

Text Classification with KoBERT

Jiyoung Kim, Yoonjin Oh

Artificial Intelligence in KU (AIKU)

Department of Computer Science and Engineering, Korea University



역할

- 김지영
 - 데이터 전처리, 모델 선정, (Test Inference) 코드 작성
- 오윤진
 - (Train, Val) 모델 파이프라인 코드 작성

데이터셋 분석

song_name	adults	artist	album_id	date	genre	lyrics
긴 잠	0.0	INO	'1869'	2001.03.16	0	잠을 깨는 것이 싫었어wn눈 뜨면 또 하루wn니 곁을 살테니wn오직 내꿈 속엔 넌 ...
Alcatraz	0.0	INO	'1869'	2001.03.16	1	이젠 널 가둬놓겠어wn나의 품에wn조금은 낯설겠지만wn편해질꺼야wn두려운 내 맘 때...
해요	0.0	INO	'1869'	2001.03.16	0	그녀와 나는요 그땐 참 어렸어요wn많이 사랑했고 때론 많이 다뤘었죠wn지금 생각하면...
투비(鬪悲)	0.0	INO	'1869'	2001.03.16	0	예전처럼 다시 처음으로wn서로 몰랐던 때로 돌아가wn쉽진 않지만wn부탁이야 잊어줘w...

- 총 8개의 column 으로 구성 (위 이미지는 id column 가 제거된 상태)
- 한 곡에 대한 장르(genre) 또는 가수(artist)가 2개 이상인 샘플(인스턴스)도 존재.
 - 장르(genre) column dtype 이 str.
 - 한국어 데이터셋 (모델 선정에 제약 多)

구현 방향성

- **베이스라인 모델 선정:** Text Classification Task 에 강한 모델 탐색
 - ➡ TRY1) RoBERTa, XLNet
 - ➡ TRY2) KoBERT
- **데이터셋 전처리:** 오버샘플링
 - ➡ 한 샘플에 두 개 이상의 값이 (,로 구분되어) 들어있는 경우, 각 값이 분리되어 개별적인 인스턴스가 되게 처리
- **파인튜닝 방향성:** 기본 모델에 대해서, 본 데이터에 단순 파인튜닝하는 '구현' 이 우선, 이후 '테크닉'

1. 베이스라인 모델 선정

목표: Text Classification Task 에 강한 모델을 탐색하자.

TRY1) RoBERTa, XLNet → Korean 으로 사전학습된 해당 모델을 찾는 데 어려움

TRY2-1) Llama2(LLM) → HF 상에서 Korean 으로 사전학습된 Llama2 모델로 시도
데이터타입과 관련된 오류 이슈로 시간을 더 투자하기 어려워 보류
(훈련시킬 프롬프트까지 작성...했는데...)

TRY2-2) KoBERT → BERT 모델을 Korean 으로 사전학습시킨 모델

최종 선정

1. 베이스라인 모델 선정

목표: Text Classification Task 에 강한 모델을 탐색하자.

TRY1) RoBERTa, XLNet → Korean 으로 사전학습된 해당 모델을 찾는 데 어려움

TRY2-1) Llama2(LLM) → HF 상에서 Korean 으로 사전학습된 Llama2 모델로 시도
데이터타입과 관련된 오류 이슈로 시간을 더 투자하기 어려워 보류
(훈련시킬 프롬프트까지 작성...했는데...)

TRY2-2) KoBERT → BERT 모델을 Korean 으로 사전학습시킨 모델

최종 선정

2. 파인튜닝(Fine-tuning)

목표: 처음부터 끝까지 완전히 돌아가게, 모델을 구현해보자.

- 주요 하이퍼파라미터

```
# Config
epochs = 10
batch_size 32
warmup_ratio = 0.1
lr = 2e-5
grad_clip = 1.0
train_log_interval = 500
save_interval = 1000

# Optimizer
optimizer = torch.optim.AdamW(model.parameters(), lr=lr)
```

3. 성능 평가

10	JEEjeeyOUNG		0.54102	1	2h
----	-------------	---	---------	---	----

- HuggingFace accuracy 라이브러리를 활용하여 Accuracy 산출

4. 추가 아이디어

시간관계상,

알고리즘만 작성하고 간단하게 돌려보았습니다.

아이디어 1

- Background 가설:
곡의 artist 는 특히 genre 와 연관성이 높을 것이다.

➡ 룰베이스 기반 가중치 조정

아이디어 2

- Background 가설:
분류문제를 'QA문제'로 환원하자.
(데이터셋의 모든 feature 를 활용하기 위한 목적)

➡ 직접 Instruction / Context / Response 형식 (QA 데이터셋) 형식에 맞게 전처리 후, QA 언어모델에 훈련

```
for i, batch_data in enumerate(tqdm(val_dataloader)):
    with torch.no_grad():
        lyrics = batch_data['lyric']
        # labels = batch_data['label']
        attention_masks = batch_data['attention_mask']
        # gt_labels = batch_data['gt_label']

        lyrics = lyrics.to(device)
        # labels = labels.to(device)
        attention_masks = attention_masks.to(device)
        # encoded_lyric = labels.to(device) # 수정

        outputs = model(
            lyrics, token_type_ids=None, attention_mask=attention_masks
        )
        for j, _ in enumerate(batch_data):
            artist_idx = i*32 + j
            cur_artist = org_dataset[artist_idx]
            if cur_artist in overlapped_artists:
                genre_list = art_dict[cur_artist]

        logits = outputs.logits

        for i, logit in enumerate(logits):
            if i in genre_list: logits[i] = logit*(1.5)
            # print(logits)
            # Predict Genre (숫자값)
            predicted_labels = torch.argmax(logits, dim=1)
            predictions.append(predicted_labels)
```

회고 겸 소감

- **지영**: NLP 전반적인 아키텍처와 모듈 사용에 파악이 미흡했고, 생각보다 데이터 전처리에서 많은 시간이 소요되었다.
- **윤진**: LLM 에 너무 집착해서 시간을 약간 허비한 감이 없지 않아 있어 아쉽다.
DatasetClass 및 DataLoader 사용법에 대해 정확히 익힐 수 있는 시간이었고, Pandas 는 오랜만에 썼었는데 다시 열심히 공부해야겠다고 생각했다.

전성후 군에게 큰 감사를 전합니다

감사합니다