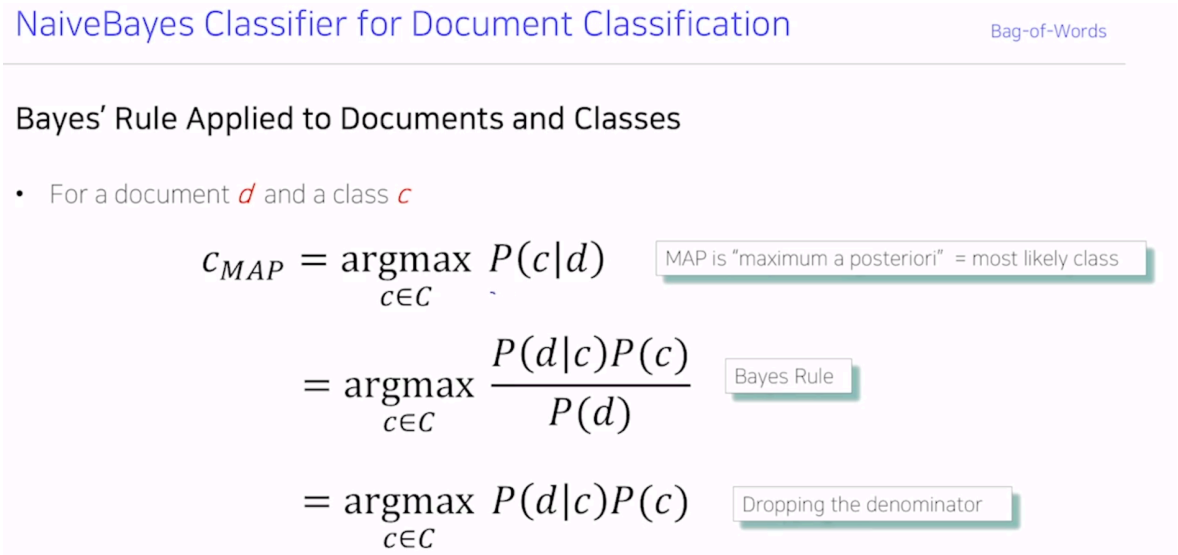
Week4-1

1. 학습정리
    1. 자연어 학문 분야
        1. NLP
            1. 단계별 프로세싱
                1. 로우레벨
                2. 단어레벨, 구레벨
                3. 문장레벨
                4. 문단레벨
        2. 텍스트 마이닝
            1. 트렌드 분석
            2. 사회과학 적인 인사이트 분석
        3. 검색 기술
            1. 추천 시스템
    2. NLP
        1. 트랜스포머 모델 사용
    3. Bag-of-Words
        1. 단어를 및 문서를 벡터로 표현
        2. 단계
            1. 각 단어를 카테고리화 함
            2. 원 핫 벡터로 변환
            3. 각 단어의 원핫 벡터를 모두 더해서 문장을 나타낼 수 있음 - bag-of-words 벡터
    4. NaiveBayes classifier



NaiveBayes Classifier for Document Classification

Bag-of-Words

- Bayes' Rule Applied to Documents and Classes
  - For a document d, which consists of a sequence of words w, and a class c
  - The probability of a document can be represented by multiplying the probability of each word appearing
  - $P(d|c)P(c) = P(w_1, w_2, \ldots, w_n|c)P(c) \rightarrow P(c)\prod_{w_i \in W} P(w_i|c)$ (by conditional independence assumption)

1. 백오브워즈 벡터를 분류할 수 있는 대표적 방법
2. 클래스가 c개 주어졌을때 문서 d 가 어느 클래스에 속할지
3. 가장 높은 확률을 보이는 클래스 c를 선택
4. 각 클래스가 나타날 확률과 클래스가 고정되어 있을 때 각 단어가 나타날 확률을 추정
5. 나이브 = 멍청
    1. 모든 단어가 독립적이라고 가정하고 베이즈 정리를 적용한 것
6. 예제

or a test document d_5 = "Classification task uses transformer"

We calculate the conditional probability of the document for each class

We can choose a class that has the highest probability for the document

$$P(c_{CV}|d_5) = P(c_{CV})\prod_{w \in W} P(w|c_{CV}) = \frac{1}{2} \times \frac{1}{14} \times \frac{1}{14} \times \frac{1}{14} \times \frac{1}{14}$$

$$P(c_{NLP}|d_5) = P(c_{NLP})\prod_{w \in W} P(w|c_{NLP}) = \frac{1}{2} \times \frac{1}{10} \times \frac{2}{10} \times \frac{1}{10} \times \frac{1}{10}$$

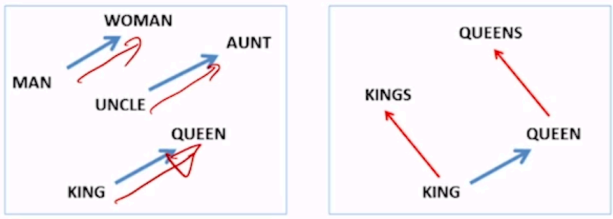
Word	Prob	Word	Prob
$P(w_{\text{"classification"}}\|c_{CV})$	$\frac{1}{14}$	$P(w_{\text{"classification"}}\|c_{NLP})$	$\frac{1}{10}$
$P(w_{\text{"task"}}\|c_{CV})$	$\frac{1}{14}$	$P(w_{\text{"task"}}\|c_{NLP})$	$\frac{2}{10}$
$P(w_{\text{"uses"}}\|c_{CV})$	$\frac{1}{14}$	$P(w_{\text{"uses"}}\|c_{NLP})$	$\frac{1}{10}$

5. word embedding
    1. 각 단어들을 특정한 차원의 공간상의 한 점(벡터)로 나타냄
    2. 비슷한 의미의 단어가 좌표공간상의 비슷한 위치로 매핑
    3. 의미상의 유사도를 반영한 벡터 표현
6. Word2Vec

Property of Word2Vec

Word Embedding: Word2Ve

- The word vector, or the relationship between vector points in space, represents the relationship between the words.
- The same relationship is represented as the same vectors.



(Mikolov et al., NAACL HLT, 2013)

- e.g.,
- vec[queen] – vec[king] = vec[woman] – vec[man]

1. 워드임베딩을 학습하는 알고리즘
2. 주변 단어들로 부터 의미를 유추할 수 있다는 것에 착안
3. 주변단어를 숨긴채 의미를 예측하는 방식으로 학습
4. 단어끼리의 관계에 따라 일관된 벡터 속성 보여줌
5. 이걸로 할 수 있는 일
    1. word intrusio 어졌을 때 나머지 단어와 가장 상이한 단어를 찾아내는 것

7. GloVe
    1. 워드투벡과 더불어 많이 쓰이는 임베딩 방법
    2. 워드투벡과 동일한 기능, 비등비등한 성능
    3. 워드투벡과 차이점
        1. 각 입력 및 출력 단어쌍들에 대해 학습데이터에서 한 윈도우 내에서 두 단어가 총 몇 번 동시에 등장했는지 사전에 계산
        2. 입력벡터와 출력벡터의 내적값이 위에서 구한 것에 최대한 가까워 질 수 있도록 학습
        3. 중복되는 계산을 줄여줌
        4. 학습이 빠름
        5. 적은 데이터에 대해서도 잘 동작
8. 실습
    1. NaiveBayes classifier
        1. 스무딩 (k)
            1. 테스트에는 특정 클래스에 단어가 있는데 트레이닝에는 없었을 때 유의미하게 학습하게 하기 위함
    2. Word2Vec
        1. 두 가지 구현법
            1. CBOW (Continueous Bag of words)
                1. 주변단어를 통해 중심단어를 예측
            2. Skip-gram
                1. 중심단어를 통해 주변 단어를 예측

2. 피어세션
    1. 앞으로의 계획 토의
        1. 퍼더퀘스쳔 후 복습과 인터뷰

    2. 조교님 참여 시간엔 인터뷰
2. 퍼더퀘스쳔
    1. 워드2벡 단점
    2. GloVe
    3. 동시에 가지는 단점
        1. 학습하는 데이터의 양이 충분치 않으면 단어간의 관계를 부정확하게 학습할 수 있음
        2. 사용자가 지정한 윈도우 내에서만 학습/분석이 이루어져서 단어가 전체의 문맥적 정보를 반영하기 힘듦
        3. 학습이 안 된 단어에 대해서는 벡터값을 만들어 낼 수 없음
            1. 오타난 단어의 원형 예측 불가
                1. fasttext
                    1. 언어의 형태학적 요소를 반영한 모델
                    2. 스펠링 비슷하면 비슷한 벡터공간상에 위치 -> 오타
3. 과제
    1. 간단한 텍스트 전처리 실습을 하면서 익힐 수 있었음