A Proof for Claim 1

Proof. We start with recalling some related notations and definitions. In Section 4.3, we design a novel soft contrastive loss by incorporating a pair-wise *positiveness* score $\boldsymbol{w}$ in Eq. 1:

$$\mathcal{L}_{\mathrm{CL}}^{soft}(\mathcal{D}) = \frac{1}{|\mathcal{D}|}\sum_{i\in\mathcal{D}}\left[\frac{1}{\sum_{j\in A(i)} w_{ij}}\sum_{j\in A(i)} -w_{ij}\cdot\log\frac{\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_j/\tau)}{\sum_{a\in A(i)}\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_a/\tau)}\right], \tag{8}$$

where $\mathcal{D}$ is the training set, $A(i)$ is the index of the set of instances involved in the same batch, $\boldsymbol{z}_i$ is the contrastive feature for instance $\boldsymbol{x}_i$, and τ is the temperature hyper-parameter.

For each instance $\boldsymbol{x}_i$, the soft contrastive loss $\mathcal{L}_{\mathrm{CL}}^{soft}(\boldsymbol{x}_i)$ can be written as:

$$\mathcal{L}_{\mathrm{CL}}^{soft}(\boldsymbol{x}_i) = C_i\sum_{j\in A(i)}\left[-w_{ij}\cdot\log\frac{\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_j/\tau)}{\sum_{a\in A(i)}\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_a/\tau)}\right], \tag{9}$$

where $C_i = \frac{1}{\sum_{j\in A(i)} w_{ij}}$ is a constant value for fixed $\boldsymbol{w}$, then the contrastive logit p_{ij} is defined as:

$$p_{ij} = \frac{\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_j/\tau)}{\sum_{a\in A(i)}\exp(\boldsymbol{z}_i\cdot\boldsymbol{z}_a/\tau)} \tag{10}$$

Then, we omit the constant C in Eq. 9 and solve the optimization problem with the Lagrange multiplier. The problem is defined as follows:

$$\begin{cases} \text{Minimize:} & f(p_{i1},\cdots,p_{in}) = -\sum_{k=1}^{n}(w_{ik}\cdot\log p_{ik}), \\ \text{Subject to:} & g(p_{i1},\cdots,p_{in}) = \sum_{k=1}^{n} p_{ik} - 1 = 0, \end{cases} \tag{11}$$

where n refers to the number of instances in the training batch, i.e., $n = |B| = |A(i)|$. The Lagrangian function of Eq. 11 and its corresponding partial derivatives are:

$$\mathcal{L}(p_{i1},\cdots,p_{in},\lambda) = -\sum_{k=1}^{n}(w_{ik}\cdot\log p_{ik}) + \lambda(\sum_{k=1}^{n} p_{ik} - 1), \tag{12}$$

$$\begin{cases} \frac{\partial\mathcal{L}}{\partial p_{ij}} & = -\frac{w_{ij}}{p_{ij}} + \lambda = 0 \\ \frac{\partial\mathcal{L}}{\partial\lambda} & = \sum_{k=1}^{n} p_{ij} - 1 = 0 \end{cases} \tag{13}$$

From Eq. 13, the optimal value of p_{ij} is $p_{ij}^{*} = \frac{w_{ij}}{\sum_{k=1}^{n} w_{ik}}$, which concludes the proof for Claim 1.

Claim 1 indicates that minimizing Eq. 6 encourages the similarity of features between two samples to be proportional to the corresponding *positiveness* score. In this way, we effectively transfer knowledge from the pseudo-labeling branch to contrastive learning.

B Pseudo-code

We summarize the pipeline of self-**Ba**lanced **Co**-advice co**n**trastive framework (BaCon) in Algorithm 1, and the source code can be found here.

Algorithm 1 The overall pipeline of BaCon.

Input: Train set $\mathcal{D} = \{\mathcal{D}^l \cup \mathcal{D}^u\}$, model parameter f_{con}, f_{cls}, train epoch T, warm-up epochs w, estimation interval r, sampling rate α, β, similarity metric $\mathrm{Sim}(\cdot)$, hyper-parameter p and k.
Output: Trained model parameter θ_T.
Initialize: Load DINO [6] pre-trained parameters for the backbone in two branches and the classifier and projector is randomly initialized.

```
for epoch = 0, ··· , T − 1 do
  if epoch%r = 0 then
    Re-estimate the training set distribution and re-place π_e (Section 4.1);
  end if
  Regularize the predictions of the classifier with Eq. 2;
  Train f_cls with Eq. 3;
  Compute the self-supervised CL loss L^u_CL(D) and supervised CL loss L^s_CL(D^l) with Eq. 1;
  if epoch ≥ w then
    Debiasing pseudo-labels with Eq. 4;
    Obtain the sampled pseudo-labels M(D^u) with sampling rate in Eq. 5;
    Calculate the pair-wise positiveness score w with w = Sim(p̃_i, p̃_j);
    Compute the soft CL loss L^soft_CL(D^l ∪ M(D^u)) with Eq. 6;
    Train f_con with Eq. 7.
  else
    Train f_con with L^u_CL(D) + γ_1 L^s_CL(D^l).
  end if
end for
```

After training, we test the model via k-means clustering on the backbone feature of the contrastive-learning branch. The test strategy is summarized in Algorithm 2.

Algorithm 2 The test stage strategy of BaCon.

Input: The balanced test set $\mathcal{D}^t = \{(\boldsymbol{x}_i, y_i)\} \in \mathcal{X} \times \mathcal{Y}$, fine-tuned backbone of the contrastive-learning branch f_b.
Output: Classification accuracy Acc.

```
for x_i ∈ D^t do
  Obtain the feature of x_i via f_b(x_i)
end for
Perform k-means clustering on all the features of test samples with the cluster number C = |D^t|;
Calculate the optimal assignment between clusters and classes by Hungarian algorithm [32];
Compute the test accuracy Acc based on the optimal assignment.
```

C Detailed Experimental Settings

In this section, we provide a comprehensive overview of the experimental setup, expanding upon the details presented in Section 5.1.

C.1 Benchmark Datasets

In Section 5.3, we present our experimental findings based on four generic long-tailed image recognition datasets. CIFAR-10-LT and CIFAR-100-LT, which are first introduced by [10]. Both datasets are long-tail subsets sampled from the original CIFAR-10 and CIFAR-100. The imbalance ratio is defined as the ratio between the number of instances in the largest class and the smallest class. In our experiments, we default the imbalance ratio to 100 to reflect the performance disparities between classes. ImageNet-100-LT, proposed by [27], which comprises 12K images sampled from the ImageNet-100 dataset [57] using a Pareto distribution. The instance number of each class in the training set ranges from 1,280 to 5. We also incorporate the Places dataset [80], a large-scale scene-centric dataset, in our experiments. We utilize Places-LT [40], which consists of approximately 62.5K images sampled from the Places dataset using a Pareto distribution. The training set of Places-LT is ranging from 4,980 to 5. A summary of the dataset statistics is provided in Table 7.

Table 7: Statistics of datasets.

Dataset	Long-tail type	Imbalance ratio ρ	#class	#train images	#test images
CIFAR-10-LT	Exp	100	10	11.2K	10.0K
CIFAR-100-LT	Exp	100	100	9.8K	10.0K
ImageNet-100-LT	Pareto	256	100	12.0K	5.0K
Places-LT	Pareto	996	365	62.5K	36.5K

C.2 Construction of Training and Testing Sets

To construct the training set for distribution-agnostic generalized category discovery (DA-GCD), we first sub-sample $|\mathcal{Y}_k|$ classes from the entire long-tail datasets (introduced in Section C.1) to stimulate known classes and treat the rest as novel classes, then sub-sample 50% instances in each known class as $\mathcal{D}^l$ and concentrate others with all samples in novel classes to form $\mathcal{D}^u$. The default $|\mathcal{Y}_k|$ for CIFAR-10-LT, CIFAR-100-LT, ImageNet-100-LT, and Places-LT are 5, 80, 50, and 182 respectively. Our default training set of each dataset is summarized in Table 8. We also evaluate BaCon with different $\mathcal{Y}_k$, different labeled ratios r_l (default $r_l = 50\%$), and different imbalance ratios of both labeled set and unlabeled set.

Table 8: The default setting of DA-GCD's training set.

Dataset	Known classes $\lvert\mathcal{Y}_k\rvert$	Novel classes $\lvert\mathcal{Y}_n\rvert$	Labeled images $\lvert\mathcal{D}^l\rvert$	Unlabeled images $\lvert\mathcal{D}^u\rvert$
CIFAR-10-LT	5	5	4.5K	6.7K
CIFAR-100-LT	80	20	3.8K	6.0K
ImageNet-100-LT	50	50	3.2K	9.0K
Places-LT	182	183	20.8K	41.7K

For the test set, we use a disjoint balanced set as a common practice in the field of long-tail learning [29, 31, 27]. It's worth noting that the test set contains *both* known and novel classes, and the model is required to classify test set instances to every single class.

C.3 Implementation Details of BaCon

We implement all our techniques using PyTorch [47] and conduct the experiments using a RTX3090 GPU. Following GCD [60], we load a DINO [6] pre-trained ViT-B/16 [13] as the backbone for both

branches. The classification head and projector are randomly initialized. We only fine-tune the last block of the backbone, the classification head in the pseudo-labeling branch, and the projector in the contrastive-learning branch. We use the output of $[\mathrm{CLS}]$ token with a dimension of 768 as the backbone feature for an input image. The classifier is implemented with a single fully-connected layer, and we use the cosine classifier due to its competent empirical results in long-tail learning [29]. The projector is an MLP with an output feature dimension of 65536. We train with a batch size of 256, and an initial learning rate of 0.1 decayed with a cosine schedule. We train for 200 epochs on each dataset. The temperature τ is set to 1.0, the hyper-parameter p and k is set to 0.5, the sampling rates $\alpha = 0.8$, $\beta = 0.5$, and the re-estimate interval r is 10 (epochs). We use dot product as the similarity metric to define the *positiveness* score $\boldsymbol{w}$.

D Introduction of Baseline Methods

In our paper, we formally define a more realistic task called distribution-agnostic generalized category discovery (DA-GCD) and design a novel framework BaCon for the challenge setting. To evaluate the effectiveness of BaCon, we compare the proposed method with SOTA techniques in two closely related fields: imbalanced semi-supervised learning (imbalanced SSL) and generalized category discovery (GCD). In this section, we provide a detailed introduction of baseline methods (including ABC [33] and DARP [31] in imbalanced SSL; GCD [60], ORCA [4], OpenCon [56], and SimGCD [70] in GCD).

Imbalanced SSL [31, 33, 69, 19] considers the scenario where the training set has a long-tailed distribution and only a small subset of data is labeled. DARP [31] suggests refining pseudo-labels to the long-tailed distribution of the training set, and further designing a method to estimate the unknown distribution based on the assumption that the confusion matrix for labeled data and the unlabeled part are almost the same. Unfortunately, their assumption does not always hold in practice since the model tends to overfit the labeled samples. ABC [33] introduces an auxiliary balanced classifier to mitigate the data imbalance issue. Nevertheless, they also require the class-wise data distribution which is agnostic (due to the presence of open-set class samples) in DA-GCD.

Generalized category discovery (GCD), formulated by [60], is a new-raised domain in open-set learning. GCD manages to jointly recognize *known* categories contained in the manually annotated subset as well as *novel* (open-set) classes which appeared in the unlabeled set. Note that GCD classified the open-set samples into fine-grained categories according to their visual concept, which is different from OOD detection [37, 24, 39]. The pioneering work GCD [60] proposes to leverage the pre-trained ViT and contrastive learning to learn a discriminative feature space and further obtain predictions by the proposed semi-k-means clustering strategy. ORCA [4] suggests balancing the learning rate of known and novel classes by an uncertainty adaptive margin mechanism. OpenCon [56] attempts to supervise the unlabeled instances by generating pseudo-positive pairs for closely aligned representations. But this paradigm could lead to another dilemma: the representation and pseudo-label are interdependent, which means, an inferior feature space (e.g., lack of intra-class consistency) could lead to false positive pairs, which deteriorate the learning of feature space in turn. A recent work SimGCD [70], argues that parametric classification could achieve better performance with elaborately designed pseudo-labeling techniques. However, the aforementioned methods in GCD are built on the class balance assumption, and we observe a consistent performance degradation when generalizing to DA-GCD scenarios.

E More Empirical Results

In this Section, we provide more empirical results on different datasets, and experimental settings to further verify the effectiveness of the proposed BaCon.

E.1 Results with Different Ratio of Known and Novel Classes

Here we show the accuracy when changing the number of known (close-set) and novel (open-set) classes. A small $|\mathcal{Y}_k|$ means fewer categories have manually labeled samples (illustrated in Figure 1), making the learning process more difficult. The performance is reported in Table 9. It's revealed that BaCon achieves the best accuracies on all settings, and brings larger performance gain under more challenging settings, i.e., few known classes and a large number of novel categories (outperforms best baseline $\sim$7% overall accuracy on CIFAR-100-LT when $|\mathcal{Y}_k| = 20$). Note the default $|\mathcal{Y}_k|$ for CIFAR-10-LT, CIFAR-100-LT, ImageNet-100-LT, and Places-LT are 5, 80, 50, and 182 respectively.

Table 9: Test accuracy (%) on CIFAR-100-LT with different $|\mathcal{Y}_k|$.

	$\lvert\mathcal{Y}_k\rvert = 20$			$\lvert\mathcal{Y}_k\rvert = 50$			$\lvert\mathcal{Y}_k\rvert = 80$		
Methods	Old	New	All	Old	New	All	Old	New	All
TRSSL†	**65.5**	30.7	37.6	62.2	30.5	46.4	58.7	35.8	54.1
ORCA†	43.5	29.3	32.1	56.6	30.2	43.4	55.0	30.8	50.1
SimGCD	51.2	27.3	32.1	66.3	19.6	42.9	59.8	24.2	52.8
GCD	63.4	50.8	53.3	67.0	53.0	60.0	65.5	49.0	62.2
OpenCon†	60.3	37.2	41.8	**69.0**	33.3	51.2	64.2	40.9	59.6
BaCon-O	62.7	56.9	58.1	66.8	59.4	63.1	66.5	**69.6**	67.1
BaCon-S	64.2	**59.2**	**60.2**	67.9	**60.2**	**64.0**	**67.4**	66.5	**67.2**

E.2 Results on Few-annotated Scenarios

In Table 10, we also conduct experiments under few-annotated scenarios, i.e., $r_l = \{10\%, 30\%, 50\%\}$. The annotation ratio r_l is computed by $r_l = \frac{\#\text{labeled samples in a known class}}{\#\text{all samples in a known class}}$, since we only have access to manually annotated images from known categories (illustrated in Figure 1c), and all novel class instances are unlabeled. Note that we set $r_l = 50\%$ in default in the main paper.

Table 10: Test accuracy (%) with different annotation ratio r_l.

	$r_l = 10\%$			$r_l = 30\%$			$r_l = 50\%$		
Methods	Old	New	All	Old	New	All	Old	New	All
TRSSL†	53.7	32.6	49.5	56.0	32.6	51.4	58.7	35.8	54.1
ORCA†	43.8	41.3	43.3	51.2	32.2	47.4	55.0	30.8	50.1
SimGCD	51.2	17.6	44.5	55.7	25.6	49.7	59.8	24.2	52.8
GCD	61.7	54.9	60.4	63.1	58.8	62.2	65.5	49.0	62.2
OpenCon†	52.1	40.9	49.9	61.8	43.9	58.2	64.2	40.9	59.6
BaCon-O	63.2	58.8	62.3	64.9	58.8	62.3	66.5	**69.6**	67.1
BaCon-S	**64.0**	**60.5**	**63.3**	**65.6**	**65.6**	**65.6**	**67.4**	66.5	**67.2**

E.3 Results with Different Imbalance Ratio on Labeled and Unlabeled Set

In Table 5, we report the performance of BaCon and baseline methods in GCD when the training set has a different imbalance ratio ρ. Now, we further set the imbalance ratio of labeled data and unlabeled data to be unequal ($\rho_l \neq \rho_u$), which is common in practice. For example, in medical image processing, the labeled data and unlabeled parts may come from hospitals in different countries. When it comes to autonomous driving, the occurrence frequency of stones or pedestrians may slightly change when the training data is from city areas or mountain roads.

The results are summarized in Table 11. We set the imbalance ratio of labeled data to 100 and vary the value of ρ_u across different settings, namely $\{20, 50, 100, 150\}$. It's observed that the proposed BaCon outperforms all baseline methods by a large margin, indicating that BaCon is robust to the shift of imbalance ratio between labeled and unlabeled samples.

Table 11: Test accuracy (%) on CIFAR-100 ($\rho_l = 100$) with different ρ_u

	$\rho_u = 20$			$\rho_u = 50$			$\rho_u = 100$			$\rho_u = 150$		
Methods	Old	New	All	Old	New	All	Old	New	All	Old	New	All
TRSSL†	63.1	38.8	58.2	57.2	44.0	54.6	58.7	35.8	54.1	55.5	39.0	52.2
ORCA†	56.4	45.5	54.2	56.9	42.1	53.9	55.0	30.8	50.1	56.9	31.9	51.8
SimGCD	61.1	27.9	54.4	59.2	29.2	53.2	59.8	24.2	52.8	59.6	24.7	52.6
GCD	65.6	56.1	63.7	65.5	54.1	63.2	65.5	49.0	62.2	64.5	50.9	61.8
OpenCon†	66.8	48.8	63.2	64.7	51.7	62.1	64.2	40.9	59.6	64.1	40.2	59.3
BaCon-O	67.1	63.4	66.4	67.0	61.0	65.8	66.5	**69.6**	67.1	65.3	54.4	63.1
BaCon-S	**69.2**	**66.2**	**68.6**	**68.7**	**65.3**	**68.0**	**67.4**	66.5	**67.2**	**67.8**	**64.2**	**67.1**