Linking GloVe with `word2vec`

Tianze Shi and Zhiyuan Liu

`stz11@mails.tsinghua.edu.cn, liuzy@tsinghua.edu.cn`

November 20, 2014

The Global Vectors for word representation (GloVe), introduced by Jeffrey Pennington et al. [3][1] is reported to be an efficient and effective method for learning vector representations of words. State-of-the-art performance is also provided by skip-gram with negative-sampling (SGNS) [2] implemented in the `word2vec` tool[2].

In this note, we explain the similarities between the training objectives of the two models, and show that the objective of SGNS is similar to the objective of a specialized form of GloVe, though their cost functions are defined differently.

1   Introduction and Notation

By representing words as vectors, similarities between words and other valuable features can be calculated directly with vector arithmetics. The goal of word embedding algorithms is to find vectors for the words and their contexts in the corpus to meet some pre-defined criterion (e.g. to predict the surrounding context of a given word), where the contexts are often defined as the words surrounding a given word.

Let the word and context vocabularies be V_W and V_C respectively. For each word $w \in V_W$ and each context $c \in V_C$, the goal is to find a vector $\vec{w} \in \mathbb{R}^d$ and $\vec{c} \in \mathbb{R}^d$, where d denotes the vector dimension. Embeddings of all words in the vocabulary can be combined into a $\|V_W\| \times d$ matrix W, with the ith row W_i being the embedding of the ith word in the vocabulary. Similarly, a $\|V_C\| \times d$ matrix C gathers all the embeddings of the contexts, with C_j representing the embedding of the jth context.

Word-context pairs are denoted as (w, c), and $\#(w, c)$ counts all observations of (w, c) from the corpus. We use $\#(w) = \Sigma_c \#(w, c)$ and $\#(c) = \Sigma_w \#(w, c)$ to refer to the count of occurrences of a word (context) in all word-context pairs. Either $\Sigma_c \#(c)$ or $\Sigma_w \#(w)$ may represent the count of all word-context pairs.

---

[1] `http://nlp.stanford.edu/projects/glove/`
[2] `https://code.google.com/p/word2vec/`

2 Training Objectives of the Two Models

2.1 GloVe

GloVe explicitly factorizes the word-context co-occurrence matrix. The following equation gives the local cost function of GloVe model.

$$l_G\left(w_i, c_j\right) = f\left(\#\left(w_i, c_j\right)\right)\left(W_i \cdot C_j^T + b_{W_i} + b_{C_j} - \log \#\left(w_i, c_j\right)\right)^2, \quad (1)$$

where b_{W_i} and b_{C_j} are the unknown bias terms only relevant to the words and contexts respectively. $f(x)$ is a weighting function which down-weights rare co-occurrences. The function chosen by Pennington et al. is

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

The cost function is minimized by optimizing $W_i \cdot C_j^T$ to $\log \#\left(w_i, c_j\right) - b_{W_i} - b_{C_j}$, an ideal solution to which is given by

$$W_i \cdot C_j^T = \log \#\left(w_i, c_j\right) - b_{W_i} - b_{C_j} \quad (3)$$

for each row in W and C.

2.2 Skip-gram with Negative Sampling (SGNS)

As shown by Levy and Goldberg [1], SGNS implicitly factorizes a word-context matrix, whose cells are the shifted point-wise mutual information (PMI). The local objective for a given word-context pair is

$$l_S\left(w_i, c_j\right) = \#\left(w_i, c_j\right) \log \sigma\left(W_i \cdot C_j^T\right) + k \cdot \#\left(w_i\right) \cdot \frac{\#\left(c_j\right)}{\Sigma_w \#(w)} \log \sigma\left(-W_i \cdot C_j^T\right), \quad (4)$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ and k is the number of "negative" samples.

To optimize the objective, we find its partial derivative with respect to $x := W_i \cdot C_j^T$ and compare it to zero:

$$\frac{\partial l_S}{\partial x} = \#\left(w_i, c_j\right) \sigma\left(-x\right) - k \cdot \#\left(w_i\right) \cdot \frac{\#\left(c_j\right)}{\Sigma_w \#(w)} \sigma(x) = 0. \quad (5)$$

This equation is solved by

$$\begin{aligned} W_i \cdot C_j^T &= PMI\left(w_i, c_j\right) - \log k \\ &= \log \#(w_i, c_j) - \log \#(w_i) - \log \#(c_j) + \log \Sigma_w \#(w) - \log k. \end{aligned} \quad (6)$$

2.3 Similarities between the Two Objectives

By comparing Equation 3 and 6, we find that they show somewhat similar forms. The $\log \#(w_i)$ and $\log \#(c_j)$ terms in Equation 6 can be absorbed into the bias terms b_{W_i} and b_{C_j} respectively, and the $\log \Sigma_w \#(w) - \log k$ term is independent of i and j and can be viewed as a global bias term, which may be divided into the word and context bias terms.

The bias terms in the GloVe objective function are unknown and are to be determined by matrix factorization algorithms. They may or may not converge to the values given in the SGNS objective function. From this perspective, the GloVe model is more general and has a wider domain for optimization.

2.4 Differences between the Two Objectives

The GloVe model and the SGNS model are different in the following two aspects.

First, they define different cost functions though they share similar objectives, which may affect the performance when the vector dimensionality is not high enough.

They are also different in weighting strategies. With a well-chosen weighting function $f(x)$, the GloVe model down-weights the significance of rare word-context pairs and pays no attention to the unobserved pairs. Explicitly expressed in "negative-sampling", the SGNS model gains its success by assuming that randomly-chosen word-context pairs takes little or even no appearance in the corpus. Meanwhile, Levy and Goldberg [1] also point out that rare words are down-weighted in SGNS's objective shown in Equation 4.

The choice of weighting function $f(x)$ neglecting the unobserved word-context pairs is for the sake of efficiency and also avoiding the appearance of undefined $\log(0)$. Whether defining an objective for the unobserved word-context pairs and taking advantage of the "negative-sampling" can improve the performance remains an open question.

3 Observations of the Bias Terms in the GloVe Model

Curious about the optimized values of the bias terms in the GloVe model and the validity of "fixing the bias terms" to be $\log \#(w_i)$ and $\log \#(c_j)$ in the SGNS model, we observe the trained bias terms in GloVe and compare them to the fixed term in SGNS.

We train the GloVe model on a Wikipedia dump with 1.5 billion tokens and build a vocabulary of words occurring no less than 100 times in the corpus. We set x_{max} to be 10 or 100 and α to be 3/4 in the weighting function $f(x)$. Word-context pairs are counted symmetrically using the same techniques given by [3]. We run 50 iterations to train 300-dimensional vectors for words and contexts.

Figure 1 shows the Pearson correlation between b_{W_i} and $\log \#(w_i)$, between b_{C_j} and $\log \#(c_j)$ and between $b_{W_i} + b_{C_j}$ and $\log \#(w_i) + \log \#(c_j)$ with different
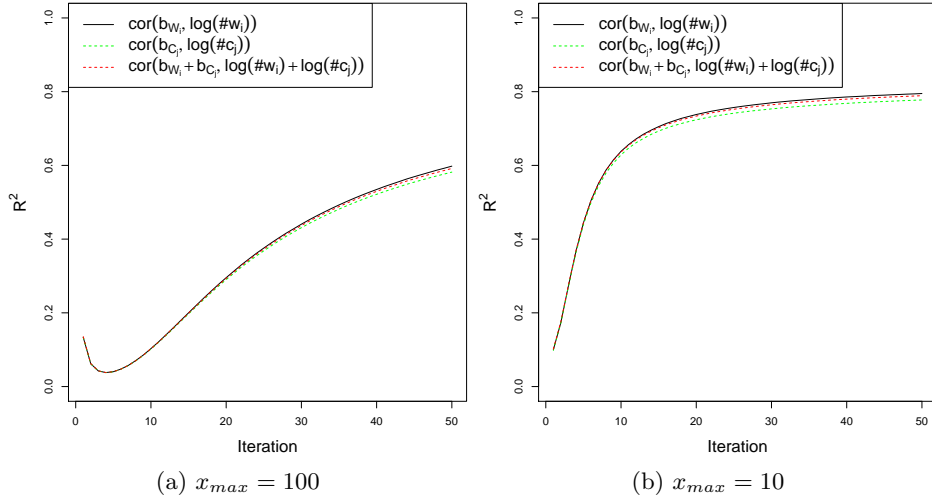
Figure 1: Pearson correlation coefficient R^2 as a function of the number of iterations.

x_{max} values.

Figure 2 depicts the distribution of b_{W_i} with respect to $\log \#(w_i)$ after the first iteration and after all 50 iterations. The two bands in the graph may be due to truncation of less frequent words.

We see that b_{W_i} correlates well to $\log \#(w_i)$ after 50 iterations, and that less weighting effect (with smaller x_{max}) results in a higher correlation. Though not explicitly written in the objective function, GloVe is actually optimizing $W_i \cdot C_j^T$ towards a shifted-PMI, just like what is done in the SGNS model.

4   Discussion

We show that interestingly, GloVe and SGNS, one explicitly factorizing a co-occurrence matrix and one implicitly factorizing a shifted-PMI matrix, are actually sharing similar objectives, though not completely the same. The training objective of SGNS is similar to the one of a specialized form of GloVe. Their differences mainly come from different cost functions and weighting strategies. Further we observe that in empirical experiments, the bias terms in the GloVe model tend to converge toward the corresponding terms in the SGNS model. We suppose that this may be a good approximation for the globally optimized value.

Future investigation may focus on the choices of the weighting function and their effect on the two models.
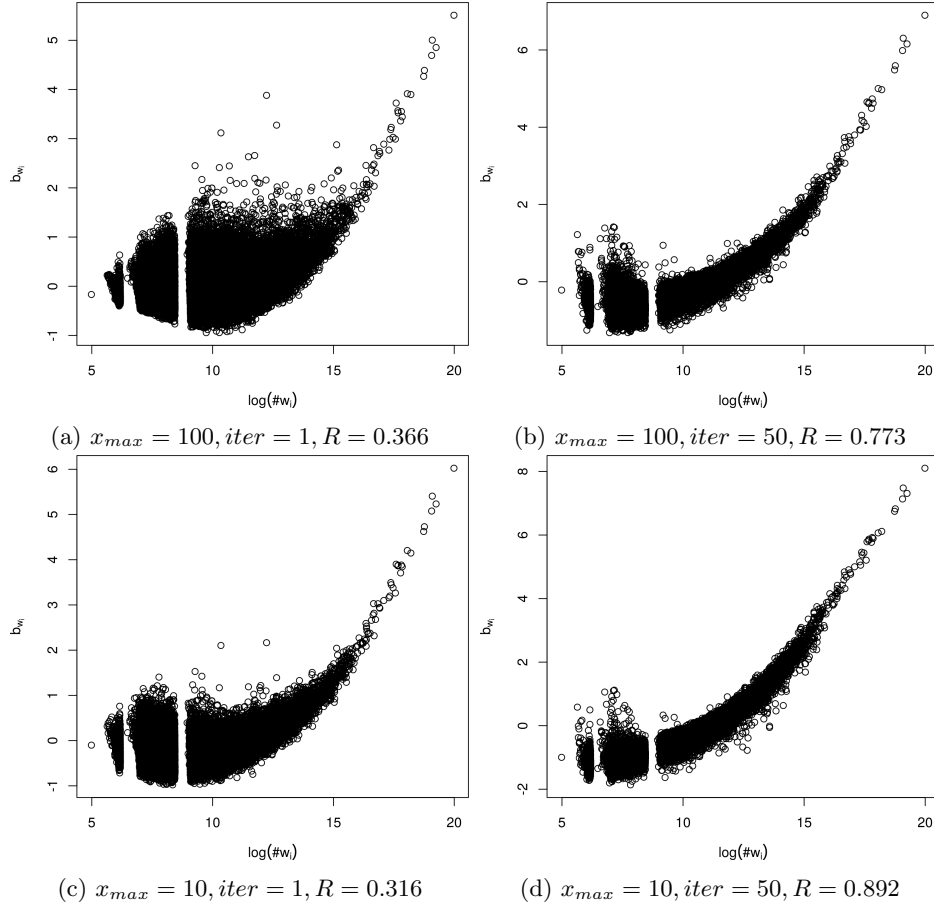
4

(a) $x_{max} = 100, iter = 1, R = 0.366$

(b) $x_{max} = 100, iter = 50, R = 0.773$

(c) $x_{max} = 10, iter = 1, R = 0.316$

(d) $x_{max} = 10, iter = 50, R = 0.892$

Figure 2: Distribution of b_{W_i} as a function of $\log \#(w_i)$ after the first iteration and after all 50 iterations. Pearson correlation coefficients R are given.

References

[1] Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2177–2185, 2014.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119, 2013.

[3] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.