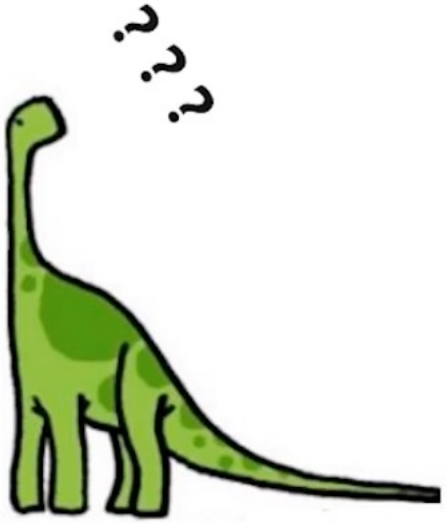
Long-tailed Distributions

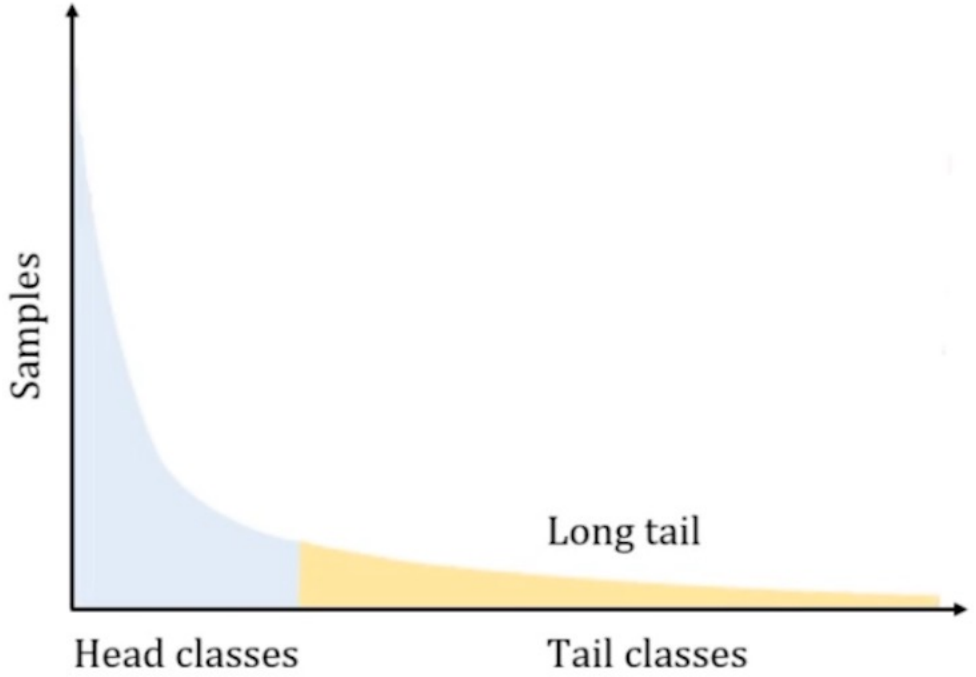
by im0qianqian

Outline

- 什么是长尾分布？
- 为什么会有长尾分布？
- 长尾分布带来什么危害？
- 论文分享

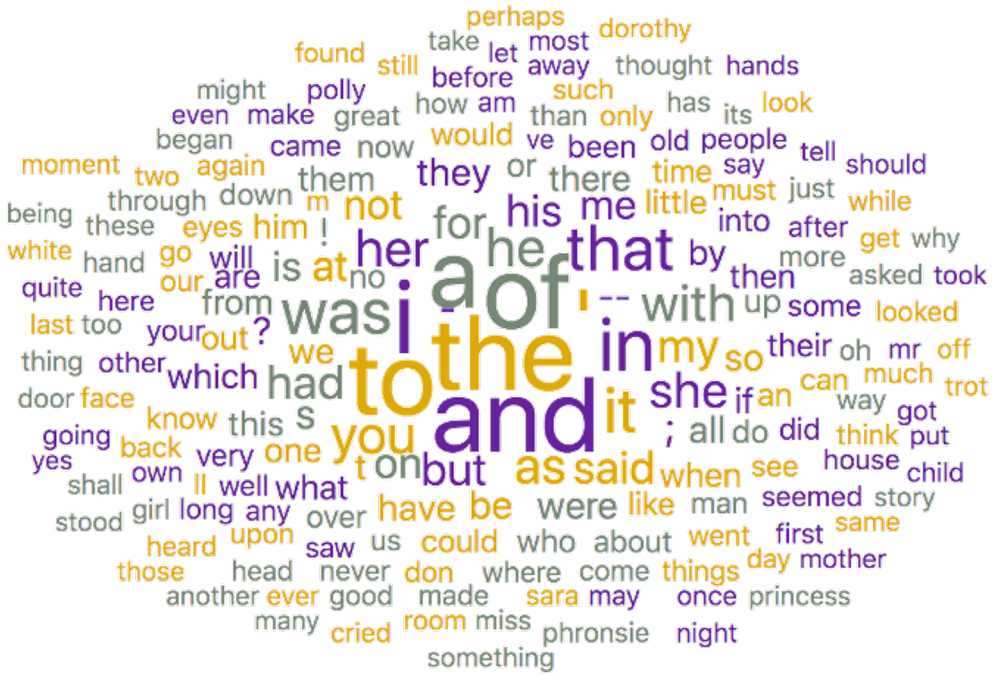
What? Long-tailed distribution

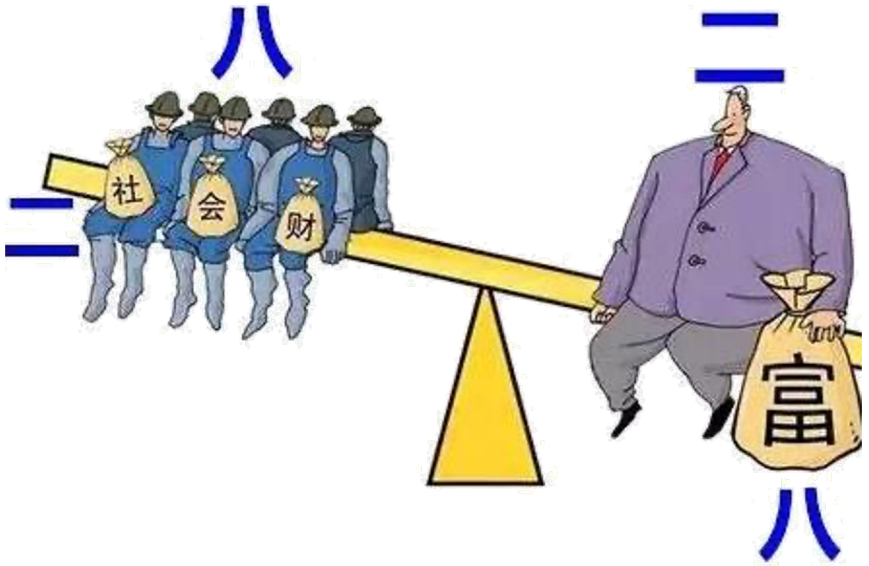


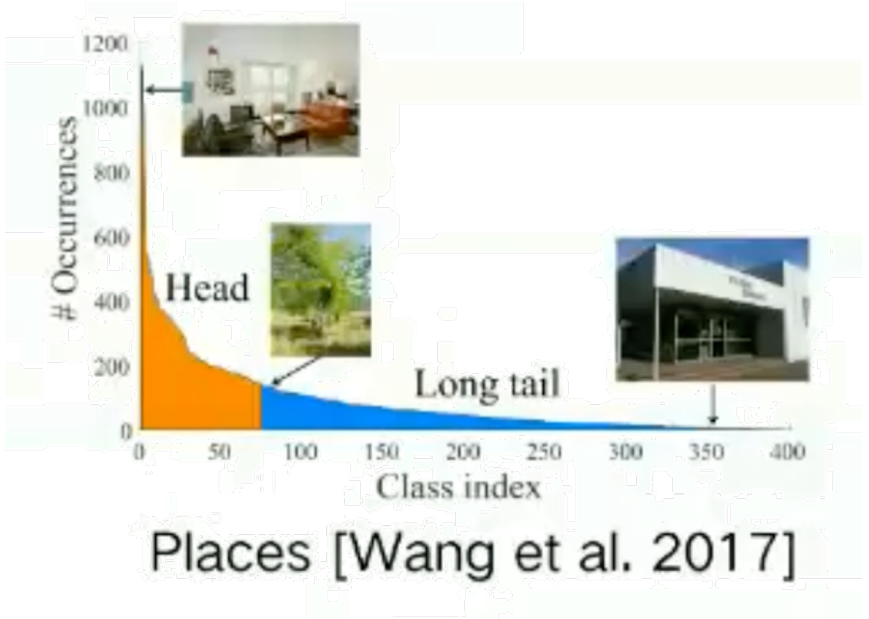


Why? Long-tailed distribution

- 人类本身生活在一个不均衡的物理世界中
- 人类的认知来源于世界本身

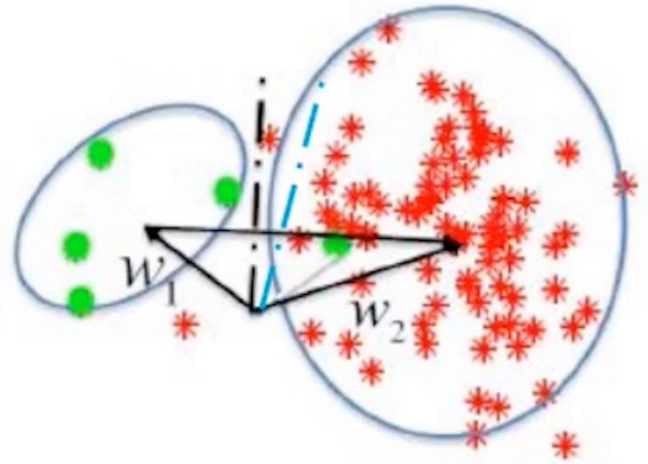


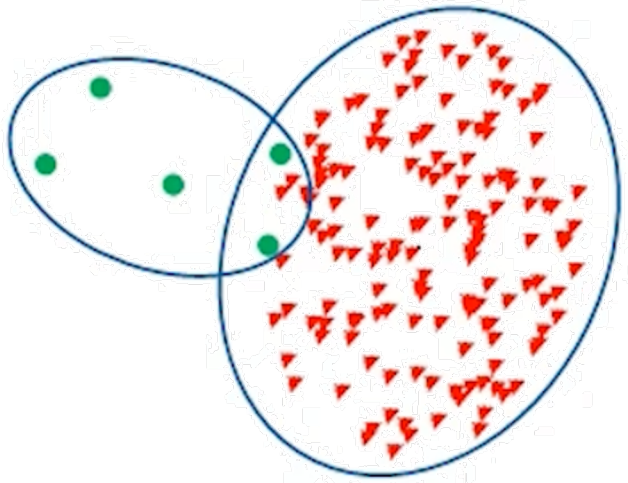




Problems caused by Long-tailed distribution

- 模型会更迁就于头部类的决策
  - 推荐中会出现，强者愈强，弱者愈弱的现象
- 头部类别特征学习较好，而尾部类则学习不充分





Re-sampling

思路：

- 头部类别欠采样
- 尾部类别过采样
- 类别均衡采样

缺点：

- 过采样可能导致尾部类别的过拟合
- 欠采样可能会漏掉一些重要的样本
- 现在很多工作都是将 re-sampling 当作整个 pipeline 的一个组件使用

Re-weighting

主要对损失函数做文章

比如比较早的方法是用样本的频率倒数给 loss 在类别层面上加权

相关工作：

- Focal loss
- Seesaw loss

Transfer Learning

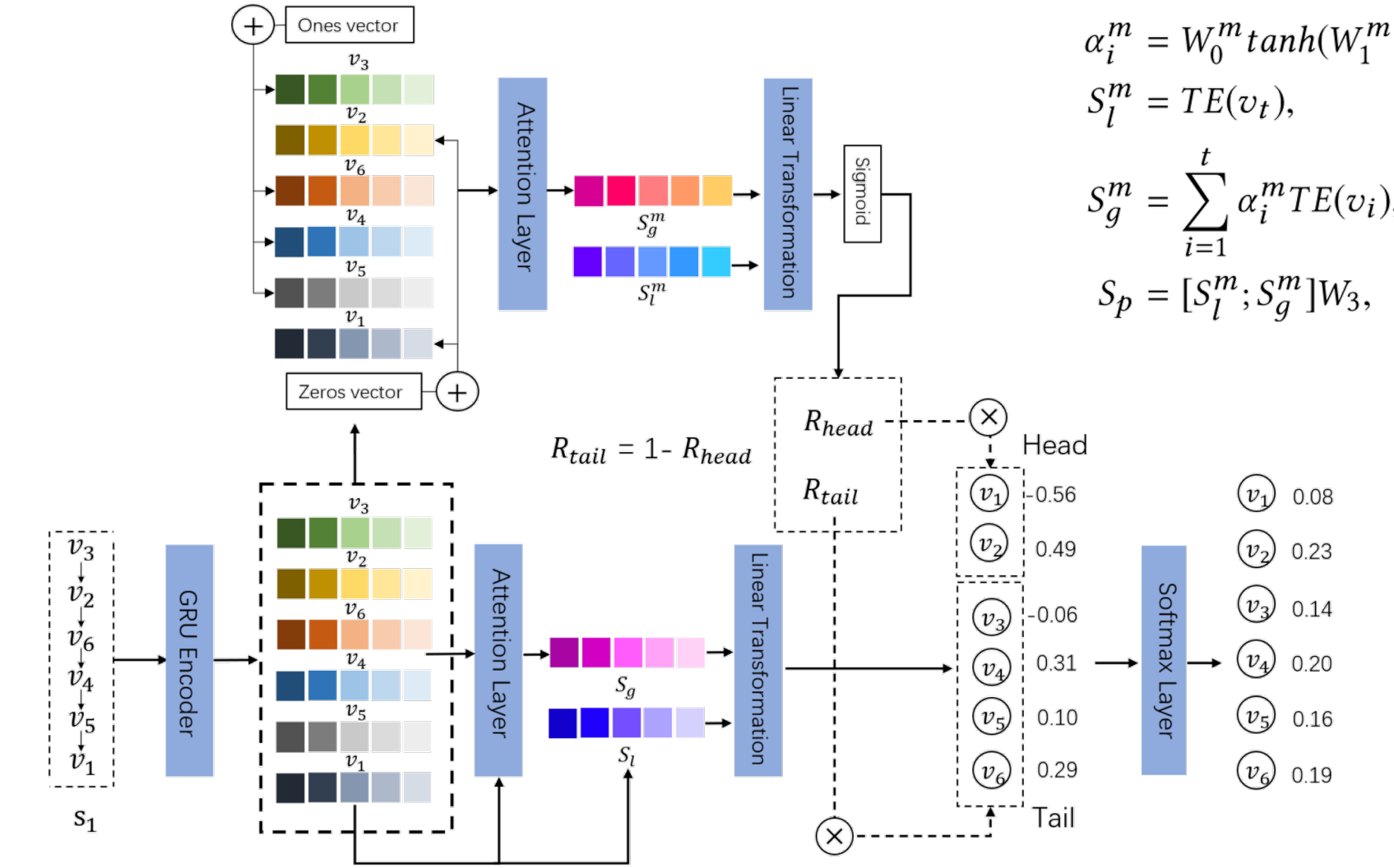
头部有足够多的样本，因此能不能从头部把 Knowledge 迁移到尾部的学习中。

从而弥补尾部样本数量的不足。

方法

- 比如先在不完整的数据集上做预训练，再把数据集分为一个个样本平衡的子集做 finetune。

TailNet



$$\alpha_i^m = W_0^m tanh(W_1^m TE(v_t) + W_2^m TE(v_i) + b^m),$$
$$S_l^m = TE(v_t),$$
$$S_g^m = \sum_{i=1}^{t} \alpha_i^m TE(v_i),$$
$$S_p = [S_l^m; S_g^m]W_3,$$

Long-tail Session-based Recommendation. RecSys 2020

TailNet

Table 1: The performance of TailNet with other baseline methods on two datasets. (PM: Preference Mechanism)

Methods	30MUSIC					YOOCHOOSE 1/4				
	MRR@20	Recall@20	Coverage@20	Tail_Coverage@20	Tail@20	MRR@20	Recall@20	Coverage@20	Tail_Coverage@20	Tail@20
POP	0.18	0.69	0.01	0	0	0.3	1.36	0.06	0	0
S-POP	8.20	18.98	15.52	10.55	20.16	17.85	27.18	19.59	9.29	1.82
Item-KNN	15.71	37.68	75.14	79.04	54.60	21.68	53.01	65.93	60.72	14.65†
FPMC	9.17	14.47	84.37†	97.45†	60.00†	19.17	46.69	70.97†	84.74†	6.83
BPR-MF	6.97	12.25	49.62	86.95	19.94	16.37	23.77	44.82	68.32	2.96
GRU4REC	20.45	**39.12**†	36.60	26.18	18.64	22.41	59.58	25.79	11.71	1.87
NARM	20.19	36.68	45.51	36.99	30.84	28.88	69.29	44.05	30.65	6.49
STAMP	13.12	23.34	14.10	3.81	12.95	30.33	70.55	41.11	26.89	6.34
RepeatNet	18.01	33.01	32.24	24.49	23.06	**31.01**†	70.69	33.82	19.00	4.62
SR-GNN	27.28	37.86	40.59	28.71	20.48	30.64	**71.39**†	33.90	19.71	6.43
TailNet without PM	28.62	39.00	34.53	21.33	11.56	30.64	69.29	42.75	28.70	6.92
TailNet-propotion	27.91	37.29	38.55	25.37	19.85	28.99	67.04	43.86	30.32	8.43
TailNet	**28.70**†	38.34	**47.86**	**40.10**	**32.13**	30.97	69.41	**45.51**	**32.31**	**8.88**

Boldface indicates the best result in neural network based methods. † indicates the best results in all methods. The scores reported in [23] on the YOOCHOOSE dataset differ because of the discrepancy in test data. For neural network based methods, we choose the values of Coverage, Tail_Coverage and Tail when they are most accurate.

TailNet

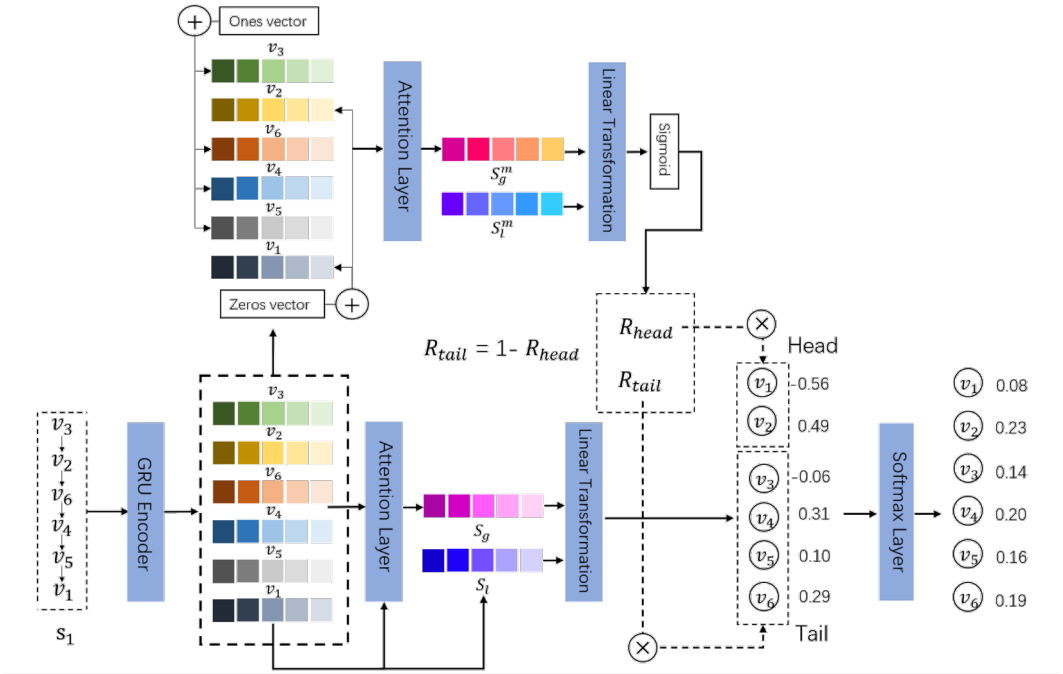


Table 2: The performance of STAMP and SR-GNN, both with and without preference mechanism, on YOOCHOOSE 1/4 dataset. (PM: Preference Mechanism)

Methods	@20(%)				
	MRR	Recall	Coverage	Tail_Coverage	Tail
STAMP	30.33	**70.55**	41.11	26.89	6.34
STAMP with PM	**30.83**	70.46	**43.80**	**30.33**	**7.08**
SR-GNN	**30.64**	**71.39**	33.90	19.71	6.43
SR-GNN with PM	30.62	71.37	**35.46**	**21.99**	**6.57**

Long-tail Session-based Recommendation. RecSys 2020

DCL: Dynamic Curriculum Learning

Sampling Scheduler

调整训练集的数据分布，将采样数据集的样本分布从原先的不平衡调整到后期的平衡状态。

Metric Learning with Easy Anchors

利用度量学习的方式，通过置信度高的简单样本将困难的正样本拉近，推远负样本，从而使不同类别之间的距离增大，便于分类。

Dynamic Curriculum Learning for Imbalanced Data Classification. IEEE/CVF International Conference on Computer Vision 2019

DCL: Sampling Scheduler

基本思想：

原始数据分布能够学习数据的主要特性，而平均采样后的分布能够学习样本量少的类别特性。

因此动态调整这一变化可以同时兼顾这两种特性。

$$D = 1 : \frac{\#C_1}{\#C_{min}} : \frac{\#C_2}{\#C_{min}} : \ldots : \frac{\#C_{K-1}}{\#C_{min}}$$

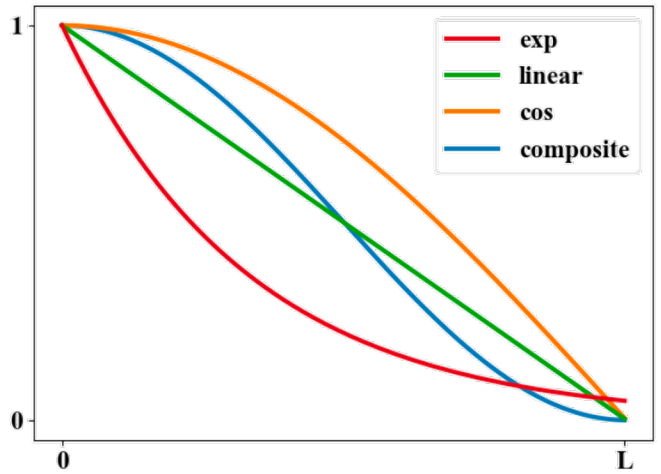
$$D_{target}(l) = D_{train}^{\,g(l)}$$

$$SF_{cos}(l) = cos(\frac{l}{L} * \frac{\pi}{2})$$

$$SF_{composite}(l) = \frac{1}{2}cos(\frac{l}{L}\pi) + \frac{1}{2}$$

$$SF_{linear}(l) = 1 - \frac{l}{L}$$

$$SF_{exp}(l) = \lambda^l$$

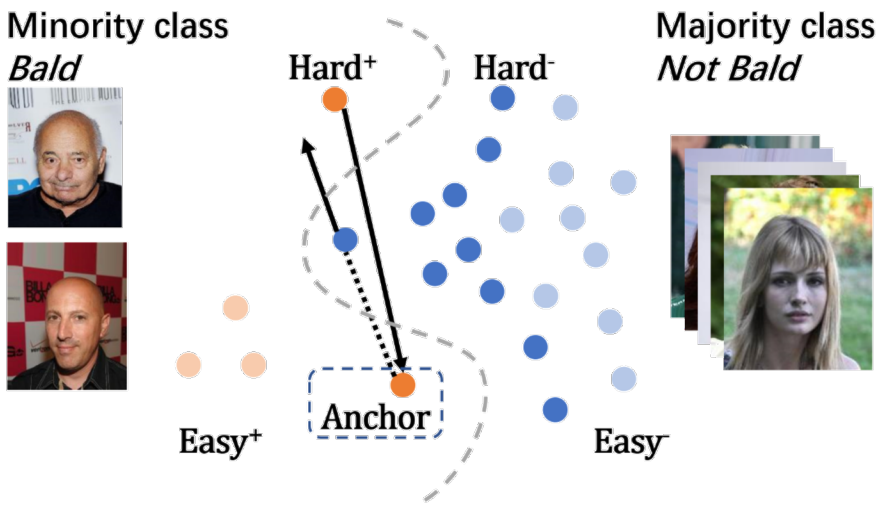


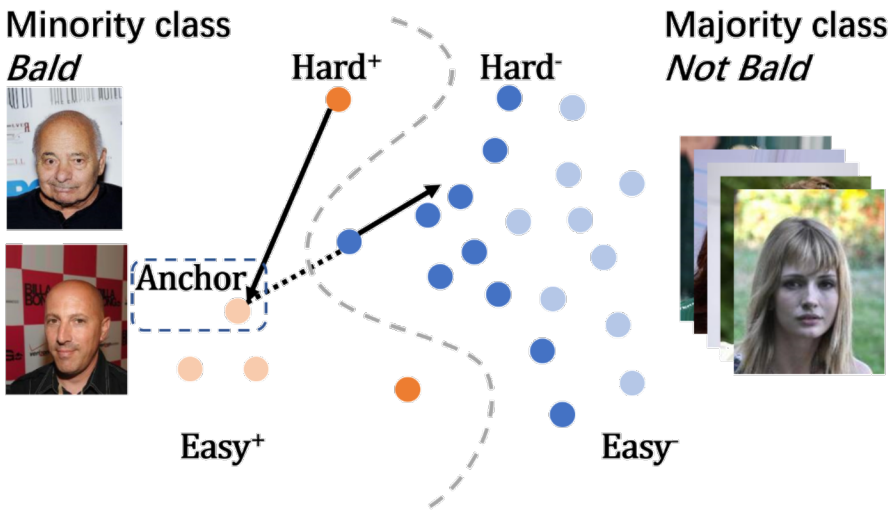
$$\mathcal{L}_{\mathrm{DSL}} = -\frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N_j} w_j * \log\left(p(y_{i,j} = \bar{y}_{i,j}|\mathbf{x}_{i,j})\right) \quad (7)$$

$$w_j = \begin{cases} \frac{D_{target,j}(l)}{D_{current,j}} & \text{if } \frac{D_{target,j}(l)}{D_{current,j}} \geq 1 \\ 0/1 & \text{if } \frac{D_{target,j}(l)}{D_{current,j}} < 1 \end{cases} \quad (8)$$

Dynamic Curriculum Learning for Imbalanced Data Classification. IEEE/CVF International Conference on Computer Vision 2019

DCL: Dynamic Curriculum Learning





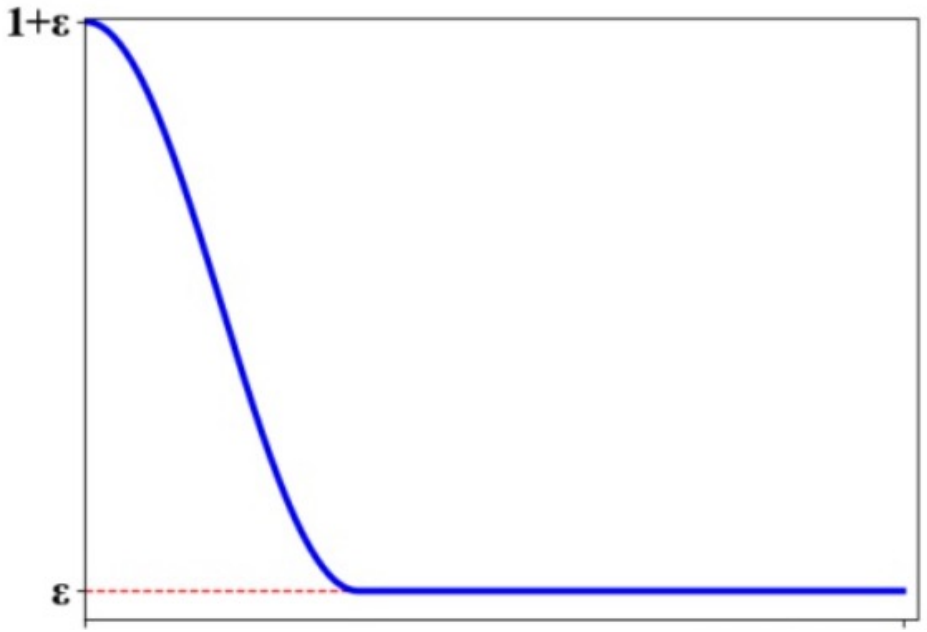
$$\mathcal{L}_{\text{TEA}} = \frac{\sum_T \max\left(0,\ m_j + d(\mathbf{x}_{easy,j}, \mathbf{x}_{+,j}) - d(\mathbf{x}_{easy,j}, \mathbf{x}_{-,j})\right)}{|T|} \tag{10}$$

Dynamic Curriculum Learning for Imbalanced Data Classification. IEEE/CVF International Conference on Computer Vision 2019

DCL: Dynamic Curriculum Learning

$$\mathcal{L}_{\text{DCL}} = \mathcal{L}_{\text{DSL}} + f(l) * \mathcal{L}_{\text{TEA}}$$

$$f(l) = \begin{cases} \frac{1}{2}cos(\frac{l}{L}\pi) + \frac{1}{2} + \epsilon & \text{if } l < pL \\ \epsilon & \text{if } l \geq pL \end{cases}$$



Dynamic Curriculum Learning for Imbalanced Data Classification. IEEE/CVF International Conference on Computer Vision 2019

DCL: Dynamic Curriculum Learning

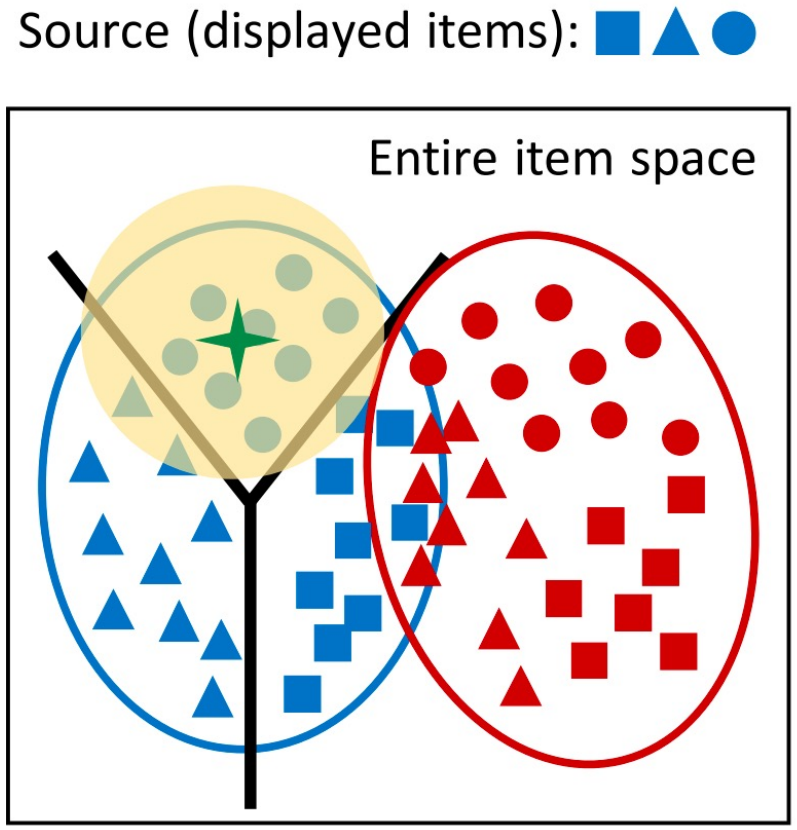
Table 2. Class-balanced Mean Accuracy (mA) for each class (%) and class imbalance level (majority class rate-50%) of each of the attributes on CelebA dataset. The 1st/2nd best results are highlighted in red/blue.

	Attractive	Mouth Open	Smiling	Wear Lipstick	High Cheekbones	Male	Heavy Makeup	Wavy Hair	Oval Face	Pointy Nose	Arched Eyebrows	Black Hair	Big Lips	Big Nose	Young	Straight Hair	Brown Hair	Bags Under Eyes	Wear Earrings	No Beard	Bangs
Imbalance level	1	2	2	3	5	8	11	18	22	22	23	26	26	27	28	29	30	30	31	33	35
DeepID2(CE) [46]	78	89	89	92	84	94	88	73	63	66	77	83	62	73	76	65	79	74	75	88	91
Over-Sampling [9]	77	89	90	92	84	95	87	70	63	67	79	84	61	73	75	66	82	73	76	88	90
Down-Sampling [9]	78	87	90	91	80	90	89	70	58	63	70	80	61	76	80	61	76	71	70	88	88
Cost-Sensitive [16]	78	89	90	91	85	93	89	75	64	65	78	85	61	74	75	67	84	74	76	88	90
Selective-Learning [14]	81	91	92	93	86	97	90	78	66	70	79	87	66	77	83	72	84	79	80	93	94
CRL-I [7]	83	95	93	94	89	96	84	79	66	**73**	80	**90**	**68**	**80**	84	73	86	80	83	94	95
LMLE [19]	**88**	**96**	**99**	**99**	**92**	**99**	**98**	**83**	68	72	79	92	60	**80**	**87**	73	**87**	73	83	**96**	**98**
CLMLE [20]	**90**	**97**	**99**	**98**	**94**	**99**	**98**	**87**	**72**	**78**	**86**	**95**	66	**85**	**90**	**80**	**89**	**82**	**86**	**98**	**99**
DCL (ours)	83	93	93	95	88	98	92	81	**70**	**73**	**82**	89	**69**	**80**	86	**76**	86	**82**	**85**	95	96

	Blond Hair	Bushy Eyebrows	Wear Necklace	Narrow Eyes	5 o'clock Shadow	Receding Hairline	Wear Necktie	Eyeglasses	Rosy Cheeks	Goatee	Chubby	Sideburns	Blurry	Wear Hat	Double Chin	Pale Skin	Gray Hair	Mustache	Bald		**Average**
Imbalance level	35	36	38	38	39	42	43	44	44	44	44	44	45	45	45	46	46	46	48		
DeepID2(CE) [46]	90	78	70	64	85	81	83	92	86	90	81	89	74	90	83	81	90	88	93		**81.17**
Over-Sampling [9]	90	80	71	65	85	82	79	91	90	89	83	90	76	89	84	82	90	90	92		**81.48**
Down-Sampling [9]	85	75	66	61	82	79	80	85	82	85	78	80	68	90	80	78	88	60	79		**77.45**
Cost-Sensitive [16]	89	79	71	65	84	81	82	91	**92**	86	82	90	76	90	84	80	90	88	93		**81.60**
Selective-Learning [14]	93	85	73	**74**	89	**87**	92	97	90	94	**87**	**94**	**86**	96	**89**	92	94	92	95		**85.93**
CRL-I [7]	95	84	**74**	72	90	**87**	88	96	88	96	**87**	92	85	98	**89**	92	95	**94**	**97**		**86.60**
LMLE [19]	**99**	82	59	59	82	76	90	98	78	95	79	88	59	**99**	74	80	91	73	90		**83.83**
CLMLE [20]	**99**	**88**	69	71	**91**	82	**96**	**99**	86	**98**	85	**94**	72	**99**	87	**94**	**96**	82	95		**88.78**
DCL (ours)	95	**87**	**76**	**79**	**93**	**90**	**95**	**99**	**92**	**97**	**93**	**97**	**93**	**99**	**94**	**96**	**99**	**97**	**99**		**89.05**

Dynamic Curriculum Learning for Imbalanced Data Classification. IEEE/CVF International Conference on Computer Vision 2019

ESAM



长尾商品的特征分布和那些热门商品的不一致

ESAM: Discriminative Domain Adaptation with Non-Displayed Items to Improve Long-Tail Performance. SiGIR 2020

ESAM: Attribute correlation alignment (ACA)

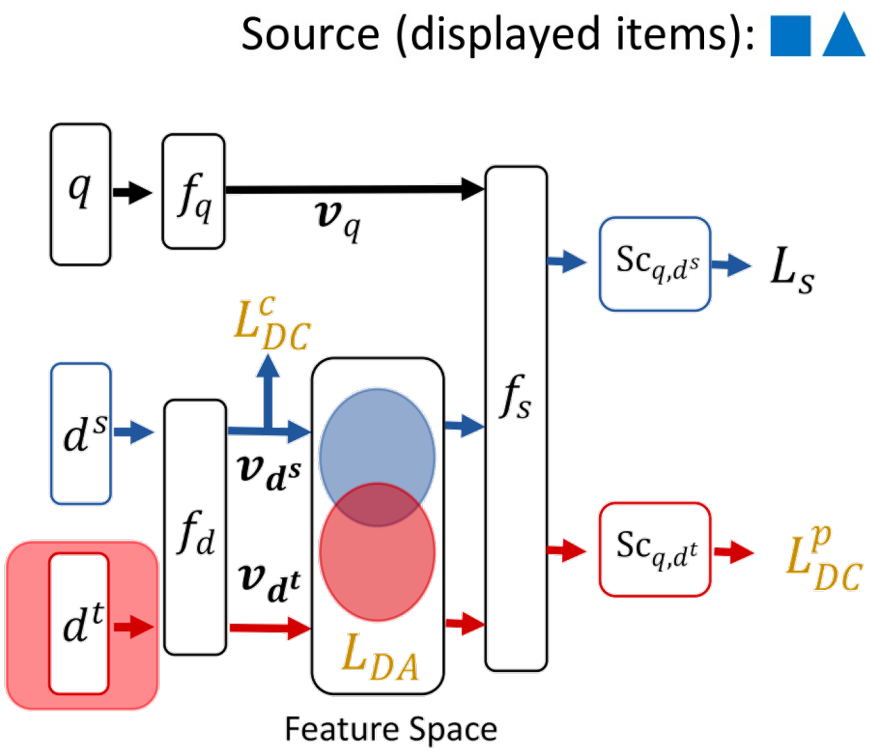


Figure 3: Overview of the ESAM. The red flow represents the non-displayed item stream.

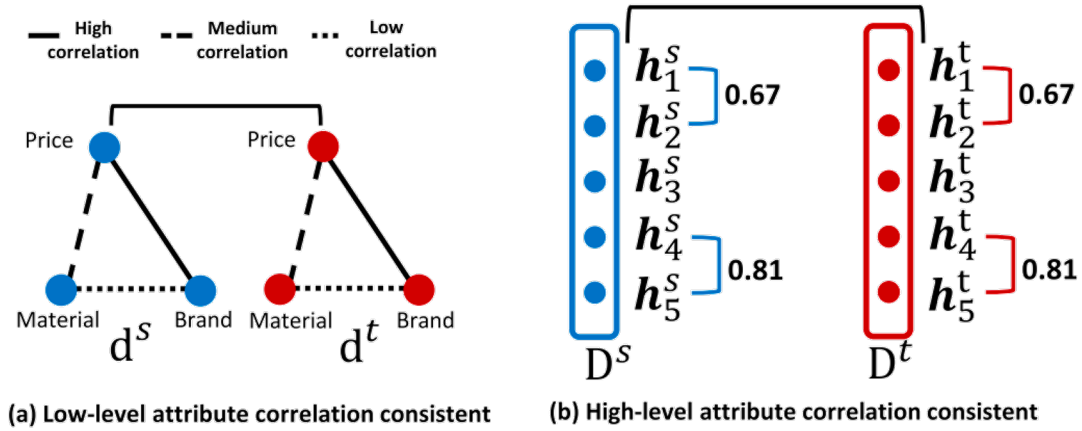
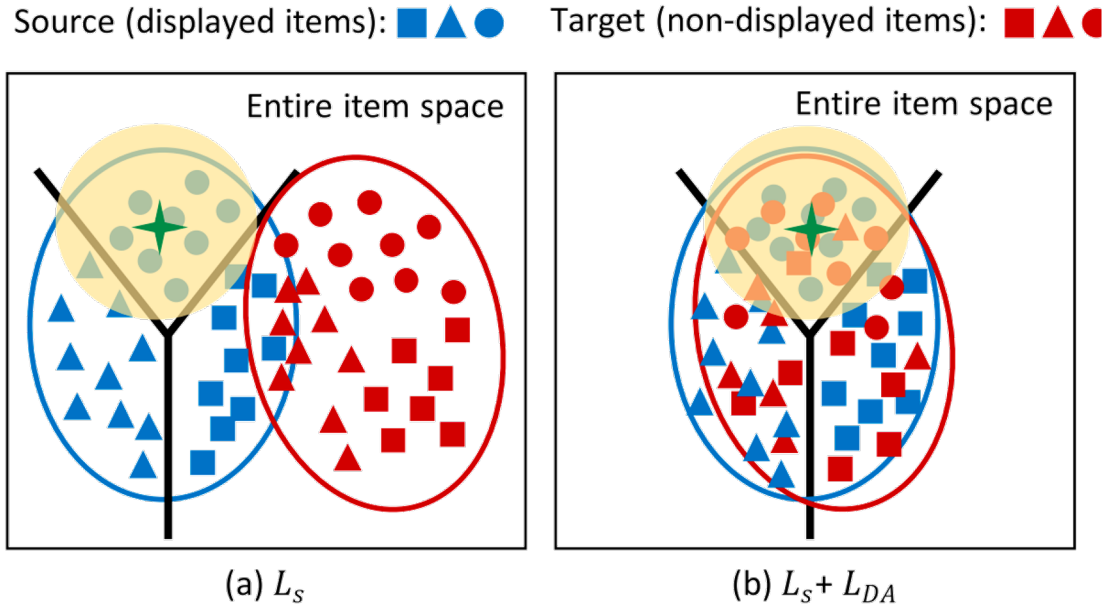


Figure 4: The correlation between low-level item attributes in the source domain is consistent with the target domain. Based on such constraint, we believe that the distributions of source and target domains are aligned when the correlation between high-level attribute vectors of item matrix in the source domain is consistent with the target domain.

ESAM: Attribute correlation alignment (ACA)

$$L_{DA} = \frac{1}{L^2} \sum_{(j,k)} (\boldsymbol{h}_j^{s\top} \boldsymbol{h}_k^s - \boldsymbol{h}_j^{t\top} \boldsymbol{h}_k^t)^2$$

$$= \frac{1}{L^2} \|Cov(\boldsymbol{D}^s) - Cov(\boldsymbol{D}^t)\|_F^2,$$



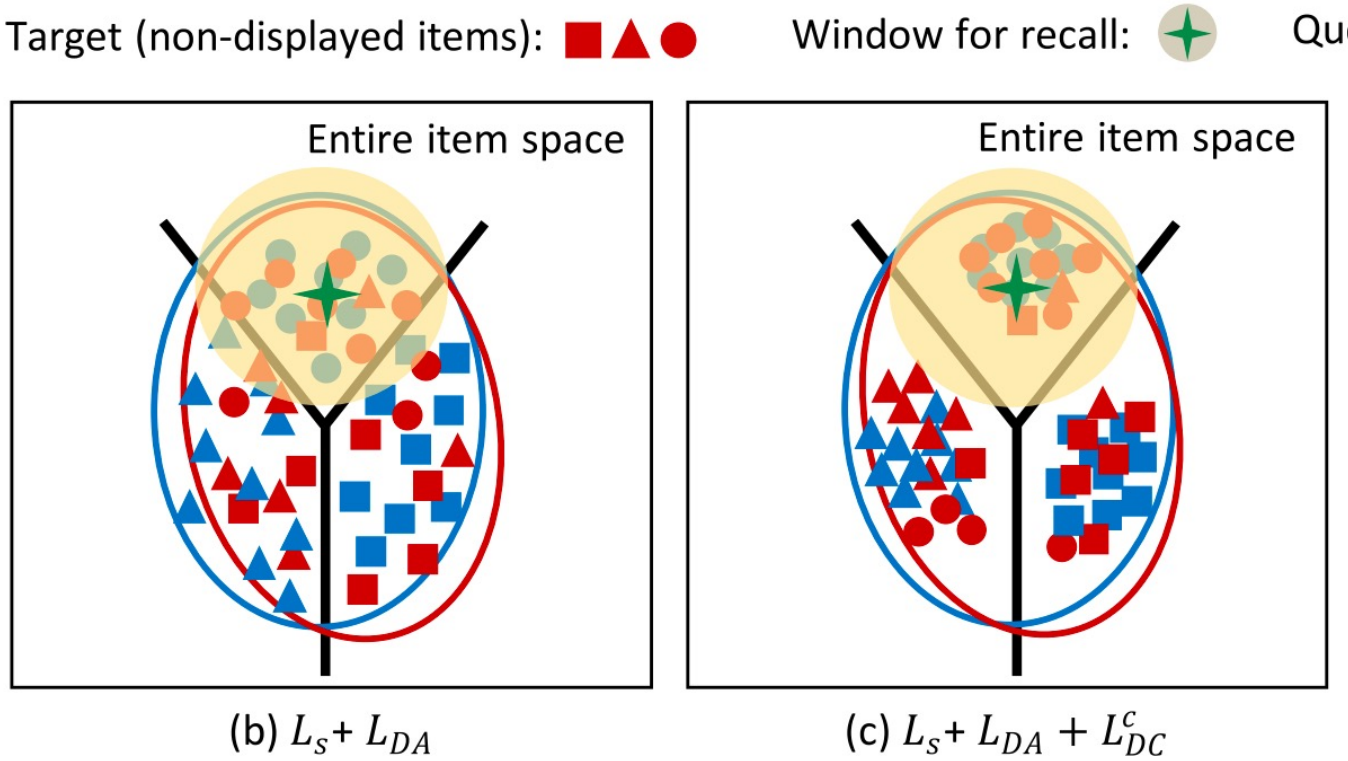
ESAM: Discriminative Domain Adaptation with Non-Displayed Items to Improve Long-Tail Performance. SiGIR 2020

ESAM: Center-Wise Clustering for Source Clustering

$$L_{DC}^{c} = \sum_{j=1}^{n} \max(0, \| \frac{\boldsymbol{v}_{d_j^s}}{\|\boldsymbol{v}_{d_j^s}\|} - \mathbf{c}_q^{y_j^s} \|_2^2 - m_1)$$

$$+ \sum_{k=1}^{n_y} \sum_{u=k+1}^{n_y} \max(0, m_2 - \|\mathbf{c}_q^k - \mathbf{c}_q^u\|_2^2)$$

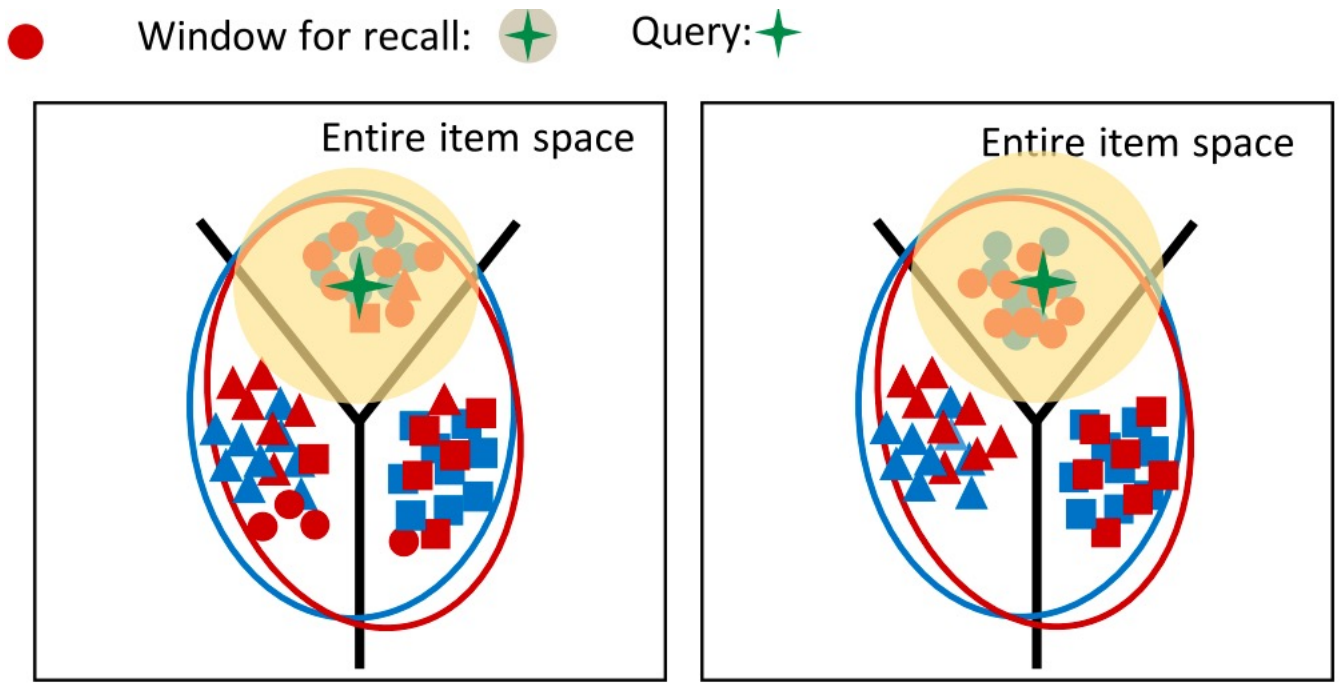
$$\mathbf{c}_q^k = \frac{\sum_{j=1}^{n} (\delta(y_j^s = Y_k) \cdot \frac{\boldsymbol{v}_{d_j^s}}{\|\boldsymbol{v}_{d_j^s}\|})}{\sum_{j=1}^{n} \delta(y_j^s = Y_k)}$$



ESAM: Discriminative Domain Adaptation with Non-Displayed Items to Improve Long-Tail Performance. SiGIR 2020

ESAM: Self-Training for Target Clustering

$$L_{DC}^{p} = -\frac{\sum_{j=1}^{n} \delta(Sc_{q,d_j^t} < p_1 | Sc_{q,d_j^t} > p_2) Sc_{q,d_j^t} \log Sc_{q,d_j^t}}{\sum_{j=1}^{n} \delta(Sc_{q,d_j^t} < p_1 | Sc_{q,j}^{t} > p_2)}, \tag{7}$$



(c) $L_s + L_{DA} + L_{DC}^{c}$ (d) $L_s + L_{DA} + L_{DC}^{c} + L_{DC}^{p}$

ESAM

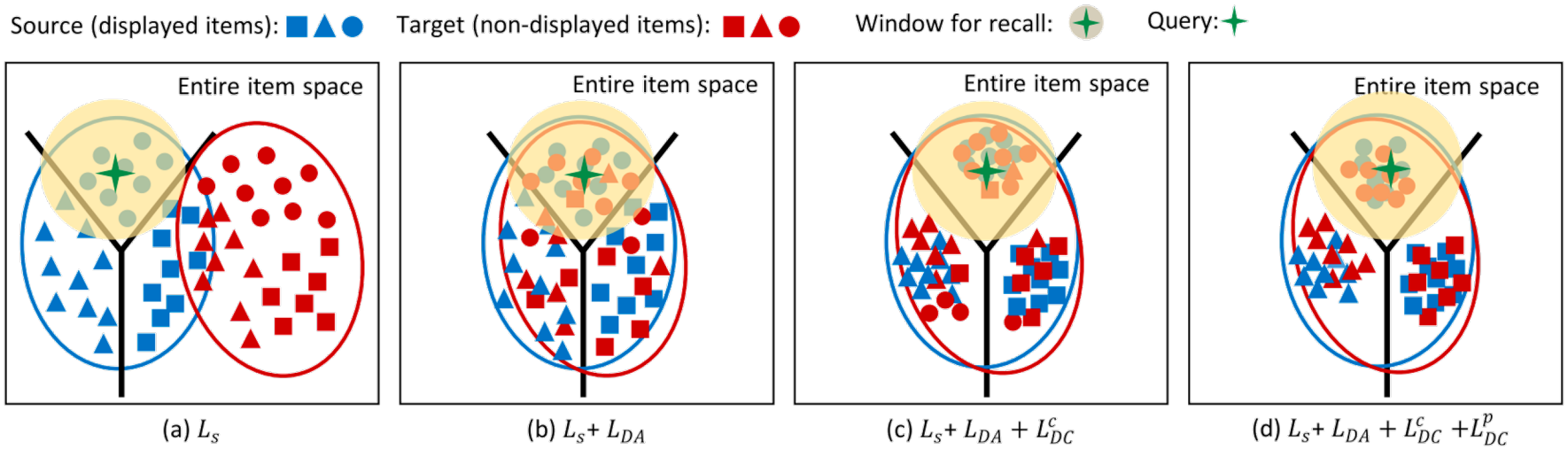


Figure 2: Three shapes represent three feedbacks, the star with a shadow represents a query feature with its retrieval range. (a) Domain shift between displayed and non-displayed item features. (b) Although the domain shift is alleviated, the poor neighborhood relationship makes the model easy to retrieve outliers. (c) The proposed center-wise clustering encourages better manifold structure. However, the class-agnostic ACA may cause negative transfer, *e.g.*, some target circles are aligned to source triangles. (d) ESAM extracts the best item feature representations.

ESAM: Discriminative Domain Adaptation with Non-Displayed Items to Improve Long-Tail Performance. SiGIR 2020

ESAM

Table 3: Performance comparison between the methods without and with ESAM on MovieLens-1M dataset. "Hot" represents hot items in the test set, "Long-tail" represents long-tail items in the test set, and "Entire" represents all items in the test set. The best results are highlighted in boldface. The improvements over the baseliens are statistically significance at 0.05 level.

Group	Method	w/o ESAM									w/ ESAM								
		NDCG@20			Recall@20			MAP			NDCG@20			Recall@20			MAP		
		Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire
Single Domain	NeuralMF [19]	0.2296	0.0508	0.2017	0.2660	0.0732	0.2453	0.1953	0.0486	0.1813	**0.2431**	**0.0812**	**0.2238**	**0.2753**	**0.1005**	**0.2644**	**0.2014**	**0.0674**	**0.1962**
	YoutubeNet [12]	0.2519	0.0983	0.2385	0.2976	0.1142	0.2691	0.2138	0.0874	0.2057	**0.2671**	**0.1264**	**0.2613**	**0.3125**	**0.1369**	**0.2947**	**0.2282**	**0.1108**	**0.2270**
	RALM [29]	0.2658	0.1237	0.2584	0.3081	0.1294	0.2883	0.2259	0.1037	0.2236	**0.2784**	**0.1493**	**0.2738**	**0.3249**	**0.1455**	**0.3057**	**0.2376**	**0.1271**	**0.2412**
	BST[8]	0.2816	0.1371	0.2835	0.3292	0.1373	0.3104	0.2429	0.1230	0.2481	**0.2935**	**0.1627**	**0.2984**	**0.3453**	**0.1516**	**0.3295**	**0.2579**	**0.1451**	**0.2608**
Domain Adaptation	DALRF [38]	0.2731	0.1503	0.2467	0.3092	0.1424	0.2953	0.2376	0.1431	0.2359	**0.2816**	**0.1584**	**0.2503**	**0.3158**	**0.1507**	**0.3012**	**0.2431**	**0.1479**	**0.2418**
	DARec [44]	0.2748	0.1565	0.2497	0.3124	0.1483	0.2996	0.2418	0.1488	0.2401	**0.2824**	**0.1631**	**0.2614**	**0.3182**	**0.1549**	**0.3079**	**0.2503**	**0.1579**	**0.2495**

Table 4: Performance comparison between methods without and with ESAM on CIKM Cup 2016 dataset. The best results are highlighted in boldface. The improvements over the baseliens are statistically significance at 0.05 level.

Group	Method	w/o ESAM									w/ ESAM								
		NDCG@20			Recall@20			MAP			NDCG@20			Recall@20			MAP		
		Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire	Hot	Long-tail	Entire
Single Domain	NeuralMF [19]	0.1840	0.0521	0.1296	0.3145	0.0947	0.2265	0.1596	0.0559	0.1158	**0.2043**	**0.1085**	**0.1543**	**0.3420**	**0.1495**	**0.2652**	**0.1718**	**0.0857**	**0.1409**
	YoutubeNet [12]	0.2196	0.0942	0.1572	0.3481	0.1422	0.2594	0.1928	0.0974	0.1437	**0.2347**	**0.1295**	**0.1821**	**0.3702**	**0.1735**	**0.2859**	**0.2139**	**0.1266**	**0.1703**
	RALM [29]	0.2341	0.1195	0.1794	0.3592	0.1583	0.2739	0.2164	0.1219	0.1682	**0.2457**	**0.1408**	**0.2049**	**0.3796**	**0.1841**	**0.2985**	**0.2295**	**0.1451**	**0.1879**
	BST[8]	0.2483	0.1392	0.2067	0.3824	0.1793	0.3021	0.2319	0.1436	0.1928	**0.2607**	**0.1586**	**0.2241**	**0.3976**	**0.2058**	**0.3249**	**0.2465**	**0.1620**	**0.2194**
Domain Adaptation	DALRF [38]	0.2213	0.1456	0.1861	0.3684	0.1921	0.2973	0.2024	0.1583	0.1895	**0.2362**	**0.1578**	**0.2014**	**0.3805**	**0.2052**	**0.3097**	**0.2152**	**0.1735**	**0.2008**
	DARec [44]	0.2259	0.1502	0.1923	0.3705	0.1982	0.3018	0.2079	0.1604	0.1936	**0.2348**	**0.1683**	**0.2091**	**0.3859**	**0.2053**	**0.3084**	**0.2137**	**0.1762**	**0.2041**

ESAM: Discriminative Domain Adaptation with Non-Displayed Items to Improve Long-Tail Performance. SiGIR 2020

SGL

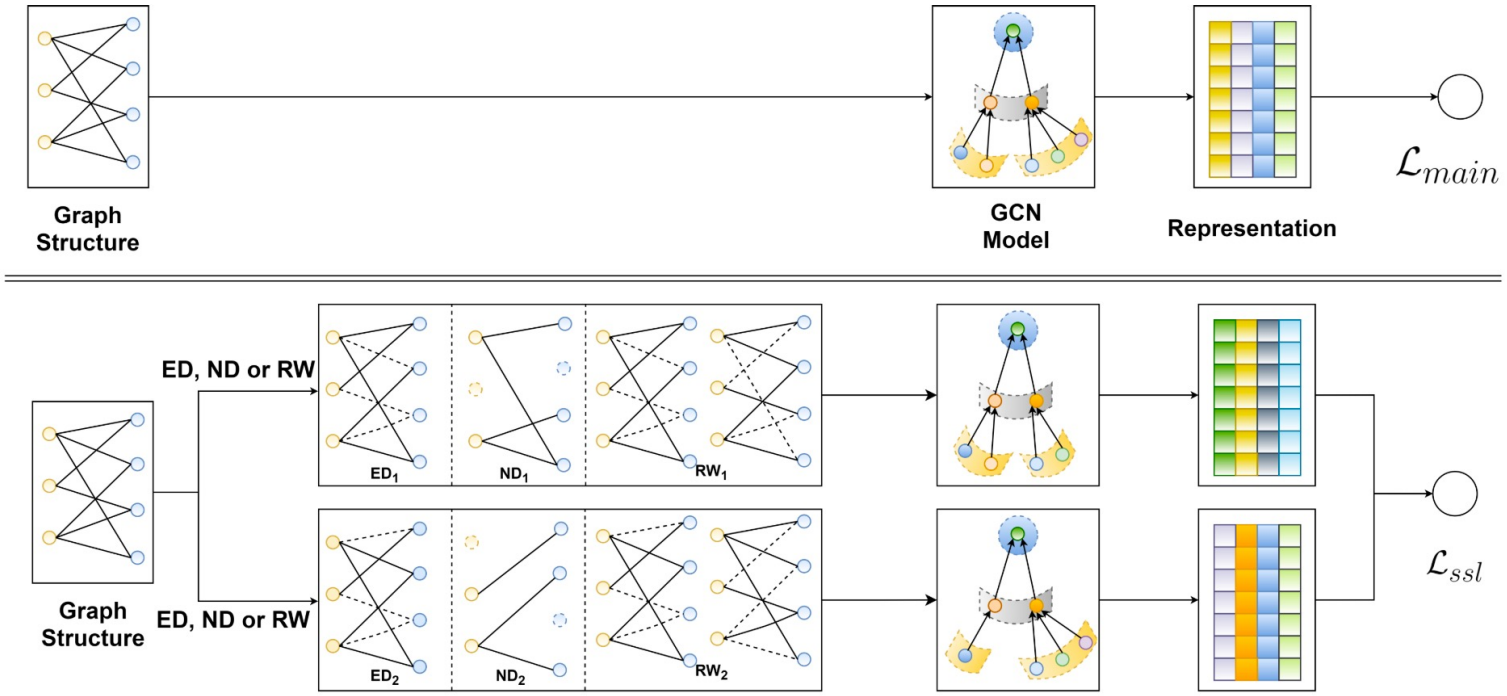


Figure 1: The overall system framework of SGL. The upper layer illustrates the working flow of the main supervised learning task while the bottom layer shows the working flows of SSL task with augmentation on graph structure.

Self-supervised Graph Learning for Recommendation. SiGIR 2021

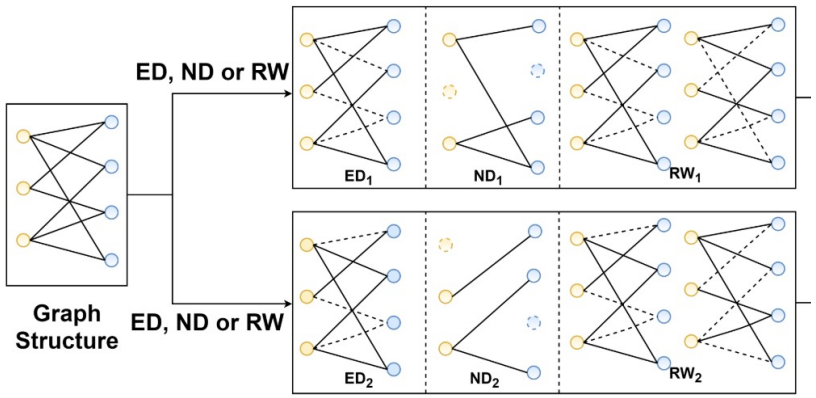
SGL (Data Augmentation on Graph Structure)

$$\mathbf{Z}^{(l)} = H(\mathbf{Z}^{(l-1)}, \mathcal{G})$$

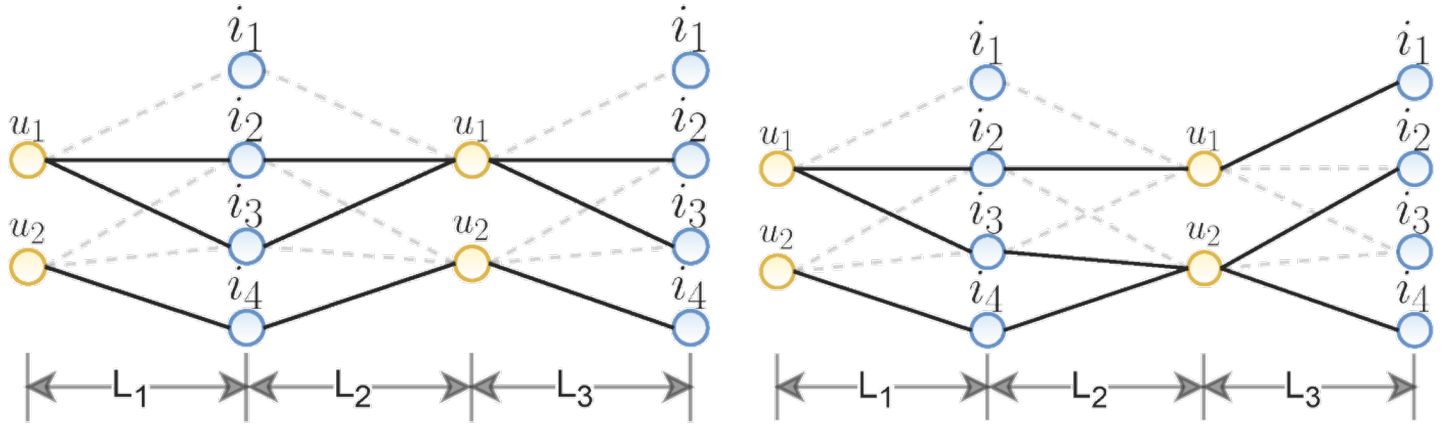
Dropout

$$\mathbf{Z}_1^{(l)} = H(\mathbf{Z}_1^{(l-1)}, s_1(\mathcal{G}))$$

$$\mathbf{Z}_2^{(l)} = H(\mathbf{Z}_2^{(l-1)}, s_2(\mathcal{G}))$$



- Node Dropout：一定概率丢掉图中节点及其邻接的边
- Edge Dropout：一定概率丢掉图中的边
- Random Walk：不考虑原始图，随机重新生成图上哪里有边



(a) Edge Dropout　　(b) Random Walk

Self-supervised Graph Learning for Recommendation. SiGIR 2021

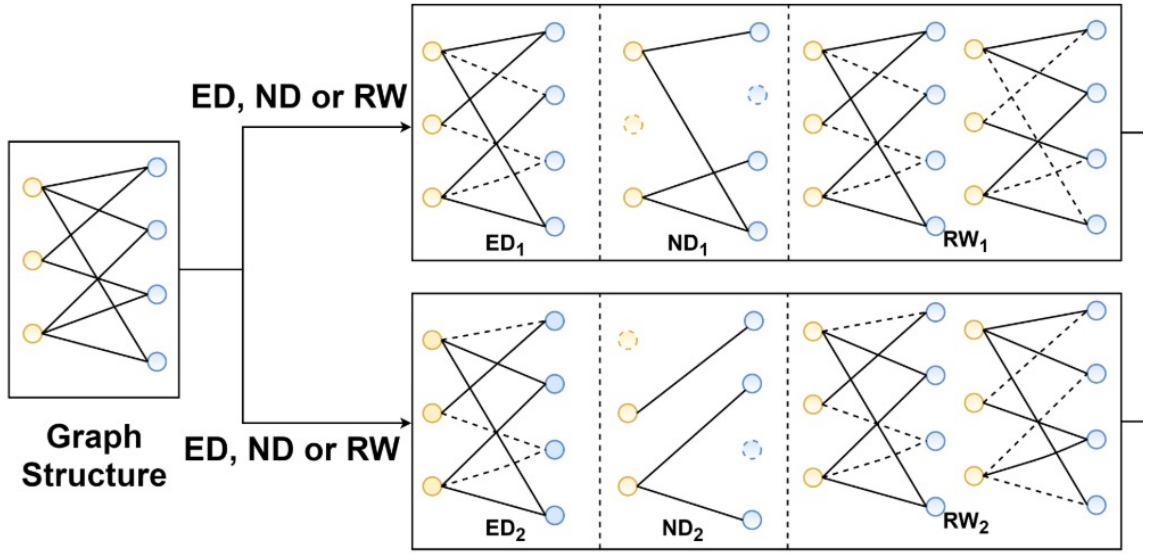
SGL (Contrastive Learning)

同一节点不同视图下的 embedding 对

$$\{(z'_u, z''_u)|u \in \mathcal{U}\}$$

不同节点不同视图下的 embedding 对

$$\{(\mathbf{z}'_u, \mathbf{z}''_v)|u, v \in \mathcal{U}, u \neq v\}$$



$$\mathcal{L}_{ssl}^{user} = \sum_{u \in \mathcal{U}} -\log \frac{\exp(s(\mathbf{z}'_u, \mathbf{z}''_u)/\tau)}{\sum_{v \in \mathcal{U}} \exp(s(\mathbf{z}'_u, \mathbf{z}''_v)/\tau)}$$

$$\mathcal{L}_{ssl} = \mathcal{L}_{ssl}^{user} + \mathcal{L}_{ssl}^{item}$$

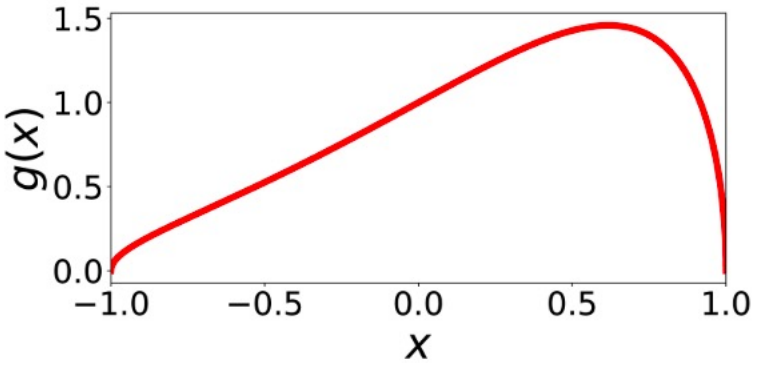
$$\mathcal{L} = \mathcal{L}_{main} + \lambda_1 \mathcal{L}_{ssl} + \lambda_2 \|\Theta\|_2^2$$

Self-supervised Graph Learning for Recommendation. SiGIR 2021

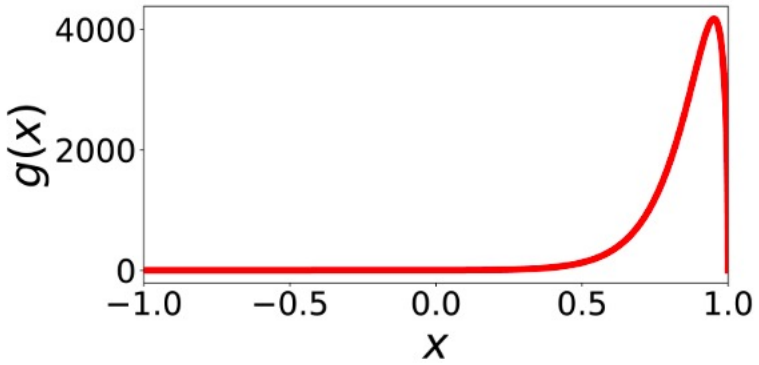
SGL (Theoretical Analyses of SGL)

$$\mathcal{L}_{ssl}^{user} = \sum_{u \in \mathcal{U}} -\log \frac{\exp(s(\mathbf{z}'_u, \mathbf{z}''_u)/\tau)}{\sum_{v \in \mathcal{U}} \exp(s(\mathbf{z}'_u, \mathbf{z}''_v)/\tau)}$$

$$\frac{\partial \mathcal{L}_{ssl}^{user}(u)}{\partial z'_u} \qquad \|c(v)\|_2 \propto \sqrt{1 - (s'^{T}_u s''_v)^2} \exp(s'^{T}_u s''_v/\tau) \qquad g(x) = \sqrt{1-x^2} \exp\left(\frac{x}{\tau}\right)$$



(a) $g(x), \tau = 1$



(b) $g(x), \tau = 0.1$

SGL (Evaluation)

Table 4: Overall Performance Comparison.

Dataset	Yelp2018		Amazon-Book		Alibaba-iFashion	
Method	Recall	NDCG	Recall	NDCG	Recall	NDCG
NGCF	0.0579	0.0477	0.0344	0.0263	0.1043	0.0486
LightGCN	0.0639	0.0525	0.0411	0.0315	0.1078	0.0507
Mult-VAE	0.0584	0.0450	0.0407	0.0315	0.1041	0.0497
DNN+SSL	0.0483	0.0382	0.0438	0.0337	0.0712	0.0325
SGL-ED	**0.0675**	**0.0555**	**0.0478**	**0.0379**	**0.1126**	**0.0538**
%Improv.	5.63%	5.71%	9.13%	12.46%	4.45%	6.11%
p-value	5.92e-8	1.89e-8	5.07e-10	3.63e-10	3.34e-8	4.68e-10

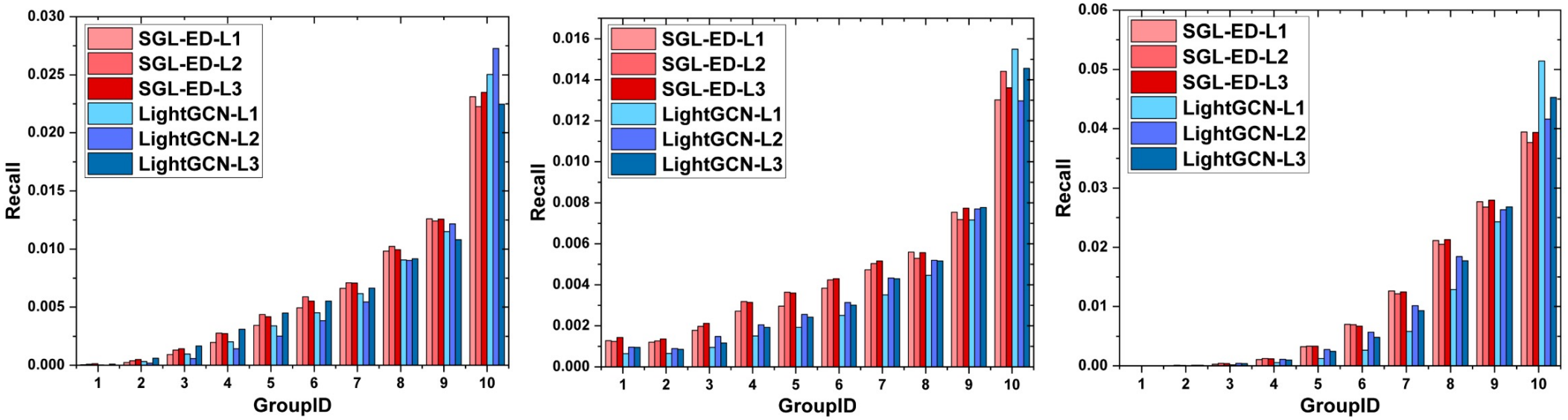
Self-supervised Graph Learning for Recommendation. SiGIR 2021

SGL (Ablation Study)

Table 3: Performance comparison with LightGCN at different layers. The performance of LightGCN on Yelp2018 and Amazon-Book are copied from its original paper. The percentage in brackets denote the relative performance improvement over LightGCN. The bold indicates the best result.

Datase		Yelp2018		Amazon-Book		Alibaba-iFashion	
#Layer	Method	Recall	NDCG	Recall	NDCG	Recall	NDCG
1 Layer	LightGCN	0.0631	0.0515	0.0384	0.0298	0.0990	0.0454
	SGL-ND	**0.0643(+1.9%)**	**0.0529(+2.7%)**	0.0432(+12.5%)	0.0334(+12.1%)	**0.1133(+14.4%)**	**0.0539(+18.7%)**
	SGL-ED	0.0637(+1.0%)	0.0526(+2.1%)	**0.0451(+17.4%)**	**0.0353(+18.5%)**	0.1125(+13.6%)	0.0536(+18.1%)
	SGL-RW	0.0637(+1.0%)	0.0526(+2.1%)	**0.0451(+17.4%)**	**0.0353(+18.5%)**	0.1125(+13.6%)	0.0536(+18.1%)
2 Layers	LightGCN	0.0622	0.0504	0.0411	0.0315	0.1066	0.0505
	SGL-ND	0.0658(+5.8%)	0.0538(+6.7%)	0.0427(+3.9%)	0.0335(+6.3%)	**0.1106(+3.8%)**	**0.0526(+4.2%)**
	SGL-ED	**0.0668(+7.4%)**	**0.0549(+8.9%)**	**0.0468(+13.9%)**	**0.0371(+17.8%)**	0.1091(+2.3%)	0.0520(+3.0%)
	SGL-RW	0.0644(+3.5%)	0.0530(+5.2%)	0.0453(+10.2%)	0.0358(+13.7%)	0.1091(+2.3%)	0.0521(+3.2%)
3 Layers	LightGCN	0.0639	0.0525	0.0410	0.0318	0.1078	0.0507
	SGL-ND	0.0644(+0.8%)	0.0528(+0.6%)	0.0440(+7.3%)	0.0346(+8.8%)	0.1126(4.5%)	0.0536(+5.7%)
	SGL-ED	**0.0675(+5.6%)**	**0.0555(+5.7%)**	**0.0478(+16.6%)**	**0.0379(+19.2%)**	0.1126(+4.5%)	0.0538(+6.1%)
	SGL-RW	0.0667(+4.4%)	0.0547(+4.2%)	0.0457(+11.5%)	0.0356(+12.0%)	**0.1139(+5.7%)**	**0.0539(+6.3%)**

SGL (Long-tail Recommendation)



(a) Yelp2018 (b) Amazon-Book (c) Alibaba-iFashion

Figure 4: Performance comparison over different item groups between SGL-ED and LightGCN. The suffix in the legend indicates the number of GCN layers.

Focal Loss

$$cross\_entropy = -\sum_{k=1}^{N}(y_t \times \log(p_t))$$

$$focal\_loss = -\sum_{k=1}^{N}(y_t \times (1-p_t)^{\gamma} \log(p_t))$$

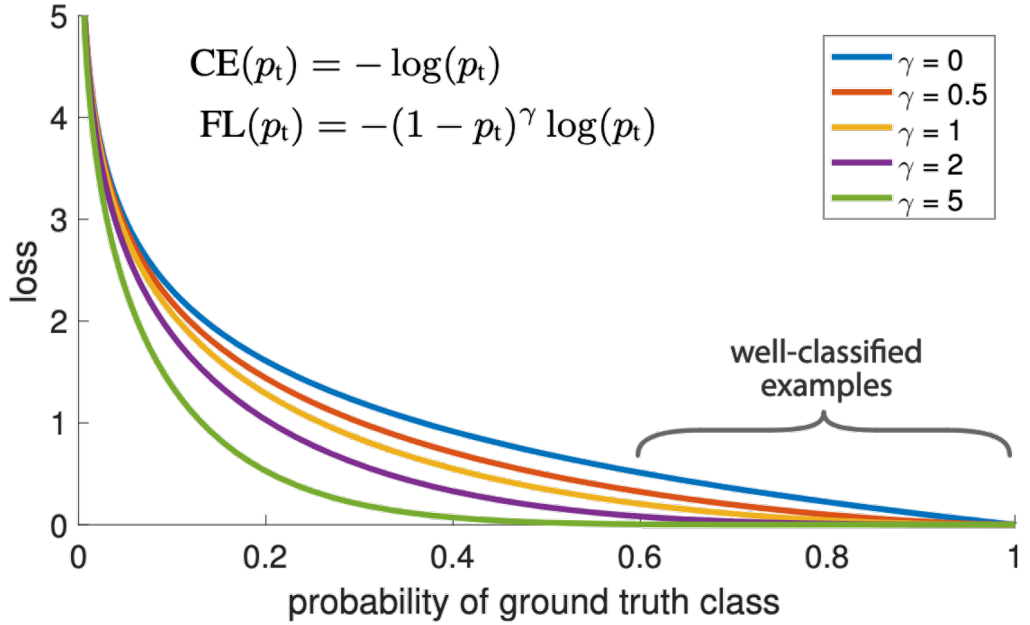
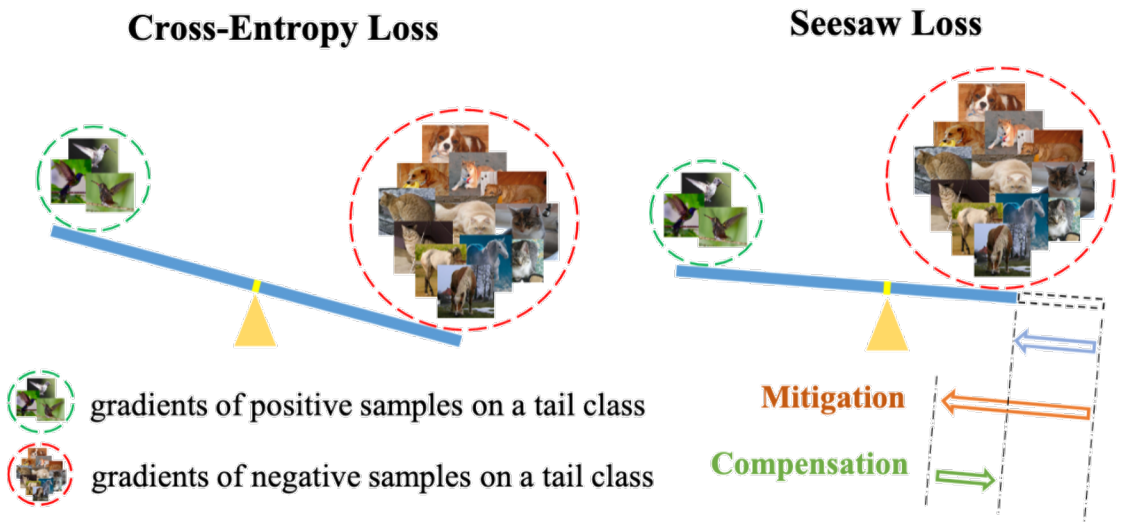
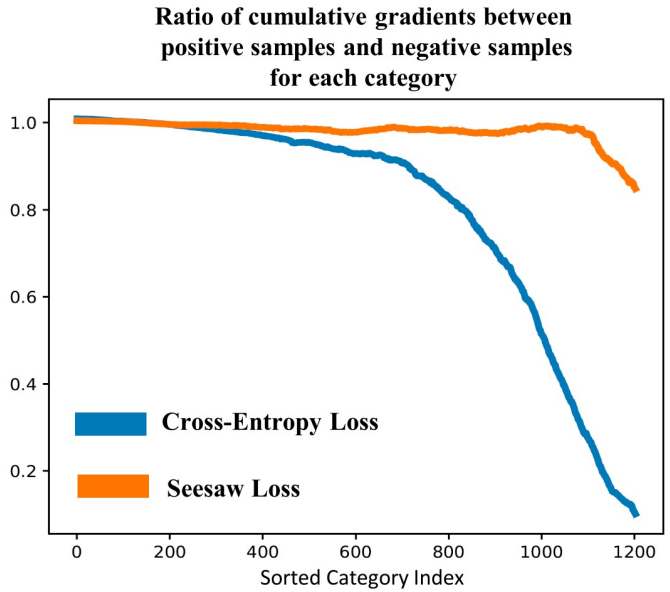


Figure 1. We propose a novel loss we term the *Focal Loss* that adds a factor $(1-p_t)^{\gamma}$ to the standard cross entropy criterion. Setting $\gamma > 0$ reduces the relative loss for well-classified examples ($p_t > .5$), putting more focus on hard, misclassified examples. As our experiments will demonstrate, the proposed focal loss enables training highly accurate dense object detectors in the presence of vast numbers of easy background examples.

Focal Loss for Dense Object Detection. ICCV 2017

Seesaw Loss





Seesaw Loss for Long-Tailed Instance Segmentation. LVIS Challenge 2020 of team MMDet

Seesaw Loss

$$L_{seesaw}(\mathbf{z}) = -\sum_{i=1}^{C} y_i \log(\widehat{\sigma}_i),$$

$$\text{with } \widehat{\sigma}_i = \frac{e^{z_i}}{\sum_{j \neq i}^{C} \mathcal{S}_{ij} e^{z_j} + e^{z_i}}.$$

$$\mathcal{S}_{ij} = \mathcal{M}_{ij} \cdot \mathcal{C}_{ij}.$$

M：缓解尾部类别过量的负样本梯度

C：补充错误分类样本的惩罚

$$\frac{\partial L_{seesaw}(\mathbf{z})}{\partial z_j} = \mathcal{S}_{ij} \frac{e^{z_j}}{e^{z_i}} \widehat{\sigma}_i.$$

Seesaw Loss for Long-Tailed Instance Segmentation. LVIS Challenge 2020 of team MMDet

Seesaw Loss

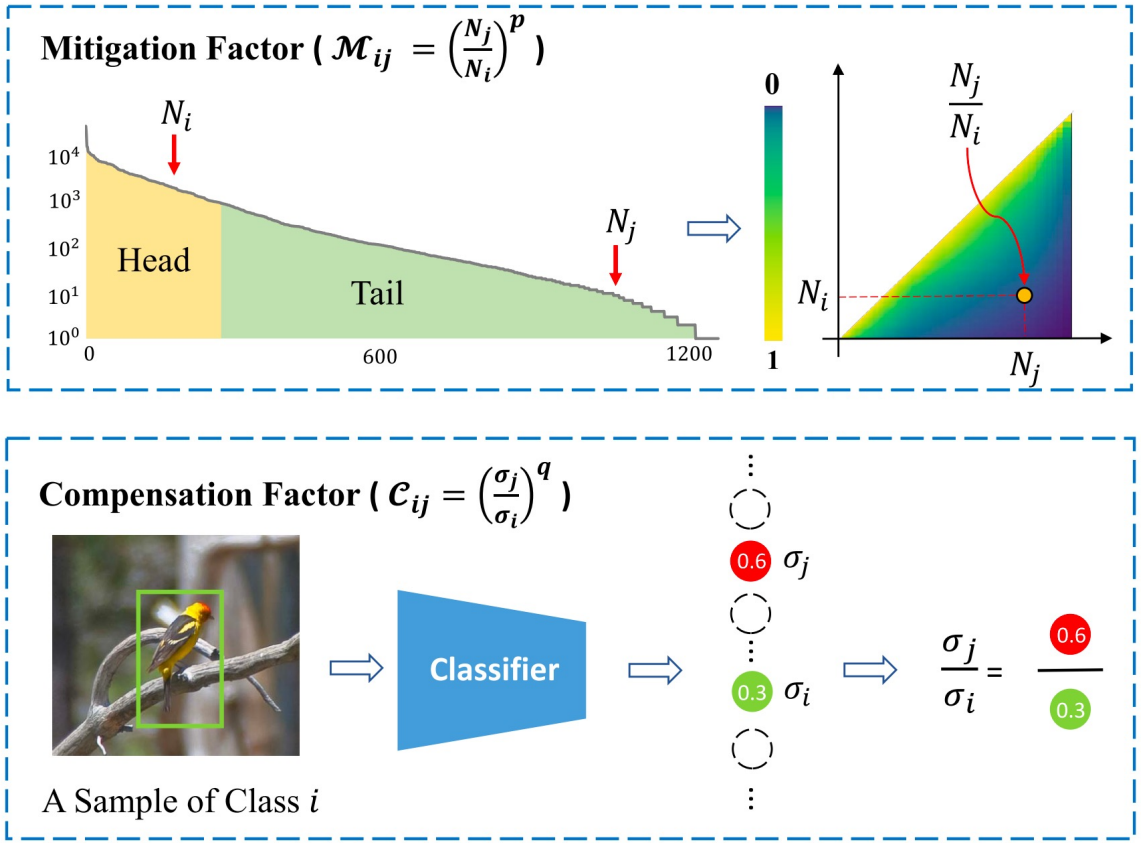
$$\mathcal{S}_{ij} = \mathcal{M}_{ij} \cdot \mathcal{C}_{ij}.$$

M：缓解尾部类别过量的负样本梯度

C：补充错误分类样本的惩罚

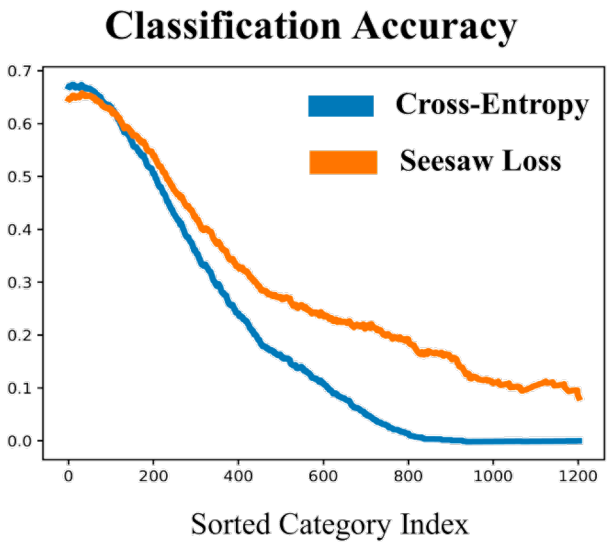
$$\mathcal{M}_{ij} = \begin{cases} 1, & \text{if } N_i \leq N_j \\ \left(\frac{N_j}{N_i}\right)^p, & \text{if } N_i > N_j \end{cases}$$

$$\mathcal{C}_{ij} = \begin{cases} 1, & \text{if } \sigma_j \leq \sigma_i \\ \left(\frac{\sigma_j}{\sigma_i}\right)^q, & \text{if } \sigma_j > \sigma_i \end{cases}$$



Seesaw Loss for Long-Tailed Instance Segmentation. LVIS Challenge 2020 of team MMDet

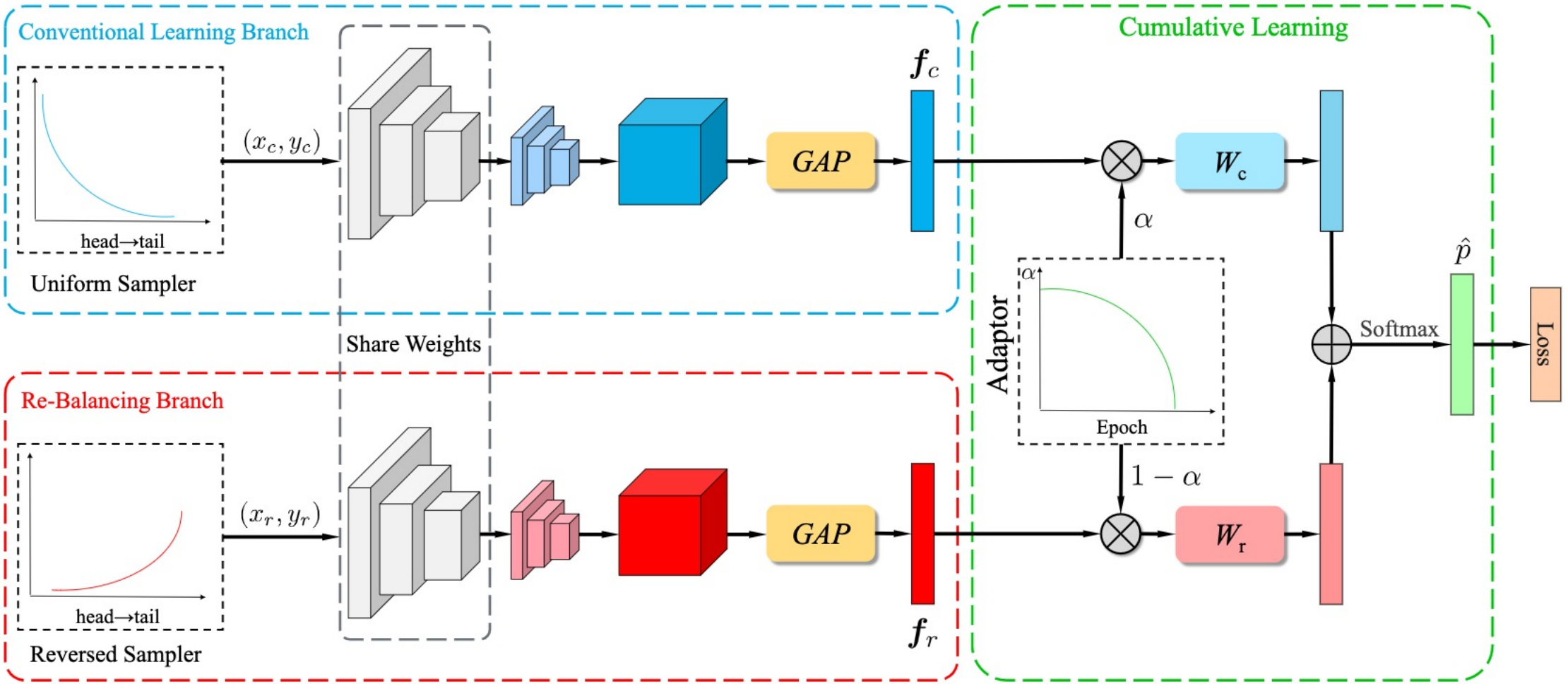
Seesaw Loss



Framework	Sampler	Loss	Split	AP	AP_r	AP_c	AP_f	AP^{box}
Mask R-CNN [17]	Random	Cross-Entropy (CE)	val	20.6	0.8	19.3	30.7	21.7
		Equalization Loss (EQL) [44]		22.7	3.7	23.3	30.4	24.0
		Balanced Group Softmax (BAGS) [26]		25.6	17.3	25.0	30.1	26.4
		Seesaw Loss		26.6	18.1	25.8	31.2	27.4
		Seesaw Loss + Norm Mask		**27.1**	18.7	26.3	31.7	27.4
Mask R-CNN [17]	RFS [14]	Cross-Entropy (CE)	val	25.5	16.6	24.5	30.6	26.6
		Equalization Loss (EQL) [44]		26.2	17.0	26.2	30.2	27.6
		Balanced Group Softmax (BAGS) [26]		25.8	16.5	25.7	30.1	26.5
		Seesaw Loss (Ours)		27.6	20.6	27.3	31.1	28.9
		Seesaw Loss + Norm Mask (Ours)		**28.1**	20.0	28.0	31.9	28.9
Cascade Mask R-CNN [1]	Random	Cross-Entropy (CE)	val	22.6	2.4	22	32.2	25.5
		Equalization Loss (EQL) [44]		24.3	5.1	25.3	31.7	27.3
		Balanced Group Softmax (BAGS) [26]		27.9	19.6	27.7	31.6	31.5
		Seesaw Loss (Ours)		29.0	21.1	28.6	33.0	32.8
		Seesaw Loss + Norm Mask (Ours)		**29.6**	20.3	29.3	34.0	32.8
Cascade Mask R-CNN [1]	RFS [14]	Cross-Entropy (CE)	val	27.0	16.6	26.7	32.0	30.3
		Equalization Loss (EQL) [44]		27.1	17.0	27.2	31.4	30.4
		Balanced Group Softmax (BAGS) [26]		27.0	16.9	26.9	31.7	30.2
		Seesaw Loss (Ours)		29.3	21.7	29.2	32.8	32.8
		Seesaw Loss + Norm Mask (Ours)		**30.1**	21.4	30.0	33.9	32.8
Mask R-CNN [17]	RFS [14]	Cross-Entropy (CE)	test-dev	25.1	13.0	24.8	30.8	-
		Seesaw Loss + Norm Mask (Ours)		27.9	20.3	27.1	32.2	-
Cascade Mask R-CNN [1]	RFS [14]	Cross-Entropy (CE)	test-dev	26.4	15.5	25.5	32.3	-
		Seesaw Loss + Norm Mask (Ours)		**30.0**	23.0	29.3	34.1	-

Seesaw Loss for Long-Tailed Instance Segmentation. LVIS Challenge 2020 of team MMDet

BBN



BBN: Bilateral-Branch Network with Cumulative Learning for Long-Tailed Visual Recognition. CVPR 2020

Conclusion

- Re-sampling（很多工作当作 pipeline 的一个组件使用）
  - DCL
- Re-weighting（迁移方便，实现方便）
  - TailNet
  - Focal loss
  - Seesaw loss
- Transfer learning（贴近数据分布，需求更加可控）
  - ESAM
- Metric learning trips?（需设计巧妙的方式）
  - DCL
  - SGL

THANKS