[파머완 8.1~8.3, 8.5]

1. 텍스트 분석 이해

   피처 벡터화(피처 추출): 텍스트를 word 기반의 다수의 피처로 추출하고 이 피처에 단어 빈도수와 같은 숫자 값을 부여하면 텍스트는 단어의 조합인 벡터값으로 표현될 수 있는데, 이렇게 텍스트를 변환하는 것

   대표적 2가지 방법: BOW, Word2Vec

   1) 텍스트 분석 수행 프로세스
      (1) 텍스트 사전 준비작업(텍스트 전처리): 텍스트를 피처로 만들기 전에 미리 클렌징, 대/소문자 변경, 특수문자 삭제 등의 클렌징 작업, 단어(Word) 등의 토큰화 작업, 의미 없는 단어(Stop word) 제거 작업, 어근 추출(Stemming/Lemmatization) 등의 텍스트 정규화 작업을 수행하는 것을 통칭
      (2) 피처 벡터화/추출: 사전 준비 작업으로 가공된 텍스트에서 피처를 추출하고 여기에 벡터 값 할당, 대표적 방법 BOW, Word2Vec, BOW는 대표적으로 Count 기반과 TF-IDF 기반 벡터화가 있음
      (3) ML 모델 수립 및 학습/예측/평가: 피처 벡터화된 데어터 세트에 ML 모델을 적용해 학습/예측 및 평가를 수행

   2) 파이썬 기반의 NLP, 텍스트 분석 패키지

NLTK	가장 대표적인 NLP 패키지 방대한 데이터와 서브 모듈 가짐 수행 속도가 느리다는 단점
Gensim	토픽 모델링 분야에서 가장 두각을 나타내는 패키지 Word2Vec 구현 등의 다양한 신기능도 제공
SpaCy	최근 가장 많은 주목 받음

2. 텍스트 사전 준비 작업(텍스트 전처리) – 텍스트 정규화

   텍스트 정규화: 입력 데이터로 사용하기 위해 클렌징, 정제, 토큰화, 어근화 등의 다양한 텍스트 데이터의 사전 작업을 수행하는 것

   1) 클렌징: 불필요한 문자, 기호 등을 사전에 제거하는 작업 (HTML, XML, 태그, 특정 ㅇ

기호 등)

2) 텍스트 토큰화

(1) 문장 토큰화

문장의 마침표, 개행문자(₩n) 등의 기호 따라 분리함

(2) 단어 토큰화

문장을 단어로 토큰화 하는 것

기본적으로 공백, 콤마, 마침표, 개행문자 등으로 분리하지만 정규 표현식 사용시 더 다양한 유형으로 수행 가능

3) 스톱 워드 제거

스톱 워드: 분석에 큰 의미가 없는 단어를 지칭

4) Stemming, Lemmatization

문법적 또는 의미적으로 변화하는 단어의 원형을 찾는 것

3. Bag of Words – BOW

문서가 가지는 모든 단어를 문맥이나 순서를 무시하고 일괄적으로 단어에 대해 빈도 값을 부여하여 피처 값을 추출하는 모델

1) 문장 1과 문장 2에 있는 모든 단어에서 중복을 제거하고 각 단어를 칼럼 형태로 나열합니다. 그러고 난 후 각 단어에 고유의 인덱스를 다음과 같이 부여합니다.

→ 개별 문장에서 해당 단어가 나타나는 횟수를 각 단어에 기재

2) 장점: 쉽고 빠른 구축

3) 단점: 문맥의미 반영 부족, 희소 행렬 문제

4) BOW 피처 벡터화 (가중치가 높을수록 중요한 단어로 인식)

(1) 카운트 기반의 벡터화

(2) TF-IDF 기반의 벡터화

5) 사이킷런의 Count 및 TF-IDF 벡터화 구현: CountVectorizer, TfidVectorizer

파라미터 명	파라미터 설명
max_df	전체 문서에 걸쳐서 너무 높은 빈도수를 가지는 단어 피처를 제외하기 위한 파라미터입니다. 너무 높은 빈도수를 가지는 단어는 스톱 워드와 비슷한 문법적인 특성으로 반복적인 단어일 가능성이 높기에 이를 제거하기 위해 사용됩니다. max_df = 100과 같이 정수 값을 가지면 전체 문서에 걸쳐 100개 이하로 나타나는 단어만 피처로 추출합니다. Max_df = 0.95와 같이 부동소수점 값(0.0 ~ 1.0)을 가지면 전체 문서에 걸쳐 빈도수 0~95%까지의 단어만 피처로 추출하고 나머지 상위 5%는 피처로 추출하지 않습니다.
min_df	전체 문서에 걸쳐서 너무 낮은 빈도수를 가지는 단어 피처를 제외하기 위한 파라미터입니다. 수백~수천 개의 전체 문서에서 특정 단어가 min_df에 설정된 값보다 적은 빈도수를 가진다면 이 단어는 크게 중요하지 않거나 가비지(garbage)성 단어일 확률이 높습니다. min_df = 2와 같이 정수 값을 가지면 전체 문서에 걸쳐서 2번 이하로 나타나는 단어는 피처로 추출하지 않습니다. min_df = 0.02와 같이 부동소수점 값(0.0 ~ 1.0)을 가지면 전체 문서에 걸쳐서 하위 2% 이하의 빈도수를 가지는 단어는 피처로 추출하지 않습니다.
max_features	추출하는 피처의 개수를 제한하며 정수로 값을 지정합니다. 가령 max_features = 2000으로 지정할 경우 가장 높은 빈도를 가지는 단어 순으로 정렬해 2000개까지만 피처로 추출합니다.
stop_words	'english'로 지정하면 영어의 스톱 워드로 지정된 단어는 추출에서 제외합니다.
n_gram_range	Bag of Words 모델의 단어 순서를 어느 정도 보강하기 위한 n_gram 범위를 설정합니다. 튜플 형태로 (범위 최솟값, 범위 최댓값)을 지정합니다. 예를 들어 (1, 1)로 지정하면 토큰화된 단어를 1개씩 피처로 추출합니다. (1, 2)로 지정하면 토큰화된 단어를 1개씩(minimum 1), 그리고 순서대로 2개씩(maximum 2) 묶어서 피처로 추출합니다.
analyzer	피처 추출을 수행한 단위를 지정합니다. 당연히 디폴트는 'word'입니다. Word가 아니라 character의 특정 범위를 피처로 만드는 특정한 경우 등을 적용할 때 사용됩니다.
token_pattern	토큰화를 수행하는 정규 표현식 패턴을 지정합니다. 디폴트 값은 '\b\w\w+\b로, 공백 또는 개행 문자 등으로 구분된 단어 분리자(\b) 사이의 2문자(문자 또는 숫자, 즉 영숫자) 이상의 단어(word)를 토큰으로 분리합니다. analyzer= 'word'로 설정했을 때만 변경 가능하나 디폴트 값을 변경할 경우는 거의 발생하지 않습니다.
tokenizer	토큰화를 별도의 커스텀 함수로 이용시 적용합니다. 일반적으로 CountTokenizer 클래스에서 어근 변환 시 이를 수행하는 별도의 함수를 tokenizer 파라미터에 적용하면 됩니다.

6) BOW 벡터화를 위한 희소 행렬

사이킷런의 CountVectorizer/TfidfVectorizer 를 이용해 텍스트를 피처 단위로 벡터화해 변환하고 CSR 형태의 희소 행렬을 반환

희소 행렬: 대규모 행렬의 대부분의 값을 0이 차지하는 행렬
BOW 형태를 가진 언어 모델의 피처 벡터화는 대부분 희소 행렬
대표적 방식 COO, CSR

(1) 희소 행렬 – COO 형식

0이 아닌 데이터만 별도의 데이터 배열에 저장하고, 그 데이터가 가리키는 행과 열의 위치를 별도의 배열로 저장하는 방식

(2) 희소 행렬 – CSR 형식

COO 형식이 행과 열의 위치를 나타내기 위해서 반복ㅈㄱ인 위치 데이터를 사용해야 하는 문제점을 해결한 방식

4. 감성 분석

1) 지도학습 기반 감성 분석 실습 – IMDB 영화평

2) 비지도학습 기반 감성 분석 소개

Lexicon을 기반으로 함

감성 지수를 이용하여 분석

- SentiWordNet: NLTK 패키지의 WordNet과 유사하게 감성 단어 전용의 WordNet을 구현한 것입니다. WordNet의 Synset 개념을 감성 분석에 적용한 것입니다. WordNet의 Synset별로 3가지 감성 점수(sentiment score)를 할당합니다. 긍정 감성 지수, 부정 감성 지수, 객관성 지수가 그것입니다. 긍정 감성 지수는 해당 단어가 감성적으로 얼마나 긍정적인가를, 부정 지수는 얼마나 감성적으로 부정적인가를 수치로 나타낸 것입니다. 객관성 지수는 긍정/부정 감성 지수와 완전히 반대되는 개념으로 단어가 감성과 관계없이 얼마나 객관적인지를 수치로 나타낸 것입니다. 문장별로 단어들의 긍정 감성 지수와 부정 감성 지수를 합산하여 최종 감성 지수를 계산하고 이에 기반해 감성이 긍정인지 부정인지를 결정합니다.
- VADER: 주로 소셜 미디어의 텍스트에 대한 감성 분석을 제공하기 위한 패키지입니다. 뛰어난 감성 분석 결과를 제공하며, 비교적 빠른 수행 시간을 보장해 대용량 텍스트 데이터에 잘 사용되는 패키지입니다.
- Pattern: 예측 성능 측면에서 가장 주목받는 패키지입니다. 아쉽게도 현재 기준으로 파이썬 3.X 버전에서 호환이 되지 않고, 파이썬 2.X 버전에서만 동작합니다. 이 책에서는 사용 예제를 소개하지는 않습니다만, 감성 분석에 관심이 많은 사람이라면 적용해 보는 것도 좋습니다

3) SentiWordNet을 이용한 감성 분석