딥러닝 심화

Homework#2

학과 : 산업인공지능학과
학번 : 2024254022
이름 : 정현일

2025.06.18.

충북대학교 산업인공지능학과

ViT- CIFAR10학습 모델

1. 필요 라이브러리 import

```
import torch
import evaluate
import numpy as np
from itertools import chain
from collections import defaultdict
from torch.utils.data import Subset
from torchvision import datasets
from torchvision import transforms
from transformers import AutoImageProcessor
from transformers import CvtForImageClassification
from transformers import TrainingArguments, Trainer
```

2. 데이터셋 각 클래스별 최대 개수 제한하는 샘플링 함수

```
def subset_sampler(dataset, classes, max_len):
    target_idx = defaultdict(list)
    labels = dataset.targets
    for idx, label in enumerate(labels):
        target_idx[int(label)].append(idx)

    indices = list(
        chain.from_iterable(
            [target_idx[idx][:max_len] for idx in range(len(classes))]
        )
    )
    return Subset(dataset, indices)
```

ViT- CIFAR10학습 모델

3. 이미지 분류를 위한 CvT 모델 초기화 함수

```python
def model_init(classes, class_to_idx):
    model = CvtForImageClassification.from_pretrained(
        pretrained_model_name_or_path="microsoft/cvt-21",
        num_labels=len(classes),
        id2label={idx: label for label, idx in class_to_idx.items()},
        label2id=class_to_idx,
        ignore_mismatched_sizes=True
    )
    return model
```

4. 함수정의

Collect 함수

```python
def collator(data, transform):
    images, labels = zip(*data)
    pixel_values = torch.stack([transform(image) for image in images])
    labels = torch.tensor([label for label in labels])
    return {"pixel_values": pixel_values, "labels": labels}
```

모델 평가 메트릭스 함수

```python
def compute_metrics(eval_pred):
    metric = evaluate.load("f1")
    predictions, labels = eval_pred
    predictions = np.argmax(predictions, axis=1)
    macro_f1 = metric.compute(
        predictions=predictions, references=labels, average="macro"
    )
    return macro_f1
```

ViT- CIFAR10학습 모델

5. 데이터 준비

```python
train_dataset = datasets.CIFAR10(root="./datasets", download=True, train=True)
test_dataset = datasets.CIFAR10(root="./datasets", download=True, train=False)

classes = train_dataset.classes
class_to_idx = train_dataset.class_to_idx

print(f"CIFAR10 클래스: {classes}")
print(f"클래스 수: {len(classes)}")

subset_train_dataset = subset_sampler(
    dataset=train_dataset, classes=train_dataset.classes, max_len=1000
)
subset_test_dataset = subset_sampler(
    dataset=test_dataset, classes=test_dataset.classes, max_len=100
)

print(f"학습 데이터 크기: {len(subset_train_dataset)}")
print(f"테스트 데이터 크기: {len(subset_test_dataset)}")

image_processor = AutoImageProcessor.from_pretrained(
    pretrained_model_name_or_path="microsoft/cvt-21"
)
```

데이터셋 로드

클래스 정보 확인

데이터 부분집합 생성

데이터 크기 확인

사전학습된 cvt모델 로드

클래스 정보 확인 내역

```
CIFAR10 클래스: ['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck']
클래스 수: 10
학습 데이터 크기: 10000
테스트 데이터 크기: 1000
```

ViT- CIFAR10학습 모델

6. CIFAR10용 이미지 전처리 파이프라인 정의함수

```python
transform = transforms.Compose([
    transforms.ToTensor(),
    transforms.Resize(
        size=(
            image_processor.size["shortest_edge"],
            image_processor.size["shortest_edge"]
        )
    ),

    transforms.Normalize(
        mean=image_processor.image_mean,
        std=image_processor.image_std
    )
])
```

7. 모델 Train 설정

```python
args = TrainingArguments(
    output_dir="./models/CvT-CIFAR10",
    save_strategy="epoch",
    evaluation_strategy="epoch",
    learning_rate=1e-5,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    num_train_epochs=3,
    weight_decay=0.001,
    load_best_model_at_end=True,
    metric_for_best_model="f1",
    logging_dir="logs",
    logging_steps=125,
    remove_unused_columns=False,
    seed=7
)

trainer = Trainer(
    model_init=lambda x: model_init(classes, class_to_idx),
    args=args,
    train_dataset=subset_train_dataset,
    eval_dataset=subset_test_dataset,
    data_collator=lambda x: collator(x, transform),
    compute_metrics=compute_metrics,
    tokenizer=image_processor,
)
trainer.train()
```

Train 매개변수 설정

Trainer 설정

학습수행

8. 모델 Train 결과 (epoch 3)

[1875/1875 03:03, Epoch 3/3]

Epoch	Training Loss	Validation Loss	F1
1	0.805200	0.315930	0.904773
2	0.630800	0.207533	0.930983
3	0.580100	0.195211	0.938852

ViT- CIFAR10학습 모델

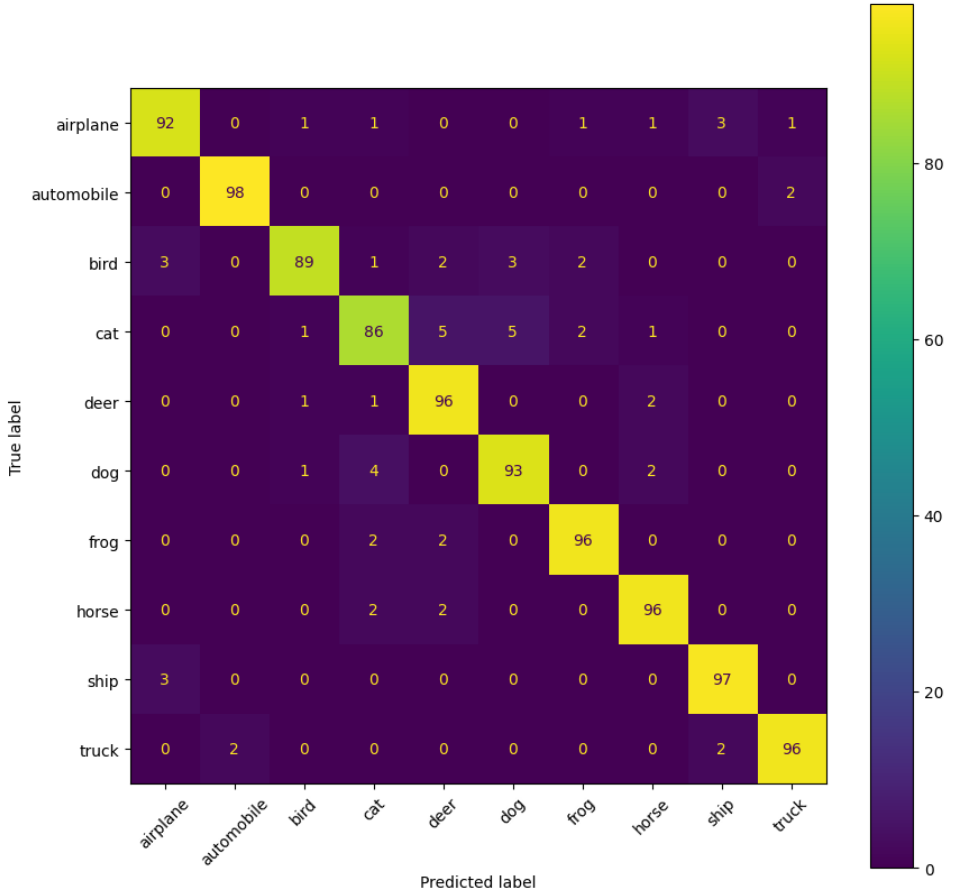
9. 훈련모델 평가 및 confusion matrix 시각화

```python
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay


outputs = trainer.predict(subset_test_dataset)
print(outputs)

y_true = outputs.label_ids
y_pred = outputs.predictions.argmax(1)

labels = list(classes)
matrix = confusion_matrix(y_true, y_pred)
display = ConfusionMatrixDisplay(confusion_matrix=matrix, display_labels=labels)
_, ax = plt.subplots(figsize=(10, 10))
display.plot(xticks_rotation=45, ax=ax)
plt.show()
```

예측 수행

예측 결과 추출

Confusion matrix 시각화



10. 예측결과 시각화 함수

```python
def visualize_predictions(dataset, y_true, y_pred, classes, num_images_per_class=5):
    class_to_images = defaultdict(list)

    # 클래스별로 예시 이미지 수집
    for i, (true, pred) in enumerate(zip(y_true, y_pred)):
        if len(class_to_images[true]) < num_images_per_class:
            class_to_images[true].append((i, true, pred))

    num_classes = len(classes)
    fig, axs = plt.subplots(num_classes, num_images_per_class, figsize=(num_images_per_class * 2, num_classes * 2))
    fig.suptitle("Prediction Visualization(True → Pred)", fontsize=16)

    for class_idx in range(num_classes):
        images_info = class_to_images[class_idx]
        for j in range(num_images_per_class):
            ax = axs[class_idx, j]
            ax.axis('off')
            if j < len(images_info):
                idx, true, pred = images_info[j]
                image, _ = dataset[idx]

                image = transforms.ToTensor()(image)
                image = image.permute(1, 2, 0).numpy()

                ax.imshow(image)
                title = f"{classes[true]} ({classes[pred]})"
                color = "green" if true == pred else "red"
                ax.set_title(title, color=color, fontsize=8)

    plt.tight_layout()
    plt.subplots_adjust(top=0.95)
    plt.show()
```

ViT- CIFAR10학습 모델

11. 예측결과 시각화 : 실제(예측)

```
visualize_predictions(subset_test_dataset, y_true, y_pred, classes)
```

Prediction Visualization(True → Pred)