
CS 224N Class Project

Automatic Hypernym Classification

Rion L. Snow and Kayur D. Patel

Department of Computer Science

Stanford University

Stanford, CA 94305

{rion,kdpatel}@cs.stanford.edu

Abstract

Hypernym classification is the task of deciding whether, given two words, one word “is a kind of” the other. We present a classifier that learns the noun hypernym relation based on automatically-discovered lexico-syntactic patterns between a set of provided hyponym/hypernym noun pairs. This classifier is shown to outperform two previous methods for automatically identifying hypernym pairs (using WordNet as a gold standard), and is shown to outperform those methods as well as WordNet on a hand-labeled data set.

1 Introduction

The classification of general relationships between concepts has long been a subject of intense study in linguistics. WordNet [9] is one of the largest such projects for cataloguing relationships between concepts in English and several other languages. Chief among the relationships catalogued between pairs of nouns are the synonym (same meaning as), antonym (opposite meaning as), hypernym (is a kind of), holonym (is a part of), and coordinate term (is the same kind of thing as) relations. Machine learning techniques have been applied to capture a subset of these relationships automatically; in particular, the problem of discovering sets of synonyms and coordinate terms has been studied at length (see for example, [5]). Work in automatically inducing other relationships has been less successful; in particular, in the study of the hypernym relationship some hand-designed patterns have been found to be effective in discovering novel hyponym/hypernym pairs[4]; further, preliminary hypernym ontologies have been constructed using a small number of hand-constructed patterns [1], [2]. However, no reliable method for automatically discovering such patterns has yet been implemented. Some algorithms have been sketched, however; most notably the algorithm originally proposed for discovering new patterns in [4]:

“...In order to find new patterns automatically, we sketch the following procedure:

1. Decide on a lexical relation, R , that is of interest, e.g., group/member” (in our formulation this is a subset of the hyponymy relation).
2. Gather a list of terms for which this relation is known to hold, e.g., England-country”. This list can be found automatically using the method described here, bootstrapping from patterns found by hand, or by bootstrapping from an existing lexicon or knowledge base.

3. Find places in the corpus where these expressions occur syntactically near one another and record the environment.
4. Find the commonalities among these environments and hypothesize that common ones yield patterns that indicate the relation of interest.
5. Once a new pattern has been positively identified, use it to gather more instances of the target relation and go to Step 2."

We apply this algorithm for discovering the lexico-syntactic patterns relating noun hypernym pairs. As mentioned in [4], the problem of "finding the commonalities" in word "environments" is underdetermined, i.e. there is no canonical way to reliably capture the 'commonalities' among 'word environments' in natural text. Nonetheless recent advances in automatic parsing technology allows us to represent natural language sentences in a structurally reliable way; we propose one possible method of reliably 'finding commonalities' in the following section.

2 Automatically Discovering Hypernym Relationships

Recording the lexico-syntactic environment between a specific pair of words has thus far been limited to collecting the counts of hand-designed features, for example, [12], [3]. Unfortunately this method of lexicon construction is tedious and subject to the bias of the designer; further these lexicons are necessarily a very small subset of the actual 'patterns' found to occur in natural text. Some recent attempts have applied a novel method for automatically discovering relationships between pairs of nouns in the context of automatic inference rule discovery [5], using the dependency relations produced by MINIPAR, a broad-coverage principle-based parser for English described at length in [7]. We propose a similar, entirely automatic method of capturing the word environment between related words in a repeatable, consistent fashion. In particular, we use Lin's MINIPAR parser to produce directed dependency trees of sentences in text, and then for each sentence record as our 'environment' the shortest path in the dependency tree between the nouns of interest (with optional 'satellite' nodes). For example, given the sentence fragment (from the TIPSTER 1 corpus) "Oxygen is the most abundant element on the moon," MINIPAR yields the dependency tree partially depicted in Figure 1:

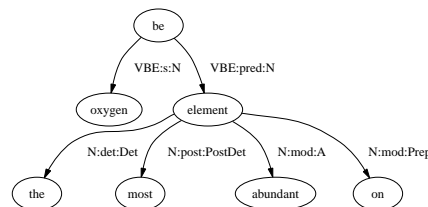


Figure 1: Dependency Tree from MINIPAR

We then remove the noun pair information to produce a general path, for example the first feature above becomes simply: "-N:VBE:V,be,be,V:VBE:N". The full list of extracted features is given in Figure 1.

We apply MINIPAR in this way to a corpus of over 6 million newswire sentences (consisting of articles from the Associated Press, Wall Street Journal, and Los Angeles Times, drawn from the Tipster 1, 2, 3, and Trec 5 corpora). From these we construct a feature lexicon containing all dependency paths discovered between all pairs of nouns, such that the path occurs between at least five unique noun pairs in our corpus. This feature lexicon consists of about 70,000 dependency paths. We then construct feature count vectors for every pair of nouns occurring within any sentence in our corpus; each row in the vector is

Table 1: Paths extracted from Figure 1

OXYGEN,-N:VBE:V,BE,BE,V:VBE:N,ELEMENT]
 OXYGEN,-N:VBE:V,BE,BE,V:VBE:N,ELEMENT,(THE,DET:DET:N)]
 OXYGEN,-N:VBE:V,BE,BE,V:VBE:N,ELEMENT,(MOST,POSTDET:POST:N)]
 OXYGEN,-N:VBE:V,BE,BE,V:VBE:N,ELEMENT,(ABUNDANT,A:MOD:N)]
 OXYGEN,-N:VBE:V,BE,BE,V:VBE:N,ELEMENT,(ON,PREP:MOD:N)]

simply the count of the number of occurrences of a particular feature in conjunction with the noun pair. Using this formalism we have been able to capture a wide variety of repeatable patterns between hyponym/hypernym noun pairs; in particular, we have been able to 'rediscover' many of the hand-designed patterns proposed in [4], in addition to a number of new patterns not discussed by Hearst.

Table 2: Exact Corpora List:

Tipster 1: AP 1989, WSJ 1987-1989
 Tipster 2: AP 1988, WSJ 1990-1992
 Tipster 3: AP 1990
 Trec 5: LA Times 1989-1990

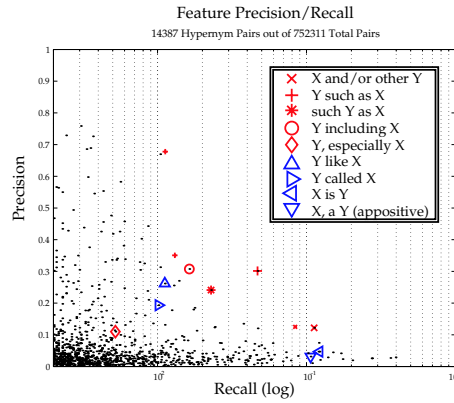


Figure 2: Hypernym Pre/Re for all Features

Figure 2 depicts the precision and recall of each feature on the hypernym pair recall task. As far as we know, this is the first comparison of the precision and recall of Hearst's patterns for hyponym discovery vs. a large subset of possible relationships. Note that while Hearst's patterns (in red) are some of the best individual patterns, they are far from comprehensive. The specific MINIPAR representation of Hearst's patterns are as follows:

Our analysis reveals several important findings: first, that there is no 'silver bullet' pattern that can reliably identify a large portion of hypernym pairs. This analysis justifies Hearst's initial intuition as to the power of hand-selected patterns; nonetheless, this analysis shows that Hearst's intuitive listing of lexico-syntactic patterns covers only a fraction of useful patterns, and we should expect that a trained classifier may use the additional features to its advantage. In addition we have discovered a number of high precision paths relating hypernyms, in blue in Figure 2, for example:

Table 3: Dependency Path Representations of Hearst’s Patterns

NP such as NP:	-N:PCOMP-N:PREP,SUCH_AS,SUCH_AS,-PREP:MOD:N
Such NP as NP:	-N:PCOMP-N:PREP,AS,AS,-PREP:MOD:N,(SUCH,PRED:PRE:N)
NP including NP:	-N:OBJ:V,INCLUDE,INCLUDE,-V:I:C,DUMMY_NODE,DUMMY_NODE,-C:REL:N
NP and other NP:	(AND,U:PUNC:N),N:CONJ:N,(OTHER,A:MOD:N)
NP or other NP:	(OR,U:PUNC:N),N:CONJ:N,(OTHER,A:MOD:N)
NP, especially NP:	N:APPO:N,(ESPECIALLY,A:APPO-MOD:N)

Table 4: Dependency Path Representations of other Hypernym-Specific Patterns

NP(,) called NP:	-N:DESC:V,CALL,CALL,-V:VREL:N
NP like NP:	-N:PCOMP-N:PREP,LIKE,LIKE,-PREP:MOD:N
NP is a NP:	-N:S:VBE,BE,BE,VBE:PRED:N
NP, a NP: (appositive)	-N:APPO:N

3 The Hypernym Classifier

3.1 Constructing the Hypernym-only Classifier

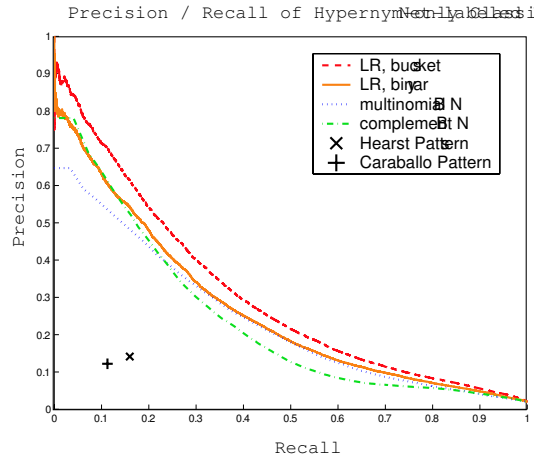


Figure 3: Performance of Hypernym-only Classifiers on WordNet-Labeled Dev Set

We would like to use this feature space to train a classifier to distinguish between hypernym and non-hypernym noun pairs. We perform this by creating feature vectors for every noun pair in our training set, where each feature is simply the count of times a particular dependency path is used to connect that noun pair. For our training and test sets we used a large newswire corpus consisting of a total of approximately 50 million sentences (over 500 million words). We labeled each noun pair as one of the following four disjoint classes, as determined by WordNet:

greater than hypernymy: Word a and word b are related by a greater than hypernymy relation if a is a kind of b .

less than hypernymy: Word a and word b are related by a less than hypernymy relation if word b is a kind of a .

coordinate: Word a and word b are coordinate if a and b share the same parent hypernym. For example apples and oranges are coordinate terms because they are both kinds of fruit.

unrelated: If word a and word b do not fit any of the relations above, they are categorized as unrelated.

Definition of the Training Set:

If a particular noun pair was listed in WordNet as belonging to more than one of these classes (a frequent occurrence due to polysemy), we removed this pair from our training set. We expect future versions of our algorithm will be able to deal with multiple classes for different senses of the same word; in this experiment, however, we restrict our training set to the single sense case. After creating our feature space, we implement a multinomial Naive Bayes classifier and a logistic regression classifier. 10-fold cross validation: 752311 vectors, 65952 features. Model selected based on Max F-Score Data:

Table 5: Max Average F-Score: Multinomial and Complement Naive Bayes

<i>Smoothing Parameter</i>	<i>Multinomial Naive Bayes</i>	<i>Complement Naive Bayes</i>
0.1000	0.2995	0.3024
0.2000	0.2961	0.2992
0.5000	0.3032	0.3002
1.0000	0.3088	0.3023
2.0000	0.3160	0.3008
4.0000	0.3175	0.2948
8.0000	0.2735	0.2826

Table 6: Max Average F-score: Summary

Logistic Regression (Buckets):	0.3480
Logistic Regression (Binary):	0.3200
Best Multinomial Naive Bayes:	0.3175
Best Complement Naive Bayes:	0.3024
Hearst Patterns:	0.1500
Caraballo Pattern:	0.1170

3.2 Constructing the Coordinate Classifier

We would like to apply a word similarity model to enhance our model of hypernym classification; using the insight that distributional information is much easier to obtain, we wish to supplement our sparse hypernym information with dense distributional information. We define the coordinate relation as a symmetric relation between words that are "the same kind of thing". Prior work for classifying the coordinate relation include automatic word sense clustering methods based on *distributional* similarity (e.g. [10]) or on pattern-based techniques, specifically using the *coordination* pattern (e.g. [2]). We construct a vector-space model similar to [10] using a single connecting dependency link from MINIPAR as our distributional features; we create a feature count vector for every individual word consisting of all dependency links, normalize these feature counts with pointwise mutual information, and compute the cosine coefficient between the normalized vectors as a measure of similarity. We evaluate our coordinate classifier on our hand-labeled test set of 5,387 pairs (of which 131 are labeled as "coordinate"). For purposes of comparison we constructed a classifier from WordNet, which has a simple binary decision of determining that two words are coordinate if they share a common ancestor precisely n words higher up

in the hypernym ontology, for n up to 6. Also we compare a simple pattern-based classifier based on the *conjunct* pattern (e.g. “X and Y”), which thresholds simply on the number of conjunct patterns found between the pair.

Table 7: Summary of F-scores on Hand-labeled Coordinate Pairs

Interannotator Average:	0.6405
Best Vector Space F-score on Coordinate-only Classification:	0.3327
Conjunct Classifier on Coordinate-only Classification:	0.2857
Best WordNet F-score on Coordinate-only Classification:	0.2630

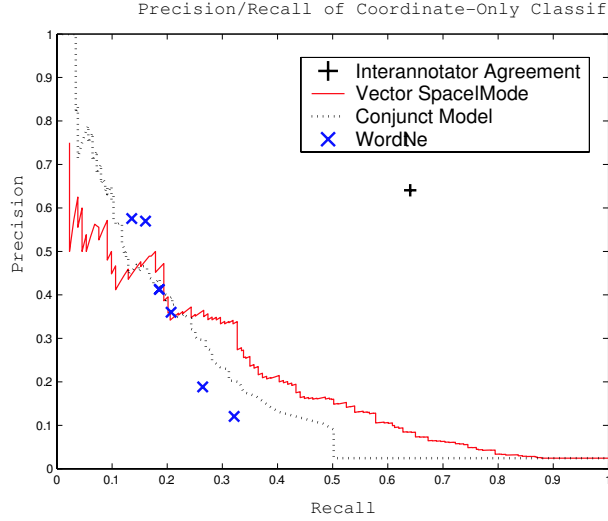


Figure 4: Vector space model vs. Conjunct-pattern model vs. WordNet Precision/Recall on hand-labeled coordinate term testset

3.3 The Combined Classifier

Finally we would like to combine our hypernym and coordinate models in order to improve our hypernym classification. Defining the probability produced by our best hypernym-only classifier as

$$P_{old}(e_i <_H e_k)$$

, and a normalized probability obtained from our coordinate classifier as

$$P(e_i \sim_C e_j)$$

, we apply a simple linear interpolation scheme to compute a new probability: that e_k is a hypernym of e_i :

$$P_{new}(e_i <_H e_k) = \lambda_1 P_{old}(e_i <_H e_k) + \lambda_2 \sum_j P(e_i \sim_C e_j) P(e_j <_H e_k)$$

We restrict our parameters λ_1, λ_2 such that $\lambda_1 + \lambda_2 = 1$, and set these parameters using 10-fold cross-validation on our hand-labeled test set. For our final evaluation we use $\lambda_1 = 0.3$.

4 Evaluation and Comparison to Past Work

Having trained our hypernym classifier on the full dependency path feature space we have created, and having constructed our coordinate classifier over more than 10,000 of the most frequent words encountered in our training corpus, we would now like to compare our results to other methods for classifying hypernyms. To do this we have constructed a hand-labeled test set of 5,387 noun pairs from randomly-selected paragraphs within our corpus. Each noun pair is labeled as “hyponym-to-hypernym”, “hypernym-to-hyponym”, “coordinate”, and “unrelated”. The breakdown is as follows:

Table 8: Makeup of 5,387 pairs in Hand-labeled Test Set

Unrelated Pairs:	5124
Coordinate Pairs:	131
Hyponym/Hypernym Pairs:	133

Interannotator agreement was obtained on a held-out agreement set of 500 pairs. Agreement is averaged precision/recall across all pairs of four labelers on class of focus.

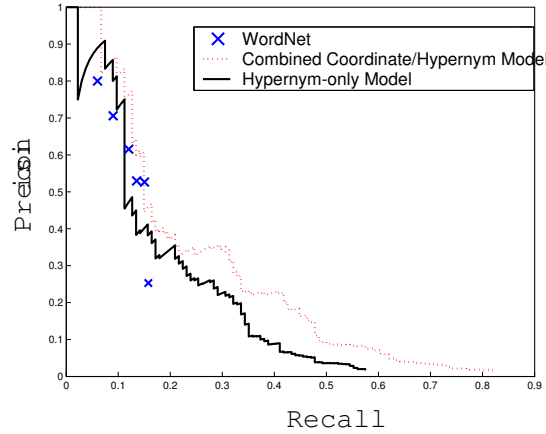


Figure 5: Pre/Re on Hand-Labeled Test Set

Human-Human agreement, Hypernym: 83%
Human-Human agreement, Coordinate: 64%

Ideally, we would like also to compare our learned classifier against WordNet itself; we expect that, despite having only trained on WordNet, we may be able generalize and perform better than WordNet in a large number of cases; this is due to the fact that our classifier is context-dependent, and thus, given sufficient data, it may infer the appropriate classification even for very specific, domain-limited terms (e.g. proper names) that have not been (or ever will be) catalogued in WordNet. Unfortunately, it is clear that for this comparison it is uninformative to use WordNet as ground truth, and thus we must create a hand-labeled data set for this test set. We have performed such a blind-labeling for 5,387 noun pairs, selected at random from a thresholded set of all within-sentence noun-pairs collected from our corpora. Figure 5 contains a plot of precision / recall vs. WordNet, as well as the methods in the previous comparison, now using the human labels as ground truth.

Interannotator agreement obtained on a held-out agreement set of 500 pairs. Agreement is averaged precision/recall across all pairs of four labelers on class of focus.

Table 9: Final Evaluation of Combined Model

Combined Linear Interpolation Hypernym/Coordinate Model:	0.3268
Hypernym-only Logistic Regression (buckets):	0.2714
Best WordNet F-Score:	0.2339

Best WordNet results were found using the First Sense, All Syns, 4-Ancestor model. Hypernym-only model is a 16% F-score improvement over WordNet, while the combined Hypernym/Coordinate model has a 40% F-score improvement over WordNet.

We observe that our classifier consistently does as well or better than all other methods presented, and in particular it has a large f-score improvement over any classifier built from WordNet. While our test corpus (newswire) is still too specific to broadly generalize from these results, this data does indicate the possibility that our classifier may perform better than WordNet for a much larger hand-labeled data set over more general corpora such as encyclopediae. Automatic hypernym classification is but one step on the path towards our larger goal of constructing full semantic hierarchies. Our future work will involve incorporating what we have learned from building our classifier into a larger ontology-induction methodology, with the ambition of creating a full hypernym ontology directly from natural language.

Acknowledgments

We thank Daniel Jurafsky and Andrew Ng for helpful discussion and guidance. We thank George Miller et al. for WordNet 2.0, Jason Rennie for the WordNet::QueryData Perl Module, Dekang Lin for the MINIPAR dependency parser, and the fine folks at the Auton Lab for their sparse logistic regression package. Rion Snow is supported by the NDSEG Fellowship sponsored by the DOD and AFOSR.

References

- [1] Caraballo, S.A. (2001) Automatic Acquisition of a Hypernym-Labeled Noun Hierarchy from Text. Brown University Ph.D. Thesis.
- [2] Cederberg, S. & Widdows, D. (2003) Using LSA and Noun Coordination Information to Improve the Precision and Recall of Automatic Hyponymy Extraction. *Seventh Conference on Computational Natural Language Learning (CoNLL-2003)*, Edmonton, Canada, pp. 111-118.
- [3] Girju, R., Badulescu A., & Moldovan D. (2003) Learning Semantic Constraints for the Automatic Discovery of Part-Whole Relations. In *Proceedings of the Human Language Technology Conference*, Edmonton, Canada.
- [4] Hearst, M. (1992) Automatic Acquisition of Hyponyms from Large Text Corpora. *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes, France.
- [5] Lin, D. (1998) Automatic Retrieval and Clustering of Similar Words. *COLING-ACL98*, Montreal, Canada.
- [6] Lin, D. (1998) Dependency-based Evaluation of MINIPAR. *Workshop on the Evaluation of Parsing Systems*, Granada, Spain
- [7] Lin, D. & Pantel P. (2001) Discovery of Inference Rules for Question Answering. *Natural Language Engineering*, 7(4):343-360.
- [8] Lin, D., Zhao, S., Qin, L., & Zhou M. (2003) Identifying Synonyms among Distributionally Similar Words. In *Proceedings of IJCAI-03*, pp.1492-1493.
- [9] Miller, G. (1995) WordNet: a lexical database for English. *Communications of the ACM*

- [10] Pantel, P. (2003) Clustering by Committee. Ph.D. Dissertation. Department of Computing Science, University of Alberta.
- [11] Rennie J., Shih, L., Teevan, J., & Karger, D. (2003) Tackling the Poor Assumptions of Naive Bayes Text Classifiers. *Proceeding of the Twentieth International Conference on Machine Learning (ICML-2003)*, Washington DC.
- [12] Turney, P.D., Littman, M.L., Bigham, J. & Shanyder, V. (2003) Combining independent modules to solve multiple-choice synonym and analogy problems. *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP-03)* , Borovets, Bulgaria, pp. 482-489.