A review on Word Embeddings: GloVe

Eunju Park(yello.ejp@gmail.com)

ABSTRACT

This paper introduces a word embedding method called GloVe collaborating count-based and distribution-based methods. In intrinsic and extrinsic evaluations, GloVe succeeds in capturing strengths of both word embedding methods showing better performance than other state-of-the-art word embedding methods. Nevertheless, it is too early to conclude that GloVe is the most effective word embedding method because a grounded method to evaluate the ability of word embeddings has not found yet. It is recommended to keep additional experiments on real world datasets from different domain until a solution for the evaluations is proved.

1. INTRODUCTION

Many methods have been introduced to represent features and contents of a document in NLP(Natural Language Processing) tasks. Recent vector space models for word representations have succeeded in capturing semantics and syntactics of languages. These representations are powerful tools widely used in NLP tasks, which can be applied to various applications such as information retrieval, document classification, question answering, named entity recognition and dependency parsing.

Word embedding methods deriving vector representations for words can be categorized into two parts depending on its approach: count-based method and distribution-based method. In Count-based method, a word is represented by co-occurrence of words in contexts. LSA(Latent Semantic Analysis) and HAL(Hyperspace Analogues to Language) efficiently use statistical information of a document and are believed to be fast in training. In spite of the strengths, it is poor in capturing relations between words and causes a disproportionate importance if a word occurs frequently. In terms of a bag of vocabulary, it is hard to represent an unseen word if it does not exist in the corpus. On the other hand, distribution-based methods believe that a representation of a word can be predicted from its neighbors. The popular methods such as Skip-gram and CBOW(Continuous Bag-Of-Words) seem to generate improved performance on a variety of NLP tasks such as word analogy and named entity recognition. Nevertheless, it predicts a word representation by using some neighbors' information instead of utilizing global statistical information, which is not efficient to deal with scalability with a corpus size

The GloVe(Global Vector for Word Representation) method takes advantages of count-based and distribution-based methods, which the drawbacks of count-based and distribution-based methods.

2. RELATED WORKS

2.1. Count-based method

Count-based methods utilize statistical information such as frequency of each word and co-occurrence of words in contexts. LSA method builds word-document matrix where rows correspond to words, columns correspond to documents and entries correspond to the number of occurrences of each word. Unlike LSA, HAL method builds word-word matrix where rows and columns correspond to words and entries correspond to the number of occurrences of a given word occurs in the context of another given word. However, the shortcoming of LSA and HAL is that the most frequent words have bigger weights comparing to other less frequent words causing a disproportionate. To address this shortcoming, normalized methods such as PPMI(Positive Pointwise Mutual Information) and HPCA(Hellinger PCA) have been suggested. In order to avoid the sparsity of matrix computed through above methods, it can be solved by reducing the original matrix into lower dimensionality matrix.

2.2. Count-based method

The idea of distribution-based methods started from averaging random vectors. It utilizes a given word or word's context to predict word representations. Skip-gram method predicts a word's context given the word itself while CBOW predict a word given its context. In CBOW, each word is represented by random vector, e.g., one-hot-vector, and the word's context is represented by the sum of vectors of words within the context window. At the end, a classifier predicts the target word. Skip-gram method is the other way around of CBOW method. Skip-gram is better for rare words because it has more opportunities to train the word while CBOW trains the word with context words once. But, CBOW is faster than Skip-gram. Instead of using one-hot vector to represent input words in Skip-gram and CBOW, vLBL and ivLBL methods use a normalized word-document matrix. The rest of the method architecture is same to Skip-gram and CBOW.

3. PROPOSED MODEL

GloVe is a word embedding method that leverages global co-occurrence and linear substructure for analogies. In order to obtain word embeddings, it requires several processing steps. First of all, it builds word-word co-occurrence count matrix X of a text corpus within a certain number of window-size. $X_{i,j}$ is the number of times word j appear in context with word i. $X_{i,.}$ is the number of contexts with word i. Each row or column vector denote a representation of a word in a corpus. On the left in figure 1, i, j and k denote words Germany, France and Paris. With assumption that word vector can be derived from function F regarding to co-occurrence ratio, the difference between words can be measured in a sense of analogy. On the right in figure 1, it can derive the relation between *Paris* and *Berlin* the relation between *France* and *Germany*. The difference between two words can be seen as analogy matter, which can be denoted like following formula.

$$F(w_i, w_j, \widetilde{w}_k) = \frac{Prob(k|i)}{Prob(k|j)} = \frac{x_{i,k}/x_i}{x_{j,k}/x_j}$$

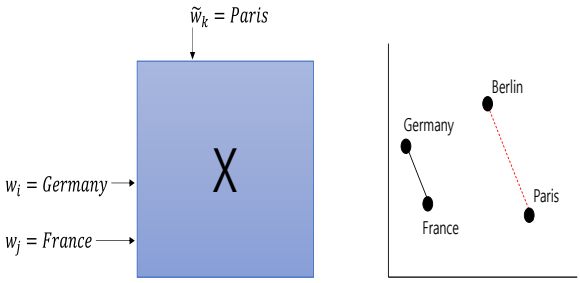


Figure 1. The difference between Germany and France is an analogy matter.

The original formula can be transformed with several steps like following.

$$F(w_i - w_j, \tilde{w}_k) \propto F\left((w_i - w_j)^T \tilde{w}_k\right)$$

$$\propto F(w_i^T \tilde{w}_k - w_j^T \tilde{w}_k)$$

$$= \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}$$

As exponential function satisfies the function F above, the final form can be derived like below by applying biases and adding constant to avoid potential problems.

$$w_i^T \tilde{w}_k + b_i + b_k = \log(X_{i,k} + 1)$$

Algorithm 1 describes the word embedding process. As mentioned in section 2.1, it applied weighting function to normalize the number of occurrences to dismiss the size effect of words which frequently occur in a corpus. By optimizing the objective function, it obtains two sets of word vectors w_t and $\widetilde{w_t}$. Although both sets should perform equivalently, it adopts using summation of w_t and $\widetilde{w_t}$. as finalized word embeddings for a small boost in performance with an evidence that combining results can help reducing overfitting and improve results.

Algorithm 1: Training GloVe

Input: a text corpus C; word tokens T; window-size N;

Output: Vector representations v_t ,

$\forall t \in T$

1. Build word-word co-occurrence count matrix X of C within N

2. $w_t \leftarrow X_{t,.}$, $\widetilde{w_t} \leftarrow X_{.,t}$, $\forall t \in T$;

3. $argmin_{w,\tilde{w}} = \sum_{i,j=1}^{T} f(X_{i,j})(w_i^T \widetilde{w_j} + b_i + \tilde{b_j} - \log(X_{i,j} + 1))^2$

$where\ f(X_{i,j}) = \begin{cases} \left(\frac{X_{i,j}}{\max(X_{i,j})}\right)^{\alpha} & , X_{i,j} < \max(X_{i,j}) \\ 1 & , otherwise \end{cases}$

4. $v_t \leftarrow w_t + \widetilde{w_t}$, $\forall t \in T$

4. EXPERIMENT AND RESULT

There are two mechanism to evaluate word embeddings: intrinsic evaluation and extrinsic evaluation. Intrinsic evaluation uses sets of words labeled manually by experts and compare judged datasets with word embeddings. On the other hand, extrinsic evaluations pay attention to the ability of word embeddings to be used in various downstream NLP tasks. Word similarity and word analogy for intrinsic evaluation and named entity recognition for extrinsic evaluation are used.

4.1. Word similarity

Word similarity between two words can be computed by cosine similarity. It uses datasets containing word pairs with similarity measured by experts. For example, it compares the cosine similarity

between the word Car and Train defined by GloVe to the similarity in the dataset.

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

Table 1. Comparison of accuracy on word similarity

Table 1 shows comparison of performance of different word embedding methods including GloVe on five different datasets. GloVe outperforms other methods in a different size of corpus.

4.2. Word analogy

Word analogy is based on the arithmetic operations of word embedding. It uses datasets containing a set of three words: w_a, w_b and w_c. It assumes that the vector of word w_d is described like following.

$$w_b - w_a + w_c$$

Datasets for word analogy tasks consist of semantic subset and syntactic subset. In a sentence "Paris is France as Berlin is to (unknown)", it computes vector operations of $w_{Paris} - w_{France} + w_{Berlin}$ to get the unknown word that should be *Germany* as an example of semantic tasks. A Syntactic task can be described a sentence "Fly is to flying as dance is to (unknown), which goal is to obtain a vector corresponding to *dancing.* For evaluation, it uses 3CosMul dataset providing 19,544 questions, divided into a semantic subset and a syntactic subset.

Table 2 shows that GloVe performs better than the other methods with small vector dimension and corpus.

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**

Table 2. Comparison of accuracy on analogy task

4.3. Named entity recognition

Named entity recognition is one of downstream NLP tasks. It identifies types of named entities such as names of people, organizations, etc. in a corpus. It uses prepared datasets having tags to distinguish named entities from normal words. For instance, the following sentence "[ORG U.N.] official [PER Ekeus] heads for [LOC Baghdad]" has three named entities. U.N. is an organization, Ekeus is a person and Baghdad is a location. By utilizing word embedding methods, it evaluates the ability of the embedding method whether it is effective on solving named entity recognition tasks. It trains the word embeddings with training set of CoNLL-2003 and then test the word embeddings on testing set of that, ACE and

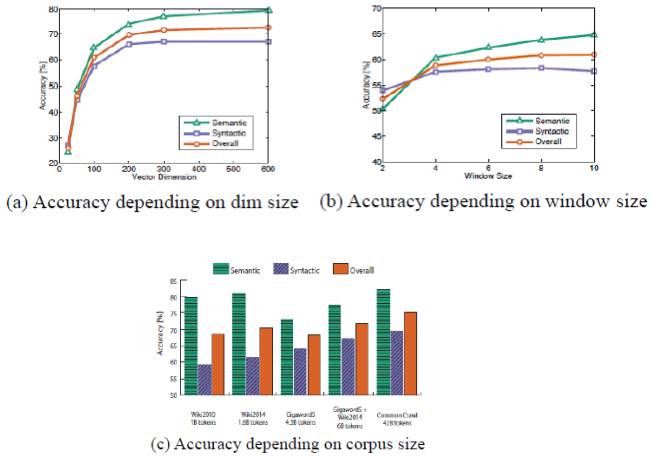
MUC7 datasets. Table 3 shows that GloVe model are useful in downstream NLP task.

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**

Table 3. Comparison of accuracy on named entity recognition

4.4. GloVe analysis

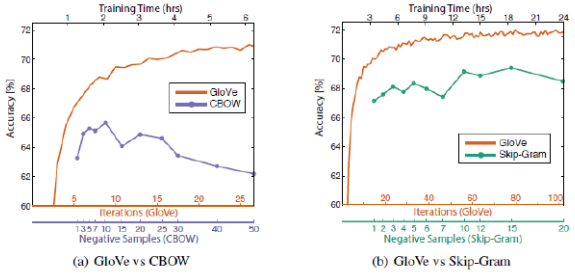
Several noticeable perspectives can be observed from the results of experiments. Although a large vector dimension shows better performance in analogy tasks, it diminishes the accuracy when the dimension is larger than about 200. In terms of influence of window size, it seems that a small windows size supports syntactic tasks well while a large window size works better on semantic tasks. Unlike the general expectation that training word embeddings with a large size of corpus large datasets would guarantee to improve the performance, the growing size of a corpus does not contribute to the performance of semantic tasks effectively. Comparing GloVe to CBOW and Skip-gram in accuracy and running time, the results are controlled by the number of iterations in GloVe and the number of negative samples in CBOW and Skip-gram. While the performance is improved in GloVe, it shows decreasing performance in CBOW and Skip-gram when the number of iterations or negative samples is decreasing. Regardless of the number of iterations, GloVe is faster. Moreover, it is getting accurate in analogy tasks.



(a) Accuracy depending on dim size (b) Accuracy depending on window size

(c) Accuracy depending on corpus size

Figure 2. Model analysis in analogy tasks



(a) GloVe vs CBOW (b) GloVe vs Skip-Gram

Figure 3. Comparison of GloVe vs CBOW and Skip-gram

According to experiments and results for comparison among different word embedding methods, it is proved that GloVe captures advantages of count-based and distribution-based methods, which addresses weaknesses of both methods. It is not only solving the drawbacks of count-based methods outperforming the distribution-based methods in accuracy but also solving weaknesses of distribution-based methods by achieving better performance with a large corpus in speed and accuracy. To sum up, GloVe outperforms other methods on word analogy, word similarity and entity recognition tasks.

5. DISCUSSION

According to the results discussed in section 4.4, the performance of GloVe is better than other methods. In spite of its obvious results, there are several things that should be considered before concluding the ability of GloVe method.

1. No grounded standard for evaluations of word embeddings. Training word representations is unsupervised learning. It means that the same word could be represented in different vector spaces depending on the word embedding methods.

2. Absence of correlation between intrinsic and extrinsic evaluations. The primary goal of intrinsic evaluations is to provide insights into word embedding to apply it for applications. However, the relation between two different evaluation approaches has not found yet.

3. Lack of intrinsic evaluations. Intrinsic evaluations can be categorized in many different methods: word similarity, word analogy, thematic fit, concept categorization, synonym detection, outlier word detection, etc. Among many evaluation methods, GloVe is checked only on word similarity and analogy tasks.

4. Weakness of intrinsic evaluations. Intrinsic evaluations use collections of word pairs with human judgements, which do not contain enough word pairs to be divided into training and testing sets. For instance, the largest dataset which the authors use has 2034 pairs. Considering the number of word pairs in each dataset in table 4, they evaluate GloVe with small datasets.

Dataset	Word pairs
WS-353	353
MC	30
RG	65
SCWS	2023
SimLex	999
RW	2034
MEN	3000
SimVerb-3500	3500

Table 4. Word similarity datasets

5. Lack of extrinsic evaluations. As the definition of extrinsic evaluations, the ability of GloVe method can be evaluated in many NLP task: named entity recognition, sentiment analysis, text classification, textual entailment detection, paraphrase detection, metaphor detection and so on. It seems not to be convincing concluding that GloVe outperforms the other models under only the result of named entity recognition tasks.

6. Expensive computing on generating the word-word co-occurrence matrix. Time consuming computing to generate the matrix is a weakness of count-based methods. Although GloVe addresses its drawbacks, it fails to solve its critical disadvantage.

7. Out-of-Vocabulary. To solve the tasks, it trains the word embedding with a certain corpus. It means that only words existing in the bag-of-word can be dealt with. If a document has a word not in the bag, it is hard to process the unseen word. Although it outperforms other methods in several tasks, it is not

suitable on online processing if unseen words occur often. In this case, other embedding methods such as character level embeddings should be considered.

6. CONCLUSION

The expensive computation for generating a matrix to train word embeddings in GloVe can be solved by distributed computing. Nevertheless, the biggest issue in evaluating word embedding methods is that there is no consensus on what the best way is to evaluate word embeddings. Until a solution is founded for grounded evaluation, word embedding methods should be evaluated how effectively it works well on task specific NLP problems.

7. REFERENCE

[1] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.

[2] Deerwester, Scott, et al. "Indexing by latent semantic analysis." Journal of the American society for information science 41.6 (1990): 391-407.

[3] Lund, Kevin, and Curt Burgess. "Producing high-dimensional semantic spaces from lexical co-occurrence." Behavior research methods, instruments, & computers 28.2 (1996): 203-208.

[4] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).

[5] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.

[6] Wang, Bin, et al. "Evaluating word embedding models: methods and experimental results." APSIPA Transactions on Signal and Information Processing 8 (2019).

[7] Nugaliyadde, Anupiya, et al. "Enhancing semantic word representations by embedding deep word relationships." Proceedings of the 2019 11th International Conference on Computer and Automation Engineering. 2019.

[8] Li, Bofang, et al. "Investigating different syntactic context types and context representations for learning word embeddings." Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017.

[9] Sang, Erik F., and Fien De Meulder. "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition." arXiv preprint cs/0306050 (2003).

[10] Bakarov, Amir. "A survey of word embeddings evaluation methods." arXiv preprint arXiv:1801.09536 (2018).

[11] Faruqui, Manaal, et al. "Problems with evaluation of word embeddings using word similarity tasks." arXiv preprint arXiv:1605.02276 (2016).

[12] Chiu, Billy, Anna Korhonen, and Sampo Pyysalo. "Intrinsic evaluation of word vectors fails to predict extrinsic performance." Proceedings of the 1st workshop on evaluating vector-space representations for NLP. 2016.