Aligning Knowledge and Text Embeddings by Entity Descriptions

Huaping Zhong[§] , **Jianwen Zhang**[†] , **Zhen Wang**[§] , **Hai Wan**[§] , **Zheng Chen**[†]
[§]{zhonghp@mail2,wangzh56@mail2,wanhai@mail}.sysu.edu.cn
[†]{jiazhan,zhengc}@microsoft.com
[§]Sun Yat-sen University [†]Microsoft Research

Abstract

We study the problem of jointly embedding a knowledge base and a text corpus. The key issue is the alignment model making sure the vectors of entities, relations and words are in the same space. Wang et al. (2014a) rely on Wikipedia anchors, making the applicable scope quite limited. In this paper we propose a new alignment model based on text descriptions of entities, without dependency on anchors. We require the embedding vector of an entity not only to fit the structured constraints in KBs but also to be equal to the embedding vector computed from the text description. Extensive experiments show that, the proposed approach consistently performs comparably or even better than the method of Wang et al. (2014a), which is encouraging as we do not use any anchor information.

1 Introduction

Knowledge base embedding has attracted surging interest recently. The aim is to learn continuous vector representations (embeddings) for entities and relations of a structured knowledge base (KB) such as Freebase. Typically it optimizes a global objective function over all the facts in the KB and hence the embedding vector of an entity / relation is expected to encode global information in the KB. It is capable of reasoning missing facts in a KB and helping facts extraction (Bordes et al., 2011; Bordes et al., 2012; Bordes et al., 2013; Socher et al., 2013; Chang et al., 2013; Wang et al., 2014b; Lin et al., 2015).

Although seeming encouraging, the approaches in the aforementioned literature suffer from two common issues: (1) Embeddings are exclusive to entities/relations within KBs. Computation between KBs and text cannot be handled, which are prevalent in practice. For example, in fact extraction, a candidate value may be just a phrase in text. (2) KB sparsity. The above approaches are only based on structured facts of KBs, and thus cannot work well on entities with few facts.

An important milestone, the approach of Wang et al. (2014a) solves issue (1) by jointly embedding entities, relations, and words into the same vector space and hence is able to deal with words/phrases beyond entities in KBs. The key component is the so-called *alignment model*, which makes sure the embeddings of entities, relations, and words are in the same space. Two alignment models are introduced there: one uses entity names and another uses Wikipedia anchors. However, both of them have drawbacks. As reported in the paper, using entity names severely pollutes the embeddings of words. Thus it is not recommended in practice. Using Wikipedia anchors completely relies on the special data source and hence the approach cannot be applied to other customer data.

To fully address the two issues, this paper proposes a new alignment method, aligning by entity descriptions. We only assume some entities in KBs have text descriptions, which almost always holds in practice. We require the embedding of an entity not only fits the structured constraints in KBs but also equals the vector computed from the text description. Meanwhile, if an entity has few facts, the description will provide information for embedding, thus the issue of KB sparsity is also well handled. We conduct extensive experiments on the tasks of triplet classification, link prediction, relational fact extraction, and analogical reasoning to compare with the previous approach (Wang et al., 2014a). Results show that our approach consistently achieves better or comparable performance.

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 267–272,
Lisbon, Portugal, 17-21 September 2015.

2 Related Work

TransE This is a representative knowledge embedding model proposed by Bordes et al. (2013). For a fact (h, r, t) in KBs, where h is the head entity, r is the relation, and t is the tail entity, TransE models the relation r as a translation vector $\mathbf{r}$ connecting the embeddings $\mathbf{h}$ and $\mathbf{t}$ of the two entities, i.e., $\mathbf{h} + \mathbf{r}$ is close to $\mathbf{t}$. The model is simple, effective and efficient. Most knowledge embedding models thereafter including this paper are variants of this model (Wang et al., 2014b; Wang et al., 2014a; Lin et al., 2015).

Skip-gram This is an efficient word embedding method proposed by Mikolov et al. (2013a), which learns word embeddings from word concurrencies in text windows. Without any supervision, it amazingly recovers the semantic relations between words in a vector space such as 'King' − 'Queen' ≈ 'Man' − 'Women'. However, as it is unsupervised, it cannot tell the exact relation between two words.

Knowledge and Text Jointly Embedding Wang et al. (2014a) combines knowledge embedding and word embedding in a joint framework so that the entities/relations and words are in the same vector space and hence operators like inner product (similarity) between them are meaningful. This brings convenience to tasks requiring computation between knowledge bases and text. Meanwhile, jointly embedding utilizes information from both structured KBs and unstructured text and hence the knowledge embedding and word embedding can be enhanced by each other. Their model is composed of three components: a *knowledge model* to embed entities and relations, a *text model* to embed words, and an *alignment model* to make sure entities/relations and words are in the same vector space. The knowledge model and text model are variants of TransE and Skip-gram respectively. The key component is the alignment model. They introduced two: alignment by entity names and alignment by Wikipedia anchors. (1) **Alignment by Entity Names** makes a replicate of KB facts but replaces each entity ID with its name string, i.e., the vector of a name phrase is encouraged to equal to the vector of the entity (identified by ID). It has problems with ambiguous entity names and observed polluting word embeddings thus it is not recommended by the authors. (2) **Alignment by Wikipedia Anchors** replaces the surface phrase v of a Wikipedia anchor with its corresponding Freebase entity e_v and defines the likelihood

$$\mathcal{L}_{AA} = \sum_{(w,v)\in\mathcal{C}, v\in\mathcal{A}} \log \Pr(w|e_v) \quad (1)$$

where $\mathcal{C}$ is the collection of observed word and context pairs and $\mathcal{A}$ refers to the set of all anchors in Wikipedia. $\Pr(w|e_v)$ is the probability of the anchor predicting its context word, which takes a form similar to Skip-gram for word embedding. Alignment by anchors works well in both improving knowledge embedding and word embeddings. However, it completely relies on the special data source of Wikipedia anchors and cannot be applied to other general data settings.

3 Alignment by Entity Descriptions

We first describe the settings and notations. Given a knowledge base, i.e., a set of facts (h, r, t), where $h, t \in \mathcal{E}$ (the set of entities) and $r \in \mathcal{R}$ (the set of relations). Some entities have text descriptions. The description of entity e is denoted as D_e. $w_{i,n}$ is the n^{th} word in the description of e_i. N_i is the length (in words) of the description of e_i. We try to learn embeddings $\mathbf{e}_i$, $\mathbf{r}_j$ and $\mathbf{w}_l$ for each entity e_i, relation r_j and word w_l respectively. The vocabulary of words is $\mathcal{V}$. The union vocabulary of entities and words together is $\mathcal{I} = \mathcal{E} \cup \mathcal{V}$. In this paper "word(s)" refers to "word(s)/phrase(s)".

We follow the jointly embedding framework of (Wang et al., 2014a), i.e., learning optimal embeddings by minimizing the following loss

$$\mathcal{L}\left(\{\mathbf{e}_i\}, \{\mathbf{r}_j\}, \{\mathbf{w}_l\}\right) = \mathcal{L}_K + \mathcal{L}_T + \mathcal{L}_A, \quad (2)$$

where $\mathcal{L}_K$, $\mathcal{L}_T$ and $\mathcal{L}_A$ are the component loss functions of the knowledge model, text model and alignment model respectively. Our focus is on a new alignment model $\mathcal{L}_A$ while the knowledge model $\mathcal{L}_K$ and text model $\mathcal{L}_T$ are the same as the counterparts in (Wang et al., 2014a). However, to make the content self-contained, we still need to briefly explain $\mathcal{L}_K$ and $\mathcal{L}_T$.

Knowledge Model Describes the plausibility of a triplet (h, r, t) by defining

$$\Pr(h|r,t) = \frac{\exp\{z(h,r,t)\}}{\sum_{\tilde{h}\in\mathcal{I}} \exp\{z(\tilde{h},r,t)\}}, \quad (3)$$

where $z(h, r, t) = b - 0.5 \cdot \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_2^2$, $b = 7$ as suggested by Wang et al. (2014a). $\Pr(r|h, t)$ and

$\Pr(t|h,r)$ are defined in the same way. The loss function of knowledge model is then defined as

$$\mathcal{L}_K = -\sum_{(h,r,t)} \big[\log \Pr(h|r,t) + \log \Pr(t|h,r) + \log \Pr(r|h,t)\big] \quad (4)$$

Text Model Defines the probability of a pair of words w and v co-occurring in a text window:

$$\Pr(w|v) = \frac{\exp\{z(w,v)\}}{\sum_{\tilde{w}\in\mathcal{V}} \exp\{z(\tilde{w},v)\}} \quad (5)$$

where $z(w,v) = b - 0.5 \cdot \|\mathbf{w} - \mathbf{v}\|_2^2$. Then the loss function of text model is

$$\mathcal{L}_T = -\sum_{(w,v)} \log \Pr(w|v) \quad (6)$$

Alignment Model This part is different from Wang et al. (2014a). For each word w in the description of entity e, we define $\Pr(w|e)$, the conditional probability of predicting w given e:

$$\Pr(w|e) = \frac{\exp\{z(e,w)\}}{\sum_{\tilde{w}\in\mathcal{V}} \exp\{z(e,\tilde{w})\}}, \quad (7)$$

where $z(e,w) = b - 0.5 \cdot \|\mathbf{e} - \mathbf{w}\|_2^2$. Notice that $\mathbf{e}$ is the same vector of entity e appearing in the knowledge model of Eq. (3).

We also define $\Pr(e|w)$ in the same way by revising the normalization term

$$\Pr(e|w) = \frac{\exp\{z(e,w)\}}{\sum_{\tilde{e}\in\mathcal{E}} \exp\{z(\tilde{e},w)\}} \quad (8)$$

Then the loss function of alignment model is

$$\mathcal{L}_A = -\sum_{e\in\mathcal{E}} \sum_{w\in D_e} [\log \Pr(w|e) + \log \Pr(e|w)] \quad (9)$$

Training We use stochastic gradient descent (SGD) to minimize the overall loss of Eq. (2), which sequentially updates the embeddings. Negative sampling is used to calculate the normalization items over large vocabularies. We implement a multi-threading version to deal with large data sets, where memory is shared and lock-free.

4 Experiments

We conduct experiments on the following tasks: link prediction (Bordes et al., 2013), triplet classification (Socher et al., 2013), relational fact extraction (Weston et al., 2013), and analogical reasoning (Mikolov et al., 2013b). The last one evaluates quality of word embeddings. We try to study whether the proposed alignment model, without using any anchor information, is able to achieve comparable or better performance than alignment by anchors. As to the methods, "Separately" denotes the method of separately embedding knowledge bases and text. "Jointly(anchor)" and "Jointly(name)" denote the jointly embedding methods based on Alignment by Wikipedia Anchors and Alignment by Entity Names in (Wang et al., 2014a) respectively. "Jointly(desp)" is the joint embedding method based on alignment by entity descriptions.

Table 1: Link prediction results.

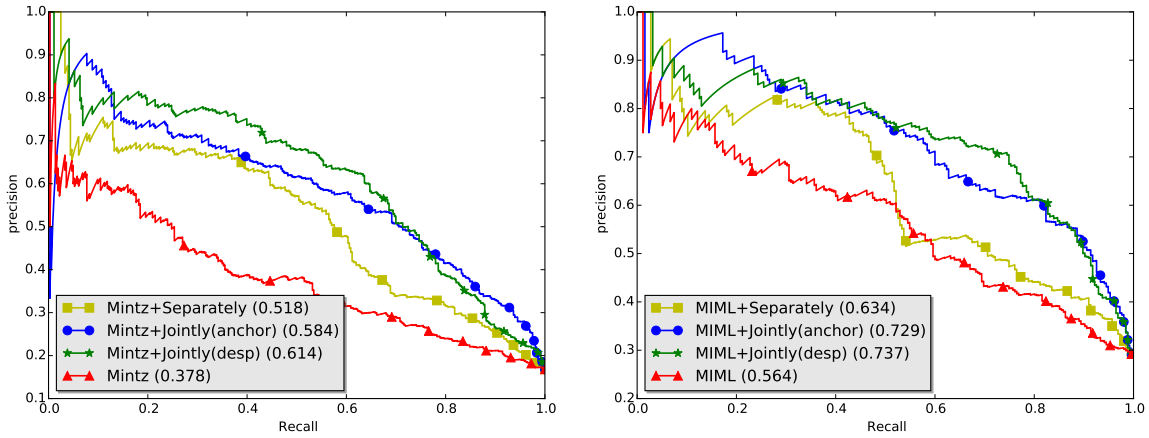
Metric	MEAN		HITS@10	
	Raw	Filtered	Raw	Filtered
TransE	243	125	34.9	47.1
Jointly(anchor)	**166**	47	49.9	72.0
Jointly(desp)	167	**39**	**51.7**	**77.3**

Table 2: Triplet classification results.

Type	e - e	w - e	e - w	w - w	all
Separately	94.0	51.7	51.0	69.0	73.6
Jointly(anchor)	95.2	65.3	65.1	76.2	79.9
Jointly(desp)	**96.1**	**66.7**	**66.1**	**76.4**	**80.9**

Data For link prediction, FB15K from (Bordes et al., 2013) is used as the knowledge base. For triplet classification, a large dataset provided by (Wang et al., 2014a) is used as the knowledge base. Both sets are subsets of Freebase. For all tasks, Wikipedia articles are used as the text corpus. As many Wikipedia articles can be mapped to Freebase entities, we regard a Wikipedia article as the description for the corresponding entity in Freebase. Following the settings in (Wang et al., 2014a), we apply the same preprocessing steps, including sentence segmentation, tokenization, and named entity recognition. We combine the consecutive tokens covered by an anchor or identically tagged as "Location/Person/Organization" and regard them as phrases.

Link Prediction This task aims to complete a fact (h,r,t) in absence of h or t, simply based on $\|\mathbf{h}+\mathbf{r}-\mathbf{t}\|$. We follow the same protocol in (Bordes et al., 2013). We directly copy the results of the baseline (TransE) from (Bordes et al., 2013) and implement "Jointly(anchor)". The results are in Table 1. "MEAN" is the average rank of the true absent entity. "HITS@10" is accuracy of the top



(a) (b)

Figure 1: Precision-recall curves for relation extraction. (a) Mintz (Mintz et al., 2009) as base extractor (b) MIML (Surdeanu et al., 2012) as base extractor.

10 predictions containing the true entity. Lower "MEAN" and higher "HITS@10" is better. "Raw" and "Filtered" are two settings on processing candidates (Bordes et al., 2013).

We train "Jointly(anchor)" and "Jointly(desp)" with the embedding dimension k among $\{50, 100, 150\}$, the learning rate α in $\{0.01, 0.025\}$, the number of negative examples per positive example c in $\{5, 10\}$, the max skip-range s in $\{5, 10\}$ and traverse the text corpus with only 1 epoch. The best configurations of "Jointly(anchor)" and "Jointly(desp)" are exactly the same: $k = 100, \alpha = 0.025, c = 10, s = 5$.

From the results, we observe that: (1) Both jointly embedding methods are much better than the baseline TransE, which demonstrates that external textual resources make entity embeddings become more discriminative. Intuitively, "Jointly(anchor)" indicates "*how to use an entity in text*", while "Jointly(desp)" shows "*what is the definition/meaning of an entity*". Both are helpful to distinguish an entity from others. (2) Under the setting of "Raw", "Jointly(desp)" and "Jointly(anchor)" are comparable. In other settings "Jointly(desp)" wins.

Triplet Classification This is a binary classification task, predicting whether a candidate triplet (h, r, t) is a correct fact or not. It is used in (Socher et al., 2013; Wang et al., 2014b; Wang et al., 2014a). We follow the same protocol in (Wang et al., 2014a).

We train their models via our own implementation on our dataset. The results are in Table 2. "e-e" means both sides of a triplet (h, r, t) are entities in KB, "e-w" means the tail side is a word out of KB entity vocabulary, similarly for "w-e" and "w-w". The best configurations of the models are: $k = 150, \alpha = 0.025, c = 10, s = 5$ and traversing the text corpus with 6 epochs.

The results reveal that: (1) Jointly embedding is indeed effective. Both jointly embedding methods can well handle the cases of "e-w", "w-e" and "w-w", which means the vector computation between entities/relations and words are really meaningful. Meanwhile, even the case of "e-e" is also improved. (2) Our method, "Jointly(desp)", outperforms "Jointly(anchor)" on all types of triplets. We believe that the good performance of "Jointly(desp)" is due to the appropriate design of the alignment mechanism. Using entity's description information is a more straightforward and effective way to align entity embeddings and word embeddings.

Relational Fact Extraction This task is to extract facts (h, r, t) from plain text. Weston et al. (2013) show that combing scores from TransE and some text side base extractor achieved much better precision-recall curve compared to the base extractor. Wang et al. (2014a) confirm this observation and show that jointly embedding brings further encouraging improvement over TransE. In this experiment, we follow the same settings as (Wang et al., 2014a) to investigate the performance of our new alignment model. We use the

Table 3: Analogical reasoning results

Metric	Words		Phrases	
	Acc.	Hits@10	Acc.	Hits@10
Skip-gram	67.4	86.7	22.0	63.6
Jointly(anchor)	**69.4**	87.7	26.2	68.1
Jointly(name)	44.5	69.7	11.5	46.0
Jointly(desp)	69.3	**88.3**	**49.0**	**86.5**

same public dataset NYT+FB, released by Riedel et al. (2010) and used in (Weston et al., 2013) and (Wang et al., 2014a). We use Mintz (Mintz et al., 2009) and MIML (Surdeanu et al., 2012) as our base extractors.

In order to combine the score of a base extractor and the score from embeddings, we only reserve the testing triplets whose entitites and relations can be mapped to the embeddings learned from the triplet classification experiment. Since both Mintz and MIML are probabilistic models, we use the same method in (Wang et al., 2014a) to linearly combine the scores.

The precision-recall curves are plot in Fig. (1). On both base extractors, the jointly embedding methods outperform separate embedding. Moreover, "Jointly(desp)" is slightly better than "Jointly(anchor)", which is in accordance with the results from the link prediction experiment and the triplet classification experiment.

Analogical Reasoning This task evaluates the quality of word embeddings (Mikolov et al., 2013b). We use the original dataset released by (Mikolov et al., 2013b) and follow the same evaluation protocol of (Wang et al., 2014a). For a true analogical pair like ("France", "Paris") and ("China", "Beijing"), we hide "Beijing" and predict it by selecting the word from the vocabulary whose vector has highest similarity with the vector of "China" + "Paris" - "France". We use the word embeddings learned for the triplet classification experiment and conduct the analogical reasoning experiment for "Skip-gram", "Jointly(anchor)", "Jointly(name)" and "Jointly(desp)".

Results are presented in Table 3. "Acc" is the accuracy of the predicted word. "HITS@10" is the accuracy of the top 10 candidates containing the ground truth. The evaluation analogical pairs are organized into two groups, "Words" and "Phrases", by whether an analogical pair contains phrases (i.e., multiple words). From the table we observe that: (1) Both "Jointly(anchor)" and "Jointly(desp)" outperform "Skip-gram". (2) "Jointly(desp)" achieves the best results, especially for the case of "Phrases". Both "Jointly(anchor)" and "Skip-gram" only consider the context of words, while "Jointly(desp)" not only consider the context but also use the whole document to disambiguate words. Intuitively, the whole document is also a valuable resource to disambiguate words. (3) We further verify that "Jointly(name)", i.e., using entity names for alignment, indeed pollutes word embeddings, which is consistent with the reports in (Wang et al., 2014a).

The above four experiments are consistent in results: without using any anchor information, alignment by entity description is able to achieve better or comparable performance, compared to alignment by Wikipedia anchors proposed by Wang et al. (2014a).

5 Conclusion

We propose a new alignment model based on entity descriptions for jointly embedding a knowledge base and a text corpus. Compared to the method of alignment using Wikipedia anchors Wang et al. (2014a), our method has no dependency on special data sources of anchors and hence can be applied to any knowledge bases with text descriptions for entities. Extensive experiments on four prevalent tasks to evaluate the quality of knowledge and word embeddings produce very consistent results: our alignment model achieves better or comparable performance.

References

Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. 2011. Learning structured embeddings of knowledge bases. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*.

Antoine Bordes, Xavier Glorot, Jason Weston, and Yoshua Bengio. 2012. A semantic matching energy function for learning with multi-relational data. *Machine Learning*, pages 1–27.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795.

Kai-Wei Chang, Wen-tau Yih, and Christopher Meek. 2013. Multi-relational latent semantic analysis. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1602–1612, Seattle, Washington, USA, October. Association for Computational Linguistics.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Zheng Chen. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 2181–2187.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

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

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems*, pages 926–934.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014a. Knowledge graph and text jointly embedding. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics*, pages 1591–1601.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014b. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1112–1119.

Jason Weston, Antoine Bordes, Oksana Yakhnenko, and Nicolas Usunier. 2013. Connecting language and knowledge bases with embedding models for relation extraction. *arXiv preprint arXiv:1307.7973*.