Visual Recognition Meets Long-tail Data Distribution

Songyang Zhang
zhangsongyang@megvii.com
www.zhangsongyang.com

PLUS Lab, ShanghaiTech University
2021/08/19

上海科技大学
ShanghaiTech University

中国科学院大学
University of Chinese Academy of Sciences

MEGVII 旷视

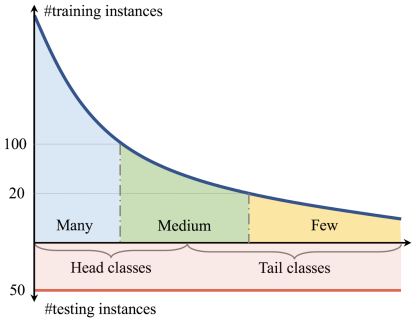
Outline

Outline

Problem Definition

Long-tail Classification

- Goal: Learn a mapping from input to label space. $y = \mathcal{M}(\mathbf{I}; \Theta)$
- Label space: $\mathcal{C} = \{c_1, \cdots, c_K\}$
- Model: feature extractor $f(\cdot)$ and classifier head $h(\cdot)$
  - Dot-product: $z_j = \mathbf{w}_j^\top \mathbf{x}$
  - Cosine Similarity: $z_j = s \cdot \frac{\mathbf{w}_j^\top \mathbf{x}}{||\mathbf{w}_j^\top|| ||\mathbf{x}||}$
- Train Data: Imbalanced $\mathcal{D}_{tr} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N, \mathbf{x} = f(\mathbf{I}, \theta_f), \mathbf{z} = h(\mathbf{x}, \theta_h)$
- Evaluation: Balanced metric.



Long-tail data distribution

Outline

Tasks and Benchmarks

- Image Classification
  - Benchmarks: CIFAR-10/100, ImageNet-LT[1], iNaturalist[2], Places-LT[3]
  - Evaluation: F1-score, Mean Accuracy
- Object Detection/Instance Segmentation
  - Benchmarks: Visual Genome, LVIS v0.5/v1.0[4]
  - Evaluation: mAP
- Visual Relation Detection
  - Benchmarks: Visual Genome, Open Image V4/V6
  - Evaluation: Mean Recall

[1] Ziwei Liu et al. "Large-Scale Long-Tailed Recognition in an Open World". In: *CVPR*. 2019.

[2] Grant Van Horn et al. "The inaturalist species classification and detection dataset". In: *CVPR*. 2018.

[3] Ziwei Liu et al. "Large-Scale Long-Tailed Recognition in an Open World". In: *CVPR*. 2019.

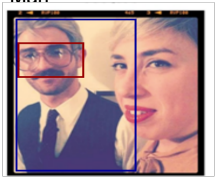
[4] Agrim Gupta, Piotr Dollar, and Ross Girshick. "LVIS: A Dataset for Large Vocabulary Instance Segmentation". In: *CVPR*. 2019.

Visual Relation Detection

Visual Relationship

- The interacting relationship between the entities(instances) that can be recognized by the visual clues.
- Triplets: (Subject entity,**Predicate**, Object entity)
- A structured, high-level visual information

Who were Glasses?
Man



(Man, Wear, Glasses)

Scene Graph

- A graph composed by the multiple visual relationships
- **Nodes: Entities(subject/object)**
- **Edges: Predicates**

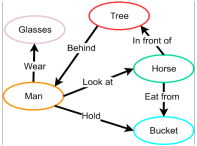


Illustration of SGG

Outline

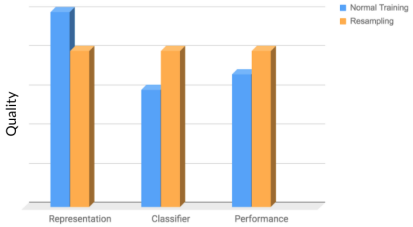
One-stage Learning Framework

Classification Performance = Representation Quality + Classifier Head Quality

One-stage Learning

Joint learn feature extractor $f(\cdot)$ and classifier head $h(\cdot)$

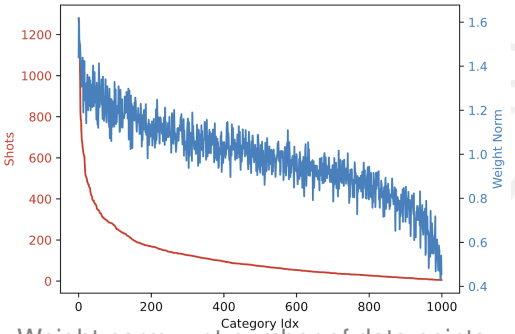
- With re-balance strategies/knowledge transfer ideas
- End-to-end training and the model is jointly optimized.



Classification Performance

Two-stage Framework

- Step-1: Joint learn feature extractor and classifier
- Step-2: Fix feature extractor and update classifier with class distribution prior
  - Non-parametric Logit Adjustment
  - Parametric Learned Classifier



Weight norm w.r.t number of data points.

Bound analysis

Non-parametric Logit Adjustment

- NCM(Nearest Centroid Mean)[a]
  - $\hat{z}_j = \lambda \cdot \frac{\mathbf{e}_j \cdot \mathbf{x}}{|\mathbf{e}_j||\mathbf{x}|}$
  - $\mathbf{e}_j$ is the mean feature of j-th categories in training set.
- τ-normalized[b]
  - $\hat{z}_j = \frac{\mathbf{w}_j^\top \mathbf{x}}{||\mathbf{w}_j||^\tau}$
- Logit Adjustment[c]
  - $\hat{z}_j = z_j - \lambda \log(\rho_j)$
  - ρ_j is the frequency of j-th categories.
- Deconfound[d](cls&det)
  - Cosine similarity classifier head. $\hat{z}_j = z_j - \lambda d(\mathbf{x}, \mathbf{e}) \mathbf{w}_j^\top \mathbf{e}$
  - $\mathbf{e}$ is mean feature of training data.

[a] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

[b] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

[c] Aditya Krishna Menon et al. "Long-tail learning via logit adjustment". In: *arXiv preprint* (2020).

[d] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. "Long-tailed classification by keeping the good and removing the bad momentum

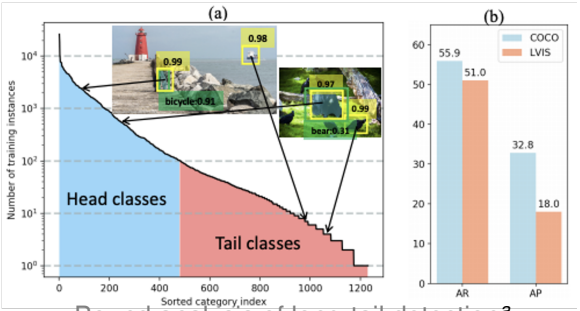
Parametric Learned Classifier(Proposed in Image Classification)

Parametric[a]

[a] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

- τ-normalized
  - $\hat{z}_j = \frac{\mathbf{w}_j^\top \mathbf{x}}{||\mathbf{w}_j||^\tau}$
- cRT(Classifier re-training)
  - $\hat{z}_j = \hat{\mathbf{w}}_j^\top \mathbf{x}$
  - ignore original classifier head.
- LWS
  - $\hat{z}_j = \alpha_j \cdot z_j$
  - α_j is class-wise learnable scale.

Parametric Learned Classifier(SimCal[5])



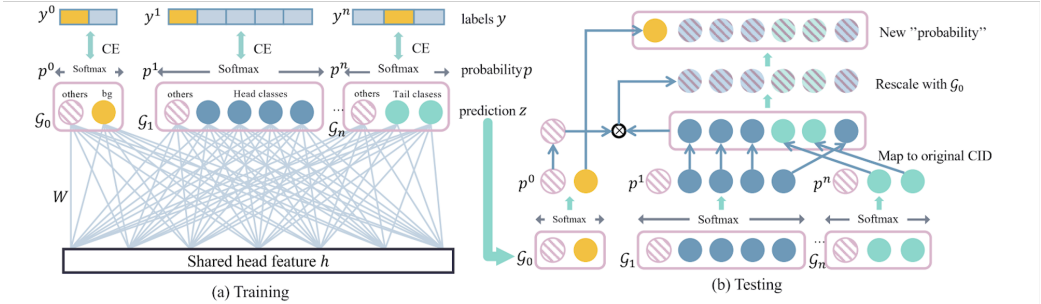
Bound analysis of long-tail detection[a]

- Like cRT[b]
- Introduce a bi-level balanced-sampling to create a batch:
  - First sample *n* categories uniformly.
  - Then sample images for each category
  - Only keep the proposals of sampled categories.

[a] Tao Wang et al. "The Devil is in Classification: A Simple Framework for Long-tail Instance Segmentation". In: *ECCV*. 2020.

[b] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

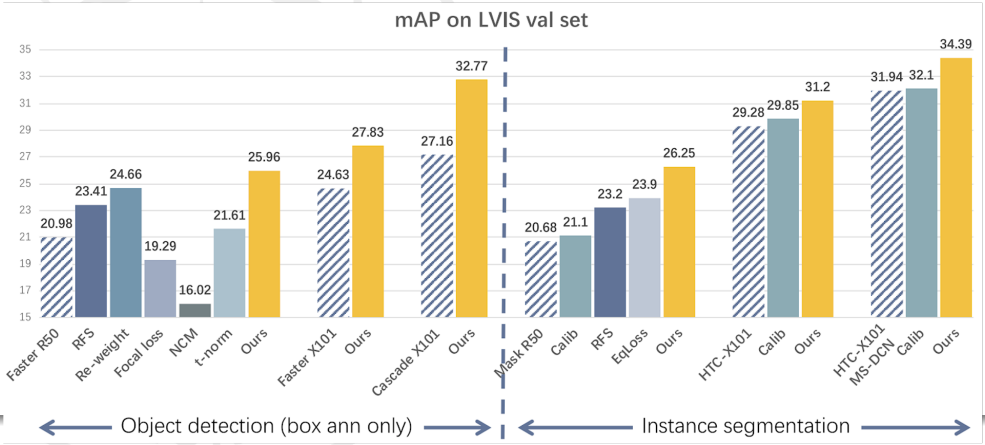
Parametric Learned Classifier(BalancedGroupSoftmax[6])



- Rely on the heuristic group split
- Too many hyper-parameters

[6] Yu Li et al. "Overcoming Classifier Imbalance for Long-Tail Object Detection With Balanced Group Softmax". In: *CVPR* 2020.

Parametric Learned Classifier(BalancedGroupSoftmax[7])



[7] Yu Li et al. "Overcoming Classifier Imbalance for Long-Tail Object Detection With Balanced Group Softmax". In: *CVPR* 2020.

Outline

Data: Re-sampling strategy[8]

$$p_j = \frac{n_j^q}{\sum_{i=1}^{C} n_i^q}, \quad q \in [0, 1]$$

- Instance-balanced re-sampling($q = 1$)
- Class-balanced re-sampling($q = 0$)
- Square-root re-sampling($q = 1/2$)
- Progressively-balanced re-sampling

$$p_j^{\mathrm{PB}}(t) = \left(1 - \frac{t}{T}\right) p_j^{\mathrm{IB}} + \frac{t}{T} p_j^{\mathrm{CB}}$$

  - t is the current epoch
  - T is the total epcoh
  - CB and IB are hyper-parameters.

[8] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

Loss: Re-weight

- Focal Loss[9]:

$$\mathcal{L}_{\text{focal}} := (1-\tilde{p})^{\gamma}\,\mathcal{L}_{CE} = -(1-\tilde{p})^{\gamma}\log(\tilde{p})$$

- Effective Number Re-weight[10]:

$$\mathcal{L}^{\text{CB}}_{\text{softmax}}(\mathbf{z}, y) = -\frac{1-\beta}{1-\beta^{n_y}}\log\left(\frac{\exp(z_y)}{\sum_{j=1}^{C}\exp(z_j)}\right)$$

- LDAM(Label-distribution-aware margin)[11]

$$\mathcal{L}_{\text{LDAM}} := -\log\frac{e^{z_j-\Delta_j}}{e^{z_j-\Delta_j}+\sum_{c}\neq j e^{z_c-\Delta_c}}$$

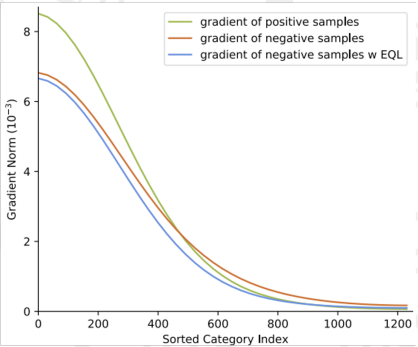
  - Δ_j is a class-aware margin, inversely proportional to number of j-th categories data.

[9] Tsung-Yi Lin et al. "Focal loss for dense object detection". In: *ICCV*. 2017.

[10] Yin Cui et al. "Class-balanced loss based on effective number of samples". In: *CVPR*. 2019.

[11] Kaidi Cao et al. "Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss". In: *NeurIPS*. 2019.

Loss: Re-weight



Gradient norm w.r.t data distribution

Equalization Loss[a]

[a] Jingru Tan et al. "Equalization Loss for Long-Tailed Object Recognition". In: *CVPR*. 2020.

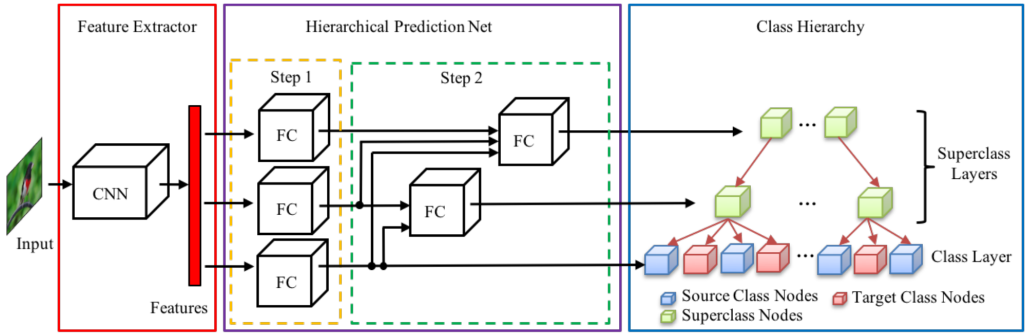
$$\mathcal{L}_{SEQL} = -\sum_{j=1}^{C} y_j \log\left(\tilde{p}_j\right) \tag{1}$$

$$\tilde{p}_j = \frac{e^{z_j}}{\sum_{k=1}^{C} \tilde{w}_k e^{z_k}} \tag{2}$$

$$\tilde{w}_k = 1 - \beta T_\lambda\left(f_k\right)\left(1 - y_k\right) \tag{3}$$

- β is a random variable with a probability of γ to be 1 and $1-\gamma$ to be 0

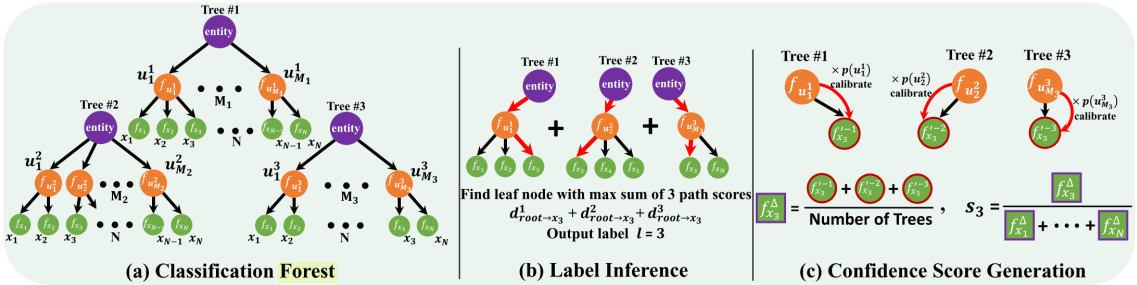
Category Hierarchy(Knowledge Transfer With Class Hierarchy)[12]



Overview of the framework

[12] Aoxue Li et al. "Large-Scale Few-Shot Learning: Knowledge Transfer With Class Hierarchy". In: *CVPR*. 2019.

Category Hierarchy(Forest R-CNN)[13]



Overview of the framework

[13] Jialian Wu et al. "Forest R-CNN: Large-vocabulary long-tailed object detection and instance segmentation". In: *ACM MM*. 2020.

Category Hierarchy(Taxonomic Classifier)[14]

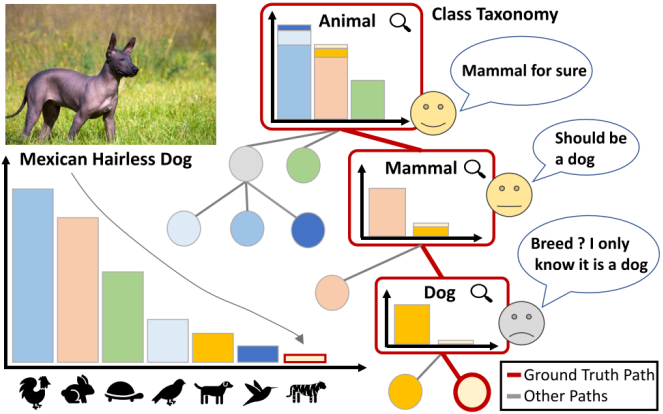
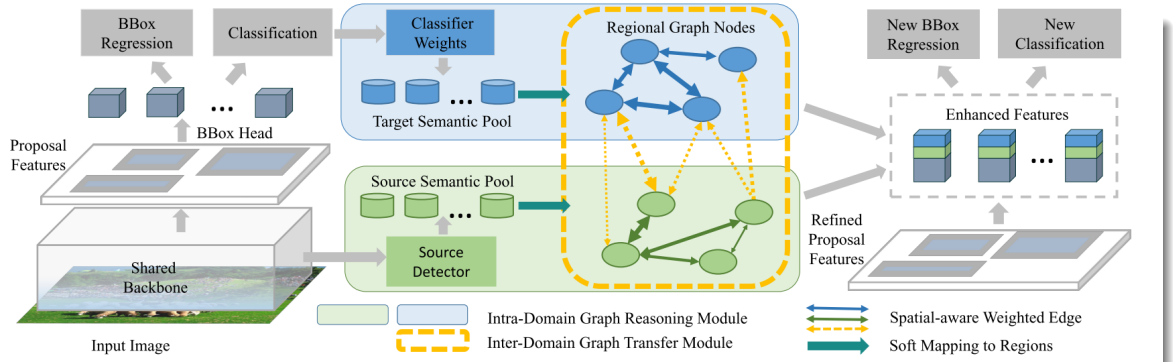


Illustration of the idea

[14] Tz-Ying Wu and Pedro Morgado. "Solving Long-tailed Recognition with Deep Realistic Taxonomic Classifier". In: *ECCV*. 2020.

Knowledge Transfer(Universal R-CNN)[15]



Overview of the framework

[15] Hang Xu et al. "Universal-RCNN: Universal Object Detector via Transferable Graph R-CNN.". In: *AAAI*. 2020.

Outline

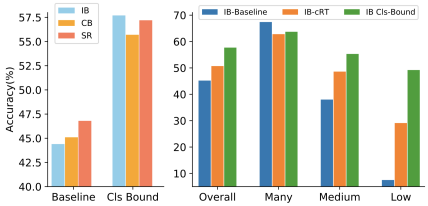
Distribution Alignment: A Unified Framework for Long-tail Visual Recognition

Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, Jian Sun

ShanghaiTech University
Megvii Technology
University of Chinese Academy of Sciences

In Conference on Computer Vision and Pattern Recognition(CVPR) 2021

Empirical Study



Empirical Bound Experiment

- Training feature extractor with various strategies on ImageNet-LT dataset.
- Re-training the classifier only on ImageNet dataset.
- IB: instance-balanced; CB: class-balanced; SR: square-root

Conclusion

- Feature extractor learned with instance-balanced sampling can provide **sufficient** quality.
- **Bottleneck** is the classifier head.
- Previous works remain **sub-optimal**

Empirical Study

- The first stage produces a **strong feature representation** that can potentially lead to large performance gain and the **instance-based** sampling achieves better overall results
- The **biased decision boundary** in the feature space seems to be the **performance bottleneck** of the existing long-tail methods.

Outline

Our Method

Two-stage Learning Framework

1. Joint Learning Stage.
   Learn the feature extractor $f(\cdot)$ and original classifier head $h_o(\cdot)$ with **instance-balanced** strategy jointly on imbalanced $\mathcal{D}_{tr}$
2. Distribution Calibration Stage
   Freeze $f(\cdot)$ and **adjust the decision boundary** of the classifier head.
   - Adaptive Calibration Function.
   - Distribution Alignment with Generalized Re-weighting.

Our Method

Adaptive Calibration Function

- Original prediction: $\mathbf{z}^o = [z_1^o, \cdots, z_K^o]$ from $h_o(\cdot)$
- Reduce adverse impact from the limited data for tail.
- $s_j = \alpha_j \cdot z_j^o + \beta_j, \quad \forall j \in \mathcal{C}, \quad \alpha_j$ and β_j are the calibration parameters for each class, which will be learned from data
- Introduce a confidence score function $\sigma(\mathbf{x})$ to adaptively adjust the score.

$$\hat{z}_j = \sigma(\mathbf{x}) \cdot s_j + (1 - \sigma(\mathbf{x})) \cdot z_j^o \tag{4}$$

$$= (1 + \sigma(\mathbf{x})\alpha_j) \cdot z_j^o + \sigma(\mathbf{x}) \cdot \beta_j \tag{5}$$

- Prediction distribution after calibration.

$$p_m(y = j|\mathbf{x}) = \frac{\exp(\hat{z}_j)}{\sum_{k=1}^{C} \exp(\hat{z}_k)}. \tag{6}$$

Our Method

Alignment with Generalized Re-weighting

- Align the model prediction $p_m(\cdot)$ with a reference distribution $p_r(\cdot)$, by minimizing the expected KL-divergence

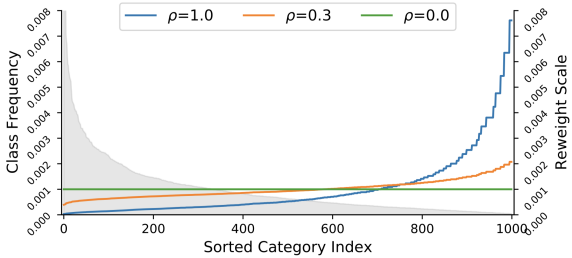
$$\mathcal{L} = \mathbb{E}_{\mathcal{D}_{tr}} \left[\mathcal{KL}(p_r(y|\mathbf{x}) || p_m(y|\mathbf{x})) \right] \tag{7}$$

$$\approx -\frac{1}{N} \sum_{i=1}^{N} \left[\sum_{y \in \mathcal{C}} p_r(y|\mathbf{x}_i) \log(p_m(y|\mathbf{x}_i)) \right] + C \tag{8}$$

- Reference distribution: $p_r(y = c|\mathbf{x}_i) = w_c \cdot \delta_c(y_i), \quad \forall c \in \mathcal{C}$, w_c is the class weight.
- Define empirical class frequencies $\mathbf{r} = [r_1, \cdots, r_K]$ on the training set.

$$w_c = \frac{(1/r_c)^\rho}{\sum_{k=1}^{K} (1/r_k)^\rho}, \quad \forall c \in \mathcal{C} \tag{9}$$

Connection with Previous Work



Method	Align Method			
	Type	Balance	Magnitude	Margin
Joint	-	-	-	-
LWS[6]	L	CB-RS	α_j	0
τ-Normalized[6]	H	CB-RS	$1/\|\|\mathbf{w}_j\|\|^\tau$	0
Logit Adjust[11]	H	-	1.0	$-\lambda \log(r_j)$
Deconfound*[14]	H	-	1.0	$-\lambda d(\mathbf{x}, \mathbf{e})\mathbf{w}_j^\top \mathbf{e}$
DisAlign	L	G-RW	$1 + \sigma(\mathbf{x})\alpha_j$	$\sigma(\mathbf{x})\beta_j$
DisAlign*	L	G-RW	$1 + \sigma(\mathbf{x})\alpha_j$	$\sigma(\mathbf{x})\beta_j$

Outline

Experiment on Classification

Method	Align Type	Top-1 Accuracy@R-50				Top-1 Accuracy@X-50			
		Average	Many	Medium	Few	Average	Many	Medium	Few
Baseline[6]	-	41.6	64.0	33.8	5.8	44.4	65.9	37.5	7.7
Baseline*	-	48.4	68.4	41.7	15.2	49.2	68.9	42.8	15.6
NCM[6]	Hand-Craft	44.3	53.1	42.3	26.5	47.3	56.6	45.3	28.1
τ-Norm[6]		46.7	56.6	44.2	27.4	49.4	59.1	46.9	30.7
Logit Adjust(post)[11]		50.4	-	-	-	-	-	-	-
Deconfound*[14]		-	-	-	-	51.8	62.7	48.8	31.6
cRT[6]	Learnable	47.3	58.8	44.0	26.1	49.6	61.8	46.2	27.4
cRT*[6]		-	-	-	-	49.7	60.4	46.8	29.3
LWS[6]		47.7	57.1	45.2	29.3	49.9	60.2	47.2	30.3
DisAlign		51.3	59.9	49.9	31.8	52.6	61.5	50.7	33.1
DisAlign*		**52.9**	61.3	52.2	31.4	**53.4**	62.7	52.1	31.4

Quantitative results on ImageNet-LT. $*$ denotes the model uses cosine classifier. **R-50** and **X-50** means the ResNet-50 and ResNeXt-50, respectively.

Experiment on Semantic Segmentation

Framework	Method	B	Mean IoU(%)				Mean Accuracy(%)			
			Average	Head	Body	Tail	Average	Head	Body	Tail
FCN[12]	Baseline	R-50	38.1	64.6	40.0	29.6	46.3	78.6	49.3	35.4
	DisAlign		40.1(+2.0)	65.0(+0.4)	42.8(+2.8)	31.3(+1.7)	51.4(+5.1)	78.6(+0.0)	56.1(+6.8)	40.6(+5.2)
	Baseline	R-101	41.4	67.0	43.3	33.2	50.2	80.6	52.9	40.1
	DisAlign		43.7(+2.3)	67.4(+0.4)	46.1(+2.8)	35.7(+2.5)	55.9(+5.7)	80.6(+0.0)	59.7(+6.8)	46.4(+6.3)
	Baseline	S-101	46.2	67.6	48.0	39.1	57.3	79.4	61.7	48.2
	DisAlign		46.9(+0.7)	67.7(+0.1)	48.2(+0.2)	40.3(+1.2)	60.1(+2.8)	79.7(+0.3)	64.2(+2.5)	51.9(+3.7)
DeepLabV3$^+$[2]	Baseline	R-50	44.9	67.7	48.3	36.4	55.0	80.1	60.8	44.1
	DisAlign		45.7(+0.8)	67.7(+0.0)	48.6(+0.3)	37.8(+1.4)	57.3(+2.3)	80.8(+0.7)	63.0(+2.2)	46.9(+2.8)
	Baseline	R-101	46.4	68.7	49.0	38.4	56.7	80.9	61.5	46.7
	DisAlign		47.1(+0.7)	68.7(+0.0)	49.4(+0.4)	39.6(+1.2)	59.5(+2.8)	81.4(+0.5)	64.2(+2.7)	50.3(+3.6)
	Baseline	S-101	47.3	69.0	49.7	39.7	58.1	80.8	63.4	48.2
	DisAlign		47.8(+0.5)	68.9(-0.1)	49.8(+0.1)	40.7(+1.0)	60.1(+2.0)	81.0(+0.2)	65.5(+2.1)	52.0(+3.8)

Performance of semantic segmentation on ADE-20K: All baseline models are trained with an image size of 512×512 and 160K iteration in total. **B** is backbone network(R-50, R-101, S-101 denote ResNet-50, ResNet-101 and ResNeSt-101, respectively).

Experiment on Detection/Instance Segmentation

Pre-Train	Method	BBox AP				Mask AP			
		AP_{bbox}	AP^r_{bbox}	AP^c_{bbox}	AP^f_{bbox}	AP_{mask}	AP^r_{mask}	AP^c_{mask}	AP^f_{mask}
ImageNet	Baseline	20.8	3.3	19.5	29.4	21.2	3.7	21.6	28.4
	Baseline*	22.8	10.2	21.1	30.1	23.8	11.5	23.7	28.9
	Focal Loss[9]	21.9	-	-	-	21.0	9.3	21.0	25.8
	SimCal[16]	22.6	13.7	20.6	28.7	23.4	16.4	22.5	27.2
	LST[5]	22.6	-	-	-	23.0	-	-	-
	RFS[4]	23.6	12.8	22.3	29.4	24.3	14.6	24.0	28.5
	EQL[13]	23.3	-	-	-	22.8	11.3	24.7	25.1
	DisAlign	23.9	7.5	25.0	29.1	24.3	8.5	26.3	28.1
	DisAlign*	**25.6**	13.7	25.6	30.5	**26.3**	14.9	27.6	29.2
COCO	Baseline	22.8	2.6	21.8	32.0	23.9	2.8	23.4	30.5
	Baseline*	25.0	10.2	23.9	32.3	25.3	11.0	25.5	30.7
	GroupSoftmax[8]	25.8	15.0	25.5	30.4	26.3	18.0	26.9	28.7
	DisAlign	25.5	8.2	26.3	32.4	25.7	9.4	27.6	29.7
	DisAlign*	**27.6**	14.8	27.9	32.4	**27.9**	16.2	29.3	30.8

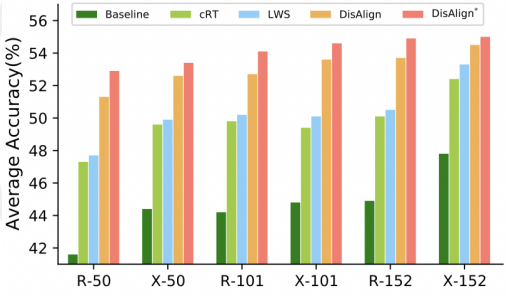
Comparison with the-state-of-art on LVIS with Mask-R-CNN-FPN(ResNet-50 backbone). All results are evaluated on the LVIS v0.5 validation set with the score threshold at 0.0001. ($*$ denotes cosine classifier for bbox classification.)

Ablation Study

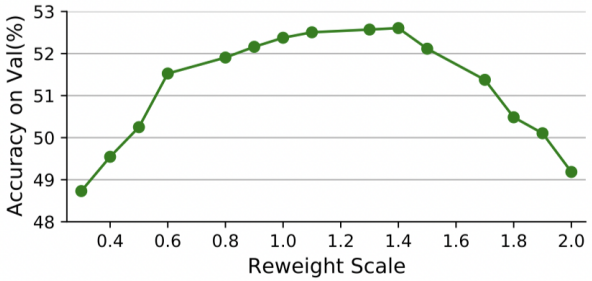
Backbone	Method	BBox AP				Mask AP			
		AP_{bbox}	AP^r_{bbox}	AP^c_{bbox}	AP^f_{bbox}	AP_{mask}	AP^r_{mask}	AP^c_{mask}	AP^f_{mask}
ResNet-50	Baseline*	26.5	8.7	25.0	36.0	23.5	8.1	22.4	31.5
	DisAlign*	30.5	17.9	30.1	36.5	27.0	15.7	27.0	31.9
ResNet-101	De-confound[14]	25.8	-	-	-	23.5	5.2	22.7	32.3
	De-confound TDE[14]	30.0	-	-	-	27.1	16.0	26.9	32.1
	Baseline*	28.9	11.8	27.7	37.8	25.6	10.5	24.9	33.0
	DisAlign*	32.7	20.5	32.8	38.1	28.9	18.0	29.3	33.3
ResNeXt-101	Baseline*	30.7	14.2	29.3	39.6	27.3	13.0	26.4	34.6
	DisAlign*	33.7	21.4	33.1	39.7	29.7	18.4	29.7	34.7

Results on LVIS v1.0 dataset with Cascade R-CNN. * denotes cosine classifier head.

Ablation Study



Different Backbone



Influence of the G-RW scale

Ablation Study

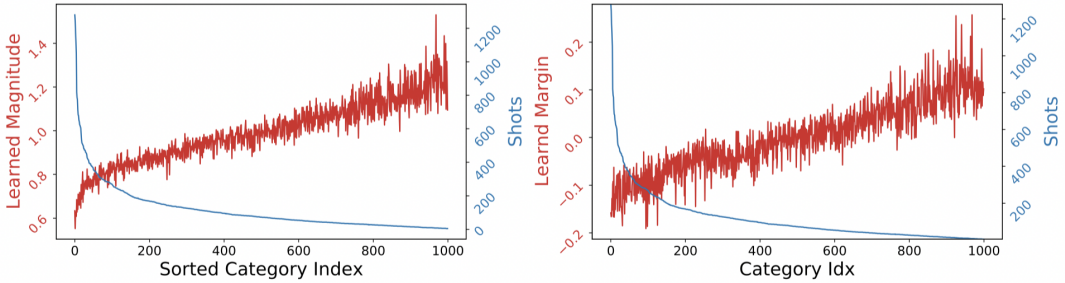


Figure 6: **Analysis of the Calibration.** We use model trained on ImageNet-LT with ResNeXt-50 for analysis.

Ablation Study

GR	MT	MG	Average	Many	Medium	Few
✗	✗	✗	41.6	64.0	33.8	5.8
✓	✓	✗	50.1	60.4	48.0	28.8
✓	✗	✓	49.9	63.9	46.9	21.2
✓	✓	✓	51.3	59.9	49.9	31.8

Ablation study of DisAlign. GR means the generalized reweight strategy. **MT** means the learnable magnitude parameter $(1+\sigma(\mathbf{x})\alpha)$ and **MG** is the learnable margin parameter $\sigma(\mathbf{x})\beta$.

Method	Calibration	G-RW	Top-1 Acc
Baseline*	-	-	49.2
cRT*	✗	✗	49.7
-	✗	✓	51.9
DisAlign*	✓	✓	53.4

Influence of Model Components. Backbone is ResNeXt-50, $*$ means cosine classifier.

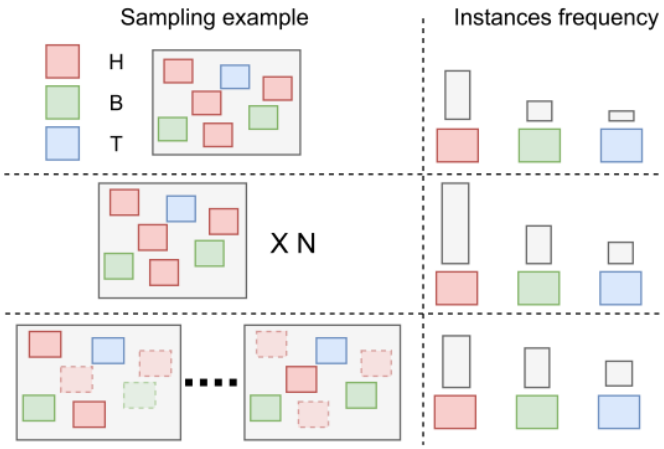
Bipartite Graph Network with Adaptive Message Passing for Unbiased Scene Graph Generation

Rongjie Li, Songyang Zhang, Bo Wan, Xuming He

ShanghaiTech University
University of Chinese Academy of Sciences
KU Leuven

In Conference on Computer Vision and Pattern Recognition(CVPR) 2021

Bi-level Data-sampling



Bi-level Sampling

Future Direction

Future Direction

- Theoretical analysis of the optimization dynamics(feature extractor and classifier)
- Contrastive learning to alleviate the data imbalance
- Strong augmentation along with transformer structure
- Semi-supervised learning

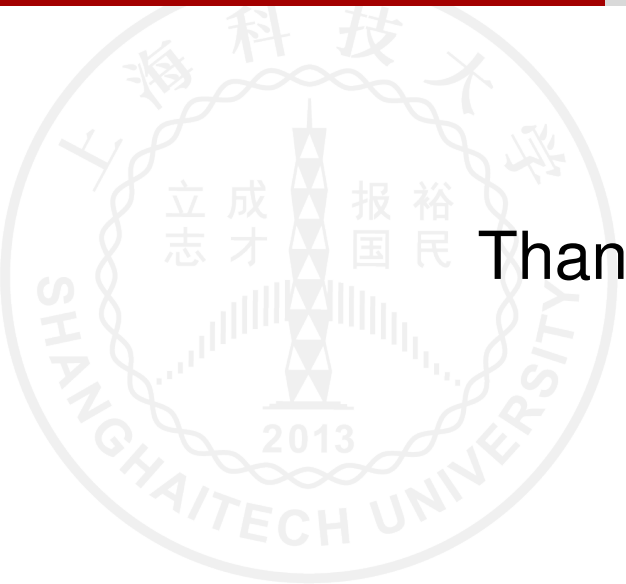
Take Home Message

Learning Framework

- **Two-stage** is more powerful.
- How to introduce **reference category distribution** is important.
- Distribution Alignment/Logit Adjustment/BalancedGroupSoftmax are effective.

Imbalance Strategy

- Sampling is more effective than re-weight in image classification
- Re-weight is more convenient than sampling in detection/instance segmentation
- Sampling and re-weight can be adopted jointly but suffer from overfitting.
- Knowledge transfer ideas are weak when using more powerful framework(Cascade R-CNN) or stronger backbones(X-101,S-101)
- **Strong augmentation** is significantly helpful in engineering.



Thanks!