GloVe: Global Vectors for Word Representation

최희정

0 Abstract

- The result is a new global log-bilinear regression model that 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.

1 Introduction

- The two main model families for learning word vectors are:
  1) global matrix factorization methods, such as latent semantic analysis (LSA) (Deerwester et al., 1990)
  2) local context window methods, such as the skip-gram model of Mikolov et al. (2013c).

- Currently, both families suffer significant drawbacks.
  1) While methods like LSA efficiently leverage statistical information, they do relatively poorly on the word analogy task, indicating a sub-optimal vector space structure.
  2) Methods like skip-gram may do better on the analogy task, but they poorly utilize the statistics of the corpus since they train on separate local context windows instead of on global co-occurrence counts.

- Global log-bilinear regression models
  : We propose a specific weighted least squares model that trains on global word-word co-occurrence counts and thus makes efficient use of statistics.

2 Related Work

2.1 Matrix Factorization Methods

- These methods utilize low-rank approximations to decompose large matrices that capture statistical information about a corpus.

- The Hyperspace Analogue to Language (HAL) (Lund and Burgess, 1996) utilizes matrices of "term-term" type, i.e., the rows and columns correspond to words and the entries correspond to the number of times a given word occurs in the context of another given word.

- A main problem with HAL and related methods is that the most frequent words contribute a disproportionate amount to the similarity measure:
  the number of times two words co-occur with the or and, for example, will have a large effect on their similarity despite conveying relatively little about their semantic relatedness.

- A number of techniques exist that addresses this shortcoming of HAL
  1) COALS method (Rohde et al., 2006), in which the co-occurrence matrix is first transformed by an entropy- or correlation-based normalization.
  2) Hellinger PCA (HPCA) (Lebret and Collobert, 2014), a square root type transformation

2 Related Work

2.2 Shallow Window-Based Methods

- Another approach is to learn word representations that aid in making predictions within local context windows.

- The skip-gram and continuous bag-of-words (CBOW) models of Mikolov et al. (2013a) propose a simple single-layer architecture based on the inner product between two word vectors.
  Mnih and Kavukcuoglu (2013) also proposed closely-related vector log-bilinear models, vLBL and ivLBL.

- Through evaluation on a word analogy task, these models demonstrated the capacity to learn linguistic patterns as linear relationships between the word vectors.

- Unlike the matrix factorization methods, the shallow window-based methods suffer from the disadvantage that they do not operate directly on the co-occurrence statistics of the corpus.
  Instead, these models scan context windows across the entire corpus, which fails to take advantage of the vast amount of repetition in the data.

3 The GloVe Model

- Notation

$\boldsymbol{X}$	: the matrix of word-word co-occurrence counts
$\boldsymbol{X_{ij}}$	: the number of times word j occurs in the context of word i
$\boldsymbol{X_i}$ $(= \sum_k X_{ik})$	: the number of times any word appears in the context of word i
$\boldsymbol{P_{ij}}$ $(= P(j\|i) = \frac{X_{ij}}{X_i})$	: the probability that word j appear in the context of word i

3 The GloVe Model

- The relationship of these words can be examined by studying the ratio of their co-occurrence probabilities with various probe words, k.

- the comparison between the probability that word i appear in the context of word k and that of word j

: $\boldsymbol{P_{jk}/P_{jk}} = \begin{cases} > 1 & ,word\ i\ is\ more\ related\ to\ word\ k \\ \approx 1 & ,both\ are\ related\ or\ unrelated\ to\ word\ k \\ < 1 & ,word\ j\ is\ more\ related\ to\ word\ k \end{cases}$

i

- Compared to the raw probabilities, the ratio is better able to distinguish relevant words($\boldsymbol{P_{jk}/P_{jk}} \neq 1$) from irrelevant words ($\boldsymbol{P_{jk}/P_{jk}} \approx 1$) and it is also better able to discriminate between the two relevant words.

3 The GloVe Model

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

- ice is more related to $solid$.
- $steam$ is more related to gas.
- Both ice and $steam$ are related or unrelated to $water$ and $fashion$.
- $solid$ and gas are relevant words but $water$ and $fashion$ are irrelevant words.

3 The GloVe Model

- The ratio, $\boldsymbol{P_{jk}/P_{jk}}$, depends on three words $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$.
  We would like F to encode the information present the ratio $\boldsymbol{P_{jk}/P_{jk}}$ in the word vector space.

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

- Since vector spaces are inherently linear structures, the most natural way to do this is with vector differences.

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

- We note that the arguments of $\boldsymbol{F}$ in Eqn. (2) are vectors while the right-hand side is a scalar.
  To prevent $\boldsymbol{F}$ from mixing the vector dimensions in undesirable ways, we can first take the dot product of the arguments.

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

3 The GloVe Model

- For word-word co-occurrence matrices, the distinction between a word and a context word is arbitrary and we are free to exchange the two roles.
  To do so consistently, we must not only exchange $\boldsymbol{w} \leftrightarrow \tilde{\boldsymbol{w}}$ but also $\boldsymbol{X} \leftrightarrow \boldsymbol{X^T}$.

1) First, we require that $\boldsymbol{F}$ be a homomorphism between the groups $(R, +)$ and $(R > 0, \times)$

→ $\boldsymbol{F(a+b) = F(a) \times F(b)}$

→ $\boldsymbol{F = exp}$ or $w_i^T \tilde{w}_k = \log(P_{ik}) = \log(X_{ik}) - \log(X_i)$. (6)

$$F(w_i^T \tilde{w}_k) = P_{ik} = \frac{X_{ik}}{X_i}. \quad (5) \longrightarrow F\left((w_i - w_j)^T \tilde{w}_k\right) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}, \quad (4)$$

3 The GloVe Model

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

2) Next, we note that Eqn. (6) would exhibit the exchange symmetry if not for the log(Xi) 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 $\widetilde{b_k}$ for $\widetilde{w_k}$ restores the symmetry.

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

- However, it is actually ill-defined since the logarithm diverges whenever its argument is zero.
One resolution to this issue is to include an additive shift in the logarithm which maintains the sparsity of X while avoiding the divergences.

$$\log(X_{ik}) \rightarrow \log(1 + X_{ik})$$

3 The GloVe Model

- A main drawback of model is that it weighs all co-occurrences equally, even those that happen rarely or never. We propose a new weighted least squares regression model that addresses these problems.
  Casting Eqn. (7) as a least squares problem and introducing a weighting function $f(X_{ij})$ into the cost function gives us the model.

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

- The weighting function should obey the following properties:
  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 over-weighted.
  3. $f(x)$ should be relatively small for large values of x, so that frequent co-occurrences are not over-weighted.

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

3 The GloVe Model

3.1 Relationship to Other Models

- The starting point for the skip-gram or ivLBL methods is a model Q_{ij} for the probability that word j appears in the context of word i.

$$Q_{ij} = \frac{\exp(w_i^T \tilde{w}_j)}{\sum_{k=1}^{V} \exp(w_i^T \tilde{w}_k)} . \qquad (10)$$

- They attempt to maximize the log probability as a context window scans over the corpus.

$$J = - \sum_{\substack{i \in \text{corpus} \\ j \in \text{context}(i)}} \log Q_{ij} . \qquad (11)$$

3 The GloVe Model

- However, the sum in Eqn. (11) can be evaluated much more efficiently if we first group together those terms that have the same values for i and j.

$$J = -\sum_{\substack{i \in \text{corpus} \\ j \in \text{context}(i)}} \log Q_{ij} \,. \qquad (11) \longrightarrow J = -\sum_{i=1}^{V}\sum_{j=1}^{V} X_{ij} \log Q_{ij} \,, \qquad (12)$$

- Recalling our notation $X_i = \sum_k X_{ik}$ and $P_{ij} = \frac{X_{ij}}{X_i}$, we can rewrite J with $H(P_i, Q_i)$, the cross entropy of the distributions P_i and Q_i.

$$J = -\sum_{i=1}^{V} X_i \sum_{j=1}^{V} P_{ij} \log Q_{ij} = \sum_{i=1}^{V} X_i H(P_i, Q_i), \qquad (13)$$

$$\left(\because -\sum_{j=1}^{V} P_{ij} \log Q_{ij} = E_p(-\log Q_{ij}) \right)$$

3 The GloVe Model

- Furthermore, for the measure to be bounded it requires that the model distribution Q be properly normalized. This presents a computational bottleneck owing to the sum over the whole vocabulary in Eqn. (10), and it would be desirable to consider a different distance measure that did not require this property of Q.

- A natural choice would be a least squares objective in which normalization factors in Q and P are discarded.

$$\hat{J} = \sum_{i,j} X_i (\hat{P}_{ij} - \hat{Q}_{ij})^2 \qquad (14) \qquad \text{where } \hat{P}_{ij} = X_{ij} \text{ and } \hat{Q}_{ij} = \exp(w_i^T \tilde{w}_j)$$

- X_{ij} often takes very large values, which can complicate the optimization.
An effective remedy is to minimize the squared error of the logarithms of $\hat{P}$ and $\hat{Q}$ instead.

$$\hat{J} = \sum_{i,j} X_i (\log \hat{P}_{ij} - \log \hat{Q}_{ij})^2 = \sum_{i,j} X_i (w_i^T \tilde{w}_j - \log X_{ij})^2 . \qquad (15)$$

3 The GloVe Model

- In fact, Mikolov et al. (2013a) observe that performance can be increased by filtering the data so as to reduce the effective value of the weighting factor for frequent words.
  With this in mind, we introduce a more general weighting function, which we are free to take to depend on the context word as well.

$$\hat{J} = \sum_{i,j} f(X_{ij}) (w_i^T \tilde{w}_j - \log X_{ij})^2 , \qquad (16)$$

- The result is equivalent to the cost function of Eqn. (8), which we derived previously.

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

(8)

3 The GloVe Model

3.2 Complexity of the model

- As can be seen from Eqn. (8) and the explicit form of the weighting function $f(X)$, the computational complexity of the model depends on the number of nonzero elements in the matrix X.
  As this number is always less than the total number of entries of the matrix, the model scales no worse than $O(|V|^2)$.

- However, typical vocabularies have hundreds of thousands of words, so that $|V|^2$ can be in the hundreds of billions, which is actually much larger than most corpora.
  For this reason it is important to determine whether a tighter bound can be placed on the number of nonzero elements of X.

3 The GloVe Model

- In order to make any concrete statements about the number of nonzero elements in X, it is necessary to make some assumptions about the distribution of word co-occurrences.
  In particular, we will assume that the number of co-occurrences of word i with word j, X_{ij}, can be modeled as a power-law function of the frequency rank of that word pair, r_{ij}.

$$X_{ij} = \frac{k}{(r_{ij})^{\alpha}} . \qquad (17)$$

- The total number of words in the corpus is proportional to the sum over all elements of the co-occurrence matrix X.

$$|C| \sim \sum_{ij} X_{ij} = \sum_{r=1}^{|X|} \frac{k}{r^{\alpha}} = kH_{|X|,\alpha} , \qquad (18)$$

note) $H_n = \sum_{k=1}^{n} \frac{1}{k} = 1+\frac{1}{2}+\frac{1}{3}+\ldots+\frac{1}{n}$

4 Experiments

4.1 Evaluation methods

- Word analogies
  : We answer the question “a is to b as c is to ?” by finding the word d whose representation w_d is closest to $w_b - w_a + w_c$ according to the cosine similarity.

- Word similarity
  : We also evaluate our model on a variety of word similarity tasks.

- Named entity recognition (NER)
  : a collection of documents from Reuters newswire articles, annotated with four entity types
  - person, location, organization, and miscellaneous.

4 Experiments

4.2 Corpora and training details

- We trained our model on five corpora of varying sizes.

- We tokenize and lowercase each corpus with the Stanford tokenizer, build a vocabulary of the 400,000 most frequent words, and then construct a matrix of cooccurrence counts X.
  In constructing X, we must choose how large the context window should be and whether to distinguish left context from right context.

- In all cases we use a decreasing weighting function, so that word pairs that are d words apart contribute 1/d to the total count.
  This is one way to account for the fact that very distant word pairs are expected to contain less relevant information about the words' relationship to one another.

4 Experiments

- For all our experiments, we set x_{max} = 100, α = 3/4, and train the model using AdaGrad(Duchi et al., 2011), stochastically sampling nonzero elements from X, with initial learning rate of 0.05.
  We run 50 iterations for vectors smaller than 300 dimensions, and 100 iterations otherwise (see Section 4.6 for more details about the convergence rate).

- The model generates two sets of word vectors, $\boldsymbol{w}$ and $\widetilde{\boldsymbol{w}}$. When X is symmetric, $\boldsymbol{w}$ and $\widetilde{\boldsymbol{w}}$ are equivalent and differ only as a result of their random initializations; the two sets of vectors should perform equivalently.

- On the other hand, there is evidence that for certain types of neural networks, training multiple instances of the network and then combining the results can help reduce overfitting and noise and generally improve results (Ciresan et al., 2012).
  With this in mind, we choose to use the sum $\boldsymbol{w} + \widetilde{\boldsymbol{w}}$ as our word vectors.
  Doing so typically gives a small boost in performance, with the biggest increase in the semantic analogy task.

4 Experiments

4.3 Results

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**

[Word analogies]

- We present results on the word analogy task in Table2.
- The GloVe model performs significantly better than the other baselines, often with smaller vector sizes and smaller corpora.

4 Experiments

Table 3: Spearman rank correlation on word similarity tasks. All vectors are 300-dimensional. The CBOW* vectors are from the `word2vec` website and differ in that they contain phrase vectors.

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

[Word similarity]

- Table 3 shows results on five different word similarity datasets.

- A similarity score is obtained from the word vectors by first normalizing each feature across the vocabulary and then calculating the cosine similarity.

4 Experiments

Table 4: F1 score on NER task with 50d vectors. *Discrete* is the baseline without word vectors. We use publicly-available vectors for HPCA, HSMN, and CW. See text for details.

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**

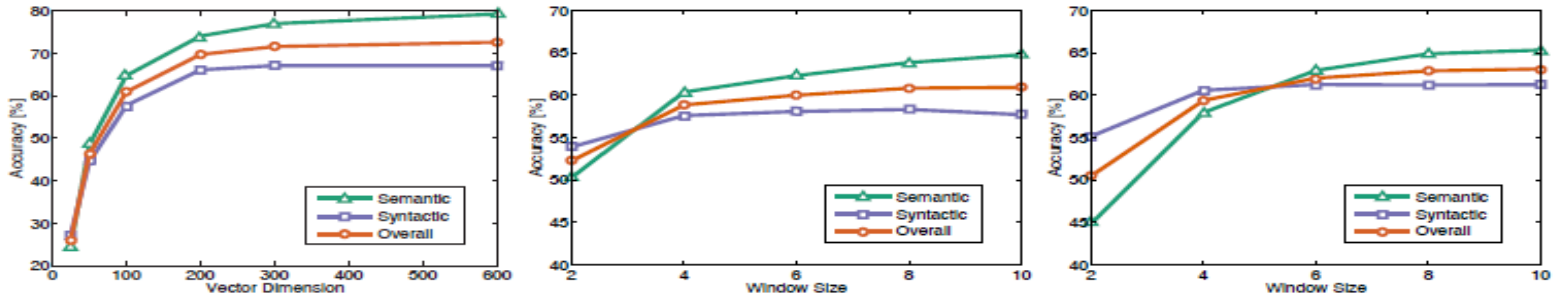
[NER]

- Table 4 shows results on the NER task with the CRF-based model.

- The GloVe model outperforms all other methods on all evaluation metrics, except for the CoNLL test set, on which the HPCA method does slightly better.

4 Experiments

4.4 Model Analysis: Vector Length and context Size



(a) Symmetric context (b) Symmetric context (c) Asymmetric context

Figure 2: Accuracy on the analogy task as function of vector size and window size/type. All models are trained on the 6 billion token corpus. In (a), the window size is 10. In (b) and (c), the vector size is 100.

- In (a), we observe diminishing returns for vectors larger than about 200 dimensions. In (b) and (c), we examine the effect of varying the window size for symmetric and asymmetric context windows.

- Performance is better on the syntactic subtask for small and asymmetric context windows, which aligns with the intuition that syntactic information is mostly drawn from the immediate context and can depend strongly on word order.

- Semantic information, on the other hand, is more frequently non-local, and more of it is captured with larger window sizes.

4 Experiments

4.5 Model Analysis: Corpus Size

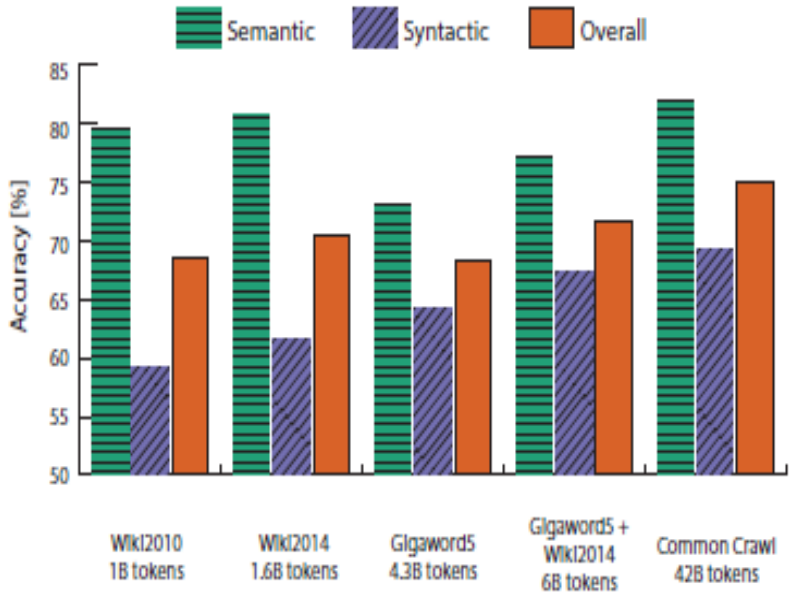


Figure 3: Accuracy on the analogy task for 300-dimensional vectors trained on different corpora.

- On the syntactic subtask, there is a monotonic increase in performance as the corpus size increases.
  This is to be expected since larger corpora typically produce better statistics.

- Interestingly, the same trend is not true for the semantic subtask, where the models trained on the smaller Wikipedia corpora do better than those trained on the larger Gigaword corpus.

4 Experiments

4.6 Model Analysis: Run-time

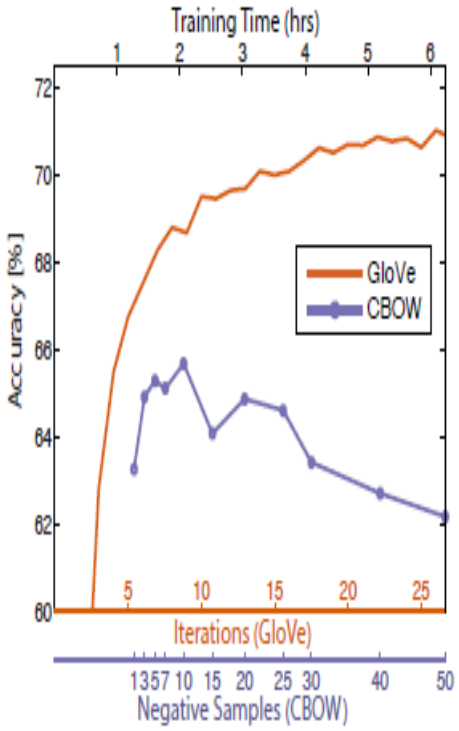
- The total run-time is split between populating X and training the model. The former depends on many factors, including window size, vocabulary size, and corpus size.

- Using a single thread of a dual 2.1GHz Intel Xeon E5-2658 machine, populating X with a 10 word symmetric context window, a 400,000 word vocabulary, and a 6 billion token corpus takes about 85 minutes.

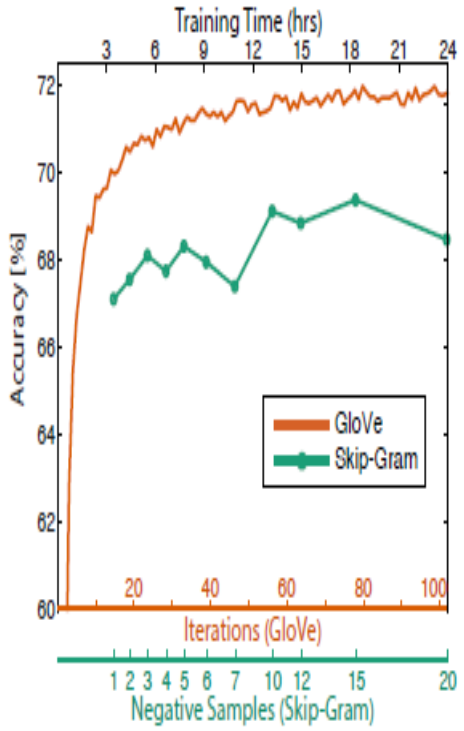
- For 300-dimensional vectors with the above settings (and using all 32 cores of the above machine), a single iteration takes 14 minutes.

4 Experiments

4.7 Model Analysis: Comparison with word2vec



(a) GloVe vs CBOW



(b) GloVe vs Skip-Gram

- In Fig. 4, we plot the overall performance on the analogy task as a function of training time. The two x-axes at the bottom indicate the corresponding number of training iterations for GloVe and negative samples for word2vec.

- We note that word2vec's performance actually decreases if the number of negative samples increases beyond about 10. Presumably this is because the negative sampling method does not approximate the target probability distribution well.

- For the same corpus, vocabulary, window size, and training time, GloVe consistently outperforms word2vec.

5 Conclusion

- We construct a model that utilizes this main benefit of count data while simultaneously capturing the meaningful linear substructures prevalent in recent log-bilinear prediction-based methods like word2vec.

- The 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.