

---

**한국어 임베딩** (이기창 저. 2019)

## **Chapter 2. How Vector Becomes Meaningful**

Spring Semester Capstone Study

TEAM Kai.Lib

발표자 : 배세영

2020.04.08 (WED)

이기창님 DevFest 2019 Seoul PPT 참고([링크](#))

## 1. 자연어 계산과 이해

임베딩에 자연어의 의미를 함축해 넣으려면?  
자연어의 통계적 패턴 정보를 통째로 집어넣으면 된다!

구분	Bag of Words 가정	언어 모델	분포 가정
내용	어떤 단어가 많이 쓰였는가	단어가 어떤 순서로 쓰였는가	어떤 단어가 같이 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, BERT	Word2Vec

## 2. 어떤 단어가 많이 쓰였는가

- Bag of Words 가정
  - “저자가 생각한 주제가 문서에서의 단어 사용에 녹아 있다” 는 가정
  - Bag은 중복 원소를 허용한 집합
  - 단어의 등장 순서에 관계없이 문서 내 단어의 등장 빈도를 임베딩으로 쓰는 기법
  - 간단한 방법이지만 정보 검색 분야에서 여전히 많이 쓰이고 있음

별 하나 에 추억 과  
별 하나 에 사랑 과  
별 하나 에 쓸쓸함 과  
별 하나 에 동경 과  
별 하나 에 시 와  
별 하나 에 어머니 , 어머니



어머니  
별 과 하나 사랑  
별  
시 하나 과 함 쓸쓸  
과  
하나 동경 별 하나 추억 하나  
과 별 하나 별  
별 와 어머니



별	하나	에	추억	과	사랑	쓸쓸	함	동경	시	와	어머니	,
6	6	6	1	4	1	1	1	1	1	1	2	1

---

## 2. 어떤 단어가 많이 쓰였는가

---

- TF-IDF (Term Frequency-Inverse Document Frequency)
  - 단어의 빈도, 또는 등장 여부를 그대로 임베딩으로 사용하는 것에는 큰 단점이 있음
  - 특정 단어가 많이 나타났다 하더라도 문서의 주제를 가늠하기 어려운 경우가 있기 때문 ('을/를', '이/가')
  - 이러한 단점을 보완하기 위해 만들어진 기법

$$TF - IDF(w) \\ = TF(w) \times \log\left(\frac{N}{DF(w)}\right)$$

$N$  : 전체 문서 수

$TF$  : 어떤 단어가 특정 문서에 등장한 횟수

$DF$  : 어떤 단어가 나타난 문서의 수

## 2. 어떤 단어가 많이 쓰였는가

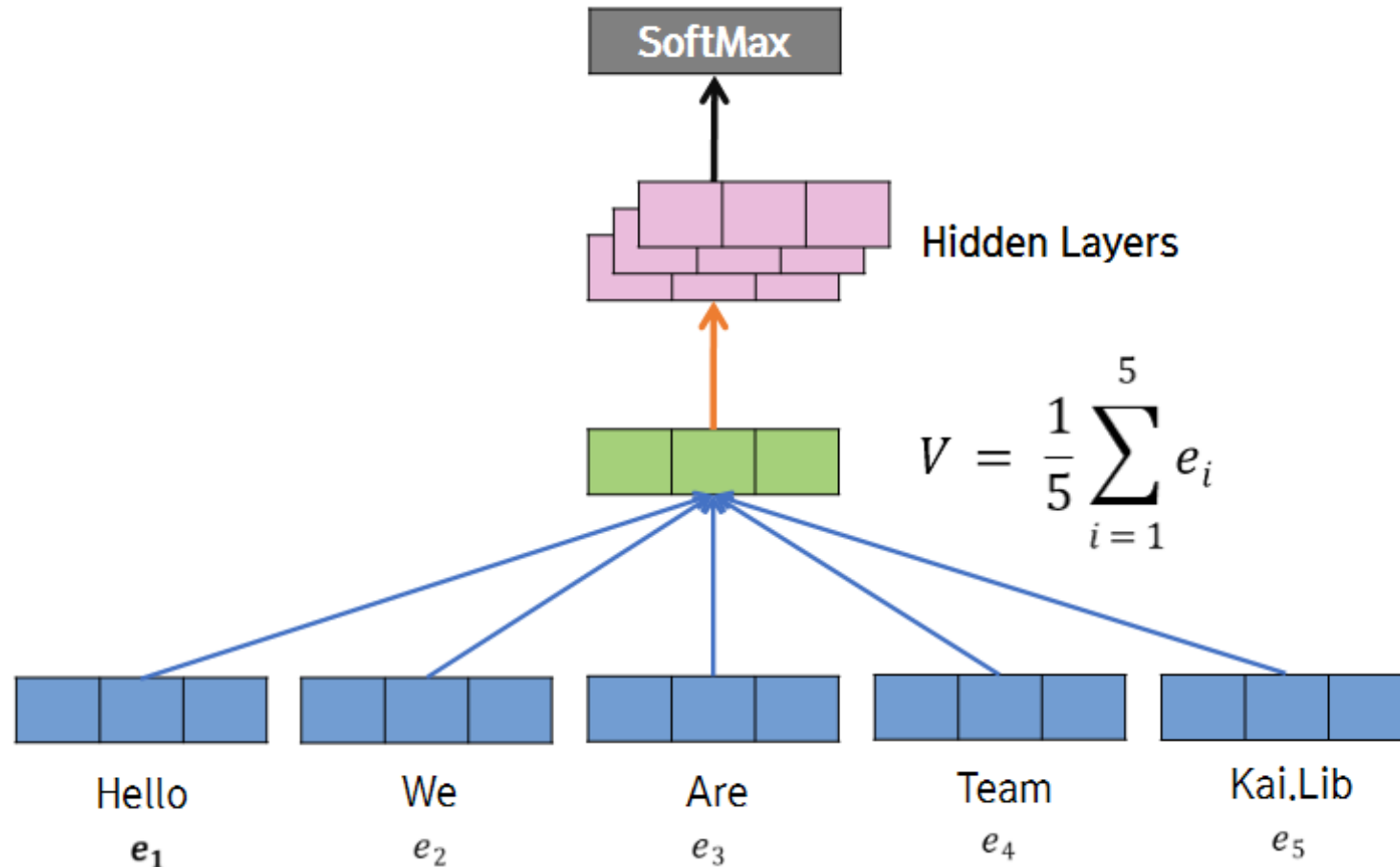
- TF-IDF (Term Frequency-Inverse Document Frequency)
  - 단어의 빈도, 또는 등장 여부를 그대로 임베딩으로 사용하는 것에는 큰 단점이 있음
  - 특정 단어가 많이 나타났다 하더라도 문서의 주제를 가늠하기 어려운 경우가 있기 때문 ('을/를', '이/가')
  - 이러한 단점을 보완하기 위해 만들어진 기법

구분	메밀꽃 필 무렵	운수 좋은 날	사랑 손님과 어머니	삼포 가는 길
담배	0.2603	0.2875	0.0364	0.2932
를	0	0.0034	0	0

TF-IDF 값이 높은 단어는  
해당 문서의 주제 파악을 좀 더 용이하게 해 준다

## 2. 어떤 단어가 많이 쓰였는가

- Deep Averaging Network
  - Bag of Words 가정의 Neural Network 버전



---

### 3. 단어가 어떤 순서로 쓰였는가

---

- 통계 기반 언어 모델
  - 언어 모델(Language Model)은 단어 시퀀스에 확률을 부여하는 역할
  - 등장 순서, 즉 시퀀스 정보를 명시적으로 학습하므로 BoW의 대척점에 있음
- 말뭉치에서  $n$ 개 단어가 연속된 시퀀스가 나타날 확률을 반환

**이전에 진행했던 LM & Fusioning 발표자료 참고!**

<https://github.com/sooftware/TIL/blob/master/Capstone-Study/LM-%26-Fusioning.pdf>

---

### 3. 단어가 어떤 순서로 쓰였는가

---

- 통계 기반 언어 모델
  - n-gram 모델에서도 희소 문제(sparsity problem)은 존재
  - 희소 문제를 해결하기 위한 백오프(Back-off), 스무딩(smoothing) 기법 제안
- Back-off
  - 목표한 n-gram의 출현 빈도가 0이면, n의 값을 줄여 탐색한 후 확률값을 보정한다

$$\begin{aligned} &Freq(\text{"이렇게 긴 시퀀스가 존재할 리 없지"}) \\ &\approx \alpha \times Freq(\text{"존재할 리 없지"}) + \beta \end{aligned}$$



### 3. 단어가 어떤 순서로 쓰였는가

- 통계 기반 언어 모델
  - n-gram 모델에서도 희소 문제(sparsity problem)은 존재
  - 희소 문제를 해결하기 위한 백오프(Back-off), 스무딩(smoothing) 기법 제안
- Smoothing
  - 출현 빈도 표에 k만큼의 값을 더해 빈도 값 자체를 조정 (Add-k Smoothing)

표현	빈도(조정 전)	k	빈도(조정 후)
이렇게	6	2	8
긴	11	2	13
시퀀스가	5	2	7
존재할	8	2	10
...			
긴 시퀀스가 존재할 리 없지	0	2	2
이렇게 긴 시퀀스가 존재할 리 없지	0	2	2

---

### 3. 단어가 어떤 순서로 쓰였는가

---

- 신경망 기반
  - 주어진 단어 시퀀스를 가지고 다음 단어를 맞추는 과정에서 학습
  - 학습이 완료되면 이들 모델의 중간 혹은 말단 계산 결과물을 단어나 문장의 임베딩으로 활용
  - ELMo, GPT등의 모델이 이에 해당

**발 없는 말이 ( )**  


- **Masked Language Model**은 위의 방식과 약간 다른 방식
  - 문장 중간에 들어갈 단어를 예측하는 과정에서 학습
  - 태생적으로 일방향 학습인 위의 모델과는 달리 양방향 학습이 가능
  - 기존 언어 모델 기법들 대비 임베딩 품질이 좋다
  - BERT가 이에 해당

**발 없는 말이 ( ) 간다**  


---

## 4. 어떤 단어가 같이 쓰였는가

---

- 분포 가정
  - 분포 : 특정 범위, 즉 Window 내에 동시에 등장하는 이웃 단어 또는 문맥의 집합
- 어떤 단어 쌍이 비슷한 문맥 환경에서 자주 등장한다면 그 의미 또한 유사할 것 이라는 가정
- 모국어 화자들이 해당 단어를 실제 어떻게 사용하고 있는지 문맥을 살펴 그 단어의 의미를 유추한다

**“단어의 의미는 곧 그 언어에서의 활용이다”**  
비트겐슈타인 (1889~1951)

---

## 4. 어떤 단어가 같이 쓰였는가

---

- 분포 가정
  - 분포 : 특정 범위, 즉 Window 내에 동시에 등장하는 이웃 단어 또는 문맥의 집합
- 어떤 단어 쌍이 비슷한 문맥 환경에서 자주 등장한다면 그 의미 또한 유사할 것 이라는 가정
- 모국어 화자들이 해당 단어를 실제 어떻게 사용하고 있는지 문맥을 살펴 그 단어의 의미를 유추한다

에서 속옷 빨래 를 하는  
물 로 빨래 할 때  
청소 와 빨래 지만 요리

Word2Vec, FastText, Glove, ...