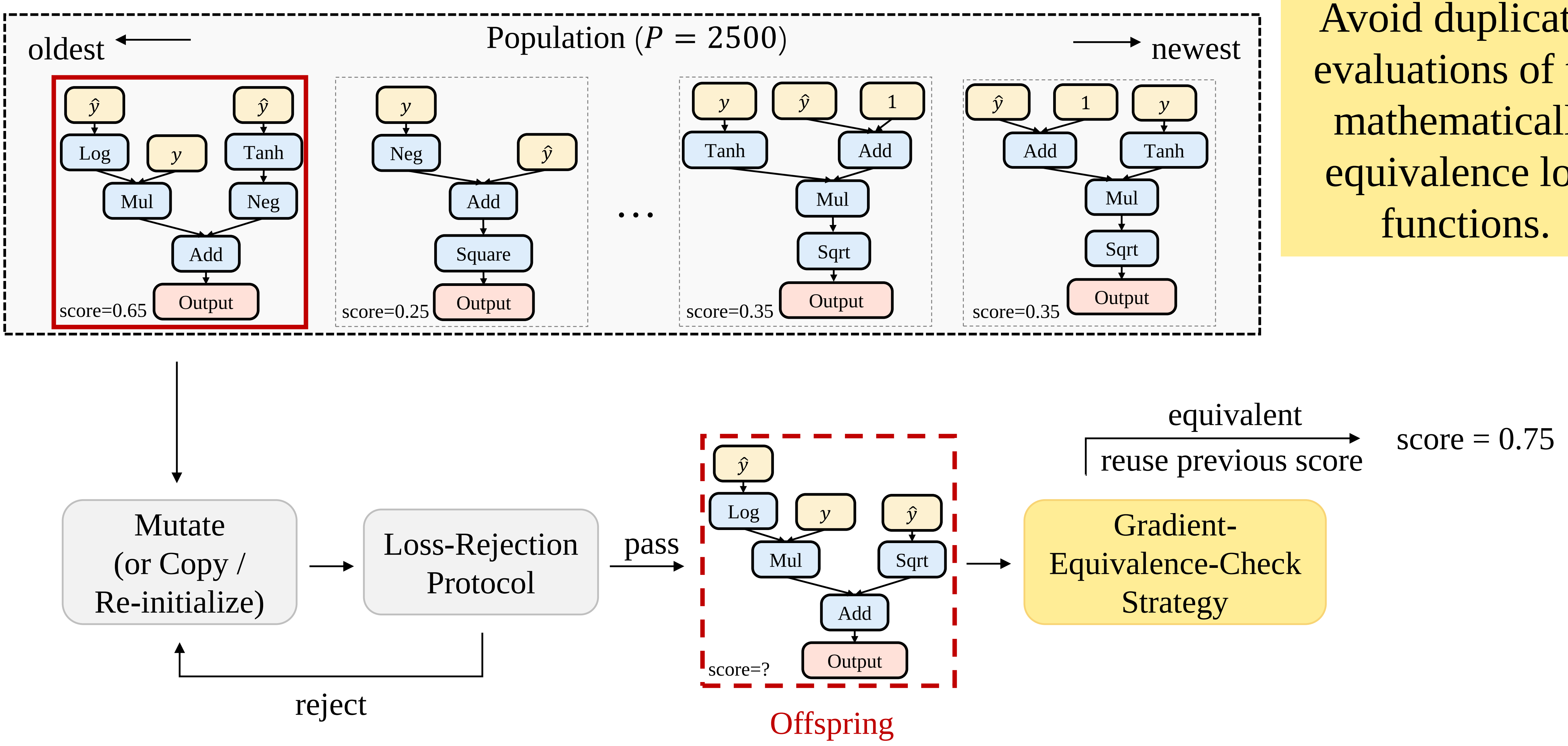
Search Algorithm – Loss Rejection Protocol



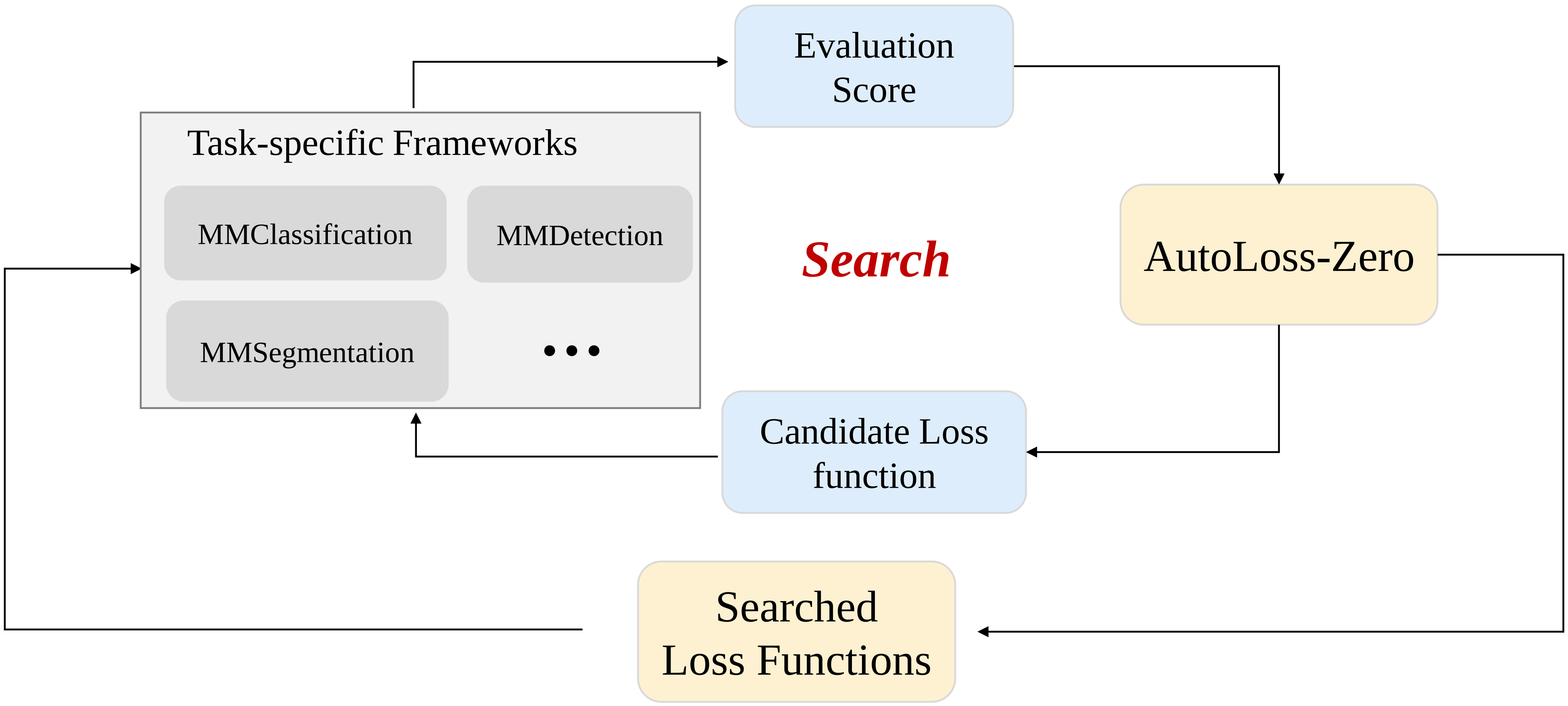
Reject the unpromising loss functions until the mutant passes the Loss-Rejection Protocol.

Search Algorithm – Gradient-Equivalence-Check Strategy

Avoid duplicated evaluations of the mathematically equivalence loss functions.



Open-MMLab



Evaluation Score
Task-specific Frameworks
MMClassification
MMDetection
MMSegmentation
• • •
Search
AutoLoss-Zero
Candidate Loss function
Searched Loss Functions

Experiments – Semantic Segmentation

Loss Function		mIoU	FWIoU	gAcc	mAcc	BIoU	BF1
Cross Entropy		78.7	91.3	**95.2**	87.3	70.6	65.3
WCE [49]		69.6	85.6	91.1	**92.6**	61.8	37.6
DiceLoss [34]		77.8	91.3	95.1	87.5	69.9	64.4
Lovàsz [2]		79.7	**91.8**	**95.4**	88.6	72.5	66.7
DPCE [4]		79.8	**91.8**	**95.5**	87.8	71.9	66.5
SSIM [40]		79.3	**91.7**	**95.4**	87.9	71.5	66.4
mIoU	ASL [27]	**81.0**	**92.1**	**95.7**	88.2	73.4	68.9
	Ours	**80.7**	**92.1**	**95.7**	89.1	74.1	66.0
FWIoU	ASL [27]	80.0	**91.9**	**95.4**	89.2	75.1	65.7
	Ours	78.7	**91.7**	**95.2**	87.7	72.9	64.6
gAcc	ASL [27]	79.7	**91.8**	**95.5**	89.0	74.1	64.4
	Ours	79.4	**91.7**	**95.3**	88.7	73.6	64.8
mAcc	ASL [27]	69.8	85.9	91.3	**92.7**	72.9	35.6
	Ours	75.3	89.2	93.7	**92.6**	73.7	44.1
BIoU	ASL [27]	49.0	69.9	62.6	81.3	**79.2**	39.0
	Ours	39.8	66.6	79.7	47.8	77.6	45.5
BF1	ASL [27]	1.9	1.0	2.7	6.5	7.4	74.8
	Ours	6.0	54.6	73.8	7.3	9.4	**79.8**

Experiments – More tasks

Object Detection

Loss Function				mAP
Cls_{RPN}	Reg_{RPN}	Cls_{RCNN}	Reg_{RCNN}	
CE	L1	CE	L1	37.3
CE	L1	CE	IoULoss [59]	37.9
CE	L1	CE	GIoULoss [48]	37.6
CE	L1	CSE-Auto-A [33]		38.5
CE	L1	Ours		38.0
Ours				38.1

Instance Segmentation

Loss Function	mAP
CE + L1 + CE + L1 + CE	34.6
CE + L1 + CE + IoULoss [59] + CE	34.4
CE + L1 + CE + GIoULoss [48] + CE	34.7
Ours	34.8

Pose Estimation

Loss Function	mAP
MSE	71.5
Ours	72.0

Our searched losses are on-par or better compared with the previous handcrafted / searched losses on various vision tasks!

Generalization of the Searched Losses

- Semantic Segmentation

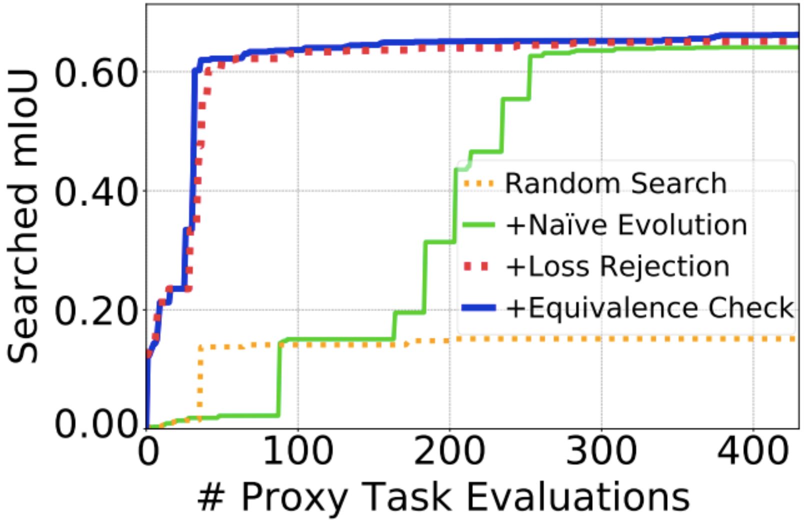
Dataset		Cityscapes		VOC			
Network		R101-DLv3+		R50-DLv3+		R101-PSP	
Loss Function		mIoU	BF1	mIoU	BF1	mIoU	BF1
Cross Entropy		80.0	62.2	76.2	61.8	77.9	64.7
mIoU	ASL [27]	**80.7**	66.5	**78.4**	66.9	**78.9**	65.7
	Ours	**80.4**	63.8	**78.0**	62.8	**78.5**	64.9
BF1	ASL [27]	6.7	**78.0**	1.4	70.8	1.6	**71.8**
	Ours	9.3	**77.7**	7.2	**78.3**	5.1	**71.3**

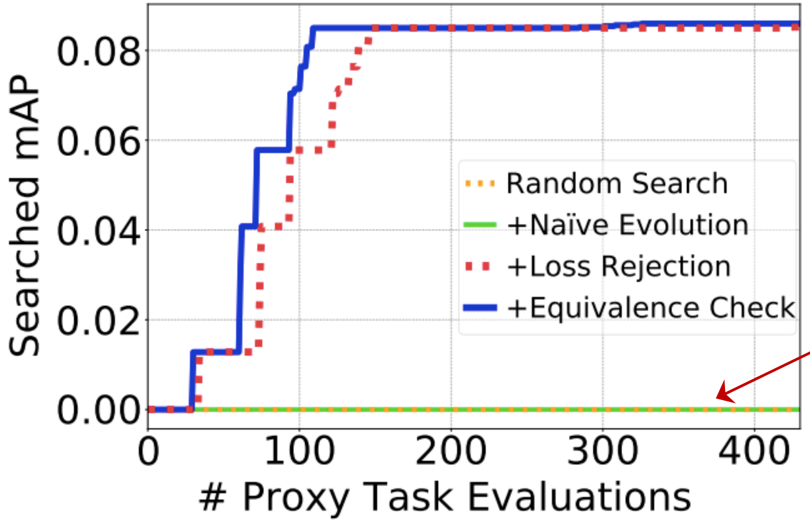
- Object Detection

Dataset	COCO	VOC
Network	ResNet-101	ResNet-50
Loss Function	mAP	mAP
CE + L1 + CE + IoULoss [59]	39.7	80.4
Ours	39.9	80.6

Experiments – Search Efficiency

	Speed-up	# Explored Losses in 48h
Naïve Evolution	$1\times$	~ 300
+ Loss-Rejection Protocol	$\sim 700\times$	$\sim 2.1\times 10^5$
+ Gradient-Equivalence-Check Strategy	$\sim 1000\times$	$\sim 3.2\times 10^5$
+ Stop Training for Invalid Loss Values	$\sim 5000\times$	$\sim 1.5\times 10^6$





No promising loss functions can be discovered without Loss-Rejection Protocol

Take-away

- AutoLoss-Zero is the first general framework for searching loss functions from scratch for generic tasks.

- A novel loss-rejection protocol and a gradient-equivalence-check strategy greatly improve the search efficiency, and are generally applicable to various tasks and metrics.

- The searched loss functions show competitive performance, and are transferable across different models and datasets.

Thanks!