Efficient Graph-based Word Sense Induction

Abstract

The correct resolution of multiple senses of a polysemous word is crucial for a lot of downstream NLP applications. In this work we propose a more efficient and interpretable way to perform word sense induction (WSI) by building a global non-negative vector embedding bases [1] (which are interpretable like topics) and clustering them for each polysemous word.

We then try to extend the success of this approach to the sentiment analysis task. We propose a novel method to jointly solve WSI and sentiment analysis: efficiently injecting sentiment information during the WSI stage in order to discover sentiment-aware senses of each word. Our experiments show that this semi-supervised method provides a more interpretable solution for the sentiment analysis problem.

Approach

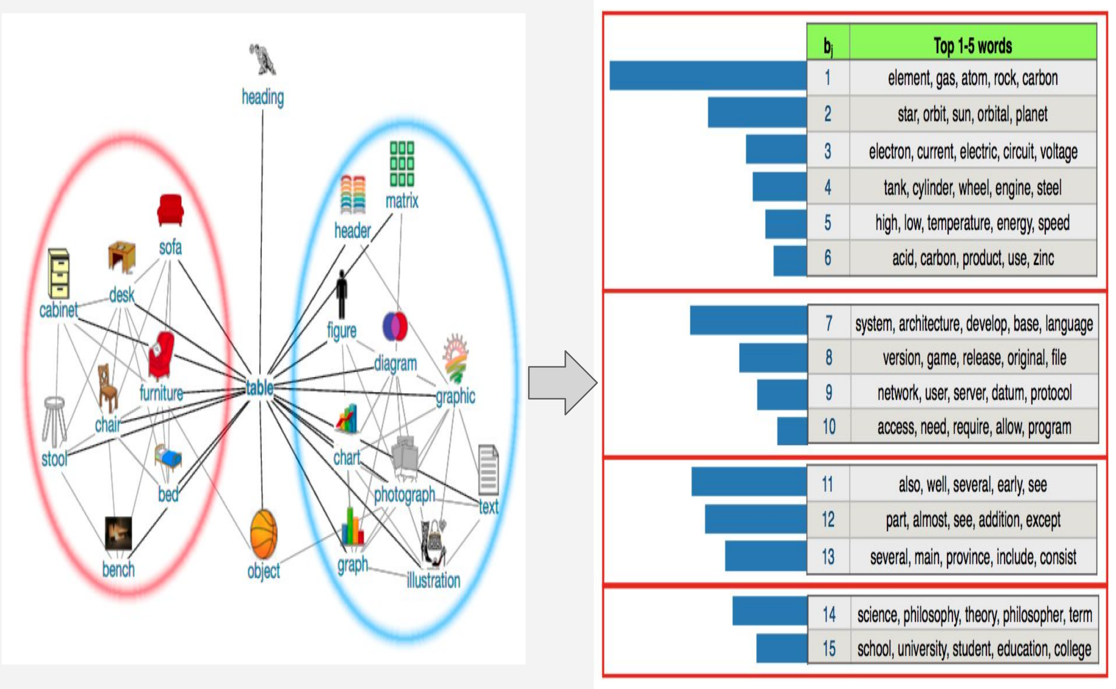

Fig. 1. Clustered DIVE topics. Ego network as shown in [2]

1. Generate *D* dimensional DIVE embeddings from the dataset
2. For each query word:
   a. Find relevant bases(topics)
   b. Use topical similarity for clustering
   c. Each cluster represents a sense
   d. Find a sense embedding per cluster
3. Improve these sense embeddings using EM over skip-gram training

$$SIM(b_i, b_j, q) = (1-w)\cos(\mathrm{f}_{(b_i,q)}, \mathrm{f}_{(b_j,q)})\cdot\log(\min(\mathrm{w}_q[b_i], \mathrm{w}_q[b_j])) + (w)\,sent\_sim(b_i, b_j, q)$$

b_i, b_j	basis referenced by index *i, j*
$w_q[b_i]$, $w_q[b_j]$	DIVE value for word *q* for basis b_i, b_j
cos(a, b)	cosine similarity betwn two feature vectors for word *q*
sent_sim(a, b, q)	sentiment similarity between two feature vectors for word *q*

Model

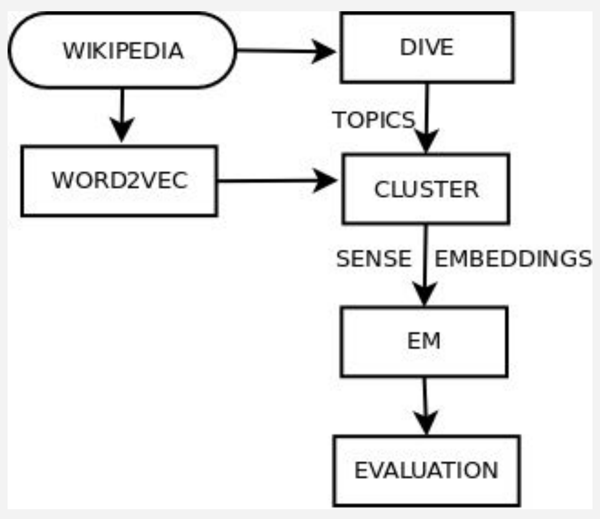

Fig 2. Pipeline of the WSD approach

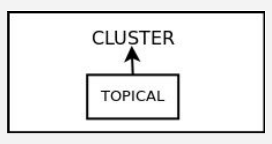

Fig 3: Clustering for WSD

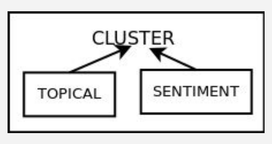

Fig 4: Clustering for Sentiment Analysis

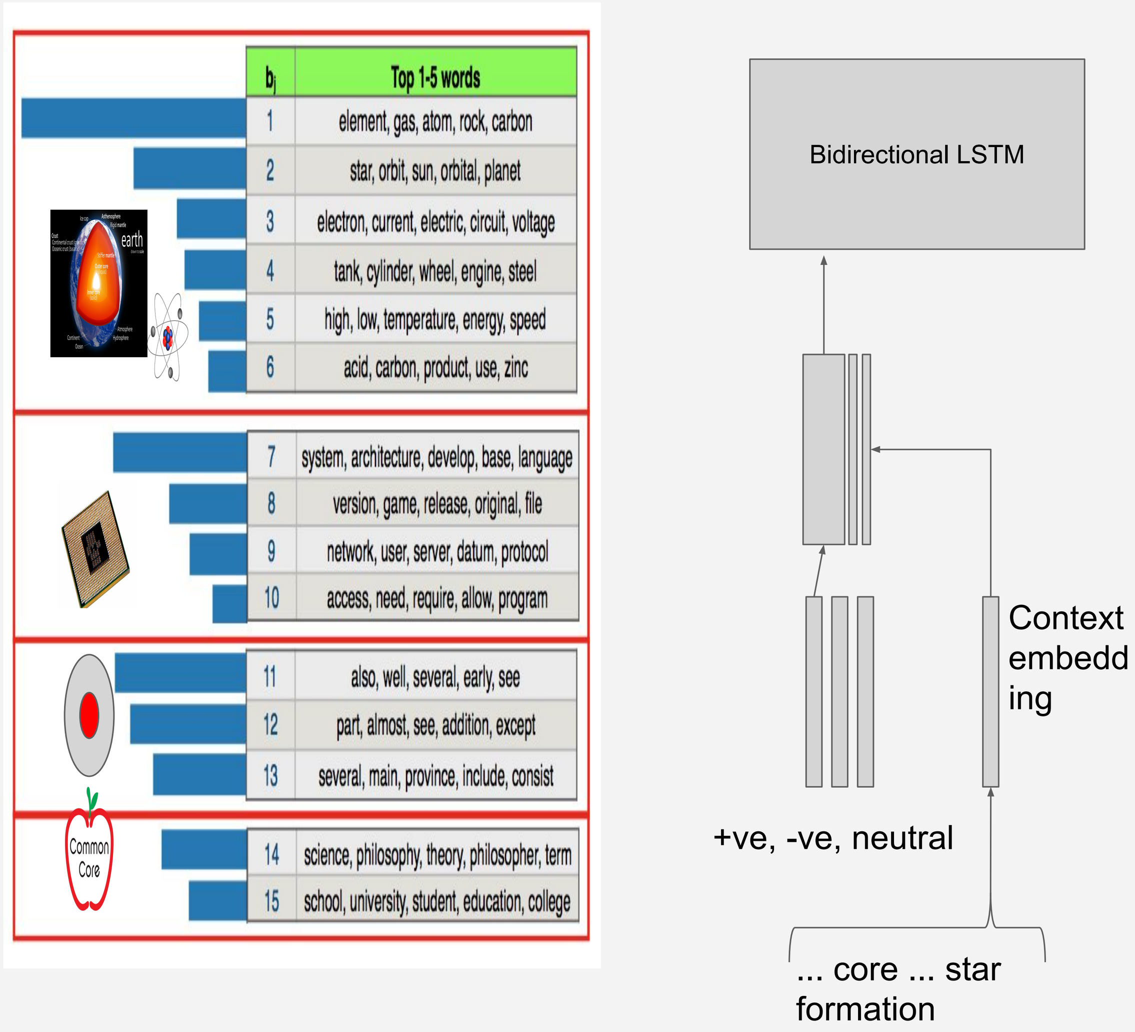

Fig 5. Sentiment Classification Pipeline

Results

WSD Results (on Wikipedia) [3]

Model	JI	Tau	WNDCG	FNMI	FB-C
All-1	19.2	60.9	28.8	0	**62.3**
Rnd	21.8	62.8	28.7	2.8	47.4
MSSG	**22.2**	**62.9**	29.0	3.2	48.9
WG	21.2	61.2	29.0	1.6	58.1
WG+EM	21.0	61.5	29.0	1.3	57.8
DIVE (100)	21.9	61.9	**29.3**	3.1	50.6
DIVE (300)	22.1	62.8	29.1	**3.5**	49.9

Table 1. Semeval 2013 task

Model	TWSI			balanced TWSI		
	P	R	F1	P	R	F1
MSSG rnd	66.1	65.7	65.9	33.9	33.7	33.8
MSSG	66.2	65.8	66.0	34.3	34.2	34.2
WG	**68.6**	**68.1**	**68.4**	38.7	38.5	38.6
WG+EM	68.3	67.8	68.0	38.4	38.2	38.3
DIVE rnd	63.4	63.0	63.2	33.4	33.2	33.3
DIVE (100)	67.6	67.2	67.4	**39.7**	**39.5**	**39.6**
DIVE (300)	67.4	66.9	67.2	39.0	38.8	38.9

Table 2 TWSI task

Sentiment Classifcation Results (on IMDb Sentiment Dataset)

Dataset	Random	Skip gram	**Senti_Dive**
20%	0.730	0.867	0.820
100%	0.840	0.880	0.843

Table 3: Accuracy of sentiment classification

Qualitative Results (on IMDb Sentiment Dataset)

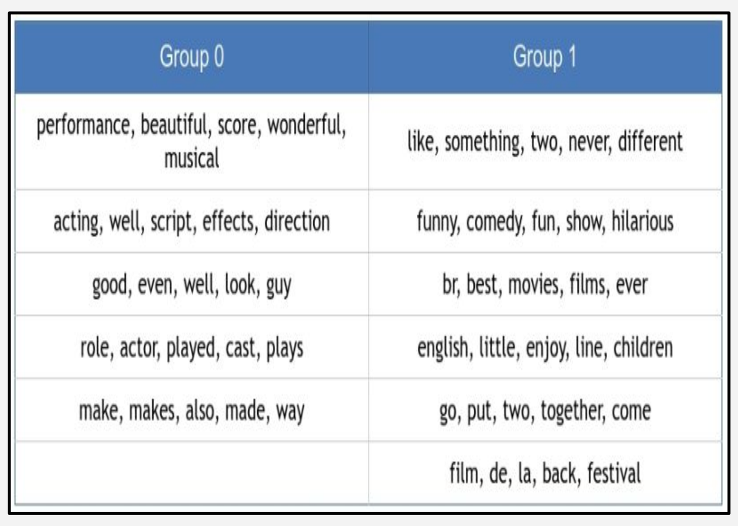

Fig 6. Clusters based only on WSD for the word *Great*

Fig 7. Clusters with induced sentiment for the word *Great*

Fig 8. Sentiment scores assigned to bases

Impact

- For the word sense disambiguation tasks, this approach betters the state of the art (SOTA) in some experiments while achieving comparable results in others.
- This approach is much more efficient than the previous SOTA approaches while being more interpretable.
- For the sentiment task while achieving comparable results this approach makes the process much more interpretable.

Future Work

- Experiment with explicit dynamic cluster count selection.
- Attention based end-to-end approach for sentiment analysis

References

1. “Unsupervised Hypernym Detection by Distributional Inclusion Vector Embedding”, *CoRR-2017*
2. M. Pelevina and N. Arefiev and C. Biemann and A. Panchenko, "Making sense of word embeddings." *ACL representation workshop 2017.*
3. H.-S. Chang, A. Agrawal, A. Ganesh, A. Desai, V. Mathur and A. McCallum, “Efficient Graph-based Word Sense Induction by Distributional Inclusion Vector Embeddings,” *TextGraphs 2018 (NAACL workshop).*