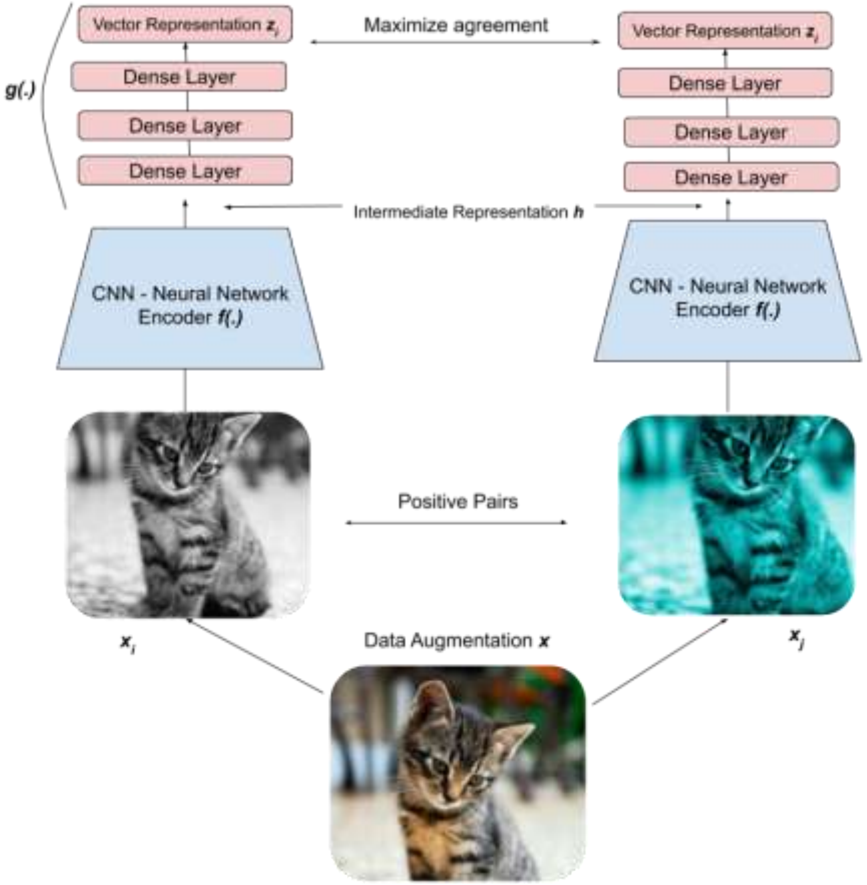
Improving Contrastive Learning on Imbalanced Seed Data via Open-World Sampling

Introduction

- Contrastive learning has been successfully applied to learning strong visual representations in an unsupervised manner.


Vector Representation z_i
Maximize agreement
Vector Representation z_j
$g(.)$
Dense Layer
Dense Layer
Dense Layer
Dense Layer
Dense Layer
Dense Layer
Intermediate Representation h
CNN - Neural Network Encoder $f(.)$
CNN - Neural Network Encoder $f(.)$
Positive Pairs
x_i
Data Augmentation x
x_j

- Learning with massive unannotated data, e.g., from internet-scale sources (expensive), limited computing budget(the **out-of-distribution** data would suppress the learning of relevant features)
- The data distribution in the open world are extremely **diverse** and always exhibits **long tails**

Sampling open-world unlabeled data for improving the representation learning, not just for the head classes but also for the tailed classes.

Problem Setting

- Start from a relatively small (“seed”) set of **unlabeled** training data(highly skewed yet unspecified)
- Aim to **retrieve an extra set**, with a given sampling budget, of freely available images from some external sources, to enhance self-supervised representation learning for targeted distribution (of seed set)

Challenges

- The actual class imbalancedness is unknown, making the most approaches handling imbalance in the (semi-)supervised setting inapplicable.
- Adopting a pre-trained backbone trained on imbalanced seed data with tail classes under-learned may amplify unfairness.
- Widely existing irrelevant outlier samples in the open world are harder to detect given the lack of label information.

Principles

- **Tailness**: Using each sample's training loss to identify "hard samples", which is weaker and noisier. Therefore, we propose to instead use an **empirical contrastive loss expectation(ECLE)** of sample loss over multiple random augmentations as the proxy.
- **Proximity**: We incorporate **a feature distance regularizer** between new external samples and seed training samples to reject too "far-away" samples from the former.
- **Diversity**: We include another **diversity-promoting term** in sample selection.

Eventually, our ideas could be mathematically unified into one framework called **Model-Aware K-center (MAK)**

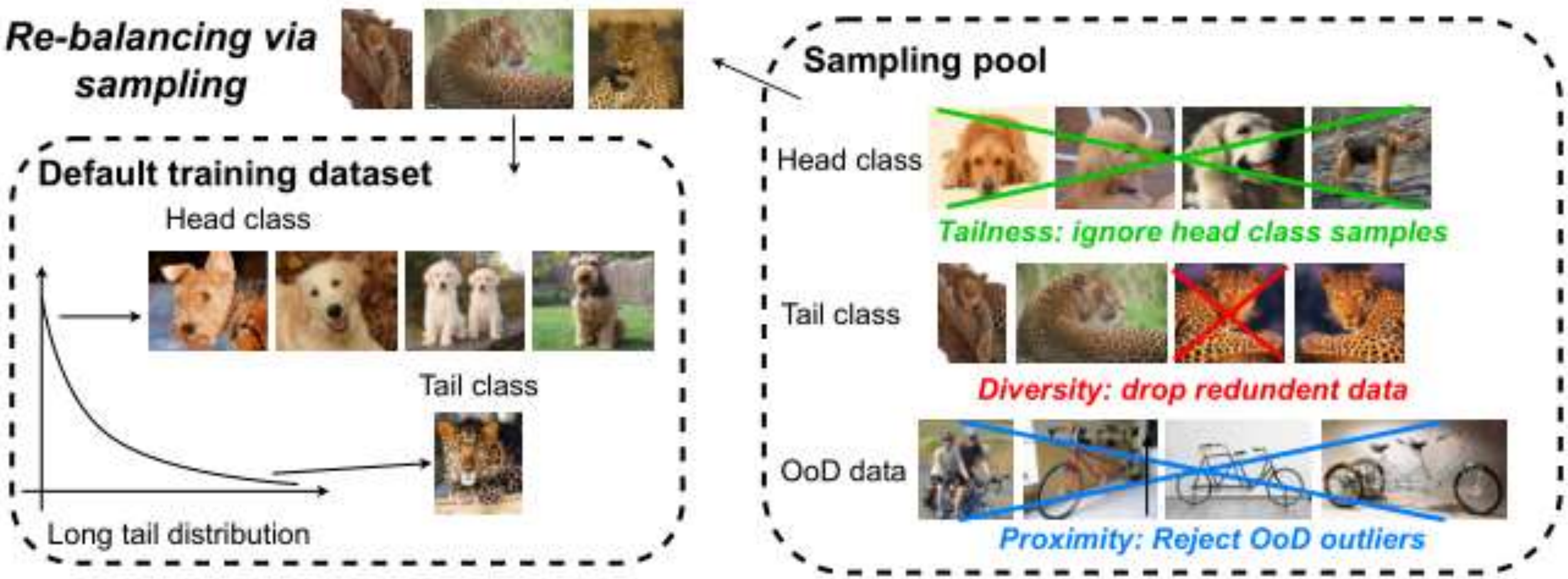
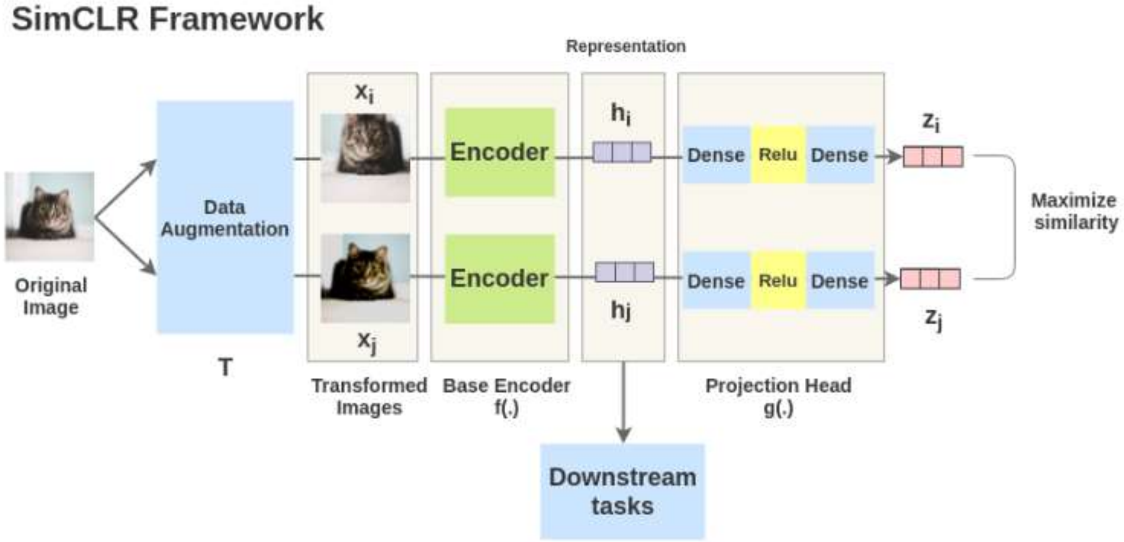

Re-balancing via sampling
Sampling pool
Default training dataset
Head class
Long tail distribution
Tail class
Head class
Tailness: ignore head class samples
Tail class
Diversity: drop redundent data
OoD data
Proximity: Reject OoD outliers

Figure 1: The overview of the proposed MAK sampling framework. MAK re-balances the long tail distribution via sampling additional data from a sampling pool. MAK are composed by three components: *tailness*, *proximity* and *diversity*.

Backbone to develop our framework——SimCLR

- positive pairs: two augmented views of the same data
- negative samples: all other augmented samples in the same batch
- enforcing an anchor sample v_i to be similar to another positive sample while being different from negative samples.


SimCLR Framework

The SimCLR loss associated with the i-th sample in the batch:

$$\mathcal{L}_{\text{CL,i}} = -\log \frac{s^{\tau}\left(A(v_i,\theta_{i,1}), A(v_i,\theta_{i,2})\right)}{s^{\tau}\left(A(v_i,\theta_{i,1}), A(v_i,\theta_{i,2})\right) + \sum_{v_i^- \in V^-} s^{\tau}\left(A(v_i,\theta_{i,1}), v_i^-\right)} \qquad s^{\tau}(a,b) = \exp\left(a \cdot b/\tau\right)$$

NT-Xnet(the normalized temperature-scaled cross entropy loss)

Spotting Hard Samples from Tail Classes: A New Proxy for **Tailness**

- The contrastive loss largely depends on the random augmentations A(·, θ) and thus display high randomness.
- To eliminate the randomness, we turn to the following new proxy value for the i-th sample, that is designed to “smooth out'' random augmentations by integrating over them.

$$\mathcal{L}^{\mathcal{E}}{}_{\mathrm{CL,i}} = \mathbb{E}_{\theta_{i,1},\theta_{i,2}\sim\Theta}\left(\mathcal{L}_{\mathrm{CL,i}}(\theta_{i,1},\theta_{i,2};\tau,v_i,V^{-})\right)$$

- In practice, the expectation is approximated by the sample mean, e.g., drawing $\{\theta_{i,1},\theta_{i,2}\}$ for M times and then averaging corresponding $L_{CL,\mathrm{i}}$ values.
- Sort and choose those with the largest ECLE (empirical contrastive loss expectation)values as hard samples.

Proximity

- Adopting only the ECLE proxy might easily pick those outliers, hurting feature learning and generalization on the underlying distribution.
- We construct a regularization term that promotes proximity via rejecting OoD outliers.

$$D(s^0, s^1) = \frac{1}{|s^1|} \sum_{j \in s^1} \min_{i \in s^0} \Delta(x_i, x_j)$$

s^1 be the new additional set, s^0 be the seed training set

$\Delta(x_i, x_j)$ denote the feature distance between two samples

- In practice, to compute $\Delta(x_i, x_j)$, we use the normalized cosine distance.

$$\text{cosine similarity} = S_C(A,B) := \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2} \sqrt{\sum\limits_{i=1}^{n} B_i^2}} \qquad \text{cosine distance} = D_C(A,B) := 1 - S_C(A,B)$$

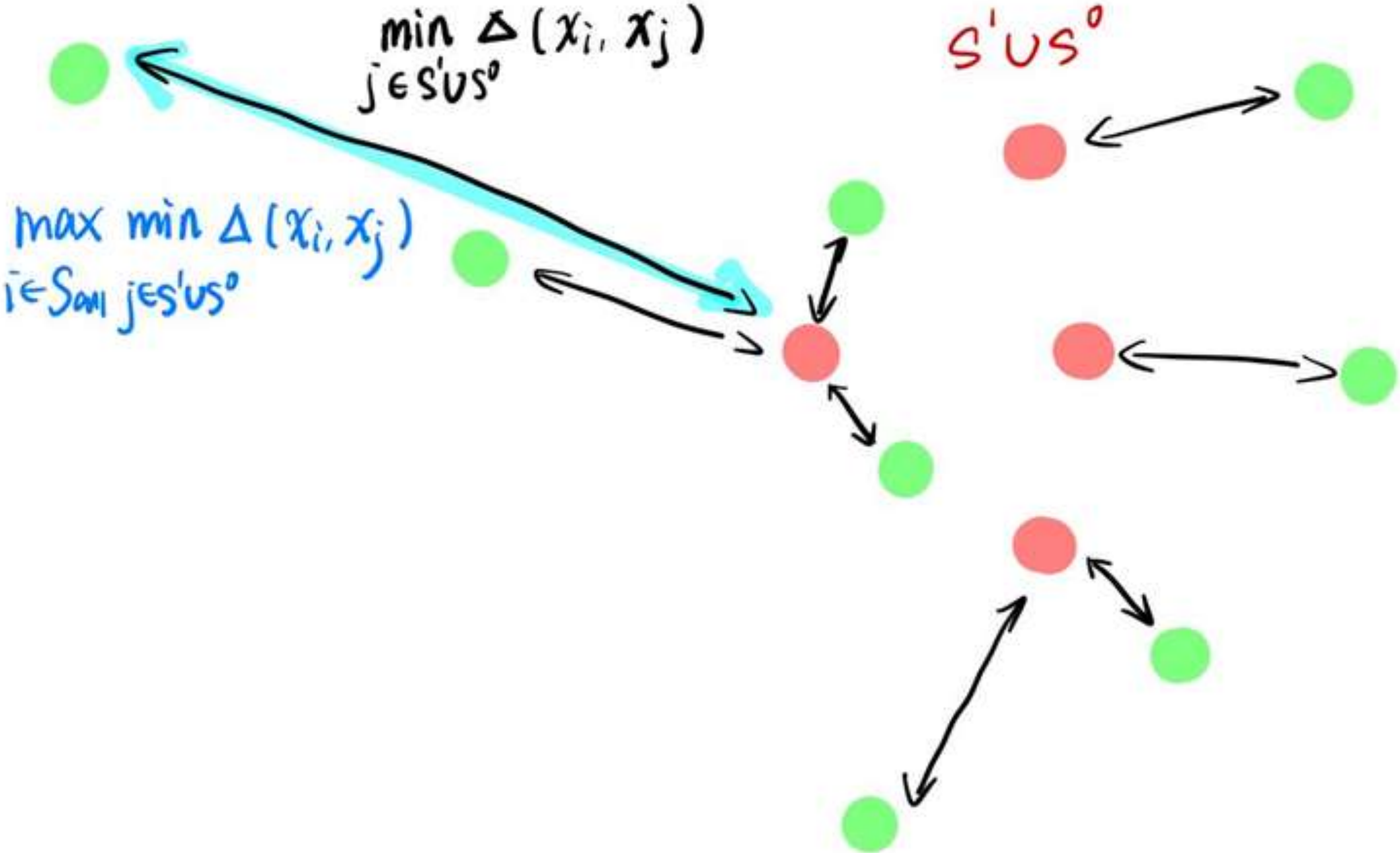
- For further efficiency, we pre-compute the set of feature prototypes from $\boldsymbol{s^0}$ using K-means clustering, denoted as $\boldsymbol{s_p^0}$, and then compute D($\boldsymbol{s_p^0}$, $\boldsymbol{s^1}$).

Diversity

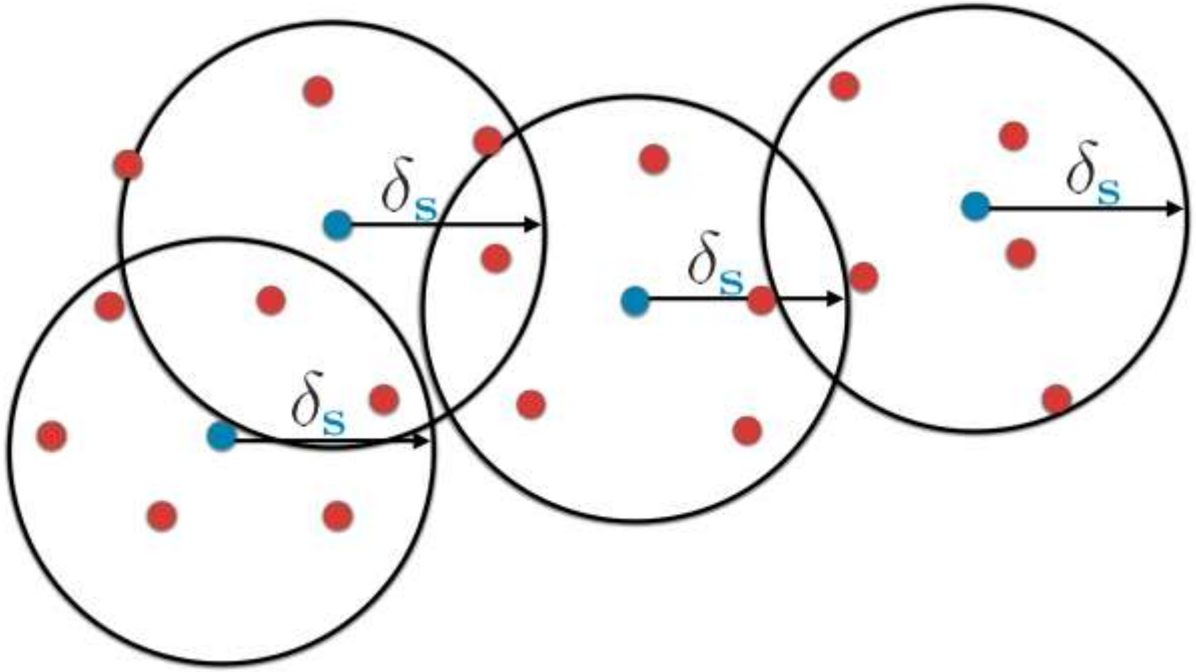
- Oversampling too many external images both would add to the training overhead, and might not necessarily help.
- Attain the informative samples within the size limit | s^1 | ⩽ K
- We introduce the following regularization term:

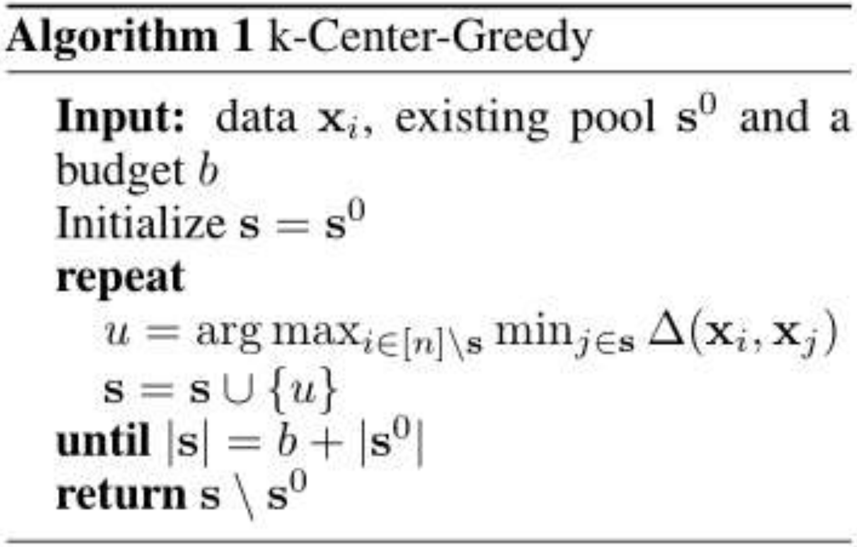
$$H(s^1 \cup s^0, S_{all}) = \max_{i \in S_{all}} \min_{j \in s^1 \cup s^0} \Delta(x_i, x_j)$$

- Minimizing it boils down to choosing | s^1 | center points on top of the given | s^0 | points, such that the largest distance between any data point from s_{all} and its nearest center point from $s^1 \cup s^0$ is minimized.


$\min_{j \in S' \cup S^0} \Delta(x_i, x_j)$
$S' \cup S^0$
$\max_{i \in S_{all}} \min_{j \in S' \cup S^0} \Delta(x_i, x_j)$

K-Center





Algorithm 1 k-Center-Greedy

```
Input: data $\mathbf{x}_i$, existing pool $\mathbf{s}^0$ and a budget $b$
Initialize $\mathbf{s} = \mathbf{s}^0$
repeat
    $u = \arg\max_{i \in [n] \setminus \mathbf{s}} \min_{j \in \mathbf{s}} \Delta(\mathbf{x}_i, \mathbf{x}_j)$
    $\mathbf{s} = \mathbf{s} \cup \{u\}$
until $|\mathbf{s}| = b + |\mathbf{s}^0|$
return $\mathbf{s} \setminus \mathbf{s}^0$
```

Model-Aware K-Center: A Unified Framework

$$\max_{s^1:|s^1|\leq K}\{\sum_{i\in s^1}\mathcal{L}^{\mathcal{E}}{}_{\mathrm{CL,i}} - D(s^0, s^1) - H(s^1 \cup s^0, S_{all})\}$$

- Find a sample set s^1 from the external source, such that :

(i) mine more data for tail classes while overcoming augmentation randomness, by sorting external samples by their ECLE values (tailness);

(ii) reject the out-of-distribution outliers that might distract training, by constraining feature distances from the seed set (proximity);

(iii) control the sample volume under K while ensuring sample diversity, by K-center sample selection (diversity).

Algorithm 1: A greedy heuristic to efficiently solve MAK.

```
Require : seed training set $s^0$ , external data $S_{all} \backslash s^0$ , sampling budget $K$, candidate set size $C$,
          feature distance function $\Delta$ (cosine distance in practice), coefficient $\alpha \in (0, 1)$.
Train feature extractor $f$ on $s^0$ with self-supervised method;
Calculate ECLE $\mathcal{L}_{\mathrm{CL,i}}^{\mathcal{E}}$ and average feature distance $D\left(s^0, s^1\right)$ with $f$ as in Equation 2 and 3,
  respectively;
Summarize $\mathcal{L}_{\mathrm{CL,i}}^{\mathcal{E}}$ and $D\left(s^0, s^1\right)$ with score $q = \alpha N(\mathcal{L}_{\mathrm{CL,i}}^{\mathcal{E}}) - (1 - \alpha) N(D\left(s^0, s^1\right))$ where
  $N(v) = \frac{v - mean(v)}{std(v)}$ is the normalization function;
Construct set $S'$: from $S_{all} \backslash S^0$, find all samples whose score $q$ are top $C$ largest among all;
// tailness & proximity
Construct set $s$: Initialize $s = s^0$ ;
while $|s \backslash s^0| \leq K$ do // Apply K-center greedy algorithm for diversity
    $u = \arg\max_{i \in S'} \min_{j \in s} \Delta(x_i, x_j)$ ;
    $s = s \cup \{u\}$ ;
end
return $s^1 = s \backslash s^0$
```

Experiment

- **Seed Training Datasets**: ImageNet-100-LT
- **Sampling Datasets**:

(i) ImageNet-900 (ii) ImageNet-Places-Mix

- **Evaluation protocol**:

(1) linear separability performance:

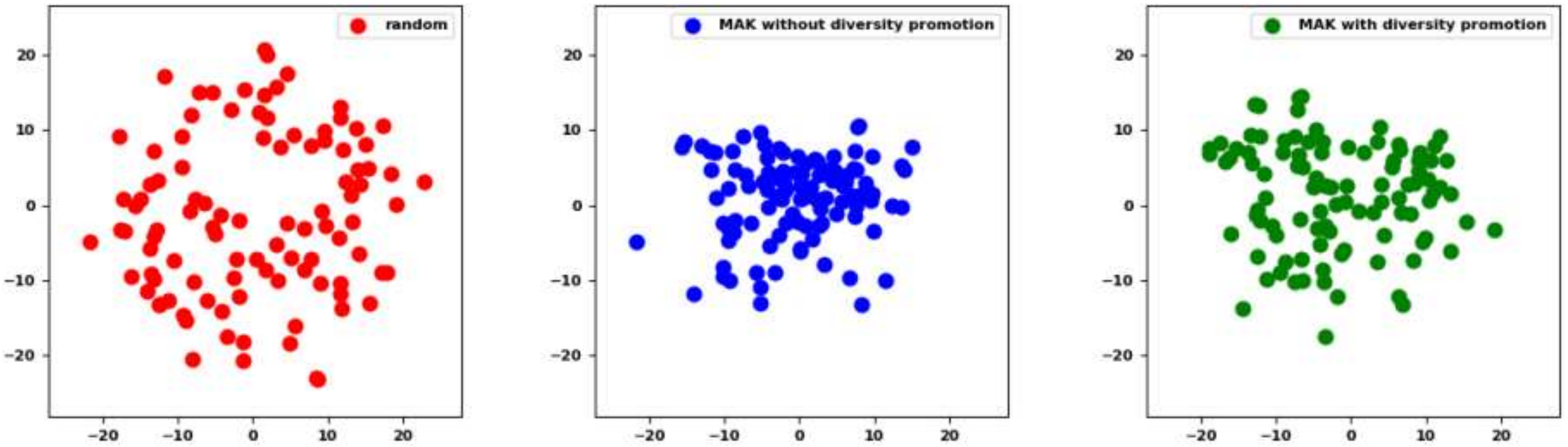
- Pre-train a model f with contrastive learning on the imbalanced ImageNet dataset
- Fine-tune a linear classifier with visual representation produced with a balanced dataset
- Testing the accuracy on testing dataset for the linear classifier

(2) few-shot performance:

The whole model are fine-tuned on 1% samples of the full dataset from where the long tail dataset is sampled.

Sampling dataset	Budget	method	Protocol	*Many* ↑	*Medium* ↑	*Few* ↑	*Std*↓ (imbalancedness)	All ↑
None	-	-	*linear separability*	71.2±0.8	65.3±0.7	62.7±0.9	3.6±0.5	67.3±0.7
			few-shot	52.6±0.3	40.5±1.5	32.5±1.1	8.3±0.4	44.2±1.0
IN900	10K	random	*linear separability*	74.6±0.3	69.7±0.4	66.1±1.2	3.5±0.5	71.2±0.2
			few-shot	56.6±1.2	48.6±0.4	43.7±1.7	5.3±1.2	51.1±0.1
		K-center	*linear separability*	73.6±0.3	68.6±0.8	64.5±0.9	3.8±0.3	70.0±0.4
			few-shot	55.0±0.4	45.8±0.3	39.1±1.1	6.5±0.6	48.5±0.2
		MAK	*linear separability*	**76.1±0.6**	**70.8±0.5**	**69.3±0.8**	**3.0±0.1**	**72.7±0.4**
			few-shot	**57.4±0.6**	**48.9±0.2**	**46.3±1.5**	**4.8±0.2**	**51.9±0.4**
	20K	random	*linear separability*	75.7±0.2	71.8±0.1	69.6±1.1	2.6±0.4	73.0±0.1
			few-shot	57.4±0.7	49.9±0.3	45.9±0.4	4.8±0.2	52.3±0.5
		MAK	*linear separability*	**78.0±0.8**	**73.4±0.6**	**72.4±0.3**	**2.4±0.3**	**75.1±0.6**
			few-shot	**59.0±0.9**	**52.9±0.5**	**50.0±0.4**	**3.8±0.5**	**54.9±0.4**
IPM	10K	random	*linear separability*	73.8±0.7	67.9±0.5	65.1±0.9	3.6±0.2	69.8±0.5
			few-shot	55.5±0.5	**45.8±0.8**	38.9±0.7	6.9±0.3	48.7±0.4
		K-center	*linear separability*	73.0±0.6	67.7±0.1	65.4±1.5	**3.2±0.4**	69.5±0.4
			few-shot	54.2±0.1	45.6±0.4	38.4±0.9	6.5±0.3	48.0±0.3
		MAK	*linear separability*	**74.7±0.2**	**69.2±0.7**	**66.6±0.7**	**3.3±0.3**	**71.1±0.5**
			few-shot	**56.8±0.7**	45.1±0.9	**42.6±0.8**	**6.2±0.1**	**49.3±0.7**

Tailness	*Proximity*	*Diversity*	Protocol	*Many* ↑	*Medium* ↑	*Few* ↑	*Std* ↓ (imbalancedness)	All ↑
			linear separability	74.6±0.3	69.7±0.4	66.1±1.2	3.5±0.5	71.2±0.2
			few-shot	56.6±1.2	48.6±0.4	43.7±1.7	5.3±1.2	51.1±0.1
✓			*linear separability*	74.5±0.6	69.2±0.6	66.3±1.1	3.4±0.6	70.9±0.4
			few-shot	55.7±0.4	46.5±0.5	40.2±1.6	6.4±0.4	49.3±0.4
	✓		*linear separability*	74.0±0.9	68.4±0.7	65.5±1.3	3.6±0.3	70.2±0.4
			few-shot	55.0±0.2	46.6±0.3	40.8±1.4	5.8±0.5	49.1±0.3
		✓	*linear separability*	73.6±0.3	68.6±0.8	64.5±0.9	3.8±0.3	70.0±0.4
			few-shot	55.0±0.4	45.8±0.3	39.1±1.1	6.5±0.6	48.5±0.2
✓	✓		*linear separability*	75.8±0.4	69.9±0.3	**69.8±1.3**	**2.9±0.5**	72.2±0.2
			few-shot	**57.7±0.7**	48.0±1.0	**46.4±0.7**	5.0±0.7	51.5±0.3
✓	✓	✓	*linear separability*	**76.1±0.6**	**70.8±0.5**	69.3±0.8	**3.0±0.1**	**72.7±0.4**
			few-shot	57.4±0.6	**48.9±0.2**	**46.3±1.5**	**4.8±0.2**	**51.9±0.4**


random
MAK without diversity promotion
MAK with diversity promotion
20
10
0
−10
−20
−20
−10
0
10
20

(a)

(b)

(c)

Conclusion

- The data sampled from open-world always show a long tail distribution, further hurting the balancedness of contrastive learning.
- We propose a unified sampling framework called MAK. It significantly boosts the balancedness and accuracy of contrastive learning via strategically sampling additional data.
- On the other hand, when applying on real applications, there are also problems like fair or private. This reminds us to carefully check if our method has risk of producing unfair or biased outputs in the future.