Negative Sampling Improves Hypernymy Extraction Based on Projection Learning

Dmitry Ustalov†, Nikolay Arefyev§, Chris Biemann‡, and Alexander Panchenko‡

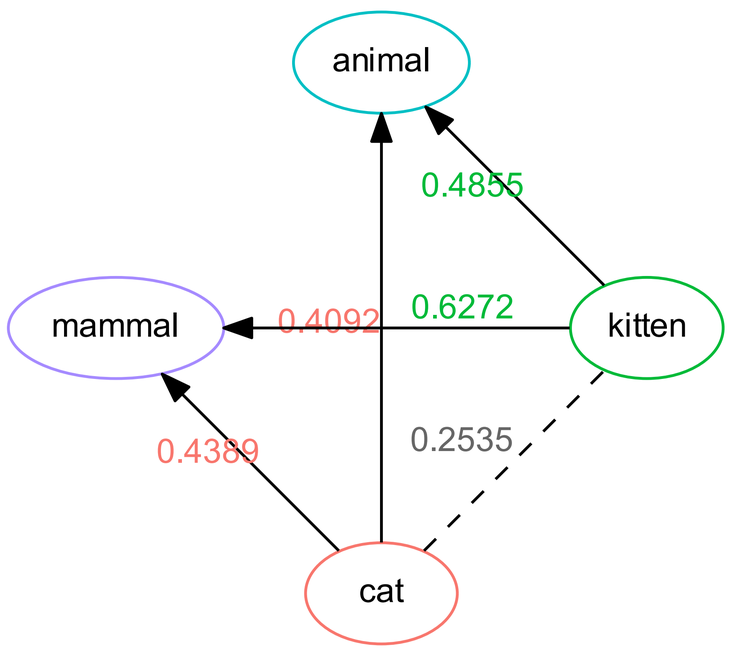
† Ural Federal University, Russia
§ Moscow State University, Russia
‡ University of Hamburg, Germany

Introduction

We present a new approach to extraction of hypernyms based on projection learning and word embeddings. In contrast to classification-based approaches, projection-based methods require no candidate hyponym-hypernym pairs. We show that **explicit negative examples used for regularization of the model significantly improve performance** compared to the state-of-the-art approach of Fu et al. (2014) on three datasets from different languages.

Key Ideas

- Hypernymy is an asymmetric relation.
- Regularization enforces the linguistic constraints.
- Negative sampling is used in the loss function.



Evaluation

We adopted the hit@l measure proposed by Frome et al. (2013) which was originally used for image tagging.

For each subsumption pair $(\boldsymbol{x}, \boldsymbol{y})$ composed of the hyponym $\boldsymbol{x}$ and the hypernym $\boldsymbol{y}$ in the test set $\mathcal{P}$, we compute l nearest neighbors for the projected hypernym $\boldsymbol{x}\boldsymbol{\Phi}^*$.

The pair is considered matched if the gold hypernym $\boldsymbol{y}$ appears in the computed list of the l nearest neighbors $\mathrm{NN}_l(\boldsymbol{x}\boldsymbol{\Phi}^*)$. To obtain the quality score, we average the matches in the test set $\mathcal{P}$:

$$\text{hit@}l = \frac{1}{|\mathcal{P}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{P}} \mathbb{1}\big(\boldsymbol{y} \in \mathrm{NN}_l(\boldsymbol{x}\boldsymbol{\Phi}^*)\big),$$

where $\mathbb{1}(\cdot)$ is the indicator function. To consider also the rank of the correct answer, we compute the area under curve measure as the area under the $l-1$ trapezoids:

$$\text{AUC} = \frac{1}{2}\sum_{i=1}^{l-1}(\text{hit@}(i) + \text{hit@}(i+1)).$$

In our experiments, we use the model of Fu et al. (2014) as the baseline: $\lambda = 0$.

Acknowledgments

We acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) foundation under the "JOIN-T" project, the Deutscher Akademischer Austauschdienst (DAAD), the Russian Foundation for Basic Research (RFBR) under the project no. 16-37-00354 mol_a, and the Russian Foundation for Humanities under the project no. 16-04-12019 "RussNet and YARN thesauri integration". We also thank Microsoft for providing computational resources under the Microsoft Azure for Research award. Finally, we are grateful to Benjamin Milde, Andrey Kutuzov, Andrew Krizhanovsky, and Martin Riedl for discussions and suggestions related to this study.

Hypernymy Extraction via Regularized Projection Learning

The projection matrix $\boldsymbol{\Phi}^*$ is obtained similarly to the linear regression problem, i.e., for the given row word vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ representing correspondingly hyponym and hypernym, the square matrix $\boldsymbol{\Phi}^*$ is fit on the training set of positive pairs $\mathcal{P}$:

$$\boldsymbol{\Phi}^* = \arg\min_{\boldsymbol{\Phi}} \frac{1}{|\mathcal{P}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{P}} \|\boldsymbol{x}\boldsymbol{\Phi} - \boldsymbol{y}\|^2 + \lambda R,$$

where $|\mathcal{P}|$ is the number of training examples, $\|\boldsymbol{x}\boldsymbol{\Phi} - \boldsymbol{y}\|$ is the distance between a pair of row vectors $\boldsymbol{x}\boldsymbol{\Phi}$ and $\boldsymbol{y}$, and λ is the constant controlling the importance of the regularization term R. In the original method, the L^2 distance is used. To improve performance, k projection matrices $\boldsymbol{\Phi}$ are learned one for each cluster of relations in the training set. One example is represented by a hyponym-hypernym offset. Clustering is performed using the k-means algorithm.

Asymmetric Regularization. As hypernymy is an asymmetric relation, our first method enforces the asymmetry of the projection matrix. Applying the same transformation to the predicted hypernym vector $\boldsymbol{x}\boldsymbol{\Phi}$ should not provide a vector similar $(\cdot)$ to the initial hyponym vector $\boldsymbol{x}$. Note that, this regularizer requires only positive examples $\mathcal{P}$:

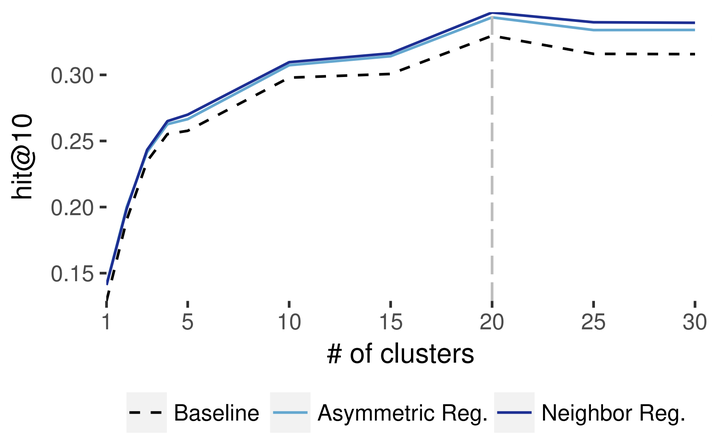
$$R = \frac{1}{|\mathcal{P}|} \sum_{(\boldsymbol{x},\_)\in\mathcal{P}} (\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi} \cdot \boldsymbol{x})^2.$$

Neighbor Regularization. This approach relies on the negative sampling by explicitly providing the examples of semantically related words $\boldsymbol{z}$ of the hyponym $\boldsymbol{x}$ that penalizes the matrix to produce the vectors similar to them:

$$R = \frac{1}{|\mathcal{N}|} \sum_{(\boldsymbol{x},\boldsymbol{z})\in\mathcal{N}} (\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi} \cdot \boldsymbol{z})^2.$$

Note that this regularizer requires negative samples $\mathcal{N}$. In our experiments, we use synonyms of hyponyms as $\mathcal{N}$, but other types of relations can be also used such as antonyms, meronyms or co-hyponyms. Certain words might have no synonyms in the training set. In such cases, we substitute $\boldsymbol{z}$ with $\boldsymbol{x}$, gracefully reducing to the previous variation. Otherwise, on each training epoch, we sample a random synonym of the given word.

Results: Russian (skip-gram, 500 dimensions, RDT)



The dataset is composed of the Russian Wiktionary and a set of subsumptions extracted using the Hearst patterns from `lib.rus.ec`. We use the optimal $k = 20$ tuned on the validation set.

Model	hit@1	hit@5	hit@10	AUC
Baseline	0.209	0.303	0.323	2.665
Asym. Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	0.213	0.300	0.322	2.659
Asym. Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	0.212	0.312	0.334	2.743
Neig. Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	**0.214**	0.304	0.325	2.685
Neig. Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	0.211	**0.315**	**0.338**	**2.768**

Results: English (skip-gram, 300 dimensions, Google News)

		EVALution				EVALution, BLESS, K&H+N, ROOT09				
Model	k	**hit@1**	**hit@5**	**hit@10**	**AUC**	k	**hit@1**	**hit@5**	**hit@10**	**AUC**
Baseline	1	0.109	0.118	0.120	1.052	1	0.104	0.247	0.290	2.115
Asymmetric Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	1	0.116	0.125	0.132	1.140	1	0.132	0.256	0.292	2.204
Asymmetric Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	1	0.145	0.166	0.173	1.466	1	0.112	**0.266**	0.314	2.267
Neighbor Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	1	0.134	0.141	0.150	1.280	1	**0.134**	0.255	0.306	2.267
Neighbor Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	1	**0.148**	**0.168**	**0.177**	**1.494**	1	0.111	0.264	**0.316**	**2.273**
Baseline	30	0.327	0.339	0.350	3.080	25	0.546	0.614	0.634	5.481
Asymmetric Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	30	0.336	0.354	0.366	3.201	25	0.547	0.616	0.632	5.492
Asymmetric Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	30	0.341	0.364	0.368	3.255	25	**0.553**	0.621	**0.642**	5.543
Neighbor Reg. $\boldsymbol{x}\boldsymbol{\Phi}$	30	0.339	0.357	0.364	3.210	25	0.547	0.617	0.634	5.494
Neighbor Reg. $\boldsymbol{x}\boldsymbol{\Phi}\boldsymbol{\Phi}$	30	**0.345**	**0.366**	**0.370**	**3.276**	25	**0.553**	**0.623**	0.641	**5.547**

References

[1] R. Fu et al. Learning Semantic Hierarchies via Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209, 2014.

[2] A. Panchenko, O. Morozova, and H. Naets. A Semantic Similarity Measure Based on Lexico-Syntactic Patterns. In *Proceedings of KONVENS 2012*, pages 174–178, 2012.

[3] A. Frome et al. DeViSE: A Deep Visual-Semantic Embedding Model. In *Advances in Neural Information Processing Systems 26*, pages 2121–2129, 2013.

[4] A. Panchenko et al. Human and Machine Judgements for Russian Semantic Relatedness. In *Analysis of Images, Social Networks and Texts: 5th International Conference, Revised Selected Papers*, AIST 2016, pages 221–235, 2017.