GloVe: Global Vectors for Word Representation

집현전 초급반 김지은

2021.01.17(일)

Introduction

단어 벡터는 다양한 분야에서 사용되며, 대부분 단어 벡터 사이의 관계나 각도를 이용하여 벡터의 성능을 평가한다.

Mikolov는 2013년에 **차원의 차이**로 평가하는 방법을 제안했다.

(예시) analogy: "king is to queen as man is to woman"
vector equation: "king" - "queen" = "man" – "woman"

이는 차원의 의미를 기반으로 하는 모델의 도입이 가능해지게 만들었다.

Introduction

단어 학습 방법은 크게 **두 가지**로 분류된다.

1) 전체적인 통계 정보를 사용하는 방법

장점
- 빠른 학습
- Global statistics의 효율적 활용

단점
- 단어 유사도 파악만 가능하고, analogy는 불가능하다.
- 빈도수가 큰 단어들에 대해 불균형하다.

종류
- LSA, HAL, COALS 등

Introduction

단어 학습 방법은 크게 **두 가지**로 분류된다.

2) 지정한 크기의 window 안에 위치하는 문맥을 파악하는 방법

Skip-gram과 같은 방법이 있으며, 해당 방법은 analogy task에는 효과적으로 사용되나 각각의 corpus에 대한 통계량만을 반영하기 때문에 전체적인 데이터를 반영하지 못한다는 단점이 있다.

종류
- NNPM, HLBL, RNN, CBOW 등

본 논문에서는 global word-word-co-occurrence count를 이용한 weighted least squares model을 소개한다. 의미의 선형 방향을 생성하며, LSA의 메커니즘이었던 **카운트 기반의 방법**과 Word2Vec의 메커니즘이었던 **예측 기반**의 방법론 두 가지를 모두 사용한다.

The GloVe Model

notation

- ✓ X: $matrix\ of\ word - word - co - occurrence\ counts$
- ✓ X_{ij}: 단어 i가 나타난 문맥에서 단어 j가 나타난 횟수
- ✓ X_i: 단어 i가 존재하는 모든 문맥에서 다른 모든 단어들이 나타난 횟수
- ✓ $Pij = P(j|i) = X_{ij}/X_i$: 단어 j 가 단어 i 의 문맥에서 나타날 확률

단어의 의미는 co-occurrence 확률에서 직접 구할 수 있다.

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

위는 $i = ice, j = steam$ 일때 확률값이다. 두 단어의 관계를 새로운 단어 k를 통해 파악할 수 있다.

The GloVe Model

해석

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

1. 확률 값이 작거나 큰 경우 단어 i, j 둘 중 하나와 k가 더 가깝다.
2. 확률 값이 1에 가까운 경우 k가 i, j와 모두 가깝거나 모두와 멀다.

The GloVe Model

co-occurrence probability를 활용하여 유의미한 결과를 얻을 수 있다.

여기서 확률의 비율은 i, j, k 세 단어에 대해 의존적이다.

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} \qquad \textbf{(1)}$$

단어에 대한 일반식

$$F\left((w_i - w_j)^T \tilde{w}_k\right) = \frac{P_{ik}}{P_{jk}} \qquad \textbf{(3)}$$

식 (3)은 선형 구조식에 대해 차이에 대해 식으로 나타낸 것이다. 함수 F의 결과는 vector이기 때문에 우변의 항과 맞추기 위해 내적한다. 또한 임의로 선택된 단어들에 대해 두 단어의 위치가 바뀌어도 동일한 결과를 보여야 하는 조건을 충족해야한다.

The GloVe Model

식 (3)을 준동형 사상을 만족하도록 변경한 수식의 결과는 다음과 같다.

$$F\left((w_i - w_j)^T \tilde{w}_k\right) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)} \quad \textbf{(4)}$$

모든 조건을 만족하는 F를 exponantial 함수로 정의하면 다음과 같다.

$$w_i^T \tilde{w}_k = \log(P_{ik}) = \log(X_{ik}) - \log(X_i) \quad \textbf{(6)}$$

The GloVe Model

수식 (6)에 bias를 추가한 후 i와 k가 동시에 등장하지 않는 경우를 해결하기 위해 $f(X_{ij})$ 함수를 이용한다. Weighing function을 사용하여 손실 함수를 다음과 같이 정의할 수 있다.

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2 \qquad \textbf{(8)}$$

V는 단어들의 크기이다. 또한 다음의 조건을 만족해야한다.

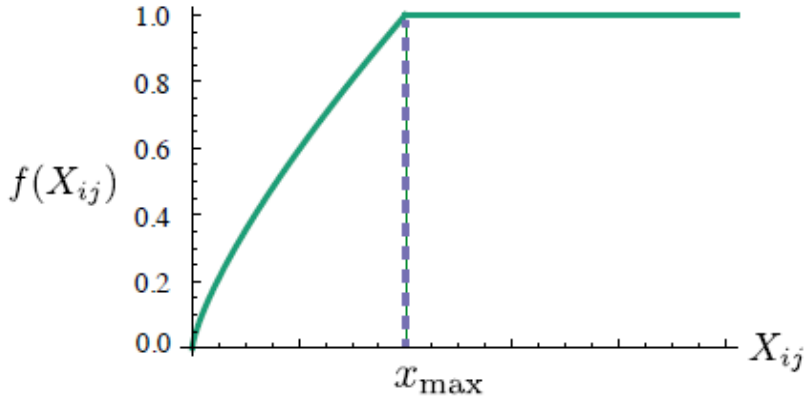
- ✓ $f(0) = 0$, f가 연속형 함수일 때 0으로 향하는 속도가 더 빨라야한다.
- ✓ $f(x)$는 감소 함수가 아니어야 한다.
- ✓ $f(x)$는 자주 나타나는 x에 대해 너무 큰 가중치를 부여하면 안된다.

The GloVe Model

본 논문에서 정의한 $f(x)$는 다음과 같다.

$$f(x) = \begin{cases} (x/x_{\max})^{\alpha} & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \quad \textbf{(9)}$$

α = 3/4일 때 식 (9)를 시각화한 결과이다.



Experiments

실험에서 사용한 데이터는 다음과 같다.

- ✓ word analogy task of Mikolov et al. (2013a)
- ✓ a variety of word similarity tasks
- ✓ on the CoNLL-2003 shared benchmark

word analogy

"a is to b as c is to __?" 에서 빈칸에 들어갈 단어를 찾는 문제이다. 데이터 셋은 19,544개의 질문들로 구성되어 있고, semantic한 자료와 syntactic한 자료로 구분된다.

코사인 유사도를 통해 $w_b - w_a + w_c$와 가장 가까운 w_d를 찾아내는 문제이다.

Experiments

Word Similarity

모형을 평가하기 위해 다음과 같은 데이터 셋을 사용한다.

- ✓ WordSim-353(Finkelstein et al., 2001)
- ✓ MC(Miller and Charles, 1991)
- ✓ RG(Rubenstein and Goodenough, 1965)
- ✓ SCW(Huang et al.,2012)
- ✓ RW(Luong et al., 2013)

Experiments

Named Entity Recognition

NER을 위한 데이터 셋인 CoNLL-2003은 뉴스 기사로 구성되어있다. 4가지의 그룹을 분류하기위한 데이터이다.
사용한 데이터는 아래와 같다.

- ✓ ConLL-03 testing data
- ✓ ACE Phase2 (2001-02) and ACE-2003 data
- ✓ MUC7 Formal Run test set

Experiments

실험과정

각 corpus를 소문자로 전환하고 token화를 진행한다. 빈도를 기준으로 400,000개의 단어를 선정하고 co-occurrence counts X를 생성한다.

이후 window size, 고려할 단어의 방향을 결정하고 weighting function을 이용하여 단어의 거리를 기준으로 가중치를 부여한다.

실험에서 사용한 파라미터

- ✓ X_{max}=100
- ✓ a=¾
- ✓ AdaGrad
- ✓ Iteration: 300d 미만 50, 300d이상 100
- ✓ 좌측 및 우측으로 10개의 단어를 window로 지정

Results

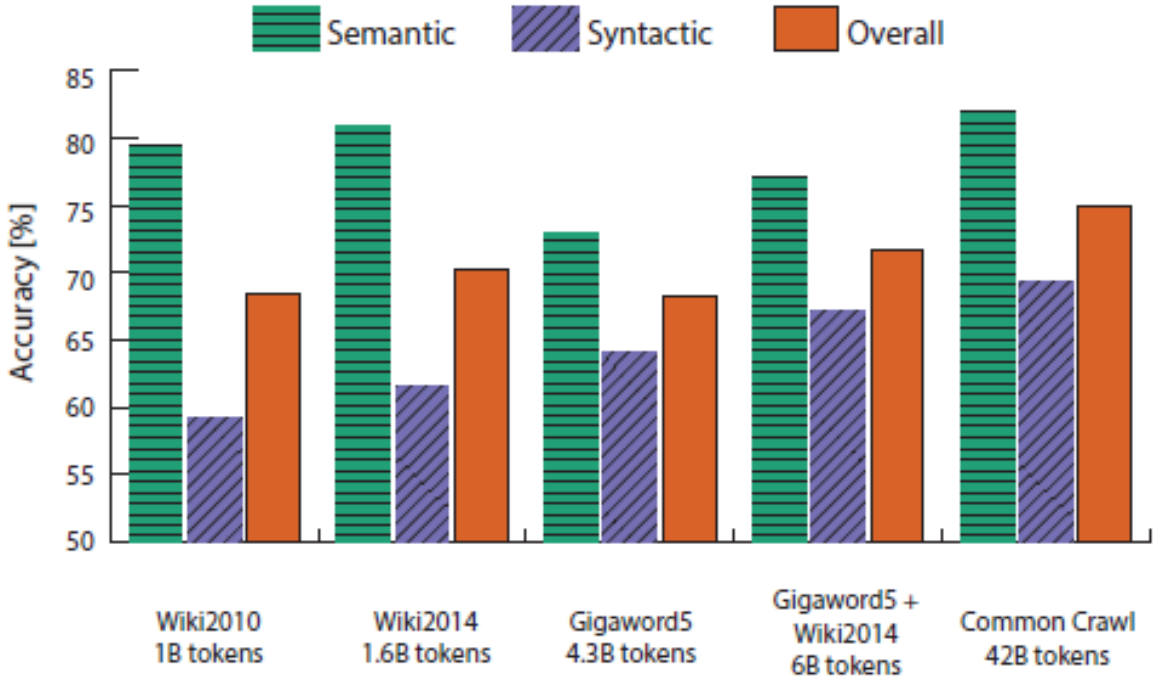
Word-similarity task에 대한 실험 결과이다.

Model	Size	WS353	MC	RG	SCWS	RW
SVD	6B	35.3	35.1	42.5	38.3	25.6
SVD-S	6B	56.5	71.5	71.0	53.6	34.7
SVD-L	6B	65.7	72.7	75.1	56.5	37.0
CBOW†	6B	57.2	65.6	68.2	57.0	32.5
SG†	6B	62.8	65.2	69.7	58.1	37.2
GloVe	6B	65.8	72.7	77.8	53.9	38.1
SVD-L	42B	74.0	76.4	74.1	58.3	39.9
GloVe	42B	**75.9**	**83.6**	**82.9**	**59.6**	**47.8**
CBOW*	100B	68.4	79.6	75.4	59.4	45.5

모든 벡터는 300 차원을 가지며 L은 큰 corpus에 대한 모형을 의미한다.

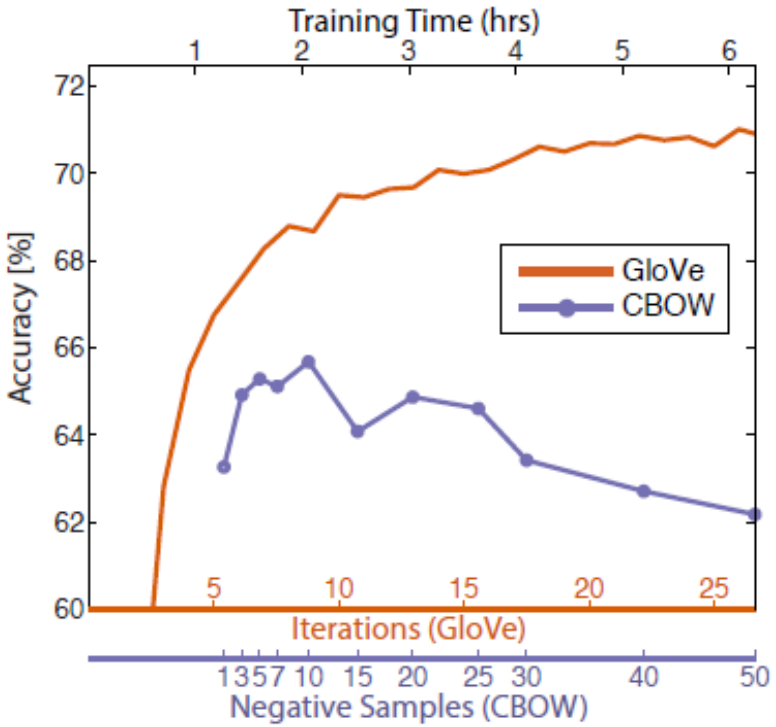
Results

Analogy task에 대해 서로 다른 corpora를 적용한 결과이다.

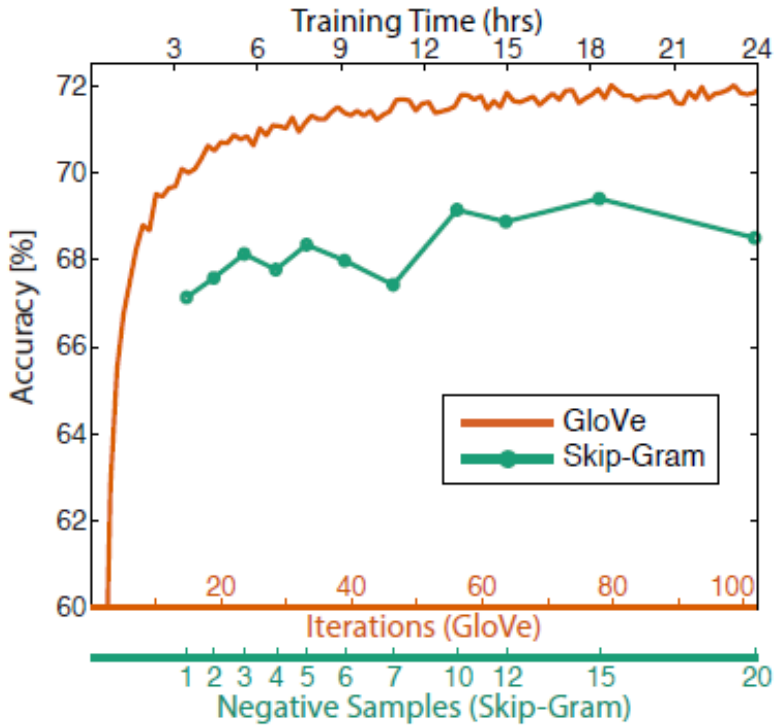


Results

CBOW와 Skip-Gram과의 반복횟수에 따른 시간과 정확도를 비교한 그래프이다.



(a) GloVe vs CBOW



(b) GloVe vs Skip-Gram