University of Stuttgart
Germany

Institut für Maschinelle Sprachverarbeitung

Integrating Distributional Lexical Contrast into Word Embeddings for Antonym–Synonym Distinction

Kim Anh Nguyen, Sabine Schulte im Walde, Ngoc Thang Vu

Institute for Natural Language Processing (IMS)
University of Stuttgart

ACL-2016, Berlin

Overview

1. Introduction
   1.1. Word Vector Representations
   1.2. Antonym-Synonym Distinction Task

2. Contributions
   2.1. Improving Weights of Feature Vectors
   2.2. Distributional Lexical Contrast Embeddings Model

3. Experiments

4. Conclusion

1.1. Word Vector Representations

1.1.1. Distributional Semantic Model (DSM)

- A means to represent meaning vectors of words.
- DSM rely on the *distributional hypothesis.* (Harris, 1954)
- Words with similar distributions have related meanings.
- Each weighted feature can be:
  - Co-occurrence frequency.
  - Association measure: local mutual information (LMI) (Evert, 2005)

1.1.2. Word Embeddings

- Representing words as low-dimensional dense vectors.
- Words with similar distributions have similar vectors.

1.2. Antonym-Synonym Distinction Task

- Goal
  - Distinguishing antonyms from synonyms.
- Problems
  - DSM tend to capture both antonyms (`formal-informal`) and synonyms (`formal-conventional`).
  - Word embeddings represent vectors of both antonyms and synonyms as similar vectors.
- Causes
  - Antonymy and synonymy are paradigmatic relations.
  - Antonyms and synonyms often occur in similar contexts.

Outline

1. Introduction
   1.1. Word Vector Representations
   1.2. Antonym-Synonym Distinction Task

2. Contributions
   2.1. Improving Weights of Feature Vectors
   2.2. Distributional Lexical Contrast Embeddings Model

3. Experiments

4. Conclusion

Institut für Maschinelle Sprachverarbeitung

2.1. Improving Weights of Feature Vectors

- Goal:
  - Improving the quality of weighted feature vectors.
- Solution:
  - Strengthening most salient features in the vectors.
  - Using the lexical contrast information of the target words and their contexts.
  - Proposing the new weight for feature vectors.
- Representing words based on DSM with positive LMI.
- For each target word w:
  - Determining the sets of antonyms $A(w)$ and synonyms $S(w)$.
  - Determining the set of shared words $W(f)$ for each feature f.
- Computing the new weight (called $weight^{SA}$) as follows:

2.1. Improving Weights of Feature Vectors

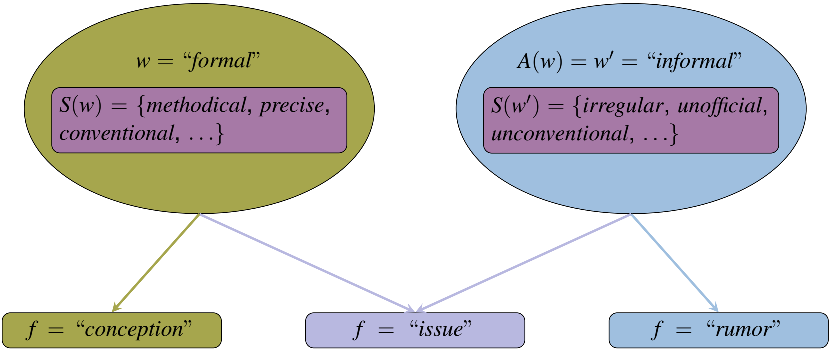
Average similarity to synonyms

$$weight^{SA}(w,f) = \frac{1}{\#(w,u)} \sum_{u \in W(f) \cap S(w)} sim(w,u)$$

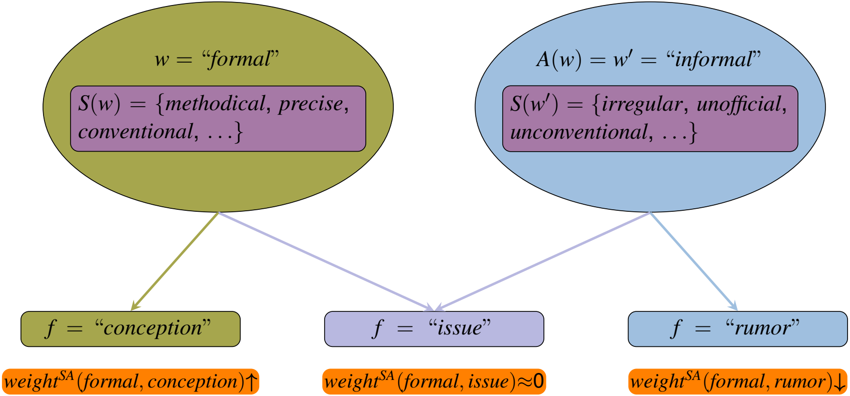
$$- \frac{1}{\#(w',v)} \sum_{w' \in A(w)} \sum_{v \in W(f) \cap S(w')} sim(w',v)$$

Average similarity to antonyms

2.1. Improving Weights of Feature Vectors



2.1. Improving Weights of Feature Vectors



$$weight^{SA}(w,f) = \frac{1}{\#(w,u)} \sum_{u \in W(f) \cap S(w)} sim(w,u) - \frac{1}{\#(w',v)} \sum_{w' \in A(w)} \sum_{v \in W(f) \cap S(w')} sim(w',v)$$

2.2. Distributional Lexical Contrast Embeddings Model (dLCE)

- Aims:
  - Learning word embeddings.
  - Moving synonyms closer to each other in space.
  - Moving antonyms further away from each other in space.
- Solution:
  - Integrating distributional lexical contrast into the transformation of Skip-gram model (Mikolov et al. 2013, Levy et al., 2014).
  - Applying lexical contrast to every single context of the target word.
- The proposed objective function as follows:

2.2. Distributional Lexical Contrast Embeddings Model (dLCE)

Distribution of target word and contexts

Distribution of negative contexts

$$\sum_{w \in V} \sum_{c \in V} \{ \boxed{ \left(\#(w,c) \log \sigma(sim(w,c)) + k\#(w) P_0(c) \log \sigma(-sim(w,c)) \right) }$$

$$\boxed{ + \left(\frac{1}{\#(w,u)} \sum_{u \in W(c) \cap S(w)} sim(w,u) - \frac{1}{\#(w,v)} \sum_{v \in W(c) \cap A(w)} sim(w,v) \right) } \}$$

Distribution of synonymous pairs

Distribution of antonymous pairs

Outline

1. Introduction
    1.1. Word Vector Representations
    1.2. Antonym-Synonym Distinction Task

2. Contributions
    2.1. Improving Weights of Feature Vectors
    2.2. Distributional Lexical Contrast Embeddings Model

3. Experiments

4. Conclusion

Institut für Maschinelle Sprachverarbeitung

3. Experiments

- Evaluating $weight^{SA}$ on Antonym-Syntonym distinction task.
- Evaluating effects of dLCE model:
  - Antonym-Syntonym distinction task.
  - Similarity task.

3.1. Antonym–Synonym Distinction

- Corpus: ENCOW14A (Schäfer and Bildhauer, 2012) contains 14.5 billion tokens.
- Dataset: a gold standard resource of paradigmatic relation pairs (Roth and Schulte im Walde, 2014)

Word Class	Ant-pairs	Syn-pairs	Total
Adjective	300	300	600
Noun	350	350	700
Verb	400	400	800

- Using average precision (AP) to evaluate.
- Using box-plots to compare the cosine medians of antonymous *vs.* synonymous pairs.

3.1. Antonym–Synonym Distinction

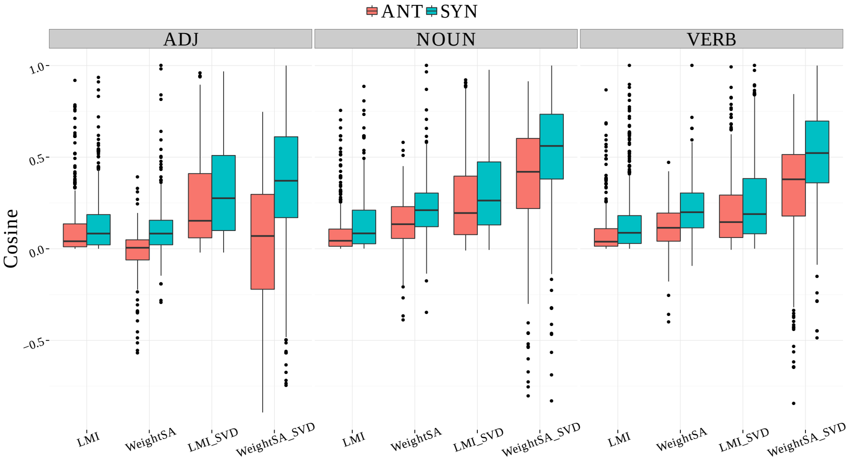
- AP evaluation results[1]:

	Adjectives		Nouns		Verbs	
	ANT	SYN	ANT	SYN	ANT	SYN
LMI	0.46	0.56	0.42	0.60	0.42	0.62
$weight^{SA}$	**0.36****	**0.75****	**0.40**	**0.66**	**0.38***	**0.71***
LMI + SVD	0.46	0.55	0.46	0.55	0.44	0.58
$weight^{SA}$ + SVD	**0.36*****	**0.76*****	**0.40***	**0.66***	**0.38*****	**0.70*****

[1] $\chi^2, {}^{***}p < .001, {}^{**}p < .005, {}^{*}p < .05$

3.1. Antonym–Synonym Distinction

- Results in box-plots:



3.2. Effects of dLCE model

3.2.1. Antonym-Synonym Distinction:

- Dataset: the gold standard resource of paradigmatic relation pairs (Roth and Schulte im Walde, 2014).
- Using area under curve (AUC) to identify antonyms.
- Comparison Models: Skip-gram (SGNS), mLCM (Pham et al., 2015)
- Results:

	Adjectives	Nouns	Verbs
SGNS	0.64	0.66	0.65
mLCM	0.85	0.69	0.71
dLCE	**0.90**	**0.72**	**0.81**

3.2. Effects of dLCE model

3.2.2. Similarity Task:

- Dataset: SimLex-999 (Hill et al., 2015)
- Using Spearman correlation coefficient ρ to evaluate.
- Comparison Models: SGNS, mLCM.
- Results:

SGNS	mLCM	dLCE
0.38	0.51	**0.59**

Outline

1. Introduction
   1.1. Word Vector Representations
   1.2. Antonym-Synonym Distinction Task
2. Contributions
   2.1. Improving Weights of Feature Vectors
   2.2. Distributional Lexical Contrast Embeddings Model
3. Experiments
4. Conclusion

4. Conclusion

- We have presented two methods to address the task of antonym-synonym distinction:
  - Improving the quality of weighted feature vectors.
  - Integrating distributional lexical contrast into word embeddings.
- The results from the experiments show that our approaches can model semantic similarity and distinguish between antonyms and synonyms.

Thank you!