**기계학습 기말고사 대체 과제 보고서**

**소프트웨어학과 201920979 윤성주**

**1. 감성 분석의 기본 이해**

감성 분석은 텍스트나 문서의 감정적인 내용을 이해하고 분류하는 자연어 처리 기술입니다. 이 기술은 주로 기계학습과 자연어 처리 기법을 활용하여 특정 텍스트의 긍정적인지 부정적인지, 혹은 중립적인지를 분석한다.

감성 분석의 응용으로는 금융 시장 예측(뉴스 기사나 금융 보고서를 분석하여서 시장의 감정적 반응을 파악하고 주가 등 금융 변수의 변동성을 예측), 의료 응용(환자의 의료 기록이나 의견을 분석하여 질병 관리나 치료의 효과를 평가), 사회적 미디어 분석(SNS등에서 사용자들의 감정을 분석하여 트렌드를 파악)등에 사용할 수 있다.

**2. 데이터 전처리**

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

우선 데이터를 전처리 하기위해서 데이터를 가져와야 하는데 홀수 학번이므로 amazon\_uk\_shoes\_products\_dataset\_2021\_12.csv 데이터를 사용했다.

그래서 아래 데이터를 잘 로드한 모습이다. 그러나 내가 필요한 데이터만 뽑아서 쓰기위해서 아래 사진처럼 필요한 열만 선택 후 데이터를 확인했다.

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

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

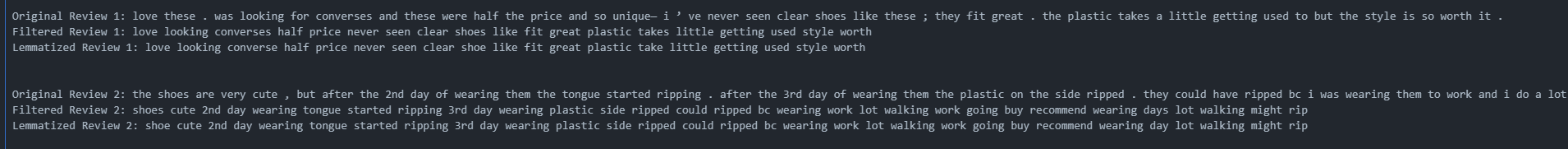
자동 생성된 설명

df.info()를 통해서 데이터 정보를 확인하였고 다음과 같이 출력됐다.

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

결측 값을 제거한 후 데이터 정보를 확인했다.



결측 값 제거, 토큰화 및 불용어를 제거 그리고 텍스트 정규화를 진행하고 나온 결과값이다. 출력 값을 보면 처음 Original Review 1과 filtered Review 1, Lemmatized Review 1로 출력이 됐다. 하나를 예시로 보자면 오리지날 리뷰에는 사용자가 적은 텍스트가 모두 적혀 있었으나 filtered Review를 보면 불용어가 제거되어서 출력되었다. 마지막으로 Lemmatized Review를 보면 takes가 take로 바뀌는 등 표제어로 추출해서 나타냈다.

**3. 모델 선택 및 학습**

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

감정 분석에 대한 레이블을 생성했다. 80%는 학습으로 나머지 20%는 테스트로 설정하였다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

학습 세트와 테스트 세트의 일부 데이터를 출력하였다. 결과 값처럼 review\_text에 대한 sentiment\_score가 출력되었다. 1에 가까울수록 감정 점수가 긍정적이고 -1에 가까울수록 감정 점수가 부정적이다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

하이퍼파라미터와 그리드를 설정하였다. 그리고 모델 학습을 진행시켰다.



최적의 파라미터가 다음과 같이 출력되었고, 최고의 교차검증 정확도는 0.857460이 나왔다. 점수가 높게 나온 걸 보니 견고성이 높은 것 같다.

**4. 평가**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

다음은 테스트 세트를 예측, 분류 리포트, 정확도를 출력하였다.

Precision은 정밀도 recall은 재현율 F1-score은 F1점수 support는 각 클래스에 속하는 샘플의 수를 나타낸다.

Accuracy는 정확도를 나타내는데 모델이 전체 샘플 중에서 올바르게 예측한 샘플의 비율이다. 전체 클래스에 대해 평균적으로 정확도를 계산한다.

이 모델의 전체적인 정확도는 약 88.7%가 나온다. 각 클래스별로는 positive 클래스가 가장 높은 정밀도, 재현율, F1-score를 보이고, negative 클래스가 가장 낮은 성능을 보여준다.

따라서, 이 모델은 positive 클래스에 대해 높은 정확도와 성능을 보이고 있고, neutral 클래스 역시 상당히 높은 성능을 보여준다. 하지만 negative 클래스에 대해서는 좀 더 개선이 필요한 상황이다.

텍스트, 폰트, 소프트웨어, 스크린샷이(가) 표시된 사진

자동 생성된 설명

ROC-AUC를 계산을 하였더니 0.949라는 높은 점수가 나왔다. 오버피팅은 너무 많은 것을 학습시켜서 학습자료를 제외한 나머지를 제대로 인지하지 못하는 것이고 언더피팅은 너무 적은 데이터를 학습시켜서 대상이 아닌 자료도 대상으로 판단하는 것인데 ROC-AUC 점수가 높게 나와서 오버피팅과 언더피팅을 하지 않아도 될 것 같다.

**5. 시각화 및 해석**

텍스트, 스크린샷, 번호, 도표이(가) 표시된 사진

자동 생성된 설명

다음은 혼동 행렬을 시각화한 모습이다. X축이 예측, Y축이 실제이다.

실제 부정이고 부정적이라고 예측한 것은 121개, 실제 부정이나 중립으로 예측한 것은 36개 실제 부정이나 긍정이라고 예측한 것은 68개이다.

실제 중립이고 부정적이라고 예측한 것은 12개, 실제 중립이고 중립이라고 예측한 것은 333개, 실제 중립이고 긍정이라고 예측한 것은 30개이다.

실제 긍정이고 부정적이라고 예측한 것은 12개, 실제 긍정이고 중립이라고 예측한 것은 47개, 실제 긍정이고 긍정이라고 예측한 것은 704이다.

기계가 학습할 시 긍정적으로 학습시킨 것이 제법 많이 보였다. 실제 부정인데 긍정이라 예측한 것이 좀 많으므로 이부분을 주의해서 기계학습을 진행해야 할 것 같다. 텍스트, 라인, 그래프, 도표이(가) 표시된 사진

자동 생성된 설명

다음은 ROC 곡선을 시각화해서 출력한 값이다.

negative 클래스의 점수는 0.93

neutral 클래스의 점수는 0.96

positive 클래스의 점수는 0.96

Micro-average의 점수는 0.96

Macro-average의 점수는 0.95가 나왔다.

Micro-average는 클래스별로 성능 지표를 계산하기 전에 모든 클래스를 하나의 클래스로 간주하고 전체적인 성능 지표를 계산해준다.

Macro-average는 클래스별 성능 지표를 각각 계산 후 평균을 내는 방식이다.

그래프의 점수가 다들 1에 가깝게 나와서 모두 점수가 높게 나오는 모습이였다. 그러나 역시 negative클래스가 다른것에 비해서 점수가 낮게나오는 모습을 볼 수 있다.

다음에 이 데이터를 학습시킬때는 negative부분에 더 유의해서 학습을 진행하여야겠다고 생각했다.