ELECTRA 기반 **영화평 감성분석**

1st Implementation

<T6>
201611251 공민정
201611276 이규은
201611309 최지현
201612368 이지우

담당교수님 : 김학수 교수님

Contents

프로젝트 소개

- 목적
- Data Set
- Model / System

프로젝트 소개 - 팀명

Review process using

Artificial intelligence tech. to

Classify emotion in

Creative ways for

Opinion mining

Operated by

Neural network

RACCOON

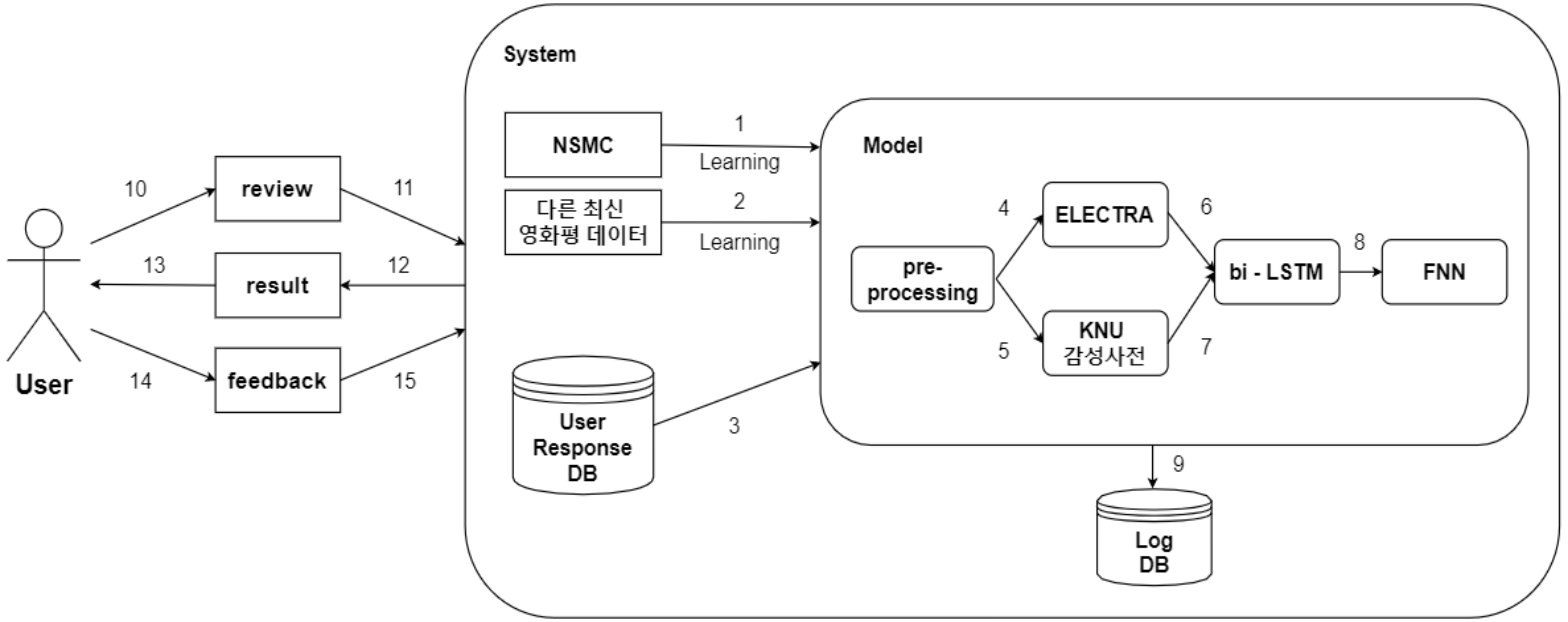
인공지능 기술을 이용해
영화평 긍/부정 판단에 대한 분류 Model에 대해 연구하는
저희의 프로젝트를 나타내는 이름입니다.

프로젝트 소개

영화평 감성분석 시스템

학습된 데이터를 기반으로 하는 영화평 감성분석을 통해 영화평의 긍/부정을 분류하는 모델 연구

연구하는 모델을 통해 사용자에게 결과값을 바로 제공하고, 모델에 대한 피드백을 받을 수 있는 시스템 개발



프로젝트 소개 - 목적

MODEL

Model 정확도 85% 이상

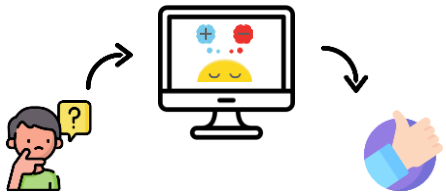
Layer를 쌓은 후의 정확도가
ELECTRA 단독 정확도인 85% 이상

한국어 사전을 사용한 한국형 Model 연구

자연어처리 분야에서 한국어 연구는
타 국가에 반해 잘 이루어지지 않음.
새로운 언어학습 Model인 ELECTRA와
한국어의 긍/부정 정도를 계산하는 감성사전 Model을 사용해
정확도가 높은 한국어 자연어처리 시스템 구축

프로젝트 소개 - 목적

SYSTEM



User에게 영화평에 대한 판단 제공 / 피드백 요청

User를 통해 1개의 영화평에 대한
Model의 결과와 피드백을 받아 DB에 저장

DB를 통한 관리로 Model 정확도 향상

저장된 DB를 통해 부분적/주기적으로
Model의 정확도 향상을 위한 Data 제공

프로젝트 소개 – Data Set

Pre-train하는 NSMC(Naver Sentiment Movie Corpus) Data의 Trend 반영

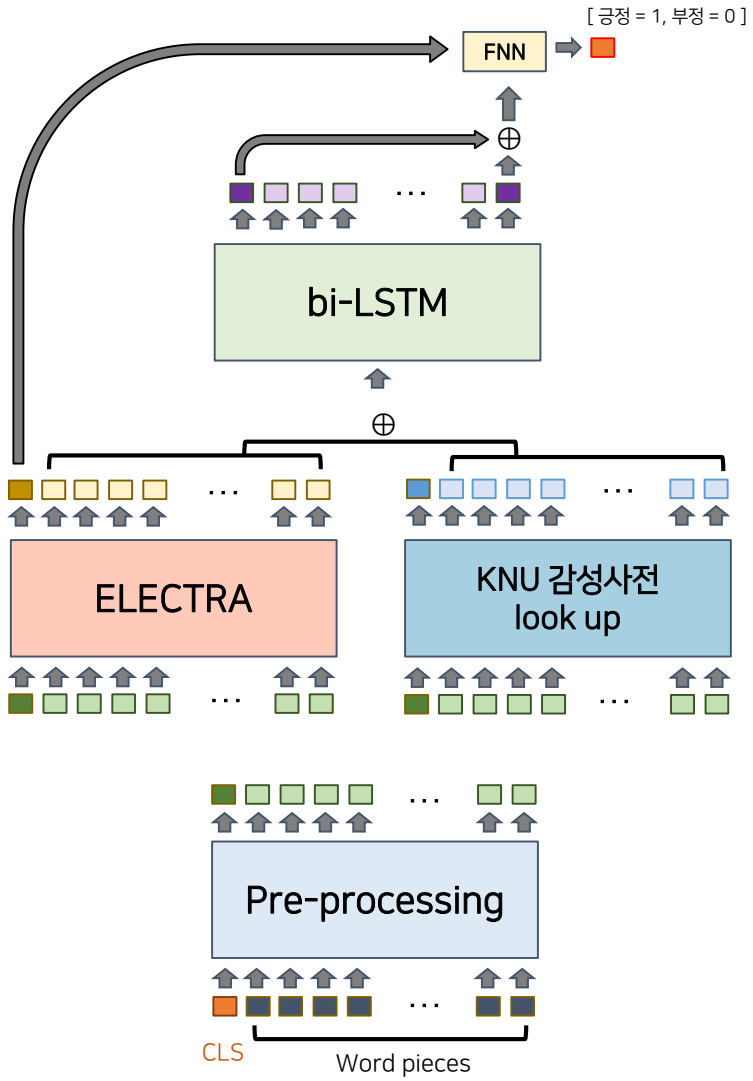
사용 가능한 NSMC Data[1]는 2016년 이전의 영화평으로

새롭게 생긴 신조어, 사회 경향 등의 영향이 있을 것으로 판단.

=> 2017년 이후의 1000여개 영화에 대해 Naver 영화평 Data[2] 수집

1) https://github.com/e9t/nsmc
2) https://github.com/TeamRaccoon/Crawling

프로젝트 소개 - Model



FNN

최종적으로 긍정(1)/부정(0) 산출

bi-LSTM

1차 Pre-train된 2개의 vector을 mapping해 2차 Pre-train

ELECTRA

2020.3 발표 – 모든 Input에 대한 token학습으로 기존 Pre-train Model보다 효율성/정확도가 높음.

1차 Pre-train

KNU 감성사전

2018.11 – 군산대학교에서 연구. 단어에 대한 긍/부정어에 대한 극성출력. => 판단의 정확성을 위해 사용

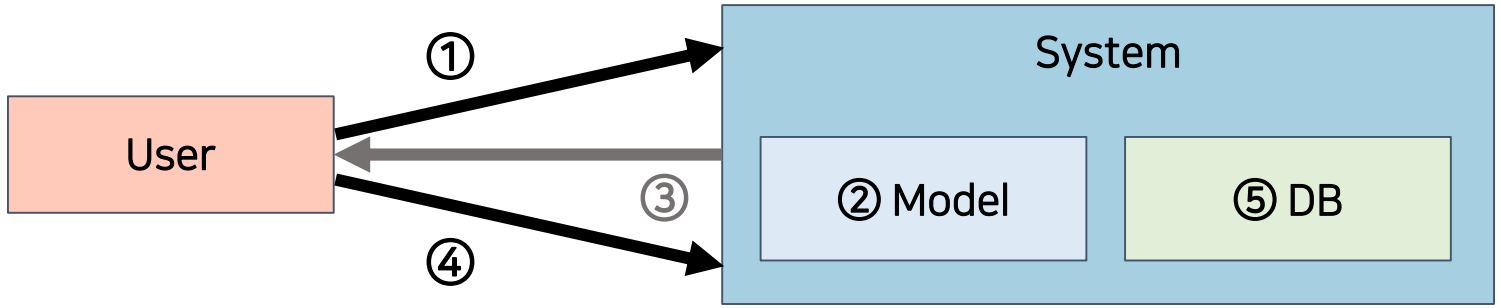
1차 Pre-train

Pre-processing

중복 데이터 제거, 불용어 제거, Null값 제거

프로젝트 소개 - System

A. 영화평 입력



① User -> System

1개의 영화평 입력

② System (Model)

학습된 Model을 통해 입력된 영화평에 대한 결과 산출

③ System -> User

입력된 영화평에 대한 긍정(Positive) / 부정(Negative) 결과 노출

④ User -> System

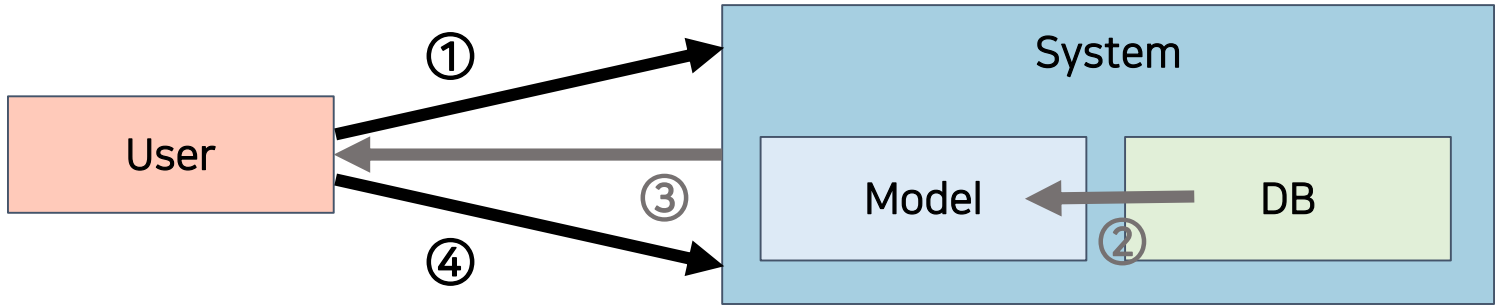
결과에 대한 피드백 (의도와 일치/ 불일치)

⑤ System (DB)

피드백 결과에 따른 영화평, 긍/부정 정보를 DB에 저장

프로젝트 소개 - System

B. Model Update



① User -> System

Update할 Model Version과

DB에 저장된 영화평 데이터(Start No./End No.) 선택

② System (DB -> Model)

DB에 저장된 데이터를 Train-set으로 사용하여 Model Training

③ System -> User

Update한 결과 노출

④ User -> System

System에서 사용할 Model Version 선택

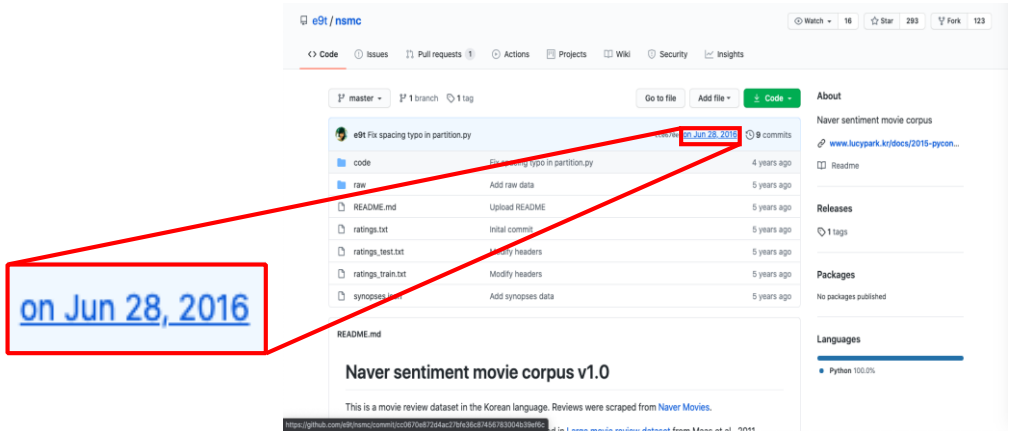
프로젝트 진행상황

- Golden Data
- Model / System
- Prototype

프로젝트 진행상황 - Golden Data

새로운 Data 수집

사용가능한 NSMC Data는 2016년까지의 Data



2017년 이후의 영화평을 Naver 영화에서 크롤링

```
def get_data(url, year):
    try:
        resp3 = requests.get(url)
        html3 = BeautifulSoup(resp3.text, "lxml")

        trs = html3.select("table.list_netizen > tbody >tr")

        list_pos = []
        list_neg = []
        list_neu = []

        for tr in trs:
            review_id = tr.find('td', {'class': 'ac num'}).getText()
            score = tr.find("div",{"class":"list_netizen_score"}).find("em").text
            star = int(score)

            review = tr.find('td', {'class': 'title'})

            review.find("a").extract()
            review.find("a").extract()
            review.find("div").extract()
            review.find("br").extract()

            content = review.text.strip()
```

프로젝트 진행상황 - Golden Data

Naver sentiment movie corpus v1.0

Characteristics

- All reviews are shorter than 140 characters
- Each sentiment class is sampled equally (i.e., random guess yields 50% accuracy)
  - 100K negative reviews (originally reviews of ratings 1-4)
  - 100K positive reviews (originally reviews of ratings 9-10)
  - Neutral reviews (originally reviews of ratings 5-8) are excluded

기존 Data 기준 => 평점 5~8점인 데이터 제외

사람마다 평점을 매기는 기준이 달라
5~8점이 완전히 중립적인 의견이라고 볼 수 없음.
(부정적인 영화평이더라도 7점을 줄 수 있고,
긍정적인 영화평이더라도 5점을 줄 수 있기 때문)

=> 5~8점인 Data를 직접 라벨링할 필요성 느낌
(라벨링 : 영화의 긍/부정 값이 있는 Train-set으로 사용할 Data)

NSMC의 기준을 참고하여
평점 1~4는 부정(0), 평점 8~10은 긍정(1)로 판단

```
def star_score(star):
    if star>=8:
        return 1
    elif star<=4:
        return 0
    else:
        return
```

positive, negative, neutral(평점 5~7) 분류하여 저장
(비율을 알기 위해)

```
if star_score(star) == 1:
    list_pos.append([review_id, content, 1])
elif star_score(star) == 0:
    list_neg.append([review_id, content, 0])
else:
    list_neu.append([review_id, content, None])
```

```
f1 = open('review_'+year+'_pos.csv', 'at' , newline='')
```

프로젝트 진행상황 - Golden Data

기존 NSMC에서 제외되었던 평점 5~7에 대해 직접 Data 라벨링

- 0: 부정, 1: 긍정
- 긍/부정을 판단할 수 없거나 영화와 관계없는 영화평(정치, 의미 없는 단어의 반복) 등은 삭제

13411821	생각보다 재미있었습니다신하균은 당연했고 디오도 연기잘해서 놀랐어요	1
13411707	정말 재미없습니다. 결말도 어이없고 흐지부지 끝나고 비추입니다.	0
13411320		
13411317	재밋게봤어용 ㅎㅎ시체가웃겼음...	1
13410774	평점보고 봤는데 진짜 똥싸다가 덜싼기분저예산 영화도 아닌데 연출면에서 너무 아쉬움배우들 연기는 좋았지만 결국기억남는건 조선족 알바생 대사 중국사람입니다 이것뿐	0
13410415	재미없는거같아요 . 그냥 그래요 내영이해 안되요	0

부정적 → 0

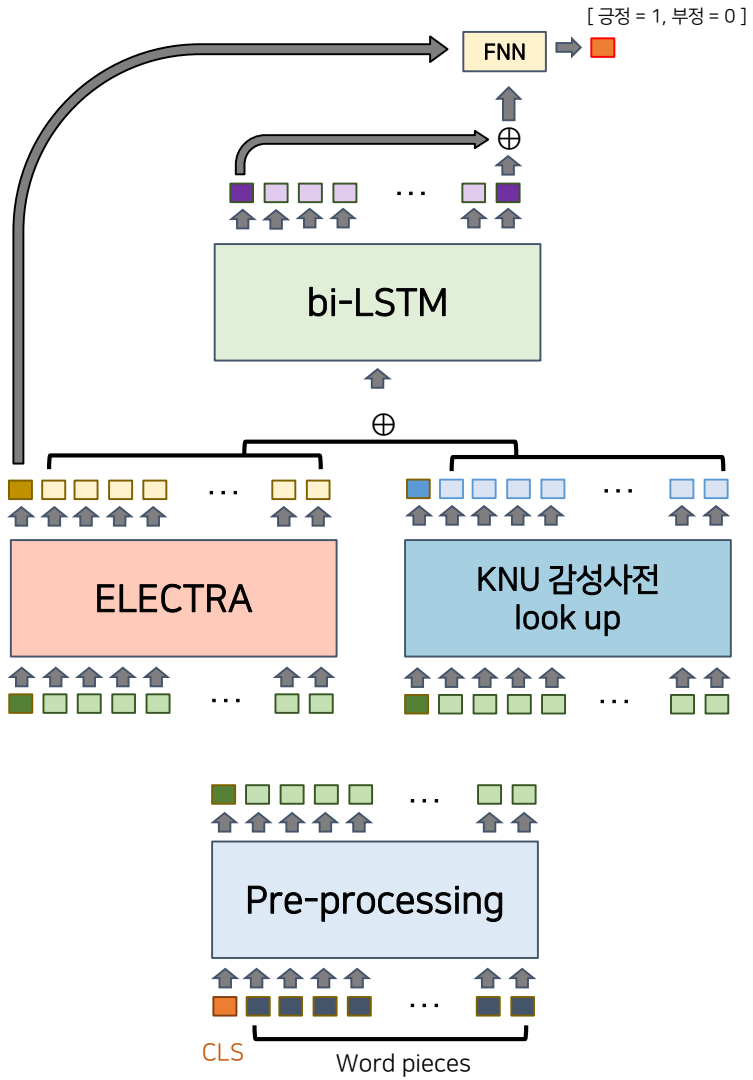
긍/부정 판단 불가능 → 삭제

긍정적 → 1

(새로 수집한 데이터의 유효성은 17-18 슬라이드의 Model 진행상황에서 설명)

프로젝트 진행상황 - Model

Model Layer별 분석



FNN

bi-LSTM

ELECTRA => Token이 가장 정확하고 자세하게 잘려 내부 수정사항 없음.

KNU 감성사전

ELECTRA와 KNU 감성사전의 Token이 다르게 형성되어 Vector mapping 작업이 더 정확하게 이루어져야 함.
=> % mapping / 다르게 나뉘는 Token에 대한 분석 진행 중

원형이 아닌 어근을 찾지 못하는 문제 발생
=> 원형이 아닌 Token 처리에 대한 분석 진행 중

Pre-processing => 중복 데이터, 불용어, Null값 제거
+ 상위 Layer에서의 문제점 보완 가능한 추가 분석 진행 중

프로젝트 진행상황 - Model

Pre-processing + ELECTRA Model

> 기존 NSMC로 학습한 모델

```
convert_data2dataset: 100%|██████████| 50000/50000 [00:07<00:00, 6655.69it/s]
do_train(epoch_1): 100%|██████████| 1172/1172 [04:04<00:00,  4.80it/s]
do_evaluate:   0%|          | 2/500 [00:00<00:25, 19.69it/s]train_accuracy : 0.8822     average_loss : 0.2835

do_evaluate: 100%|██████████| 500/500 [00:27<00:00, 18.22it/s]
test_accuracy : 0.8788

do_train(epoch_2): 100%|██████████| 1172/1172 [04:02<00:00,  4.84it/s]
do_evaluate:   0%|          | 2/500 [00:00<00:25, 19.77it/s]train_accuracy : 0.9006     average_loss : 0.2449

do_evaluate: 100%|██████████| 500/500 [00:27<00:00, 18.23it/s]
test_accuracy : 0.8824
```

정확도 88.24%

새롭게 수집한 Data가

더 높은 Model 정확도를 보임

> Golden Data로 학습한 모델

```
convert_data2dataset: 100%|██████████| 94393/94393 [00:13<00:00, 6763.67it/s]
do_train(epoch_1): 100%|██████████| 4704/4704 [16:22<00:00,  4.79it/s]
do_evaluate:   0%|          | 2/944 [00:00<00:48, 19.32it/s]train_accuracy : 0.9155     average_loss : 0.2084

do_evaluate: 100%|██████████| 944/944 [00:52<00:00, 18.14it/s]
test_accuracy : 0.9028

do_train(epoch_2): 100%|██████████| 4704/4704 [16:16<00:00,  4.82it/s]
do_evaluate:   0%|          | 2/944 [00:00<00:47, 19.70it/s]train_accuracy : 0.952      average_loss : 0.1286

do_evaluate: 100%|██████████| 944/944 [00:51<00:00, 18.21it/s]
test_accuracy : 0.9072
```

정확도 90.72%

프로젝트 진행상황 - Model

Pre-processing + ELECTRA Model

> NSMC => Golden Data

기존 NSMC 데이터를 기반으로 새로 수집한 데이터를 평가

```
convert_data2dataset: 100%|██████████| 50000/50000 [00:07<00:00, 6493.65it/s]
do_evaluate: 100%|██████████| 500/500 [00:27<00:00, 18.21it/s]test_accuracy : 0.8587
```

정확도 85.87%

> Golden Data => NSMC

새로 수집한 데이터를 기반으로 기존 NSMC 데이터를 평가

```
convert_data2dataset: 100%|██████████| 94393/94393 [00:14<00:00, 6545.25it/s]
do_evaluate: 100%|██████████| 944/944 [00:31<00:00, 29.59it/s]test_accuracy : 0.8901
```

정확도 89.01%

수집한 Data가 유효한 Data임을 확인할 수 있다

=> 데이터 유효성 확인

프로젝트 진행상황 - Prototype

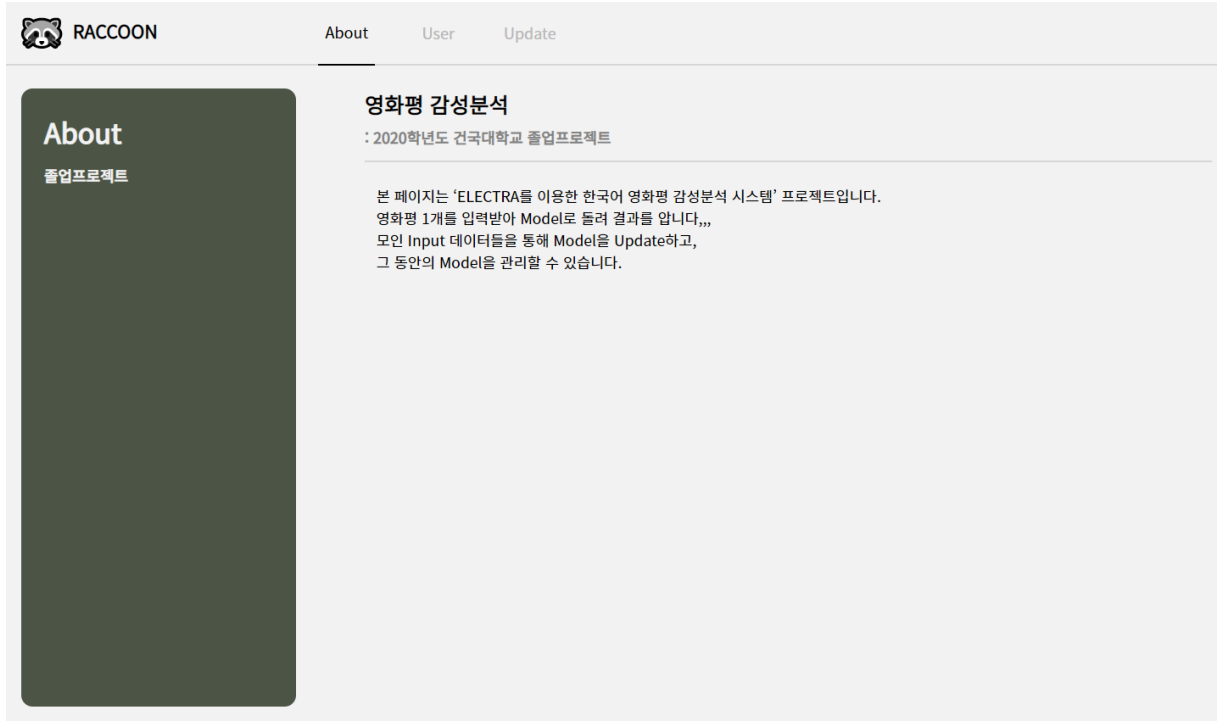
UI

Python Flask를 통해 목표 System의 UI - Web Page 구현, AWS Server / Mysql 사용

Main

팀 정보

프로젝트 소개



프로젝트 진행상황 - Prototype

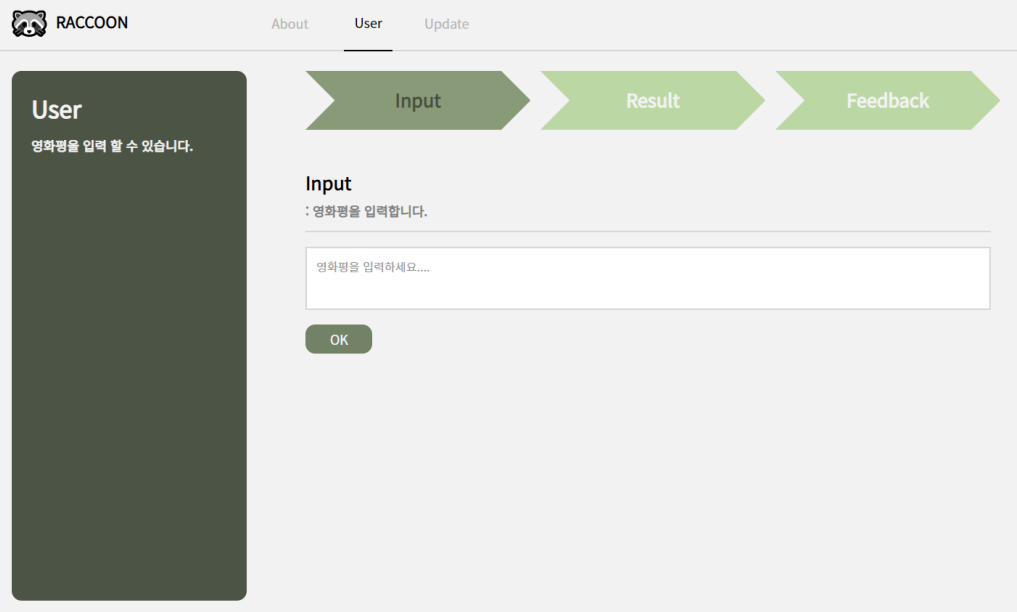
User Input

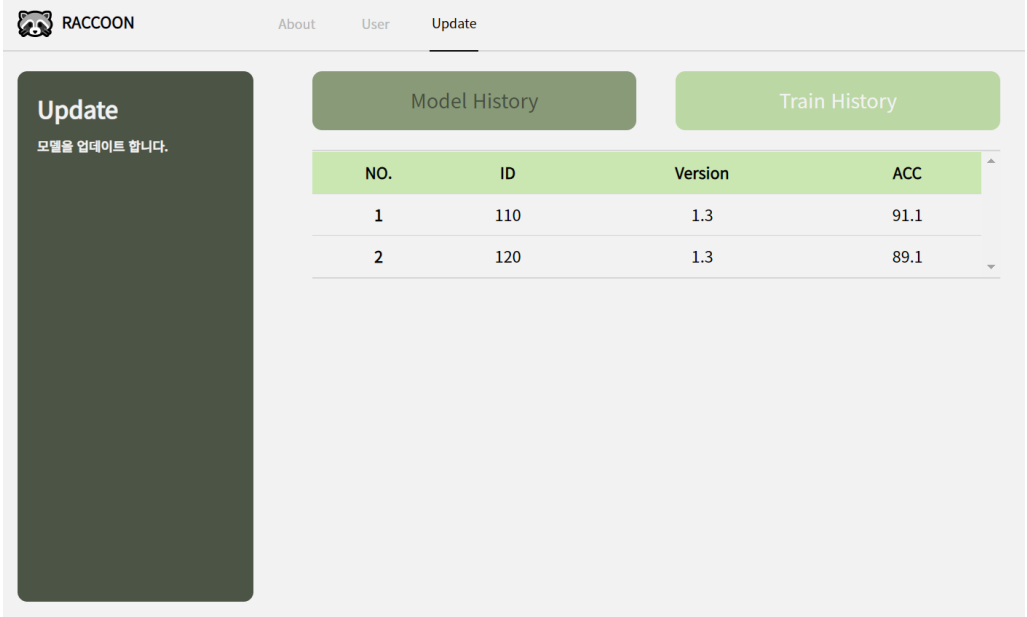
영화평 입력 후 피드백 DB에 저장

Update Model

DB기반 Model Update

Model 미완성으로
Update기능 구현하지 못함





프로젝트 진행상황 - Prototype

User의 영화평 입력 Demo

Input

영화평 입력 후
서버에 전송

프로젝트 진행상황 - Prototype

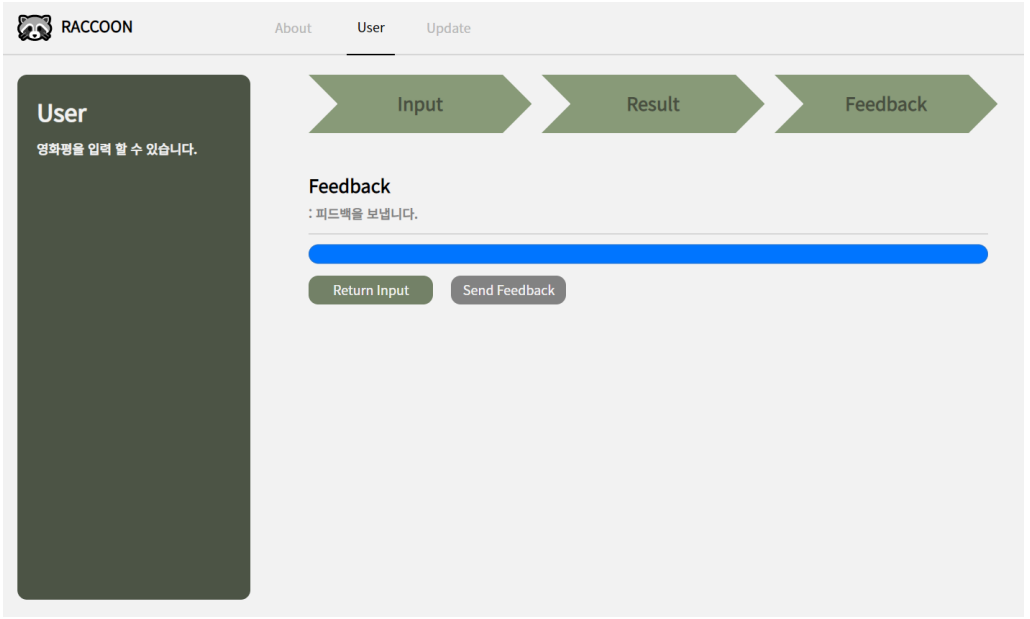
User의 영화평 입력 Demo

Result

임시 Model의 결과를 User에게 보여주고 판단 요청

Feedback

User의 판단에 따른 긍/부정 결과와 영화평 DB에 저장



향후 목표

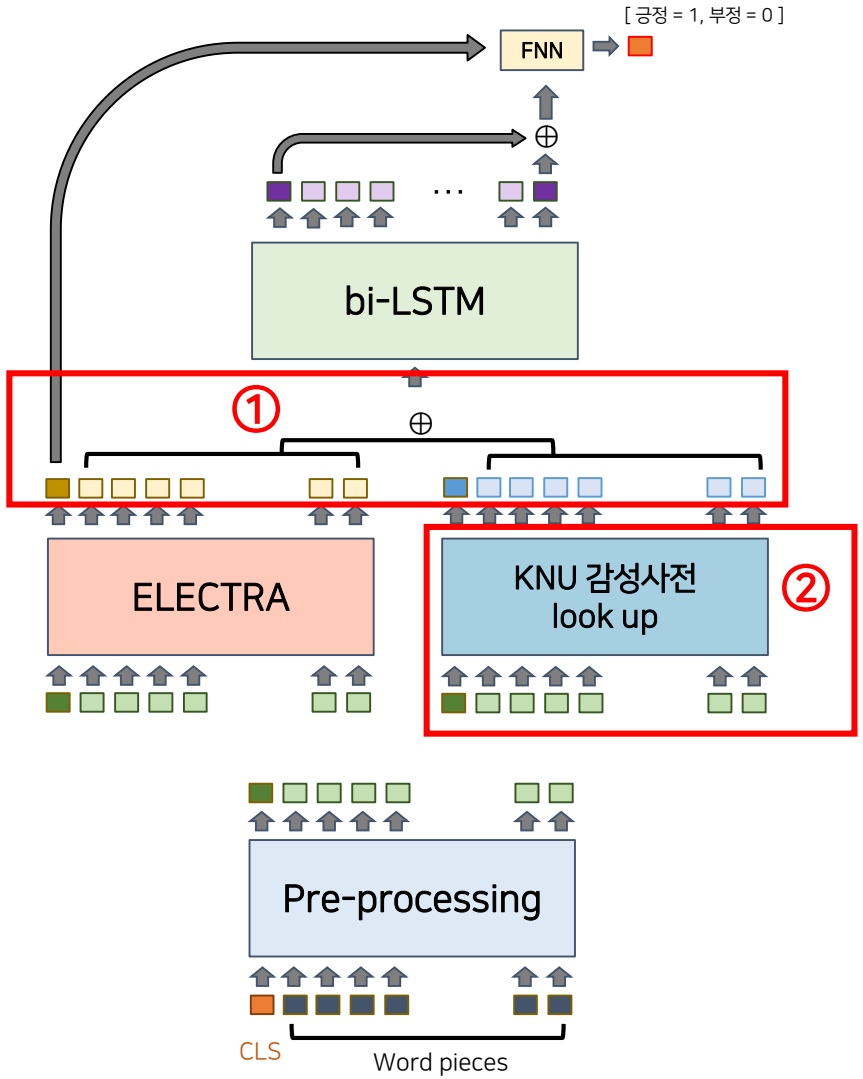
향후 목표

추가 Layer 구축

기존 설계했던 Model을 바탕으로 Layer 쌓기
(+ 각 Layer의 문제점 분석 후 보완)

=> ①KNU 감성사전과 ELECTRA의 vector mapping
(+ Input의 Token 매치)

=> ②KNU 감성사전의 Input Token의 원형화



향후 목표

정확도 개선

1. Golden Data

- 긍/부정으로 자동 분류한 영화평에서 잘못 분류되거나 긍/부정을 알 수 없는 영화평 수정

  ex. 평점은 9점이나 영화평은 '재미없어요' 인 경우

- 띄어쓰기를 추가, 삭제하거나 이모티콘을 문자로 대체하는 등의 영화평 보정
- 추가적인 golden data 생성

2. Model

- Test-set과 일치하지 않는 Data들의 특징 분석

향후 목표

KNU 감성사전 활용

- token의 형태가 원형이 아닌 경우, 간혹 어근을 찾지 못하는 문제점 발견
  ex. '재미있다'에서는 '재미있'이라는 어근을 찾지만, '재미있어'에서는 어근을 찾지 못함

- 감성사전에 등록된 척도와 문맥 내에서의 실제 감정이 일치하지 않는 경우 문제점 발생
  ex. 감성사전에서 '슬프다'는 항상 부정으로 판별되지만, 슬픈 영화의 경우 '슬프다'가 긍정으로 판별될 수 있음

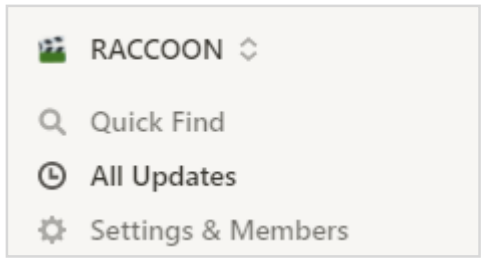
=> 문제점들을 해결할 방안 고려

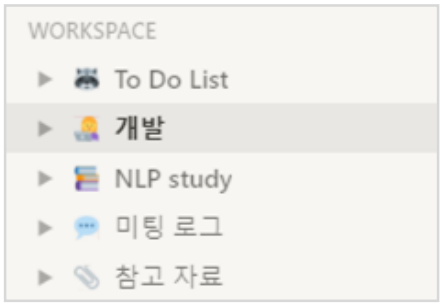
향후 목표

팀원과의 소통

- Hangout을 통한 주 1회, 5시간 이상 정기 회의

- Notion을 통한 프로젝트 진행 상황 및 문서 정리





향후 목표

국어 정보 처리 시스템 경진 대회 준비

- 유튜브 업로드
  출품작 설명, 실행 내용, 시스템 설명 등을 포함한 영상 제작

- 시스템에 관한 보고서 작성

- 출품작 1차 접수

*대회 홈페이지: http://hkd.or.kr

감사합니다