Finding Cars, Goddesses and Enzymes: Parametrizable Acquisition of Labeled Instances for Open-Domain Information Extraction

Benjamin Van Durme*
University of Rochester
Rochester, New York 14627
vandurme@cs.rochester.edu

Marius Paşca
Google Inc.
Mountain View, California 94043
mars@google.com

Abstract

A method is given for the extraction of large numbers of semantic classes along with their corresponding instances. Based on the recombination of elements clustered through distributional similarity, experimental results show the procedure allows for a parametric trade-off between high precision and expanded recall.

Introduction

Motivation

The goal of automatically constructing large knowledge bases from text (Schubert 2006) has reached new momentum, thanks to a combination of large text corpora being publicly available (Web), new sources of textual data being explored (query logs), and new efforts for large-scale information extraction being pursued. A common, unifying theme in recent work is that knowledge of various types (classes of named entities, class attributes, class-to-class relations) can be acquired from unstructured text, if the type of knowledge to be extracted is manually specified in advance. Often, the input to Web-scale extraction methods consists of small sets of hand-picked instances that are representative of a class of interest for which knowledge (e.g., other instances within the same class; or attributes of the class; or relations which involve the class) needs to be extracted. Progress is being hampered by the lack of a reliable resource containing a diverse set of classes represented through sets of representative instances. Because such a resource did not exist, the authors were forced to either focus on previously-studied, coarse-grained classes such as Location, Person, Organization, etc., or create new, experimental classes manually, limiting both the scale of experiments on such classes and the likelihood that the newly-created classes are as diverse as the classes of interest to a wide population, such as Web search users.

A number of proposals have been given for the automatic accumulation of classes (Hearst 1992; Caraballo 1999; Snow, Jurafsky, & Ng 2005), the common theme being the dependence on so-called *is-a* patterns as introduced by Hearst (whether manually specified or otherwise). To increase precision, recent efforts have made use of externally supplied semantic constraints, such as WordNet (Snow, Jurafsky, & Ng 2006), or term clusters derived via distributional similarity (Pantel & Ravichandran 2004).

The work presented in this paper was motivated by the intuition that the method given by Pantel & Ravichandran is overly trusting in the quality and granularity of automatically derived semantic clusters. Underlying the approach described in the following section is the assumption that such clusters tend to be reasonable, yet imperfect, collections of related terms which with only minimal constraints applied, may give rise to subsets of terms representing more strongly coherent classes.

Contributions

A TF×IDF like method is introduced for deriving labeled classes of instances from unstructured, open domain text. Based on the filtering of *is-a* extraction pairs through the use of distributionally similar terms, the method allows for a smooth trade-off between precision and recall, giving results such as 440 classes at 91% accuracy and 8,572 classes with 86% accuracy, representing an improvement over the state of the art in label extraction. Resulting classes may be successfully used as input to existing information extraction techniques, as demonstrated by experiments in class attribute extraction.

Extraction Method

Algorithm

The algorithm described in Figure 1 assumes access to a large collection of pairs, $\mathcal{P}$, suggesting mappings from *instances* (e.g., *george bush*) to *class labels* (e.g., *president*). When $\langle I, L \rangle \in \mathcal{P}$, L is considered a proposed *label* of *instance* I. Collections of such pairs may be extracted using pattern based methods originating from Hearst (1992), in which template patterns such as *[X is a Y]* and *[Y such as X, X', and X"]* are applied to a large set of documents. For example, "*Sales of little cars, such as the Fiat Panda, are booming.*" gives $\{\langle \textit{Fiat Panda}, \textit{car} \rangle, \ldots\}$. This method is prone to mistakes, such as with a sentence like "*Ford has product lines beyond small cars, such as the F-150.*", creating the need for a filtering procedure as described in this

*Contributions made during an internship at Google.

Given: $\mathcal{I}$: set of instance phrases
$\mathcal{L}$: set of label phrases
$\mathcal{C}$: partitioning of $\mathcal{I}$ by distributional similarity
$\mathcal{P} \subseteq \mathcal{I} \times \mathcal{L}$: set of *is-a* phrase pairs
Returns: $\mathcal{P}_{JK} \subseteq \mathcal{P}$: set of filtered phrase pairs
Parameters: $J \in [0, 1]$: label freq. constraint (intra-cluster)
$K \in \mathbb{N}$: label freq. constraint (inter-cluster)
Algorithm:
Let $\mathcal{P}_{JK} = \{\}$
For each semantic cluster $S \in \mathcal{C}$:
For each class label L, where $\exists I \in S$ s.t. $\langle I, L\rangle \in \mathcal{P}$:
Let $S_L = \{I | I \in S, \langle I, L\rangle \in \mathcal{P}\}$
Let $\mathcal{C}_L = \{S' | S' \in \mathcal{C}, \exists I \in S' : \langle I, L\rangle \in \mathcal{P}\}$
If $|S_L| > J \times |S|$:
If $|\mathcal{C}_L| < K$:
Set $\mathcal{P}_{JK} = \mathcal{P}_{JK} \cup \{\langle I, L\rangle | I \in S, \langle I, L\rangle \in \mathcal{P}\}$

Figure 1: Algorithm for extracting ⟨instance, class label⟩ pairs.

paper.

Additionally required are clusters of semantically related phrases, $\mathcal{C}$, such as $\{$*george bush*, *bill clinton*, ... $\}$. These clusters taken together are a partitioning of $\mathcal{I}$, the instance vocabulary. Lin & Pantel (2002) gave a method for building such clusters based on *distributional similarity*. For example, the sentences "*Clinton vetoed the bill*" and "*Bush vetoed the bill*" suggest that *Clinton* and *Bush* may be semantically related.

The algorithm begins by initializing the return value $\mathcal{P}_{JK}$ to the empty set. Each semantic cluster S contained within $\mathcal{C}$ is then considered in turn. For each label L that labels at least one instance in S, the algorithm verifies whether the number of such instances paired with L is at least J of the size of S. For example, if 37 instances in a cluster of 50 elements each had the label *president*, then if $37/50 > J$, *president* would be viable.

If a label is viable based on the intra-cluster constraint, we then verify whether it is acceptable according to the inter-cluster constraint K. $\mathcal{C}_L$ is the set of all clusters where at least one member of each cluster is paired with the label L. If the number of such clusters, $|\mathcal{C}_L|$, is less than K, we consider L to be a good label for the supporting instances in S. Each instance in S that is paired with L is added to our filtered collection $\mathcal{P}_{JK}$, representing an assignment of those instances to the class specified by L.

Continuing our example, if there were 5 clusters that each had at least one element labeled *president*, and $K > 5$, then each of the elements in the cluster under consideration having the label *president* would be recorded as true instances of that class.

Discussion

From an information retrieval perspective, the clusters provided as input to the extraction algorithm can be seen as documents, whereas the class labels are equivalent to document terms. In this light, the extraction algorithm offers what the traditional TF×IDF weighting scheme offers in information retrieval. The normalized term frequency, TF, is the number of instances in a cluster initially assigned a given label divided by the total number of instances in that cluster. Note that while this TF-like score is a 0-1 measure on relative frequency, it is possible in our case for the TF of distinct labels assigned to members of the same cluster to sum to a value greater than one.[1] Our parameter J directly constrains the TF as described.

IDF is usually considered to be the log of the total number of documents, first divided by the number of documents with the given term. This value is used based on the belief that terms (labels) with wide distribution across documents (clusters) are less significant than those occurring more narrowly. As done with IDF values, our use of the parameter K allows for limiting the "spread" of a term (class label). However, we desired the ability to regulate this spread directly in terms of the number of clusters covered (e.g., 1, 2, ..., 30, ...), rather than the log of the relative percentage.

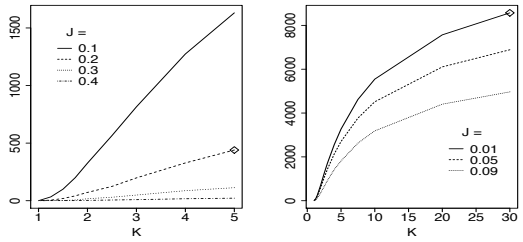

Figure 2: Number of classes extracted at strict, and less prohibitive settings of J and K.

size	number	size	number
$\leq \infty$	8,572	≤ 25	4,322
≤ 500	8,311	≤ 10	1,681
≤ 50	6,089	≤ 5	438

Table 1: For $J = 0.01$, $K = 30$, number of classes whose size $\leq N$.

Experimental Setting

Data

Experiments relied on the unstructured text available within a collection of approximately 100 million Web documents in English, as available in a Web repository snapshot from 2006 maintained by the Google search engine. The textual portion of the documents were cleaned of HTML, tokenized, split into sentences and part-of-speech tagged using the TnT tagger (Brants 2000). Clusters of related terms were collected similarly to Lin & Pantel (2002). Initial instance, label pairs were extracted in a manner similar to Hearst (1992).

J	K	$\lvert\mathcal{P}_{JK}\rvert$	$\lvert\mathcal{I}_{JK}\rvert$	$\lvert\mathcal{I}_{JK} \setminus \mathcal{I}_{400k}\rvert$	$\lvert\mathcal{L}_{JK}\rvert$
0	∞	44,178,689	880,535	744,890	7,266,464
0.01	30	715,135	262,837	191,012	8,572
0.2	5	52,373	36,797	21,309	440

Table 2: For given values of J and K, the number of: instance-label pairs ($\mathcal{P}_{JK}$); instances ($\mathcal{I}_{JK}$); instances after removing those also appearing in WN400k ($\mathcal{I}_{JK} \setminus \mathcal{I}_{400k}$); class labels ($\mathcal{L}_{JK}$). Note that when $J = 0$ and $K = \infty$, $\mathcal{P}_{JK} = \mathcal{P}$.

Extraction

Classes were extracted across a range of parameter settings. As one should expect, Figure 2 shows that as J (the requirement on the number of instances within a class that must share a label for it to be viable) is lowered, one sees a corresponding increase in the number of resultant classes. Similarly, the more distributed across clusters a label is allowed to be (K), also the larger the return. Table 1 shows the distribution of class sizes at a particular parameter setting.

Evaluation

Unless otherwise stated, evaluations were performed based on results gathered using either *wide* ($J = 0.01, K = 30$) or *narrow* ($J = 0.2, K = 5$) settings of J and K, under the procedure described. Certain experiments made use of an automatically expanded version of WordNet (Snow, Jurafsky, & Ng 2006) containing 400,000+ synsets, referred to here as WN400k. In those cases where the judgement task was binary (i.e., *good* or *bad*), subscripts given for precision scores reflect Normal based, 95% confidence intervals.

Instance Vocabulary

To determine the quality of the underlying instance vocabulary, one hundred randomly selected instances were assessed for narrow and wide settings, independent of their proposed class labels (relevant population sizes shown in Table 2). As can be seen in Table 3, the vocabulary was judged to be near perfect. Quality was determined by both authors coming to agreement on the usefulness of each term. As a control, an additional set of ten elements, drawn from WN400k, were mixed within each sample (for a total of 110). Of these control items, only one was deemed questionable.[2]

Examples of both positive and negative judgements can be seen in Table 4. Instances such as *fast heart rate*, *local produce*, and *severe itching* were considered proper, despite adjectival modification, as they are common enough terms to warrant being treated as unique vocabulary items (each of these appear in unmodified form in WordNet 3.0). Allowing instances such as *moles and voles* is complicated by the worry of needing to allow an exponential number of such conjunctive pairings. While the presence of a pair in text gives evidence towards considering it as a stand-alone instance, conjunctives were conservatively rejected unless they formed a proper name, such as a movie title or a band name.[3]

Types of events, such as *finding directions* or *swimming instruction*, were explicitly allowed as these may have their own attributes.[4]

J	K	Good	%
0.01	30	97/100	$97_{\pm 3.3}\%$
0.2	5	98/100	$98_{\pm 2.7}\%$

Table 3: Assessed quality of underlying instances.

Instance	Good?	Instance	Good?
fast heart rate	yes	*electric bulb*	yes
local produce	yes	*south east*	yes
severe itching	yes	*finding directions*	yes
moles and voles	no	*h power*	no

Table 4: Select examples from instance assessment.

Class Labels

Table 5 summarizes the results of manually evaluating sets of 100 randomly selected pairs for both wide ($J = 0.01, K = 30$) and narrow ($J = 0.2, K = 5$) parameter settings. In order to establish a baseline, a similar sample was assessed for pairs taken directly from the input data. To evaluate novelty, an additional assessment was done after removing all pairs containing a term occurring in WN400k.

As shown, our method was successful in separating high quality class/instance pairs from the lower average quality input data. Even when removing many of the more common instances as found in WN400k, quality remained high. Also in this case there appeared a statistically significant difference between the wide and narrow settings.

That the number of classes dramatically rose inversely to J, yet still retained high quality at 0.01, supports the intuition that the labeling method of Pantel & Ravichandran (2004) might ignore potentially useful sets of instances that have the misfortune of being scattered across a small number of semantically related clusters.

Examples from the evaluation on pairs can be seen in Table 6. Subjective labels such as *favorite authors* were considered bad due to difficulties in assigning a clear interpretation.[5] Similarly disallowed were class labels overly reliant on context, such as *main settlements* and *applicable laws*.

[1] That is, it may occur that many of the instances in a single cluster share the same labels. For example, the labels *politician*, *public speaker*, and *elected official* may each be assigned to the same 50% of a cluster (giving them each a TF of 0.5).

[2] The questionable item was *christopher reeves*, whose proper spelling, *christopher reeve*, is not contained in WN400k.

[3] Thus excluding such gems as *oil and water* which (as popularly known) *don't mix*.

[4] E.g., *Finding directions [is frustrating]*, or *Swimming instruction [is inexpensive]*.

[5] For instance, *Albert Einstein* may be a *famous scientist*, suggesting the label as a worthwhile class, but what about *Alan Turing*? Under a conjunctive reading, Famous(x) & Scientist(x), we might say no, but under a functional reading, Famous(Scientist)(x), it may be more appropriate (e.g., *Alan Turing is famous amongst scientists*, or *Compared to other scientists, Alan Turing is famous*).

J	K	$\mathcal{P}_{JK}$		$\mathcal{P}_{JK} \setminus \{\langle \mathcal{I}_{400k}, \cdot \rangle\}$	
		Eval	Precision	Eval	Precision
0	∞	34/100	$34_{\pm 9.3}\%$	27/100	$27_{\pm 8.1}\%$
0.01	30	86/100	$86_{\pm 6.9}\%$	75/100	$75_{\pm 8.5}\%$
0.2	5	91/100	$91_{\pm 5.6}\%$	95/100	$95_{\pm 4.3}\%$

Table 5: Quality of pairs, before and after removing instances already in WN400k.

Instance	Class	Good?
go-karting	*outdoor activities*	yes
ian and sylvia	*performers*	yes
italian foods	*foods*	yes
international journal	*professional journal*	no
laws of florida	*applicable laws*	no
farnsfield	*main settlements*	no
ellroy	*favorite authors*	no

Table 6: Interesting or questionable pairs.

The instance *international journal* is an example of imperfect input data; likely it was a substring of, e.g., *international journal of epidemiology*. Conversely, *ian and sylvia* are a pair of singers that performed as a group, which we allow.

One observed pair, $\langle$*wild turkeys*, *small mammals*$\rangle$, led to a manual search to determine possible source sentences. The double-quoted query *"mammals such as * wild turkeys"*[6] was submitted to the Google search engine, giving a total of six results, including:

- [White Ash] Provides a late winter food source for birds and small mammals such as wild turkey, evening grosbeak, cedar waxwings and squirrels.

- A wide diversity of songbirds and birds of prey, as well as mammals such as deer, wild turkeys, raccoons, and skunks, benefit from forest fragmentation.

- Large mammals such as deer and wild turkeys can be seen nearly year-round.

The first sentence highlights the previously mentioned pitfalls inherent in using template based patterns. The second sentence is arguably ambiguous, although the most natural interpretation is false. The final sentence is worse yet, exemplifying the intuition that Web text is not always trustworthy.

Expanding a Class

In some cases it may be necessary or desired to expand the size of a given class through select relaxation of constraints (i.e., less restrictive values of J and/or K for pre-specified a pre-specified label L).

To understand the potential effects on quality such relaxation may have, three classes based on size were randomly selected from three separate ranges: small classes (< 50), *prestigious private schools*, *telfair homebuilders*, *plant tissues*; medium classes (< 500), *goddesses*, *organisms*, *enzymes*; and large classes ($>= 500$), *flavors*, *critics*, *dishes*.

[6]Query was performed with and without wildcard.

writers	*american*, *ancient*, *british*, *christian*, *classical*, *contemporary*, *english*, *famous*, *favorite*, *french*, *great*, *greek*, *indian*, *prominent*, *roman*, *romance*, *spanish*, *talented*, *veteran*
weapons	*advanced*, *bladed*, *current*, *dangerous*, *deadly*, *lethal*, *powerful*, *projectile*, *small*, *smart*, *sophisticated*, *traditional*

Table 7: Candidate refinements discovered for the classes *writers* and *weapons*.

Each of these classes were required to have shown a growth in size greater than 50% between the most and least restrictive parameter settings explored.

Up to 50 instances were sampled from the minimum sized versions of each class. From the respective maximum sized versions, a similar number of instances were sampled from those left remaining once elements also appearing in the minimum set were removed (i.e., the sample came only from the instances added as a result of loosening parameter constraints).[7]

For each of the three small classes, accuracy was judged 100% both before and after expansion; proving that even small, precise classes do not always necessarily land together in distributionally similar clusters.

If a class expands as constraints are loosened, then new members must derive from clusters previously not contributing to the class. In some cases this might result in a class being incorrectly "spread out" over clusters of instances that are related to, but not precise members of, the given class. For example: for the class *goddesses*, many of the additional instances were actually male deities; in the case of *enzymes*, most of the newly added instances were amino acids which made up, but did not fully constitute, an enzyme. Quantitatively, the average number of such *nearly correct* instances increased from 40% to 66% for the class *enzymes* and 29% to 44% for the class *goddesses*.

Handling Pre-nominal Adjectives

Many of the classes obtained by this method, especially those with few members, had labels containing pre-nominal adjective modification. For example, Table 7 gives each of the discovered refinements for the classes *writers* and *weapons*, most of which would be evaluated as being distinct, useful subclasses.

Table 8 shows the number of classes, by size, whose label contained two words or more, and where the first word appeared in the adjective listing of WordNet 3.0. For instance, of the 32 classes containing between 2^{11} and 2^{12} instances, only one label (3%) was adjective initial. Compare this to the 936 classes containing between 4 and 8 instances, where 455 (49%) of the labels began with a term with an adjective reading.

[7]For example, a class of 10 members under restrictive settings that grew to a size of 16 as constraints were relaxed would be a viable "small" candidate, with 8 elements then sampled each from the min (=10) and max (=16) versions of the class.

S	Ratio	%	S	Ratio	%
2^{12}	0/4	0%	2^{6}	342/961	36%
2^{11}	1/32	3%	2^{5}	627/1566	40%
2^{10}	4/73	5%	2^{4}	860/1994	43%
2^{9}	19/143	13%	2^{3}	852/1820	47%
2^{8}	68/259	26%	2^{2}	455/936	49%
2^{7}	170/541	31%	2^{1}	108/243	44%

Table 8: For each range, the number of classes with a label whose first term has an adjective reading. $S = \lfloor size \rfloor$.

Instance	Class
lamborghini murcielago	*real cars*
spanish primera division	*domestic leagues*
dufferin mall	*nearby landmarks*
colegio de san juan de letran	*notable institutions*
fitness exercise	*similar health topics*

Table 9: Examples of ⟨instance, class⟩ pairs sampled from amongst classes with less than 10 members, and where the label was deemed unacceptable.

By limiting the number of clusters (via K) that may contain instances of a class, this necessarily penalizes classes that may legitimately be of exceptionally large size. Discovery of these classes is sacrificed in order that large numbers of bad labels, which characteristically tend to occur across many clusters, are filtered out. However, it is possible to recover some of these large classes by recognizing and removing adjectival refinements from accepted labels, and merging the results into a larger, coarser class; for example each of the sub-types of *writers* being considered members of single, simplified class.[8]

When leading adjectives were removed, with the exception of lexicalized terms as found in WordNet 3.0[9], 8,572 classes reduced to 3,397.

Informal sampling gave evidence that the smallest classes being extracted tended to be of lower quality than those of larger size, primarily due to leading subjective adjectives (examples seen in Table 9). This prompted an evaluation of 100 pairs taken from classes with less than 10 elements, which were judged to have an average quality of just $71_{\pm 5}\%$. After removing initial adjectives this improved to 91%. Adding a check for lexicalized terms raised this to 92% (where a 1% gain is not statistically significant for a sample of this size).

Task-Based Evaluation

For a better understanding of the usefulness of the extracted classes beyond the high accuracy scores achieved in manual evaluations, a separate set of experiments used the extracted classes as input data for the task of extracting attributes (e.g., *circulatory system*, *life cycle*, *evolution* and *food chain*) of various classes (e.g., *marine animals*). The experiments followed an approach introduced in (Paşca 2007), which acquires lists of ranked class attributes from query logs, based on a set of instances and a set of seed attributes provided as input for each class. The only modification was a tweak in the internal representation and ranking of candidate attributes, to allow for the extraction of attributes when five seed attributes are provided for only one class, rather than for each input class as required in the original approach. Thus, a ranked list of attributes were extracted automatically for each of the classes generated by our algorithm when J is set to 0.01 and K is set to 30, from a random sample of 50 million unique, fully-anonymized queries in English submitted by Web users to the Google search engine in 2006.

Each attribute in the extracted ranked lists was assigned a score of 1, if the attribute was *vital*, i.e., it must be present in an ideal list of attributes of the class; 0.5, if the attribute was *okay*, as it provides useful but non-essential information; or 0, if the attribute was incorrect. Precision at some rank N in a list was thus measured as the sum of the assigned values of the first N candidate attributes, divided by N. When evaluated over a random sample of 25 classes out of the larger set of classes acquired from text, the open-domain classes extracted in this paper produced attributes at accuracy levels reaching 70% at rank 10, and 67% at rank 20. For example, for the class label *forages*, which was associated to a set of instances containing *alsike clover*, *rye grass*, *tall fescue*, *sericea lespedeza* etc., the ranked list of extracted attributes was [*types*, *picture*, *weed control*, *planting*, *uses*, *information*, *herbicide*, *germination*, *care*, *fertilizer*, ...]. A more detailed analysis of the benefits of automatically-derived open-domain classes of instances in the task of class attribute extraction is presented in (Paşca & Van Durme 2008).

Previous Work

The given algorithm is most similar to that described by Pantel & Ravichandran (2004), in that both begin with clusters of instances grouped by distributional similarity. However, where the goal of those authors was to assign labels that best fit for a given cluster, the method presented here is meant to assign instances that best fit for a given label. To highlight this distinction, consider a hypothetical cluster of proper nouns, each standing for a US politician. Some of these may be senators, some of them presidents, a few may be persons of political importance that have never actually held office. The most coherent label for this set may be *politician* or *representative*, which Pantel & Ravichandran apply universally to all members of the set. Meanwhile, a second cluster may exist containing mostly historical American leaders such as *Abe Lincoln*, *George Washington*, and *Ben Franklin*, this set having the dominant label of *president*. The method presented here is aimed at teasing apart these related sets in order to assign the label, e.g., *president*, to instances of multiple clusters, and only to those instances where there is direct evidence to support it. This allows for more conservative class assignment (leading to higher precision), and a greater diversity of classes. On 1,432 clusters, Pantel & Ravichandran reported a labelling precision

[8] This heuristic fails for nonsubsective adjectives, such as those known as *privatives*, exemplified by *fake flower* and *former democracy* (Kamp & Partee 1995).

[9] E.g., {*new york*} *clubs*, {*romance languages*}, {*gold medal*} *winners*.

of 72%, with the average cluster size not provided. The relevant entries in Table 10 refer to their success at hypernym labeling based on labels extracted for top three members of each cluster; a task more directly comparable to the focus here. *All* refers to their precision for all hypernym labels collected, while *Proper* is for the subset of *All* dealing with proper nouns.

Snow, Jurafsky, & Ng (2006) gave a model for instance tagging with the added benefit of possibly choosing the proper *sense* of a proposed label. By taking advantage of the pre-existing structural constraints imposed by WordNet, the authors report a *fine grain* (sense differentiated) labelling precision of 68% on 20,000 pairs. While not directly reported, one may derive the *non* sense differentiated precision based on given fine grain, and disambiguated precision scores. For the task targeted in this paper their system achieved 69.4% accuracy on 20,000 pairs (termed WN 20K in Table 10).[10] By comparison, at narrow settings of J and K, over 50,000 pairs were labeled with a judged precision of 91%, and without the use of a manually specified taxonomy.

The comparative evaluation scores reported in Table 10 derive from judgements made by the respective authors independently, on different datasets. As such, the scores inherently include subjective components which suggest that the scores should be taken as only reference, with respect to estimated coverage and precision of different methods. It is reasonable to interpret the results from Table 10 as an indication that the method given here is competitive with state of the art, though the results should not be used for strict objective ranking against previous work.

Wang & Cohen (2007) gave state of the art results for a seed-based approach to *class clustering*, a subpart of the problem considered here. The authors need specify three examples for a given class in order for their system to automatically induce a *wrapper* for scraping similar instances from semi-structured content in Web documents. Average precision scores of 93-95% were reported for experiments on 12 classes, across English, Chinese and Japanese texts. In comparison, our method requires no seeds, does not make use of semi-structured data, and provides class labels along with representative instances. Note that one would be required to specify roughly 10,000 examples for this seed-based method to acquire even the 3,397 merged, adjective pruned classes described earlier. Future work may consider using instances as provided by the approach given in this paper in order to bootstrap such a seed-based system.

Conclusion

A method was given for the extraction of large numbers of concept classes along with their corresponding instances. Through the use of two, simply regulated constraints, imperfect collections of semantically related terms and ⟨instance, label⟩ pairs may be used together to generate large numbers of classes with state of the art precision.

[10] Snow, Jurafsky, & Ng give $c_1/total = 68/100$ as fine grain prec. when $total = 20,000$, with an associated disambiguated prec. of $c_1/(c_1 + c_2) = 98/100$. Label precision prior to sense selection must therefore be $(c_1 + c_2)/total = 69.4\%$.

Source	$\|\mathcal{P}\|$	Precision
WN 1K	1,000	93%
WN 20K	20,000	69.4%
CBC Proper	65,000	81.5%
CBC All	159,000	68%
JK Narrow	52,373	91%
JK Wide	715,135	86%

Table 10: Comparison to coverage and precision results reported in the literature, CBC (Pantel & Ravichandran 2004), WN (Snow, Jurafsky, & Ng 2006).

Acknowledgements

The authors are grateful to: the anonymous referees for suggestions of previous work; Dekang Lin for providing experimental support; and Deepak Ravichandran for general comments and feedback.

References

Brants, T. 2000. TnT - a statistical part of speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP-00)*, 224–231.

Caraballo, S. 1999. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics (ACL-99)*, 120–126.

Hearst, M. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING-92)*, 539–545.

Kamp, H., and Partee, B. 1995. Prototype theory and compositionality. *Cognition* 57(2):129–191.

Lin, D., and Pantel, P. 2002. Concept discovery from text. In *Proceedings of the 19th international conference on Computational linguistics (COLING-02)*, 1–7.

Paşca, M., and Van Durme, B. 2008. Weakly-supervised acquisition of open-domain classes and class attributes from Web documents and query logs. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*.

Paşca, M. 2007. Organizing and searching the World Wide Web of facts - step two: Harnessing the wisdom of the crowds. In *Proceedings of the 16th World Wide Web Conference (WWW-07)*.

Pantel, P., and Ravichandran, D. 2004. Automatically labeling semantic classes. In *Proceedings of the 2004 Human Language Technology Conference (HLT-NAACL-04)*, 321–328.

Schubert, L. 2006. Turing's dream and the knowledge challenge. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-06)*.

Snow, R.; Jurafsky, D.; and Ng, A. Y. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Proceedings of the 19th Conference on Neural Information Processing Systems (NIPS-17)*. MIT Press.

Snow, R.; Jurafsky, D.; and Ng, A. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL-06)*, 801–808.

Wang, R. C., and Cohen, W. W. 2007. Language-Independent Set Expansion of Named Entities using the Web. In *Proceedings of IEEE International Conference on Data Mining (ICDM 2007)*.