Learning Bi-lingual Word Representations using Distributed Neural Models

First Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Second Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Abstract

Word representations and specially word feautres induced by distributed models are shown to be able to boost the performance of various NLP tasks such as: word sense disambiguation, named entity recognition and parsing. Previously a model have been proposed to learn representation for entities of a structured lexical resource such as WordNet.

Here in this paper, we follow the previous work and extend their idea by incorporating multiple resources in order to induce richer representations jointly for two different languages. We have evaluated both mono-lingual and bi-lingual embeddings on four different gold datasets for word-pair similarity task and shown that bi-lingual embeddings perform similarly or better than mono-lingual embeddings. For example on one of datasets, bi-lingual embeddings is 10 percent(??) more correlated (Pearson correlation) to human judgements than mono-lingual embeddings of the same model and up to 40 percent more than the other models.

1 Introduction

In a large number of machine learning methods and their application to computational linguistics feature engineering or extracting informative features is a crucial part and it is done mostly manually. *Representation Learning* is an umberella term for a family of unsupervised methods to learn features from data to decrease the manual labour. Most of recent works related to this idea focus on inducing word representations. *Word representation* or *Word embedding* "is a mathematical object, usually a vector, which each dimension in this vector represents a grammatical or semantical feature to identify this word and is induced automatically from data" (**?**). Recently, it has been shown in (**?**) and (**?**) that using induced word representations can be helpful to improve state-of-the-art methods in variouse tasks. While these word embeddings are induced for a single language, Klementiev et al. [Inducing Crosslingual Distributed Representations of Words] have a model which learns cross-lingual representations and is shown to have superior performance for text classification task over strong baselines. In contrast to previous similar works which word embeddings learnt from a corpus, Bordes et al. proposed a method (**?**) to learn distributed representations from multi-relational knowledge bases(KB) like Freebase or lexical resources like WordNet. They encode information in KBs as binary relations between entities which each relation is instantiated from a set of relation types available in the KB. Since we are following their methodology, a description of their work is presented in 2.1.

The main motivation of this paper is to propose a framework to induce crosslingual word representations which benefit from aggregation of information in two different resource in two (or more) different languages. We will show in Section 2.2 that using machine learning tool developd in (Bordes2011 and Bordes2012) and specific encoding of information we will be able to do so.

In order to evalute our embeddings, we have chosen (??grammar) word pairs similarity task which helps us to investigate on effectiveness of our embeddings to capture different aspects and features of words meanings.

In Section 3 we will also show the comparison between our embeddings and the embeddings induced by other models both for English and German. This paper will be concluded by our analysis of result and models we used, as well as our suggestions for future work.

2 Representation Learning from Knowledge Bases

In this section, we will first review a framework proposed in (Bordes2011 and Bordes2012) and then will show how to use information encoded in or inferred from multiple lexicons to induce cross-lingual word representation with those models.

2.1 Bordes model for word embedding

Two major models are proposed in [Bordes2011] and [Bordes2012] to learn features in continues vector space from a Knowledge Bases(KB) which information is usually representated in form of triples of (e_i, r_k, e_j) where e_i and e_j are i_{th} and j_{th} entities related by a binary relation of type r_k. The purpose of the models is to induce a vector space and associate each entity or relation to an embedding vector or a matrix. The dimensions of such an embedding vector are supposed to reflect a set of informative features of entities and relations.

In the first model, **structured embeddings(SE)**, entities are modeled as d-dimensional vectors. An associated vector to the i_{th} entity, e_i, is $E_i \in \mathbb{R}^d$. Each relation r_k is decomposed to two operators each represented as $d \times d$ matrix, $R_k = (R_k^{left}, R_k^{right})$. These operators transform the left and right entities to a new space induced by each relation and by using a p-norm measure (L1 norm in this work) they associate a similarity value or a score to each triple. This similarity value is being calculated by Equation (1).

$$Sim(E_i, E_j, R) = ||R_k^{left} E_i - R_k^{right} E_j||_1 \quad (1)$$

The similarity between transformed entities works as a score to measure the strength of a relation holds between two entities.

Using the idea of contrastive learning , the model will be trained to increase similarity of embeddings for a positive triple (a triple which exists in the KB) or lowering its rank among other training samples and decrease the similarity of embeddings when the relation doesn't hold (negative triple) or raising its rank . For each positive triple, two negative triples will be generated by randomly alternating the right entity or left entity with other entities. Inspired from large margin methods a constraint is introdcued on the model that forces negative triples to have lower associated similarity value than correspondent positive triples by a large margin.

The second model, **Semantic Matching Energy using Bilinear layers(SME-Bil)**, is using a different represention for relations,weighted bilinear transformation of embeddings and dot product similarity function instead of L1 norm. In this model, each relation is represented by a d-dimensional vector R_k same as entities. For triple (e_i, r_k, e_j) , the model combines the weighted transformation of each entity embedding with the weighted embedding of relation using element-wise vector product. as it is shown in Equation (2).

$$E'_{left} = (W_i E_i) \odot (W_k R_k) + b_{left} \quad (2)$$

W_i and W_k are $d \times d$ weight matrices and b_{left} is a d-dimensional bias vector. The same equation holds for transforming the right entity embeddings to E'_{right}. Finally, the associated score for the triple can be calucated by dot product of E'_{left} and E'_{right} which is shown in Equation 3.

$$Sim(E_i, E_j, R_k) = -E'_{left} E'_{right} \quad (3)$$

Similar constraints to the first model are also applied to this model and both models can be trained by stochahstic gradient descent (SGD) (??)

2.2 Creating of Dataset

In this part of paper we describe the methodology we followed to encode available information in two different lexical resources, WordNet and GermaNet, that makes it possible to learn bi-lingual embeddings of word senses in German and English. The main idea is to relate two senses from two different resources using cross-lingual sense alignments. This is an additional information which can play a role of bridge between two different tasks, learning German embeddings and English embeddings, and can help to transfer knowledge from one to the another. Using this new feature we make our WordNet-GermaNet dataset which contains three type of relations (1) WordNet relations (2) GermaNet relations (3) Cross-lingual sense alignments between WordNet and GermaNet

First two types of relations are directly extracted from WordNet and GermaNet and for the cross-lingual relations we used Interlingual Index mappings between WordNet and GermaNet.
Example of relations:

WN-sense-A	WN-rel-1	WN-sense-B

GN-sense-C	GN-rel-2	GN-sense-D
WN-sense-A	ILI-rel-1-2	WN-sense-B

Left and right entities are WordNet and GermaNet senses and relations are current semantical relations in each of lexicons such as: meronymy, holonymy and . . . .

We have created four different dataset, each divided to train, test and validation separated subsets. Our four datasets are:

1. Only WordNet triples (WN)
2. Only GermaNet triples (GN)
3. WordNet-GermaNet triples with one-direction cross-lingual alignments (WN-GN)
4. WordNet-GermaNet with double-direction cross-lingual alignments (WN-GN DD)

Dataset 3 includes both relations extracted from WordNet and GermaNet and also the mapping between senses. Dataset 4 is same as dataset 3 but since the models we will use are assuming all the relations are assymetric we will try to encode the symmetry of cross-lingual alignments by reversing each of them and include the reverse in the dataset. Datasets 3 and 4 contains two different variants: the first variants contains only WordNet relations (test on WN) in the held-out test dataset and the second variant contains only GermaNet triples (test on GN). In this way we can observe the direction of possibly transferring information from English to German or vice versa.

For reducing the sparsity of data and boosting the learning runtime we filtered out all the entities that appeared less than 3 times in our datasets.

(version of wordnet, germanet, ILI and role of uby should be described here)

3 Evaluation

To show the effectiveness of joint learning of features from multiple knowledge bases we suggest two experiment setups. In the first schema we follow Bordes et al. ranking task. The goal of this task is to show how well information in knowledge bases can be preserved by learned features. On the other hand, the second setup is investigating on this question that if bi-lingual word embeddings from multiple resources are able to improve the performance of mono-lingual embeddings in a standard NLP task, here word-pair similarity or not. In this setup we will look to contribution of the learned features in predicting similarity of words.

3.1 Intrinstic Evaluation

Bordes et al. (Bordes2011) proposed a ranking task that for each triple (e_i, r_k, e_j) in the data set, all the entities will be ranked as a candidate for being right entity of the triple given the relation and the left entity. Depends on which one of the models is used, SE or SME-Bil, all the entities will be sorted based on their score regarding Equation (1) or Equation 3 previously introduced in section 2.1. By keeping the statistics of difference between the predicted rank of e_j and its true rank and also repeat the same process for left entities, we will be able to report the mean and median predicted rank of entities per relation and in total. Bordes et al. proposed to schema for calculating the average rank, micro averaging which emphasis on more frequent relations by weighted averaging with frequency of relations as weights and macro averaging which consider all the relations equally , either frequent or infrequent ones. The third statistic that we report following their work, r@100, is the ratio of number of times that an entity is correctly among top 100 entities ranked and predicted for a triple to the number of occurances of this entity in the dataset. We applied SE and SME-Bil models on our created datasets and the ranking performance on each of them is presented in Table 1.

3.2 Extrinsic Evaluation

We are interested to further analyze the effectiveness of learned embeddings to capture semantic features of words, therefore we compare the mono-lingual and bi-lingual embeddings against human judgments and also other embeddings learned from corpus. The other embeddings which we used for our comparison are (Turian et al., HLBL and Klementiev et al.). To measure the similarity between any given wordpair (w_1, w_2) we find all vectors associated to different senses of the given words in our embedding dictionary and pick the pair of embeddings that maximize cosine similarity between two words. We can motivate this by saying that for each word pair any of words works as a context for disambiguating the sense of the other word.

Four datasets of word-pair similarity are used to compare correlation of predicted similairty of pair of words against human judgments. RG-65 [rubensteinGoodenough], Yang and Powers verb similarity dataset[yangPowers], MC-30[millerCharles] and WS-353 [finkelstein] are English datasets that we used for this task. RG-65 and its subset, MC-30 are providing human scored datasets for measuring synonymy among word-pairs (nouns),. WS-353 has broader notaion of semantic similarity and include word pairs for measuirng semantic relatedness too. Yang and Powers have provided a dataset for measuring semantic similarity between verbs.

Both Pearson and Spearman correlation of predicted and gold similarities are calculated and is reported in table 2 for English. We can see that bi-lingual embeddings learned by SME-Bil model outperformed all other embeddings both in Pearson and Spearman correlation in all four datasets. Another observation is that SME-Bil models performes better than SE models in most cases. Among SME-Bil models, bi-lingual embeddings are always more correlated to human hudgments than mono-lingual embeddings. All models perform poorly on WS-353, bi-lingual SME-Bil model still performs better than the rest. We believe this could due to including notion of relatedness in this dataset. Poor performance of knowledge based models on semantic relatedness task is previously known and discussed in [Szulmanski2013].

For German, we use [this and that].

4 Conclusion and Future Work

References

Table 1: Intrinsic Evaluation (Ranking Score Performance)

Dataset	#relations	#entities		Micro	Macro
GN SE	16	64025	lhs	84.42	72.59
			rhs	84.04	72.38
			mean	1003.59	3739.85
			median	5.0	2213.37
			global	84.23	72.49
GN SME-Bil	16	64025	lhs	79.06	58.58
			rhs	83.30	81.11
			mean	407.90	308.01
			median	10.0	54.18
			global	81.18	69.85
WN SE	23	148976	lhs	91.90	89.47
			rhs	92.30	90.25
			mean	148.72	623.10
			median	5.0	4.69
			global	92.10	89.86
WN SME-Bil	23	148976	lhs	83.08	72.21
			rhs	85.2	77.92
			mean	128.82	511.21
			median	10.0	26.63
			global	84.14	75.57
WN-GN SE (WN held out)	32	213002	lhs	90.82	89.14
			rhs	91.56	88.76
			mean	293.16	1356.30
			median	5.0	5.10
			global	91.19	88.95
WN-GN SME-Bil(WN held out)	32	213002	lhs	82.42	73.65
			rhs	83.40	73.44
			mean	124.85	331.82
			median	11.0	33.86
			global	82.91	73.55
WN-GN SE (GN held out)	32	213002	lhs	81.82	70.56
			rhs	79.92	70.06
			mean	3031.44	15470.56
			median	7.0	10080.5
			global	80.87	70.313
WN-GN SME-Bil(GN held out)	32	213002	lhs	63.54	41.64
			rhs	64.78	70.32
			mean	984.79	1021.37
			median	40.0	428.90
			global	64.16	55.98
WordNet-GermaNet-DD (WN held out)	32	213002	lhs	77.06	64.98
			rhs	77.08	65.17
			mean	166.18	466.91
			median	18.0	55.41
			global	77.07	65.082
WordNet-GermaNet-DD (GN held out)	32	213002	lhs	57.72	38.06
			rhs	60.72	63.61
			mean	932.49	719.47
			median	56.0	175.56
			global	59.22	50.84

Table 2: Word-pair Similarity Performance for English

Dataset		WN-SE	WN-GN-SE	WN-SME-Bil	WN-GN-SME-Bil	WN-GN-SME-Bil-DD	HLBL	Turian*	Klementiev*
RG-65	P	0.682	0.666	0.703	0.833	0.725	-0.115	0.233	-0.380
	S	0.769	0.741	0.741	0.811	0.825	-.083	0.118	-0.398
MC-30	P	0.611	0.644	0.601	0.740	0.599	-0.363	0.150	-0.768
	S	0.720	0.648	0.756	0.846	0.954	-.450	-0.198	-0.522
WS-353	P	0.181	0.206	0.239	0.246	0.238	0.233	0.236	0.029
	S	0.093	0.146	0.185	0.224	0.201	0.197	0.210	0.040
YangPowers-130	P	0.482	0.637	0.584	0.627	0.610	-0.130	-0.076	0.154
	S	0.401	0.472	0.406	0.553	0.533	-0.186	-0.116	0.113

Table 3: Word-pair Similarity Performance for German

Dataset		GN-SE	WN-GN-SE	GN-SME-Bil	WN-GN-SME-Bil	WN-GN-SME-Bil-DD	Klementiev*
wortpaare222	P	-0.010	0.156	0.073	0.130	0.196	0.107
	S	-0.125	0.234	0.152	0.175	0.111	0.152
wortpaare30	P	0.865	0.984	0.185	0.287	0.301	-0.887
	S	1.0	1.0	-0.500	-0.500	0.500	-1.0
wortpaare350	P	-0.085	0.063	0.185	0.188	0.127	0.187
	S	-0.157	0.009	0.172	0.194	0.142	0.142
wortpaare65	P	0.800	0.558	-0.572	0.485	-0.166	0.233
	S	0.800	0.800	-0.8	0.399	0.200	0.200