algorithms

Article

RGloVe: An Improved Approach of Global Vectors for Distributional Entity Relation Representation

Ziyan Chen [1,2,*], Yu Huang [2], Yuexian Liang [1,2], Yang Wang [2], Xingyu Fu [2] and Kun Fu [2]

[1] Department of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China; liangyuexian@126.com

[2] The Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China; huangyu@mail.ie.ac.cn (Y.H.); primular@163.com (Y.W.); iecasfxy@163.com (X.F.); kunfuiecas@gmail.com (K.F.)

[*] Correspondence: zychen0207@163.com; Tel.: +86-10-5888-7208-8931

Academic Editor: Evangelos Kranakis
Received: 10 January 2017; Accepted: 13 April 2017; Published: 17 April 2017

Abstract: Most of the previous works on relation extraction between named entities are often limited to extracting the pre-defined types; which are inefficient for massive unlabeled text data. Recently; with the appearance of various distributional word representations; unsupervised methods for many natural language processing (NLP) tasks have been widely researched. In this paper; we focus on a new finding of unsupervised relation extraction; which is called distributional relation representation. Without requiring the pre-defined types; distributional relation representation aims to automatically learn entity vectors and further estimate semantic similarity between these entities. We choose global vectors (GloVe) as our original model to train entity vectors because of its excellent balance between local context and global statistics in the whole corpus. In order to train model more efficiently; we improve the traditional GloVe model by using cosine similarity between entity vectors to approximate the entity occurrences instead of dot product. Because cosine similarity can convert vector to unit vector; it is intuitively more reasonable and more easily converge to a local optimum. We call the improved model RGloVe. Experimental results on a massive corpus of Sina News show that our proposed model outperforms the traditional global vectors. Finally; a graph database of Neo4j is introduced to store these relationships between named entities. The most competitive advantage of Neo4j is that it provides a highly accessible way to query the direct and indirect relationships between entities.

Keywords: distributional relation representation; co-occurrence matrix; Neo4j; global vectors; cosine similarity

1. Introduction

With the explosive growth and easy accessibility of web documents, extracting the useful nuggets from the irrelevant and redundant messages becomes a cognitively demanding and time consuming task. Under this circumstance, information extraction is proposed to extract the structured data from text documents. The automatic content extraction (ACE) program [1] provides annotated corpus and evaluation criteria for a series of information extraction tasks. As an important level of information extraction, relation extraction aims to extract the relationships between named entities. Relation extraction is widely used in many fields such as automatic construction of knowledge base and information retrieval.

Traditional relation extraction is often limited to extracting the pre-defined types. For example, ACE 2003 defines five relation types, including AT (location relationships), NEAR (to identify relative locations), PART (part-whole relationships), ROLE (the role a person plays in an organization) and

SOCIAL (such as parent and sibling), which are further separated into 24 relation sub-types. However, traditional relation extraction leaves the question open whether it is still efficient for massive and heterogeneous corpora such as web documents [2–4]. To combat these problems, many researches have been done to extract more abundant relations without requiring any specific relation type or a vocabulary such as verbs [5–8].

In light of the above, this paper aims to answer the following research questions:

RQ1 How to learn the distributional entity representations only by using the statistical information of entity-entity co-occurrences?

RQ2 How to build, store and query the relationships between entities without extracting the predefined relation types or vocabulary?

To answer above questions, we have made the following contributions in this paper. For RQ1, this paper presents an improved model of global vectors called RGloVe based on the idea of distributed representation. Global vectors (GloVe) [9] is an effective method to train distributional word representations from the global statistics of word occurrences in the whole corpus. In order to train model more efficiently, we improve the traditional GloVe model by using cosine similarity between entity vectors to approximate the entity occurrences instead of dot product. Because cosine similarity can convert vector to unit vector, it is intuitively more reasonable and more easily converge to a local optimum. For RQ2, instead of extracting relation types or a vocabulary, the task of distributional relation representation aims to extract a series of triples (e_1, e_2, ω). The weight ω is a real value which indicates the correlation of two entities e_1 and e_2. In order to store these triples and facilitate their retrival, we introduce a graph database of Neo4j where nodes represent the entities and edges represent the relationships between entities. The cypher query language of Neo4j provides a highly accessible way to query different levels of relationships (e.g., friends of a friend).

The rest of this paper is as follows. We review related work in traditional relation extraction and distributional word representation in Section 2. In Section 3, we will present all the details of Neo4j and RGloVe. Section 4 shows the experiment results of quantitative representation between named entities. Finally, we conclude our work and point out future work in Section 5.

2. Related Work

Our work is inspired by traditional entity relation extraction. Our model of RGloVe has its root in distributional word representation. In this section, we will briefly review some related works of these two aspects.

2.1. Entity Relation Extraction

Relation extraction, as an important level of information extraction, has been widely researched. These proposed methods can be classified into three categories: supervised learning, semi-supervised learning and unsupervised learning. The most typical methods of supervised learning are kernel-based methods [10–15]. Although supervised learning-based methods perform very well, their performance much relies on the availability of a large amounts of manually labeled data. So, many researchers begin to focus on the semi-supervised learning [16–19], which can make full use of unlabeled data given a small amount of labeled data.

Above works are often limited to extracting the pre-defined types, which make it difficult for open domain applications. Recently, many unsupervised learning-based methods [20–25] of open domain relation extraction have been proposed to reduce the heavy manual labor. TextRunner [5] was the first Open IE (OIE) system, where a large set of relational entity tuples were extracted without requiring any human labor and then these tuples were assigned a probability. Fader, Soderland and Etzioni [7] developed another Open IE system of REVERB by introducing two syntactic and lexical constraints on verb-centered relations. Tseng, Lee, Lin, Liao, Liu, Chen, Etzioni and Fader [8] presented the first Chinese Open IE system (CORE) which can extract entity relation triples from Chinese texts by combining many NLP techniques. Kalyanpur and Murdock [6] described entity-relation analysis in IBM Watson, which aimed to detect noun-centric-phrases in the text. Distributional relation representation is similar with unsupervised learning-based methods. Both of them aim to extract entity relations without requiring any specific relation types. Compared with previous works on Open IE, distributional relation representation focuses on the training of entity vectors instead of extracting more abundant features by introducing a series of NLP tools.

2.2. Distributional Word Representation

There are many effective methods of distributional word representations such as one-hot representation, latent semantic analysis and distributed representation. One-hot representation is a sparse word vector which dimension is equal to the size of the vocabulary. In the vector, there is only one 1 where the corresponding word appear in the vocabulary and a lot of zeroes. So one-hot representation often suffers from the curse of dimensionality. Another problem of one-hot representation is that it is hard to find the relationship between word vectors. To solve above problems, many researchers try to transform words into low dimensional semantic space. For example, Landauer, et al. [26] reported the results of using latent semantic analysis (LSA), a high-dimensional linear associative model to analyze a large corpus of natural text and generate a representation that captures the similarity of words and text. Turney [27] introduced Latent Relational Analysis (LRA), a method for measuring relational similarity which is correspondence between relations. Sebastian, et al. [28] presented a novel framework for constructing semantic spaces that takes syntactic relations into account. Gamallo, et al. [29] concluded that Singular Value Decomposition (SVD) is a more efficient model for a number of word similarity extraction tasks. Distributed representation is another effective low dimensional vector representation. For example, Bengio, et al. [30] combined n-gram model into a simple neural network architecture to learn distributional word representation. Collobert, et al. [31] presented a multilayer neural network architecture to learn distributional word representation in the window-based context of a word instead of the preceding context. Mikolov, et al. [32] proposed continuous bag-of-words model (CBOW) and continuous skip-gram model for learning distributional word representation. Pennington, Socher and Manning [9] proposed a global vector model by training only on the nonzero elements in co-occurrence matrix.

3. Methods

For the task of distributional relation representation, we propose an improved global vectors model called RGloVe which can train the word vectors more effectively. Finally, a graph database is introduced to build, store and query these extracted entity relationships.

3.1. Co-Occurrence Matrix

We use a preprocessing tool to extract all the named entities in the whole corpus. It is assumed that if entity i and entity j occur in the same document, these two entities will be regarded as co-occurrence. Let the co-occurrence matrix be denoted by X, whose element X_{ij} represents the co-occurrence frequency of entity i and entity j. X_{ij} can be computed as,

$$X_{ij} = \sum_{d=1}^{D} \frac{1}{\left| L_{di} - L_{dj} \right|} \tag{1}$$

where L_{di} is the location of entity i in a document d. It is an effective method to show that the more distant two entities in a document, the less relevant these entities.

3.2. Distributional Entityrepresentation of RGloVe

Without abundant features, the statistics of entity occurrences in a corpus is the primary source of information available to distributional relation representation between entities. Global vectors method has been proposed to train word vectors by efficiently leveraging statistical information. In order to train the entity vectors more efficiently for distributional entity relation representation, we make some improvements of global vectors. Firstly, RGloVe uses cosine similarity between entity vectors to approximate the entity occurrences instead of dot product in the traditional global vectors. Secondly, RGloVe reduces the weight funcation to linear function which value is limited to between 0 and 1. Finally, RGloVe train the entity vectors by AdaGrad [33].

3.2.1. Brief Review of Global Vectors

Global vectors aim to design a series of functions F, which are equivalent to the ratios of co-occurrence probabilities. For vector space with inherently linear structures, these functions depend only on the difference of two target word vectors. This idea can be expressed as,

$$F(\omega_i, \omega_j, \widetilde{\omega}_k) = F(\omega_i - \omega_j, \widetilde{\omega}_k) = \frac{P_{ik}}{P_{JK}} \tag{2}$$

where $\omega \in R^d$ are word vectors and P_{ik} is the co-occurrence probability of entity i and entity k. To achieve the symmetry, it is required that F be a homomorphism, modifying Equation (2) to,

$$\frac{F\left(\omega_i^T \widetilde{\omega}_k\right)}{F\left(\omega_j^T \widetilde{\omega}_k\right)} = \frac{P_{ik}}{P_{jk}} \tag{3}$$

Let F be exponential function and adding two bias items $b_i, \widetilde{b}_k$ respectively for $w_i, \widetilde{w}_k$,

$$w_i^T \widetilde{w}_k + b_i + \widetilde{b}_k = \log(X_{ik}) \tag{4}$$

To weight all co-occurrences differently, a non-decreasing weight function can be designed as,

$$f(x) = \left\{ \begin{array}{cc} \left(x/x_{cutoff}\right)^{\alpha} & \text{if } x < x_{cutoff} \\ 1 & \text{otherwise} \end{array} \right\} \tag{5}$$

where α and x_{cutoff} can be provided with empirical value. Finally, the cost function, which combines a least squares regression model with the weight function F, is presented as,

$$J = \sum_{i,j=1}^{V} f\left(X_{i,j}\right) \left(w_i{}^T \widetilde{w}_j + b_i + \widetilde{b}_j - \log X_{i,j}\right)^2 \tag{6}$$

where V is the size of the vocabulary. The word vectors can be trained by AdaGrad. More details of derivation can be found in [9].

3.2.2. Global Vectors for Distributional Relation Representation

Cosine similarity between entity vectors is a very effective quantitative representation of entity relations, which inspires us to study the ratio of co-occurrence probabilities from the point of cosine function. If two entity vectors w_i and w_j have a very high degree of similarity, entity i will occur more frequently in the context of j. This idea can be expressed as,

$$F\left(\cos\langle w_i, \widetilde{w}_j \rangle\right) = F\left(\frac{w_i{}^T \widetilde{w}_j}{|w_i||\widetilde{w}_j|}\right) = P_{ij} \tag{7}$$

Let F also be exponential function and adding two bias items $b_i, \widetilde{b}_j$ respectively for $w_i, \widetilde{w}_j$, Equation (4) will be changed to,

$$\cos\langle w_i, \widetilde{w}_j \rangle + b_i + \widetilde{b}_j = \log(X_{ij}) \tag{8}$$

Compared with Equation (4), we can conclude that it is more natural to approximate co-occurrence matrix by cosine similarity than dot product.

For the weighting function in global vectors, the cutoff is designed to ensure that large values of x are not overweighted. The main drawback to this weighting function is that it is hard to choose the empirical values of α and x_{cutoff}. In addition, unlike word co-occurrences such as *the* and *and*, entity co-occurrences will not suffer from extremely frequent co-occurrences. So, we simplify the weighting function as,

$$f(x) = x/x_{\max} \tag{9}$$

where $x_{\max}$ is the maximum value in the co-occurrence matrix. Figure 1 shows the two different weighting functions, where the blue one represents our simplified weighting function.

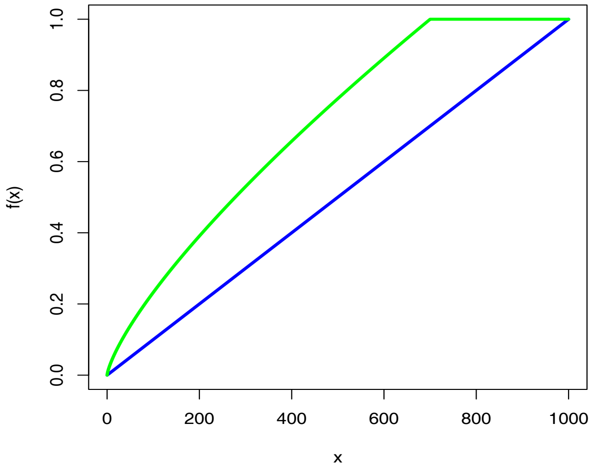

Figure 1. Weighting functions.

Finally, a new cost function can be showed by,

$$J = \sum_{i,j=1}^{V} \frac{X_{i,j}}{X_{\max}} \left(\cos\langle w_i, \widetilde{w}_j \rangle + b_i + \widetilde{b}_j - \log X_{i,j} \right)^2 \tag{10}$$

3.2.3. Training by AdaGrad

The goal of training is to obtain optimal entity vectors by minimizing the cost function. Stochastic gradient descent is an effective gradient descent optimization method for minimizing an objective function. But standard stochastic gradient descent methods only depend on the same initial learning rate. The adaptive gradient algorithm (AdaGrad) is proposed to solve this problem. AdaGrad can adaptively assign different learning rates to each parameter by,

$$x_{t+1} = x_t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_\tau + \varepsilon}} g_t \tag{11}$$

where η is the initial learning rate and ε is a small positive number. g_t is the gradient of cost function, which can be showed by,

$$J'\left(w_i{}^k\right) = \frac{2X_{i,j}}{X_{\max}} \cdot \left(\cos\langle w_i, \widetilde{w}_j\rangle + b_i + \widetilde{b}_j - \log(X_{ij})\right) \cdot \frac{\widetilde{w}_j^k |w_i|^2 - (w_i \cdot \widetilde{w}_j) w_i{}^k}{|w_i|^3 |\widetilde{w}_j|} \tag{12}$$

It can be concluded from Equation (11) that AdaGrad updates each parameter more slowly with larger update distance. Through above training, we can obtain all the entity vectors, which inherently present the relations between entities. So it is natural to establish entity relationships by computing the cosine similarity of entity vectors.

3.3. Entity Relational Storage of Neo4j

Neo4j is a commercially supported open-source graph database. It stores data in a graph, the most generic of data structures, capable of elegantly representing any kind of data in a highly accessible way. For entity relation representation, Neo4j records entities in nodes which have two properties: entity name and entity frequency. These nodes are organized by relation type which has the property of co-occurrence weight (X_{ij} in the co-occurrence matrix). Figure 2 shows the graph structure that we use for entity relation representation.

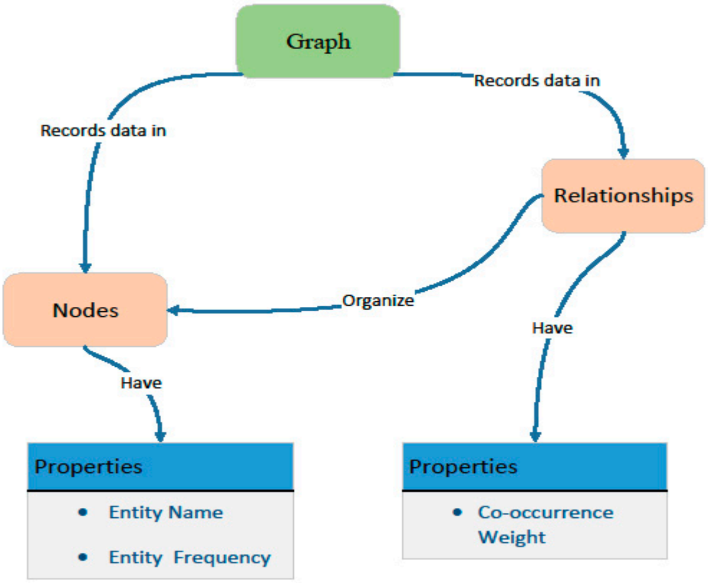

Figure 2. The graph structure for visual representation.

The cosine similarity between entity vectors is one of the most common methods for distributional entity relation representation. However, Neo4j is suitable for storing relational types instead of real numbers. In order to classify the relational types from distributional entity relation representation between entities, this paper presents an unsupervised method based on entity vectors. Firstly, we choose 24 relation sub-types in the task of ACE relation extraction as our relational types. Then we use the traditional global vector model to train the relational type vectors in a large-scale corpus of Sina News. Finally, we choose the most probable entity relationship type by calculating the cosine similarity between entity vectors and relationship type vector.

Neo4j can store hundreds of millions of nodes and relations. Querying from huge data needs a powerful query language. The declarative graph query language of cypher is designed to allow for expressive and efficient querying. Cypher is a humane query language which is similar with SQL.

There are four common distinct clauses for querying: START (starting pointing in the graph), MATCH (the graph pattern to match), WHERE (filtering criteria) and RETURN (what to return).

4. Experiments

In this section, we present and discuss the experimental results on the Chinese data set of Sina News. The flow diagram of our experiments is showed by Figure 3 First, an open tool of ICTCLAS [34] is employed to conduct word segmentation, POS tagging and named entity recognition. Co-occurrence matrix is obtained by making the statistics of entity co-occurrences in the whole corpus. Then, we use our improved model of RGloVe to train the entity vectors. Finally, we use the graph database of Neo4j to build, store and query these extracted relationships.

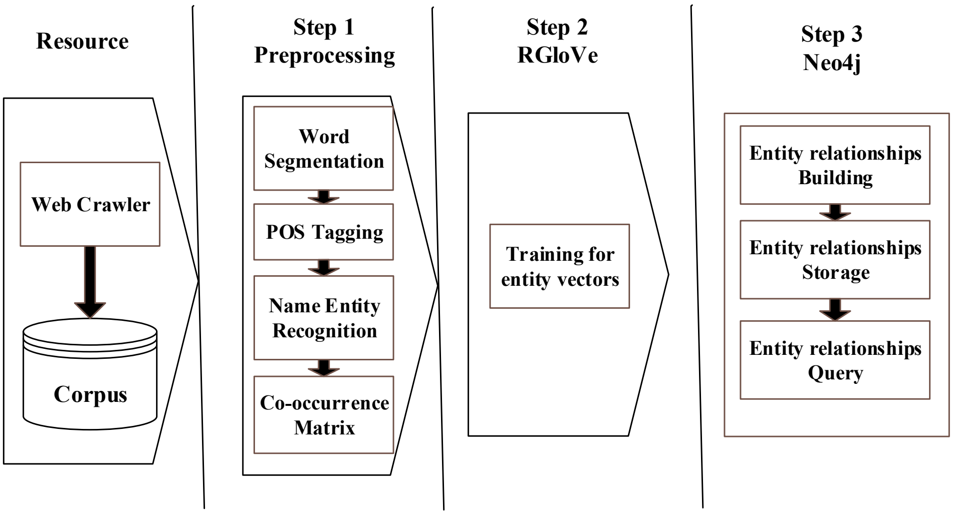

Figure 3. Our experimental framework for distributional entity relation representation.

4.1. Data Set and Experimental Settings

We choose the data set of Sina News, which contains 121,157 documents between 1 March and 31 August 2015. These documents are different in length and cover various categories, including politics, economy, sports, entertainment, etc. After preprocessing, 127,128 named entities and 3,230,441 entity pairs are extracted from the whole corpus.

We perform a comparative experiment among Word2Vec [32], GloVe [9] and RGloVe. Word2Vec is a very popular model based on neural networks to train word vectors. For Word2Vec, we choose the model of CBOW with 25 iterations for relationship type of 300 dimentions. For GloVe and RGloVe, we train the models using AdaGrad with initial learning rate of 0.05. We run 100 iterations for entity vectors of 300 dimensions. For global vectors, we set x_{cutoff} = 100 and α=3/4. Each model generates two sets of word vectors W and $\widetilde{W}$, which are equivalent. The final results of our entity vectors are decided by the sum $W + \widetilde{W}$.

4.2. Entity Vectors Presentation

Figures 4 and 5 intuitively present the vectors of GloVe and RGloVe. From the result of RGloVe, we can see clearly that *Jams* and *Curry* have the similar vector curve because they are all famous basketball players. Also, we can see that *Obama* and *Trump* have the similar vector curve because they are presidents of USA. But it is hard to find the rules from the result of GloVe.

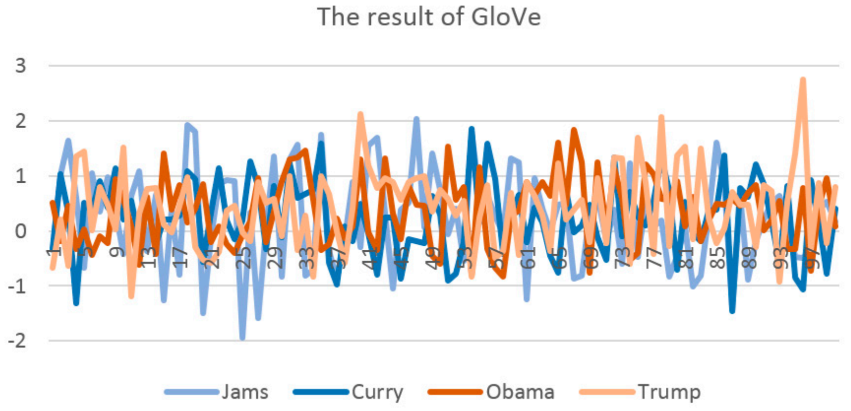

Figure 4. The result of GloVe.

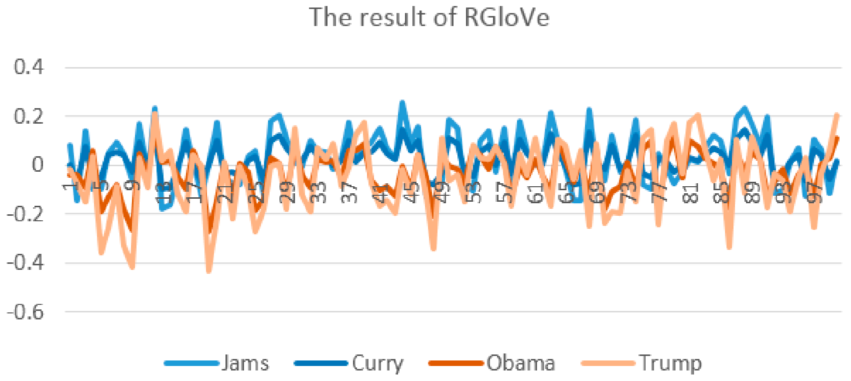

Figure 5. The result of RGloVe.

4.3. Quantitative Representation Result and Discussion

Cosine similarity between entity vectors provides a very effective quantitative representation of entity relations. In this paper, we use three methods of Word2Ve, GloVe and RGloVe to obtain the entity vectors. In order to compare the performance of these models, we make a series of assumptions and evaluation parameters: error rate, top N precision and average accuracy of relationship classification.

4.3.1. Error Rate

It is assumed that if the cosine similarity of two entities in the co-occurrence matrix is less than zero, the tuple will be regarded as a negative instance. Error rate is the ratio of all negative instances to the size of the co-occurrence matrix. Table 1 shows that our improved model of RGloVe achieves a 9.59% lower error rate than traditional global vectors.

Table 1. Error rates.

	Word2Vec (%)	**GloVe (%)**	**RGloVe (%)**
error rate	16.76	13.68	4.09

4.3.2. Top N Precision

We first select top N weight triples T_g from co-occurrence matrix as our ground truth. Then we define the similarity matrix, whose element tabulates the cosine similarity between two entity vectors.

Finally, we choose top N similarity triples T_c from similarity matrix as our comparative result. Top N precision is defined by,

$$precision = \frac{N_{T_g \cap T_c}}{N} \tag{13}$$

Top N precision is an effective approximate estimation of co-occurrence weights by computing the similarity of entity vectors. Table 2 shows the experimental results with different sample sizes. We can see from the results that our improved global vectors model can achieve better estimation to ground truth. But the top N precision is very low in both models of Word2Vec and GloVe because of the weakening of extremely frequent co-occurrences. In our improved model, we relax this weakening effect by using linear weighted function.

Table 2. Top *N* precision.

	Word2Vec (%)	GloVe (%)	RGloVe (%)
N = 1000	0.18	0.2	4.7
N = 5000	0.65	0.7	12.4
N = 10,000	1.13	1.24	15.98

4.3.3. Average Accuracy of Relationship Classification

In order to evaluate our performance of relationship classification, we conduct a manual labeling scheme to annotate the relationship types between extracted entities. Three independent annotators are instructed to distinguish 100 entity pairs of each relation sub-types. To measure the reliability our annotation scheme, we construct an agreement study by computing a value of Fleiss' kappa [35]. Fleiss's kappa is a statistical method for measuring the reliability of agreement between a fixed number of raters. For our annotation, we achieve a Fleiss's kappa value of 0.69, which is considered substantial agreement. Table 3 shows that our improved model of RGloVe achieves a 2.5% higher average accuracy than traditional global vectors and is close to the supervised method of SVM.

Table 3. Average accuracy of relationship classification.

	SVM (%)	Word2Vec (%)	GloVe (%)	RGloVe (%)
average accuracy	90.7	75.6	76.8	79.3

5. Conclusions and Future Work

In this paper, we have proposed an improved method of global vectors RGloVe for distributional entity relation representation. Unlike traditional relation extraction, distributional relation representation aims to train the entity vectors and measure the degree of closeness of relationship between two entities. The major advantage of distributional relation representation is that it is no longer limited to predefined relation types, which makes it easy to be applied to open domain question answering and information retrieval.

The statistics of entity co-occurrences in a corpus is the primary source of information available to distributional relation representation between entities. We first obtain a co-occurrence matrix, each of whose elements represents the co-occurrence weight of two entities. Then, in order to train the entity vectors more efficiently, we have developed an improved global vectors model of RGloVe by using the cosine similarity to approximate the entity occurrences instead of dot product. Finally, a graph database of Neo4j is introduced for building, storing and querying the relationships between named entities. The final comparative experiments show the superiority of our methods. In the future work, we will explore better classification criteria than cosine similarity between entity vectors and relationship type vector. In addition, it is significant to extend our model to perform experiments on the English corpus.

Acknowledgments: We are very grateful to our word group of text mining for their help with the data sets and evaluation. The work described in this paper was supported by National Natural Science Foundation of China (NO.61331017).

Author Contributions: Ziyan Chen and Kun Fu conceived and designed the experiments; Ziyan Chen performed the experiments; Yu Wang analyzed the data; Yu Wang and Xingyu Fu contributed analysis tools; Ziyan Chen, Yuexian Liang and Yang Wang wrote the paper. Authorship must be limited to those who have contributed substantially to the work reported. All authors have read and approved the final manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Doddington, G.; Mitchell, A.; Przybocki, M.; Ramshaw, L.; Strassel, S.; Weischedel, R. The automatic content extraction (ACE) program-tasks, data, and evaluation. *LREC* **2004**, *2*, 837–840.
2. Banko, M.; Etzioni, O.; Center, T. The Tradeoffs between Open and Traditional Relation Extraction. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, Columbus, OH, USA, 15–20 June 2008; pp. 28–36.
3. Etzioni, O.; Banko, M.; Soderland, S.; Weld, D.S. Open information extraction from the web. *Commun. ACM* **2008**, *51*, 68–74. [CrossRef]
4. Etzioni, O.; Fader, A.; Christensen, J.; Soderland, S.; Mausam, M.I. Open Information Extraction: The Second Generation. In Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16–22 July 2011; pp. 3–10.
5. Banko, M.; Cafarella, M.J.; Soderland, S.; Broadhead, M.; Etzioni, O. Open Information Extraction for the Web. In Proceedings of the 20th International Joint Conference on Artifical Intelligence, Hyderabad, India, 6–12 January 2007; pp. 2670–2676.
6. Kalyanpur, A.; Murdock, J.W. Unsupervised Entity-Relation Analysis in IBM Watson. In Proceedings of the Third Annual Conference on Advances in Cognitive Systems ACS, Atlanta, GA, USA, 28–31 May 2015; p. 12.
7. Fader, A.; Soderland, S.; Etzioni, O. Identifying Relations for Open Information Extraction. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, 27–31 July 2011; pp. 1535–1545.
8. Tseng, Y.-H.; Lee, L.-H.; Lin, S.-Y.; Liao, B.-S.; Liu, M.-J.; Chen, H.-H.; Etzioni, O.; Fader, A. Chinese Open Relation Extraction for Knowledge Acquisition. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, 26–30 April 2014; pp. 12–16.
9. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global Vectors for Word Representation. In Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
10. Zhou, G.; Zhang, M. Extracting relation information from text documents by exploring various types of knowledge. *Inf. Process. Manag.* **2007**, *43*, 969–982. [CrossRef]
11. Khayyamian, M.; Mirroshandel, S.A.; Abolhassani, H. Syntactic Tree-Based Relation Extraction Using a Generalization of Collins and Duffy Convolution Tree Kernel. In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, CO, USA, 31 May–5 June 2009; pp. 66–71.
12. Choi, M.; Kim, H. Social relation extraction from texts using a support-vector-machine-based dependency trigram kernel. *Inf. Process. Manag.* **2013**, *49*, 303–311. [CrossRef]
13. Choi, S.-P.; Lee, S.; Jung, H.; Song, S.-K. An intensive case study on kernel-based relation extraction. *Multimed. Tools Appl.* **2014**, *71*, 741–767. [CrossRef]
14. Zhang, C.; Xu, W.; Gao, S.; Guo, J. A Bottom-Up Kernel of Pattern Learning for Relation Extraction. In Proceedings of the Chinese Spoken Language Processing (ISCSLP), Singapore, 12–14 September 2014; pp. 609–613.
15. Nguyen, T.H.; Plank, B.; Grishman, R. Semantic Representations for Domain Adaptation: A Case Study on the Tree Kernel-Based Method for Relation Extraction. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Beijing, China, 27–31 July 2015; pp. 635–644.

16. Zhou, G.; Qian, L.; Zhu, Q. Label propagation via bootstrapped support vectors for semantic relation extraction between named entities. *Comput. Speech Lang.* **2009**, *23*, 464–478.
17. Sun, A.; Grishman, R.; Sekine, S. Semi-Supervised Relation Extraction with Large-Scale Word Clustering. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; pp. 521–529.
18. Fukui, K.-I.; Ono, S.; Megano, T.; Numao, M. Evolutionary Distance Metric Learning Approach to Semi-Supervised Clustering with Neighbor Relations. In Proceedings of the 2013 IEEE 25th International Conference on Tools with Artificial Intelligence (ICTAI), Herndon, VA, USA, 4–6 November 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 398–403.
19. Maziero, E.; Hirst, G.; Pardo, T. Semi-Supervised Never-Ending Learning in Rhetorical Relation Identification. In Proceeding of the Recent Advances in Natural Language Processing, Hissar, Bulgaria, 5–11 September 2015; pp. 436–442.
20. Min, B.; Shi, S.; Grishman, R.; Lin, C.-Y. Ensemble Semantics for Large-Scale Unsupervised Relation Extraction. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, 12–14 July 2012; pp. 1027–1037.
21. Wang, J.; Jing, Y.; Teng, Y.; Li, Q. A Novel Clustering Algorithm for Unsupervised Relation Extraction. In Proceedings of the Seventh International Conference Digital Information Management (ICDIM), Macau, Macao, 22–24 August 2012; pp. 16–21.
22. De Lacalle, O.L.; Lapata, M. Unsupervised Relation Extraction with General Domain Knowledge. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Seattle, Washington, USA, 18–21 October 2013; pp. 415–425.
23. Takase, S.; Okazaki, N.; Inui, K. Fast and large-scale unsupervised relation extraction. In Proceedings of 29th Pacific Asia Conference on Language, Information and Computation, Shanghai, China, 30 October–1 November 2015; pp. 96–105.
24. Remus, S. Unsupervised Relation Extraction of In-Domain Data From Focused Crawls. In Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics, Gothenburg, Sweden, 26–30 April 2014; pp. 11–20.
25. Alicante, A.; Corazza, A.; Isgrò, F.; Silvestri, S. Unsupervised entity and relation extraction from clinical records in Italian. *Comput. Biol. Med.* **2016**, *72*, 263–275. [CrossRef] [PubMed]
26. Landauer, T.K.; Dumais, S.T. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* **1997**, *104*, 211–240. [CrossRef]
27. Turney, P.D. Similarity of semantic relations. *Comput. Linguist.* **2006**, *32*, 379–416. [CrossRef]
28. Sebastian; Lapata, M. Dependency-based construction of semantic space models. *Comput. Linguist.* **2007**, *33*, 161–199.
29. Gamallo, P.; Bordag, S. Is singular value decomposition useful for word similarity extraction? *Lang. Resour. Eval.* **2011**, *45*, 95–119. [CrossRef]
30. Bengio, Y.; Ducharme, R.; Vincent, P.; Janvin, C. A neural probabilistic language model. *Mach. Learn. Res.* **2003**, *3*, 1137–1155.
31. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *Mach. Learn. Res.* **2011**, *12*, 2493–2537.
32. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
33. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Mach. Learn. Res.* **2011**, *12*, 2121–2159.
34. Zhang, H.-P.; Liu, Q.; Cheng, X.-Q.; Zhang, H.; Yu, H.-K. Chinese Lexical Analysis Using Hierarchical Hidden Markov Model. In Proceedings of the second SIGHAN workshop on Chinese language processing, Sapporo, Japan, 11–12 July 2003; pp. 63–70.
35. Fleiss, J.L. Measuring nominal scale agreement among many raters. *Psychol. Bull.* **1971**, *76*, 378. [CrossRef]