1. Introduction

- Motivation: use output of coreference resolution system as a resource for semantic tasks
- Coreference chains: complementary properties compared to other resources, such as cooccurrence statistics, e.g., “cows” vs. “cattle” vs. “milk”
- Coreference-based similarity can be used as an additional feature for any task that distributional similarity is useful for (e.g., finding alternative names for entities, knowledge base population)
- Task here: detecting antonyms
  - Antonyms: distributionally similar but semantically dissimilar words
  - Distributional models often cannot distinguish them from synonyms

2. Word embeddings

2.1 Word-based and coreference-based embeddings

- Calculation of word embeddings with word2vec (skip-gram model) [Mikolov et al., 2013]
- text-based embeddings: calculated on raw text data (English Gigaword, LDC2012T21, Agence France-Press 2010)
- CoreNLP
- word2vec
- Antonym detection: SVM

2.2 Qualitative analysis of word vectors

- Illustration after t-SNE [Van der Maaten and Hinton, 2008]

2.3 Quantitative analysis of word vectors

- Split coreference resource into two parts (85% - 15%)
- First part: used for training embeddings
- Second part: used for computing cosine similarities for each possible word pair in the same coreference chain
- Results:
  - text-based vectors: 0.350 0.998 0.166
  - coref-based vectors: 0.318 0.999 0.161

3. Experiment: Antonym detection

3.1 Classification features

- Supervised classification with SVMs
- Features for SVM (to classify \( w \) and \( \neq w \) as antonyms or non-antonyms):
  1. Cosine similarity of text-based embeddings of \( w \) and \( \neq w \)
  2. Inverse rank of \( \neq w \) in the nearest text-based neighbors of \( w \)
  3. Cosine similarity of coreference-based embeddings of \( w \) and \( \neq w \)
  4. Inverse rank of \( \neq w \) in the nearest coreference-based neighbors of \( w \)
  5. Difference of (1) and (3)
  6. Difference of (2) and (4)
- Feature subsets for experiments: text-based (1-2), coreference-based (3-4), all (1-6)

3.2 Data set

- Set of word pairs: target word \( w \) and antonym candidate \( \neq w \)
- Possible target words: all word types of our vocabulary with at least one antonym in Merriam-Webster (www.merriam-webster.com)
- Target words and their antonyms: available at https://code.google.com/p/cistern
- Positive training examples: target word and one of its antonyms which is also one of its 500 nearest text-based neighbors
- Negative training examples: same target word with a random word of its 500 nearest text-based neighbors

3.3 Experimental results and discussion

- All word classification: slightly better performance for combination of all features
- Noun classification: using coreference-based features in addition to text-based features improves results
- Mined coreference chains provide complementary information to cooccurrence statistics

4. Conclusion

- Coreference-based word embeddings capture a type of semantic similarity that is complementary to the one captured by text-based embeddings
- Coreference-based embeddings improve performance on antonym classification by 0.09 F1