

2022학년도 1학기 컴퓨터언어학

제5강 벡터의미론과 임베딩 (1)

박수지

서울대학교 인문대학 언어학과

2022년 3월 21일 월요일

오늘의 목표

- 1 분포 가설이 무엇인지 설명할 수 있다.
- 2 주어진 코퍼스에서 공기행렬을 도출할 수 있다.
- 3 두 단어 벡터 사이의 코사인 유사도를 계산할 수 있다.
- 4 TF-IDF 방식으로 벡터의 가중치를 구할 수 있다.

분포 가설(Distributional hypothesis, 1950s-)

유사한 문맥에서 나타나는 단어는 유사한 의미를 가진다.

유사한 문맥 이웃하는 단어들의 목록이 겹친다.

유사한 의미 ...

단어의 의미 관계

유사성(Similarity) 두 단어의 이웃이 비슷한 경우 (paradigmatic association)

- 예: 커피-차(茶) (이웃: 마시다, 우리다, 컵, 잔, 따뜻한, ...)

연관성(Relatedness) 두 단어가 서로 이웃인 경우 (syntagmatic association)

- 예: 커피-컵

단어 임베딩(Word embedding)

단어의 의미를 벡터공간에 표상하는 것

의미가 비슷한 단어들이 서로 가까운 벡터로 표상되도록…

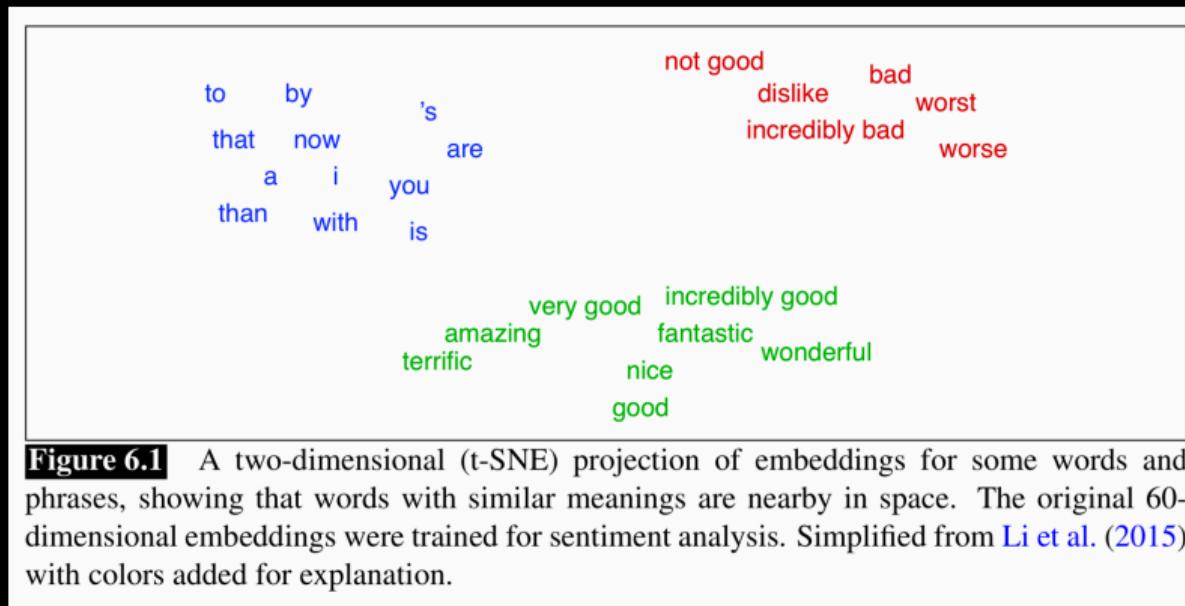


Figure 6.1 A two-dimensional (t-SNE) projection of embeddings for some words and phrases, showing that words with similar meanings are nearby in space. The original 60-dimensional embeddings were trained for sentiment analysis. Simplified from Li et al. (2015) with colors added for explanation.

문제

단어의 분포를 어떻게 벡터로 나타낼 것인가?

전통적인 해법

공기행렬을 이용한다.

공기행렬(Co-occurrence matrix)

단어의 공기 관계를 나타내는 행렬

- 1 단어-문서 행렬(Term-document matrix)
- 2 단어-단어 행렬(Term-term matrix)

벡터와 문서

단어-문서 행렬

단어-문서 행렬 얻는 법

- 1 여러 문서로 이루어진 코퍼스에서 각 문서에 각 단어가 나타난 횟수를 센다.
- 2 단어를 행에, 문서를 열에 대응시켜서 해당하는 성분에 출현 횟수를 넣는다.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Figure 6.2 The term-document matrix for four words in four Shakespeare plays. Each cell contains the number of times the (row) word occurs in the (column) document.

벡터와 문서

단어-문서 행렬의 활용: 문서 벡터

문서 벡터

- 단어-문서 행렬의 열벡터.
- 코퍼스의 어휘 집합(vocabulary)을 V 라고 할 때 $|V|$ 차원 벡터가 된다.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Figure 6.3 The term-document matrix for four words in four Shakespeare plays. The red boxes show that each document is represented as a column vector of length four.

벡터와 문서

단어-문서 행렬의 활용: 문서 벡터

서로 “비슷한” 문서 찾기

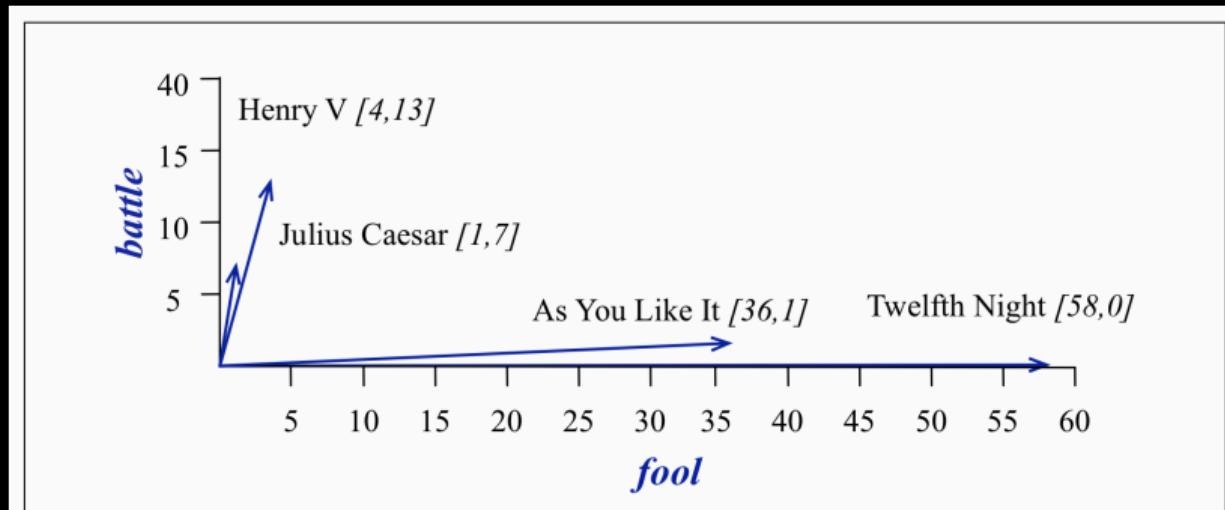


Figure 6.4 A spatial visualization of the document vectors for the four Shakespeare play documents, showing just two of the dimensions, corresponding to the words *battle* and *fool*. The comedies have high values for the *fool* dimension and low values for the *battle* dimension.

단어 벡터: 문서 차원

단어 벡터

- 단어-문서 행렬의 행벡터.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Figure 6.5 The term-document matrix for four words in four Shakespeare plays. The red boxes show that each word is represented as a row vector of length four.

단어 벡터: 단어 차원

단어-단어 행렬 얻는 법

- 1 대상 단어(target word)가 문맥 단어(context word)와 공기하는 횟수를 센다.
 - 일반적으로 $\pm L$ 개 단어의 이웃 내에 있으면 공기한다고 간주한다.
- 2 대상 단어를 행에, 문맥 단어를 열에 대응시켜서 해당하는 성분에 공기 횟수를 넣는다.

공기 예시

$L = 4$ 일 때 대상 단어 **cherry**의 문맥 단어 **is, traditionally, ..., traditional, dessert**

is traditionally followed by	cherry	pie, a traditional dessert
often mixed, such as	strawberry	rhubarb pie. Apple pie
computer peripherals and personal	digital	assistants. These devices usually
a computer. This includes	information	available on the internet

단어 벡터: 단어 차원

또 다른 단어 벡터

- 단어-단어 행렬의 행벡터.

	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

Figure 6.6 Co-occurrence vectors for four words in the Wikipedia corpus, showing six of the dimensions (hand-picked for pedagogical purposes). The vector for *digital* is outlined in red. Note that a real vector would have vastly more dimensions and thus be much sparser.

주의: 공기행렬은 차원이 크고 희소하다(대부분의 성분이 0의 값을 가진다).

단어 벡터: 단어 차원

서로 “비슷한” 단어 찾기

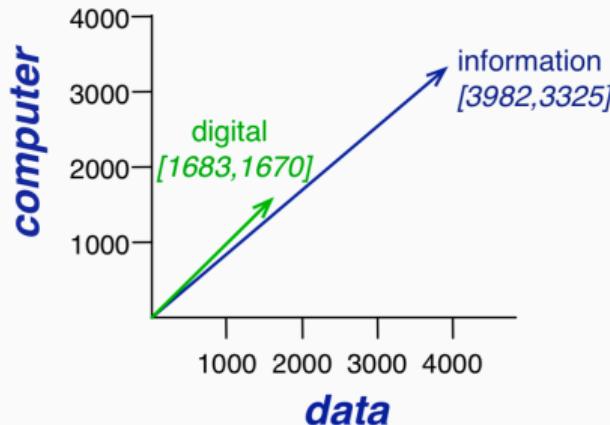


Figure 6.7 A spatial visualization of word vectors for *digital* and *information*, showing just two of the dimensions, corresponding to the words *data* and *computer*.

문제

두 벡터가 얼마나 유사한지를 어떻게 측정하는가?

해법

두 벡터 사이의 각이 작을수록 가깝다고 설정한다.

⇒ 코사인 유사도(Cosine similarity)라는 척도를 사용한다.

코사인 유사도

벡터 $\vec{v} = [v_1, v_2, \dots, v_N]$ 와 $\vec{w} = [w_1, w_2, \dots, w_N]$ 사이의 각이 θ 일 때 ($0 \leq \theta \leq \pi$)

$$\text{cos-sim}(\vec{v}, \vec{w}) = \cos \theta = \frac{\vec{v} \cdot \vec{w}}{|\vec{v}| |\vec{w}|} = \frac{v_1 w_1 + v_2 w_2 + \dots + v_N w_N}{\sqrt{v_1^2 + v_2^2 + \dots + v_N^2} \sqrt{w_1^2 + w_2^2 + \dots + w_N^2}}$$

$$\cos \theta = +1 \Rightarrow \theta = 0 = 0^\circ \text{ (같은 방향)}$$

$$\cos \theta = 0 \Rightarrow \theta = \frac{\pi}{2} = 90^\circ \text{ (수직인 방향)}$$

$$\cos \theta = -1 \Rightarrow \theta = \pi = 180^\circ \text{ (반대 방향)}$$

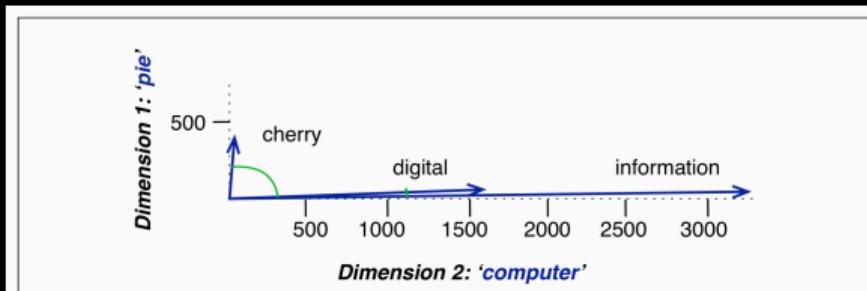


Figure 6.8 A (rough) graphical demonstration of cosine similarity, showing vectors for three words (*cherry*, *digital*, and *information*) in the two dimensional space defined by counts of the words *computer* and *pie* nearby. The figure doesn't show the cosine, but it highlights the angles; note that the angle between *digital* and *information* is smaller than the angle between *cherry* and *information*. When two vectors are more similar, the cosine is larger but the angle is smaller; the cosine has its maximum (1) when the angle between two vectors is smallest (0°); the cosine of all other angles is less than 1.

공기행렬의 한계

단순 빈도수만으로는 유의미한 정보를 담기 어렵다.

- the, it 등 자주 나오는 단어의 공기 횟수가 가장 많이 나오지만…
- 이러한 단어는 정보량이 적다.

단어-문서 행렬의 보완

TF-IDF: Term Frequency - Inverse Document Frequency

- 단어-문서 행렬에서 단순히 출현 횟수를 세었던 것을 보강해서…
- 정보량이 많은 단어와 적은 단어의 가중치를 다르게 준다.

TF-IDF

개념

$$\text{TF-IDF}(t, d) = \text{tf}_{t,d} \times \text{idf}_t$$

$\text{tf}_{t,d} = \log_{10}[\text{단어 } t \text{가 문서 } d \text{에 출현한 횟수}] + 1$

■ 문서에 자주 나타날수록 중요한 단어(주제어)다.

- 문서에 한 번도 나오지 않은 $\text{tf}_{t,d}$ 의 값은 0이다.
- tf값이 높을수록 문서 내에서 중요하다.

$$\text{idf}_t = \log_{10} \frac{[\text{전체 문서의 개수}]}{[\text{단어 } t \text{를 포함하는 문서의 개수}]}$$

■ 이 문서 저 문서에 다 나오는 단어(예: the)은 정보량이 적다.

- 모든 문서에 나오면 idf_t 값이 0이 된다.
- idf값이 높을수록 일반적으로 중요하다.

100개 문서에 1번씩 나오기 vs. 10개 문서에 10번씩 나오기

TF-IDF(Term Frequency - Inverse Document Frequency)

특징

- 차원 수가 크다. (코퍼스 내 전체 문서의 개수와 같음)
- 대부분의 값이 0이다. (한 문서에 들어가는 어휘는 전체의 일부에 불과함)

희소벡터(sparse vector)의 문제

- 1 기계학습의 특성값으로 사용하기 어렵다.
- 2 계산해야 할 매개변수의 수가 많다.
- 3 동의어를 포착하기 어렵다.
 - car와 automobile이 같은 문서에 출현하는 일은 적다. → 벡터의 차이가 크다.

오늘 배운 것

- 1 분포 가설: 두 단어가 서로 유사한 문맥을 가지면 유사한 의미를 가진다.
- 2 공기행렬: 유사한 문맥은 단어의 공기 횟수로 포착할 수 있다.
- 3 코사인 유사도: 서로 유사한 벡터는 서로 작은 각을 이룬다.
- 4 TF-IDF: 공기행렬에 단어의 정보량을 반영한다.

다음 시간에 할 일

SLP3 6.8 Word2Vec 읽어 오기