Understanding GloVe (Global Vectors for Word Representation)

Negar Nejatishahidin
CS 747

https://www.slideshare.net/JiHyunPark18/understanding-glove?from_action=save

Contents

1. Introduction

- The statistics of word occurrences in a corpus is the primary source of information available to all unsupervised methods for learning word representations.

- Although many such methods now exist, the question still remains as to how meaning is generated from these statistics, and how the resulting word vectors might represent that meaning.

1. Introduction

- Recent methods for learning vector space representations of words have succeeded in capturing fine-grained semantic and syntactic regularities using vector arithmetic,

- But the origin of these regularities has remained opaque.

1. Introduction

- Matrix Factorization Methods : methods that reduce a matrix into constituent parts that make it easier to calculate more complex matrix operations .

- Shallow Window-Based Methods: Another approach is to learn word representations that aid in making predictions within local context windows.

1. Introduction : pros & cons

Count based vs direct prediction

LSA, HAL (Lund & Burgess),
COALS (Rohde et al),
Hellinger-PCA (Lebret & Collobert)

- Fast training
- Efficient usage of statistics
- Primarily used to capture word similarity
- Disproportionate importance given to large counts

NNLM, HLBL, RNN, Skip-gram/CBOW, (Bengio et al; Collobert & Weston; Huang et al; Mnih & Hinton; Mikolov et al; Mnih & Kavukcuoglu)

- Scales with corpus size
- Inefficient usage of statistics
- Generate improved performance on other tasks
- Can capture complex patterns beyond word similarity

2. GloVe model

- Combines the advantages of the two major model families in the literature:
  - global matrix factorization and,
  - local context window methods

- Our model efficiently leverages statistical information by training only on the nonzero elements in a word-word co-occurrence matrix, rather than on the entire sparse matrix or on individual context windows in a large corpus.

Matrix of word-word co-occurrence counts

- I like deep learning.
- I like NLP.
- I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

3. GloVe cost function

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

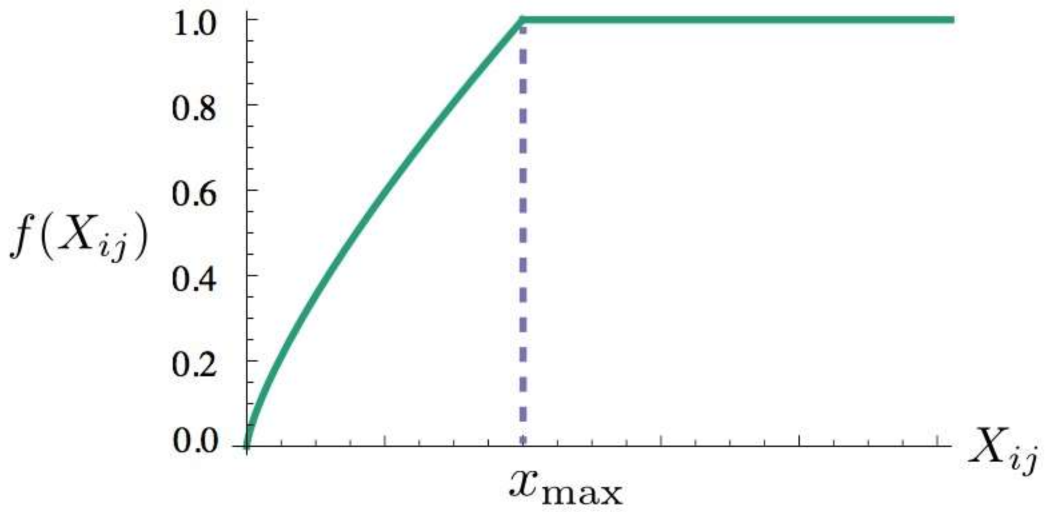


Figure 1: Weighting function f with $\alpha = 3/4$.

Matrix of word-word co-occurrence counts

- I like deep learning.
- I like NLP.
- I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

3. GloVe cost function

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

- Deriving the Cost Function
  - The above argument suggests that the appropriate starting point for word vector learning should be with ratios of co-occurrence probabilities rather than the probabilities themselves.

3. GloVe cost function

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

- Establish some notations
  - let the matrix of word-word co-occurrence counts be denoted by X,
  - whose entries X_{ij} tabulate the number of times word j occurs in the context of word i
  - let $X_i = \sum_k X_{ik}$ be the number of times any word appears in the context of word i
  - let $P_{ij} = P(i|j) = X_{ij}/X_i$ be the probability that word j appear in the context of word i

3. GloVe cost function

- Deriving the Cost Function
  - set a **function F** that represents ratios of co-occurrence probabilities rather than t he probabilities themselves

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

✓ Note that w_i and $\tilde{w}_k$ are v ectors from **different vector-spaces**

  - we would like F to encode the information present the ratio P_{ik}/P_{jk} in the word vector space. Since vector spaces are inherently linear structures, the most natural way to do this is with vector differences.

$$F(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}. \quad (2)$$

3. GloVe cost function

- Deriving the Cost Function
  - We note that the arguments of F in Eqn. (2) are **vectors** while the right-hand sid e is a **scalar**.
  - While F could be taken to be a **complicated function parameterized** by, e.g., a neural network, doing so would obfuscate the linear structure we are tryin g to capture.
  - To avoid this issue, we can first take the dot product of the arguments, which prevents F from mixing the vector dimensions in undesirable ways.

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

3. GloVe cost function

- Deriving the Cost Function
  - for word-word co-occurrence matrices, the distinction between a word and a context word is arbitrary and that we are free to exchange the two roles.

  - the symmetry can be restored in two steps.
  - First, we require that F be a homomorphism between the groups ($\mathbb{R}$,+) and ($\mathbb{R}$ >0, ×), i.e.,

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

  - which, by Eqn. (3), is solved by,

$$F(w_i^T \tilde{w}_k) = P_{ik} = \frac{X_{ik}}{X_i}. \qquad (5)$$

3. GloVe cost function : homomorphism

Homomorphisms

$G \; *$ $\qquad x, y \in G$ $\qquad f : G \rightarrow H$

$H \; \diamond$ $\qquad x * y = z$ $\qquad x \mapsto f(x)$

$y \mapsto f(y)$

$z \mapsto f(z)$

$$x * y = z \Rightarrow f(x) \diamond f(y) = f(z)$$

$$\Rightarrow \boxed{f(x) \diamond f(y) = f(x * y)}$$

3. GloVe cost function : homomorphism

Example 1

$G = \mathbb{R}$ under $+$

abelian, identity $= 0$

$H = \mathbb{R}^+$ under $\times$

abelian, identity $= 1$

$$f : G \longrightarrow H$$
$$x \longmapsto e^x$$

$$f(x+y) = f(x) \times f(y)$$
$$e^{x+y} = e^x \times e^y \quad \checkmark$$

Homo (same)
+
Morph (shape)

3. GloVe cost function

- Deriving the Cost Function
  - The solution to Eqn. (4) is $F = exp$ or,

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

  - Next, we note that Eqn. (6) would exhibit the exchange symmetry if not for the $\log(X_i)$ on the right-hand side.
  - However, this term is independent of k so it can be absorbed into a bias b_i for w_i.
  - Finally, adding an additional bias $\tilde{b}_i$ for $\tilde{w}_i$ restores the symmetry.

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik}) . \quad (7)$$

3. GloVe cost function : weighting function

- Deriving the Cost Function

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

1. $f(0) = 0$. If f is viewed as a continuous function, it should vanish as $x \to 0$ fast enough that the $\lim_{x\to 0} f(x) \log^2 x$ is finite.
2. $f(x)$ should be non-decreasing so that rare co-occurrences are not overweighted.
3. $f(x)$ should be relatively small for large values of x, so that frequent co-occurrences are not overweighted.

3. GloVe cost function : weighting function

- Deriving the Cost Function

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

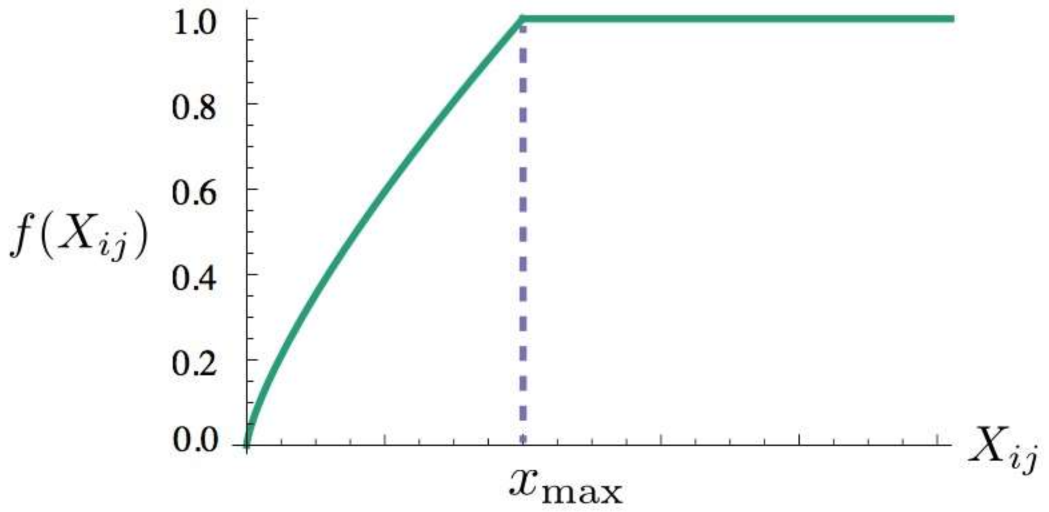


Figure 1: Weighting function f with $\alpha = 3/4$.

4. Experiments & Results

- Experiments
  - Word analogy
  - Word similarity
  - Named entity recognition (NER)

- Result

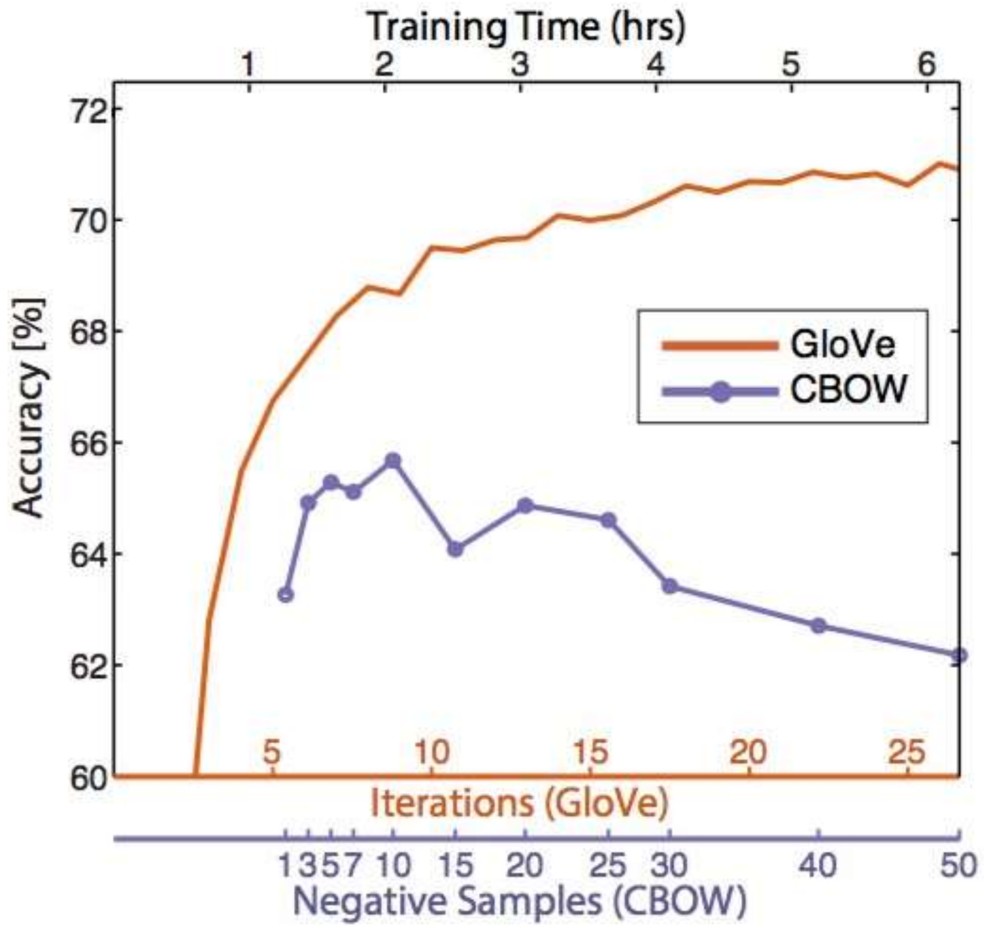
4. **Experiments** : analogy task

- The word analogy task consists of questions like, “a is to b as c is to ___?”
- The semantic questions, like “Athens is to Greece as Berlin is to ___?”.
- The syntactic questions like, “dance is to dancing as fly is to ___?”

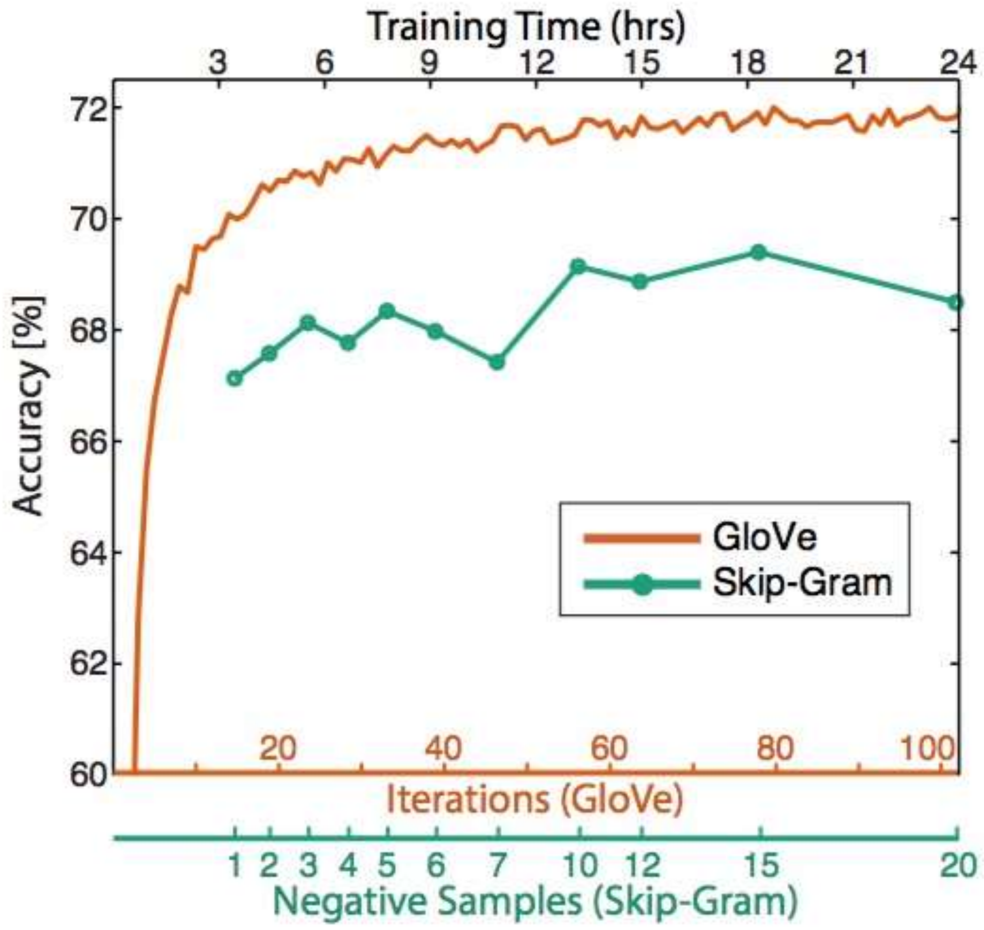
4. **Experiments** : analogy task

Model	Dim.	Size	Sem.	Syn.	Tot.
ivLBL	100	1.5B	55.9	50.1	53.2
HPCA	100	1.6B	4.2	16.4	10.8
GloVe	100	1.6B	67.5	54.3	60.3
SG	300	1B	61	61	61
CBOW	300	1.6B	16.1	52.6	36.1
vLBL	300	1.5B	54.2	64.8	60.0
ivLBL	300	1.5B	65.2	63.0	64.0
GloVe	300	1.6B	80.8	61.5	70.3
SVD	300	6B	6.3	8.1	7.3
SVD-S	300	6B	36.7	46.6	42.1
SVD-L	300	6B	56.6	63.0	60.1
CBOW†	300	6B	63.6	67.4	65.7
SG†	300	6B	73.0	66.0	69.1
GloVe	300	6B	77.4	67.0	71.7
CBOW	1000	6B	57.3	68.9	63.7
SG	1000	6B	66.1	65.1	65.6
SVD-L	300	42B	38.4	58.2	49.2
GloVe	300	42B	**81.9**	**69.3**	**75.0**

4. Experiments : analogy task



(a) GloVe vs CBOW



(b) GloVe vs Skip-Gram

4. Experiments : similarity task

- A similarity score is obtained from the word vectors by first normalizing each feature across the vocabulary and then calculating the cosine similarity.
- We compute Spearman's rank correlation coefficient between this score and the human judgments.

4. Experiments : similarity task

Glove results

Nearest words to frog:

1. frogs
2. toad
3. litoria
4. leptodactylidae
5. rana
6. lizard
7. eleutherodactylus

litoria

leptodactylidae

rana

eleutherodactylus

4. Experiments : similarity task

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	<u>72.7</u>	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	<u>58.1</u>	37.2
GloVe	6B	<u>65.8</u>	<u>72.7</u>	<u>77.8</u>	53.9	<u>38.1</u>
SVD-L	42B	74.0	76.4	74.1	58.3	39.9
GloVe	42B	**<u>75.9</u>**	**<u>83.6</u>**	**<u>82.9</u>**	**<u>59.6</u>**	**<u>47.8</u>**
CBOW*	100B	68.4	79.6	75.4	59.4	45.5

4. Experiments : NER task

- The CoNLL-2003 English benchmark dataset for NER is a collection of documents from Reuters newswire articles, annotated with four entity types:
  - Person
  - Location
  - Organization
  - Miscellaneous
- We train models on CoNLL-03 training data on test on three datasets:
  1) ConLL-03 testing data
  2) ACE Phase 2 (2001-02) and ACE-2003 data
  3) MUC7 Formal Run test set.

4. Experiments : NER task

Model	Dev	Test	ACE	MUC7
Discrete	91.0	85.4	77.4	73.4
SVD	90.8	85.7	77.3	73.7
SVD-S	91.0	85.5	77.6	74.3
SVD-L	90.5	84.8	73.6	71.5
HPCA	92.6	**88.7**	81.7	80.7
HSMN	90.5	85.7	78.7	74.7
CW	92.2	87.4	81.7	80.2
CBOW	93.1	88.2	82.2	81.1
GloVe	**93.2**	88.3	**82.9**	**82.2**

4. Result

- GloVe, is a new global log-bilinear regression model for the unsupervised learning of word representations that outperforms other models on word analogy, word similarity, and named entity recognition tasks.

5. References

- GloVe
  - https://nlp.stanford.edu/projects/glove/
- Stanford NLP lecture
  - http://web.stanford.edu/class/cs224n/
  - https://www.youtube.com/watch?v=ASn7ExxLZws&t=43s
- Socratica's video about homomorphism
  - https://www.youtube.com/watch?v=cYzp5IWqCsg