Proceedings of the 7th Workshop on Linked Data in Linguistics (LDL-2020), pages 75–81
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Supervised Hypernymy Detection in Spanish through Order Embeddings

Gun Woo Lee, Mathias Etcheverry, Daniel Fernández Sánchez, Dina Wonsever
InCo, Fing, Universidad de la República
Montevideo, Uruguay
{gun.woo.lee, mathiase, daniel.fernandez.sanchez, wonsever}@fing.edu.uy

Abstract

This paper addresses the task of supervised hypernymy detection in Spanish through an order embedding and using pretrained word vectors as input. Although the task has been widely addressed in English, there is not much work in Spanish, and according to our knowledge there is not any available dataset for supervised hypernymy detection in Spanish. We built a supervised hypernymy dataset for Spanish using WordNet and corpus statistics, with different versions according to the lexical intersection between its partitions: random and lexical split. We show the results of using the resulting dataset within an order embedding consuming pretrained word vectors as input. We show the ability of pretrained word vectors to transfer learning to unseen lexical units according to the results in the lexical split dataset. To finish, we study the results of giving additional information in training time, such as, co-hyponymy links and instances extracted through lexico-syntactic patterns.

Keywords: Hypernymy Detection in Spanish, Order Embedding, Word Embedding

1. Introduction

Hierarchical organizations are key in language semantics. Hypernymy refers to the general-specific relationship between two lexical terms. Such is the case of biology taxonomies (e.g. mammal-vertebrate, pangolin-mammal), seasons (e.g. spring-season) and colors (e.g. green-color), among many others. The general term is called the hypernym and the specific one the hyponym. In natural language processing, automatic hypernymy detection (or taxonomy learning) is an active NLP research area, that has applications in several tasks such as question answering (Clark et al., 2007), textual entailment (Chen et al., 2017) and image detection (Marszalek and Schmid, 2007).

A well known hand-crafted resource is WordNet (Miller, 1995). It is a large lexical database that contains semantic relations, including hypernymy among them. Manual resources consume a considerable human effort for its creation and maintenance, and suffer from incompleteness and inadequacies. Furthermore, different applications require the expansion of the hypernymy relationship to particular instances like celebrities, song names, movies, and so on. Hence, it is clear the importance of automatic mechanisms to overcome or assist manual ones.

Regarding Spanish, the resources available for supervised hypernymy detection are quite scarce. WordNet was originally created for English and later translated into other languages, among which is Spanish (Atserias et al., 2004). This consists in the main source of hypernyms for Spanish. *Hypernymy detection* has been evaluated mainly through binary classification relying on datasets that contain a number of pairs of terms and a label for each pair indicating if hypernymy relation is held between the terms (Shwartz et al., 2016).

A complementary evaluation benchmark for modeling hypernymy is given by *hypernymy discovery* (Espinosa-Anke et al., 2016). It consists on given a domain's vocabulary and an input term, discover its hypernyms. This formulation is beneficial to avoid the lexical memorization phenomena (Levy et al., 2015). Regarding to hypernymy discovery, a dataset in Spanish (among other languages) was introduced for the task 9 of SemEval-2018 (Camacho-Collados et al., 2018).

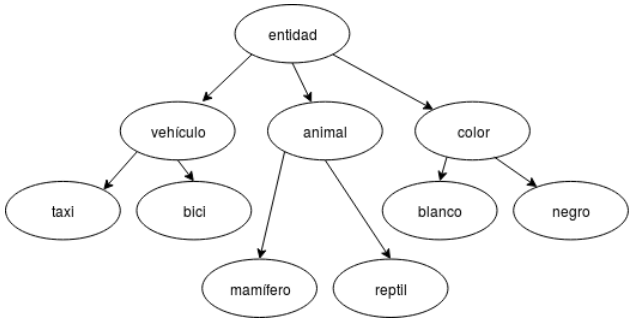

Figure 1: Example of a very simplified taxonomy in Spanish.

In this work we does not pursuit hypernymy discovery and we are aware that it is not clear how realistic hypernymy detection is, since in many scenarios the potential pairs may not be given and need to be discovered. However, we believe that a dataset for hypernymy detection in Spanish can be useful for model comparisons, and according to our knowledge there is no such resource available for Spanish at the time of this work.

We introduce a dataset for supervised hypernymy detection for Spanish built using Spanish WordNet and corpus statistics. We describe its creation process and we made it available to the NLP community as a complementary benchmark for hypernymy detection in Spanish. In addition, we train and evaluate using the created dataset an order embedding (Vendrov et al., 2015) based model using pretrained word embeddings as input, and we report the obtained results for future comparisons. Also, we show that this model, disregarding the use of Hearst patterns, outperforms other distributional approaches and the much more complex hybrid LSTM-based model, that combines distributional and path-based information, proposed by Shwartz et al. (2016).

2. Related Works

Hypernymy detection in NLP can be focused as a supervised or an unsupervised learning task. Supervised approaches relies on pairs annotated with the information of whether they belong to the relationship or not. On the contrary, unsupervised approaches do not use annotated instances, they rely solely in the distributional inclusion hypothesis (Zhitomirsky-Geffet and Dagan, 2005) or entropy based measures (Santus et al., 2014).

Supervised approaches have been addressed mainly using two types of information: paths and contexts distributions (or word embeddings). Path-based (or pattern-based) approaches use the paths of words that connect instances of the relationship. Hearst (1992) presents the first path-based approach where hand-crafted patterns were used for hypernymy extraction. For example, the path "is a type of" would match cases like "tuna is a type of fish" allowing to detect that "tuna" is an hyponym of "fish", etc. Also, paths of joint occurrences in syntactic dependency trees result useful for hypernymy detection (Snow et al., 2004). Path patterns were generalized using part-of-speech tags and ontology types by Nakashole et al. (2012). A different kind of pattern-based approach is proposed in the work of Navigli and Velardi (2010), they consider word lattices to extract definitional sentences in texts and then extract hypernymy related pairs from them, or learning lexical taxonomies (Navigli et al., 2011). The main disadvantage of path-based approaches is that both candidates must occur simultaneously in the same context.

In the other hand, the distributional approaches relies in the contexts of each word independently. Many methods propose supervised classification after applying a binary vector operation on the pair of representations, such as vector concatenation (Baroni et al., 2012) and difference (Roller et al., 2014; Fu et al., 2014; Weeds et al., 2014). Vylomova et al. (2016) studied vector difference behavior in a wider set of lexical relations and they remarked the importance of negative training data to improve the results. Ustalov et al. (2017) performed hypernyms extraction based on projection learning. Instead of classifying the pair of representations, they learned a mapping to project hyponyms embeddings to their respective hypernyms, remarking also the importance of negative sampling. A related approach is presented by Dash et al. (2019), where a neural network architecture is designed to enforce asymmetry and transitivity through non-linearities and residual connection. These last two approaches present some overlap with the work of Vendrov et al. (2015), that its order embedding approach is the one considered in this work.

Shwartz et al. (2016) combined path-based and distributional information in supervised hypernymy detection, concatenating the embedding of each term independently with a distributional representation of all paths between the terms in a dependency parsed corpus. The representation was built with the average of the LSTM resulting representation of each path. Additionally, they introduced a dataset for lexical entailment where they tested their model.

LEAR (Lexical Entailment Attract-Repel) (Vulic and Mrksic, 2017) gives great performance on hypernymy detection specializing word embeddings based on WordNet constraints. The direction of the asymmetric relation was encoded in the resulting vector norms while cosine distance jointly enforces synonyms semantic similarity. The resulting vectors were specialized simultaneously for lexical relatedness and entailment.

3. Hypernymy Dataset for Spanish

In this section we describe the dataset construction process. The dataset consists of pairs of words and a boolean label associated to each pair that is true when the first element is an hyponym of the second and false otherwise. We will refer as positive instances to those pairs that are labelled as true (e.g. summer-season) and as negative instances to those that are labelled as false (e.g. cat-fish).

In the dataset construction process we use a variety of sources to obtain positive and negative instances. In the following we describe each source and technique used; and we give a measure of the quality of the dataset based on a random sampling.

In addition and based on the dataset built by Shwartz et al. (2016), we performed a random split (in train, validation and test) and a split without terms occurring in more than one partition to deal with the lexical memorization (Levy et al., 2015). The latter is referred as lexical split.

3.1. Related Pairs

The extraction of positive pairs was performed using Spanish WordNet, patterns against a Spanish Corpus, and Shwartz dataset translation.

In addition to these sources, it is possible to consider the transitive links as positive instances, since the hypernym relation fulfills the transitive property. However, this assumption may not be satisfied when different senses are faced in the transitive link. So, we decided to not consider inferred transitive instances in this work, and the dataset discard word sense information.

In the following we describe how we use each source:

- Spanish WordNet:
  The main source of positive instances of our dataset is the Spanish version of the WordNet of the Open Multilingual Wordnet (OMW). We consider the hypernymy relation defined in WordNet between synsets, and then we perform a selection of pairs, taking one word of each synset, to obtain hypernymic pairs that will belong to the dataset.

  We considered the following two heuristics:

  1. We choose from each synset those words that are most frequently used according to its frequency in the corpus of Cardellino (2016)[1].
  2. Based on Santus et al. (2014) work, we filtered the resulting candidate pairs that the hyponyms has a frequency greater than the frequency of it proposed hypernym.

[1] Spanish Billion Word Corpus and Embeddings by Cristian Cardellino: https://crscardellino.github.io/SBWCE/

k	Size (# pairs)	% Correct
1	15695 / 10103	83.9 / 84.3
2	29180 / 19258	82.2 / 83.3
3	35103 / 22851	77.6 / 83.5

Table 1: Size and percentage of correct hypernyms of a sample of the resulting pairs considering 1, 2 and 3 most frequent words of each synset. We show the results applying (right) and without applying (left) the second heuristic filtering.

Regarding the first heuristic, we observe the result of considering the pairs from an all-vs-all of the k most frequent lemmas of each synset. In table 1 we report the respective sizes and percentage of correct pairs of a 0.5% random sample, where can be observed that taking into account more than the two most frequent words of each synset the results degrade considerably.

We filter the output of the first heuristic using the second heuristic and we observe a quality improvement in the resulting pairs. The values on the right in table 1 details the obtained results. According to this minimal evaluation criterion we decide to consider the most three frequent words of each synset filtering the pairs where the hyponym is more frequent than the hypernym.

To finish with WordNet extracted hypernyms, we eliminate the cycles that are generated due to the multiple senses of certain words and the transitivity of the hypernym relation. The resulting pairs are the final set of the WordNet positve instances of the dataset.

- Pattern-based:
  Relying on the well known importance of the pattern (or path) based approaches to detect and discover hypernyms, originated by Hearst (1992), we consider to include in our dataset positive instances extracted using high confidence patterns. We consider the following two patterns for Spanish built by Ortega et al. (2011) they found to present a high confidence in their experiments (confidence value near to 1):

  1. "el <hyponym> es el único <hyperonym>"
  2. "de <hyponym> y otras <hyperonym>"

  We use these patterns to extract candidate pairs from the corpus of Cardellino (2016). Unfortunately, the quality of the resulting pairs was poor. Subsequently, we achieve a little improvement filtering the obtained candidates using the part of speech. Even so, we did not obtain good enough results to be included in the final dataset. However, we consider that despite the poor quality the extracted instances, it may become useful to study the behavior of including them as training data. For that purpose it is available along with the dataset.

- Shwartz dataset translation:
  In the dataset built by Shwartz et al. (2016), they obtained the hypernymy relation instances from English WordNet, DBPedia, Wikidata and Yago. Their dataset contains a considerable number of instances like shakespeare-writer. Therefore, we consider to select those pairs that contain proper names as hyponym candidate. We limit our selection to the instances of: "village", "city", "company", "town", "place", "river" and "person"; and we translate the instances through Google's translation library. We include the resulting candidates as positive instances in our dataset.

3.2. Unrelated Pairs

The unrelated pairs, or negative instances, are those pairs that does not hold an hypernymic relation between them. We consider for the procurement of unrelated pairs the following approaches:

- Random sampling:
  Since most of the words are not hypernym between them, we can randomly pick two words from a given vocabulary and we probably will get a non hypernymic pair. So, we obtain the noun words from the Cardellino's Corpus, with at least 4 characters and a frequency greater than to 200, jointly with the vocabulary of the positive part, above mentioned, of the dataset. Then we proceed to generate tuples, that were not already included in the dataset, till complete the desired ratio of 1:3 of positive:negative instances.

  The dataset resulting of WordNet, Shwartz translation and random pairs is what we refer as our base dataset, presented in its two versions: random and lexical split, as we will detail later.

- Cohyponyms:
  Cohyponymy is the relation between hyponyms that share the same hypernym. They are words that have properties in common, but which in turn have their own characteristics that differentiate them well from each other. Cohyponymy can be seen as words belonging to a same class (e.g male-female, march-november). Given a pair of cohponyms it is highly probably that an hypernymy relation is not fullfilled between them. Therefore, it is possible to obtain negative pairs from cohyponymic relations entailed from the positive instances.

- Inverted links:
  The hypernym relation is asymmetric. Therefore, if a tuple satisfies the hypernym relation, its inverse not. Then, having our positive dataset already, a simple way to build negative dataset is exchanging the order of the pairs of the positive dataset. However, synonyms may become a problem in this assumption. We can think between some synonyms that an hypernymic relation is fulfilled in both directions (e.g. neat-tidy). For this reason we does not include inverted links in the distributed dataset.

- Antonymy:
  Words that have an opposite meaning are called Antonyms. We assume that if there is an antonymy relationship, the hypernym relationship is not satisfied. Therefore, we include the antonyms extracted from WordNet as negative instances.

Positive Pairs		
WordNet	Pattern-based	Shwartz
27861	2731	3798

Negative Pairs			
Random	Cohyponym	Antonym	Meronym
~ 90000	~ 45000	1107	5940

Table 2: Total amount of positive and negative instances from where each version of the dataset is built.

3.3. Dataset Splits

As usual in supervised training, we split the whole dataset (positive and negative pairs) into train, validation and test partitions. Following the work of Shwartz et al. (2016), we consider two splits of the data: random and lexical split. While the random split is performed randomly, the lexical split does not allow lexical intersection between the partitions. In the following section we describe each one.

3.3.1. Random Split

The random split consists in splitting the dataset randomly, without taking into account any consideration. We perform a random split with the following ratio: 70 % for training set, 25 % for test set and 5 % for validation set.
This splitting process has the advantage that any tuple is discarded, leading to a larger dataset, but may suffer of the phenomena of lexical memorization (Levy et al., 2015). The lexical memorization phenomenon occurs when different pairs of hypernym, instead of learning the semantic relationship between words, learn a specific word independently as a strong indicator of the label. For example, given the positive pairs such as: (cat, animal), (dog, animal), (horse, animal), the algorithm tends to learn that the word "animal" is a "prototype" and given any new (x, animal) classifies it as a positive pair.

3.3.2. Lexical Split

To avoid the phenomenon of lexical memorization, the training, validation, and test sets are split with different vocabularies. We split the dataset with the same methodology of (Shwartz et al., 2016). The approximate division ratio was 70-25-5. The respective sizes of the random and lexical splits of our base dataset are shown in Table 3.

		Train	Val	Test	Total
Rnd. Split	P	18654	1332	6662	106592
	N	55962	3996	19986	
Lex. Split	P	8221	513	2506	44960
	N	24663	1539	7518	

Table 3: Spanish dataset sizes for each split: lexical and random. The sizes are discriminated in terms of positive (P) and negative (N) instance. This sizes does not contain cohyponyms or pattern extracted positive instances.

4. Experiments using Order Embeddings

To automatically detect hypernymy we consider a simple feed forward network trained as an order embeddings (Vendrov et al., 2015). This network takes the word embedding to a non negative vector with a partial order relation defined and trained to take hypernym pairs to related vectors.
In this work we show that without path or any additional information than the proper word embedding of each word, and a feed forward network trained as above mentioned, fairly good results can be achieved.
We first give an introduction to the order embedding proposal and our experiments configuration.

4.1. Order Embedding Model

An order embedding is a function between two partially ordered sets $f : (X, \preceq_X) \to (Y, \preceq_Y)$ that preserves and reflects its order relationships. That is to say, $x_1 \preceq_X x_2$ if and only if $f(x_1) \preceq_Y f(x_2)$.
Vendrov et al. (2015) introduce a method to train an order embedding into $\Re^m_{\geq 0}$ considering the *reversed product order*, defined as follows:

$$x \preceq y \iff \bigwedge_{i=1}^{m} x_i \geq y_i, \tag{1}$$

where $x, y \in \Re^m_{\geq 0}$ and x_i and y_i correspond to the i-th component of x and y, respectively. By definition this relationship is antisymmetric and transitive, being $\vec{0}$ the top element of the hierarchy.

4.1.1. Contrastive Loss Function

The partial order relation $(\preceq, \Re^m_{\geq 0})$ defined above allows to define measures to quantify the degree to which a pair of two elements does not satisfy the relationship. Let us consider

$$E_p(\vec{x}, \vec{y}) = ||max(\vec{0}, \vec{y} - \vec{x})||^2, \tag{2}$$

where $\vec{x}, \vec{y} \in \Re^m_+$ and max is the maximum function element-wise. Note that E_p indicates the relation satisfaction degree and $E_p(x, y) = 0$ iff $\vec{x} \preceq \vec{y}$.
Then, E_p can be forced to be higher than a threshold α for unrelated terms through the max-margin loss as follows:

$$E_n(\vec{x}, \vec{y}) = max\{0, \alpha - E_p(\vec{x}, \vec{y})\}, \tag{3}$$

guaranteeing that $E_n(\vec{x'}, \vec{y'})$ is 0 when $E_p(\vec{x'}, \vec{y'}) \geq \alpha$ and therefor $\vec{x'}\vec{y'}$.
Then, summing (2) and (3) the resulting contrastive loss function, which consists of minimizing E_p and E_n jointly, stands as follows:

$$L = \sum_{(x,y)\in P} E_p(\vec{x}, \vec{y}) + \sum_{(x',y')\in N} E_n(\vec{x'}, \vec{y'}), \tag{4}$$

where P and N are sets of positive and negative examples, respectively. Note that L is differentiable allowing to fit a mapping to an order embedding through gradient descent based techniques.

4.2. Hyperparameter Configuration

We search for a good hyperparameter configuration through random search. We search for an hyperparameter configuration according to the validation set and report the evaluation results on the test set partition. We consider feed

		P_{rand}	R_{rand}	F_{rand}	P_{lex}	R_{lex}	F_{lex}
(a)	OrdEmb	0.855	0.904	0.879	**0.823**	0.674	0.741
	OrdEmb +cohyp	0.857	**0.932**	**0.893**	0.809	**0.827**	**0.818**
	OrdEmb +pattern	**0.860**	0.885	0.872	0.798	0.766	0.782
	OrdEmb +pattern +cohyp	0.859	0.930	0.893	0.802	0.821	0.811

		P_{rand}	R_{rand}	F_{rand}	P_{lex}	R_{lex}	F_{lex}
(b)	OrdEmb	0.719	**0.946**	0.817	0.744	0.841	**0.789**
	OrdEmb +cohyp	0.847	0.869	0.858	**0.781**	0.716	0.747
	OrdEmb +pattern	0.742	0.931	0.826	0.666	**0.857**	0.749
	OrdEmb +pattern +cohyp	**0.848**	0.870	**0.859**	0.759	0.678	0.716

Table 4: Results on test set on Spanish. The upper table (a) shows the result of evaluating without introducing inferred cohyponymy instances in the test partition and the lower table (b) shows the results including cohyponymy instances in the test partition. The labels +cohyp and +pattern stand for cohyponymy and pattern-extracted instances in the training data.

	P_{rand}	R_{rand}	F_{rand}	P_{lex}	R_{lex}	F_{lex}
Best Distributional (Shwartz et al., 2016)	0.901	0.637	0.746	0.754	0.551	0.637
HypeNET Integrated (Shwartz et al., 2016)	0.913	**0.890**	0.901	0.809	0.617	0.700
OrdEmb ReLU	0.936	0.876	**0.905**	**0.958**	0.615	0.749
OrdEmb SELU-ReLU	0.932	0.845	0.887	0.740	**0.872**	**0.801**
OrdEmb tanh-sigm	**0.967**	0.836	0.897	0.788	0.756	0.771

Table 5: Order embedding results with different activation functions on test of Shwartz English dataset, and we include HypeNET and Best Distributional results reported by Shwartz.

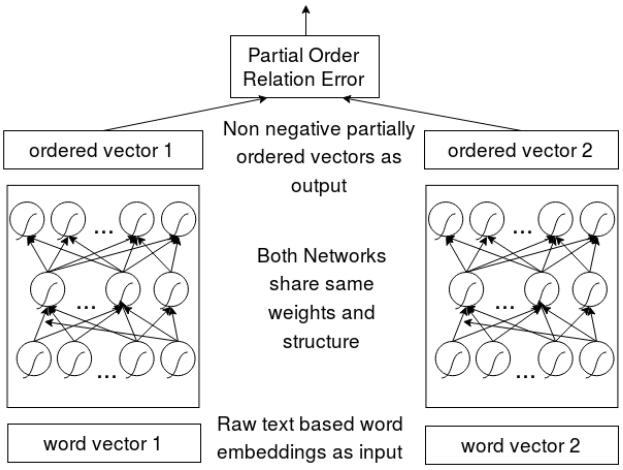

Figure 2: Order embedding diagram.

forward networks using pretrained fastText (Joulin et al., 2016) word vectors for Spanish and English.

We evaluate our models using precision, recall and F measures. The best configuration consisted on a three layered feed forward networks, with 150 neurons and SELU activation function on the first two layers and 100 ReLU units for the output layer. For the training we consider Adam (Kingma and Ba, 2014), with a learning rate of 0.005, and we conclude the training by early stopping, with a patience of 5. We checkout the best performing model against the validation set along the whole training.

4.3. Results for English

We include for comparison the results of the best distributional model reported by Shwartz et al. (Shwartz et al., 2016) and HypeNET integrated mdoel. In the Table 5 can be seen how the order embedding achieves considerable good results in comparison to the best distributional model reported by Shwartz and also in comparison to HypeNET, that is a pattern-based and distributional combined model.
We found interesting the good performance of the order embedding model taking as input general purpose word embeddings and without considering any explicit paths information on a corpus.

4.3.1. Results for Spanish

In this section we show the results obtained with the above described model in the introduced dataset for Spanish. We report order embedding results as a baseline in the dataset for future comparisons.
In order to show the behavior of pattern-extracted and cohyponymy instances we consider the following different variants of the training data:

- As base, the positive instances from WordNet and the translated instances of Shwartz dataset, and the negative instances randomly, sampling words from the vocabularies of Cardellino and WordNet. (OrdEmb)
- The base dataset adding cohyponyms as negative instances for training. (OrdEmb +cohyp)
- The base dataset adding positive instances extracted by patterns. (OrdEmb +pattern)

- The base dataset adding for training cohyponyms as negative instances and pattern extracted pairs as positive. (OrdEmb +pattern+cohyp)

We show the obtained results in the table 4. We evaluate the model against the base test partition and including cohyponymy instances on the test data. In the results can be observed that both cohyponyms and pattern-extracted instances during the training give some improvement in most cases, where cohyponyms are most beneficial, with the exception of the lexical split evaluating with cohyponyms addition in test partition.

5. Conclusion

In this paper we show the results obtained on supervised hypernymy detection in Spanish. Given the lack of resources in Spanish for hypernymy detection we build a dataset based on previous work for English. We included two versions of the dataset according to its train, validation and test partitions, and the lexical intersection between them: random and lexical split. The former is done randomly while the lexical split does not contain lexical intersection between the partitions, tackling the lexical memorization problem of the hypernymy detection. We train an order embedding using general purpose word vectors and we obtain that considerable good results. We show the behavior of including cohyponyms pairs for the training considerably improves the overall result.

6. Bibliographical References

Atserias, J., Villarejo, L., and Rigau, G. (2004). Spanish wordnet 1.6: Porting the spanish wordnet across princeton versions. In *LREC*.

Baroni, M., Bernardi, R., Do, N., and Shan, C. (2012). Entailment above the word level in distributional semantics. In *EACL*, pages 23–32. The Association for Computer Linguistics.

Camacho-Collados, J., Delli Bovi, C., Espinosa-Anke, L., Oramas, S., Pasini, T., Santus, E., Shwartz, V., Navigli, R., and Saggion, H. (2018). Semeval-2018 task 9: Hypernym discovery. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018); 2018 Jun 5-6; New Orleans, LA. Stroudsburg (PA): ACL; 2018. p. 712–24.* ACL (Association for Computational Linguistics).

Cardellino, C. (2016). Spanish billion words corpus and embeddings. *Spanish Billion Words Corpus and Embeddings*.

Chen, Q., Zhu, X., Ling, Z., Inkpen, D., and Wei, S. (2017). Natural language inference with external knowledge. *CoRR*, abs/1711.04289.

Clark, P., Fellbaum, C., and Hobbs, J. (2007). Using and extending wordnet to support question-answering. 01.

Dash, S., Chowdhury, M. F. M., Gliozzo, A., Mihindukulasooriya, N., and Fauceglia, N. R. (2019). Hypernym detection using strict partial order networks.

Espinosa-Anke, L., Camacho-Collados, J., Delli Bovi, C., and Saggion, H. (2016). Supervised distributional hypernym discovery via domain adaptation. In *Conference on Empirical Methods in Natural Language Processing; 2016 Nov 1-5; Austin, TX. Red Hook (NY): ACL; 2016. p. 424-35.* ACL (Association for Computational Linguistics).

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *ACL (1)*, pages 1199–1209. The Association for Computer Linguistics.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *14th International Conference on Computational Linguistics, COLING 1992, Nantes, France, August 23-28, 1992*, pages 539–545.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., and Mikolov, T. (2016). Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Levy, O., Remus, S., Biemann, C., and Dagan, I. (2015). Do supervised distributional methods really learn lexical inference relations? In Rada Mihalcea, et al., editors, *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 970–976. The Association for Computational Linguistics.

Marszalek, M. and Schmid, C. (2007). Semantic hierarchies for visual object recognition. *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference*, pages 1–7, June.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Nakashole, N., Weikum, G., and Suchanek, F. M. (2012). PATTY: A taxonomy of relational patterns with semantic types. In *EMNLP-CoNLL*, pages 1135–1145. ACL.

Navigli, R. and Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden, July. Association for Computational Linguistics.

Navigli, R., Velardi, P., and Faralli, S. (2011). A graph-based algorithm for inducing lexical taxonomies from scratch. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

Ortega, R. M. A.-a., Aguilar, C. A., VillaseÃ, L., Montes, M., and Sierra, G. (2011). Hacia la identificaciÃde relaciones de hiponimia/hiperonimia en Internet. *Revista signos*, 44:68 – 84, 03.

Roller, S., Erk, K., and Boleda, G. (2014). Inclusive yet selective: Supervised distributional hypernymy detection. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1025–1036, Dublin, Ireland, August.

Santus, E., Lenci, A., Lu, Q., and Schulte im Walde, S. (2014). Chasing hypernyms in vector spaces with entropy. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 38–42.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. *CoRR*, abs/1603.06076.

Snow, R., Jurafsky, D., and Ng, A. Y. (2004). Learning syntactic patterns for automatic hypernym discovery. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, pages 1297–1304.

Ustalov, D., Arefyev, N., Biemann, C., and Panchenko, A. (2017). Negative sampling improves hypernymy extraction based on projection learning. In *EACL (2)*, pages 543–550. Association for Computational Linguistics.

Vendrov, I., Kiros, R., Fidler, S., and Urtasun, R. (2015). Order-embeddings of images and language. *CoRR*, abs/1511.06361.

Vulic, I. and Mrksic, N. (2017). Specialising word vectors for lexical entailment. *CoRR*, abs/1710.06371.

Vylomova, E., Rimell, L., Cohn, T., and Baldwin, T. (2016). Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *ACL (1)*. The Association for Computer Linguistics.

Weeds, J., Clarke, D., Reffin, J., Weir, D. J., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *COLING*, pages 2249–2259. ACL.

Zhitomirsky-Geffet, M. and Dagan, I. (2005). The distributional inclusion hypotheses and lexical entailment. In *ACL*.