[ Paper Review ]

# VL-LTR: Learning Class-wise Visual-Linguistic Representation for Long-Tailed Visual Recognition

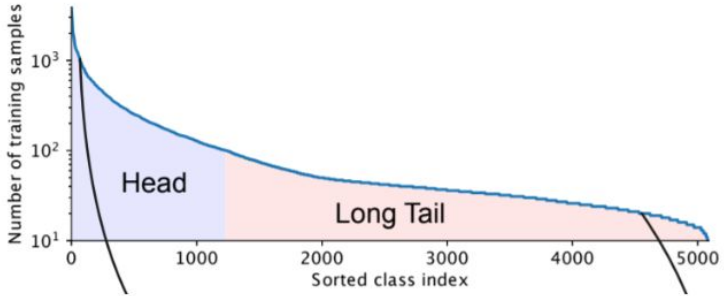
**ECCV 2022**

2023.03 김다혜

# Index

# Introduction

## ❖ Long-tailed distribution in Real-world




- 소수의 head class가 데이터의 대부분을 차지하고, 나머지 tail class는 데이터가 부족한 long-tailed 분포를 띈다.
- 데이터 수가 많은 head class에 편향이 생기고, 데이터가 적은 tail class로 인해 성능이 떨어지게 됨

## ● Long-tailed 문제 해결하기 위한 Previous work

1) re-sampling the training data
2) re-weighting the loss functions
3) transfer learning methods..

➔ 기존의 연구는 **image modality** 에만 의존한 solution 이고, text modal을 불균형 문제에 통합한 시도는 거의 없다.

# Introduction

## ❖ Language modality 활용 가능성

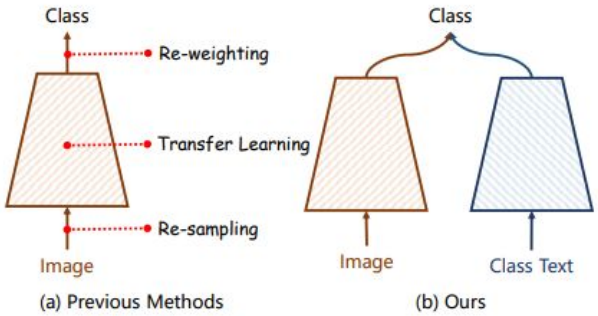
- **Image modality**

  - 구체적이고, low-level 특징을 표현한다. (ex. 모양, 색, 질감 ..)

- **Language modality**

  - 추상적이고, high-level 특징까지 표현 가능하다.
  - 전문가에 의한 사전지식이 포함할 수 있다.

  **=> class 별 표현 학습에 필요한 이미지가 충분하지 않을 때 활용해볼 수 있다**




## VL-LTR

: Learning Class-wise Visual-Linguistic Representation for Long-Tailed Visual Recognition

➔ 본 논문은 long-tailed recognition을 위한 visual-linguistic framework인 **VL-LTR**을 소개한다.

# Introduction

## ❖ Main Component

### 1) Class-wise visual-linguistic pre-training **(CVLP)**

클래스 별 visual-linguistic 관련성을 학습하는 pre-training 과정

- 기존의 pre-train visual-linguistic 모델과 달리, **클래스 별로 표현 학습**함으로써 long-tailed visual recognition 성능을 향상시킴

### 2) Language-guided recognition **(LGR)**

사전학습된 visual-linguistic 표현에 기반한 long-tailed recognition 수행

- Visual recognition에 언어 정보 활용, noisy text 에 강한 method

## ❖ Contribution

1) long-tailed visual recognition에서 텍스트 정보가 이미지 정보를 보충하는 새로운 방법론 제시

2) long-tailed visual recognition의 새로운 프레임워크 제시

**" class-wise text-image pre-training (CVLP) + language-guided recognition (LGR) "**

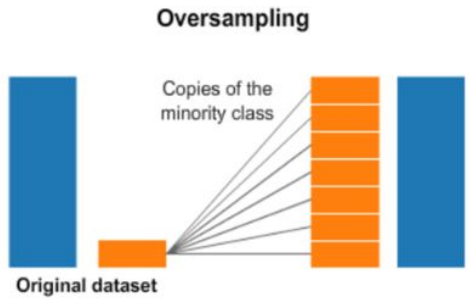
3) 다양한 long-tailed recognition benchmarks (ImageNet-LT, Places-LT, and iNaturalist 2018)에서 SOTA 달성

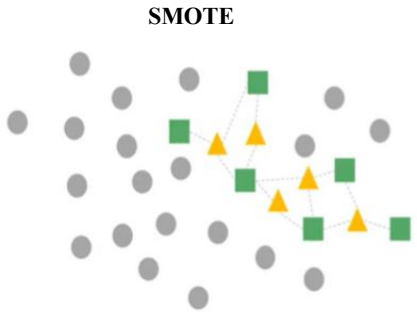
# Related Work

## ❖ Class re-balanced Strategy

### 1) Data Resampling

- head나 tail의 sample 비율을 조정하여 균형을 맞추는 방식
  ex) over/under sampling, smote..
  그러나, augment 된 소수 클래스에서 overfitting 가능성 높음




- overfitting 완화하기 위해,
  다수 클래스의 feature space 에서 소수 클래스를 up-sampling 하거나,
  다수 클래스의 데이터를 소수 클래스 데이터로 변형하여 소수 클래스를 up-sampling 하는 접근 방식 연구됨

# Related Work

## 2) Re-weighting loss function

- 클래스 별 반영 비율을 loss function을 통해 조정하는 방법

  ex) **Focal loss** : 높은 probability로 예측한 sample의 loss에 가중치를 주어 어려운 sample을 보다 잘 학습할 수 있도록 도움

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

  ex) **LDAM loss** : 속한 데이터 개수가 작은 few-shot class가 더 넓은 margin을 가지게 함

## 3) Transfer learning

- 충분한 데이터를 포함한 head class에서 얻은 feature을 이용해, tail class의 representation learning에 이용하는 방법

➔ 세 가지 Class re-balanced strategy 모두 image modality에 한정된 rebalancing method 이다.

# Related Work

## ❖ Previous Visual-Linguistic Model

- Image classification에서의 기존 visual-linguistic 접근 방식
  -> image-text 간 표현 격차 해소에 한계 존재

- 최근 pre-trained Visual-linguistic 모델은 다양한 vision/multimodal task에서 좋은 성능을 보임
  ex. Vinvl, Oscar, Vl-bert ..

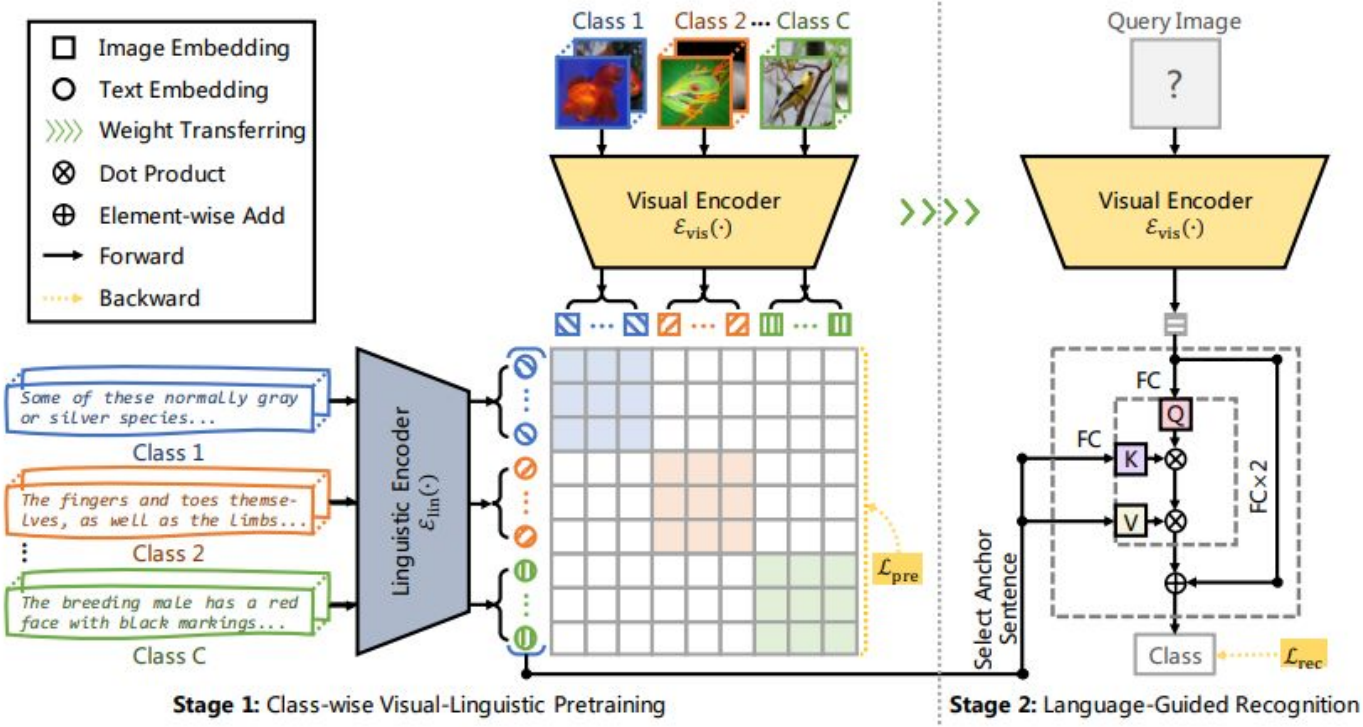
- 또한, **contrastive learning**을 통해 visual-linguistic 표현을 효율적으로 학습한 거대 pre-trained 모델 등장했다.
  ex. CLIP, ALIGN

➔ 본 논문에서는 visual recognition task에서 linguistic modality를 효율적으로 활용하는 **VL-LTR**를 소개한다.

# Methodology

## Overall Architecture

VL-LTR은 two-stage 프레임워크를 갖는다.



# Methodology

## Stage 1) Class-wise Visual-Linguistic Pre-training (CVLP)

**Contrastive Learning 기반 표현 학습**을 활용하여 long-tailed data 에서도 효율적인 학습을 목표로 한다.

- Pre-training 목표
  - Visual-Linguistic representation을 학습하여, **클래스 별 언어적 정보를 visual recognition에 활용**

$$\mathcal{L}_{\text{pre}} = \lambda \mathcal{L}_{\text{ccl}} + (1 - \lambda)\mathcal{L}_{\text{dis}},$$

λ : 두 loss의 비율을 조절하기 위한 하이퍼파라미터

- Pre-train Loss 함수 $\mathcal{L}_{\text{pre}}$는 $\mathcal{L}_{\text{ccl}}$ 와 $\mathcal{L}_{\text{dis}}$ 로 이루어져있다.

# Methodology

## Stage 1) Class-wise Visual-Linguistic Pre-training (CVLP)

$$\mathcal{L}_{\text{ccl}} = \mathcal{L}_{\text{vis}} + \mathcal{L}_{\text{lin}}$$
$$= -\frac{1}{|\mathcal{T}_i^+|}\sum_{T_j \in \mathcal{T}_i^+} \log \frac{\exp(S_{i,j}/\tau)}{\sum_{T_k \in \mathcal{T}} \exp(S_{i,k}/\tau)} - \frac{1}{|\mathcal{I}_i^+|}\sum_{I_j \in \mathcal{I}_i^+} \log \frac{\exp(S_{j,i}/\tau)}{\sum_{I_k \in \mathcal{I}} \exp(S_{k,i}/\tau)},$$

- **Pre-training Process**

1) batch별 image, text 샘플링 하여 각각 visual,linguistic encoder를 통해 이미지, 텍스트 임베딩을 생성한다.

$$E_i^I = \mathcal{E}_{\text{vis}}(I_i), \quad E_i^T = \mathcal{E}_{\text{lin}}(T_i),$$

2) $E_i^I$ $E_i^T$ 간의 코사인 유사도 Si,j 를 구한다.

3) 구한 Si,j 통해 Visual loss와 Linguistic loss 정의하여 최종 $\mathcal{L}_{\text{ccl}}$ 를 정의한다.

=> 구한 loss를 통해 visual / linguistic 인코더 optimizing

# Methodology

## Stage 1) Class-wise Visual-Linguistic Pre-training (CVLP)

- **Distillation**

  제한된 text corpus로 인한 overfitting 방지하기 위해, CLIP에서 pre-train 된 정보를 활용한다.

  => $\mathcal{L}_{\text{dis}}$ 정의

$$\mathcal{L}_{\text{dis}} = -\frac{\exp(S'_{i,i}/\tau)}{\sum_{T_j \in \mathcal{T}} \exp(S'_{i,j}/\tau)} \log \frac{\exp(S_{i,i}/\tau)}{\sum_{T_k \in \mathcal{T}} \exp(S_{i,k}/\tau)} - \frac{\exp(S'_{i,i}/\tau)}{\sum_{I_j \in \mathcal{I}} \exp(S'_{j,i}/\tau)} \log \frac{\exp(S_{i,i}/\tau)}{\sum_{I_k \in \mathcal{I}} \exp(S_{k,i}/\tau)}.$$

S' = CLIP 기반 코사인 유사도 matrix

S = class-wise contrastive learning (CCL)에서 구한 코사인 유사도 matrix

# Methodology

## Stage 1) Class-wise Visual-Linguistic Pre-training (CVLP)

- 최종 Pre loss function

$$\mathcal{L}_{\text{pre}} = \lambda \mathcal{L}_{\text{ccl}} + (1 - \lambda)\mathcal{L}_{\text{dis}},$$

λ : 두 loss의 비율을 조절하기 위한 하이퍼파라미터

- pre-training framework 장점

  class level 이미지 샘플에 대한 text는 독립적이고, 매 반복마다 달라질 수 있다.

  -> fixed image-text pair로 학습할 때보다 **정규화된 모델** 얻을 수 있고, noisy text에 강하다.

# Methodology

## Stage 2) Language-Guided Recognition (LGR)

학습된 visual-linguistic representation 활용하여 image classification 진행할 수 있도록 fine-tuning 하는 과정

### (1) Anchor Sentence Selection

- 인터넷에서 수집한 noise text는 recognition 성능 저해하므로, 가장 구별되는 중심 문장을 선별한다.

- **Process**

1) 클래스 당 최대 50개의 이미지 갖는 image batch I' 생성한다.

2) I' 와 text T 사이의 $\mathcal{L}_{\text{lin}}$ 계산 ( $\mathcal{L}_{\text{lin}}$ = stage1에서 정의한 코사인 유사도 기반 Linguistic loss )

3) 각 이미지에서 $\mathcal{L}_{\text{lin}}$ 값이 가장 작은 M개의 text sentences => **"Anchor sentences"**

# Methodology

## Stage 2) Language-Guided Recognition (LGR)

### (2) Language-Guided Recognition Head

- **LGR Head를 optimizing** 하기 위해, 이미지와 문장의 attention score를 구한다.
  - Q, K, V = Attention 연산에서의 query, key, value
  - Image의 Q와 Anchor sentence의 K,V 를 사용하여 attention score 연산
  - G = M개의 Anchor sentences에 대한 attention score

$$Q = \text{Linear}(\text{LayerNorm}(E^I)),$$
$$K = \text{Linear}(\text{LayerNorm}(E^T)), \quad V = E^T,$$
$$G = \sigma(\frac{QK^\intercal}{\sqrt{D}})V,$$

  - PI, PT = Visual / Linguistic representation에 기반한 classification 확률

Attention score(G) 와 Image embedding 간 코사인 유사도

$$P = P^I + P^T = \sigma(\text{MLP}(E^I)) + \sigma(\langle E^I, G \rangle / \tau).$$

# Methodology

## Stage 2) Language-Guided Recognition (LGR)

- 최종 rec loss function

**Visual / Linguistic Representation에 기반한 확률 (PI, PT)**과 **ground truth label (y)**의 Cross Entropy 연산을 통해 구한 최종 $\mathcal{L}_{\text{rec}}$

$$\mathcal{L}_{\text{rec}} = \mathcal{L}_{\text{CE}}(P^I, \mathbf{y}) + \mathcal{L}_{\text{CE}}(P^T, \mathbf{y}).$$

stage 2 (LGR) 에서는,

1) 앵커 문장 선택 후
2) 구한 앵커 문장과 이미지와의 attention score에 기반한 loss값을 통해 LGR Head를 optimizing 한다.

# Experiments

## ❖ Datasets

- 세 가지 **long-tailed visual recognition benchmarks**을 사용했다.
  - ImageNet-LT, Places-LT, iNaturalist 2018
- 추가적으로, 세 가지 datasets 에 대한 **class-level text descriptions** 을 수집하였다.
  - Wikipedia 에서 class 에 대한 descriptions 수집 후 전처리

## ❖ Settings

- visual encoder : ResNet-50 또는 ViT-Base/16
- linguistic encoder : 12-layer Transformer
- optimizer : AdamW
- pre-training
  - CLIP의 pre-trained weights 사용 ( 50 epochs, mini-batch size = 256 )
- fine-tuning
  - class 마다 64 sentences 선별 ( 50 epochs, mini-batch size = 128 )

# Experiments

## ❖ Results (ImageNet-LT)

### I. 대표적인 long-tailed recognition methods 와 비교

| Method | Backbone | Accuracy (%) | | | |
|---|---|---|---|---|---|
| | | Overall | Many | Medium | Few |
| Cross Entropy [26] | ResNeXt-50 | 44.4 | 65.9 | 37.5 | 7.7 |
| OLTR [29] | ResNeXt-50 | 46.3 | - | - | - |
| SSD [26] | ResNeXt-50 | 56.0 | 66.8 | 53.1 | 35.4 |
| RIDE (4 Experts) [48] | ResNeXt-50 | 56.8 | 68.2 | 53.8 | 36.0 |
| TADE [53] | ResNeXt-50 | 58.8 | 66.5 | 57.0 | 43.5 |
| smDRAGON [39] | ResNeXt-50 | 50.1 | - | - | - |
| ResLT [6] | ResNeXt-101 | 55.1 | 63.3 | 53.3 | 40.3 |
| PaCo [7] | ResNeXt-101 | 60.0 | 68.2 | 58.7 | 41.0 |
| NCM [21] | ResNeXt-152 | 51.3 | 60.3 | 49.0 | 33.6 |
| cRT [21] | ResNeXt-152 | 52.4 | 64.7 | 49.1 | 29.4 |
| $\tau$-normalized [21] | ResNeXt-152 | 52.8 | 62.2 | 50.1 | 35.8 |
| LWS [21] | ResNeXt-152 | 53.3 | 63.5 | 50.4 | 34.2 |
| NCM [21] | ResNet-50* | 49.2 | 58.9 | 46.6 | 31.1 |
| cRT [21] | ResNet-50* | 50.8 | 63.3 | 47.2 | 27.8 |
| $\tau$-normalized [21] | ResNet50* | 51.2 | 60.9 | 48.4 | 33.8 |
| LWS [21] | ResNet-50* | 51.5 | 62.2 | 48.6 | 31.8 |
| Zero-Shot CLIP [37] | ResNet-50* | 59.8 | 60.8 | 59.3 | 58.6 |
| Baseline | ResNet-50* | 60.5 | 74.4 | 56.9 | 34.5 |
| VL-LTR (ours) | ResNet-50* | **70.1** | **77.8** | **67.0** | **50.8** |
| VL-LTR (ours) | ViT-Base* | **77.2** | **84.5** | **74.6** | **59.3** |

(*) = initialized with CLIP weights

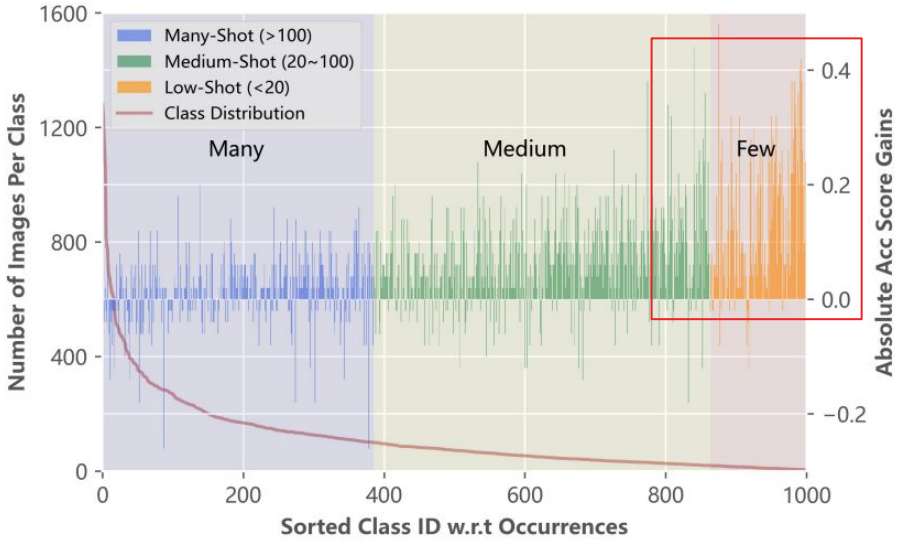
- ResNet-50 backbone 적용 시(70.1%),
  - baseline(60.5%) 보다 9.6% 나은 성능 보임
  - 기존 best model PaCo(60.0%) 보다 10.1% 나은 성능
- few-shot 에서 baseline 보다 16.3% 나은 성능 보임
- **ViT-Base/16 backbone 적용 시 77.2%**
  - ImageNet-LT 에서 SOTA 달성

# Experiments

## ❖ Results (ImageNet-LT)

**II. baseline과 비교했을 때, class 별 성능 개선도**

* baseline = VL-LTR method 에서 ***visual modality 정보만*** 반영한 버전




- **tail classes**에서 더 좋은 accuracy 점수 달성

-> class별 text descriptions 사용이 Long-tailed 문제를 완화하는데 기여했다고 볼 수 있다.

# Experiments

## ❖ Results (Places-LT)

| Method | Backbone | Accuracy (%) | | | |
|---|---|---|---|---|---|
| | | Overall | Many | Medium | Few |
| OLTR [29] | ResNet-152 | 35.9 | 44.7 | 37.0 | 25.3 |
| ResLT [6] | ResNet-152 | 39.8 | 39.8 | 43.6 | 31.4 |
| TADE [53] | ResNet-152 | 40.9 | 40.4 | 43.2 | 36.8 |
| PaCo [7] | ResNet-152 | 41.2 | 36.1 | 47.9 | 35.3 |
| NCM [21] | ResNet-152 | 36.4 | 40.4 | 37.1 | 27.3 |
| cRT [21] | ResNet-152 | 36.7 | 42.0 | 37.6 | 24.9 |
| $\tau$-normalized [21] | ResNet-152 | 37.9 | 37.8 | 40.7 | 31.8 |
| LWS [21] | ResNet-152 | 37.6 | 40.6 | 39.1 | 28.6 |
| smDRAGON [39] | ResNet-50 | 38.1 | - | - | - |
| NCM [21] | ResNet-50* | 30.8 | 37.1 | 30.6 | 19.9 |
| cRT [21] | ResNet-50* | 30.5 | 38.5 | 29.7 | 17.6 |
| $\tau$-normalized [21] | ResNet-50* | 31.0 | 34.5 | 31.4 | 23.6 |
| LWS [21] | ResNet-50* | 31.3 | 36.0 | 32.1 | 20.7 |
| Zero-Shot CLIP [37] | ResNet-50* | 38.0 | 37.5 | 37.5 | 40.1 |
| Baseline | ResNet-50* | 39.7 | 50.8 | 38.6 | 22.7 |
| VL-LTR (ours) | ResNet-50* | **48.0** | **51.9** | **47.2** | **38.4** |
| VL-LTR (ours) | ViT-Base* | **50.1** | **54.2** | **48.5** | **42.0** |

- 실험 setting은 ImageNet-LT와 동일하다.
- ResNet-50 backbone 적용 시 (48.0%),
  - 기존 SOTA 모델 PaCo(41.2%) 보다 나은 성능 보임
- **ViT-Base/16 backbone 적용 시 (50.1%)**
  - Places-LT 에서 SOTA 달성
- **medium, few-shot classes 에서도 좋은 성능** 보임

# Experiments

## ❖ Results : iNaturalist 2018

| Method | Backbone | Accuracy (%) |
|---|---|---|
| CB-Focal [2] | ResNet-50 | 61.1 |
| LDAM+DRW [2] | ResNet-50 | 68.0 |
| BBN [56] | ResNet-50 | 69.6 |
| SSD [26] | ResNet-50 | 71.5 |
| RIDE (4 experts) [48] | ResNet-50 | 72.6 |
| smDRAGON [39] | ResNet-50 | 69.1 |
| ResLT [6] | ResNet-50 | 72.3 |
| TADE [53] | ResNet-50 | 72.9 |
| PaCo [7] | ResNet-50 | 73.2 |
| NCM [21] | ResNet-50 | 63.1 |
| cRT [21] | ResNet-50 | 67.6 |
| $\tau$-normalized [21] | ResNet-50 | 69.3 |
| LWS [21] | ResNet-50 | 69.5 |
| NCM [21] | ResNet-50* | 65.3 |
| cRT [21] | ResNet-50* | 69.9 |
| $\tau$-normalized [21] | ResNet-50* | 71.2 |
| LWS [21] | ResNet-50* | 71.0 |
| Zero-Shot CLIP [37] | ResNet-50* | 3.4 |
| Baseline | ResNet-50* | 72.6 |
| VL-LTR (ours) | ResNet-50* | **74.6** |
| PaCo [7] | ResNet-152 | 75.2 |
| DeiT-B/16 [45] | - | 73.2 |
| DeiT-B/16-384 [45] | - | 79.5 |
| VL-LTR (ours) | ViT-Base* | **76.8** |
| VL-LTR-384 (ours) | ViT-Base* | **81.0** |

- pre-trained for 100 epochs, and fine-tuned for 360 epochs
- ResNet-50 backbone 적용 시 (74.6%), 기존 모델보다 나은 성능
- **ViT-Base/16 backbone 적용 시 (76.8%)**
  - 기존 SOTA 모델 PaCo(75.2%) 보다 나은 성능 보임

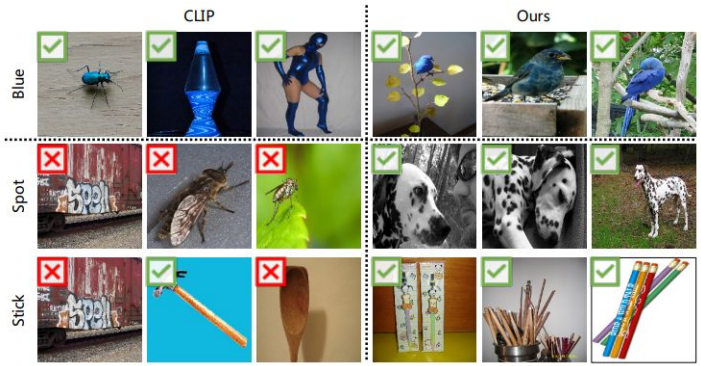
# Experiments

## Ablation Study

Class-wise Visual-Linguistic 사전학습을 수행하지 않았을 때의 결과와 비교하는 실험을 진행했다.

| # | CLIP Weights | Pre-training | | Fine-tuning | | Accuracy (%) |
|---|---|---|---|---|---|---|
| | | w/o $\mathcal{L}_{dis}$ | w/ $\mathcal{L}_{dis}$ | Head | SS | |
| 1 | ✓ | - | ✓ | LGR | AnSS | **70.1** |
| 2 | ✓ | - | - | LGR | AnSS | 62.8 |
| 3 | - | ✓ | - | LGR | AnSS | 46.8 |
| 4 | ✓ | ✓ | - | LGR | AnSS | 66.2 |
| 5 | ✓ | - | ✓ | FC | - | 62.1 |
| 6 | ✓ | - | ✓ | KNN | - | 63.9 |
| 7 | ✓ | - | ✓ | LGR | Cut Off | 69.7 |

- CVLP framework 제거했을 때 성능이 떨어지는 것을 확인할 수 있다.

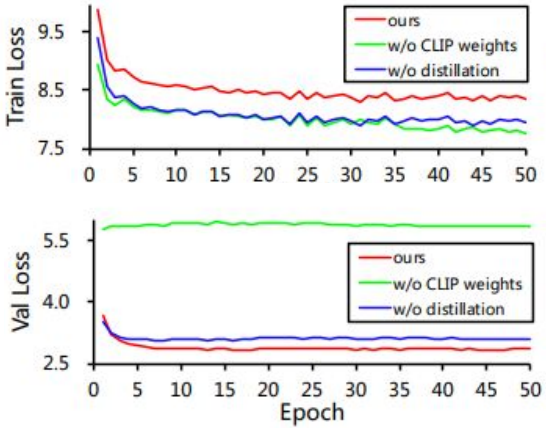



- VL-LTR vs CLIP
  - **rare 한 컨셉**을 인식할 때, (ex. "spot", "stick"), **class level에서 pre-train 한 VL-LTR**이 CLIP보다 잘 맞추는 것을 확인할 수 있다.

# Experiments

## Ablation Study : CLIP Pre-trained Weights & Distillation Loss

| # | CLIP Weights | Pre-training | | Fine-tuning | | Accuracy (%) |
|---|---|---|---|---|---|---|
| | | w/o $\mathcal{L}_{dis}$ | w/ $\mathcal{L}_{dis}$ | Head | SS | |
| 1 | ✓ | - | ✓ | LGR | AnSS | **70.1** |
| 2 | ✓ | - | - | LGR | AnSS | 62.8 |
| 3 | - | ✓ | - | LGR | AnSS | 46.8 |
| 4 | ✓ | ✓ | - | LGR | AnSS | 66.2 |
| 5 | ✓ | - | ✓ | FC | - | 62.1 |
| 6 | ✓ | - | ✓ | KNN | - | 63.9 |
| 7 | ✓ | - | ✓ | LGR | Cut Off | 69.7 |




- CLIP의 pre-trained weights를 적용한 VL-LTR 모델의 성능이 더 좋은 것을 확인 (#1, #3)
- Distillation을 수행한 모델의 성능이 더 높음 (#1, #4)
- training 과 validation loss 비교 (오른쪽 그림)
  - Fine tuning 단계에서 CLIP pre-trained weights와 Distillation Loss가 overfitting을 완화시킨다.
  - ImageNet-LT의 text description 만으론 text corpus가 제한적이기 때문

# Experiments

## Ablation Study : Anchor Sentence Selection

| # | CLIP Weights | Pre-training | | Fine-tuning | | Accuracy (%) |
|---|---|---|---|---|---|---|
| | | w/o $\mathcal{L}_{dis}$ | w/ $\mathcal{L}_{dis}$ | Head | SS | |
| 1 | ✓ | - | ✓ | LGR | AnSS | **70.1** |
| 2 | ✓ | - | - | LGR | AnSS | 62.8 |
| 3 | - | ✓ | - | LGR | AnSS | 46.8 |
| 4 | ✓ | ✓ | - | LGR | AnSS | 66.2 |
| 5 | ✓ | - | ✓ | FC | - | 62.1 |
| 6 | ✓ | - | ✓ | KNN | - | 63.9 |
| 7 | ✓ | - | ✓ | LGR | Cut Off | 69.7 |

- AnSS(Anchor Sentence Selection) 을 "Cuf Off"로 대체 (= 단순히 처음 M개의 문장을 앵커문장으로 선별) 했을 때, 성능이 떨어짐
  - AnSS가 noisy한 문장을 필터링 한 것을 보여준다.
  - “AnSS”는 training-free module 이므로, **noisy problem을 해결 할 수 있는 새로운 가능성 제시**

# Conclusion

## ❖ Summerization

VL-LTR

- Long-tailed recognition에서 새로운 visual-linguistic framework를 제안한다.

  1) class-level visual-linguistic pre-training (CVLP)

  => 이미지와 설명텍스트를 class level에서 matching시켜서 학습한다.

  2) language-guided recognition (LGR) head

  => Visual recognition에 visual-linguistic representation을 활용한다.

- image, text 두 modal을 사용하여 class imbalance 문제를 해결한 새로운 접근 방식으로,

  다양한 **long-tailed recognition benchmarks 에서 기존의 vision-based methods 보다 좋은 성능 달성**했다.

## ❖ Limitations

- 언어적 표현 학습시킬 때 **기존의 pre-trained model (CLIP)에 의존**해야 하므로 Text corpus가 제한적이다.
- Two-stage LTR method를 발전시켜, end-to-end 학습 방법에 대한 연구가 필요해보인다.