리뷰 해체 분석기

[딥러닝 기반 영화평 감성 분석]

1st Implementation Presentation

Team #4
201411273 박재범
201411275 박진호
201411283 이상민
201511244 김민우

 # 작품명: 리뷰 해체 분석기

 # 소프트웨어의 목적:

한국어로 작성된 영화평을 입력받아 해당 영화평의 긍정/부정 여부를 판단하고 통계를 확인 및 관리 가능한 웹 어플리케이션을 서비스한다. 또한 신규 데이터셋에 대해서 추가적인 학습을 진행함으로써 새로운 버전을 생성하고 선택하여 적용할 수 있으며 버전별 통계도 함께 제공하여 버전 관리와 모델 정확도 향상이 가능하고 이를 통해 지속적으로 서비스의 질을 개선할 수 있도록 한다.

해당 소프트웨어를 통해 1차적으로는 영화 산업에서 관람객 리뷰 분석을 통해 전반적인 긍/부정 비율을 파악해 상영관 확대, DVD/Blue-Ray 발매 등 영화 개봉 전후 사업 확장 또는 축소에 도움을 주는 등 다양한 분야의 시장 반응 분석에 활용할 수 있으며,

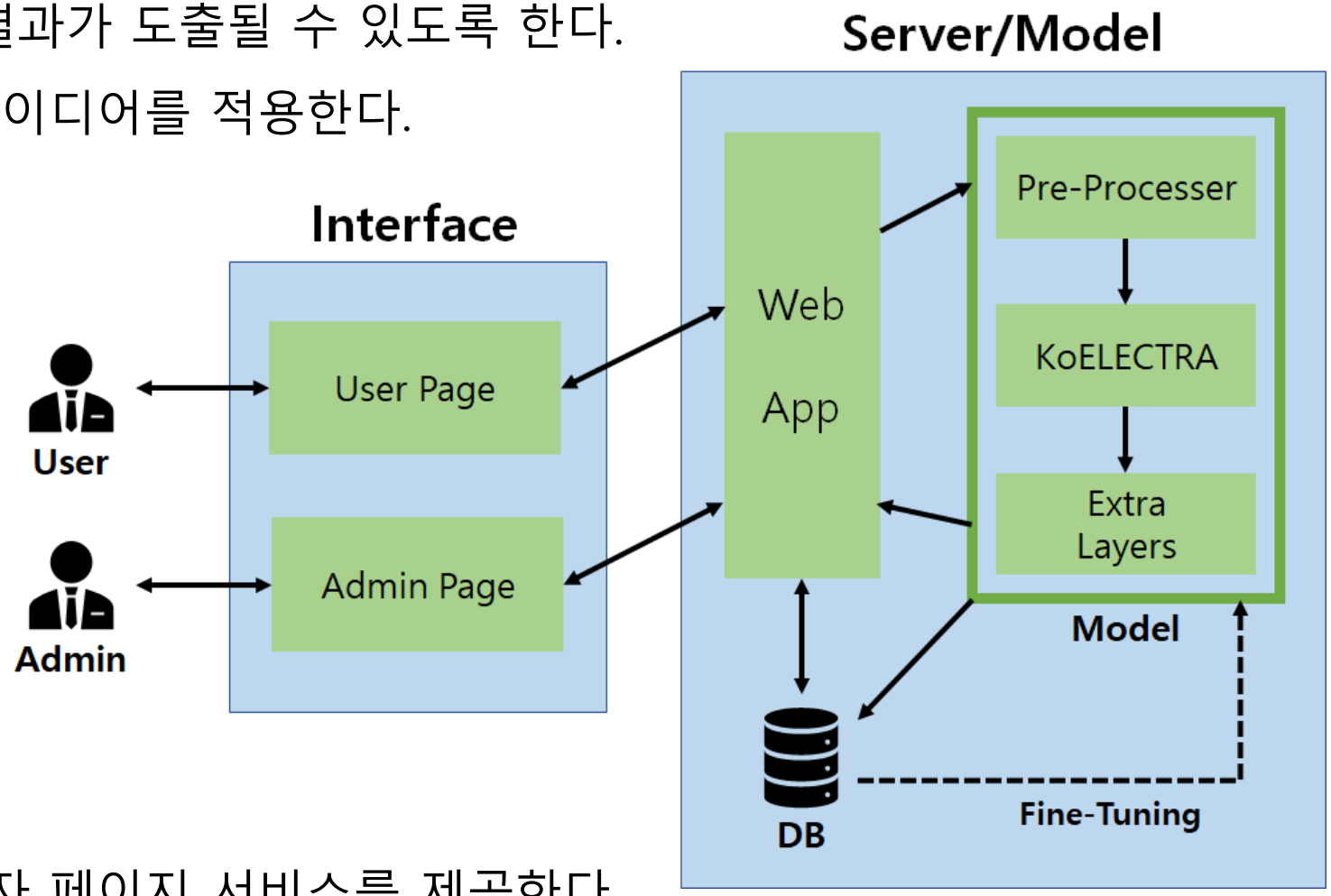
2차적으로는 한국어 감성 분석 정확도를 향상시킨 언어 모델을 얻을 수 있어 다른 분야에서도 유용하게 활용할 수 있도록 한다.

 # 언어 모델

전처리한 입력 문장에 대해 KoELECTRA 기반 인공신경망을 거쳐 긍정/부정의 Binary Classification 결과가 도출될 수 있도록 한다. 또한 정확도 향상을 위한 다양한 아이디어를 적용한다.

 # 웹 서버(Back-End)

웹 인터페이스를 통한 사용자 또는 관리자의 요청을 처리하거나 모델 서버와의 데이터 처리 및 DB 관리를 담당한다.

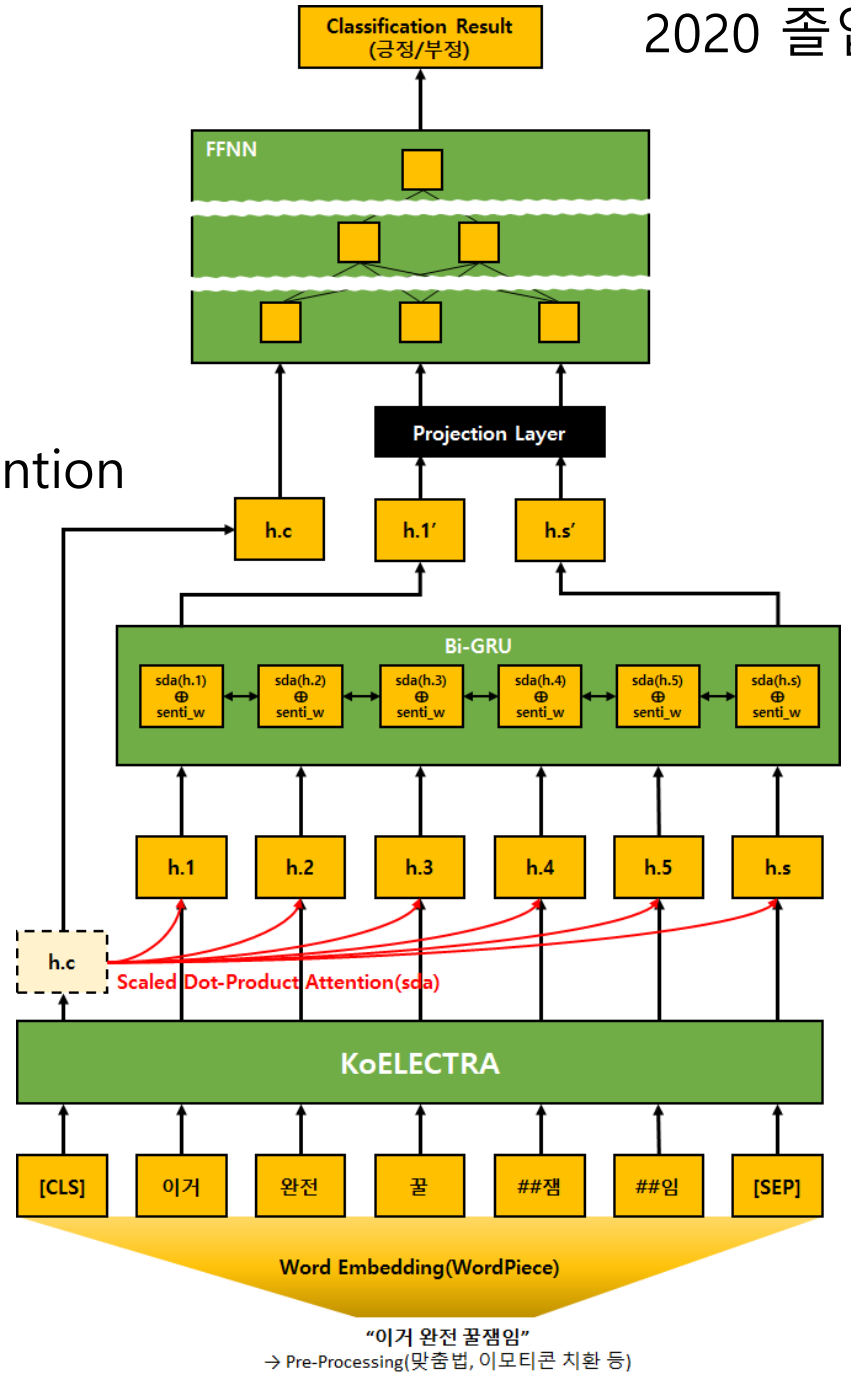


 # 웹 인터페이스(Front-End)

웹 환경에서 사용자 페이지와 관리자 페이지 서비스를 제공한다.

 # 모델 레이어의 구조 및 프로세스

1. 입력에 대한 Word Embedding(Word Piece)
2. 전체 토큰이 KoELECTRA 모델을 통과
3. CLS 토큰을 단어 토큰에 Scaled Dot-Product Attention
4. 토큰에 한국어 감성사전 가중치를 Concatenate
5. 단어 토큰이 Bi-GRU를 통과
6. Bi-GRU 양 끝 벡터의 차원을 Projection
7. CLS와 Bi-GRU 결과 토큰이 FFNN을 통과
8. 최종 Binary Classification 도출



1) 모델의 학습(Fine-Tuning)을 위한 학습 서버 확보

기본적으로 모델의 학습은 행렬, 벡터의 수많은 연산으로 이루어 진다.
이는 CPU 작업보다는 그래픽 작업을 처리하는 GPU에 최적화된 작업이다.
따라서 가정용 데스크탑이나 개인용 노트북보다 향상된 Cuda Core의 성능과 개수,
그리고 그것을 장시간 활용할 수 있는 서버 환경이 필요하다.
이를 위해 Google Colab에서 제공하는 GPU를 이용해 학습을 진행했다.

2) 성능 향상을 위한 추가적인 Neural Network Layer 설계 및 구축

본 프로젝트에서는 시간 및 자원에 대한 제한과 효율성을 고려하여 대용량 일반 데이터(말뭉치)를 통해 선행학습한 KoELECTRA를 활용한다.

이를 통해 특정 Task(본 프로젝트에서는 Binary Classification)에 대한 Fine-Tuning시 초기 학습 속도를 향상시킬 수 있으며, KoELECTRA의 출력 벡터에 추가적인 신경망 구조를 추가하여 성능을 더욱 끌어올릴 수 있도록 하였다.

첫번째 아이디어는 RNN에서 Classification에 많이 활용되는 Bi-GRU에 CLS 토큰을 제외한 단어 토큰들을 입력으로 넣고 문맥 전체 정보를 내포하고 있는 양 끝의 결과 벡터값을 취하는 것이다. 이 때, KoELECTRA의 CLS 도출에 영향을 더 많이 미친 단어일 수록 중요도를 높여주는 효과를 위해 CLS 벡터를 나머지 벡터들에 Scaled Dot-Product Attention해준다.

두번째 아이디어는 KoELECTRA의 결과 CLS 벡터와 Bi-GRU의 양 끝 벡터 2개를 FFNN을 통과시켜 2차원의 벡터로 변환하고 이를 통해 최종 Binary-Classification 결과를 도출하는 것이다.

3) 한국어 감성사전의 감성 정보 활용

표준국어대사전을 통해 14,843개의 말뭉치의 긍정/부정 값을 -2, -1, 0, 1, 2 총 5단계로 정리한 KNU 한국어 감성사전을 활용하여 모델의 정확도 향상에 활용하였다.

최종적으로 적용하기로 결정한 아이디어는 한국어 감성사전의 말뭉치를 KoELECTRA의 Vocabulary를 바탕으로 Tokenize한 뒤 출현하는 토큰들이 어떤 감성값을 가진 말뭉치에 가장 빈번하게 나타나는지를 구하고, 출현 횟수 * 감성값의 평균치를 계산하여 해당 토큰에 대응시키는 것으로 하였다.

예를 들어 '씨##'라는 토큰은 -2의 감성값, '굿'이라는 토큰은 2의 감성값을 가지게 된다.

대응값이 100% 정확하지는 않지만 통계적으로 적용할 만 하다는 판단 하에 결정하였다.

이후, KoELECTRA의 Vocabulary에서 감성값이 결정되지 않은 토큰은 가중치를 0으로, 결정된 토큰은 계산된 가중치를 설정하여 모든 토큰들에 대해 감성값을 1:1 대응시키고

5개의 감성값을 Random Initializing하여 n차원 벡터로 변환한 뒤 Scaled Dot-Product Attention을 끝낸 Bi-GRU 입력 토큰에 Concatenate해주었다.

4) 추가 학습 데이터 크롤링

기존 KoELECTRA의 학습에 활용되었던 NSMC는 15만개의 Tranning Set과 5만개의 Test Set 총 20만개이다. 이에 추가적인 학습 데이터를 제공하기 위해 네이버 영화를 기준으로 크롤링을 진행하였다. 이 때, 특정 포맷이나 감성 정보에 치우지지 않도록 17개의 장르(드라마, 판타지, 공포, 멜로/애정/로맨스, 모험, 스릴러, 느와르, 다큐멘터리, 코미디, 가족, 미스터리, 전쟁, 애니메이션, 범죄, 뮤지컬, SF, 액션)별로 대체로 긍정적인 영화 10개, 대체로 부정적인 영화 10개를 직접 선별하여 총 340개의 영화에 대해 작업하였으며, 평점 8~10은 긍정으로, 1~3은 부정으로 분류하여 최종 결과물의 긍정 부정 비율이 1:1이 되도록 하였다.

크롤링으로 추가된 데이터는 총 60만개이며, NSMC와 병합하여 Tranning Set 60만개, Test Set 20만개 총 80만개의 학습 데이터를 새롭게 구축하여 학습에 활용했다.

```
id  document    label
9479479 슬프지만 마음 따뜻해지는 영화   1
9267855 음악 연출 스토리 좋았음..대만영화 중 최고인듯   1
5775804 관객을 모욕한 영화  0
6160688 음악 하난 죽여주네  0
9460731 내생 최고의 영화라 해도 과언이 아닌 그런 삶이야기. . . 아름다운 부부애를 보며 크게 배우고, 커플이라면 부부라면,
진정한 사랑이 어떤건지 알고싶다면 꼭 봐야 할 영화.. 1
```

5) 데이터 전처리 작업

학습 데이터의 질은 모델의 정확도에 무시할 수 없는 영향을 미치는 요소이기 때문에 반드시 적절한 전처리가 필요하다.

무의미한 글자의 중복을 압축하고, 감성 정보에 영향을 미치지 않는 Html 태그 기호, 한자 등을 제거하였으며, 다음과 같은 전처리도 시도해보았다.

- 띄어쓰기 및 맞춤법 교정: Py-Hanspell 모듈의 활용을 고려하였으나 의미가 완전히 잘못 변환(ex. 너무 명감독 -> 너 무명감독)되는 경우가 많아 철회하였다.
- 이모티콘 축약: 긍정적 또는 부정적 의미를 담은 이모티콘 등을 하나의 함축된 토큰으로 치환하는 아이디어를 적용하려고 하였으나 명확한 분류가 어렵고 ㅎㅎ, ㅋㅋ 등 미묘한 차이가 있는 표현들은 유지하는 것이 낫다고 판단하여 철회하였다.

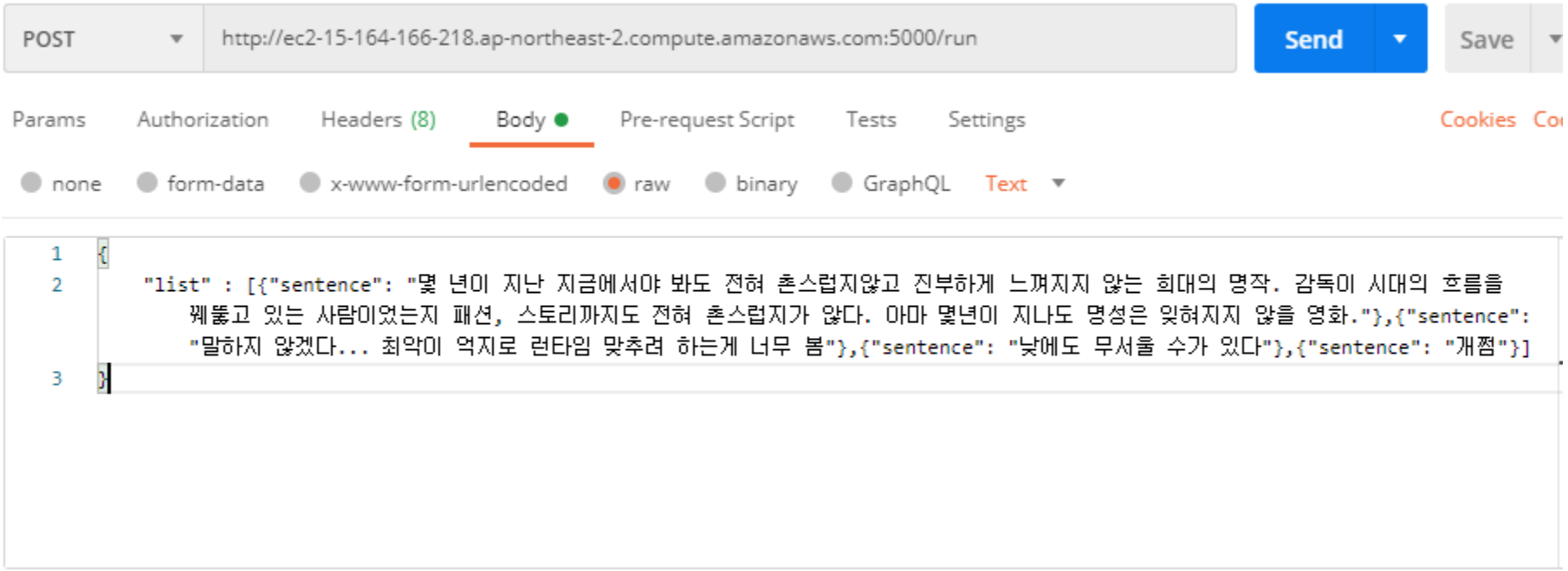
해당 전처리는 Training Set, Test Set에 적용되어 있으며, 새로운 사용자 입력에도 적용된다.

또한 크롤링 시 무의미한 데이터(ex. 123, ㅇㅇ)를 제외하기 위해 글자수 3 이하의 영화평은 제외하였다.

 # 개요

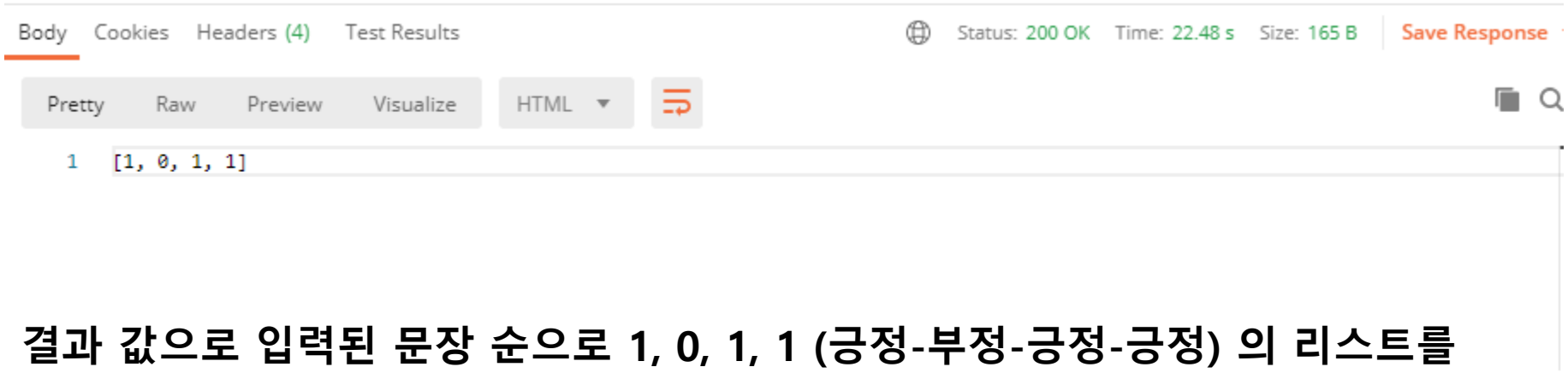
Python의 웹 프레임워크 중 하나인 Flask로 개발하였으며, 개발한 언어 모델을 서버에 올려서 인터페이스로 부터 REST API로 Json Body를 받고, Body를 Parsing 하여 Sentence를 추출한다. 추출한 Sentence들을 각 Token으로 나누어 모델을 통과시키고 긍/부정을 도출하여 인터페이스로 긍정 : 1, 부정 : 0 을 담은 리스트를 Json Body에 담아 Response를 전송한다. 구현한 서버는 AWS EC2에 탑재하여 운영한다.

1) Request



API 테스트 툴인 Postman을 사용하여 Sentence들을 Json body에 넣어 POST로 전송한다.

1) Response



결과 값으로 입력된 문장 순으로 1, 0, 1, 1 (긍정-부정-긍정-긍정) 의 리스트를 Response로 전달 받는다.

2) 서버 상에서의 처리

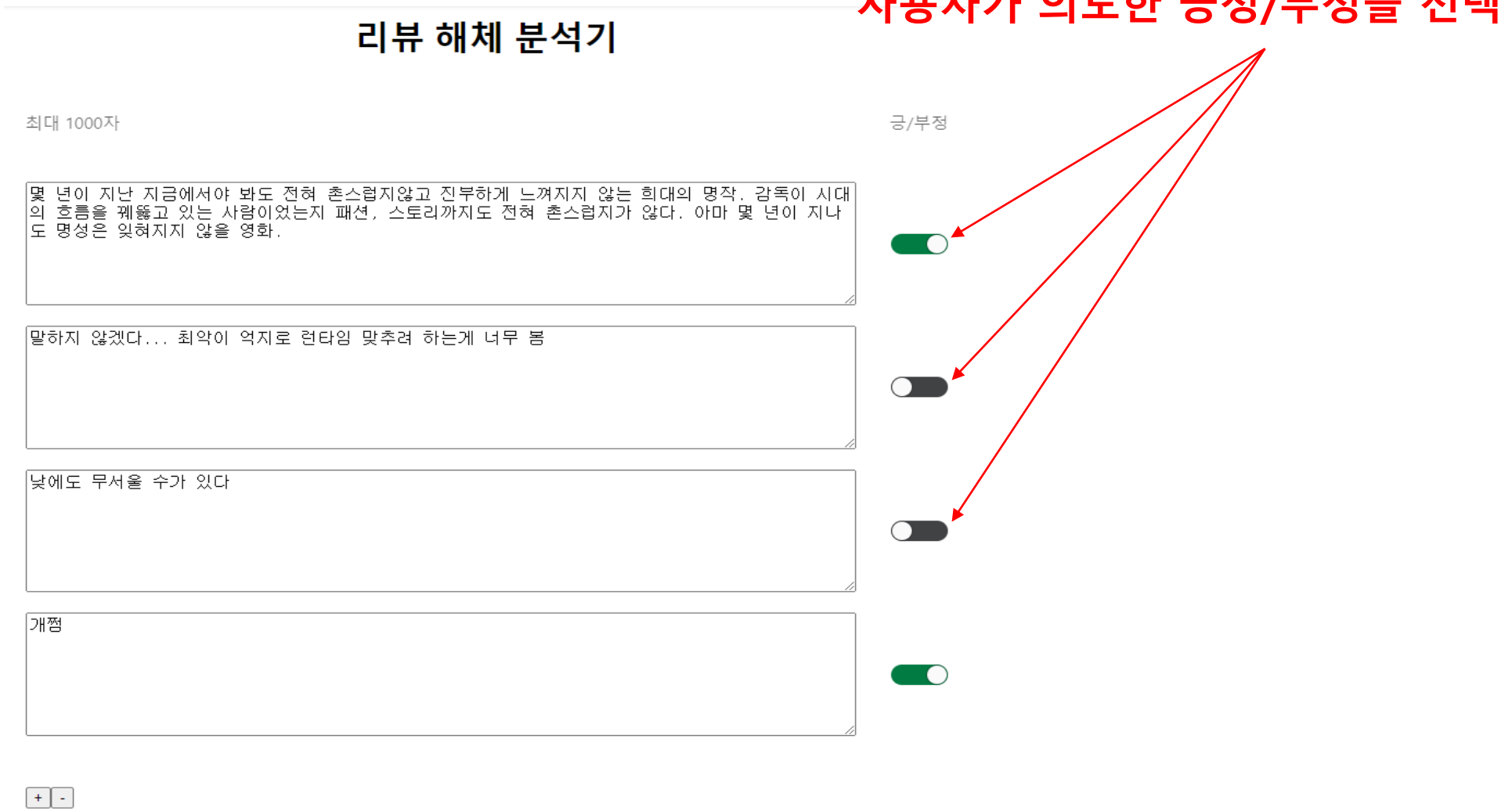
```
몇 년이 지난 지금에서야 봐도 전혀 촌스럽지않고 진부하게 느껴지지 않는 희대의 명작. 감독이 시대의 흐름을 꿰뚫고 있는 사람이었는지 패션, 스토리까지도 전혀 촌스럽지가 않다. 아마 몇년이 지나도 명성은 잊혀지지 않을 영화.
generated tokens:  ['몇', '년이', '지난', '지금', '##에서', '##야', '봐도', '전혀', '촌', '##스럽', '##지않', '##고', '진', '##부', '##하게', '느껴', '##지지', '않는', '희', '##대의', '명', '##작', '.', '감독이', '시대의', '흐름을',
 '꿰', '##뚫', '##고', '있는', '사람이', '##었는지', '패션', ',', '스토리', '##까지도', '전혀', '촌', '##스럽', '##지가', '않다', '.', '아마', '몇년', '##이', '지나', '##도', '명', '##성은', '잊혀', '##지지', '않을', '영화', '.']
말하지 않겠다... 최악이 억지로 런타임 맞추려 하는게 너무 봄
generated tokens:  ['말하지', '않겠다', '.', '.', '.', '최악', '##이', '억지로', '런', '##타임', '맞추', '##려', '하는', '##게', '너무', '봄']
낮에도 무서울 수가 있다
generated tokens:  ['낮', '##에도', '무서', '##울', '수가', '있다']
개쩜
generated tokens:  ['개', '##쩜']
[1, 0, 1, 1]
```

입력 받은 Sentence들에 대해 Token을 Generate하고 1,0,1,1 의 결과를 도출한다.
현재 서버는 EC2로 운영중이므로
http://ec2-15-164-166-218.ap-northeast-2.compute.amazonaws.com:5000/run **으로 Post Request를 하면 테스팅할 수 있다.**

 # 개요

React JS 프레임워크로 개발하였으며, MVC 모델 구조를 적용하여 유지보수 및 관리가 수월하도록 설계하였다.
사용자 페이지에서는 유저가 리뷰를 입력하고 입력한 리뷰의 긍정/부정 여부를 계산할 수 있도록 구현하고 유저가 입력한 결과와 모델이 예측한 결과를 비교하고 긍정/부정 비율의 통계를 출력해 직관적으로 확인할 수 있도록 하였으며,
관리자 페이지에서는 현재까지 입력된 데이터의 수, 버전, 모델 별 정답률을 출력하고 원하는 버전을 선택할 수 있도록 하였다.

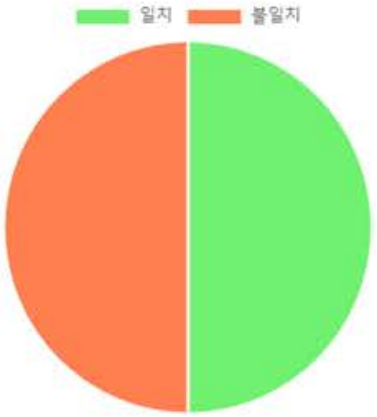
1) 사용자 페이지(메인)


사용자가 의도한 긍정/부정을 선택
리뷰 해체 분석기
최대 1000자
긍/부정
몇 년이 지난 지금에서야 봐도 전혀 촌스럽지않고 진부하게 느껴지지 않는 희대의 명작. 감독이 시대의 흐름을 꿰뚫고 있는 사람이었는지 패션, 스토리까지도 전혀 촌스럽지가 않다. 아마 몇 년이 지나도 명성은 잊혀지지 않을 영화.
말하지 않겠다... 최악이 억지로 런타임 맞추려 하는게 너무 봄
낮에도 무서울 수가 있다
개쩜
+
-
완료
TextArea를 추가, 삭제(+-버튼)
최종 제출을 위한 완료 버튼

2) 사용자 페이지(결과)

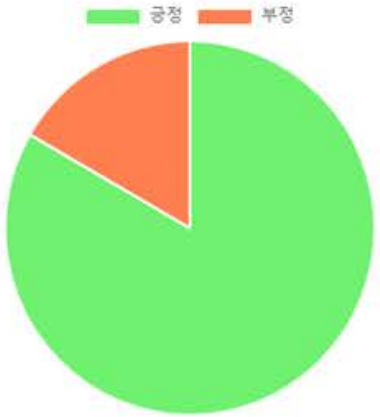
전체 분석 결과

No.	입력된 예상 결과	일치여부
1	긍정	일치
2	긍정	불일치
3	긍정	일치
4	부정	불일치
5	부정	불일치



일치 : 3건, 불일치 : 3건

정답률 50%



긍정 : 5건, 부정 : 1건

전체 결과

긍정 - 83% (5건)
부정 - 17% (1건)

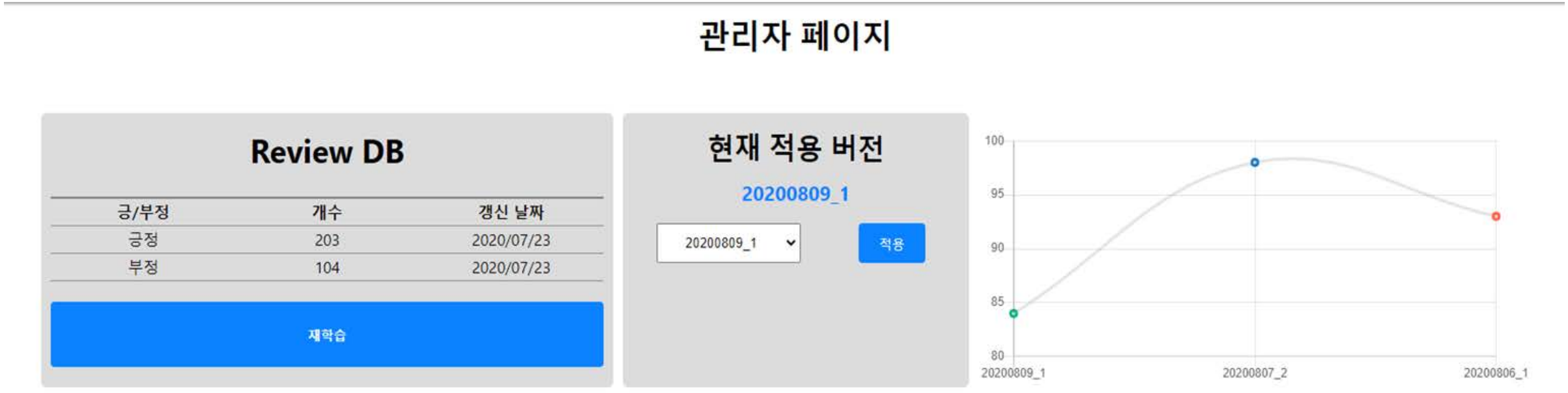
으로 전체적으로 "긍정"입니다.

유저의 입력 결과와 모델이 예측한 결과의 일치 여부

실제 모델이 예측한 결과

전체적인 긍정/부정 정도

3) 관리자 페이지(메인)



지금까지 입력받은 영화평의 수와 결과를 저장, DB의 데이터를 활용한 재학습

모델의 버전 선택

버전 별 정답률

 # 언어 모델

학습 모델의 Hyper Parameter(Batch Size, Epoch)를 조절하면서
Fine-Tuning을 진행하여 모델의 가중치 및 정확도를 최적화하는 과정을 진행하고
모델 버전 관리, DB 및 웹 서버와의 연동을 위한 추가 작업을 진행할 예정이다.

 # 웹 서버(Back-End)

웹 인터페이스 요청을 통해 모델이 학습할 수 있도록 제어하는 기능과
DB 기반 모델 버전 관리 기능 구현을 진행할 예정이다.

 # 웹 인터페이스(Front-End)

엑셀 파일을 통한 대규모 데이터 일괄 입력 기능을 추가하고
현재 구축된 템플릿을 바탕으로 UI/UX를 개선할 예정이다.

 # 2020 국어 정보 처리 시스템 경진 대회 일정을 고려하여 병행

1차 접수: ~9/15(GP: SRS 2nd 제출)

1차 발표: 9/21(GP: STP 제출)

2차 접수: ~9/29(GP: SDS 제출)

2차 심사: 10/6(GP: 2nd Demo)

 # 지속적인 디버깅 및 예외처리

프론트엔드-백엔드-모델 연동 간 발생할 수 있는 버그 또는

여러 사용자의 동시 입력 처리를 위한 제어나 예외처리를 신경써서 완성도를 높일 계획이다.