Chapter 08. 텍스트 분석

NLP: 머신이 인간의 언어를 이해하고 해석하는 데 더 중점을 둠

- 텍스트 분석을 향상하게 하는 기반 기술

텍스트 분석(텍스트 마이닝): 비정형 텍스트에서 의미 있는 정보를 추출하는 것에 더 중점을 둠

- 텍스트 분류, 감성 분석, 텍스트 요약, 텍스트 군집화에 집중

01. 텍스트 분석 이해

텍스트 분석: 비정형 데이터인 텍스트를 분석

- 피처 벡터화(피처 추출): 텍스트를 word 기반의 다수의 피처로 추출하고 이 피처에 단어 빈도 수와 같은 숫자값을 부여하여 텍스트를 벡터값으로 표현
    - BOW, Word2Vec
- 텍스트 분석 수행 프로세스
    1. 텍스트 사전 준비 작업(텍스트 전처리): 텍스트 정규화 작업 수행
    2. 피처 벡터화/추출: 사전 준비 작업으로 가공된 텍스트에서 피처를 추출하고 벡터값 할당
    3. ML 모델 수립 및 학습/예측/평가: 피처 벡터화된 데이터 세트에 ML 모델 적용, 학습/예측 및 평가 수행
- 파이썬 기반의 NLP, 텍스트 분석 패키지
    - NLTK: 방대한 데이터 세트와 서브 모듈 가짐, NLP의 거의 모든 영역 커버
    - Gensim: 토필 모델링 분야에서 두각을 보임
    - SpaCy: 뛰어난 수행 성능으로 최근 가장 주목을 받음

02. 텍스트 사전 준비 작업(텍스트 전처리) - 텍스트 정규화

텍스트 정규화: 텍스트를 머신러닝 알고리즘이나 NLP 애플리케이션에 입력 데이터로 사용하기 위해 클렌징, 정제, 토큰화, 어근화 등 다양한 텍스트 데이터의 사전 작업을 수행하는 것

- 클렌징: 분석에 방해가 되는 불필요한 문자, 기호 등을 사전에 제거하는 작업
- 텍스트 토큰화
    - 문장 토큰화: 문서에서 문장을 분리

        - 문장의 마침표, 개행문자 등 문장의 마지막을 뜻하는 기호에 따라 분리하는 것이 일반적, 정규 표현식에 따라서도 가능
        - 각 문장이 가지는 시맨틱적인 의미가 중요한 요소로 사용될 때 사용
    - 단어 토큰화: 문장에서 단어를 토큰으로 분리
        - 공백, 콤마, 마침표, 개행문자 등으로 분리하는 것이 일반적, 정규 표현식을 이용하기도 함
        - 단어의 순서가 중요하지 않은 경우 단어 토큰화만 사용해도 충분
    - n-gram: 연속된 n개의 단어를 하나의 토큰화 단위로 분리해 내는 것
- 스톱 워드 제거
    - 스톱 워드: 분석에 큰 의미가 없는 단어
- Stemming과 Lemmatization
    - 원형 단어를 찾는다는 목적을 가짐
    - Stemming: 원형 단어로 변환 시 일반적인 방법을 적용하거나 더 단순화된 방법을 적용해 원래 단어에서 일부 철자가 훼손된 어근 단어를 추출하는 경향
        - Porter, Lancaster, Snowball Stemmer
    - Lemmatization: 품사와 같은 문법적인 요소와 더 의미적인 부분을 감안 정확한 철자로 된 어근 단어를 찾아줌
        - WordNetLemmatizer

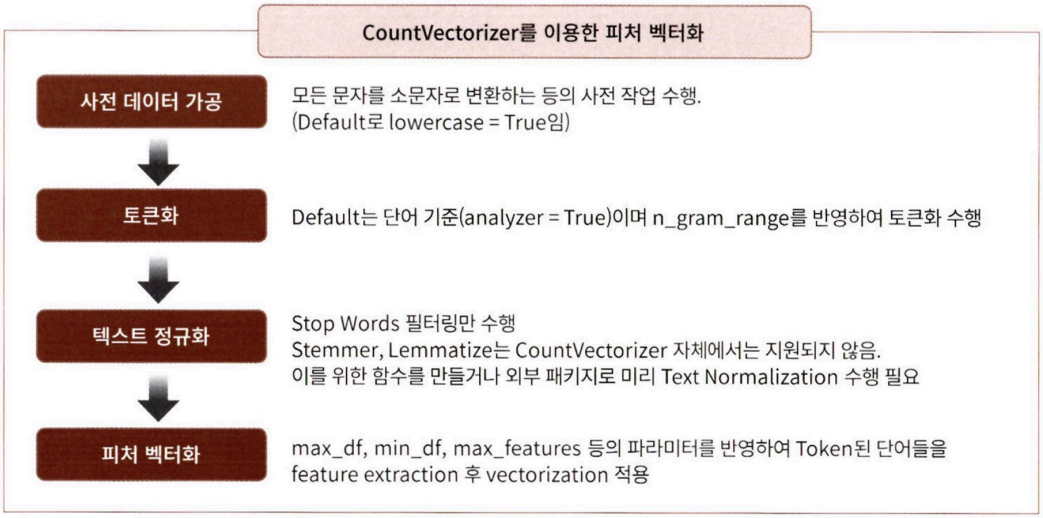
03. Bag of Words - BOW

Bag of Words: 문서가 가지는 모든 단어를 문맥이나 순서를 무시하고 일괄적으로 단어에 대해 빈도 값을 부여해 피처 값을 추출하는 모델

- BOW 장단점
    - 장점: 쉽고 빠른 구축
    - 단점: 문맥 의미 반영 부족, 희소 행렬 문제
- BOW 피처 벡터화: 모든 문서에서 모든 단어를 칼럼 형태로 나열하고 각 문서에서 해당 단어의 횟수나 정규화된 빈도를 값으로 부여하는 데이터 세트 모델로 변경하는 것

〈 BOW 피처 벡터화 〉

- BOW 피처 벡터화 방식
  - 카운트 기반의 벡터화: 해당 단어가 나타나는 횟수, count를 부여하는 경우
    - CountVectorizer: 피처 벡터화 + 소문자 일괄 변환, 토큰화, 스톱워드 필터링
      - 입력 파라미터: max_df, min_df, max_features, stop_words 등
  - TF-IDF: 개별 문서에서 자주 나타나는 단어에 높은 가중치를 주되, 모든 문서에서 전반적으로 자주 나타나는 단어에 대해서는 페널티 부여
    - TfidfVectorizer: 파라미터와 변환 방법이 CountVectorizer과 동일



- BOW 벡터화를 위한 희소 행렬
  - 희소 행렬: 대규모 행렬의 대부분의 값을 0이 차지하는 행렬
    - 메모리 공간이 많이 필요, 데이터 액세스를 위한 시간이 많이 소모
  - COO 방식: 0이 아닌 데이터만 별도의 데이터 배열에 저장하고, 그 데이터가 가리키는 행과 열의 위치를 별도의 배열로 저장

    - CSR 방식: 행 위치 배열 내에 있는 고유한 값의 시작 위치만 다시 별도의 위치 배열로 가지는 변환 방식
        - COO 형식이 행과 열의 위치를 나타내기 위해서 반복적인 위치 데이터를 사용해야 하는 문제점을 해결

05. 감성 분석

감성 분석: 문서의 주관적인 감성/의견/감정/기분 등을 파악하기 위한 방법

- 문서 내 텍스트가 나타내는 여러 가지 주관적인 단어와 문맥을 기반으로 감성 수치를 계산
- 긍정 감성/부정 감성: 긍정 감성 지수와 부정 감성 지수를 합산하여 결정
- 지도학습: 학습 데이터와 타깃 레이블 값을 기반으로 감성 분석 학습을 수행한 뒤 이를 기반으로 다른 데이터의 감성 분석을 예측하는 방법
- 비지도학습: 'Lexicon'이라는 일종의 감성 어휘 사전을 이용

Lexicon: 감성만을 분석하기 위해 지원하는 감성 어휘 사전 (감성 사전)

- 감성 지수: 긍정 감성 또는 부정 감성의 정도를 의미하는 수치
- 감성 사전: SentiWordNet, VADER, Patter

WordNet: 다양한 상황에서 같은 어휘라도 다르게 사용되는 어휘의 시맨틱 정보를 제공하며 이를 위해 각각의 품사로 구성된 개별 단어를 Synset으로 표현

SentiSynset: 단어의 감성을 나타내는 감성 지수와 객관성을 나타내는 객관성 지수를 가짐