KUBIG
DATA SCIENCE & AI

눈빛만 봐도 알 수 있잖아~♫

CLIP model을 활용한 감정 인식

Team | SO⊐TA
19기 박세훈 배지원 정하연

KUBIG
DATA SCIENCE & AI

CONTENTS

KUBIG
DATA SCIENCE & AI

01. 프로젝트 소개

01. 주제 선정

- 프로젝트 목표: 학습에 사용되지 않은 복잡하고 미묘한 감정에 대한 zero-shot learning
- Fine-tuning CLIP Model
- 프로젝트 선정 배경:
  - 심리 상담자, 혹은 face ID 사용자의 정확한 심리 분석
  - 다양한 감정으로 레이블된 표정 데이터셋 부족
  - Classification 과제를 위한 멀티모달 모델 파인튜닝:
    텍스트 ↔ 레이블 ↔ 이미지 매핑을 통하여 텍스트 간 유사도를 이미지 피쳐 간 유사도로 확장할 수 있는 모델 기대

02. 프로젝트 진행 과정

데이터 Augmentation

- 7가지 감정 레이블의 얼굴 표정 이미지 데이터 수집
- 배치당 감정별 이미지 개수 균형 맞춤

모델 구조 파인튜닝

- Embedding 모델의 레이어 깊이 추가 시도

모델 파라미터 파인튜닝

- Ray Tune를 이용한 파라미터 튜닝
- learning_rate, logit_scale, embedding_size, 모델 가중치 초기화 여부 등

KUBIG
DATA SCIENCE & AI

02. 연구 방법

01. 데이터 소개

FER-2013 (Facial Expression Recognition 2013)

- 35,887개의 흑백 이미지
- 7개의 감정 클래스

  (화남, 혐오, 두려움, 행복, 슬픔, 놀람, 중립)

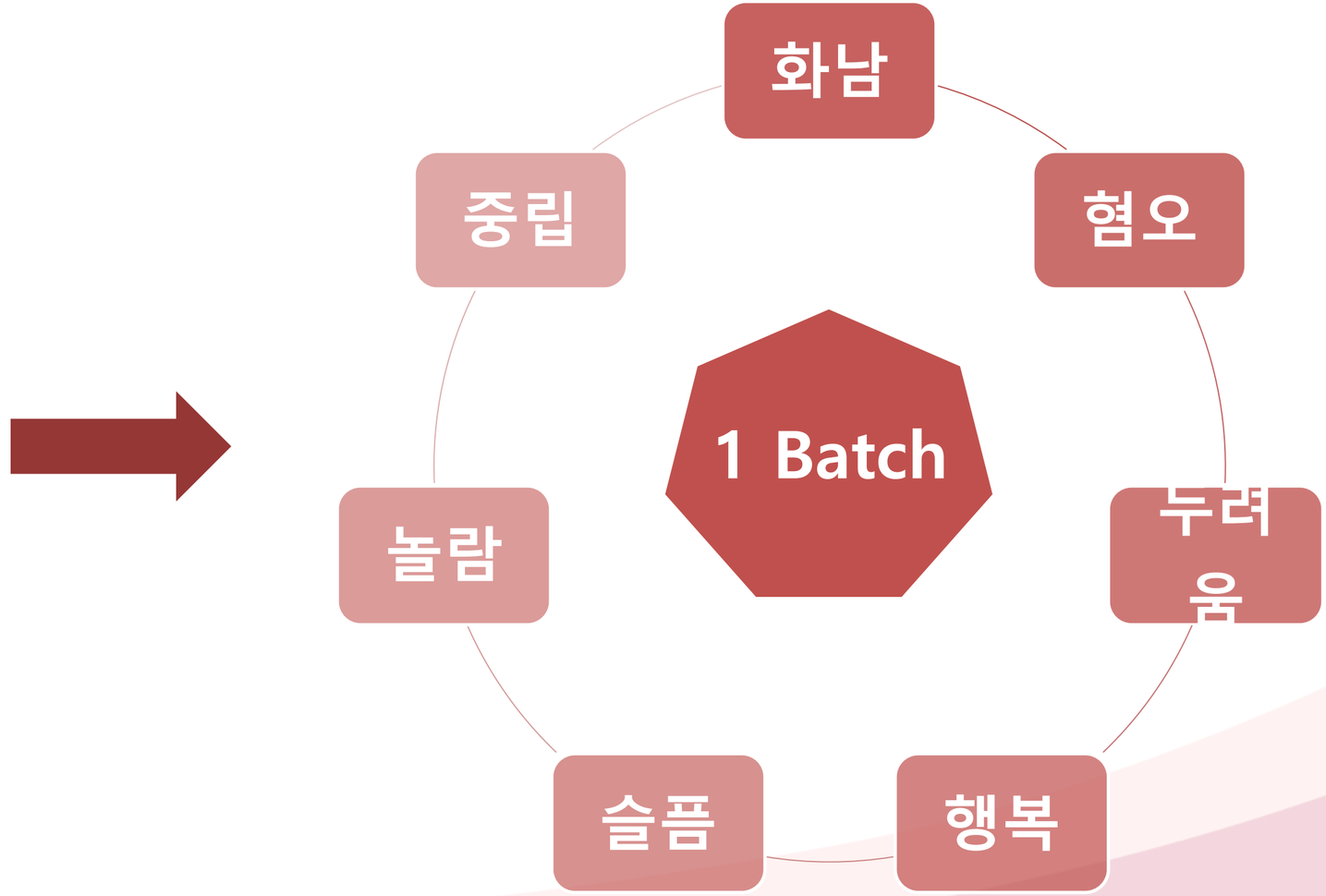
AffectNet(Affect-in-the-Wild Database, 2017)

- 1,000,000개의 이미지
- 8개의 감정 클래스

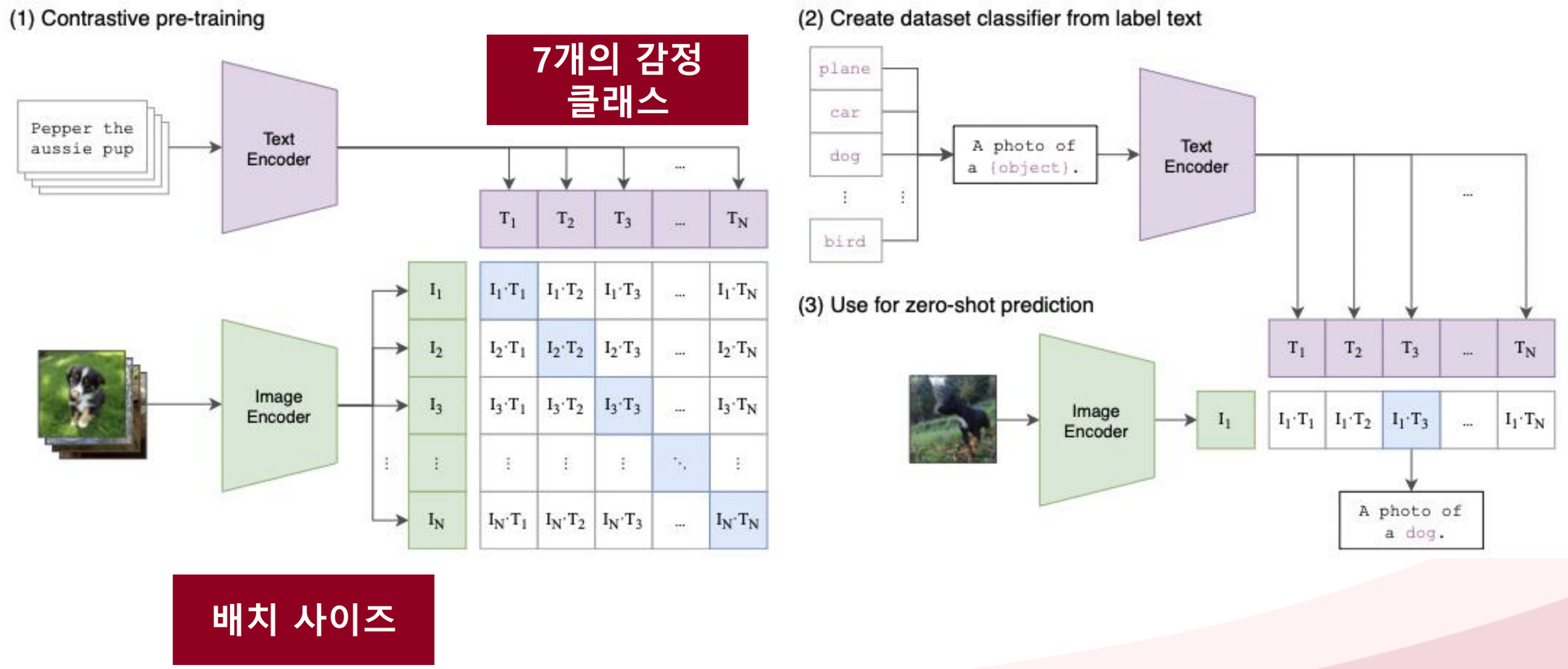
  (화남, 혐오, 두려움, 행복, 슬픔, 놀람, 중립, 경멸)

02. 데이터 전처리 및 균형

Kaggle Dataset API



03. Model Architecture



7개의 감정 클래스

배치 사이즈

KUBIG
DATA SCIENCE & AI

03. 튜닝 시도

01. Small dataset

1. 256개의 이미지로 구성된 small dataset 정의

```python
# 라벨 기반 샘플링을 위한 Sampler 정의
class LabelSampler(Sampler):
    def __init__(self, dataset):
        self.dataset = dataset
        self.labels = [sample[1] for sample in dataset]  # Assuming dataset returns (data, label)
        self.label_to_indices = {label: [] for label in set(self.labels)}
        for idx, label in enumerate(self.labels):
            self.label_to_indices[label].append(idx)

    def __iter__(self):
        indices = []
        for label in self.label_to_indices:
            indices.extend(self.label_to_indices[label])
        random.shuffle(indices)
        return iter(indices)

    def __len__(self):
        return len(self.dataset)
```

```python
small_train_dataset = create_small_dataset(train_dataset, 256)
small_val_dataset = create_small_dataset(val_dataset, 256)
small_test_dataset = create_small_dataset(test_dataset, 256)

batch_size = 7  # 고유한 라벨의 수
small_train_loader = sampler_loader(small_train_dataset, batch_size=batch_size)
small_val_loader = sampler_loader(small_val_dataset, batch_size=batch_size)
small_test_loader = sampler_loader(small_test_dataset, batch_size=batch_size)
```

2. BabySitting model

train, validation loss가 정상적으로 줄어드는지

```
Epoch 1/100, Train Loss: 2.2272, Train Accuracy: 0.1797, Validation Loss: 2.4244, Validation Accuracy: 0.1719
Epoch 2/100, Train Loss: 2.2000, Train Accuracy: 0.1836, Validation Loss: 2.4803, Validation Accuracy: 0.1719
Epoch 3/100, Train Loss: 2.1243, Train Accuracy: 0.2109, Validation Loss: 2.4794, Validation Accuracy: 0.1523
Epoch 4/100, Train Loss: 2.0474, Train Accuracy: 0.2266, Validation Loss: 2.4589, Validation Accuracy: 0.1445
Epoch 5/100, Train Loss: 1.9482, Train Accuracy: 0.2344, Validation Loss: 2.5306, Validation Accuracy: 0.1758
Epoch 6/100, Train Loss: 1.8832, Train Accuracy: 0.2539, Validation Loss: 2.4391, Validation Accuracy: 0.1836
Epoch 7/100, Train Loss: 1.8194, Train Accuracy: 0.2812, Validation Loss: 2.5958, Validation Accuracy: 0.1562
Epoch 8/100, Train Loss: 1.8132, Train Accuracy: 0.2695, Validation Loss: 2.6259, Validation Accuracy: 0.1836
Epoch 9/100, Train Loss: 1.9023, Train Accuracy: 0.2773, Validation Loss: 2.4313, Validation Accuracy: 0.1602
Epoch 10/100, Train Loss: 1.9544, Train Accuracy: 0.2383, Validation Loss: 2.2946, Validation Accuracy: 0.2148
Epoch 11/100, Train Loss: 1.8016, Train Accuracy: 0.2812, Validation Loss: 2.3049, Validation Accuracy: 0.1953
Epoch 12/100, Train Loss: 1.7070, Train Accuracy: 0.2734, Validation Loss: 2.3304, Validation Accuracy: 0.1992
Epoch 13/100, Train Loss: 1.7140, Train Accuracy: 0.2695, Validation Loss: 2.3845, Validation Accuracy: 0.2148
Epoch 14/100, Train Loss: 1.6799, Train Accuracy: 0.2695, Validation Loss: 2.4482, Validation Accuracy: 0.1953
Epoch 15/100, Train Loss: 1.6662, Train Accuracy: 0.2812, Validation Loss: 2.4303, Validation Accuracy: 0.1914
Epoch 16/100, Train Loss: 1.6552, Train Accuracy: 0.2812, Validation Loss: 2.4408, Validation Accuracy: 0.1836
Epoch 17/100, Train Loss: 1.6070, Train Accuracy: 0.2969, Validation Loss: 2.4305, Validation Accuracy: 0.1914
Epoch 18/100, Train Loss: 1.7519, Train Accuracy: 0.2930, Validation Loss: 2.4551, Validation Accuracy: 0.1992
Epoch 19/100, Train Loss: 1.6445, Train Accuracy: 0.2969, Validation Loss: 2.4667, Validation Accuracy: 0.2148
Epoch 20/100, Train Loss: 1.6099, Train Accuracy: 0.2891, Validation Loss: 2.4501, Validation Accuracy: 0.2109
Epoch 21/100, Train Loss: 1.6086, Train Accuracy: 0.2891, Validation Loss: 2.4802, Validation Accuracy: 0.1914
Epoch 22/100, Train Loss: 1.5713, Train Accuracy: 0.3203, Validation Loss: 2.4648, Validation Accuracy: 0.1914
Epoch 23/100, Train Loss: 1.6903, Train Accuracy: 0.2812, Validation Loss: 2.4957, Validation Accuracy: 0.1875
Epoch 24/100, Train Loss: 1.5985, Train Accuracy: 0.3008, Validation Loss: 2.5040, Validation Accuracy: 0.1992
Epoch 25/100, Train Loss: 1.5420, Train Accuracy: 0.3203, Validation Loss: 2.4536, Validation Accuracy: 0.1914
Epoch 26/100, Train Loss: 1.6779, Train Accuracy: 0.2891, Validation Loss: 2.4943, Validation Accuracy: 0.2109
Epoch 27/100, Train Loss: 1.6667, Train Accuracy: 0.2695, Validation Loss: 2.5045, Validation Accuracy: 0.1719
```

02. Hyperparameter Tuning

3. Learning rate, logit_scale, embedding_size에 대해 최적 하이퍼파라미터 조합

Ray Tune은 딥러닝 모델의 하이퍼파라미터 튜닝을 자동화·병렬화할 수 있는 라이브러리로, 다양한 최적화 알고리즘을 제공하여 최적의 모델 설정을 빠르게 탐색 가능. 이를 통해 실험 과정을 간소화하고 모델 성능을 효과적으로 향상할 것으로 기대

⋮

```
Trial status: 10 ERROR
Current time: 2024-08-24 12:14:37. Total running time: 30s
Logical resource usage: 0/12 CPUs, 0/1 GPUs (0.0/1.0 accelerator_type:A100)
+-----------------------------------------------------------------------------------------------+
| Trial name                status       ...al_projection_dim     logit_scale            lr |
+-----------------------------------------------------------------------------------------------+
| train_cifar_5bc46_00000   ERROR                         128        0.807396   1.47407e-05 |
| train_cifar_5bc46_00001   ERROR                         128        1.33589    0.000539544 |
| train_cifar_5bc46_00002   ERROR                         128        0.940801   1.44959e-05 |
| train_cifar_5bc46_00003   ERROR                           7        0.365229   1.17869e-05 |
| train_cifar_5bc46_00004   ERROR                           7        1.80003    7.4061e-05  |
| train_cifar_5bc46_00005   ERROR                           7        0.960642   2.5974e-05  |
| train_cifar_5bc46_00006   ERROR                         128        0.955036   8.67599e-05 |
| train_cifar_5bc46_00007   ERROR                           7        0.950397   5.07361e-05 |
| train_cifar_5bc46_00008   ERROR                         128        0.827466   0.000111792 |
| train_cifar_5bc46_00009   ERROR                           7        1.60116    2.65117e-05 |
+-----------------------------------------------------------------------------------------------+
```

02. Hyperparameter Tuning

3. Learning rate, logit_scale, embedding_size에 대해 최적 하이퍼파라미터 조합 탐색

```
learning_rate_min, learning_rate_max = 1e-5, 1e-3
logit_scale_min, logit_scale_max = 1, 4
hidden_size_min, hidden_size_max = 2, 128

combinations = 64
comb_results = {}                    # {0:(1,4), 1:(1,4), ... }


for idx in range(combinations):
    lr = random.uniform(learning_rate_min, learning_rate_max)
    logit_scale = random.uniform(logit_scale_min, logit_scale_max)
    hidden_size = random.randint(hidden_size_min, hidden_size_max)

    init_model()
    model = VisionTextModel(vit_model, kobert_model, visual_projection_dim=hidden_size,
                            text_projection_dim=hidden_size, logit_scale=logit_scale).to(device)
```

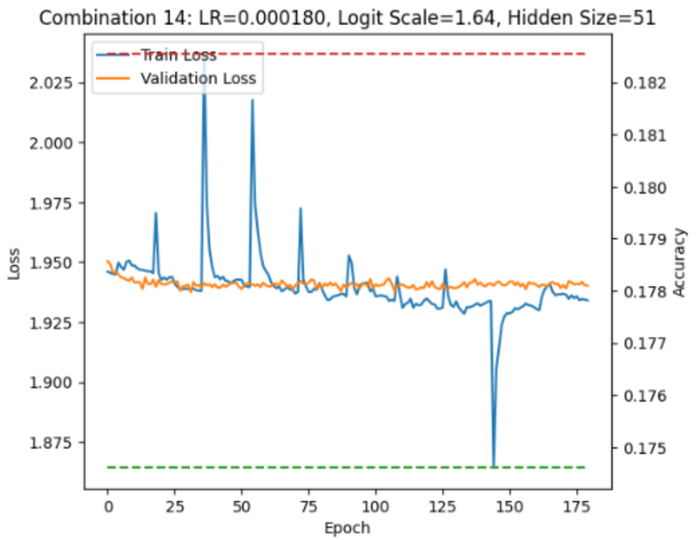
직접 구현 !!

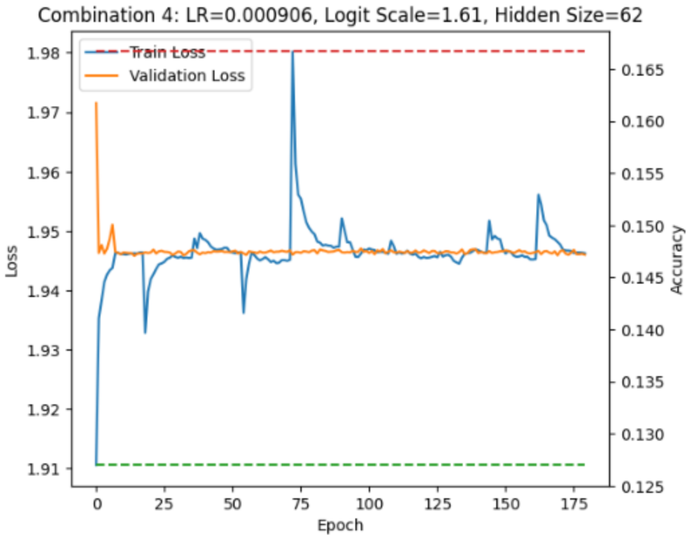
Accuracy

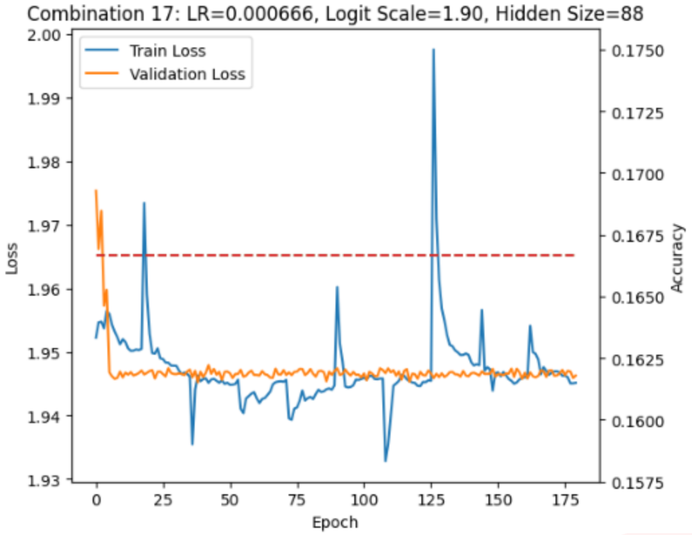
Learning rate　　Logit_scale　　Embedding_size

03. Loss and Accuracy

4. Parameter 조합 별 학습 결과







⇒ 학습이 제대로 진행되지 않음 … SO≠TA

04. Test results

KOTE Dataset에서 분류된 44개의 정서 label 중,

학습에 사용한 7가지 감정을 제외한 label 별 이미지 dataset 구축하고자 함

:“당황하는”

:“한심한”

:“비장한”

※ KOTE 감정 레이블: ['불평/불만', '환영/호의', '감동/감탄', '지긋지긋', '고마움', '슬픔', '화남/분노', '존경', '기대감', '우쭐댐/무시함', '안타까움/실망', '비장함', '의심/불신', '뿌듯함', '편안/쾌적', '신기함/관심', '아껴주는', '부끄러움', '공포/무서움', '절망', '한심함', '역겨움/징그러움', '짜증', '어이없음', '없음', '패배/자기혐오', '귀찮음', '힘듦/지침', '즐거움/신남', '깨달음', '죄책감', '증오/혐오', '흐뭇함(귀여움/예쁨)', '당황/난처', '경악', '부담/안_내킴', '서러움', '재미없음', '불쌍함/연민', '놀람', '행복', '불안/걱정', '기쁨', '안심/신뢰']

KUBIG
DATA SCIENCE & AI

Thank You