

# Path-based vs. Distributional Information in Recognizing Lexical Semantic Relations

Vered Shwartz

Computer Science Department, Bar-Ilan University, Ramat-Gan, Israel  
vered1986@gmail.com

Ido Dagan

dagan@cs.biu.ac.il

## Abstract

Recognizing various semantic relations between terms is beneficial for many NLP tasks. While path-based and distributional information sources are considered complementary for this task, the superior results the latter showed recently suggested that the former’s contribution might have become obsolete. We follow the recent success of an integrated neural method for hypernymy detection (Shwartz et al., 2016) and extend it to recognize multiple relations. The empirical results show that this method is effective in the multiclass setting as well. We further show that the path-based information source always contributes to the classification, and analyze the cases in which it mostly complements the distributional information.

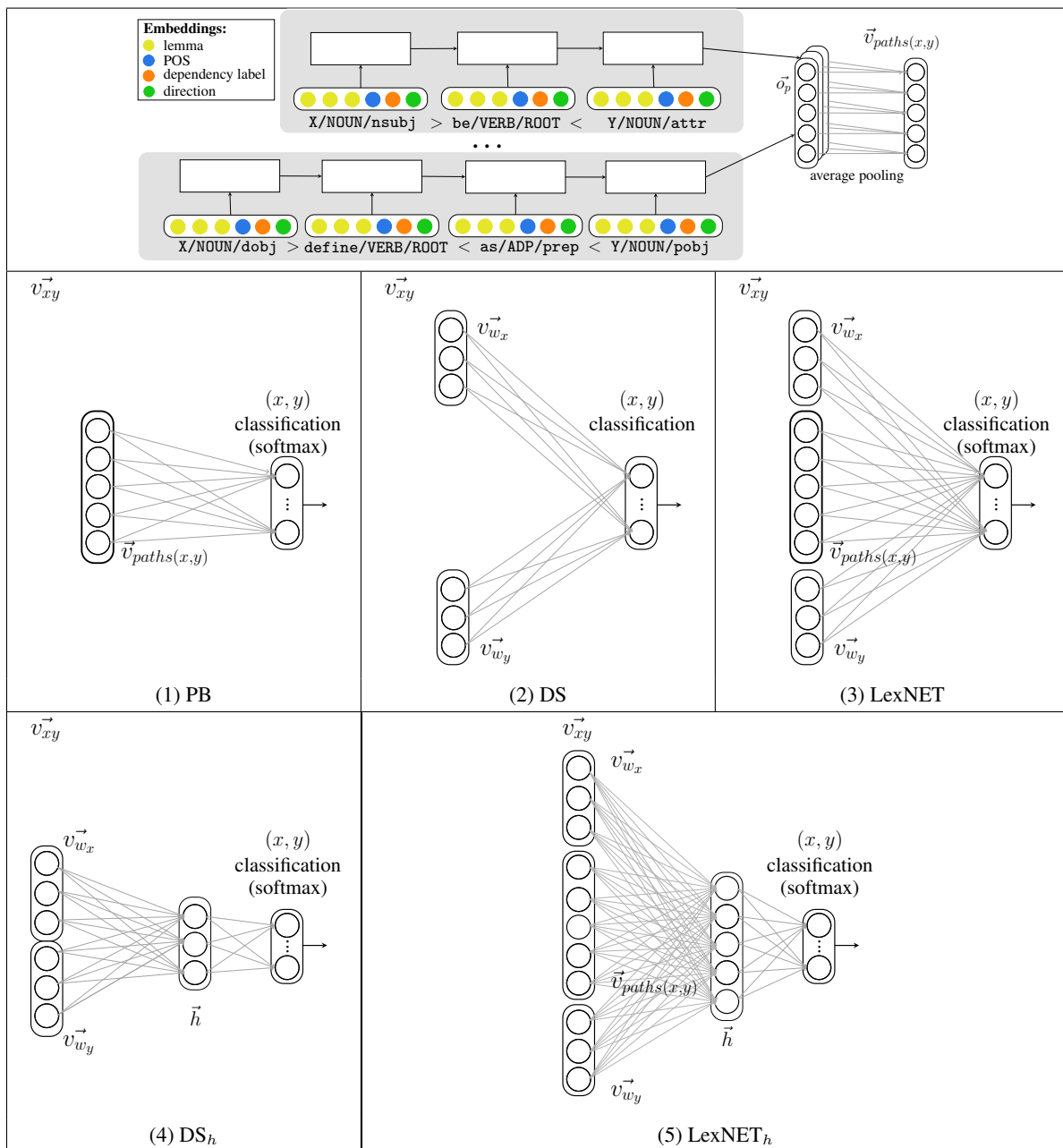
## 1 Introduction

Automated methods to recognize the lexical semantic relation the holds between terms are valuable for NLP applications. Two main information sources are used to recognize such relations: path-based and distributional. Path-based methods consider the *joint* occurrences of the two terms in a given pair in the corpus, where the dependency paths that connect the terms are typically used as features (A. Hearst, 1992; Snow et al., 2004; Nakashole et al., 2012; Riedel et al., 2013). Distributional methods are based on the *disjoint* occurrences of each term and have recently become popular using word embeddings (Mikolov et al., 2013; Pennington et al., 2014), which provide a distributional representation for each term. These embedding-based methods were reported to perform well on several common datasets (Baroni et al., 2012; Roller et al., 2014), consistently outperforming other methods (Santus et al., 2016; Necsulescu et al., 2015).

While these two sources have been considered complementary, recent results suggested that path-based methods have no marginal contribution over the distributional ones. Recently, however, Shwartz et al. (2016) presented `HyPeNET`, an integrated path-based and distributional method for hypernymy detection. They showed that a good path representation can provide substantial complementary information to the distributional signal in hypernymy detection, notably improving results on a new dataset.

In this paper we present `LexNET`, an extension of `HyPeNET` that recognizes *multiple* semantic relations. We show that this integrated method is indeed effective also in the multiclass setting. In the evaluations reported in this paper, `LexNET` performed better than each individual method on several common datasets. Further, it was the best performing system in the semantic relation classification task of the CogALex 2016 shared task (Shwartz and Dagan, 2016).

We further assess the contribution of path-based information to semantic relation classification. Even though the distributional source is dominant across most datasets, path-based information always contributed to it. In particular, path-based information seems to better capture the relationship between terms, rather than their individual properties, and can do so even for rare words or senses. Our code and data are available at <https://github.com/vered1986/LexNET>.



**Figure 1:** Illustrations of classification models. Top row: path-based component. A path is a sequence of edges, and each edge consists of four components: lemma, POS, dependency label and direction. Edge vectors are fed in sequence into the LSTM, resulting in an embedding vector  $\vec{o}_p$  for each path.  $\vec{v}_{paths(x,y)}$  is the average of  $(x, y)$ 's path embeddings.

## 2 Background: HypeNET

Recently, Shwartz et al. (2016) introduced HypeNET, a hypernymy detection method based on the integration of the best-performing distributional method with a novel neural path representation, improving upon state-of-the-art methods. In HypeNET, a term-pair  $(x, y)$  is represented as a feature vector, consisting of both distributional and path-based features:  $\vec{v}_{xy} = [\vec{v}_{w_x}, \vec{v}_{paths(x,y)}, \vec{v}_{w_y}]$ , where  $\vec{v}_{w_x}$  and  $\vec{v}_{w_y}$  are  $x$  and  $y$ 's word embeddings, providing their distributional representation, and  $\vec{v}_{paths(x,y)}$  is a vector representing the dependency paths connecting  $x$  and  $y$  in the corpus. A binary classifier is trained on these vectors, yielding  $c = \text{softmax}(W \cdot \vec{v}_{xy})$ , predicting hypernymy if  $c[1] > 0.5$ .

Each dependency path is embedded using an LSTM (Hochreiter and Schmidhuber, 1997), as illustrated in the top row of Figure 1. This results in a path vector space in which semantically-similar paths (e.g. *X is defined as Y* and *X is described as Y*) have similar vectors. The vectors of all the paths that connect  $x$  and  $y$  are averaged to create  $\vec{v}_{paths(x,y)}$ .

dataset	dataset relations	#instances
K&H+N	hypernym, meronym, co-hyponym, random	57,509
BLESS	hypernym, meronym, co-hyponym, , event, attribute, random	26,546
ROOT09	hypernym, co-hyponym, random	12,762
EVALution	hypernym, meronym, attribute, synonym, antonym, holonym, substance meronym	7,378

**Table 1:** The relation types and number of instances in each dataset, named by their WordNet equivalent where relevant.

Shwartz et al. (2016) showed that this new path representation outperforms prior path-based methods for hypernymy detection, and that the integrated model yields a substantial improvement over each individual model. While HypeNET is designed for detecting hypernymy relations, it seems straightforward to extend it to classify term-pairs simultaneously to multiple semantic relations, as we describe next.

### 3 Classification Methods

We experiment with several classification models, as illustrated in Figure 1:

**Path-based** HypeNET’s path-based model ( $\text{PB}$ ) is a binary classifier trained on the path vectors alone:  $\vec{v}_{\text{paths}(x,y)}$ . We adapt the model to classify multiple relations by changing the network softmax output  $c$  to a distribution over  $k$  target relations, classifying a pair to the highest scoring relation:  $r = \text{argmax}_i c[i]$ .

**Distributional** We train an SVM classifier on the concatenation of  $x$  and  $y$ ’s word embeddings  $[\vec{v}_{w_x}, \vec{v}_{w_y}]$  (Baroni et al., 2012) ( $\text{DS}$ ).<sup>1</sup> Levy et al. (2015) claimed that such a linear classifier is incapable of capturing interactions between  $x$  and  $y$ ’s features, and that instead it learns separate properties of  $x$  or  $y$ , e.g. that  $y$  is a *prototypical hypernym*. To examine the effect of non-linear expressive power on the model, we experiment with a neural network with a single hidden layer trained on  $[\vec{v}_{w_x}, \vec{v}_{w_y}]$  ( $\text{DS}_h$ ).<sup>2</sup>

**Integrated** We similarly adapt the HypeNET integrated model to classify multiple semantic relations ( $\text{LexNET}$ ). Based on the same motivation of  $\text{DS}_h$ , we also experiment with a version of the network with a hidden layer ( $\text{LexNET}_h$ ), re-defining  $c = \text{softmax}(W_2 \cdot \vec{h} + b_2)$ , where  $\vec{h} = \tanh(W_1 \cdot \vec{v}_{xy} + b_1)$  is the hidden layer. The technical details of our network are identical to Shwartz et al. (2016).

### 4 Datasets

We use four common semantic relation datasets that were created using semantic resources:  $\text{K\&H+N}$  (Neculescu et al., 2015) (an extension to Kozareva and Hovy (2010)),  $\text{BLESS}$  (Baroni and Lenci, 2011),  $\text{EVALution}$  (Santus et al., 2015), and  $\text{ROOT09}$  (Santus et al., 2016).

Table 1 displays the relation types and number of instances in each dataset. Most dataset relations are parallel to WordNet relations, such as hypernymy (*cat, animal*) and meronymy (*hand, body*), with an additional *random* relation for negative instances.  $\text{BLESS}$  contains the *event* and *attribute* relations, connecting a concept with a typical activity/property (e.g. (*alligator, swim*) and (*alligator, aquatic*)).  $\text{EVALution}$  contains a richer schema of semantic relations, with some redundancy: it contains both meronymy and holonymy (e.g. for *bicycle* and *wheel*), and the fine-grained substance-holonymy relation. We removed two relations with too few instances: *Entails* and *MemberOf*.

To prevent the lexical memorization effect (Levy et al., 2015), Santus et al. (2016) added negative switched hyponym-hypernym pairs (e.g. (*apple, animal*), (*cat, fruit*)) to  $\text{ROOT09}$ , which were reported to reduce this effect.

### 5 Results

Like Shwartz et al. (2016), we tuned the methods’ hyper-parameters on the validation set of each dataset, and used Wikipedia as the corpus. Table 2 displays the performance of the different methods on all datasets, in terms of recall, precision and  $F_1$ .<sup>3</sup>

Our first empirical finding is that Shwartz et al.’s (2016) algorithm is effective in the multiclass setting as well ( $\text{LexNET}$ ). The only dataset on which performance is mediocre is  $\text{EVALution}$ , which seems to be

<sup>1</sup>We experimented also with difference  $\vec{v}_{w_x} - \vec{v}_{w_y}$  and other classifiers, but concatenation yielded the best performance.

<sup>2</sup>This was previously done by Bowman et al. (2014), with promising results, but on a small artificial vocabulary.

<sup>3</sup>Additional evaluation of the method is available in our CogALex 2016 shared task submission (Shwartz and Dagan, 2016).

method	K&H+N			BLESS			ROOT09			EVALution		
	P	R	$F_1$	P	R	$F_1$	P	R	$F_1$	P	R	$F_1$
PB	0.713	0.604	0.55	0.759	0.756	0.755	0.788	0.789	0.788	0.53	0.537	0.503
DS	0.909	0.906	0.904	0.811	0.812	0.811	0.636	0.675	0.646	0.531	0.544	0.525
DS <sub>h</sub>	0.983	0.984	0.983	0.891	0.889	0.889	0.712	0.721	0.716	0.57	0.573	0.571
LexNET	0.985	0.986	0.985	0.894	<b>0.893</b>	<b>0.893</b>	<b>0.813</b>	0.814	0.813	<b>0.601</b>	<b>0.607</b>	<b>0.6</b>
LexNET <sub>h</sub>	0.984	0.985	0.984	<b>0.895</b>	0.892	<b>0.893</b>	0.812	<b>0.816</b>	<b>0.814</b>	0.589	0.587	0.583

**Table 2:** Performance scores (precision, recall and  $F_1$ ) of each individual approach and the integrated models. To compute the metrics we used scikit-learn (Pedregosa et al., 2011) with the “averaged” setup, which computes the metrics for each relation, and reports their average, weighted by support (the number of true instances for each relation). Note that it can result in an  $F_1$  score that is not the harmonic mean of precision and recall.

dataset	#pairs	x	y	gold label	DS <sub>h</sub> prediction	possible explanation
K&H+N	102	firefly	car	false	hypo	( <i>x, car</i> ) frequent label is hypo
		racehorse	horse	hypo	false	out of the embeddings vocabulary
		larvacean	salp	sibl	false	rare terms <i>larvacean</i> and <i>salp</i>
BLESS	275	tanker	ship	hyper	event	( <i>x, ship</i> ) frequent label is event
		squirrel	lie	random	event	( <i>x, lie</i> ) frequent label is event
		herring	salt	event	random	non-prototypical relation
ROOT09	562	toaster	vehicle	RANDOM	HYPER	( <i>x, vehicle</i> ) frequent label is HYPER
		rice	grain	HYPER	RANDOM	( <i>x, grain</i> ) frequent label is RANDOM
		lung	organ	HYPER	COORD	polysemous term <i>organ</i>
EVALution	235	pick	metal	MadeOf	IsA	rare sense of <i>pick</i>
		abstract	concrete	Antonym	MadeOf	polysemous term <i>concrete</i>
		line	thread	Synonym	MadeOf	( <i>x, thread</i> ) frequent label is MadeOf

**Table 3:** The number of term-pairs that were correctly classified by the integrated model while being incorrectly classified by DS<sub>h</sub>, in each test set, with corresponding examples of such term-pairs.

inherently harder for all methods, due to its large number of relations and small size. The performance differences between LexNET and DS are statistically significant on all datasets with p-value of 0.01 (paired t-test). The performance differences between LexNET and DS<sub>h</sub> are statistically significant on BLESS and ROOT09 with p-value of 0.01, and on EVALution with p-value of 0.05.

DS<sub>h</sub> consistently improves upon DS. The hidden layer seems to enable interactions between  $x$  and  $y$ ’s features, which is especially noticed in ROOT09, where the hypernymy  $F_1$  score in particular rose from 0.25 to 0.45. Nevertheless, we did not observe a similar behavior in LexNET<sub>h</sub>, which worked similarly or slightly worse than LexNET. It is possible that the contributions of the hidden layer and the path-based source over the distributional signal are redundant.<sup>4</sup> It may also be that the larger number of parameters in LexNET<sub>h</sub> prevents convergence to the optimal values given the modest amount of training data, stressing the need for large-scale datasets that will benefit training neural methods.

## 6 Analysis

Table 2 demonstrates that the distributional source is dominant across most datasets, with DS performing better than PB. Although by design DS does not consider the relation between  $x$  and  $y$ , but rather learns properties of  $x$  or  $y$ , it performs well on BLESS and K&H+N. DS<sub>h</sub> further manages to capture relations at the distributional level, leaving the path-based source little room for improvement on these two datasets.

On ROOT09, on the other hand, DS achieved the lowest performance. Our analysis reveals that this is due to the switched hypernym pairs, which drastically hurt the ability to memorize individual single words, hence reducing performance. The  $F_1$  scores of DS on this dataset were 0.91 for co-hyponyms but only 0.25 for hypernyms, while PB scored 0.87 and 0.66 respectively. Moreover, LexNET gains 10 points over DS<sub>h</sub>, suggesting the better capacity of path-based methods to capture relations between terms.

### 6.1 Analysis of Information Sources

To analyze the contribution of the path-based information source, we examined the term-pairs that were correctly classified by the best performing integrated model (LexNET/LexNET<sub>h</sub>) while being incorrectly classified by DS<sub>h</sub>. Table 3 displays the number of such pairs in each dataset, with corresponding term-pair examples. The common errors are detailed below:

<sup>4</sup>We also tried adding a hidden layer only over the distributional features of LexNET, but it did not improve performance.

**Lexical Memorization**  $DS_h$  often classifies  $(x, y)$  term-pairs according to the most frequent relation of one of the terms (usually  $y$ ) in the train set. The error is mostly prominent in `ROOT09` (405/562 pairs, 72%), as a result of the switched hypernym pairs. However, it is not infrequent in other datasets (47% in `BLESS`, 43% in `EVALution` and 34% in `K&H+N`). As opposed to distributional information, path-based information pertains to both terms in the pair. With such information available, the integrated model succeeds to overcome the most frequent label bias for single words, classifying these pairs correctly.

**Non-prototypical Relations**  $DS_h$  might fail to recognize non-prototypical relations between terms, i.e. when  $y$  is a less-prototypical relatum of  $x$ , as in *mero:(villa, guest)*, *event:(cherry, pick)*, and *attri:(piano, electric)*. While being overlooked by the distributional methods, these relations are often expressed in joint occurrences in the corpus, allowing the path-based component to capture them.

**Rare Terms** The integrated method often managed to classify term-pairs in which at least one of the terms is rare (e.g. *hyper:(mastodon, proboscidean)*), where the distributional method failed. It is a well known shortcoming of path-based methods that they require informative co-occurrences of  $x$  and  $y$ , which are not always available. With that said, thanks to the averaged path representation,  $PB$  can capture the relation between terms even if they only co-occur once within an informative path, while the distributional representation of rare terms is of lower quality. We note that the path-based information of  $(x, y)$  is encoded in the vector  $\vec{v}_{paths(x,y)}$ , which is the averaged vector representation of all paths that connected  $x$  and  $y$  in the corpus. Unlike other path-based methods in the literature, this representation is indifferent to the total number of paths, and as a result, even a single informative path can lead to successful classification.

**Rare Senses** Similarly, the path-based component succeeded to capture relations for rare senses of words where  $DS_h$  failed, e.g. *mero:(piano, key)*, *event:(table, draw)*. Distributional representations suffer from insufficient representation of rare senses, while  $PB$  may capture the relation with a single meaningful occurrence of the rare sense with its related term. At the same time, it is less likely for a polysemous term to co-occur, in its non-related senses, with the candidate relatum. For instance, paths connecting *piano* to *key* are likely to correspond to the keyboard sense of *key*, indicating the relation that does hold for this pair with respect to this rare sense.

Finally, we note that `LexNET`, as well as the individual methods, perform poorly on synonyms and antonyms. The synonymy  $F_1$  score in `EVALution` was 0.35 in `LexNET` and in  $DS_h$  and only 0.09 in  $PB$ , reassessing prior findings (Mirkin et al., 2006) that the path-based approach is weak in recognizing synonyms, which do not tend to co-occur.  $DS_h$  performed poorly also on antonyms ( $F_1 = 0.54$ ), which were often mistaken for synonyms, since both tend to occur in the same contexts. It seems worthwhile to try improving the model using insights from prior work on these specific relations (Santus et al., 2014; Mohammad et al., 2013) or additional information sources, like multilingual data (Pavlick et al., 2015).

## 7 Conclusion

We presented an adaptation to `HypeNET` (Shwartz et al., 2016) that classifies term-pairs to one of multiple semantic relations. Evaluation on common datasets shows that `HypeNET` is extensible to the multi-class setting and performs better than each individual method.

Although the distributional information source is dominant across most datasets, it consistently benefits from path-based information, particularly when finer modeling of inter-term relationship is needed.

Finally, we note that all common datasets were created synthetically using semantic resources, leading to inconsistent behavior of the different methods, depending on the particular distribution of examples in each dataset. This stresses the need to develop “naturally” distributed datasets that would be drawn from corpora, while reflecting realistic distributions encountered by semantic applications.

## Acknowledgments

This work was partially supported by an Intel ICRI-CI grant, the Israel Science Foundation grant 880/12, and the German Research Foundation through the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1).

## References

- Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*.
- Marco Baroni and Alessandro Lenci. 2011. Proceedings of the gems 2011 workshop on geometrical models of natural language semantics. pages 1–10.
- Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Proceedings of EACL 2012*, pages 23–32.
- Samuel R Bowman, Christopher Potts, and Christopher D Manning. 2014. Learning distributed word representations for natural logic reasoning. *AAAI*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Zornitsa Kozareva and Eduard Hovy. 2010. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of EMNLP 2010*, pages 1110–1118.
- Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. Do supervised distributional methods really learn lexical inference relations? In *Proceedings of NAACL-HLT 2015*, pages 970–976.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.
- Shachar Mirkin, Ido Dagan, and Maayan Geffet. 2006. Integrating pattern-based and distributional similarity methods for lexical entailment acquisition. In *Proceedings of COLING/ACL 2006*, pages 579–586.
- Saif M Mohammad, Bonnie J Dorr, Graeme Hirst, and Peter D Turney. 2013. Computing lexical contrast. *Computational Linguistics*, 39(3):555–590.
- Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: A taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference EMNLP and CoNLL*, pages 1135–1145.
- Silvia Neculescu, Sara Mendes, David Jurgens, Núria Bel, and Roberto Navigli. 2015. Reading between the lines: Overcoming data sparsity for accurate classification of lexical relationships. In *Proceedings of \*SEM 2015*, pages 182–192.
- Ellie Pavlick, Johan Bos, Malvina Nissim, Charley Beller, Benjamin Van Durme, and Chris Callison-Burch. 2015. Adding semantics to data-driven paraphrasing. In *Proceedings of ACL 2015 (Volume 1: Long Papers)*, pages 1512–1522.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of EMNLP 2014*, pages 1532–1543.
- Sebastian Riedel, Limin Yao, Andrew McCallum, and M. Benjamin Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of NAACL-HLT 2013*, pages 74–84.
- Stephen Roller, Katrin Erk, and Gemma Boleda. 2014. Inclusive yet selective: Supervised distributional hypernymy detection. In *Proceedings of COLING 2014*, pages 1025–1036.
- Enrico Santus, Qin Lu, Alessandro Lenci, and Churen Huang. 2014. Unsupervised antonym-synonym discrimination in vector space.
- Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Proceedings of the 4th workshop on linked data in linguistics: Resources and applications. pages 64–69.
- Enrico Santus, Alessandro Lenci, Tin-Shing Chiu, Qin Lu, and Chu-Ren Huang. 2016. Nine features in a random forest to learn taxonomical semantic relations. In *LREC*.
- Vered Shwartz and Ido Dagan. 2016. Cogalex-v shared task: Lexnet - integrated path-based and distributional method for the identification of semantic relations. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex-V)*.
- Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of ACL 2016 (Volume 1: Long Papers)*, pages 2389–2398.
- Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. In *NIPS*.