Multi modal representation learning and cross-modal semantic matching

Wang, X.

2

EMBEDDED REPRESENTATION OF RELATION WORDS WITH VISUAL SUPERVISION

This chapter is based on the following publication:

Wang, X., Du, Y., Li, X., Cao, F., Su, C. (2019, February). Embedded representation of relation words with visual supervision. In 2019 Third IEEE International Conference on Robotic Computing (IRC) (pp. 409-412). IEEE.

CHAPTER SUMMARY

This chapter addresses RQ1.

RQ1: To what extent is it possible to improve the representation of visual features detected by CNNs or the representation of textual features embedding and reduce the semantic gap between visual and textual information?

Word representation learned from the analysis of natural language does not usually reflect the true semantics of words. This chapter proposes a new method, named Visually Supervised Word2Vec (VS-Word2Vec) model to achieve the representation of the relation words that are important in knowledge related tasks. Our method first computes the visual feature vector of relation words based on deep networks, and then achieve the visual similarity matrix for all relation words, which we think reflects their true semantics. VS-Word2Vec model then combines the visual similarity and the CBOW and builds an optimization problem to jointly learn the word vector representation. Therefore, VS-Word2Vec fuses the visual modality and natural language together. Experiments implemented over the public datasets demonstrate that VS-Word2Vec model really changes the distribution of word representation and achieves more effective results in describing their true semantics than CBOW model.

The distributed representation of words in a vector space is an important issue in natural language processing tasks. The representation is mainly derived by modeling the context and statistic distribution of words in language documents instead of modeling their true semantics [46]. However, the data to describe the semantics of words includes many other modalities, such as visual and auditory data. Human learns how to understand the world by fusing the multiple modalities.

Many methods have been proposed to incorporate morphological information into word representations [47, 48, 49, 50]. Recently, Mikolov et al. [26, 25] proposed a set of models, such as CBOW and skip-gram, based on word analogies that probe the structure of the word embedding space. Pennington et al. [27] introduced a new global log-bilinear regression model (GloVe) based on the global matrix factorization and local context window methods. Bojanowski [51] proposed an approach by considering the morphology of words based on the skip-gram model, in which each word was represented as a bag of character n-grams and the word vector was the sum of the n-gram representations. For tasks at the intersection of vision and language, it seems prudent to model semantics as dictated by both text and vision. Kottur et al. [52] presented the Visual Word2Vec model which learned visually grounded word embedding to capture visual notions of semantic relatedness. Lu et al. [53] proposed a model that used visual relationships to improve the previous work by leveraging language priors from semantic word embedding to fine-tune the likelihood of a predicted relationship.

Most of these methods represent each word based on a single modality, and ignore the joint learning of multiple modalities. The phenomenon is much distinct in the knowledge related problems. In a knowledge triple (Arg1, Predicate, Arg2), Arg1 and Arg2 are subject and object strings, respectively, and Predicate is the string of relation [54, 55]. The vector representation of many predicate words (also called relation words) doesn't reflect their true similarity. For example, "above" and "below" is similar to each other in the representation by CBOW model. However, in the visual media, we know that they mean the opposite relative positions and show much different semantics.

This chapter proposes a new method, named Visualla y Supervised Word2Vec (VS-Word2Vec) model, to learn the vector representation of relation words. In this method, we first compute the visual feature vector of relation words based on deep networks, and then achieve the visual similarity matrix as weight for all textual relation words, which we think reflects their true semantics. VS-Word2Vec model then combines the visual similarity weight matrix and CBOW, and builds an optimization problem to jointly learn the word vector representation over visual modality and text modality. Experiments implemented over the public datasets demonstrate that our approach really changes the distribution of word representation and achieves more effective results in describing their true semantics than CBOW model.

2.1. OUR MODEL

2.1.1. BASIC CBOW MODEL

Suppose we have a sequence of training words $D = (w_1, w_2, ..., w_t, ..., w_T)$, where T is the number of words in this sequence and each word w_t belongs to a vocabulary $W = \{w_{I,1}, w_{I,2}, ..., w_{I,|W|}\}$, $|W|$ is the number of words in vocabulary. We represent the vector representation of word $w \in W$ as $\mathbf{v}_w \in \mathbf{R}^m$. The objective of the CBOW model [25] is to maximize the log probability:

$$J_T = \sum_{w_t \in D} \log p(w_t | Cont(w_t)) \tag{2.1}$$

where $Cont(w_t)$ denotes the context of a word w_t, i.e., $Cont(w_t) = \{w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}\}$ with the context window size c.

The CBOW model uses a binary tree to represent all words in the vocabulary. The $|W|$ words in vocabulary are leaf units of the tree. For each leaf unit, there exists a unique path from the root to this unit, and this path is used to estimate the probability of the word corresponding to the leaf unit. The probability in Eq.2.1 can be formulated as follows:

$$\begin{aligned} p(w|Cont(w)) &= \prod_{j=2}^{l^w} p(d_j^w | \mathbf{x}_w, \theta_{j-1}^w) \\ &= \prod_{j=2}^{l^w} [\sigma(\mathbf{x}_w^T \theta_{j-1}^w)]^{d_j^w} \cdot [1 - \sigma(\mathbf{x}_w^T \theta_{j-1}^w)]^{d_j^w} \end{aligned} \tag{2.2}$$

where l^w denotes the length of the path from root unit to the leaf one corresponding to w, d_j^w indicates that the inner unit at the j-th level is the left or right child of the inner unit at the (j-1)-th level, $d_j^w = 1$ denotes the left child case and $d_j^w = 0$ otherwise, θ_{j-1}^w is the parameter corresponding to the (j-1)-th level inner unit, on the path. $\sigma(\cdot)$ denotes the sigmoid function, i.e.,

$$\sigma(\mathbf{x}_w^T \theta_{j-1}^w) = \frac{1}{1 + e^{-\mathbf{x}_w^T \theta_{j-1}^w}} \tag{2.3}$$

and $\mathbf{x}_w = \sum_{w_i \in Cont(w)} \mathbf{v}_{w_i}$.

2.1.2. VISUALLY SUPERVISED WORD2VEC MODEL

Besides the vocabulary W defined above, we have a relation word vocabulary $R = \{w_{r,1}, w_{r,2}, \cdots, w_{r,|R|}\} \subset W$, which often describes the relation of two objects in image, i.e., "behind" and "contain".

Images visually represent the concepts in terms of their appearance, motion and space relation, etc, instead of the abstraction used in natural language. Distributed representations of words in a vector space help learning algorithms to

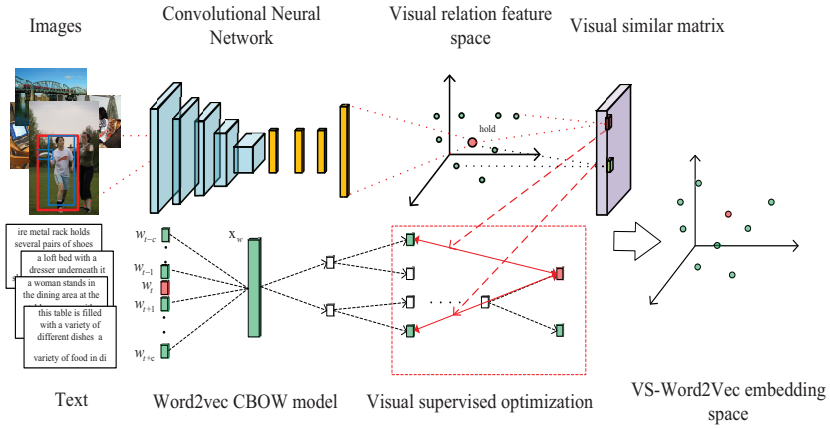

Figure 2.1: VS-Word2Vec Framework.

achieve better performance in natural language processing tasks by grouping similar word. However, we find that Word2Vec model cannot produce a good representation, especially for the relation words to represent the relationship of two entities in knowledge extraction task, in describing their semantic similarity. As shown in Fig.2.1, Our VS-Word2Vec model includes two parts: one is the basic CBOW, which learns the representation from natural language, and the second part is to compute the visual similarity of relation words based on the deep learning over image contents. Finally, our model jointly learns the representation by building an optimization problem over the above parts.

In the image data, each patch reflects a relation concept is bounded and is labeled by a relation word $w_{r,i} \in R$. The visual representation, denoted by $\mathbf{y}^q_{w_i}$, corresponding to $w_{r,i}$ can be computed over the bounded patch with Convolutional Neural Networks (CNNs)(16-layer VGG network), where the superscript q indicates the q-th patch example for the relation word. The visual representation for $w_{r,i}$ are given by the average of feature vector over its q visual examples:

$$\mathbf{y}_{w_i} = \frac{1}{Q_i} \sum_{q=1}^{Q_i} \mathbf{y}^q_{w_i} \tag{2.4}$$

where Q_i denotes the number of visual patch examples corresponding to $w_{r,i}$. The visual similarity two relation words $w_{r,i}$ and $w_{r,j}$ is defined as follows:

$$sim(w_{r,i}, w_{r,j}) = \frac{\mathbf{y}^T_{w_i} \mathbf{y}_{w_j}}{\|\mathbf{y}_{w_i}\| \cdot \|\mathbf{y}_{w_j}\|}. \tag{2.5}$$

For all $|R|$ relation words, we can obtain a similarity matrix $S_V = (s_{ij})_{|R|\times|R|}$ with $s_{ij} = sim(w_{r,i}, w_{r,j})$.

We aim to make the similarities of two relation words learned from natural language and from images consistent. Hence, we define the inconsistency of the similarities from two sources:

$$J_V(w_{r,i}, w_{r,j}) = (s_{ij} - \mathbf{x}_{w_{r,i}}^T \mathbf{v}_{w_{r,j}})^2 \tag{2.6}$$

where $\mathbf{x}_{w_{r,i}}$ has been defined in section 2.1.1. J_V is expected to be small to keep the consistency between the similarities derived from two sources. Our VS-Word2Vec can be formulated based on CBOW by the following optimization problem:

$$\max_{\{\mathbf{v}_{w_t}\}} J(\{\mathbf{v}_{w_t}\}), \tag{2.7}$$

where

$$J(\{\mathbf{v}_{w_t}\}) = \sum_{t=c+1}^{T-c} \big(\log p(w_t|Cont(w_t)) - \lambda \rho_t \sum_{w_{r,j} \in R} J_V(w_t, w_{r,j})\big) \tag{2.8}$$

where λ is a parameter to control the balance between two terms, $\rho_t = 0$ denotes that the word w_t is not the relation word defined in R, and $\rho_t = 1$ otherwise. The algorithm of VS-Word2Vec model is summarized in Algorithm 1.

2.2. EXPERIMENTAL RESULTS

2.2.1. DATASET AND EXPRERIMENT SETTINGS

We implement the experiments over the text8 dataset [1], which is often used as the test dataset for Word2Vec methods. We use the visual relationship detection data introduced by Lu et al. [53] to compute the similarity of relation words. The image relationship dataset contains 5000 images with 100 object categories and 70 predicates. In total, the dataset contains 37,993 relationships with 6,672 relationship types and 24.25 predicates per object category. In this chapter we focus on 22 predicates about 63 object categories. For both our model and the baseline method, we use the following parameters: the dimension of vector representation is 150, and the size of context window is $c = 2$. When building the word vocabulary, we keep the words those appear at least 3 times in the dataset. The learning rate is set to $\eta = 0.025$ in both CBOW and our model. The parameter λ in Eq.2.8 is experimentally set to 0.0025.

2.2.2. RESULTS AND ANALYSIS

Fig.2.2 illustrates the comparison of the similarity of 22 typical predicate words derived from images by deep networks, natural language by CBOW, and the multimodal data by our VS-Word2Vec, where all similarity values have been normalized into [0,1]. From the Figs.2.2(a) and 2.2(b), we observe that the similarity of relation words derived from image contents is different from that computed by CBOW

[1] http://mattmahoney.net/dc/text8.zip

Algorithm 1 VS-Word2Vec model.

Require: Word sequence D, word vocabulary W, relation word vocabulary R, images labeled with relation words.

Ensure: Vector representation $\{\mathbf{v}_{w_t}\}$;

Computer visual representation for each relation word based on Eq.2.4;
Compute visual similarity matrix S for all relation words with Eq.2.5;
Randomly initialize $\{\mathbf{v}_{w_t}\}$ and $\{\theta_{j-1}^{w_t}\}$;
for each $w_t \in D$ **do**
    $\mathbf{x}_{w_t} = \sum_{w_i \in Cont(w_t)} \mathbf{v}_{w_i}$; $\mathbf{e}_1 = \mathbf{0}$, $\mathbf{e}_2 = \mathbf{0}$;
    for $j = 2 : l^w$ **do**
        $g_1 = \eta(1 - d_j^{w_t} - \sigma(\mathbf{x}_{w_t}^T \theta_{j-1}^{w_t}))$;
        $\mathbf{e}_1 \leftarrow \mathbf{e}_1 + g_1 \theta_{j-1}^{w_t}$;
        $\theta_{j-1}^{w_t} \leftarrow \theta_{j-1}^{w_t} + g_1 \mathbf{x}_{w_t}$;
    end for
    if $w_t \in R$ **then**
        $\rho_t = 1$;
    else
        $\rho_t = 0$;
    end if
    for $j = 1 : q$ **do**
        $g_2 = \eta\lambda(\mathbf{x}_{w_t}^T \mathbf{v}_{w_j} - s_{tj})$;
        $\mathbf{e}_2 \leftarrow \mathbf{e}_2 + g_2 \mathbf{v}_{w_j}$;
    end for
    $\mathbf{v}_{w_t} \leftarrow \mathbf{v}_{w_t} + \mathbf{e}_1 + \rho_t \mathbf{e}_2$;
end for

over natural language. Fig.2.2(c) shows our results of VS-Word2Vec by considering the semantic similarity in images and the CBOW based natural language modeling together. The figure shows that our method really changes the similarity of relation words and tends to be close to the their true semantic similarity.

Fig.2.3 visualizes the location and distribution of 22 predicate relation words in 2-dimensional space. We observe that our approach really changes the distribution of the relation word representation compared the CBOW and describes their true semantics better. VS-Word2Vec model pushes away the location of relation words with different semantics and draw those words with similar semantics close. For example, “sit”, “lying” and “stand” represent three different kinds of motions and locate far away from each other in Fig.2.3(b) compared to Fig.2.3(a); “above” and “below” represent the opposite location of objects, and they are pushed away largely by our approach.

We also quantitatively evaluate the performance of our approach in measuring the word similarity or relatedness of semantics, which is also measured by the cosine distance shown in Eq.2.5. We choose SimVerb-3500 [56] as the ground

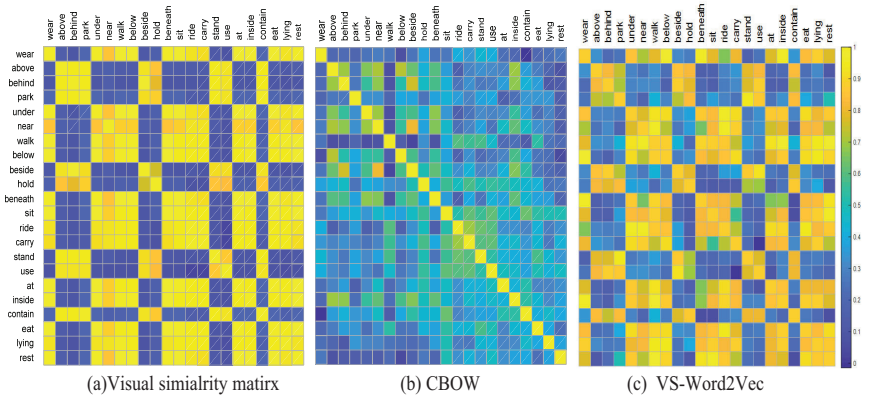

Figure 2.2: The comparison of the similarity of 22 typical relation words.

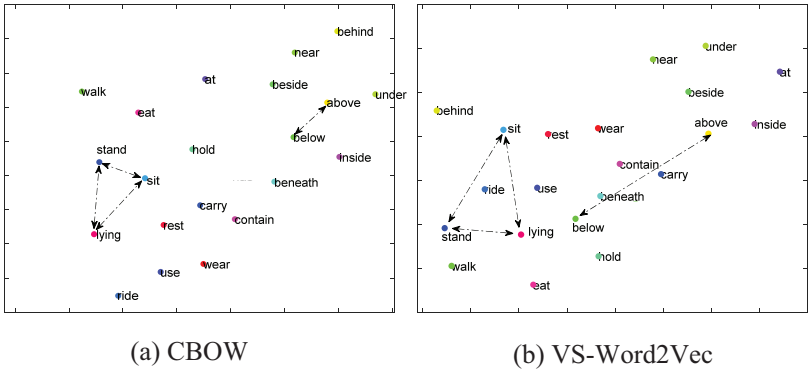

Figure 2.3: The visualization of the distribution of vector representation derived from the CBOW and our approach based on t-SNE.

truth, which gives the similarity of 3500 pairs of verbs. We normalize the similarity into [0,1] and use the following metric to evaluate the consistency of the similarity derived from our approach and the ground truth.

$$Con = \frac{\#(|S_g(p_i) - S_V(p_i)| < |S_g(p_i) - S_T(p_i)|)}{\#(p_i)} \tag{2.9}$$

where p_i denotes a pair of words, S_g, S_V and S_T are the similarity derived from ground truth, our VS-Word2Vec and CBOW Word2Vec, respectively, for the pair p_i. We choose 798 pairs that appear in both the Text8 and SimVerb-3500, and report the performance based on Eq.2.9 in Table 2.1. In this table, SYNONYMS, ANTONYMS, HYPER/HYPONYMS, COHYPONYMS and NONE are the different categories of pairs given in SimVerb-3500. We find that in three categories, the

Table 2.1: Confidence of Model

	CBOW	**Our Model**
SYNONYMS	**0.5618**	0.4492
ANTONYMS	0.3846	**0.6154**
HYPER/HYPONYMS	0.4192	**0.5808**
COHYPONYMS	**0.5231**	0.4769
NONE	0.1330	**0.8670**
Mean	0.4043	**0.5979**

similarity consistency of our approach is higher than CBOW Word2Vec model, and the average consistency of our approach is better.

2.3. CONCLUSION

In this chapter, we propose the VS-Word2Vec model to learn the vector representation of relation words by jointly compute over visual modality and natural language. In this method, we first compute the visual feature based on deep networks over an image patch that reflect a relation word, and then achieve the visual similarity matrix for all relation words. VS-Word2Vec model then resolve an optimization problem that consists of the terms related to the visual similarity and context in natural language. Experiments implemented demonstrate that our approach really changes the distribution of word representation and achieves more accurate similarity of words than CBOW model.