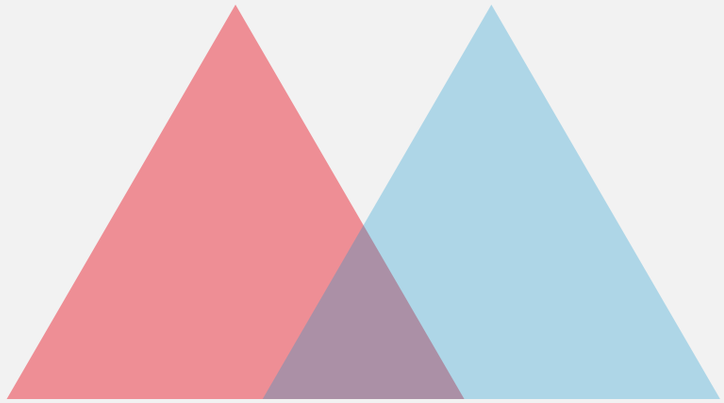
한국어 임베딩

제 1, 2장



Contents

1.서론

1.1. 임베딩이란

1.2. 임베딩의 역할

1.2.1. 단어/문장 관련도

1.2.2. 의미/문법 정보 함축

1.2.3. 전이학습

1.3. 임베딩 기법의 역사와 종류

1.4. 개발 환경

1.5. 이 책의 데이터, 주요 용어

1.6. 요약

2.벡터가 어떻게 의미를 가지게 되는가

2.1.자연어 계산과 이해

2.1.1.어떤 단어가 가장 많이 쓰였는가

2.1.1.1.백오브워즈 가정

2.1.1.2.TF-IDF

2.1.1.3.Deep Averaging Network

2.2.단어가 어떤 순서로 쓰였는가

2.2.1.통계 기반 언어 모델

2.2.2.뉴럴 네트워크 기반 언어 모델

2.3.어떤 단어가 같이 쓰였는가

2.3.1.분포의 의미 1: 형태소

2.3.2.분포의 의미 2: 품사

2.3.3.점별 상호 정보량

2.3.4.워드투벡

질문

들어가기에 앞서

0

들어가기에 앞서

언어란 무엇인가?

형태
form/
expression/
language

의미
meaning/
semantic
content

“watch”

출처 : 최재웅, 언어학과 30주년 기념 학회

0

들어가기에 앞서

질문!

- 기계가 단어나 문장의 의미를 **“이해”**할 수 있는가?
- 기계든 사람이든 단어나 문장의 의미를 이해하는지를 **어떻게** 확인할 수 있는가?

출처 : 최재웅, 언어학과 30주년 기념 학회

1장

서론

1.서론
1.1. 임베딩이란
1.2. 임베딩의 역할
1.2.1. 단어/문장 관련도
1.2.2. 의미/문법 정보 함축
1.2.3. 전이학습
1.3. 임베딩 기법의 역사와 종류
1.4. 개발 환경
1.5. 이 책의 데이터, 주요 용어
1.6. 요약

1. 임베딩이란?

- 컴퓨터는 어디까지나 빠르고 효율적인 계산기일 뿐이다.
- 따라서 **인간이 사용하는 언어** (=자연언어) 를 그대로 이해하지 못한다.
- 컴퓨터는 자연언어를 숫자로 변형하여 계산한다.
  따라서 기계의 자연언어 이해와 생성은 **연산과 처리**의 영역이다.
- 정리 : **자연어 처리** 분야에서 임베딩이란,
  사람이 쓰는 자연어를 기계가 이해할 수 있도록
  숫자의 나열인 ‘벡터’로 바꾸는 결과와 과정을 의미한다.

*자연어 처리 (NLP, Natural Language Processing)

1. 임베딩이란?

- 벡터 (vector) :
  ①여러 개의 숫자를 하나로 묶어서 사용하는 것.
  즉, n×1이나 1×n의 행렬과 같다.
  $(x_1, x_2, \ldots, x_n)$으로 표시하고,
  기호로 쓸 때는 굵은 글자로 나타낸다.

- 출처 : 한국정보통신기술협회.

1

1. 임베딩이란?

- 임베딩 (embedding) :

  단어나 문장 각각을 벡터로 변환해 벡터공간 (vector space) 으로 ‘끼워 넣는다(embed)’는 의미에서 임베딩이라는 이름이 붙여졌다.

2. 임베딩의 역할

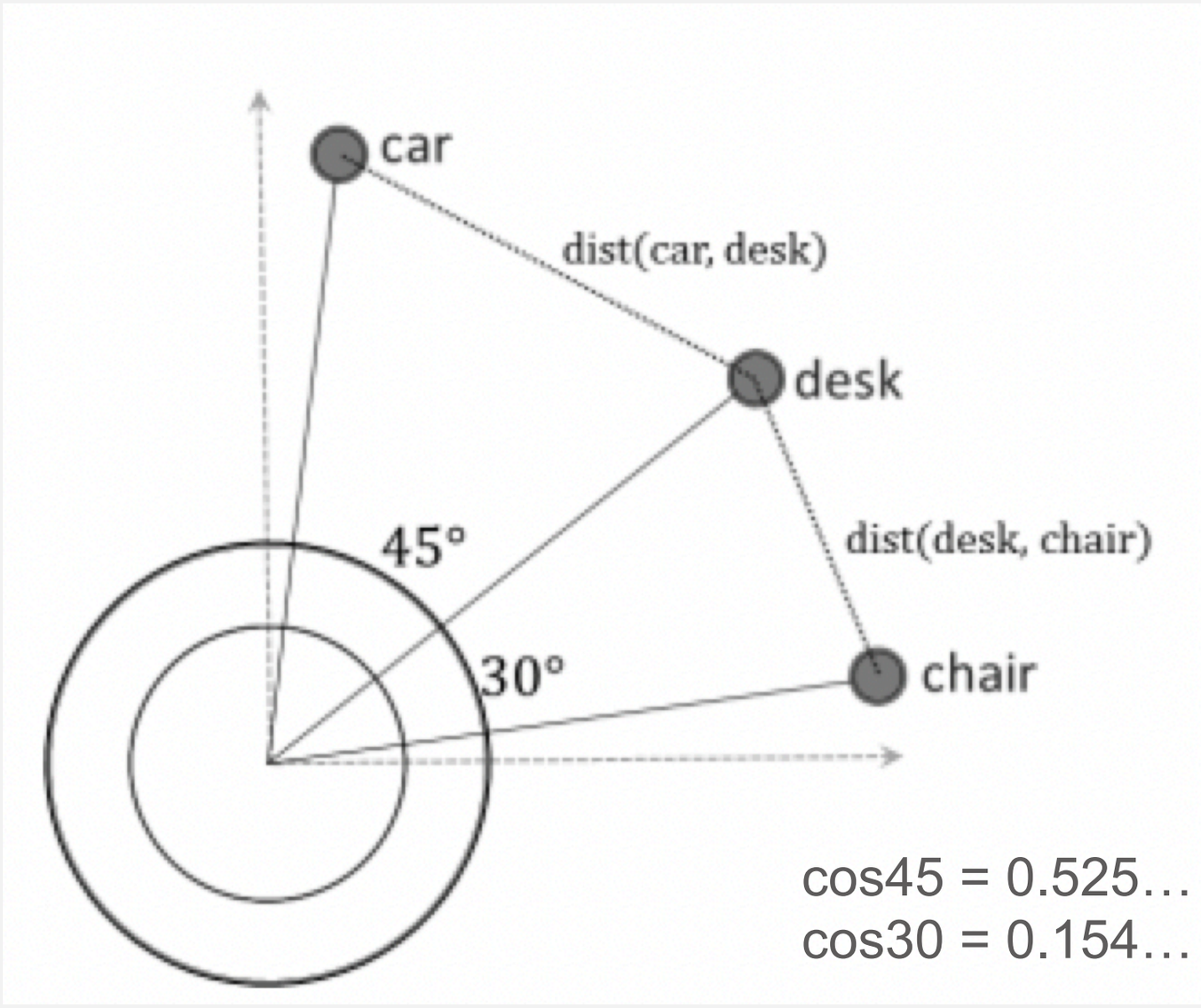
임베딩의 역할은 총 세가지로 파악할 수 있습니다.

1. 단어/문장 관련도
2. 의미/문법 정보 함축
3. 전이학습

2. 임베딩의 역할

1. **단어/문장 관련도**
2. 의미/문법 정보 함축
3. 전이학습

기본 이론 :
2차원(평면 공간) 안에
단어들의 위치를
찍을 수만 있다면,
단어들의 관련성,
어떤 단어들이
가깝고 먼지 알 수 있다.



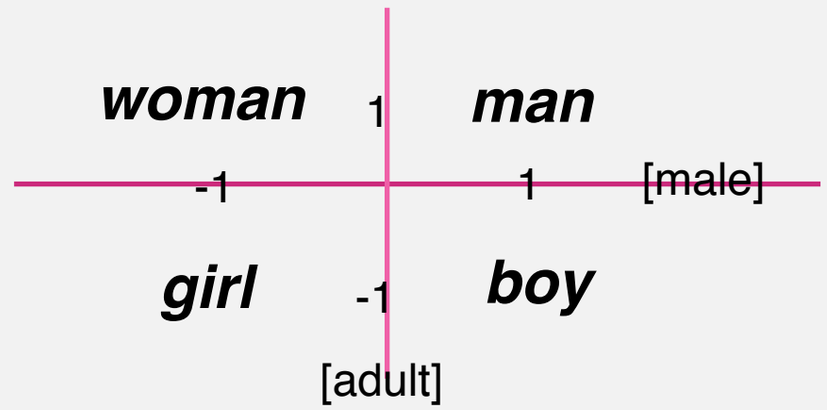
출처 : 연구계획서
송상헌. 2018.

2. 임베딩의 역할

1. 단어/문장 관련도
2. **의미/문법 정보 함축**
3. 전이학습

기본 이론 :
기하학적 표상→
수치로 표상된 의미
(의미의 수치와)

	Male	Adult
Man	+1	+1
Woman	-1	+1
Boy	+1	-1
Girl	-1	-1



1. 단어/문장 관련도
2. 의미/문법 정보 함축
3. **전이학습**

전이 학습 (transfer learning) :

(이미 만들어진) 임베딩을 다른 딥러닝 모델의 입력값으로 쓰는 기법.

설명 :
전이학습은 사람과 비슷한 학습법이다.
사람이 무언가를 배울 때, 평생 쌓아 온 지식을 바탕으로 새로운 사실을 빠르게 이해한다.
전이학습 또한 대규모 코퍼스로 미리 만들어놓은 임베딩을 입력값으로 쓴다.
이를 통해 문서 분류라는 태스크를 빠르게 잘 할 수 있게 된다.

3. 임베딩의 종류와 역사

- 변화의 흐름 :
  1. 통계 기반 → 뉴럴 네트워크 기반
  2. 단어 수준 → 문장 수준
  3. 룰 (rule) →
     엔드투엔드(end-to-end) →
     프리트레인 (pre-trained), 파인튜닝 (fine tuning)

4. 개발환경

https://ratsgo.github.io/embedding/environment.html

이곳에서 참고하세요.

(우리는 colab을 사용하지만 이 책에서는 docker을 사용하는 법을 소개합니다.)

5. 이책의 데이터와 용어

- 데이터 : 네이버 영화평가 코퍼스, 한국어 위키피디아
  - 기본 데이터 단위 : 문장
  - 책에서 다루는 가장 작은 단위 : 토큰 (token) = 형태소
- 용어:
  - **문장 :** 토큰의 집합
    문서 : 문장의 집합
    코퍼스 : 문서의 집합
  - **토크나이즈 (tokenize)** : 문장을 토큰으로 분석하는 과정을 의미.
  - **어휘 집합** : 코퍼스에 있는 모든 문서를 문장으로 나누고, 토크나이즈 실시한 후 중복을 제거한 토큰들의 집합이다.

6. 요약

- 임베딩이란 자연어를 기계가 이해할 수 있는 숫자의 나열인 **벡터**로 바꾼 결과 혹은 그 일련의 과정 전체를 가리킨다.
- 임베딩을 사용하면 **단어/ 문장 간 관련도**를 계산할 수 있다.
- 임베딩에는 **의미적/ 문법적 정보**가 함축되어있다.
- 임베딩은 **다른 딥러닝 모델의 입력값**으로 쓰일 수 있다.

6. 요약

- 임베딩 기법은
  1) 통계 기반에서 뉴럴 네트워크 기반,
  2) 단어 수준에서 문장 수준,
  3) 엔드투엔드(end-to-end)에서 프리트레인 (pre-trained),
  파인튜닝 (fine tuning) 기법으로 발전해왔다.
- 임베딩 기법은 크게 행렬 분해 모델, 예측 기반 모델, 토픽 기반 기법 등으로 나눠진다.
- 이 책이 다루는 데이터의 최소 단위는 토큰이다.
  문장은 토큰의 집합, 문서는 문장의 집합, 코퍼스는 문서의 집합을 가리킨다.

2장

벡터가 어떻게 의미를 가지게 되는가

2.벡터가 어떻게 의미를 가지게 되는가
2.1.자연어 계산과 이해
2.2.어떤 단어가 가장 많이 쓰였는가
2.2.1.백오브워즈 가정
2.2.2.TF-IDF
2.2.3.Deep Averaging Network
2.3.단어가 어떤 순서로 쓰였는가
2.3.1.통계 기반 언어 모델
2.3.2.뉴럴 네트워크 기반 언어 모델
2.4.어떤 단어가 같이 쓰였는가
2.4.1.분포의 의미 1: 형태소
2.4.2.분포의 의미 2: 품사
2.4.3.점별 상호 정보량
2.4.4.워드투벡

0

들어가기에 앞서

질문!

- 기계가 단어나 문장의 의미를 **“이해”**할 수 있는가?

- 기계든 사람이든 단어나 문장의 의미를 이해하는지를 **어떻게** 확인할 수 있는가?

출처 : 최재웅, 언어학과 30주년 기념 학회

2

2장 벡터는 어떻게 의미를 가지는가?

1. 자연어 계산과 이해

- 임베딩이 어떻게 자연어의 의미를 함축할 수 있을까?
- 이 비결은 자연어의 **통계적 패턴 (statistical pattern)** 을 통째로 임베딩에 넣는 것이다.
- 왜냐하면 자연어의 의미는 해당 화자들이 실제 사용하는 일상 언어에서 드러나기 때문이다.
- 임베딩을 만들 때 쓰는 **통계 정보**는 크게 **세가지**가 있다.

2

2장 벡터는 어떻게 의미를 가지는가?

1. 자연어 계산과 이해

구분	백오브워즈 가정 Bag of words	언어 모델	분포 가정
내용	어떤 단어가 **많이** 쓰였는가	단어가 어떤 **순서**로 쓰였는가	어떤 단어가 **같이** 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, GPT	Word2Vec

2

2. 어떤 단어가 많이 쓰였는가

구분	백오브워즈 가정 Bag of words	언어 모델	분포 가정
내용	어떤 단어가 **많이** 쓰였는가	단어가 어떤 **순서**로 쓰였는가	어떤 단어가 **같이** 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, GPT	Word2Vec

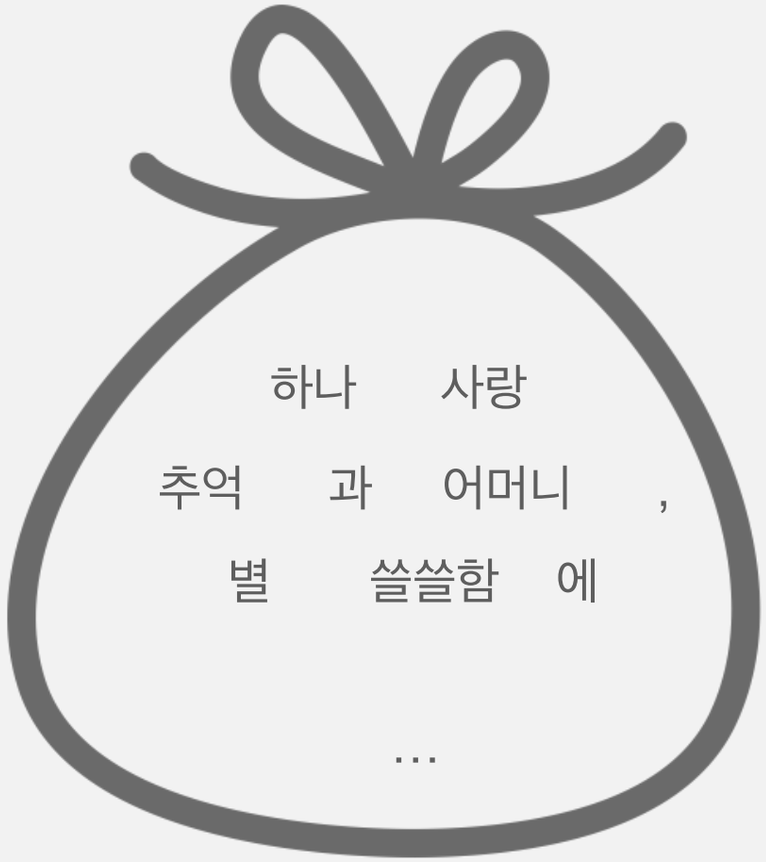
2. 어떤 단어가 많이 쓰였는가

- Bag :
  중복 원소를 허용한 집합 (multiset),
  즉, 원소들의 순서를 고려하지 않음
- Bag of words 기법 :
  단어의 등장 순서와 관계 없이
  문서 내 단어의 **등장 빈도를 임베딩으로** 쓰는 기법

2. 어떤 단어가 많이 쓰였는가

- **Bag of words 기법 :**

별 하나에 추억과
별 하나에 사랑과
별 하나에 쓸쓸함과
별 하나에 동경과
별 하나에 시와
별 하나에 어머니, 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

- 단어 빈도/ 등장여부를 그대로 임베딩에 쓰는 것은 큰 **단점** :
  단어의 빈도수가 꼭 그 문서의 주제를 나타내지는 않는다.
- 예시 : 을/를, 이/가 등의 조사는 대부분의 문서에 등장.
- 따라서 중요한 데이터란?
  다른 문서에는 안 나오는데, 해당 문서에 등장하는 데이터.

2

2장 벡터는 어떻게 의미를 가지는가?

2. 어떤 단어가 많이 쓰였는가

- 이를 보완하는 기법은
  TF-IDF
  (Term Frequency-
  Inverse Document Frequency) 이다.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

tf_{ij} = number of occurrences of i in j

df_i = number of documents containing i

N = total number of documents

- TF (Term Frequency) =
  특정 단어가 특정 문서에서 얼마나 많이 쓰였는가
- N = 전체 문서 수
- DF (Document Frequency) =
  특정 단어가 나타난 문서의 수

2

2장 벡터는 어떻게 의미를 가지는가?

2. 어떤 단어가 많이 쓰였는가

- ‘담배’ 라는 특정 단어.
- TF = ‘운수 좋은 날’ 에 등장한 ‘담배'의 수
- DF = 전체 문서에 등장한 '담배'의 수
- TF 가 클수록,
  DF 가 작을수록,
  결과값 TF-IDF 는 커진다!
- 단어 사용 빈도는 저자가 상정한 주제와 관련을 맺고 있을 것이라는 가정에 기초함.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

tf_{ij} = number of occurrences of i in j

df_i = number of documents containing i

N = total number of documents

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

구분	백오브워즈 가정 Bag of words	언어 모델	분포 가정
내용	어떤 단어가 **많이** 쓰였는가	단어가 어떤 **순서**로 쓰였는가	어떤 단어가 **같이** 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, GPT	Word2Vec

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어모델 (language model) 이란 **단어 시퀀스 (순서)** 에 확률을 부여하는 모델이다.
  - 앞선 백오브워즈 가정과는 달리,
    언어 모델은 시퀀스 정보를 명시적으로 학습한다.
- 언어 모델에는 두가지 분류가 있다.
  1. 통계 기반 언어 모델
  2. 뉴럴 네트워크 기반 언어 모델

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어 모델에는 두가지 분류가 있다.
    1. **통계 기반 언어 모델**
    2. 뉴럴 네트워크 기반 언어 모델

통계적 언어모델 (Statistical Language Model, SLM)

단어가 n개 주어진 상황이라면,

언어 모델은 n개의 단어가 동시에 나타날 확률을 반환한다.

자연스러운 한국어 문장에 높은 확률 값을 부여한다.

예시 : 누명을 쓰다 (0.41) / 누명을 당하다 (0.02)

잘 학습된 언어 모델이 있다면 어떤 문장이 그럴듯한지 (확률 값이 높은지) 알 수 있다.

3. 단어가 어떤 순서로 쓰였는가

예시 문장 : An adorable little boy is spreading smiles.

An adorable little boy 뒤에 is가 나올 확률?

최대우도추정법 (Maximum Likelihood Estimation)으로

‘is'가 나올 확률 계산:

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count(An adorable little boy is)}}{\text{count(An adorable little boy)}}$$

예를 들어 기계가 학습한 코퍼스 데이터에서 (An adorable little boy)가 100번 등장했는데
그 다음에 is가 등장한 경우는 30번이라고 합시다.
이 경우 확률은 30% 입니다.
문제 : count(An adorable little boy) = 0 이라면?

2

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

	여성	남성	계
대학생	100	80	180
교수	20	40	60
계	120	120	240

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count(An adorable little boy is)}}{\text{count(An adorable little boy)}}$$

조건부확률

A = 전체 중 여성인 사건

B = 전체 중 남성인 사건

C = 전체 중 대학생인 사건

D = 전체 중 교수인 사건

한 명을 뽑았을 때, 교수일 확률:

P(D) = 60/240 = 1/4

한 명을 뽑았을 때, 여성 교수일 확률:

P(A∩D) = 20/240 = 1/12

교수 중 한 명을 뽑았을 때, 여성일 확률:

P(A|D) = P(A∩D)/P(D) = (20/240)/(60/240)

=1/3

3. 단어가 어떤 순서로 쓰였는가

- 희소 문제 (Sparsity Problem):
  충분한 데이터를 관측하지 못하여 언어를 정확히 모델링하지 못하는 문제

- 카운트 기반 접근의 한계
  언어 모델은 **실생활에서 사용되는 언어의 확률 분포를 근사 모델링** 합니다.
  실제로 정확하게 알아볼 방법은 없겠지만 현실에서도 An adorable little boy가 나왔을 때 is가 나올 확률이라는 것이 존재합니다.
  이를 실제 자연어의 확률 분포, 현실에서의 확률 분포라고 명칭합시다.
- 기계에게 많은 코퍼스를 훈련시켜서 언어 모델을 통해 현실에서의 확률 분포를 근사하는 것이 언어 모델의 목표입니다.
  그런데 카운트 기반으로 접근하려고 한다면
  갖고있는 방대한 양의 코퍼스(corpus)가 필요합니다.

2

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count(An adorable little boy is)}}{\text{count(An adorable little boy)}}$$

- 문제 1:
  기계가 훈련한 코퍼스에 (An adorable little boy is)라는 단어 시퀀스가 없었다면
  이 단어 시퀀스에 대한 확률은 0이 됩니다. P=0
- 문제 2:
  (An adorable little boy)라는 단어 시퀀스가 없었다면
  분모가 0이 되어 확률은 정의되지 않습니다. P = nonexistent
- **희소 문제 (Sparsity Problem):**
  충분한 데이터를 관측하지 못하여
  언어를 정확히 모델링하지 못하는 문제

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어 모델에는 두가지 분류가 있다.
    1. **통계 기반 언어 모델**
    2. 뉴럴 네트워크 기반 언어 모델

N-gram 모델:
직전 n-1개 단어의 등장 확률로
전체 단어 시퀀스 등장 확률을 근사하는 모델.
(Markov assumption 을 기반으로)

*n-gram : n개의 단어들을 묶었다는 뜻. 학습한 단위를 나타낸다.

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어 모델에는 두가지 분류가 있다.
    1. **통계 기반 언어 모델**
    2. 뉴럴 네트워크 기반 언어 모델

앞에서의 방법 :

$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count(An adorable little boy is)}}{\text{count(An adorable little boy)}}$$

N-gram의 방법 :

$$P(\text{is}|\text{An adorable little boy}) \approx P(\text{is}|\text{boy})$$

$$P(\text{is}|\text{An adorable little boy}) \approx P(\text{is}|\text{little boy})$$

앞 단어 중 임의의 개수만 포함해서 카운트 후,

근사(approximate) 값을 취한다.

효과 : 갖고 있는 코퍼스에서 해당 단어의 시퀀스를 카운트할 확률이 높아진다.

2

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어 모델
    1. 통계 기반 언어 모델
    2. **뉴럴 네트워크 기반 언어 모델**
        - **2.1. 언어모델**
        - 2.2. Mask 모델

뉴럴 네트워크:
입력과 출력 사이의 관계를 유연하게 포착할 수 있고, 그 자체로 확률 모델로 기능할 수 있다.

예시 : 발 없는 말이 —> (언어 모델) —> 간다. (추측)

예시에 주어진 단어 시퀀스를 바탕으로 **다음 단어를 맞추는 (prediction) 과정에서 학습된다.**

ELMo, GPT 등 모델이 여기에 해당한다. (1/17 실습의 내용)

2

2장 벡터는 어떻게 의미를 가지는가?

3. 단어가 어떤 순서로 쓰였는가

- 언어 모델
    1. 통계 기반 언어 모델
    2. **뉴럴 네트워크 기반 언어 모델**
        - 2.1. 언어모델
        - **2.2. Mask 모델**

마스크 모델은 뉴럴 네트워크와 다르다.

예시 : 발 없는 말이 (MASK) 간다. —> (MASK) 추측 : [천리]

언어 모델은 일방향으로, 순차적으로 단어를 입력받아 다음 단어를 추측한다.
마스크 모델은 양방향 학습이 가능하다.

BERT 가 이 부류에 속한다. 다음주. (1/17 실습의 내용)

4. 어떤 단어가 같이 쓰였는가

구분	백오브워즈 가정 Bag of words	언어 모델	분포 가정
내용	어떤 단어가 **많이** 쓰였는가	단어가 어떤 **순서**로 쓰였는가	어떤 단어가 **같이** 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, GPT	Word2Vec

2장 벡터는 어떻게 의미를 가지는가?

4. 어떤 단어가 같이 쓰였는가

1.분포 가정

2.점별 상호 정보량 (Pointwise Mutual Information)

3.Word2Vec

- 분포 가정은 **문장에서 어떤 단어가 같이 쓰였는지가** 중요하다.
- 자연어 처리에서 분포 (distribution) :
  특정 범위 내에 동시에 등장하는 이웃 단어/ 문맥의 집합
- 분포 가정의 전제 :
  어떤 단어 쌍이 비슷한 **문맥** 환경에서 자주 등장한다면 그 의미 또한 유사할 것.

2

2장 벡터는 어떻게 의미를 가지는가?

4. 어떤 단어가 같이 쓰였는가

1.분포 가정

2.점별 상호 정보량 (Pointwise Mutual Information)

3.Word2Vec

- 분포 가정의 전제 :
  어떤 단어 쌍이 비슷한 **문맥** 환경에서 자주 등장한다면 그 의미 또한 유사할 것.
- 예시 :
  다리가 아프다. - leg // 팔이 아프다. 머리가 아프다
  다리가 지어졌다. - bridge // 건물이 지어졌다. 집이 지어졌다.

* 흔히 사용되는 '문맥' 개념과의 차이점 :
글월에 표현된 의미의 앞뒤 연결. (X)
특정 범위 (= 윈도우) 내에 속하는 단어들 (O)

2

2장 벡터는 어떻게 의미를 가지는가?

4. 어떤 단어가 같이 쓰였는가

1.분포 가정

2.점별 상호 정보량 (Pointwise Mutual Information)

3.Word2Vec

점별 상호 정보량 (PMI) :
두 확률변수 사이의 상관성을 숫자로 변환하는 단위다.

$$\mathrm{PMI}(x, y) = \log \frac{p(x, y)}{p(x)p(y)}$$

두 확률변수가 완전히 '독립'인 경우 그 값이 0이 된다.
PMI 는 두 단어의 등장이 독립일 때 대비해 얼마나 자주 같이 등장하는지를 수치화한 것.
조건부확률 : 사건 Y가 일어났을 때 사건 X가 일어날 확률

4. 어떤 단어가 같이 쓰였는가

1.분포 가정

2.점별 상호 정보량 (Pointwise Mutual Information)

3.Word2Vec

CBOW (왼쪽):
문맥 단어들을 통해 타깃 단어 하나를 맞추는 과정에서 학습됨.
(코퍼스 크기 작을 때 성능 좋음)

Skip-gram (오른쪽):
타깃 단어를 가지고 문맥 단어가 무엇일지 예측하는 과정에서 학습됨
(코퍼스 일정 크기 이상일 때 많이 쓰임)

4. 어떤 단어가 같이 쓰였는가

1.분포 가정

2.점별 상호 정보량 (Pointwise Mutual Information)

3.Word2Vec

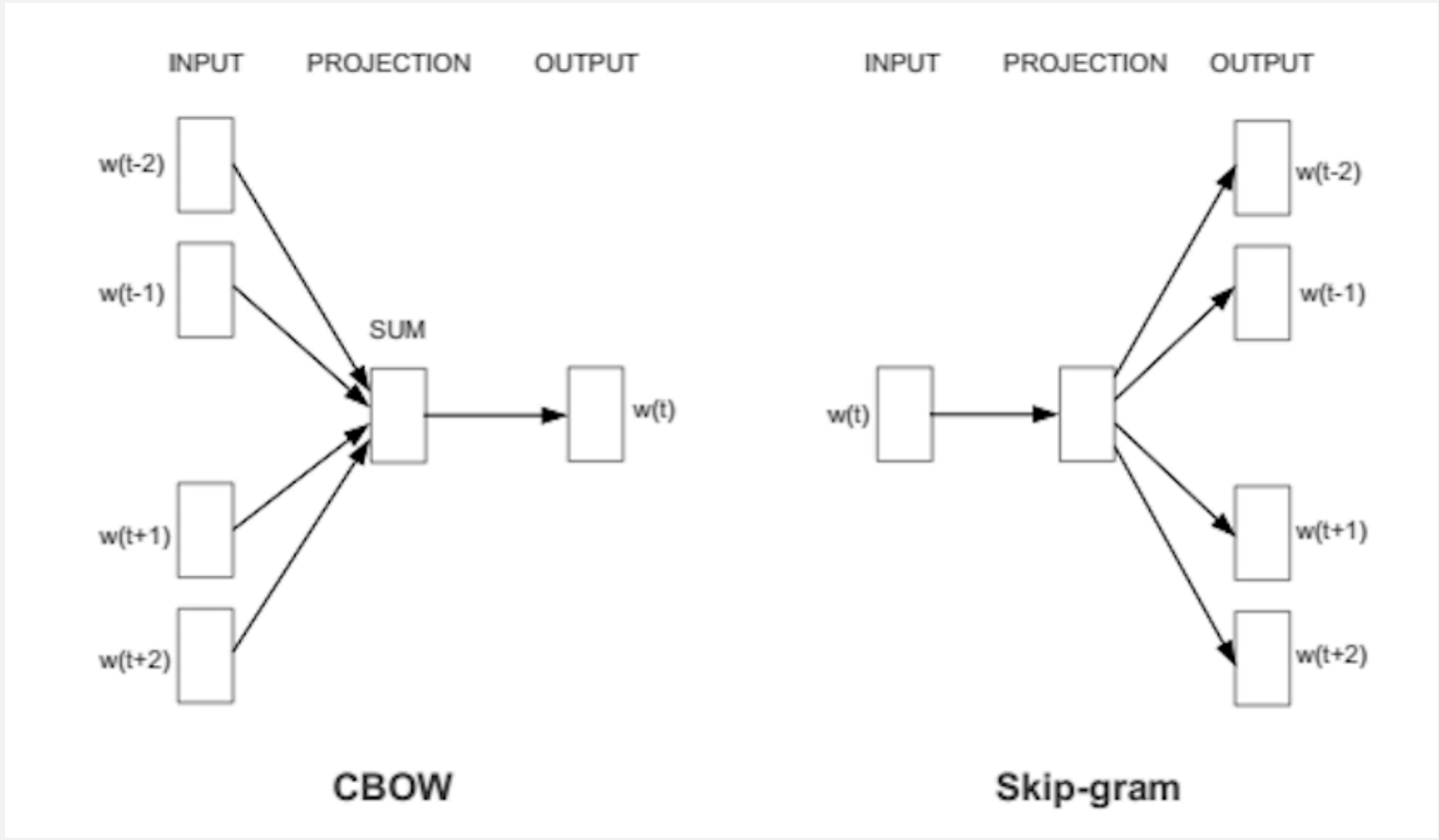


그림 : CBOW vs. Skip-gram (Mikolov et al. 2013)

둘 모두 특정 타깃 단어 주변의 **문맥, 즉 분포 정보를** 임베딩에 함축한다.

2

코사인 유사도

- 코사인 유사도 :
  두 벡터 간의 코사인 각도를 이용하여 구할 수 있는 두 벡터의 유사도.

- 단어를 임베딩하여 수치화하였으면,
  이러한 표현 방법에 대해서 코사인 유사도를 이용하여 문서의 유사도를 구하는 게 가능하다.

2

2장 벡터는 어떻게 의미를 가지는가?

코사인 유사도

- 두 벡터의 방향이 완전히 동일한 경우에는 1의 값을 가지며, 90°의 각을 이루면 0, 180°로 반대의 방향을 가지면 -1의 값을 갖게 됩니다.
- 즉, 결국 코사인 유사도는 -1 이상 1 이하의 값을 가지며

값이 1에 가까울수록 유사도가 높다고 판단할 수 있습니다.

이를 직관적으로 이해하면

두 벡터가 가리키는 방향이 얼마나 유사한가를 의미합니다.

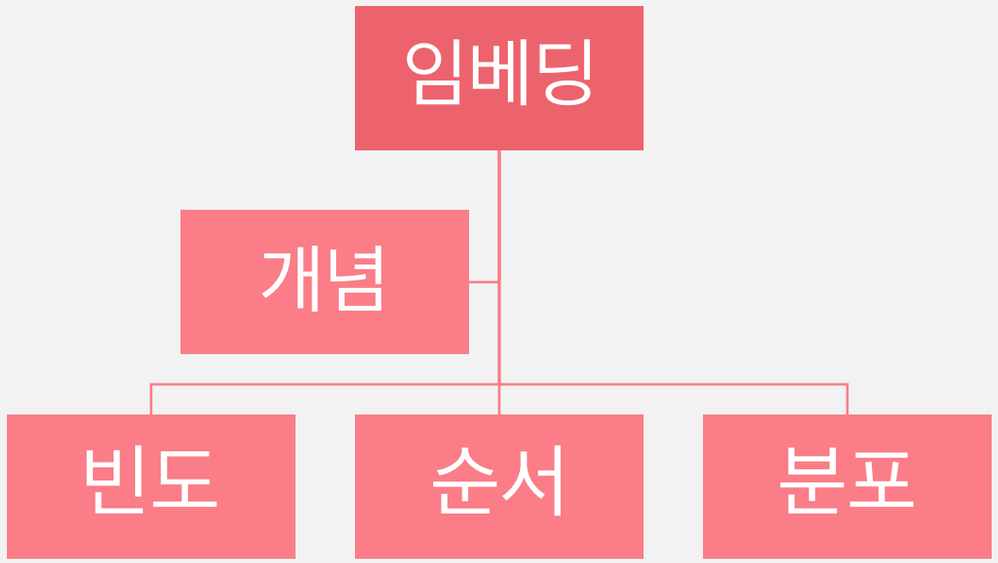
5. 요약

- 임베딩이 자연어의 의미를 함축하는 방법은
  자연어의 **통계적 패턴 (statistical pattern)** 을 통째로 임베딩에 넣는 것이다.

구분	백오브워즈 가정 Bag of words	언어 모델	분포 가정
내용	어떤 단어가 **많이** 쓰였는가	단어가 어떤 **순서**로 쓰였는가	어떤 단어가 **같이** 쓰였는가
대표 통계량	TF-IDF	-	PMI
대표 모델	Deep Averaging Network	ELMo, GPT	Word2Vec

- 백오브워즈 가정, 언어 모델, 분포 가정은
  코퍼스의 통계적 패턴을 서로 다른 각도에서 분석하는 것이며 **상호 보완적**이다.

1, 2장 전체 요약 정리



앞으로 배울 내용

- 4장 **단어** 수준 임베딩:
  - Word2Vec
  - FastText
  - GloVe
- 5장 **문장** 수준 임베딩
  - ELMo
  - BERT

*

한국어 임베딩 1, 2장

참고문헌

- 한국어 임베딩

  이기창. 2019. 에이콘 출판사.

- 딥러닝을 이용한 자연어 처리 입문

  Won Joon Yoo. 2020~.

  https://wikidocs.net/book/2155

- 한국어 임베딩 깃헙

  https://ratsgo.github.io/embedding/

한국어 임베딩
1, 2장
발표자 언어학과 17학번 최유정

경청해주셔서 감사합니다.