Week13_텍스트분석_8장_#2

토픽 모델링(Topic Modeling)

문서 집합에 숨어 있는 주제를 찾아내는 것

— 머신러닝 기반 토픽 모델: 숨겨진 주제를 효과적으로 표현할 수 있는 **중심 단어**를 함축적으로 추출

자주 사용되는 기법

- LSA(Latent Semantic Analysis)
- LDA(Latent Dirichlet Allocation)

LDA로 토픽 모델링 수행하기

(복습과제) 뉴스그룹 데이터로 사이킷런의 LatentDirichletAllocation 클래스를 이용해서 토픽 모델링

문서 군집화(Document Clustering)

비슷한 텍스트 구성의 문서를 군집화(Clustering)하는 것

텍스트 분류 기반 문서 분류와 비교

동일한 군집에 속하는 문서를 같은 카테고리 소속으로 분류할 수 있음
→ 텍스트 분류 기반 문서 분류와 유사

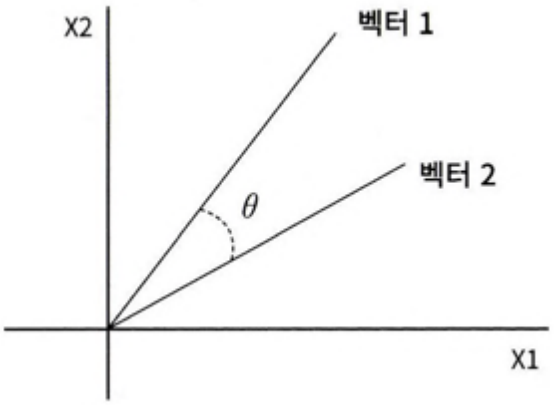
- 텍스트 분류 기반 문서 분류: 사전에 결정 카테고리 값을 가진 학습 데이터세트 필요
- 문서 군집화: 비지도학습 기반 동작

문서 유사도

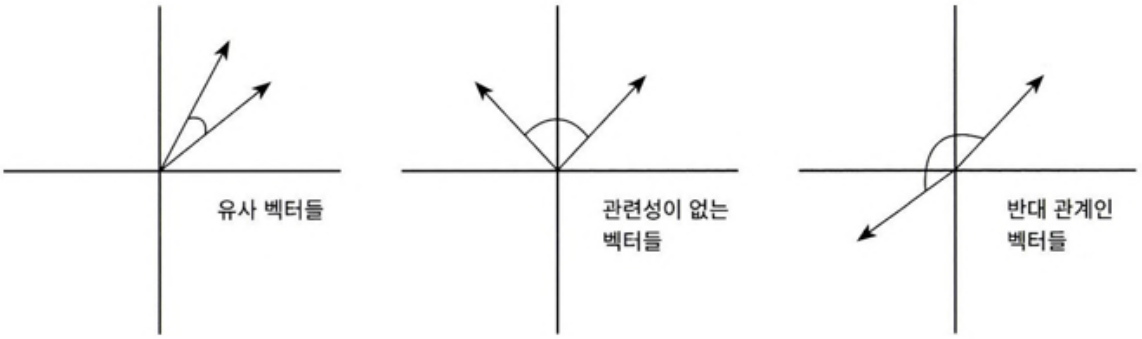
문서와 문서 간 유사도를 비교하기

문서 유사도 측정 방법 - 코사인 유사도(Cosine Similarity)

벡터와 벡터 간 유사도를 비교할 때 두 벡터 사이의 사잇각을 구해서 벡터의 상호 방향성이 얼마나 유사한지에 기반



두 벡터 사잇각



- 두 벡터 A와 B의 코사인 값 - A와 B의 내적 값을 이용

$$\text{similarity} = cos\theta = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

코사인 유사도를 문서 유사도 비교에 많이 사용하는 이유

문서를 피처 벡터화 변환 → 희소 행렬이 되기 쉬운데, 희소 행렬 기반에서 문서와 문서 벡터 간의 크기에 기반한 유사도(ex.유클리드 거리 기반 지표)는 정확도가 떨어지기 쉬움

문서가 매우 긴 경우 글이 길기 때문에 단어 빈도수도 더 ⬆️ → 공정한 비교 X

한글 텍스트 처리 - 네이버 영화 평점 감성 분석

한글 NLP 처리의 어려움

한글에는 띄어쓰기랑 다양한 조사가 있어서 라틴어 처리보다 어려움

KoNLPy

파이썬의 대표적인 한글 형태소 패키지

- **형태소 분석(Morphological analysis)**: 말뭉치를 형태소 어근 단위로 쪼개고 각 형태소에 품사 태깅을 부착하는 작업
- KoNLPy에서 꼬꼬마(Kkma), 한나눔(Hannanum), Komoran, 은전한닢 프로젝트(Mecab), Twitter 등 5개의 형태소 분석 모듈 모두 사용 가능
    - Mecab은 윈도우에서 안 됨