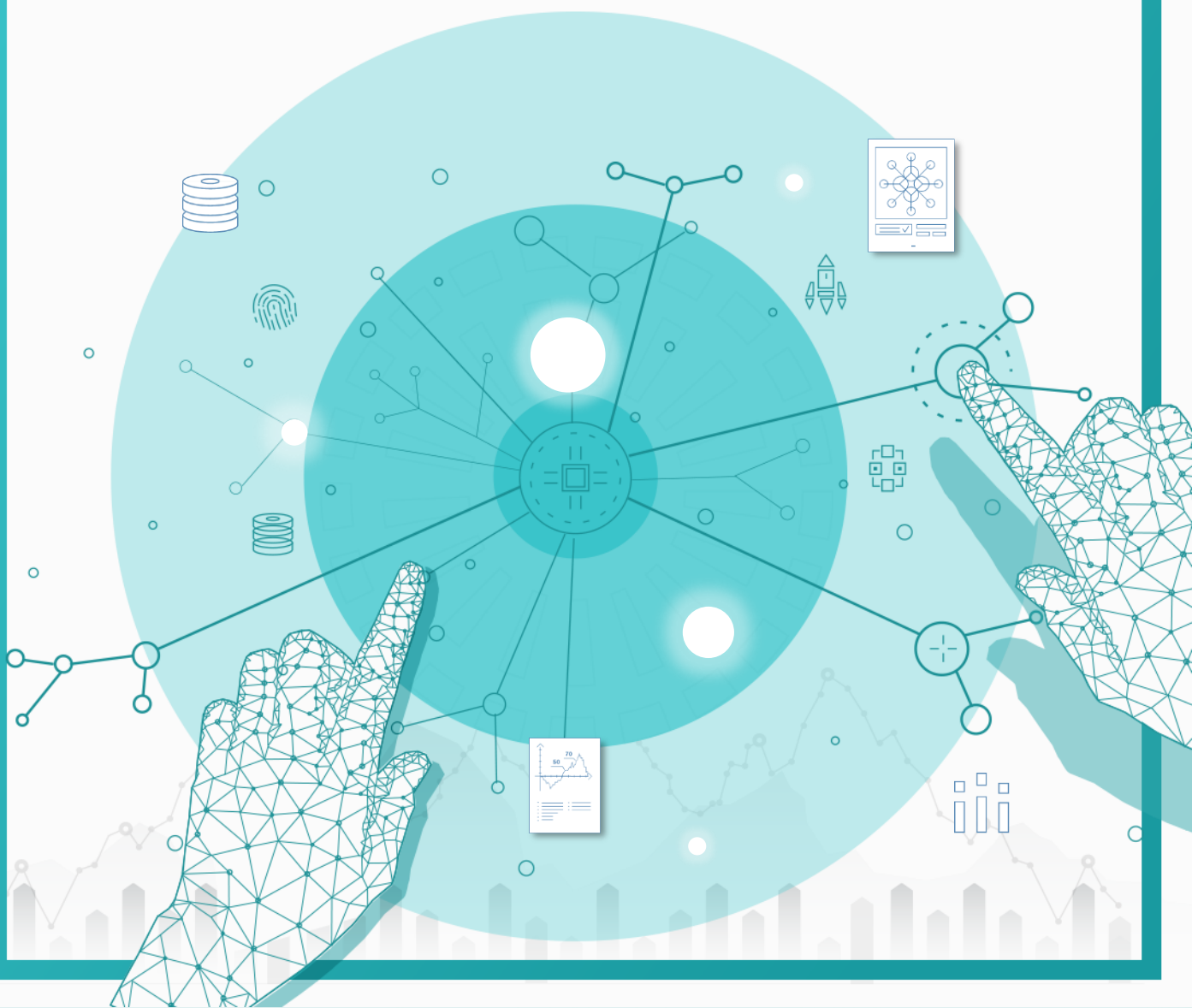




한국기술교육대학교
온라인평생교육원

파이썬을 활용한 인공지능 자연어 처리(실습)

BERT 모델을 활용한
Text Classification 모델 구현



BERT 모델을 활용한 Text Classification 모델 구현

학습 목표

1. BERT 모델을 활용해 Text Binary Classification 모델을 구현할 수 있다.
2. BERT 모델을 활용해 Text Multi Classification 모델을 구현할 수 있다.

학습 내용

1. BERT Binary Classification 모델 구현
2. BERT Multi Classification 모델 구현

1. BERT Binary Classification 모델 구현

1) 학습 및 평가 데이터 세트

(1) 데이터 세트 준비

- 영화평 데이터 세트 업로드 및 Pandas를 이용한 데이터 로드

```
import pandas as pd

dataset = pd.read_table("/content/mnt/MyDrive/ratings_

dataset.columns=['id', 'review', 'sentiment']
```

```
dataset.head()
```

	id	review	sentiment
0	9976970	더빙이 실망스럽네요..	0
1	3819312	오버연기조차 가볍지 않구나	1
2	10265843	너무재밌었다. 그래서보는것을추천한다	0
3	9045019	솔직히 재미는 없다..평점 조정	0
4	6483659	배우의 익살스런 연기가 돋보였던 영화!	1

1. BERT Binary Classification 모델 구현

1) 학습 및 평가 데이터 세트

(1) 데이터 세트 준비

- Scikit-learn을 이용한 Train(80%), Test(20%) 데이터 세트 분리
 - stratify 파라미터를 통한 층화 샘플링(Stratified Sampling)

```
from sklearn.model_selection import train_test_split

train_idx, test_idx, _, _ = train_test_split(dataset.index,
                                             test_size=0.2,
train_set = dataset.iloc[train_idx]
test_set = dataset.iloc[test_idx]
train_set.reset_index(drop=True, inplace=True)
```

- Train 데이터 세트를 이용하여 Train(70%), Validation(30%) 데이터 세트 분리
 - stratify 파라미터를 통한 층화 샘플링(Stratified Sampling)

```
train_idx, valid_idx, _, _ = train_test_split(train_set.index,
                                             test_size=0.3,
valid_set = train_set.iloc[valid_idx]
train_set = train_set.iloc[train_idx]

train_set.shape, valid_set.shape, test_set.shape

((84000, 3), (36000, 3), (30000, 3))
```

1. BERT Binary Classification 모델 구현

1) 학습 및 평가 데이터 세트

(2) 데이터 세트 클래스 정의

`torch.utils.data.Dataset` 상속

`__init__()`, `__len__()`, `__getitem__()` 함수 재정의

`__init__()` Text/Label 설정, Tokenizer, 최대 Token 수 지정

`__len__()` 전체 데이터 세트의 개수 리턴

`__getitem__()` 특정 index에 해당하는 입력 Text ID, attention mask, label 리턴

(3) Pre-trained Tokenizer 다운로드

- Huggingface로부터 Pre-trained Tokenizer 다운로드

```
bert_model_name = 'kykim/bert-kor-base'
```

```
tokenizer = BertTokenizerFast.from_pretrained(bert_model_name)
```

Downloading: 100%  336k/336k [0

Downloading: 100%  80.0/80.0 [00

Downloading: 100%  725/725 [00:

1. BERT Binary Classification 모델 구현

1) 학습 및 평가 데이터 세트

(4) 데이터 세트 클래스 객체 생성

- Train 데이터 세트와 Validation 데이터 세트를 이용하여 객체 생성

```
train_set_dataset = BertDataset(  
    reviews    = train_set.review.tolist(),  
    sentiments = train_set.sentiment.tolist(),  
    tokenizer   = tokenizer,  
)  
  
valid_set_dataset = BertDataset(  
    reviews    = valid_set.review.tolist(),  
    sentiments = valid_set.sentiment.tolist(),  
    tokenizer   = tokenizer,  
)
```


1. BERT Binary Classification 모델 구현

2) Pre-trained 모델 다운로드 및 Fine-tuning 설정

(1) Pre-trained 모델 다운로드

- Huggingface로부터 한국어 학습 모델의 하나인 'kykim/bert-kor-base' Pre-trained 모델 다운로드

```
from transformers import BertForSequenceClassification  
  
model = BertForSequenceClassification.from_pretrained(bert_model_name)
```

Downloading: 100%  454M/454M [00:11<00]

- [참고] 한국어 학습 모델 검색
 - <https://huggingface.co/models>에서 'kor' 검색

1. BERT Binary Classification 모델 구현

2) Pre-trained 모델 다운로드 및 Fine-tuning 설정

(2) Fine-tuning 설정

- `tl_strategy`를 설정하여 `param.requires_grad = False` 범위 설정

```
tl_strategy = 3

if tl_strategy == 1:
    for name, param in model.bert.named_parameters():
        print(name)
        param.requires_grad = False

elif tl_strategy == 2:
    for name, param in model.bert.named_parameters():
        if not name.startswith('pooler'):
            param.requires_grad = False

elif tl_strategy == 3:
    for name, param in model.bert.named_parameters():
        if ( not name.startswith('pooler') ) and "layer.23" not in name :
            param.requires_grad = False
```


1. BERT Binary Classification 모델 구현

3) 모델 학습 및 평가

(1) Training을 위한 하이퍼 파라미터 설정

- 모델 Output 디렉토리, Epoch 수, Batch Size 등
하이퍼 파라미터 설정

```
training_args = TrainingArguments(  
    output_dir                = model_dir,  
    num_train_epochs          = 1,  
    per_device_train_batch_size = 128,  
    per_device_eval_batch_size = 64,  
    warmup_steps              = 500,  
    weight_decay               = 0.01,  
    save_strategy              = "epoch",  
    evaluation_strategy         = "steps"  
)
```

1. BERT Binary Classification 모델 구현

3) 모델 학습 및 평가

(2) Evaluation을 위한 Test 데이터 세트 준비 및 파라미터 설정

- Test 데이터 세트를 이용하여 객체 생성

```
test_set_dataset = BertDataset(  
    reviews      = test_set.review.tolist(),  
    sentiments    = test_set.sentiment.tolist(),  
    tokenizer     = tokenizer,  
)
```

- Predict를 위한 do_predict 등 파라미터 설정

```
training_args = TrainingArguments(  
    output_dir = "./model",  
    do_predict = True  
)
```

1. BERT Binary Classification 모델 구현

3) 모델 학습 및 평가

(3) Evaluation을 위한 predict 수행

- `trainer.predict()` 함수를 이용하여 Predict 수행

```
trainer = Trainer(  
    model          = model,  
    args           = training_args,  
    compute_metrics = compute_metrics,  
)  
  
trainer.predict(test_set_dataset)
```

2. BERT Multi Classification 모델 구현

1) 학습 및 평가 데이터 세트

(1) 데이터 세트 준비

- News 데이터 세트 업로드 및 Pandas를 이용한 데이터 로드

```
import pandas as pd
```

```
dataset = pd.read_csv("/content/mnt/MyDrive/news.csv")
```

```
dataset[dataset.category==7].head()
```

	category	num	news
1054	7	7000	"중주국 자존심 지키다" 25일 도쿄올림픽 태권도 남자 68kg급에 출전하는 이대훈(...
1055	7	7001	23일 도쿄올림픽이 개막식을 열고 본격적인 스포츠 축제의 시작을 알리는 가운데, 재...
1056	7	7002	호텔 전체 빌려 급식지원센터로 ..생선 제외, 육류 뉴질랜드·호주산 ..채소·과일도...
1057	7	7003	2020 도쿄올림픽 남자 축구 대표팀 선수인 이동경이 뉴질랜드와의 첫 경기 후 상대...
1058	7	7004	코로나19로 1년 미뤄진 2020 도쿄올림픽이 23일 결국 막을 올렸다... 해를 ...

※출처: 공공데이터포털, 한국언론진흥재단_뉴스빅데이터_메타데이터_올림픽, 2021,
<https://www.data.go.kr>

2. BERT Multi Classification 모델 구현

1) 학습 및 평가 데이터 세트

(1) 데이터 세트 준비

- Label(Category)별 데이터 수

```
dataset.category.value_counts()
```

```
7    200
6    200
5    200
4    200
3    200
2    200
1    200
0    200
```

```
Name: category, dtype: int64
```

```
train_set.shape, valid_set.shape, test_set.shape
```

```
((1024, 3), (256, 3), (320, 3))
```

2. BERT Multi Classification 모델 구현

1) 학습 및 평가 데이터 세트

(2) 데이터 세트 클래스 정의

- 데이터 세트의 Text(News)와 Label(Category)에 따라 클래스 필드명 설정

```
class BertDataset(torch.utils.data.Dataset):  
  
    def __init__(self, news, category, tokenizer):  
        self.news      = news  
        self.category  = category  
        self.tokenizer = tokenizer  
        self.max_len   = tokenizer.model_max_length
```

2. BERT Multi Classification 모델 구현


2) Pre-trained 모델 다운로드 및 Fine-tuning 설정

(1) Pre-trained 모델 다운로드

- Huggingface로부터 한국어 학습 모델의 하나인 'kykim/bert-kor-base' Pre-trained 모델 다운로드
 - **num_labels = 8**은 분류할 category의 개수(class 수)

```
from transformers import BertForSequenceClassification

num_labels = 8
model = BertForSequenceClassification.from_pretrained(bert_model_name,
```

Downloading: 100%  1.25G/1.25G [00:35<00]

(2) Training을 위한 하이퍼 파라미터 설정

- 모델 Output 디렉토리, Epoch 수, Batch Size 등 하이퍼 파라미터 설정

```
training_args = TrainingArguments(
    output_dir                = model_dir,
    num_train_epochs          = 1,
    per_device_train_batch_size = 128,
    per_device_eval_batch_size = 64,
    warmup_steps              = 500,
    weight_decay               = 0.01,
    save_strategy              = "epoch",
    evaluation_strategy         = "steps",
)
```