# Supplementary Material: Long-tailed Instance Segmentation using Gumbel Optimized Loss

In the supplementary material we discuss implementation details of Gumbel activation, and we show additional experiments on long-tailed instance segmentation. In Section 1, we discuss implementation details and visualizations of Gumbel activation; in Section 2, we show detailed ablation study of *GOL* and its application to larger models.

## 1 Gumbel activation

### 1.1 Weights and biases initialization

Gumbel activation has exponential positive gradients, making it difficult to initialize due to arithmetic errors caused by the gradient overflow. For this reason, one should initialize the bias and weight terms of the classification layer with values that will produce small initial gradient. First, all weight terms $W^T$ are initialized to a small value of 0.001, which will result in that all $q_i = W^T z + b \approx b$, then the total gradient will be:

$$\nabla H(\eta_\gamma(q), y) \approx -\exp(-b) + (C-1)\frac{\exp(-b)}{\exp(\exp(-b)) - 1} \tag{1}$$

where $C$ is the total number of classes in the dataset. As the total gradient should be zero initially, we have:

$$\begin{aligned} \nabla H(\eta_\gamma(q), y) &= 0 \\ (C-1)\frac{\exp(-b)}{\exp(\exp(-b)) - 1} &= \exp(-b) \\ b &= -\log(\log(C) \end{aligned} \tag{2}$$

For the case of LVIS dataset that has 1,203 classes plus one for the background, we set the weights $W^T$ equal to 0.001 and the bias equal to $-\log(\log(1204) \approx -2$. These values produce small initial gradients and they prevent gradient overflow.

### 1.2 Temperature in Gumbel activation

We have also studied the choice of non Standard Gumbel activation, as shown in Figure 1.i, for different choices of temperature $\sigma$:

$$\eta_\gamma(q_i; \sigma) = \exp(-\exp(-\frac{q_i}{\sigma})) \tag{3}$$

We observe that, choosing a larger temperature flattens Gumbel activation curve, while choosing a smaller temperature steepens the curve. Gumbel activation has a double exponent as shown in Eq. 3, which makes it difficult to select values of $\sigma$ due to arithmetic instability. In our case, we choose values $[0.8, 0.9, 1.0, 1.1, 1.2]$ and we observe that the best choice is $\sigma = 1$ as it has better overall $AP$ and $AP^r$ as shown in 1.ii.

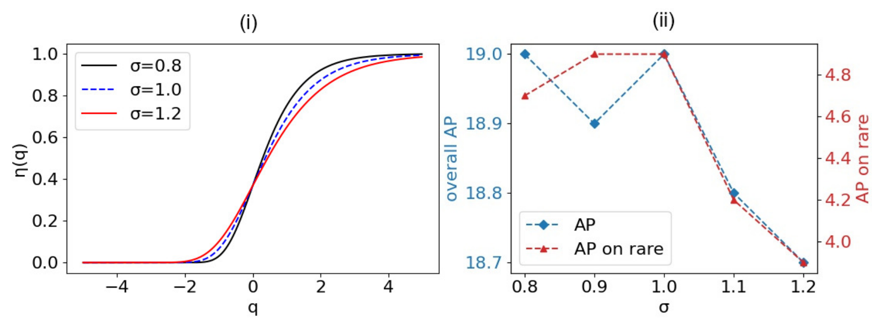


**Fig. 1.** (i): Gumbel activation using different temperature $\sigma$. Selecting a larger $\sigma$ flattens the curve, while selecting a smaller $\sigma$ makes the curve steeper. (ii): Performance of MaskRCNN-R50 on LVISv1 using training schedule 1x and random sampler, for different choices of temperature $\sigma$. The best performance is observed for $\sigma = 1.0$.

### 1.3 Gumbel activation and cut-off error

Gumbel activation has a double exponent, as shown in Eq. 3, this makes it numerically unstable for large inputs and hinders training. For this reason, we tested different ranges of values and decided to clip the input space to be within the range of $[-4, 10]$. Using this range of values the cut-off error is *e-5* and training commences without overflow errors. In the future, we will develop a solution that prevents numerical instability, so that we do not have to clip the input space.

### 1.4 Average Positive Gradient

We visualize the average positive gradient $g$, each category receives during training for 12 epochs using MaskRCNN. We use logarithmic scale to measure $g$ in $dB$ because the average gradient is small, especially for rare categories. As Figure 2 indicates, using Gumbel activation, the positive gradient is on average $7dB$ larger than the case of using Sigmoid, while for the case of rare categories, Gumbel produces gradients that are $10dB$ larger.

In conclusion, the network learns better the rare categories by using Gumbel activation than by using Sigmoid activation, as the gradient is larger with Gumbel. This is also reflected in the formula of the positive Sigmoid gradient and the positive Gumbel gradient. In detail, Sigmoid positive gradient is bounded

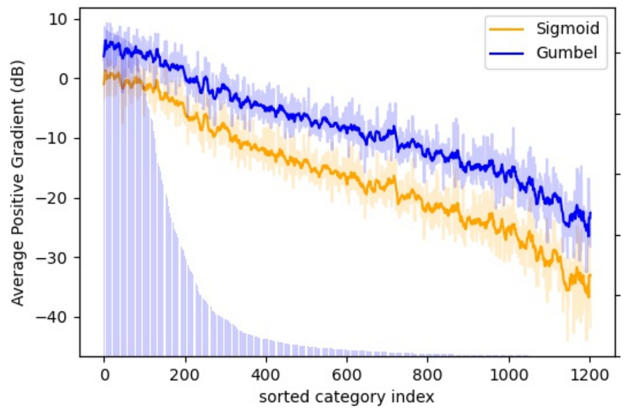


**Fig. 2.** Average Positive Gradient $g$ per category, measured in decibel, (dB). Gumbel activation produces larger gradients for rare categories and facilitates rare category learning.

to values $(-1, 0)$, while Gumbel positive gradient is exponential and has values that reach $(-\infty, 0)$. This enables Gumbel activation to produce larger gradients than Sigmoid and it is useful for rare categories, where the gradient updates are scarce.

### 1.5 Gumbel Optimised Loss

Our *GOL* method is based in DropLoss [6]. It is described as follows:

$$\mathcal{L}_{GOL} = -\sum_{j=1}^{C} \log(w_j^{Drop} \bar{p_j}), \quad \bar{p_j} = \begin{cases} \eta_\gamma(q_i), & if \;\; y_j = 1 \\ 1 - \eta_\gamma(q_i), & if \;\; y_j = 0 \end{cases} \tag{4}$$

$$w_j^{Drop} = \begin{cases} 1 - T_\lambda(f_j)(1 - y_j), & if \;\; E(r) = 1 \\ w \sim \mathrm{Ber}(\mu_{f_j}), & otherwise \end{cases} \tag{5}$$

$$\mu_{f_j} = \begin{cases} (n_{rare} + n_{common})/n_{all}, & if \;\; T_\lambda(f_j) = 1 \\ n_{frequent}/n_{all}, & otherwise \end{cases} \tag{6}$$

where $E(r)$ is a binary indicator function that outputs 1 if a region proposal $r$ is foreground, $T_\lambda(f_j)$ is a rare category indicator that outputs 1 if the frequency of category $j$ is lower than $\lambda$, $w \in \{0, 1\}$ is a random variable drawn from Bernoulli distribution and $\mu_{f_j}$ is the shape parameter that is computed according to the foreground region proposals in the training batch.

**Table 1.** Ablation study, using MaskRCNN, Resnet50 and training schedule 2x.

| RFS | Gumbel | EQL | Enh | DropLoss | $AP$ | $AP^r$ | $AP^c$ | $AP^f$ | $AP^b$ |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | 18.7 | 1.1 | 16.2 | 29.2 | 19.5 |
| | ✓ | | | | 22.0 | 8.9 | 20.3 | 29.6 | 22.4 |
| | | ✓ | | | 21.6 | 3.8 | 21.7 | 29.2 | 22.5 |
| | ✓ | ✓ | | | 23.9 | 11.4 | 23.4 | 29.9 | 24.2 |
| ✓ | | | | | 23.7 | 13.3 | 23.0 | 29.0 | 24.7 |
| ✓ | ✓ | | | | 23.5 | 13.8 | 22.2 | 29.2 | 24.3 |
| ✓ | | ✓ | | | 25.3 | 17.4 | 24.9 | 29.2 | 26.0 |
| ✓ | ✓ | ✓ | | | 26.1 | 18.4 | 25.9 | 29.8 | 26.8 |
| ✓ | ✓ | ✓ | ✓ | | 26.9 | 18.1 | 26.5 | **31.3** | 26.8 |
| | ✓ | | ✓ | ✓ | 25.6 | 14.5 | 26.1 | 29.9 | 25.1 |
| ✓ | ✓ | | ✓ | ✓ | **27.7** | **21.4** | **27.7** | 30.4 | **27.5** |

## 2 Long-tailed instance segmentation

### 2.1 Ablation Study

In Table 1, we conduct an ablation study of Gumbel activation, RFS [4], EQL [9], DropLoss [6], Normalised Mask [10] and stricter Non Maximum Suppression (NMS) threshold. We denote the stricter NMS threshold and Normalised Mask enhancements as (Enh).

As shown in Table 1 the best overall performance is achieved with Gumbel, RFS, Enh and DropLoss, we denote this pipeline as Gumbel Optimised Loss ($GOL$). The best performance on $AP^f$ is achieved using Gumbel, RFS, Enh and EQL, we denote this pipeline as $GOL$*.

**Total Performance** Our $GOL$ method significantly boosts the vanilla MaskR-CNN $AP$ by 9.0%, and it largely improves $AP^r$ by 20.3%, $AP^c$ by 11.5%, $AP^f$ by 1.2% and $AP^b$ by 8.0%.

**Table 2.** MaskRCNN with Resnet50, schedule 1x, EQLv1 loss [9], DropLoss [6], ACSL [11] and Federated Loss [13]. Gumbel activation boosts $AP$ of all models.

| Method | Activation | $AP$ | $AP^r$ | $AP^c$ | $AP^f$ | $AP^b$ |
|---|---|---|---|---|---|---|
| EQL$^\dagger$[8] | Sigmoid | 18.6 | 2.1 | 17.4 | 27.2 | 19.3 |
| EQL | Gumbel | **21.7** | **9.6** | **20.6** | **28.2** | **21.8** |
| DropLoss$^\dagger$[6] | Sigmoid | 19.8 | 3.5 | 20.0 | 26.7 | 20.4 |
| DropLoss | Gumbel | **22.0** | **10.0** | **22.1** | **27.1** | **21.9** |
| ACSL [11] | Sigmoid | 20.7 | 9.6 | 19.7 | 26.6 | **21.2** |
| ACSL | Gumbel | **21.0** | **10.9** | **19.8** | **26.7** | 21.1 |
| Federated Loss [13] | Sigmoid | 17.6 | 1.8 | 14.9 | 27.5 | 18.2 |
| Federated Loss | Gumbel | **20.1** | **6.0** | **18.5** | **28.0** | **20.5** |

**Table 3.** Comparison of activations in various frameworks using 1x schedule.

| Method | Framework | $AP$ | $AP^r$ | $AP^c$ | $AP^f$ | $AP^b$ |
|---|---|---|---|---|---|---|
| Sigmoid | MaskRCNN-ResNet50[5] | 16.4 | 0.8 | 12.7 | 27.3 | 17.2 |
| Softmax | | 15.2 | 0.0 | 10.6 | 26.9 | 16.1 |
| Gumbel | | **19.0** | **4.9** | **16.8** | **27.6** | **19.1** |
| Sigmoid | MaskRCNN-ResNet101 | 17.8 | 0.9 | 14.5 | 28.8 | 18.8 |
| Softmax | | 16.7 | 0.5 | 12.5 | 28.5 | 17.7 |
| Gumbel | | **20.6** | **6.4** | **18.5** | **29.2** | **21.0** |
| Sigmoid | MaskRCNN-ResNeXt101 | 19.6 | 1.0 | 16.5 | 31.2 | 20.7 |
| Softmax | | 18.6 | 0.6 | 14.5 | 31.1 | 19.7 |
| Gumbel | | **22.6** | **5.9** | **21.3** | **31.4** | **22.8** |
| Sofmax | Cascade MaskRCNN-Resnet101[1] | 18.8 | 0.6 | 15.7 | 30.3 | 21.3 |
| Gumbel | | **22.9** | **6.6** | **22.4** | **30.7** | **25.8** |
| Sofmax | Hybrid Task Cascade-ResNet101[2] | 19.1 | 0.6 | 15.8 | 31.0 | 21.1 |
| Gumbel | | **23.3** | **6.1** | **22.7** | **31.4** | **25.6** |

### 2.2 Results on Larger Frameworks and SOTA Losses

In Table 2, we show detailed results when using Gumbel activation and SOTA long-tailed instance segmentation loss functions. In Table 3, we show detailed experimental results using Gumbel activation and common instance segmentation frameworks. In all cases, Gumbel activation improves the overall segmentation performance of models.

### 2.3 Results on Larger Models

We report the performance of our methods using larger models such as MaskR-CNN with ResNet-101. As shown in Table 4, using MaskRCNN ResNet-50, *GOL* significantly outperforms the best method, LOCE [3] by 1.1% on $AP$, by 2.9% on $AP^r$ and by 1.5% on $AP^c$, using smaller training budget and the same enhancements.

Using MaskRCNN ResNet-101, *GOL* largely surpasses the best state-of-the-art Seesaw [10] by 0.9% in overall $AP$, 2.8% in $AP^r$, 1.0% in $AP^c$ and 0.3% in $AP^b$ using the same enhancements and RFS sampler. It also surpasses LOCE by 1.0% in overall $AP$ using fewer training epochs.

Finally, our $GOL^*$ method has the best $AP^f$ in both MaskRCNN ResNet-50 and MaskRCNN ResNet-101 backbones, thus it is useful if $AP^f$ is most important.

### 2.4 Object Distributions

We further show more examples of object distributions in LVIS v1 validation set. As shown in Figure 4, Gumbel activation produces object distributions that are closer to the target distribution as they have lower K-L divergence.

**Table 4.** Comparative results on LVISv1 using MaskRCNN-FPN and schedule 2x.

| Method | Sampler | Backbone | $AP$ | $AP^r$ | $AP^c$ | $AP^f$ | $AP^b$ |
|---|---|---|---|---|---|---|---|
| Softmax | random | MaskRCNN ResNet50 | 18.7 | 1.1 | 16.2 | 29.2 | 19.5 |
| LOCE[3] | | | 23.8 | 8.3 | 23.7 | 30.7 | 24.0 |
| EQLv2[8] | | | 25.5 | 17.7 | 24.3 | 30.2 | 26.1 |
| Seesaw[10] | | | 25.0 | 16.1 | 24.2 | 29.7 | 25.6 |
| Disalign[12] | | | 24.2 | 13.2 | 23.8 | 29.3 | 24.7 |
| *GOL* (ours) | | | 25.6 | 14.5 | 26.1 | 29.9 | 25.1 |
| LOCE[9] | MFS[9] | MaskRCNN ResNet50 | 26.6 | 18.5 | 26.2 | 30.7 | 27.4 |
| NorCal[7] | RFS | MaskRCNN ResNet50 | 25.2 | 19.3 | 24.2 | 28.6 | - |
| EQLv2[8] | | | 25.8 | 17.3 | 25.4 | 30.0 | 26.2 |
| Seesaw[10] | | | 26.4 | 19.5 | 26.1 | 29.7 | 27.6 |
| *GOL** (ours) | | | 26.9 | 18.1 | 26.5 | **31.3** | 26.8 |
| *GOL* (ours) | | | **27.7** | **21.4** | **27.7** | 30.4 | **27.5** |
| EQLv2[8] | random | MaskRCNN ResNet101 | 27.2 | 20.6 | 25.9 | 31.4 | 27.9 |
| Seesaw[10] | | | 27.1 | 18.7 | 26.3 | 31.7 | 27.4 |
| *GOL* (ours) | | | 27.0 | 16.1 | 27.4 | 31.2 | 26.8 |
| LOCE[9] | MFS[9] | MaskRCNN ResNet101 | 28.0 | 19.5 | 27.8 | 32.0 | 29.0 |
| NorCal[7] | RFS | MaskRCNN ResNet101 | 27.3 | 20.8 | 26.5 | 31.0 | 28.1 |
| Seesaw[10] | | | 28.1 | 20.0 | 28.0 | 31.8 | 28.9 |
| *GOL**(ours) | | | 28.0 | 19.3 | 27.5 | **32.4** | 28.3 |
| *GOL*(ours) | | | **29.0** | **22.8** | **29.0** | 31.7 | **29.2** |

## References


1. Cai, Z., Vasconcelos, N.: Cascade r-cnn: High quality object detection and instance segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
2. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al.: Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4974–4983 (2019)
3. Feng, C., et al.: Exploring classification equilibrium in long-tailed object detection. In: CVPR (2021)
4. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5356–5364 (2019)
5. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
6. Hsieh, T.I., Robb, E., Chen, H.T., Huang, J.B.: Droploss for long-tail instance segmentation. In: AAAI. vol. 3, p. 15 (2021)
7. Pan, T.Y., Zhang, C., Li, Y., Hu, H., Xuan, D., Changpinyo, S., Gong, B., Chao, W.L.: On model calibration for long-tailed object detection and instance segmentation. Advances in Neural Information Processing Systems **34** (2021)
8. Tan, J., Lu, X., Zhang, G., Yin, C., Li, Q.: Equalization loss v2: A new gradient balance approach for long-tailed object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1685–1694 (2021)

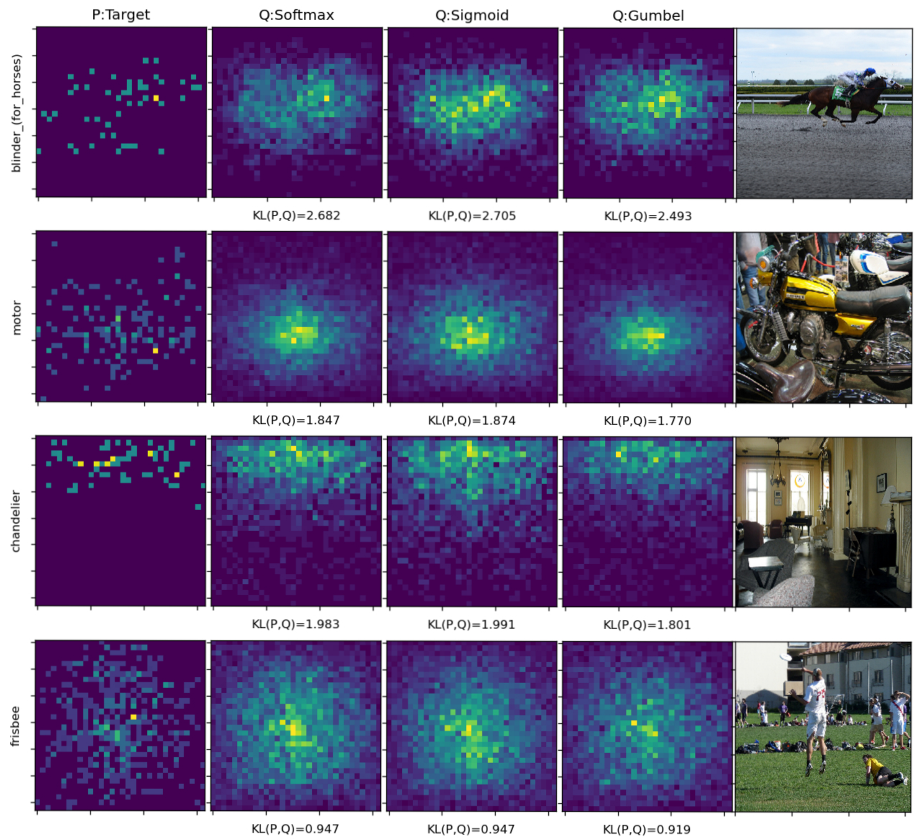


**Fig. 3.** Comparison of four object distributions in LVIS validation set, using Softmax (second column), Sigmoid (third column) and Gumbel (fourth column). Gumbel predicts distributions that have smaller K-L divergence than Sigmoid or Softmax.

9. Tan, J., Wang, C., Li, B., Li, Q., Ouyang, W., Yin, C., Yan, J.: Equalization loss for long-tailed object recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11662–11671 (2020)
10. Wang, J., Zhang, W., Zang, Y., Cao, Y., Pang, J., Gong, T., Chen, K., Liu, Z., Loy, C.C., Lin, D.: Seesaw loss for long-tailed instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9695–9704 (2021)
11. Wang, T., Zhu, Y., Zhao, C., Zeng, W., Wang, J., Tang, M.: Adaptive class suppression loss for long-tail object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3103–3112 (2021)
12. Zhang, S., Li, Z., Yan, S., He, X., Sun, J.: Distribution alignment: A unified framework for long-tail visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2361–2370 (2021)
13. Zhou, X., Koltun, V., Krähenbühl, P.: Probabilistic two-stage detection. In: arXiv preprint arXiv:2103.07461 (2021)