

HuggingFace 실습

학습 목표 및 학습 내용

- Hugging Face를 사용하여 감정 분석을 수행해 봅니다.

학습 내용

- Hugging Face 라이브러리 설치
- Hugging Face API 토큰 인증
- 모델 로드 및 감정 분석 수행
- 다른 언어(특히 한국어)에 대한 감정 분석

- 라이브러리 설치

```
In [ ]: !pip install transformers torch
!pip install huggingface_hub
```

Hugging Face 토큰 받기 및 인증

- 01 <https://huggingface.co/> 에 접속하여 Hugging Face 계정 생성
- 02 계정이 생성되면 우측 상단의 프로필 아이콘을 클릭합니다.
- 03 "Settings" 메뉴에서 "API tokens" 탭으로 이동합니다.
- "Access tokens" 버튼을 클릭하여 API 토큰을 생성합니다.
- "Create token" 버튼을 클릭하여 토큰을 생성합니다.
- 토큰은 한번만 복사가 가능하기에 임의의 장소에 복사해 둡니다.

```
In [ ]: from huggingface_hub import login
from transformers import pipeline

# Hugging Face 토큰 입력 받기
from getpass import getpass

huggingface_token = getpass("Hugging Face 토큰을 입력하세요: ")

# Hugging Face 로그인
login(token=huggingface_token)
```

모델 로드 및 감정 분석 수행하기

- DistilBERT는 BERT 모델을 기반으로 만들어진 경량화된 버전
- 감정 분석 모델을 사용하여 주어진 문장 "I love using Hugging Face Transformers!"에 대한 감정 분석을 수행

```
In [ ]: # 모델 로드
model_name = "distilbert-base-uncased"
classifier = pipeline("sentiment-analysis", model=model_name)
```

```
# 모델 테스트 - 감정분석
print(classifier("I love using Hugging Face Transformers!"))
```

다른 모델 사용으로 정확도를 높여보기

DeBERTa를 사용한 감정 분석

- DeBERTa (Decoding-enhanced BERT with disentangled attention)
- 모델 이름: microsoft/deberta-base
 - DeBERTa는 BERT 및 RoBERTa의 개선된 버전입니다.

```
In [ ]: from transformers import pipeline
        from getpass import getpass
        from huggingface_hub import login

        # Hugging Face 토큰 입력 받기
        huggingface_token = getpass("Hugging Face 토큰을 입력하세요: ")

        # Hugging Face 로그인
        login(token=huggingface_token)
```

```
In [ ]: # 모델 로드
        model_name = "microsoft/deberta-base"
        classifier = pipeline("sentiment-analysis", model=model_name)

        # 모델 테스트
        print(classifier("I love using Hugging Face Transformers!"))
```

감정 분석을 위한 다른 모델

- bert-base-uncased : BERT 모델의 기본 버전
- roberta-base : BERT의 변형으로 성능이 향상된 버전
- distilbert-base-uncased-finetuned-sst-2-english : 감정 분석을 위해 SST-2 데이터셋으로 미세 조정된 DistilBERT 모델
 - SST-2 데이터셋(SST-2, Stanford Sentiment Treebank-2)은 자연어 처리(NLP) 분야에서 감정 분석(Sentiment Analysis)을 수행하는 데 자주 사용되는 데이터셋. 이 데이터셋은 영화 리뷰 텍스트와 그에 따른 감정 레이블을 포함하고 있으며, Stanford Sentiment Treebank의 일부이다.

```
In [ ]: # 모델 로드
        model_name = "distilbert-base-uncased-finetuned-sst-2-english"
        classifier = pipeline("sentiment-analysis", model=model_name)

        # 모델 테스트 - 감정분석
        print(classifier("I love using Hugging Face Transformers!"))
```

```
In [ ]: # 모델 로드
        model_name = "distilbert-base-uncased-finetuned-sst-2-english"
        classifier = pipeline("sentiment-analysis", model=model_name)
```

```
# 모델 테스트 - 감정분석
print(classifier("나는 허깅페이스를 좋아합니다."))
```

```
In [ ]: # 모델 테스트 - 감정분석
print(classifier("오늘은 기분이 우울하다."))
print(classifier("오늘은 기분이 나쁘다."))
print(classifier("비가 오기에 기분이 꿀꿀하다"))
print(classifier("오늘은 경기에 이겼다."))
```

```
In [ ]: # Model Test - Sentiment Analysis
print(classifier("I'm feeling depressed today."))
print(classifier("I'm in a bad mood today."))
print(classifier("It's raining, so I'm feeling gloomy."))
print(classifier("I won the game today."))
```

이 모델들은 한글은 조금 성능이 떨어진다.

한글 성능이 좋은 모델을 살펴보기

- WhitePeak/bert-base-cased-Korean-sentiment : BERT 기반의 한국어 감정 분석 모델
- nlp04/korean_sentiment_analysis_klectra : ELECTRA 기반의 한국어 감정 분석 모델
- dudcjs2779/sentiment-analysis-with-klue-bert-base : KLUE BERT 기반의 한국어 감정 분석 모델

```
In [ ]: # 모델 로드
model_name = "WhitePeak/bert-base-cased-Korean-sentiment"
classifier = pipeline("sentiment-analysis", model=model_name)

# 모델 테스트 - 감정분석
print(classifier("나는 허깅페이스를 좋아합니다."))
```

```
In [ ]: # 모델 테스트 - 감정분석
print(classifier("오늘은 기분이 우울하다."))
print(classifier("오늘은 기분이 나쁘다."))
print(classifier("비가 오기에 기분이 꿀꿀하다"))
print(classifier("오늘은 경기에 이겼다."))
```

결론

이번 노트북에서는 Hugging Face의 다양한 사전 학습된 모델을 활용하여 감정 분석을 수행하는 방법을 학습했습니다. 주요 학습 성과는 다음과 같습니다:

1. **Hugging Face의 도구 사용법 이해**: Hugging Face의 `Transformers` 라이브러리와 `pipeline` 을 사용하여 간단하게 감정 분석을 수행하는 방법을 익혔습니다.
2. **다양한 모델 비교**: 여러 NLP 모델(`DistilBERT`, `DeBERTa`, `BERT`, `RoBERTa` 등) 을 사용하여 감정 분석을 수행하고, 모델별 성능을 비교해 보았습니다.
3. **언어별 모델 성능 차이 이해**: 영어와 한국어 텍스트에 대해 감정 분석 모델의 성능을 비교하고, 각 언어에 적합한 모델을 선택하는 것이 중요함을 배웠습니다.

추가적으로 탐구해볼 만한 주제

- **다양한 NLP 작업 적용:** 감정 분석 외에도 번역, 요약, 텍스트 생성 등 Hugging Face의 다양한 모델을 활용한 다른 NLP 작업에 대해 탐구.
- **모델 튜닝 및 커스터마이징:** 사전 학습된 모델을 특정 도메인에 맞게 튜닝하고 커스터마이징하는 방법을 연구.
- **다중 언어 지원:** 한국어를 포함한 다양한 언어에 대해 최적의 모델을 탐색하고 적용.

모두 수고하셨습니다. 열심히 해 주셔서 감사합니다.

In []: