

Few-shot learning을 이용한 아몬드 이미지 분류 성능 분석

작성자 : @astra.ka(가을)/인공지능

소속 : 카카오테크 부트캠프 2기 생성형AI 과정 1차-12조

작성일 : 2025.03.26

Contents

[Contents](#)

[Abstract](#)

[I. Introduction](#)

[Background](#)

[Related Work](#)

[Contribution](#)

[II. Proposed method](#)

[Overview](#)

[Few-shot learning](#)

[Prototypical Network](#)

[Backbone - resnet18](#)

[III. Datasets](#)

[Cross-Dataset Evaluation](#)

[Training dataset: CIFAR-100](#)

[Test dataset : Almond Varieties dataset](#)

[데이터전처리](#)

[IV. Experiment](#)

[실험 환경 변수](#)

[DB 연결](#)

[Pretrained Dataset](#)

[실험 개요](#)

[하이퍼파라미터 구성](#)

[실험 모델 구성표](#)

[V. Result](#)

[평가 기준](#)

[실험 별 결과](#)

[References](#)

[Acknowledge](#)

Abstract

I. Introduction

Background

아몬드 품질 분류는 유통과 가공에 필수적이며, 정확한 분류는 소비자 신뢰와 산업 발전에 중요합니다.

Related Work

기존 아몬드 연구들은 데이터 부족으로 세부적인 분류 연구에 한계가 있었습니다.

Contribution

데이터 부족 문제를 해결하기 위해, 소량의 데이터로도 효과적인 분류가 가능한 퓨샷 러닝 기법을 적용하였습니다.

II. Proposed method

Overview

graph TD

Start[실험 시작] → Pretrained[Pretrained 모델 사용
ResNet18]

Pretrained → |Train 없음| ZeroTrain[Zero-Trained Inference Only]

Pretrained → |Train 있음| MetaTraining[Meta-Training 수행]

ZeroTrain → Z1["1-shot"]

ZeroTrain → Z2["5-shot"]

MetaTraining → MBS1["Meta-Batch Size = 1"]

MetaTraining → MBS4["Meta-Batch Size = 4"]

MetaTraining → MBS8["Meta-Batch Size = 8"]

MBS1 → M11["1-shot"]

MBS1 → M12["5-shot"]

MBS4 → M41["1-shot"]

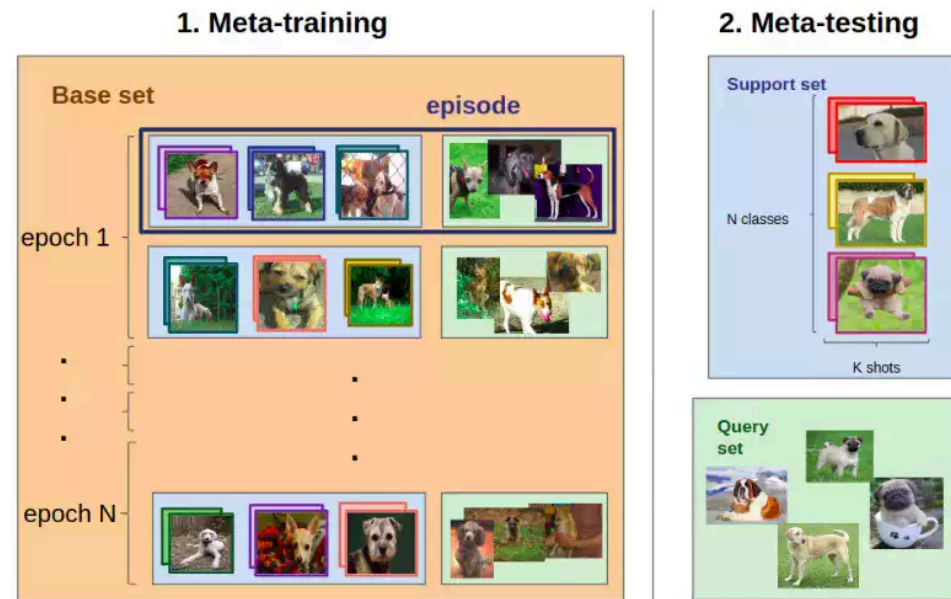
MBS4 → M42["5-shot"]

MBS8 → M81["1-shot"]

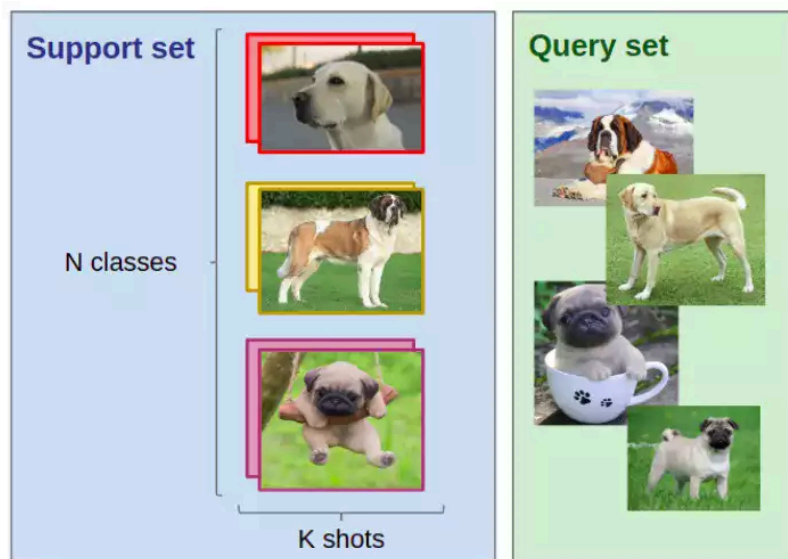
MBS8 → M82["5-shot"]

Few-shot learning

소량의 학습 데이터로도 모델이 새로운 작업을 수행하도록 학습하는 머신 러닝 기법입니다.



meta learning

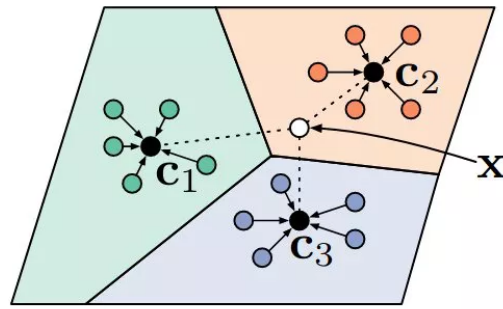


N-way K-shot image classification task

Prototypical Network

논문 링크 : <https://arxiv.org/abs/1703.05175>

각 클래스의 프로토타입을 평균 벡터로 정의하고, 새로운 샘플과의 거리를 기반으로 분류하는 모델입니다.



prototypical network 프로세스 개요

Backbone - resnet18

- transfer learning : imagenet 데이터세트로 훈련시킨 파라미터를 적용했습니다.

III. Datasets

Cross-Dataset Evaluation

본 연구는 few-shot learning을 통해 아몬드 이미지를 분류하고자 하였습니다. few-shot learning의 핵심은 소량의 학습 데이터를 기반으로도 새로운 클래스를 잘 구분하는 데 있으며, 이를 위해 이미지 임베딩 능력을 강화할 수 있도록 training을 적용하였습니다.

하지만 동일한 데이터세트 내에서 train과 test를 나누어 실험할 경우, 다음과 같은 의문이 들었습니다.

few-shot learning의 training 과정을 통해 모델이 정말 일반화되었는가, 아니면 단순히 해당 데이터세트에 적응한 것인가?

데이터는 결국 사람이 수집하기 때문에, 수집자의 의도나 무의식적인 습관으로 인해 구도나 배경 등에서 유사성이 발생할 수 있습니다. 이로 인해 모델이 '보지 않은 클래스'를 잘 구분할 수 있도록 임베딩을 학습한 것인지, 아니면 데이터세트의 특성에 맞춰 최적화된 것인지 구분하기 어려운 문제가 있습니다.

이러한 한계를 극복하고 모델의 일반화 성능을 보다 정직하게 평가하기 위해, 학습 데이터셋과 테스트 데이터셋을 아예 다른 데이터로 구성하는 **크로스-데이터셋 평가(Cross-Dataset Evaluation)** 방법을 적용하였습니다. 이는 모델이 특정 데이터세트의 특성에 과적합되지 않고, 새로운 환경에서도 잘 작동하는지 확인할 수 있는 방법입니다.

본 연구에서는 학습 데이터셋으로 **CIFAR-100**을, 테스트 데이터셋으로 **아몬드 품종 데이터셋**을 사용하였습니다. 두 데이터셋에 대한 자세한 설명은 아래에 기술하였습니다.

Training dataset: CIFAR-100

<https://www.cs.toronto.edu/~kriz/cifar.html>

CIFAR-100은 100개의 클래스로 구성된 총 60,000개의 32x32 크기의 컬러 이미지로 이루어진 데이터셋입니다. 각 클래스는 600개의 이미지를 포함하며, 500개는 학습용, 100개는 테스트용으로 제공됩니다. 이 데이터셋은 다양한 객체를 포함하고 있어 일반적인 이미지 분류 모델의 학습에 널리 사용됩니다.

Test dataset : Almond Varieties dataset

<https://www.kaggle.com/datasets/mahyeks/almond-varieties/data>

<https://link.springer.com/article/10.1007/s00217-024-04562-4>

테스트 데이터셋으로는 터키의 다양한 지역에서 수집된 Ak, Nurlu, Kapadokya, Sira 등 4가지 아몬드 품종의 이미지를 포함하는 데이터셋을 사용하였습니다. 총 1,556개의 이미지가 포함되어 있으며, 균일한 조명 조건에서 촬영되었습니다. 이러한 이미지는 아몬드 품종 분류의 정확도를 평가하는 데 활용되었습니다.

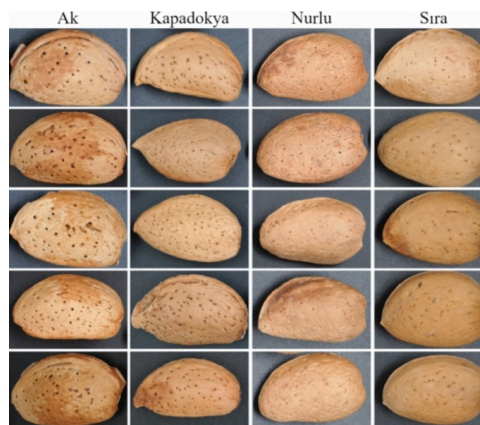


그림 1

아몬드 데이터 세트 샘플 이미지

표 1은 아몬드 품종의 수량 값에 대한 세부 정보를 제공합니다. 데이터 세트는 Kaggle 플랫폼 (<https://github.com/ymyurdakul/datasets/blob/main/almond/archive.zip>)에서 공개적으로 제공됩니다.

Table 1 Almond varieties distribution

From: *Almond (Prunus dulcis)* varieties classification with genetic designed lightweight CNN architecture

Variety	Ak	Kapadokya	Nurlu	Sira	Total
Count	401	465	306	384	1556

데이터전처리

두 데이터셋의 이미지를 ResNet 모델에 적합하도록 224x224 크기로 조정하고, RGB 3채널로 통일하였습니다. 이를 통해 일관된 입력 형식을 유지하여 모델의 학습과 평가를 효과적으로 수행할 수 있었습니다.

IV. Experiment

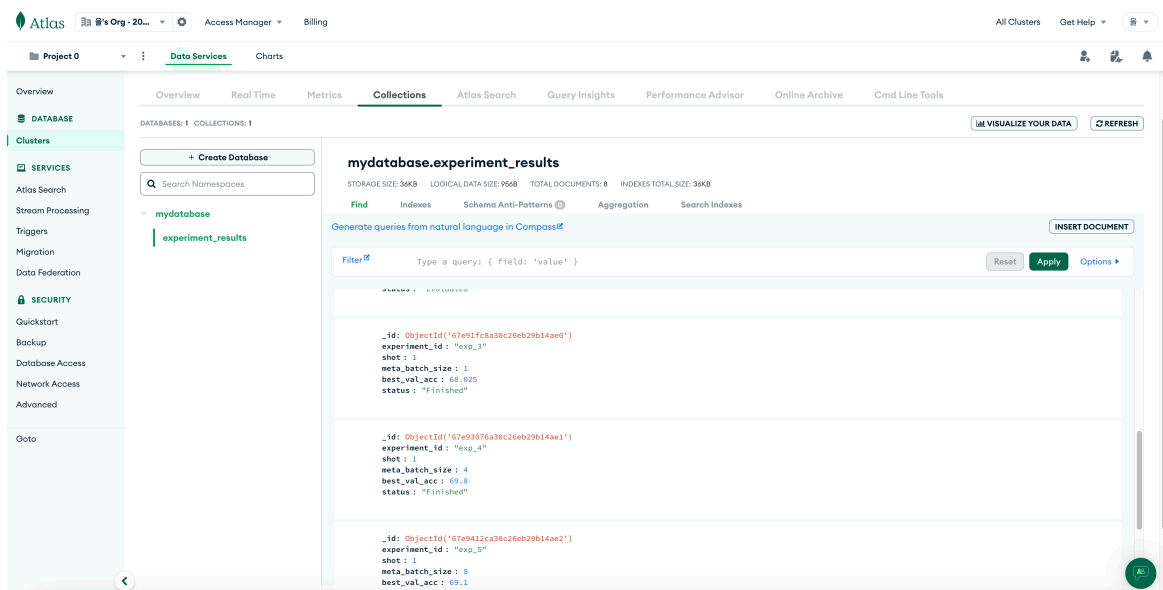
실험 환경 변수

실험은 Google Colab Pro+ 환경에서 진행되었으며, GPU는 NVIDIA T4를 사용하였습니다. 해당 환경은 중간 수준의 딥러닝 실험을 수행하기에 적합하며, 빠른 학습과 반복 실험에 용이합니다.

DB 연결

데이터셋 로딩 및 실험 결과 저장을 위해 MongoDB 와 연결하였습니다. 연결 상태는 스크린샷으로 기록하였으며, 실험 과정의 재현성과 안정성을 확보하였습니다.

▼ MongoDB atlas 스크린샷



Pretrained Dataset

사전학습에는 `torchvision.models.ResNet18_Weights.DEFAULT` 를 사용하였습니다. 이는 ImageNet의 1000개 클래스에 대해 학습된 가중치로, 도토리(acorn)가 포함되어 있습니다. 아몬드와는 시각적·의미적으로 차이가 크다고 판단하여 본 실험에 포함시켰습니다.

실험 개요

ResNet18을 백본으로 사용하는 Prototypical Network 기반 few-shot learning 실험을 수행하였습니다. 실험의 목적은 다양한 shot 수와 meta batch size 조합에 따른 임베딩 성능을 비교하고, pretrained 여부에 따른 영향을 분석하는 것입니다.

하이퍼파라미터 구성

모든 이미지는 224x224 크기로 resize하였고, backbone은 ImageNet 사전학습된 ResNet18입니다. shot 수는 1과 5를 실험하였으며, meta batch size는 1, 4, 8로 설정하였습니다. optimizer는 Adam, learning rate는 1e-3에서 step decay를 적용하였습니다.

실험 모델 구성표

다양한 설정을 실험하기 위해 pretrained 여부, train 적용 여부, shot 수, meta batch size를 조합하였습니다. 총 8개의 실험을 구성하였으며, 각 설정은 아래 표에 정리되어 있습니다. 하이퍼파라미터 최적화는 grid search로 수행하였습니다.

Table2. 실험 모델 구성 표

num	pretrained	train	test	shot	meta batch size
1	Y	N	Y	1	-
2				5	-
3	Y	Y	Y	1	1
4				1	4
5				1	8
6				5	1
7				5	4
8				5	8

▼ 중간보고용 실험

실험모델 1)

→ few shot learning중에서 meta-testing만 우선 진행

```
N_WAY = len(dataset.classes) #아몬드 클래스 수 == 4
N_SHOT = 5 #support set의 클래스당 이미지 수
N_QUERY = 10 #query set의 클래스당 이미지 수
N_EVALUATION_TASKS = len(test_dataset) # → 312
```

V. Result

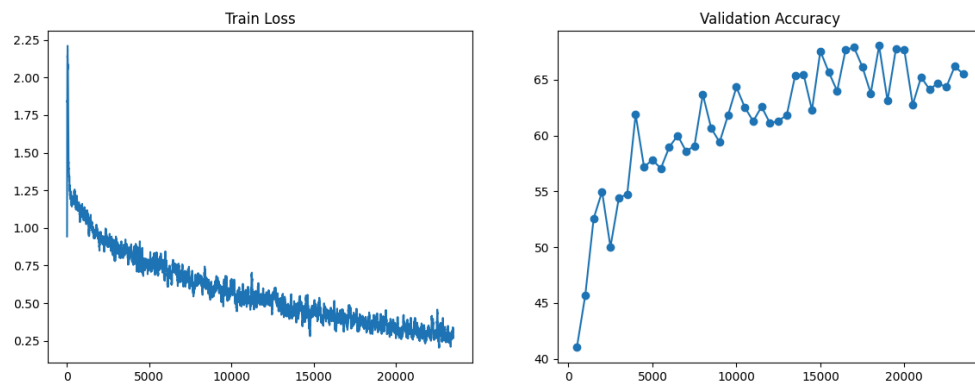
평가 기준

모델의 일반화 성능 평가를 위해 early stopping을 적용하였으며, 검증 성능이 10회 연속 개선되지 않을 경우 학습을 중단하였습니다.

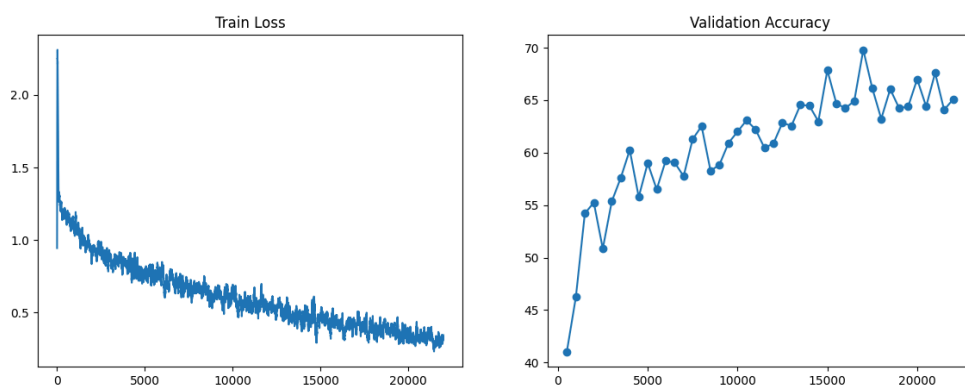
실험 별 결과

Train Loss, Val_Acc

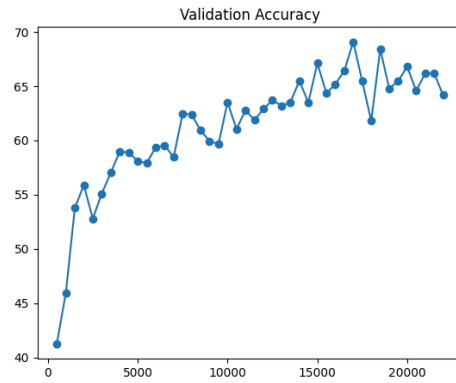
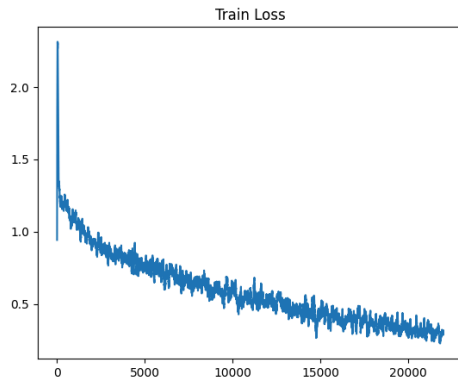
- exp_1, exp_2 : None
- exp_3 :



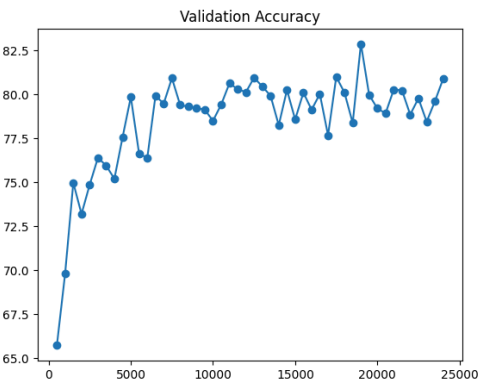
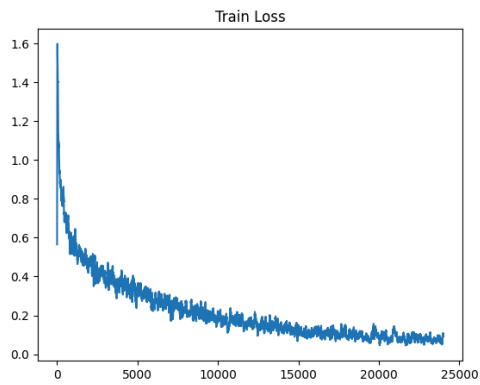
- exp_4



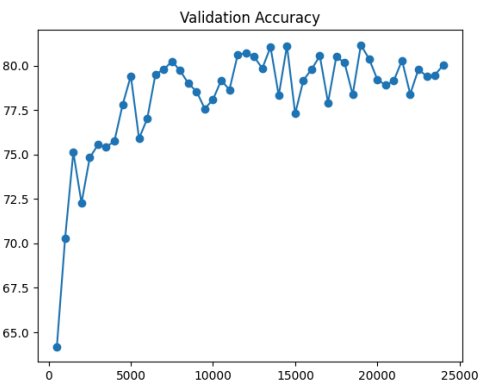
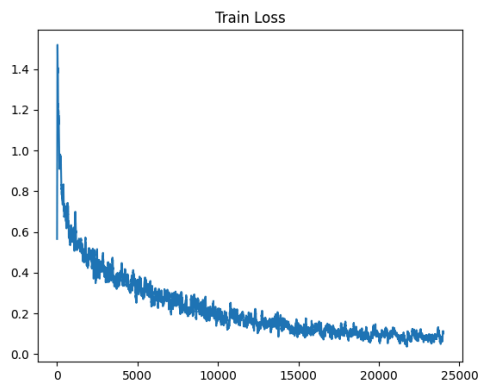
- exp_5



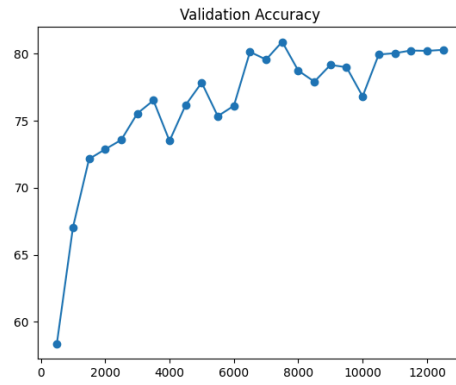
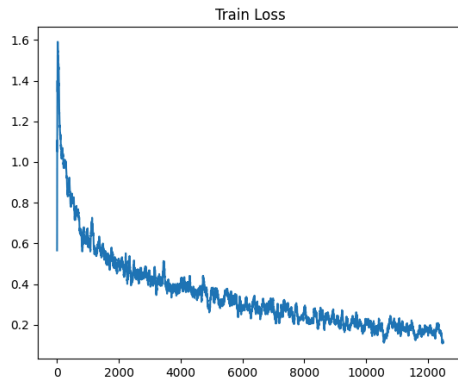
- exp_6



- exp_7



- exp_8



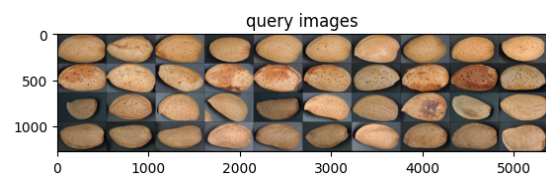
전체 결과 Table

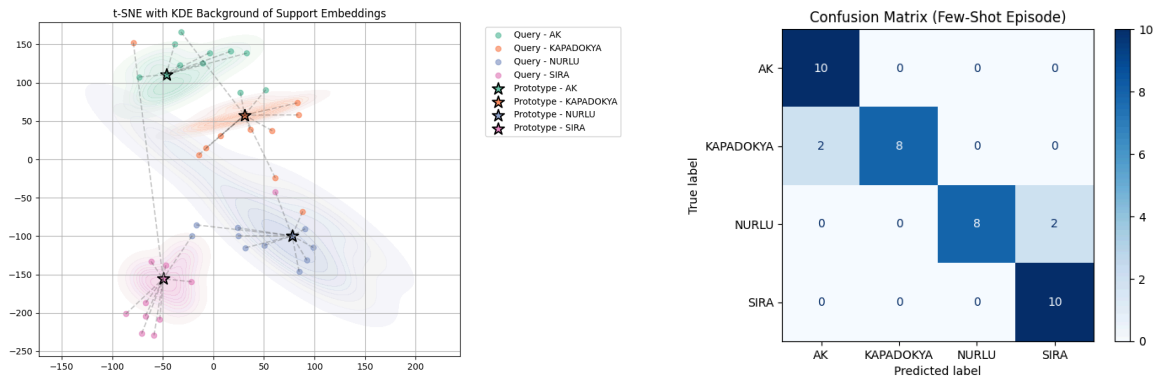
experiment_id	shot	meta_batch_size	best_val_acc	status
exp_1	1		52.47500000000001	Evaluated
exp_2	5		71.375	Evaluated
exp_3	1	1	68.025	Finished
exp_4	1	4	69.8	Finished
exp_5	1	8	69.1	Finished
exp_6	5	1	82.875	Finished
exp_7	5	4	81.175	Finished
exp_8	5	8	80.9	Finished

▼ 중간보고용 결과

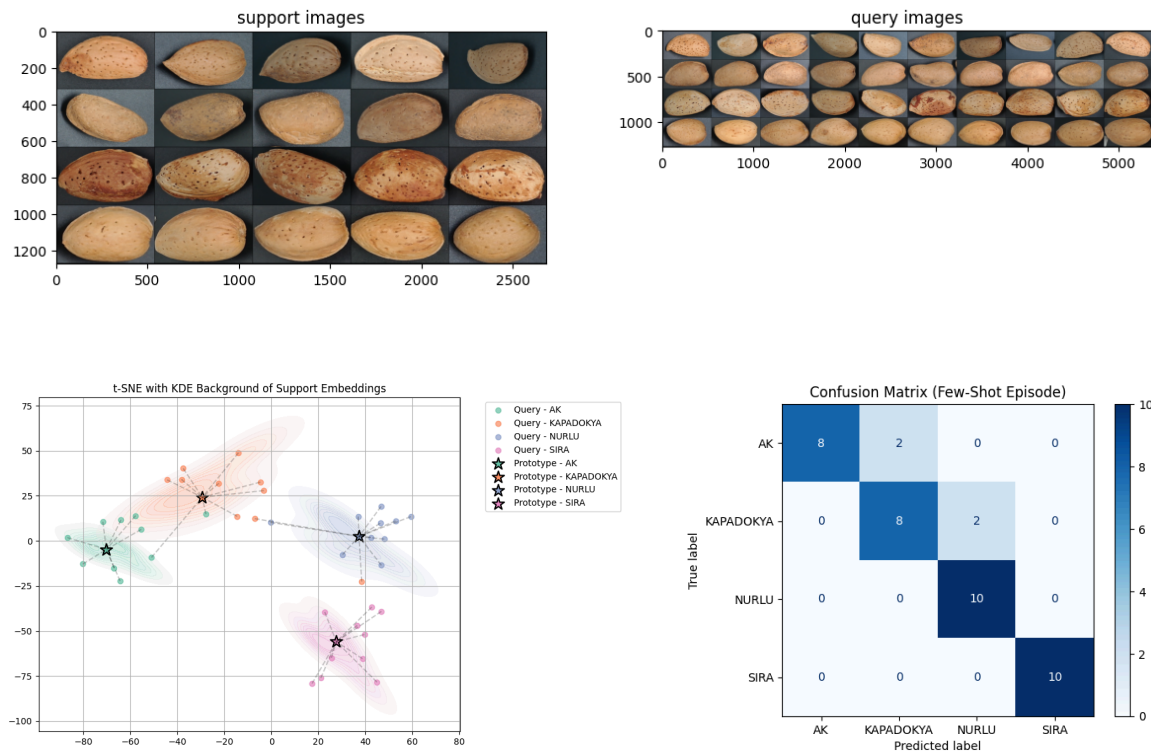
실험 1

에피소드 1.





에피소드 2.



Model tested on 312 tasks. Accuracy: 86.87%

References

1. <https://ar5iv.labs.arxiv.org/html/1703.05175>
2. <https://www.cs.toronto.edu/~kriz/cifar.html>

3. <https://www.kaggle.com/datasets/mahyek/mandarin-varieties/data>

4. <https://link.springer.com/article/10.1007/s00217-024-04562-4>

Acknowledge

본 보고서는 카카오테크 부트캠프에서 개인프로젝트(2주간 주어진 데이터셋 중 원하는 데이터셋을 사용해 원하는 사전 훈련 모델(CNN)을 훈련시키고 연구보고서를 작성할 것.)를 수행하기 위해 작성됨을 밝힌다.

본 보고서를 작성할 때 기반이 된 colab 코드를 공유한다.

중간보고 :

https://colab.research.google.com/drive/1z7bQPU1u6sK4Qmd4-orZ7_MmukGaQ-bO

최종보고 :

<https://colab.research.google.com/drive/1NA33WGcGDJxviRs4SvMOE8AtHDrRJ4g6?usp=sharing>

머신러닝 일반화 흐름도

```
graph LR
    Title["머신러닝 일반화 흐름도"]
    Title --> A
    A["<b>Few-Shot Learning</b><br>소수 예시로 학습"]
    B["<b>Zero-Shot Learning</b><br>예시 없이 일반화"]
    C["<b>Meta-Learning</b><br>Learning to Learn"]
    D["<b>Meta-RL</b><br>Task에 적응하는 RL"]
    E["<b>GPT / LLM</b><br>In-Context Learning<br>+ RLHF"]

    A --> B
    B --> C
    C --> D
    C --> E
    D --> E
```