엠블럼, 로고, 등록 상표, 원이(가) 표시된 사진

자동 생성된 설명**기계학습 기말 대체과제 보고서**

제품 review 데이터의 감성 분석

|  |  |  |  |
| --- | --- | --- | --- |
| **제출일** | 2024.06.22. | **전공** | 소프트웨어학과 |
| **과목** | 기계학습 | **학번** | 201921034 |
| **담당교수** | 백우진 교수 | **이름** | 장대호 |

**1. 감성 분석의 기본 이해 : 감성 분석이 무엇이며 그 응용에 대해 학습**

감성 분석이란 텍스트 데이터를 분석하여 그 안에 담긴 감정이나 의견을 파악하는 과정입니다. 이는 주로 긍정, 부정, 중립 등의 감정 범주로 분류됩니다. 감성 분석은 소셜 미디어, 고객 리뷰, 설문 조사 응답 등 다양한 텍스트 데이터 소스에서 사용자들의 감정이나 의견을 이해하는 데 활용됩니다.

**2. 데이터 전처리 : 머신 러닝 작업을 위해 데이터 전 처리하는 경험을 쌓음**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

2.1. 필요한 컬럼만 선택

- df = df[['reviews.text', 'reviews.rating']]

: reviews.text와 reviews.rating 컬럼만 선택하여 새로운 데이터프레임을 만듭니다.

2.2. 결측값 처리

- df.dropna(subset=['reviews.text', 'reviews.rating'], inplace=True)

: reviews.text와 reviews.rating 컬럼에 결측값이 있는 행을 제거합니다.

2.3. 텍스트 정리 함수 정의

- text.lower(): 텍스트를 소문자로 변환합니다.

- re.sub(r'[^a-zA-Z\s]', '', text): 특수 문자 및 숫자를 제거합니다.

- word\_tokenize(text): 단어 단위로 토큰화합니다.

- [word for word in words if word not in stopwords.words('english')]: 불용어를 제거합니다.

- WordNetLemmatizer().lemmatize(word): 단어를 표제어로 변환합니다.

2.4. 텍스트 전처리 적용

- df['filtering\_text'] = df['reviews.text'].apply(preprocess\_text)

: preprocess\_text 함수를 사용하여 reviews.text 컬럼의 텍스트를 전처리하고, 이를 새로운 filtering\_text 컬럼에 저장합니다.

2.5. 감성 레이블 추가

- df['sentiment'] = df['reviews.rating'].apply(lambda x: 'positive' if x >= 4 else 'negative')

: reviews.rating 값이 4 이상이면 'positive', 그렇지 않으면 'negative'로 레이블을 설정합니다.

2.6. 데이터셋 분할

x = df['filtering\_text']

y = df['sentiment']

: filtering\_text 컬럼을 입력 변수 x로, sentiment 컬럼을 출력 변수 y로 분할합니다.

2.7. 전처리된 데이터를 CSV 파일로 저장

- df.to\_csv('filtering\_review.csv', index=False)

: 전처리된 데이터프레임을 'filtering\_review.csv' 파일로 저장합니다. index=False는 인덱스 열을 파일에 포함하지 않도록 설정합니다.

2.8. 결측값 처리

- df.dropna(subset=['filtering\_text'], inplace=True)

: 전처리된 텍스트가 포함된 filtering\_text 컬럼에 결측값이 있는 행을 제거합니다.

**3. 모델 선택 및 학습 : 수업에서 활용한 모델을 활용해서 구현하고 그 강점과 약점을 이해**

**4. 평가 : 적절한 지표를 사용하여 모델의 성능을 평가**

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

4.1. 학습 세트와 테스트 세트 분할

- x와 y 데이터를 학습 세트(x\_train, y\_train)와 테스트 세트(x\_test, y\_test)로 분할합니다.

- test\_size=0.3은 전체 데이터의 30%를 테스트 세트로 사용한다는 의미입니다.

- random\_state=42는 결과 재현성을 위해 랜덤 시드를 설정합니다.

- stratify=y는 데이터의 클래스 비율을 학습 세트와 테스트 세트에서 동일하게 유지합니다.

4.2. 파이프라인 설정

- Pipeline을 사용하여 여러 단계를 연속적으로 처리할 수 있습니다.

- 첫 번째 단계는 텍스트 데이터를 TF-IDF 벡터로 변환하는 TfidfVectorizer입니다.

- 두 번째 단계는 SVM 분류기(SVC)입니다.

4.3. 하이퍼파라미터 튜닝

- param\_grid는 하이퍼파라미터 튜닝을 위해 탐색할 값들을 정의합니다.

- tfidf\_\_max\_df: TF-IDF에서 문서 빈도 임계값을 설정합니다. 0.8, 0.9, 1.0을 탐색합니다.

- tfidf\_\_ngram\_range: n-gram 범위를 설정합니다. (1, 1)과 (1, 2)를 탐색합니다.

- svc\_\_C: SVM의 정규화 파라미터입니다. 0.1, 1, 10을 탐색합니다.

- svc\_\_kernel: SVM의 커널 유형입니다. 'linear'와 'rbf'를 탐색합니다.

4.4. 그리드서치로 최적의 하이퍼파라미터 탐색:

- GridSearchCV를 사용하여 하이퍼파라미터의 모든 조합을 탐색하고 교차 검증(cv=5)을 통해 최적의 조합을 찾습니다.

- n\_jobs=-1은 모든 CPU 코어를 사용하여 병렬 처리를 수행합니다.

- scoring='accuracy'는 모델 성능 평가 기준으로 정확도를 사용합니다.

- fit(x\_train, y\_train)은 학습 세트를 사용하여 모델을 학습시킵니다.

4.5. 모델 평가

**텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명**

**텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명**

- precision: 각 클래스에 대한 정밀도. 예를 들어, 'positive' 클래스는 0.93으로, 모델이 양성으로 예측한 것 중 93%가 실제로 양성이었음을 의미합니다.

- recall: 각 클래스에 대한 재현율. 'positive' 클래스는 1.00으로, 실제 양성 중 100%를 모델이 양성으로 정확하게 예측했음을 의미합니다.

- f1-score: 정밀도와 재현율의 조화 평균값입니다. 'positive' 클래스의 f1-score는 0.96으로, 정밀도와 재현율 사이의 균형이 잘 맞는 것을 보여줍니다.

- support: 각 클래스의 샘플 수입니다.

**5. 시각화 및 해석 : 결과를 시각화하고 해석**

5.1 결과 시각화

텍스트, 스크린샷, 도표, 디자인이(가) 표시된 사진

자동 생성된 설명텍스트, 라인, 그래프, 스크린샷이(가) 표시된 사진

자동 생성된 설명

- Confusion Matrix 해석

: 혼동 행렬은 2x2 행렬로, 행은 실제 클래스를, 열은 예측된 클래스를 나타냅니다. 예를 들어, 왼쪽 위의 셀은 실제 'negative'인 샘플을 모델이 'negative'로 정확하게 예측한 수를 나타냅니다. 오른쪽 아래의 셀은 실제 'positive'인 샘플을 모델이 'positive'로 정확하게 예측한 수를 나타냅니다.

- ROC 곡선

: 그래프는 두 개의 클래스('positive'와 'negative')에 대한 ROC 곡선을 보여줍니다. 각 곡선은 각각의 클래스에 대해 진짜 양성 비율(TPR, True Positive Rate)에 대한 거짓 양성 비율(FPR, False Positive Rate)의 변화를 나타냅니다.

- 각 클래스의 ROC 곡선

: 'positive' 클래스와 'negative' 클래스 각각에 대한 ROC 곡선이 나타납니다. 각 곡선은 해당 클래스를 양성으로 간주할 때의 성능을 나타냅니다.

- 점선

: 기준 선으로, 이 선 위에 있는 점들은 랜덤한 예측의 성능을 나타냅니다. 좋은 모델은 이 점선에서 최대한 멀리 떨어져 있는 곡선을 가질 것입니다.