**REPORT**

기계학습 기말 대체과제 보고서



|  |  |
| --- | --- |
| **과목명** | 기계학습 |
| **학과** | 컴퓨터공학과 |
| **학번** | 202121859 |
| **이름** | 이을호 |
| **제출일자** | 2024.06.12 |

1. **감정 분석의 기본 이해**

감정 분석은 텍스트로 데이터가 주어지면 이를 단어 단위 및 어조로 분석을 진행하여 긍정, 부정 또는 감정이 없는지에 대한 여부를 확인하는 방법이다. 이런 감정 분석의 기법이 중요한 이유는 쇼핑몰이나 영화, 여러 제품에 대한 소비자의 감정, 고객의 서비스 응대 시에 분석을 적용하면 이를 통해 더 효율적인 대응이 가능하기에 중요한 기술로 분류된다. 해당 과제에서는 아마존 쇼핑몰의 신발 제품별 리뷰 데이터에 대해서 감정 분석에 대하여 nltk - vader 모델과 logistic regression (회귀 분석) 기법을 사용하여 모델의 생성 및 예측 그리고 성능 평가를 진행한다.

1. **데이터 전처리**
2. **우선, 필요 라이브러리를 먼저 설치를 진행한다.**

텍스트, 스크린샷, 디스플레이, 폰트이(가) 표시된 사진

자동 생성된 설명

1. **라이브러리 설치가 끝나면 데이터 로딩을 진행한다.**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

데이터를 확인해보면 구조가 하단과 같이 구성되어 있는 것을 확인 가능하다.

<amazon\_uk\_shoes\_products\_dataset\_2021\_12.csv>의 데이터 구조>

1. url (상품 URL 주소)

2. product\_name (상품명)

3. reviewer\_name (리뷰 등록자명)

4. review\_title (리뷰 제목)

5. review\_text (리뷰 내용)

6. review\_rating (상품 리뷰 별점)

7. verified\_purchase (실구매 확인 여부)

8. review\_date (리뷰 작성 날짜)

9. helpful\_count (해당 리뷰가 도움이 된 사람의 수)

10. uniq\_id (리뷰 특정 ID)

11. scraped\_at (해당 리뷰를 가져온 시점)

1. **다음은 데이터 내의 빈 데이터를 없애는 과정인 결측치 제거를 진행한다.**

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

결측치의 여부를 확인해보면 review\_title과 review\_text 그리고 helpful\_count가 결측치가 있는 것으로 확인이 가능한데, helpful\_count 행은 제거하고 나머지 결측치가 존재하는 행은 제거하는 것으로 결측치 제거 과정을 진행한다, (결측치 처리 전, 후 이미지)

결측치 제거가 완료되면 감정 분석에 사용 가능한 의미가 있는 상품명, 리뷰 제목, 리뷰 작성자, 리뷰 내용, 별정을 제외하고는 모두 제거하겠다, 하단의 이미지는 제거 후 데이터의 모습이다.

텍스트, 스크린샷, 폰트, 블랙이(가) 표시된 사진

자동 생성된 설명

1. **노이즈 제거 (정규식 활용)**

다음은 영어를 기반으로 감정 분석을 진행하기에 기본의 리뷰 제목과, 리뷰 내용에서 영어와 공백을 제외하고 다른 문자는 모두 제거하는 과정을 정규식을 통해 진행한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

1. **토큰화 진행 (nltk 활용)**

다음은 문장을 토큰화를 통해 단어 단위로 분리하여 새로 저장한다. 토큰화 기능은 nltk의 word\_tokenize를 통해 진행한다.

먼저, vader 모델을 감정 분석기로 초기화를 진행하고 감정 분석을 진행하는 함수를 정의한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

이후에는 -1 ~ 1 사이의 소수점 값으로 감정 분석 결과가 나오는 것을 이진 값으로 변환하는 change\_to\_binary 함수를 정의한다. Logistic regression에 값을 전달하기 위해서는 이진 값으로 전달해야 하기 때문에 값 전환이 필수적이다.

텍스트, 스크린샷, 디스플레이, 폰트이(가) 표시된 사진

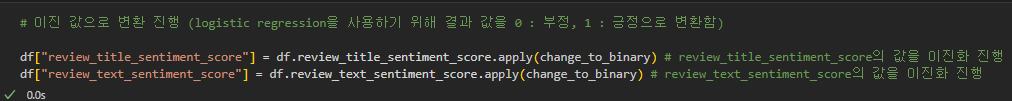
자동 생성된 설명

감정 분석을 진행하고, review\_title, review\_text의 값을 각각 분석을 진행하여 새로운 열로 데이터 프레임에 값을 추가하는 과정을 진행한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

감정 분석 값이 -1 ~ 1까지의 소수점 값으로 전달되면 이를 이전에 정의해둔 이진 함수를 통해 변환 저장하는 과정을 진행한다.



이후에는 모델에 적용할 데이터로 product\_name(상품명), review\_text\_token(리뷰 내용), review\_text\_sentiment\_score(리뷰 내용 감정 분석 결과)를 데이터 프레임으로 만들어 다음 단계로 넘어간다.

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

1. 모델 선택 및 학습

감정 분석의 경우에는 nltk의 vader 모델만으로도 가능하지만, 문제에서 요구하는 모델의 학습 및 평가 수치를 계산하는 과정은 회귀 분석을 사용해야 한다고 판단하였다. 따라서 감정 분석의 데이터 학습 및 예측에 가장 적합한 logistic regression을 선택하여 적용을 진행하였다.

1. **Logistic regression에 필요한 scikit-learn 모듈을 import를 진행**

스크린샷, 텍스트, 폰트이(가) 표시된 사진

자동 생성된 설명

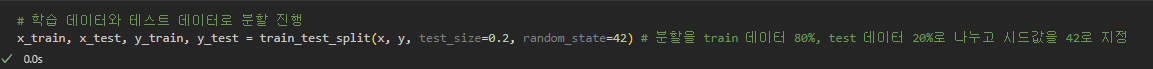
1. **데이터 지정 및 리뷰 토큰화 데이터 벡터화 진행**

벡터화를 통해 리뷰 내용의 단어의 등장 빈도를 나타내는 TF-IDF 벡터로 변환하여 적용을 진행하는데, 이런 상위 단어 등장 빈도 갯수를 의미하는 max\_features 값이 1000을 넘어가면 커널이 불완전 종료되는 현상에 1000으로 고정을 시켜두고 진행하였다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

1. **Train data, test data로 분할 진행**

위에서 x, y에 지정한 데이터를 test 데이터와 train 데이터로 2:8 비율로 분할하는 과정을 진행한다.  


1. **logistic Regression 모델 초기화 및 그리드 서치 활용 최적 모델 출력**

이 단계에서는 회귀 모델을 초기화하고, 하이퍼파라미터 값을 0.001 ~ 100 사이의 값으로 지정하여 그리드를 설정하고, 그리드 서치를 통해 최적의 하이퍼파리미터를 탐색하는 과정을 진행하고 마지막에는 모델에 성능이 가장 높고 적합한 하이퍼파라미터 값을 출력하고 교차 검증 점수인 cross-validation score를 출력하는 과정을 진행한다.

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

1. **상단 과정에서 찾은 하이퍼마라미터에 해당하는 모델을 활용하여 학습 및 예측 진행**

4)번 과정에서 찾은 최적의 하이퍼파라미터 값 C:10을 기반으로 최적 모델을 final\_model로 정의하여 최종 모델 예측을 진행한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

이제 모델에 대한 학습과 예측의 단계가 완료되었으니 모델의 예측 성능을 평가하는 과정으로 넘어가도록 하겠다.

1. **평가 (정확도, 정밀도, 재현율, F1 점수, ROC-AUC)**

모델 평가 단계에서는 위에 학습을 마친 예측 모델 y\_pred를 기반으로 정확도, 정밀도, 재현율, F1 점수, ROC-AUC 지표를 통해 모델 평가를 진행한다. 이후에는, 예측 결과를 통해 오버피팅, 언더피팅을 확인한다.

우선, 평가 지표의 출력의 코드와 결과는 하단과 같이 나온다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

세부적으로 수치 값을 통해 모델의 성능을 자세하게 분석해보겠다.

**1. 정확도 (Accuracy: 0.8892149669845928)** : 모델의 예측 성능을 의미하는 값으로 약 88.9%의 성능을 보이며, 예측의 정확도가 높음을 확인 가능하다.

**2. 정밀도 (Precision: 0.8896068492093696)** : 모델이 예측한 값 중 실제 데이터가 일치한 비율을 의미하는 값으로 약 88.9%의 성능을 보이며, 정밀도 역시 높음을 확인 가능하다.

**3. 재현율 (Recall: 0.8892149669845928)** : 실제 데이터 중 모델이 예측한 값이 일치한 비율을 의미하는 값으로 88.9% 성능으로 재현율 역시 높음을 확인 가능하다.

**4. F1 점수 (F1 Score: 0.8893294648740914)** : F1 점수는 정밀도와 재현율의 조화 평균을 의미하는데 88.9% 인 것을 확인하면 모델의 성능에 문제가 없음을 확인 가능하다.

**5. ROC-AUC 점수 (ROC-AUC Score: 0.8888145515823296)** : 모델의 성능을 평가하는 값으로 88.8%로 비교적 높은 성능을 보이는 것을 확인 가능하다.

**전체 점수 보고(Classification Report)**를 보면, 긍정 감정 분석의 예측이 부정 감정 분석의 예측 정확도 보다 높은 것을 확인 가능하다.

종합적으로 평가를 진행하면 성능이 아주 높은 편은 아니지만 그래도 어느 정도 정확한 성능인 88% 수준으로 예측이 가능한 모델이며 데이터 성능을 보면 정확도, 정밀도, 재현율의 성능이 유사하기에 오버피팅, 언더피팅에는 속하지 않으며 정상적인 결과를 내는 모델임을 확인 가능하다.

1. **시각화 및 해석 (혼동 행렬 및 ROC 곡선)**
2. **혼동 행렬 생성 및 출력**

Scikit-learn의 confusion\_matrix 기능을 통해 모델의 혼동 행렬을 생성한다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

================================

- TP : 실제 데이터 양성 -> 예측 데이터 양성

- TN : 실제 데이터 음성 -> 예측 데이터 음성

- FP : 실제 데이터 음성 -> 예측 데이터 양성

- FN : 실제 데이터 양성 -> 예측 데이터 음성

================================

혼동 행렬의 해석은 ( [[533 69][82 679]] ) : [[TN, FP][FN, TP]]으로 해석하며 예측이 맞은 데이터와 틀린 데이터를 음성, 양성에 따라 나눈 지표로 양성 데이터 679개, 음성 데이터 533개가 예측에 성공하고, 양성 데이터 69, 음성 데이터 82가 예측에 실패한 것을 확인 가능하다. 혼동 행렬로 보면 예측 실패 데이터가 아직은 많음으로 조금은 모델을 개선할 필요성이 보인다

1. **Matplotlib를 사용하여 ROC 곡선 생성**

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Roc curve의 경우 하단과 같이 출력된다.

텍스트, 라인, 스크린샷, 그래프이(가) 표시된 사진

자동 생성된 설명

Roc curve는 모델의 예측 성능을 보여주는 대표적인 시각화 지표로서 위 roc 곡선을 분석하면 모델의 성능이 왼쪽 상단의 모서리에 가까울수록 성능이 좋으나 roc\_auc 점수가 88.8%로 낮은 편은 아니며 더 성능을 개선하기 위해서는 데이터의 전처리 과정에서 조금 더 불순한 데이터를 걸러내고, 데이터의 양을 방대하게 증가시켜 많은 데이터를 모델에 학습시키면 성능이 더 높아질 것으로 기대한다. 또한, 커널 불완전 종료 이슈로 더 올리지 못한 TF-IDF 진행 과정에서 정해놓은 단어의 갯수를 1000개에서 더 많은 갯수로 조정하고 하이퍼 파라미터의 값을 더 세밀하게 조정하면 성능이 더욱 향상 될 것으로 기대한다.

건국대학교 글로컬캠퍼스

컴퓨터공학과

이을호