The Effectiveness of Simple Hybrid Systems for Hypernym Discovery

William Held and **Nizar Habash**
Computational Approaches to Modeling Language Lab
New York University Abu Dhabi, UAE
{wbh230,nizar.habash}@nyu.edu

Abstract

Hypernymy modeling has largely been separated according to two paradigms, pattern-based methods and distributional methods. However, recent works utilizing a mix of these strategies have yielded state-of-the-art results. This paper evaluates the contribution of both paradigms to hybrid success by evaluating the benefits of hybrid treatment of baseline models from each paradigm. Even with a simple methodology for each individual system, utilizing a hybrid approach establishes new state-of-the-art results on two domain-specific English hypernym discovery tasks and outperforms all non-hybrid approaches in a general English hypernym discovery task.

1 Introduction

Discovering word-level hierarchies has long been an important step in constructing language taxonomies. The most important of these hierarchical relationships is hypernymy or the ISA-relationship, i.e. 'chihuahua' *is a* 'dog', which forms the backbone of word-level taxonomies, most notably WordNet (Fellbaum, 1998).

Early works on the modeling of this relationship focused on the practical task of discovering new instances of the hypernymy relationship given a vocabulary and an existing resource with labeled data about hypernymy, described as hypernym discovery by Camacho-Collados (2017). For the purposes of discovery, Hearst (1992) developed a landmark set of lexico-syntactic patterns which indicated hypernymy.

There have been many follow-ups on this concept of identifying and utilizing patterns to identify hypernym pairs (Caraballo, 1999; Mann, 2002; Snow et al., 2005, 2006). However, by restricting the sentences of interest to only those which match patterns, even very large datasets with very loose pattern matching will often return small co-occurrence numbers, especially for more indirectly connected hypernym pairs.

To tackle the sparsity of pattern-based approaches, recent focus has turned to distributional models of hypernymy. Distributional models are attractive since they use signals drawn from every sentence of training data. Distributional approaches have focused on discovering spatial properties of embedding space which capture the hypernymy relationship (Kotlerman et al., 2010; Yamane et al., 2016; Shwartz et al., 2017; Nickel and Kiela, 2017; Vulic and Mrksic, 2017). The performance of distributional approaches in hypernymy detection shows promise to create a more broad picture of the hypernymy relationship space.

Recently, hybrid models of hypernymy, in both discovery and detection, have surpassed the performance of either paradigm individually. Similarly, the current state-of-the-art in hypernymy detection was set by a classifier which integrated information from both pattern data and distributional word embeddings (Shwartz et al., 2016). In hypernym discovery, where purely distributional methods have struggled, a model which utilized a hybrid approach of patterns and distributional representations far and away led the results of a recent SemEval Task (Camacho-Collados et al., 2018; Bernier-Colborne and Barriere, 2018).

In this paper, we study the benefits of hybrid strategies of hypernymy via a hybrid of extremely simple models of pattern-based and distributional hypernym discovery. We evaluate this model on the English sub-tasks of SemEval 2018 Task 9 for Hypernym Discovery. Overall, our results show that these paradigms have an almost directly complementary effect even when individual models are simple, a result which we support using the degrees of hypernymy each paradigm captures effectively.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3362–3367
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics

2 Pattern-Based Model

In order to make our pattern-based approach return a reasonable number of candidate hypernyms, we apply two separate methods to increase the number of candidate hypernyms presented by the pattern based model.

Extended Pattern Use First, we utilize a set of 47 extended Hearst Patterns as collected in Seitner et al. (2016). Additionally, we consider n-gram terms from our vocabulary to inherently contain a pattern co-occurrence with their sub-terms, e.g., *nuclear physics* $\rightarrow_{hyponym}$ *physics*. In English, this construction is common and accounts for a high number of "co-occurrences" between hyponyms and hypernyms.

All input sentences are tested by regular expression representations of these 47 patterns, yielding a table of candidates for the hypernymy relationship, in the form of x_{hypo}, y_{hyper}, and the number of times the pairs co-occurred in any of the extended Hearst Patterns. This stage is fully unsupervised but aims to extract lexico-syntactic information which indicates direct hypernymy. This raw co-occurrence table can be used to discover hypernym terms, with hypernym candidates scored based on their raw counts.

Hearst Matrix Singular Value Decomposition While this raw co-occurrence table can be used to discover candidate hypernyms, it still suffers from a high amount of sparsity even for terms which occur in patterns. Roller et al. (2018) showed exactly that performing singular value decomposition on co-occurrence tables can yield recall improvements, oftentimes outperforming state-of-the-art distributional methods for the hypernym detection task.

To modify this method for the hypernym discovery task, we simply sort all vocabulary terms that occur in Hearst Patterns according to the following metric:

$$sp(x, y) = U_x^T \Sigma_r V_y$$

where U, Σ, and V are taken from the singular value decomposition of the Hearst Pattern co-occurrence matrix and U_x, V_y are the row vector and the column vector for the hyponym and the hypernym respectively. Then, a similarity cutoff is tuned to maximize the F1 score of our predicted hypernyms on any labeled data that we have.

For words which never occur in any patterns, we still lack the ability to generate any reasonable candidates which causes this approach to still suffer from low total recall due to query terms never seen in patterns.

3 Distributional Modeling with Hypernyms from Nearest Neighbor

For our distributional methodology, we choose the simplest possible supervised approach to hypernym discovery - a single nearest neighbor approach - in which the hypernyms for each query term are transferred from their nearest neighbor in the training data. This approach is motivated by the work of Snow et al. (2006) where linking a new hyponym to a similar known hyponym was shown to effectively encode an enormous amount of signal about correct hypernyms.

Our method is as follows. Suppose we have a training set H consisting of a number of hyponyms and their corresponding hypernyms.

$$H : \{Hypo_i : Hyper_i^1 ... Hyper_i^j\}$$

For a given query term Q, we find the nearest neighbor $Hypo_{nn}$ from the training set by the cosine similarity of vector representations of the words. The hypernyms of the nearest neighbor are then sorted by descending frequency in the training set, such that the words which served as hypernyms to more known terms in the training data come first.

This sorting metric serves as a heuristic of the generality of the hypernyms of the nearest neighbor. Since the nearest neighbor is unlikely to be an exact synonym, it is more likely that the query and its nearest neighbor share more general hypernyms, those that would appear at a lower depth in a taxonomy, than they are to share extremely specific hypernyms.

Additionally, a similarity cutoff point is trained on tuning data, such that if there is no nearest neighbor with greater similarity the cutoff point, the nearest neighbors strategy simply returns the most frequent hypernyms from the entire training set. Contrasting to the Hearst Patterns, our distributional method instead tries to provide as many reasonable guesses to hypernyms as possible unless the nearest neighbor is very far away.

Embedding Methodology Details In theory, any word embedding model can be used for this

	General			Medical			Music		
Model Variant	MAP	MRR	P@5	MAP	MRR	P@5	MAP	MRR	P@5
Count Hearst Patterns	4.60	11.70	4.10	14.99	43.18	13.70	7.65	26.14	6.76
SVD Hearst Patterns	6.19	15.12	5.65	15.80	47.01	14.53	8.89	29.63	8.05
Hypernyms of Nearest Neighbor	9.85	24.56	8.76	29.57	48.18	34.10	38.65	72.77	38.45
Hybrid of Raw Count & NN	14.82	32.61	13.80	35.29	63.59	38.73	28.22	61.26	30.67
Hybrid of SVD & NN	**15.97**	**34.07**	**15.00**	**37.85**	**64.47**	**40.19**	**54.62**	**77.24**	**55.08**

Table 1: Comparison of model variants on all three sub-tasks of SemEval 2018 Task 9.

nearest neighbor task as it does not explicitly take advantage of any particular features of a particular word embedding. However, in practice, we found that the FastText (Bojanowski et al., 2016) algorithm is preferable since even out of vocabulary query words are able to be given reasonable embeddings due to the meaningful embeddings of sub-strings that FastText provides. This guarantees that the nearest neighbor approach always gives some form of candidate hypernyms, even for words which are out of vocabulary or word n-grams which don't have specific embeddings. For the purposes of evaluation, we used 100-dimensional embeddings with common n-grams joined together.

4 Hybrid Approach

Ultimately, we combine the methods in order to capture the valuable elements of each. While pattern-based approaches suffer from sparseness, they do tend to generate high precision results when available. Conversely, the nearest neighbor approach almost always generates a fair number of candidate hypernyms but suffers from low precision unless the nearest neighbor is an exact synonym. Therefore, we propose the following ordering rule for candidate hypernyms. When the pattern-based approach yields results, we rank them as first. Then, the hypernyms of the nearest neighbor are added until our total desired number of candidates is reached. Since this is a supervised setting, we tune a cutoff similarity value for the pattern-based approaches as described in Section 2.

5 Experiments & Results

We evaluate our model on SemEval 2018 Task 9, the only existing benchmark for the hypernym discovery task. Specifically, we focus on the 3 English sub-tasks: general English, Medical literature, and Music literature. Each task comes with a separate corpus of unlabeled text data, training and trial data of hyponyms labeled with their complete list of hypernyms, and a vocabulary of valid hypernyms. The final results of a model are tested on a dataset of equal size to the training data. Further details can be found in Camacho-Collados et al. (2018)'s paper describing the tasks and their respective data.

For each sub-task, we only use the data from the specific sub-task we are evaluating. The provided trial dataset is used to tune our cutoff points for Hearst Pattern frequency and minimum similarity for the nearest neighbor hypernyms approach. For each query word, we propose 15 candidates ranked as described in Section 4.

Our initial experiments, shown in Table 1, compare all variations of our described systems on all tasks. The hybrid models consistently outperform the individual independent models by a significant margin, except for the Music task where the raw count method seems to negatively impact the hypernyms of nearest neighbor approach. The fact that our simple combination of these two models yields improved results is a positive indication that they each contribute separate signals to the model.

For the three English sub-tasks, we evaluate our model using the evaluation script from the SemEval task, and compare our results on Mean Average Precision, Mean Reciprocal Rank, and Precision at 5, the metrics primarily discussed in the original task. We compare our system to the CRIM (Bernier-Colborne and Barriere, 2018), 300-sparsans (Berend et al., 2018), vanilla Taxoembed (Espinosa-Anke et al., 2016), and most frequent hypernym systems. The first two were the only systems to achieve state-of-the-art results on the above metrics for the three English sub-tasks, while the latter two represent the best baselines from the shared task. The comparison against these models on all three sub-tasks can be found in Table 2.

	General			Medical			Music		
Model	MAP	MRR	P@5	MAP	MRR	P@5	MAP	MRR	P@5
Hybrid of SVD & NN(Our Model)	15.97	34.07	15.00	**37.85**	**64.47**	**40.19**	**54.62**	**77.24**	**55.08**
CRIM (Bernier-Colborne and Barriere, 2018)	**19.78**	**36.10**	**19.03**	34.05	54.64	36.77	40.97	60.93	41.31
vTE* (Espinosa-Anke et al., 2016)	10.60	23.83	9.91	18.84	41.07	20.71	12.99	39.36	12.41
300-sparsans (Berend et al., 2018)	8.95	19.44	8.63	20.75	40.60	21.43	29.54	46.43	28.86
Most Frequent Hypernym*	8.77	21.39	7.81	28.93	35.80	34.20	33.32	51.48	35.76

Table 2: Results for all three English sub-tasks in SemEval 2018 Task 9. Baselines are marked with *, state-of-the-art is marked in bold.

Our simple hybrid model outperforms all systems in the competition on the general English hypernym discovery task except for CRIM. In the general task, we find it worth noting that there is a significant performance gap between our hybrid approach and all non-hybrid models, despite the simplicity of our model. The state-of-the-art model, CRIM, is also a hybrid model, but it makes much more robust use of the larger training set provided in the general English sub-task.

Perhaps more surprising, our model yields new state-of-the-art results for the Music and Medical sub-domain tasks. As these approaches are both much smaller tasks with around 1/3rd of the training data, we see that our model is able to make effective use out of smaller datasets as well.

6 Analysis

In our goal of evaluating hybrid models in isolation, we quantitatively analyze why these paradigms are beneficial in concert and manually analyze where these models fail and perform well.

Hypernymy Distance Analysis In order to explain the high degree of compatibility the hybrid model highlights, we explore the idea that each model is modeling not only signal in support of the same hypernyms but tends to model wholly different types of hypernymy.

In Section 3, we discussed the intuitions behind our sorting method that optimizes the nearest neighbor to rank general hypernyms first, as these are more likely to also apply to the query term. By contrast, the Hearst Patterns are more likely to occur when the query and hypernym are directly related. Our intuitions about the type of information captured by each model state that nearest neighbors should effectively yield higher portions of the taxonomy, while Hearst Patterns will link direct hypernyms.

In Table 3, we support this by calculating the average length of the shortest path between the hyponym and the proposed hypernym for each model. The metric is not dramatic but it clearly separates the two approaches. Correctly predicted hypernyms from the nearest neighbor approach lie on average around one step further away on Wordnet from their query words than our correctly predicted hypernyms from the Hearst Patterns.[1]

Manual Error Analysis In order to more fully understand the contributions of each model to our results, we perform manual error analysis on a randomly selected subset of the test data. 100 examples were selected from the General sub-task and 50 examples each were taken from the Music and Medical sub-tasks. Results are in Table 4. Within these examples, each candidate is labeled with which system yielded the answer. We also categorize certain types of error into their own class.

Overall, Hearst Pattern candidates account for 20% of all candidates and have a precision of 18%. Nearest Neighbor candidates are 80% of all candidates and also have a precision of 18%. The full Hearst Pattern precision numbers are 10%, 38% and 68% for the General, Music and Medical subtasks, respectively. The Nearest Neighbor precision numbers are 9%, 20%and 29%, respectively.

In all datasets, Hearst Patterns alone almost never capture all hypernyms, but especially in special topic fields they show high precision results, as projected by previous work. In the general subtask, Hearst Patterns struggle more, generally when the query term is a term that is used in versions of the patterns that do not translate well to actual hypernymy, e.g., *consumption*$\rightarrow_{bad\_pattern\_hyponym}$ *factor* from the phrase

[1]This distance is calculated when both terms exist in WordNet. For terms which lie in the other knowledge graphs used to construct the SemEval Task 9 dataset, we don't calculate a distance.

Model	All Predictions	Correct Predictions
Raw Hearst Patterns	6.33	3.64
SVD Hearst Patterns	6.33	3.63
Nearest Neighbor	7.54	4.81

Table 3: Average length of shortest path between predicted hypernyms and their input hyponyms.

	Correct		Incorrect		Near Miss		Gold Error	
Dataset	HP	NN	HP	NN	HP	NN	HP	NN
General (100)	24	58	384	893	33	39	25	39
Music (50)	30	188	50	468	7	0	5	2
Medical (50)	21	141	8	558	2	0	3	0

Table 4: Error analysis of all all candidate hypernyms in a random sub-sample (number of queries in parentheses).

"Consumption is a factor in...".

Altogether, our best answers largely come from either very similar nearest neighbors, or hybrid instances where the Hearst Patterns capture a few specific hypernyms and rough hypernyms are captured by the nearest neighbor. We view the latter instances as ideal, since they depend neither on a perfect nearest neighbor nor on patterns capturing indirect hypernyms. For example, the query *Fudan University* has three gold hypernyms {*university*, *school*, *educational institution*}, *university* and *school* are returned by the Hearst Patterns and *educational institution* is returned by the nearest neighbor.

In the general sub-task, the selection of a bad nearest neighbor when no Hearst Patterns are found is the source of a large number of major failures. Qualitative analysis generally shows that this occurs when the embedding for a rarely used query word must rely on its sub-string embedding from fastText, leading to a very incorrect nearest neighbor that still has high confidence, e.g., *Queen Elizabeth*$\rightarrow_{bad\_nearest\_neighbor}$*Elizabeth Einstein*. In the more specific sub-tasks, this type of error is less common as the domain is constrained in scope, making wildly incorrect nearest neighbors less common. However, in these more specific tasks, outliers with no strong nearest neighbor are much more frequent as the number of low confidence nearest neighbors increases in these tasks. In these cases, the model defaults to giving the most frequent hypernyms from training since the confidence cutoffs of neither Hearst Patterns nor the nearest neighbor similarity are met.

We also separate out two interesting categories of error: gold errors (occurring 2.5% of the time) and near misses (occurring 2.9% of the time). These categories have similar properties and generally form within specific queries. The prior occurs primarily when a different sense is captured than the sense in the gold data itself, e.g., *cereal*$\rightarrow_{gold\_false\_negative}${*crop*, *grain*, *snack*, *foodstuff*, *carbohydrate*}. The latter occurs primarily when an incorrect, but close, family of hypernyms is obtained from the data, e.g., *microscope*$\rightarrow_{near\_miss\_candidates}${*technology*, *facility*, *observer*, *measuring device*}.

7 Conclusions and Future Work

We studied the impact of utilizing a hybrid of pattern-based and distributional models for hypernym discovery by hybridizing simple models from each paradigm. Our results show that hybrid models of even simple systems are able to perform surprisingly well, consistently outperforming more robust single strategy models. Interestingly, a manual error analysis and metrics taken from WordNet suggests each paradigm models different types of hypernymy. We conclude that further work in hypernym discovery should utilize signals taken from both historical paradigms of hypernymy modeling, not only to improve confidence in answers but also to capture both direct and indirect hypernym relationships.

References

Gábor Berend, Márton Makrai, and Peter Földiák. 2018. 300-sparsans at semeval-2018 task 9: Hypernymy as interaction of sparse attributes. pages 928–934.

Gabriel Bernier-Colborne and Caroline Barriere. 2018. Crim at semeval-2018 task 9: A hybrid approach

to hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 725–731.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Jose Camacho-Collados. 2017. Why we have switched from building full-fledged taxonomies to simply detecting hypernymy relations. *arXiv preprint arXiv:1703.04178*.

Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. Semeval-2018 task 9: Hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 712–724.

Sharon A Caraballo. 1999. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*.

Luis Espinosa-Anke, Jose Camacho-Collados, Claudio Delli Bovi, and Horacio Saggion. 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).

Christiane Fellbaum. 1998. Wordnet: An electronic lexical database and some of its applications.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

Gideon S Mann. 2002. Fine-grained proper noun ontologies for question answering. In *Proceedings of the 2002 workshop on Building and using semantic networks-Volume 11*, pages 1–7. Association for Computational Linguistics.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems*, pages 6341–6350.

Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. *CoRR*, abs/1806.03191.

Julian Seitner, Christian Bizer, Kai Eckert, Stefano Faralli, Robert Meusel, Heiko Paulheim, and Simone Paolo Ponzetto. 2016. A large database of hypernymy relations extracted from the web.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. *arXiv preprint arXiv:1603.06076*.

Vered Shwartz, Enrico Santus, and Dominik Schlechtweg. 2017. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. pages 65–75.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Advances in neural information processing systems*, pages 1297–1304.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 801–808. Association for Computational Linguistics.

Ivan Vulic and Nikola Mrksic. 2017. Specialising word vectors for lexical entailment. *CoRR*, abs/1710.06371.

Josuke Yamane, Tomoya Takatani, Hitoshi Yamada, Makoto Miwa, and Yutaka Sasaki. 2016. Distributional hypernym generation by jointly learning clusters and projections. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1871–1879.