A Corpus to Learn Refer-to-as Relations for Nominals

Wasi Uddin Ahmad, and Kai-Wei Chang
{wasiahmad, kwchang}@cs.ucla.edu
Department of Computer Science, University of California, Los Angeles

Objective

Continuous representations of words and phrases should contain information for identifying refer-to-as relationship. In this work:

- We construct a corpus to learn continuous representations for nominals through which refer-to-as relations can be captured.
- We design a *mention* ranking task by simplifying the corefernece resolution task to evaluate the learned nominal embeddings.

Motivation

- Semantic representation of "phd candidate" and "graduate student" should indicate that they can be co-referred to each other.
- Refer-to-as relations can be resolved by taking help from knowledge source, ex., Wikipedia.

Nominal Coreference Example

"**A female motorist wearing a blue shirt** abruptly made a left turn, ignoring the officer's attempt to initiate a traffic stop. **The driver** continued to drive erratically to Annapolis Road."

- Both nominals, "A female motorist wearing a blue shirt" and "The driver" refer to the same entity.

Dataset Construction

- Each Wikipedia article is treated as an entity (or concept or idea), and the anchor text of in-links as a mention of the entity.
- Anchor texts are tagged using Stanford POS tagger and the *non-capitalized noun phrases* are considered as nominals.
- `https://github.com/wasiahmad/mining_wikipedia/tree/master/WikiMiner`

Learning Phrase Embeddings

- To evaluate the learned representation of noun phrases, we propose a ranking task:
  - Given a target mention and a list of candidate mentions, the goal is to rank the mentions in the candidate list based on how likely it is co-referred with the target mention without considering the context.
- We learned the phrase embeddings based on the following neural network architecture:
  - We use a bidirectional LSTM to learn word representations (contextualized) and use a CNN to construct phrase representations.
- Embeddings of the target mention and one of the candidate mentions are concatenated and passed through a feed-forward neural network to compute the similarity score.

Corpus Statistics

Number of articles	16,388,870
Number of redirected articles	6,466,828
Number of non-redirected articles	9,922,042
Unique noun mentions	26,660,798
Unique nominal mentions	2,512,347
Unique nominal mentions ($1 \le$ mention length ≤ 30)	1,428,441

Table 1:Corpus description extracted from Wikipedia

Train (src: Wikipedia)	Total nominal coref. chain	78,665
	Avg. candidates per chain	24
	Total unique terms	35,939
Development (src: Wikipedia)	Total nominal coref. chain	8,354
	Avg. candidates per chain	18
	Total unique terms	6,686
Test (src: CoNLL)	Total nominal coref. chain	623
	Avg. candidates per chain	12
	Total unique terms	2,839

Table 2:Data Description

Coreference Clusters generated from Wikipedia

Target Mention	Positive Candidates	Negative Candidates
protein sequence	amino acid sequencing, chain of amino acids, peptide sequence, protein primary structure	metabolic enzymes, biological mutations, periodic sequence, nucleotide sequence
general election	whole coalition, upcoming election, the previous election, election campaign, legislative election	the constitutional amendment, election win, the presidential election, democratic political values
aerial bomb	aerial bombardment, bombing, bomb attack	nuclear bomb technology, terror attacks, attack ground targets, atomic weapon
highway construction	roads, road building equipment, road work construction, street construction, road building	highway marker, construction yard, railway and highway bridge, construction superintendent

Table 3:Example of positive and negative coreference clusters generated from Wikipedia

Baseline Results

- Mention Embeddings: $\mathrm{Sim}(p_1, p_2) = \mathrm{cosine}(E(p_1), E(p_2))$ where $E(p) = \frac{1}{n_p}\sum_{k=1}^{n_p} w_k$ and $p = w_1, \ldots, w_{n_p}$
- Mention Embeddings + FFNN: $\mathrm{Sim}(p_1, p_2) = \sigma(u^T \tanh(W[E(p_1), E(p_2)] + b))$
  where $W \in R^{d_e \times d_e}, b, u \in R^{d_e}$, and $[E(p_1), E(p_2)]$ represents concatenation of the phrase embedding pair.
- Bidirectional-LSTM + CNN + FFNN: A BiLSTM followed by a CNN is used to form phrase vectors and a FFNN is used to compute the similarity score.

Model	NLL-Loss	MAP	P@1	P@5	R@1	R@5
Mention Embeddings	1.7389	**0.5452**	**0.5185**	0.2374	**0.3715**	0.7630
Mention Embeddings + FFNN	1.7836	0.4632	0.4995	0.2317	0.3516	0.7888
Bidirectional-LSTM + CNN + FFNN	**1.6731**	0.4884	0.4719	**0.2475**	0.3476	**0.8025**

Table 4:Performance of baseline methods.

Conclusion

In order to learn representations which can capture the refer-to-as relationship between nominals, we propose a corpus extracted from Wikipedia.

References

[1] P. Denis and J. Baldridge.
Specialized models and ranking for coreference resolution.
In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 660–669, 2008.

[2] S. J. Wiseman, A. M. Rush, S. M. Shieber, and J. Weston.
Learning anaphoricity and antecedent ranking features for coreference resolution.
In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2015.

Acknowledgements

This work was supported in part by National Science Foundation Grant IIS-1760523 and an NVIDIA Hardware Grant.