Natural language processing Bible

9

서수원

Business Intelligence Lab.

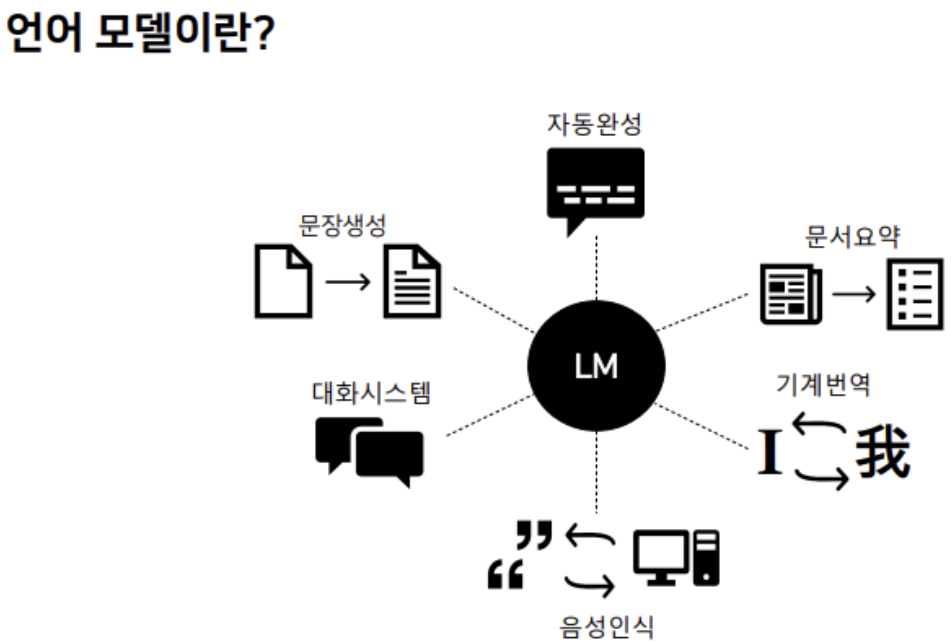
산업경영공학과, 명지대학교

01

개체명 인식

언어 모델

- 언어 모델이란?
  - 언어를 이루는 구성요소(글자,형태소,단어 등)을 문맥으로 하여 이를 바탕으로 다음 구성요소를 예측하는 모델을 의미한다.
  - 통계적 언어모델과 딥러닝 언어모델로 구분 할수 있다.



통계적 언어모델

- 통계적 언어 모델
  - 주어진 문서내 단어열의 등장확률을 기반으로 각 단어의 조합을 예측하는 모델이다.
  - 실제로 많이 사용하는 단어열의 확률 분포를 정확하게 근사하는 것이 목표이다.

조건부 확률 P(B|A): 사건 A가 일어났을 때 사건 B가 일어날 확률

언어 모델

단어 A가 등장했을 때 바로 다음에 단어 B가 등장할 확률

통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - 결합확률
  - 연쇄 법칙

연쇄 법칙(Chain rule)

$$P(A, B, C, D) = P(A)P(B|A)P(C|A, B)P(D|A, B, C)$$

↓ 언어 모델

$$P(\text{나는}, \text{사과를}, \text{먹는다}) = P(\text{나는})P(\text{사과를}|\text{나는})P(\text{먹는다}|\text{나는}, \text{사과를})$$

연쇄 법칙(Chain rule)

$$P(A, B, C, D) = P(A)P(B|A)P(C|A, B)P(D|A, B, C)$$

↓ n개 단어(w)의 결합 확률

$$P(w_1, w_2, w_3, \cdots, w_n) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)\cdots P(w_n|w_1, \cdots, w_{n-1})$$

$$= \prod_{n=1}^{n} P(w_n|w_1, \cdots, w_{n-1})$$

  - 카운트 기반 계산
    - ✓ 코퍼스 내에서 각 단어들의 조합이 나오는 횟수를 카운트 한 후 이에 기반하여 확률을 계산한다.
      - ❖ 모든 단어 조합의 경우의 수를 다 세어야 한다.
      - ❖ 계산 복잡도가 높아질 뿐만 아니라 무한한 크기의 코퍼스가 필요하다
      - ❖ 어렵다.

$$P(\text{먹는다}|\text{나는}, \text{사과를}) = \frac{count(\text{나는}, \text{사과를}, \text{먹는다})}{count(\text{나는}, \text{사과를})}$$

통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - 마르코프 가정
    - ✓ 기존 연쇄 법칙의 복잡성을 해결하고 간소화 하기 위함이다.
    - ✓ 미래 사건에 대한 조건부가 과거에 대해서는 독립이며 현재의 사건에만 영향을 받는다는 가정을 전제로 한다.

.

$$P(w_1, w_2, \cdots, w_n) \approx P(w_1)P(w_2|w_1)\cdots P(w_n|w_{n-1})$$

$$\downarrow$$

$$P(\text{나는},\ \text{사과를},\ \text{먹는다}) \approx P(\text{나는})P(\text{사과를}\,|\,\text{나는})P(\text{먹는다}\,|\,\text{사과를})$$

  - 마르코프 가정의 한계
    - ✓ 언어 현상에 적용하기에는 지나치게 단순화를 한다.
    - ✓ 언어의 장기 의존성이 간과된다. (The computer which I had just put into the machine room on the fifth floor crashed.)

통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram(언어모델)
    - ✓ 문장 내 단어는 주변의 여러 단어와 연관 된다고 가정한다.
    - ✓ N : 주변 몇 개의 단어를 볼 것인지 정하는 숫자이다.
    - ✓ N-gram : N개의 단어열을 의미한다.

    - ✓ **1-gram(unigram):** The, boy, is, looking, at, a, pretty, girl
    - ✓ **2-gram(bigram):** The boy, boy is, is looking, looking at, at a, a pretty, pretty girl
    - ✓ **3-gram(trigram):** The boy is, boy is looking, is looking at, looking at a, at a pretty, a pretty girl
    - ✓ **4-gram:** The boy is looking, boy is looking at, is looking at a, looking at a pretty, at a pretty girl

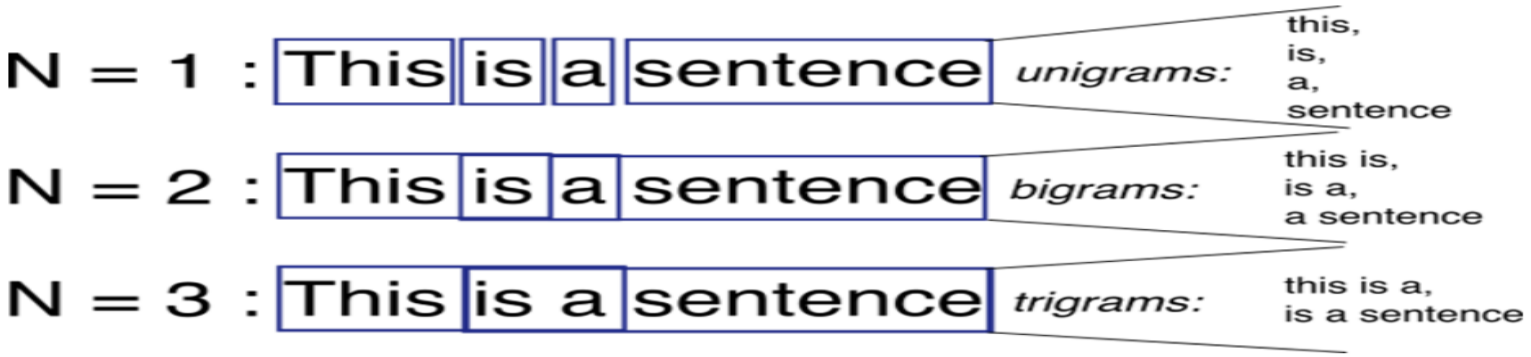
통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram(언어모델)
    - ✓ 문장 내 단어는 주변의 여러 단어와 연관 된다고 가정한다.
    - ✓ N : 주변 몇 개의 단어를 볼 것인지 정하는 숫자이다.
    - ✓ N-gram : N개의 단어열을 의미한다.
      - ❖ 1-gram (유니그램)

$$\text{1-gram(유니그램, unigram):}\ P(w_1, w_2, \cdots, w_n) \approx \prod_{i=1}^{n} P(w_i)$$

      - ❖ 2-gram(바이그램)

$$\text{2-gram(바이그램, bigram):}\ P(w_1, w_2, \cdots, w_n) \approx \prod_{i=1}^{n} P(w_i | w_{i-1})$$



통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram(언어모델)

코퍼스 예시

<s>I eat an apple</s>

<s>an apple I eat</s>

<s>I like cheese cake</s>

- 예제: 3개 문장과 2-gram 모델로 단어열의 등장 확률 계산

$$P(I|<s>) = \frac{count(<s>, I)}{count(<s>)} = \frac{2}{3} = 0.6667$$

$$P(an|<s>) = \frac{count(<s>, an)}{count(<s>)} = \frac{1}{3} = 0.3333$$

$$P(eat|I) = \frac{count(I, eat)}{count(I)} = \frac{2}{3} = 0.6667$$

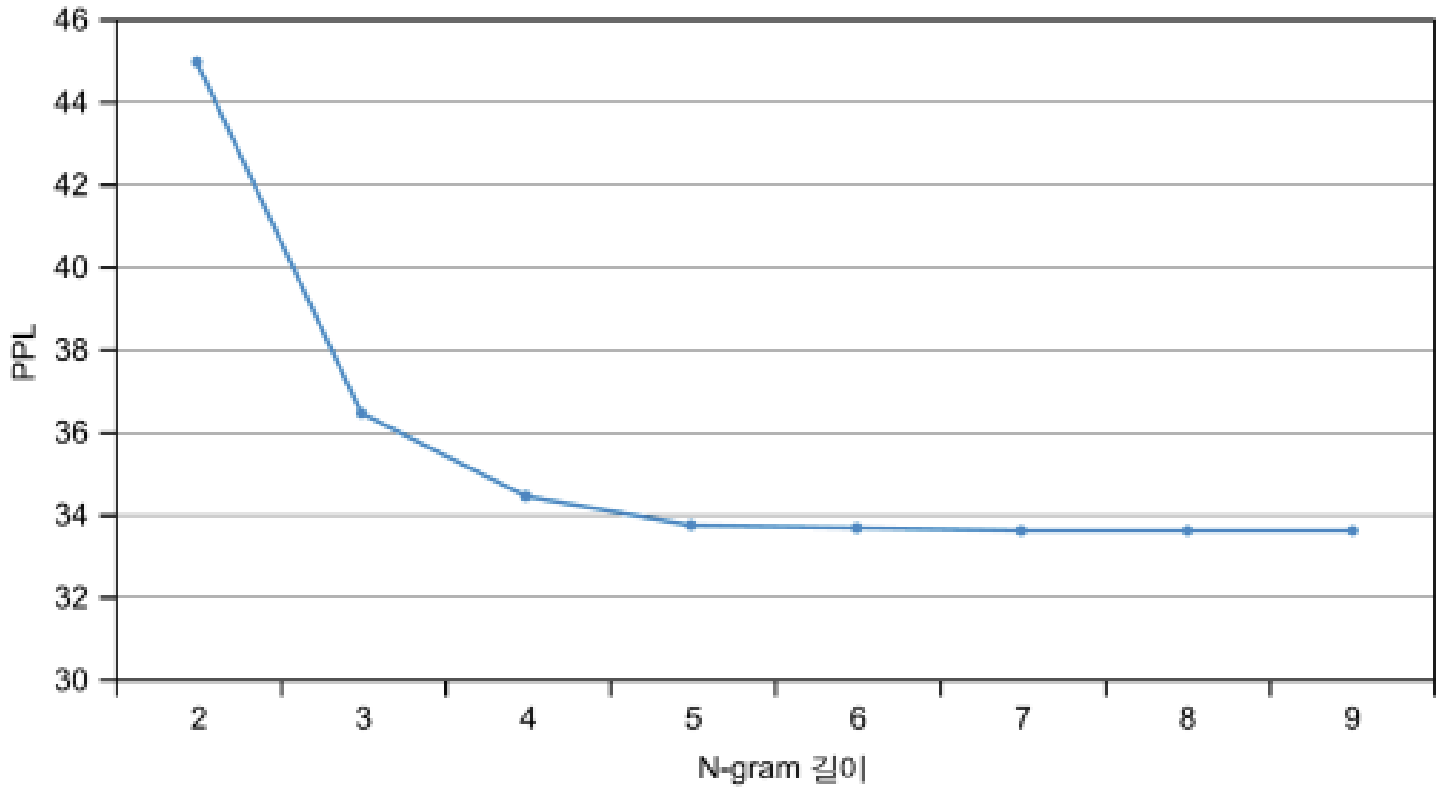
$$P(</s>|apple) = \frac{count(apple, </s>)}{count(apple)} = \frac{1}{2} = 0.5$$

$$P(like|I) = \frac{count(I, like)}{count(I)} = \frac{1}{3} = 0.3333$$

$$P(cake|cheese) = \frac{count(cheese, cake)}{count(cheese)} = \frac{1}{2} = 0.5$$

통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram(언어모델) 성능비교



통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram(언어모델)
    - ✓ 한계
      - ❖ 생성된 문장이 지니치게 부자연스럽거나 기존 코퍼스와 지나치게 유사하다.
      - ❖ 단어열의 확률값이 코퍼스에 따라 크게 달라진다.
      - ❖ 방대한 양의 코퍼스가 필요하다.
      - ❖ 희소성 문제가 생긴다.
      - ❖ 교착어인 한국어의 경우 희소성 문제가 크게 발생한다.(사과가, 사과를 사과도, 사과에 를 다 다른 단어로 처리)

통계적 언어모델

- 통계적 언어 모델(조건부 확률)
  - N-gram
    - ✓ 로그 확률
      - ❖ 언어 모델의 확률 계산시 원래 확률값에 로그를 취하는 것이 보편적이다.
      - ❖ 언더플로를 피하기 위함이다.
      - ❖ 계산이 간단해지며, 곱셉을 덧셈으로 환산 할 수 있다.

$$\log(p_1 \times p_2 \times p_3 \times p_4) = \log p_1 + \log p_2 + \log p_3 + \log p_4$$

일반화

- 일반화
  - 일반화
    - ✓ 통계적 언어 모델은 제한된 양의 코퍼스로 인해 이전에 보지 못한 단어열에 대해서는 제대로 예측하지 못하고 정확도가 떨어진다.
    - ✓ 이와 같은 희소성 문제를 해결하고 모델의 일반화 능력을 향상시키기 위해 다양한 기법이 제시 되었다.

일반화

- 일반화
  - 일반화(스무딩)
    - ✓ 모델이 한번도 본 적 없는 단어 조합에 특정 값을 부여하여 확률 분포에 변화를 주는 방법이다.
    - ✓ 코퍼스에 없는 단어열로 인해 전체 문장의 확률이 0이 되는 희소성 문제를 방지한다.

$$P(w_i | w_{<i}) \approx \frac{count(w_{<i}, w_i) + \alpha}{count(w_{<i}) + \alpha V}$$

$\mathbf{w < i}$

: i번째 단어 이전에 등장하는 모든 단어

$\boldsymbol{V}$

: 어휘(vocabulary) 사이즈 (코퍼스에 등장하는 단일 단어 개수)

일반화

- 스무딩
  - 라플라스 스무딩
    - ✓ 알파 값을 1로 지정하는 방법이다.
    - ✓ 한번도 등장하지 않은 단어열이 최소 한번은 등장 했다고 가정한다

코퍼스 예시

"I eat a strawberry"

"I eat a blueberry"

"I eat a strawberry cake"

기존 확률 계산식: $P(cake|blueberry)$ = 0, 전체 문장의 등장 확률 = 0

$$P(cake\,|\,blueberry) \approx \frac{count\,(blueberry,\,cake\,)+1}{count\,(blueberry\,)+\,V}$$

$$\approx \frac{0+1}{1+6} = \frac{1}{7} \fallingdotseq 0.143$$

라플라스 스무딩 적용시: $0 < P(cake|blueberry) \leq 1$, 0 < 전체 문장의 등장 확률 ≤ 1

일반화

- 스무딩
  - 라플라스 스무딩 한계
    - ✓ 제로 데어티가 적은 경우 유용하다.
    - ✓ 계산을 거듭할수록 원래 단어의 빈도수에서 크게 벗어난다.
    - ✓ 일반화 문제는 완전히 해소되지 않는다.

일반화

- 보간법
  - 특정 N-gram의 확률을 이전 N-gram의 확률과 섞는방법이다.
    - ✓ 3-gram의 예시 : 2-gram,1-gram 모델의 확률까지 구한 후 일정한 비율의 가중치를 곱한후 합하는 방식이다.

$$\hat{P}(w_n|w_{n-1},w_{n-2}) = \lambda_1 P(w_n|w_{n-1},w_{n-2}) + \lambda_2 P(w_n|w_{n-1}) + \lambda_3 P(w_n)$$

  - 라플라스 스무딩은 모든 제로 데이터에 똑같은 빈도수를 부여하기 때문에 문제가 발생한다.
  - 보간법 사용시, 제로 데이터들의 N-gram 정보에 따라 서로 다른 빈도를 부여할 수 있다.
  - 가중치는 검증 코퍼스에서 각 N-gram의 확률을 최대화 하는 0~1 사이의 값으로 설정한다.

일반화

- 백오프
  - 보간법과 유사하다. 여러 N-gram을 함께 고려한다.
  - 모든 N-gram의 확률을 합하지 않는다는 점이 보간법과 다르다.
    - ✓ 3-gram 모델의 예시 : 3-gram,2-gram,1-gram의 확률 중 빈도수가 0 이상이며 N의 차수가 높은 확률을 사용한다.

$$\hat{P}(w_i|w_{i-2},w_{i-1}) = \begin{cases} P(w_i|w_{i-2},w_{i-1}), \text{ if } count(w_{i-2},w_{i-1},w_i) > 0 \\ \alpha_1 P(w_i|w_{i-1}), \text{ if } count(w_{i-2},w_{i-1},w_i) = 0 \text{ and } count(w_{i-1},w_i) > 0 \\ \alpha_2 P(w_i), \ otherwise. \end{cases}$$

일반화

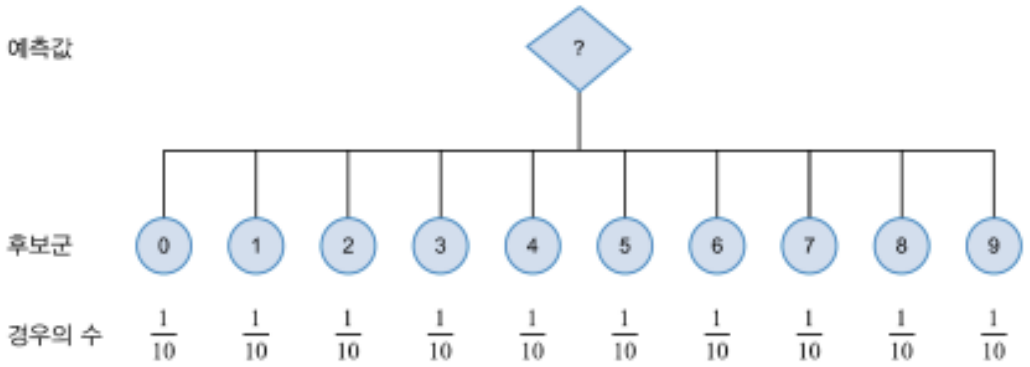
- 일반화에 대해 생각해 볼 문제
  - 비슷한 패턴의 새로운 문장에 대해 추론할 수 있는 일반화가 가능해야 한다.
    - ✓ 이는 딥러닝 언어 모델에서의 주요 과제 중 하나이다.

모델 평가와 퍼플렉서티

- 언어 모델의 평가
  - 일반적인 방법은 모델 간 비교 이지만, 상당한 시간이 소요된다.
  - 따라서 퍼플렉서티(Perplexity,PPL)을 활용한다.
    - ✓ 언어 모델의 성능을 자체적으로 평가하는 내부 평가 척도이다.
    - ✓ PPL은 확률 분포를 얼마나 확실하게 예측할 수 있는지를 나타내는 지표이다.
    - ✓ PPL점수가 낮을수록 좋다.

모델 평가와 퍼플렉서티

- PPL 계산
  - PPL은 모델이 선택할 수 있는 경우의 수를 의미하는 분기계수 이다.
  - 즉 모델이 얼마나 많은 후보군을 두고 고민 하는가를 나타낸다.
    - ✓ PPL이 높다는 것은 많은 후보군을 두고 고민 한다는 것을 의미하고, 이는 예측에 대한 확실성이 낮음을 의미한다.



✓ PPL = 10

✓ 30,000개 단어들 중 다음 단어로 올 확률이 가장 높은 단어를 예측하는 언어모델이라면 PPL = 30,000

✓ 후보군에 대한 확률의 역수를 후보군의 개수로 정규화(normalization)하여 계산

모델 평가와 퍼플렉서티

- PPL 계산
  - PPL은 모델이 선택할 수 있는 경우의 수를 의미하는 분기계수 이다.
  - 즉 모델이 얼마나 많은 후보군을 두고 고민 하는가를 나타낸다.
    - ✓ PPL이 높다는 것은 많은 후보군을 두고 고민 한다는 것을 의미하고, 이는 예측에 대한 확실성이 낮음을 의미한다.

$$PPL(W) = P(w_1, w_2, \ldots, w_N)^{-\frac{1}{N}}$$

$$= \sqrt[N]{\frac{1}{P(w_1, w_2, \ldots, w_N)}}$$

$$= \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_1 | w_1, w_2, \ldots, w_{i-1})}}$$