Learning Word Representations from a Large-Scale Unified Lexical Semantic Resource

Abstract

Keywords:

1. Introdcution

In a large number of machine learning methods and its application to natural language processing, most of the labor is dedicated to *Feature Engineering*. Extracting informative features is the crucial part of most supervised methods and it is done mostly manually. While many different applications share common learning models and classifiers, the difference in performance of competing methods mostly goes to the data representation and hand-crafted features that they use. This observation reveals an important weakness in current models, namely their inability to extract and organize discriminative features from data. *Representation learning* is an umberella term for a family of unsupervised methods to learn features from data. Most of recent works on the application of this idea in NLP focus on inducing word representations. *Word representation* 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 [1]. Recently, it has been shown in [1] and [2] that using induced feautres can be helpful to improve state-of-the-art methods in variouse NLP tasks. In Section 2, some of these methods are discussed in more details. We can see that most of the current methods for inducing word representations can only exploit from surface relation among words. Indeed, the only resource for them to capture semanitcal and grammatical aspects of words is co-occuring of them in a text. The word embeddings learned in neural language models ([] and []) and brown clustering are examples of such approach. In the contrast to these methods, Bordes et al. [4] proposed a method to learn distributed representations from relational datasets with richer information. In their work, they are attempting to induce word embeddings from knowledge bases such as WordNet and Freebase. Their datasets include binary relations between left entity and right entity and each relation is instantiated from a different relation type. Since we are following their work, a detailed description of their work is presented in 2.4.

2. Related Work

2.1. Distributional Representation

In distributional semantics, the meaning of a word is expressed by the context that it appears in it [5]. Features that are used to represent the meaning of a word are other words in its neighborhood as it is so called the context. In some approaches like LDA and latent semantic analysis (LSA), the context is defined in the scope of a document rather than a window around a word. To represent word meanings in via distributional approach, one should start from count matrix (or zero-one co-occurence matrix) which each row represents a word and each column is a context. The representation can be limited to raw usage of the very same matrix or some transforms like *tf-idf* will be applied first. A further analysis over this matrix to extract more meaningful features is applying dimensionality reduction methods or clustering models to induce latent distributional representations. A similar clustering method to k-means is used in [6] to represent phrase and word meanings and brown clustering algorithm [7] has been shown to have impact on near to state-of-the-art NLP tasks [1].

2.2. Distributed Representation

Distributed representation has been introduced in the literature for the first time in [8] where Bengio et al. introduced a first language model based on deep learning methods[9]. Deep learning is learning through several layers of neural networks which each layer is responsible to learn a different concept and each concecpt is built over other more abstract concepts. In the deep learning society, any word representation that is induced with a neural network is called *Word Embedding.* In contrast to raw count matrix in distributional representations, word embeddings are low-dimensional, dense and real-valued vectors. The term, **'Distributed'**, in this context refers to the fact that exponential number of objects (clusters) can be modeled by word embeddings. Here we will see two famous models to induce for such representations. One family will use n-grams to learn word representation jointly with a language model and the other family learns the embedding from structured resources. (Cross-lingual Word Clusters for Direct Transfer of Linguistic Structure should be mentioned ???)

2.3. Neural Language Models

In [10], Weston and Collobert use a non-probabilistic and discriminative model to jointly learn word embeddings and a language model that can separate plausible n-grams from noisy ones. For each word in a n-gram, they combine the word embeddings and use it as positive example. They put noise in the n-gram to make negative examples and then train a neural network to learn to classify postive labels from negative ones. The parameters of neural network (neural language model) and word embedding values will be learned jointly by an optimization method called *Stochastic Gradient Descent* [11].

A hierarchical dsitributed language model (HLBL) proposed by Mnih and Hinton in [12] is another influential work on word embeddings. In this model a probabilistic linear neural network(LBL) will be trained to combine word embeddings in first $n-1$ words of a n-gram to predict the n_th word.

Weston-Collobert model and HLBL by Mnih and Hinton are evaluated in [1] in two NLP tasks: chunking and named entity recognition. With using word embeddings from these models combined with hand-crafted features, the performance of both tasks are shown to be improved.

2.4. Representation Learning from Knowledge Bases

Bordes et al. in [4] and [13] have attempted to use deep learning to induce word representations from lexical resources such as WordNet and knowledge bases (KB) like Freebase. In Freebase for example, each named entity is related to another entity by an instance of a specific type of relation. In [4], each entity is represented as a vector and each relation is decomposed to two matrices. Each of these matrices transform left and right-hand-side entities to a semantic space. Similarity of transformed entities indicates that the relation holds between the entities. A prediction task is defined to evaluate the embeddings. Given a relation and one of the entities, the task is to predict the missing entity. The high accuracy (99.2%) of the model on prediciton of training data shows that learnt representation highly captures attributes of the entities and relations in Freebase.

3. Our contribution

3.1. Bilingual word embeddings

3.2. Combining semantic resources

3.3. Uby

4. Empirical 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 good the structure of knowledge bases are represented through the learned features. After we learned the word embeddings from subset(??) of Uby(??), their ability to reproduce the structure of it will be assessed. On the other hand, the second setup is investigating on this question that if the word embeddings are able to improve the performance of (??) some standard NLP tasks or not. In this setup we will look to contribution of the learned features in . . . and . . . (??) in different settings.

4.1. Ranking

Bordes et al. define a ranking task where for each triplet (e_l, r, e_r) in trianing and test set, e_l will be removed and all the entities will be ranked by using 1-norm rank function (equation ??? decomposing equation). A higher rank of e_l (lower number) reflects the better quality of learned representations. Additionally they have compared this result to another ranking schema using density estimation . In this schema, for each word embedding e the density of (e, r, e_r) will be computed (as it is described in our section???) and triplets will be sorted by their estimated probability (probability terms ??). Since we are using larger sets of triplets, instead of ranking all the training instances we sample randomly from each training dataset with size of 20% of the original dataset(??) then we test our models on these sampled training instances and all the instances from test set. Bordes et al. have followed a similar approach for ranking their embeddings on their biggest dataset. We re-run their related experiments to make the comparison to our embeddings meaningful. Table (??) shows the results.

We run another experiment to further analyze the role of multi-task learning of embeddings for transforming knowledge from a language to another one. We repeat the ranking evaluation with two different embeddings: (1) learned from GermaNet (2) jointly learned from GermaNet-WordNet. This

Table 1: Ranking Performance

Dataset	num of relations		rank e_l	rank e_r
WordNet	5	training	1	1
		testing	1	1
GermaNet	5	training	1	1
		testing	1	1
WordNet-GermaNet	5	training	1	1
		testing	1	1
WordNet-FrameNet	5	training	1	1
		testing	1	1

time we test it on a same test set which contains only tripletes extracted from GermaNet because we are interested to examine if semantic information from English (WordNet) can be transfered to German (GermaNet). Table (??) presents the comparison of ranking tasks for mono-lingual and bilingual word embeddings.

Table 2: Bilingual transfer learning

Dataset	num of relations		rank e_l	rank e_r
GermaNet	5	training	1	1
		testing	1	1
WordNet-GermaNet	5	training	1	1
		testing	1	1

Discussion about results should come here. We can not write it until we have all the results.

4.2. Some application we should think of

5. Conclusion and Future Work

References

[1] J. Turian, L. Ratinov, Y. Bengio, Word representations: a simple and general method for semi-supervised learning, ACL (2010) 384–394.

[2] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa, Natural Language Processing (almost) from Scratch, Machine Learning Research 12 (2011) 2493–2537.

[3] Generative Models for Relational Structures, ???? URL: `http://videolectures.net/ssspr2010_hancock_gmr/`.

[4] A. Bordes, J. Weston, R. Collobert, Y. Bengio, Learning structured embeddings of knowledge bases, AAAI (2011) 301–306.

[5] Z. Harris, Distributional structure, Springer Netherlands, 1981. URL: `http://link.springer.com/chapter/10.1007/978-94-009-8467-7_1`.

[6] D. Lin, X. Wu, Phrase clustering for discriminative learning, in: ACL-AFNLP, 2009, pp. 1030–1038. URL: `http://dl.acm.org/citation.cfm?id=1690290`.

[7] P. Brown, P. Desouza, R. Mercer, Class-based n-gram models of natural language, Computational Linguistics 4 (1992) 467–479.

[8] Y. Bengio, R. Ducharme, A neural probabilistic language model, The Journal of Machine ... 3 (2003) 1137–1155.

[9] Y. Bengio, Learning deep architectures for AI, Foundations and Trends in Machine Learning (2009).

[10] R. Collobert, J. Weston, A unified architecture for natural language processing: deep neural networks with multitask learning, ... international conference on Machine learning (2008).

[11] L. Bottou, Large-scale machine learning with stochastic gradient descent, Proceedings of COMPSTAT'2010 (2010).

[12] A. Mnih, G. Hinton, A scalable hierarchical distributed language model, Advances in neural information processing systems (2009) 1–8.

[13] A. Bordes, X. Glorot, J. Weston, Y. Bengio, Joint learning of words and meaning representations for open-text semantic parsing, in: Artificial Intelligence and Statistics (AISTATS), volume 22, 2012. URL: `http://redirect.subscribe.ru/_/-/jmlr.csail.mit.edu/proceedings/papers/v22/`