Weakly-Supervised Acquisition of Open-Domain Classes and Class Attributes from Web Documents and Query Logs

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

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

Abstract

A new approach to large-scale information extraction exploits both Web documents and query logs to acquire thousands of open-domain classes of instances, along with relevant sets of open-domain class attributes at precision levels previously obtained only on small-scale, manually-assembled classes.

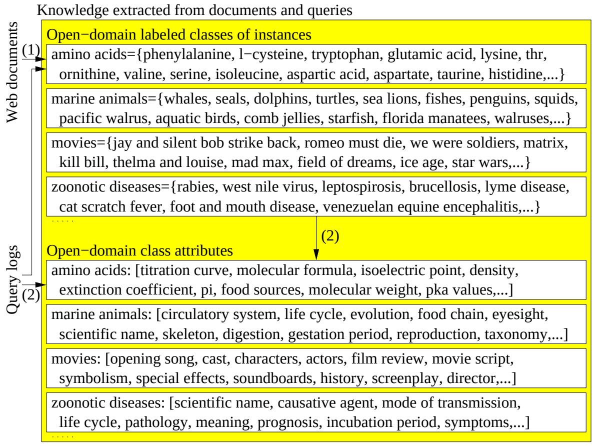
1 Introduction

Current methods for large-scale information extraction take advantage of unstructured text available from either Web documents (Banko et al., 2007; Snow et al., 2006) or, more recently, logs of Web search queries (Paşca, 2007) to acquire useful knowledge with minimal supervision. Given a manually-specified target attribute (e.g., birth years for people) and starting from as few as 10 seed facts such as (e.g., *John Lennon*, *1941*), as many as a million facts of the same type can be derived from unstructured text within Web documents (Paşca et al., 2006). Similarly, given a manually-specified target class (e.g., *Drug*) with its instances (e.g., *Vicodin* and *Xanax*) and starting from as few as 5 seed attributes (e.g., *side effects* and *maximum dose* for *Drug*), other relevant attributes can be extracted for the same class from query logs (Paşca, 2007). These and other previous methods require the manual specification of the input classes of instances before any knowledge (e.g., facts or attributes) can be acquired for those classes.

The extraction method introduced in this paper mines a collection of Web search queries and a collection of Web documents to acquire open-domain classes in the form of instance sets (e.g., {*whales*, *seals*, *dolphins*, *sea lions*,...}) associated with class labels (e.g., *marine animals*), as well as large sets of open-domain attributes for each class (e.g., *circulatory system*, *life cycle*, *evolution*, *food chain* and *scientific name* for the class *marine animals*). In this light, the contributions of this paper are fourfold. First, instead of separately addressing the tasks of collecting unlabeled sets of instances (Lin, 1998), assigning appropriate class labels to a given set of instances (Pantel and Ravichandran, 2004), and identifying relevant attributes for a given set of classes (Paşca, 2007), our integrated method from Section 2 enables the *simultaneous extraction* of class instances, associated labels and attributes. Second, by exploiting the contents of query logs during the extraction of labeled classes of instances from Web documents, we acquire thousands (4,583, to be exact) of *open-domain classes* covering a wide range of topics and domains. The accuracy reported in Section 3.2 exceeds 80% for both instance sets and class labels, although the extraction of classes requires a remarkably small amount of supervision, in the form of only a few commonly-used Is-A extraction patterns. Third, we conduct the first study in extracting attributes for thousands of open-domain, *automatically-acquired* classes, at precision levels over 70% at rank 10, and 67% at rank 20 as described in Section 3.3. The amount of supervision is limited to five seed attributes provided for only one reference class. In comparison, the largest previous

[*] Contributions made during an internship at Google.

Proceedings of ACL-08: HLT, pages 19–27,
Columbus, Ohio, USA, June 2008. ©2008 Association for Computational Linguistics


Knowledge extracted from documents and queries
Web documents
Query logs
(1)
(2)
(2)
Open−domain labeled classes of instances
Open−domain class attributes
.....
.....

amino acids={phenylalanine, l−cysteine, tryptophan, glutamic acid, lysine, thr, ornithine, valine, serine, isoleucine, aspartic acid, aspartate, taurine, histidine,...}

marine animals={whales, seals, dolphins, turtles, sea lions, fishes, penguins, squids, pacific walrus, aquatic birds, comb jellies, starfish, florida manatees, walruses,...}

movies={jay and silent bob strike back, romeo must die, we were soldiers, matrix, kill bill, thelma and louise, mad max, field of dreams, ice age, star wars,...}

zoonotic diseases={rabies, west nile virus, leptospirosis, brucellosis, lyme disease, cat scratch fever, foot and mouth disease, venezuelan equine encephalitis,...}

amino acids: [titration curve, molecular formula, isoelectric point, density, extinction coefficient, pi, food sources, molecular weight, pka values,...]

marine animals: [circulatory system, life cycle, evolution, food chain, eyesight, scientific name, skeleton, digestion, gestation period, reproduction, taxonomy,...]

movies: [opening song, cast, characters, actors, film review, movie script, symbolism, special effects, soundboards, history, screenplay, director,...]

zoonotic diseases: [scientific name, causative agent, mode of transmission, life cycle, pathology, meaning, prognosis, incubation period, symptoms,...]

Figure 1: Overview of weakly-supervised extraction of class instances, class labels and class attributes from Web documents and query logs

study in attribute extraction reports results on a set of 40 *manually-assembled* classes, and requires five seed attributes to be provided as input for each class. Fourth, we introduce the first approach to information extraction from a combination of *both* Web documents and search query logs, to extract open-domain knowledge that is expected to be suitable for later use. In contrast, the textual data sources used in previous studies in large-scale information extraction are *either* Web documents (Mooney and Bunescu, 2005; Banko et al., 2007) *or*, recently, query logs (Paşca, 2007), but not both.

2 Extraction from Documents and Queries

2.1 Open-Domain Labeled Classes of Instances

Figure 1 provides an overview of how Web documents and queries are used together to acquire open-domain, labeled classes of instances (phase (1) in the figure); and to acquire attributes that capture quantifiable properties of those classes, by mining query logs based on the class instances acquired from the documents, while guiding the extraction based on a few attributes provided as seed examples (phase (2)).

As described in Figure 2, the algorithm for deriving labeled sets of class instances starts with the acquisition of candidate pairs $\{\mathcal{M}_E\}$ of a class label and an instance, by applying a few extraction patterns to unstructured text within Web documents $\{\mathcal{D}\}$, while guiding the extraction by the contents of query logs $\{\mathcal{Q}\}$ (Step 1 in Figure 2). This is followed by the generation of unlabeled clusters $\{\mathcal{S}\}$ of distributionally similar queries, by clustering vectors of contextual features collected around the occurrences of queries $\{\mathcal{Q}\}$ within documents $\{\mathcal{D}\}$ (Steps 2 and 3). Finally, the intermediate data $\{\mathcal{M}_E\}$ and $\{\mathcal{S}\}$ is merged and filtered into smaller, more accurate labeled sets of instances (Steps 4 through 15).

```
Input: set of Is-A extraction patterns $\{\mathcal{E}\}$
.        large repository of search queries $\{\mathcal{Q}\}$
.        large repository of Web docs $\{\mathcal{D}\}$
.        weighting parameters $\mathcal{J}\in$[0,1] and $\mathcal{K}\in\overline{1..\infty}$
Output: set of pairs of a class label and an instance $\{<\mathcal{C},\mathcal{I}>\}$
Variables: $\{\mathcal{S}\}$ = clusters of distributionally similar phrases
.          $\{\mathcal{V}\}$ = vectors of contextual matches of queries in text
.          $\{\mathcal{M}_E\}$ = set of pairs of a class label and an instance
.          $\{\mathcal{C}_S\}$ = set of class labels
.          $\{\mathcal{X}\}$, $\{\mathcal{Y}\}$ = sets of queries
Steps:
01. $\{\mathcal{M}_E\}$ = Match patterns $\{\mathcal{E}\}$ in docs $\{\mathcal{D}\}$ around $\{\mathcal{Q}\}$
02. $\{\mathcal{V}\}$ = Match phrases $\{\mathcal{Q}\}$ in docs $\{\mathcal{D}\}$
03. $\{\mathcal{S}\}$ = Generate clusters of queries based on vectors $\{\mathcal{V}\}$
04. For each cluster of phrases $\mathcal{S}$ in $\{\mathcal{S}\}$
05.   $\{\mathcal{C}_S\}$ = $\emptyset$
06.   For each query $\mathcal{Q}$ of $\mathcal{S}$
07.     Insert labels of $\mathcal{Q}$ from $\{\mathcal{M}_E\}$ into $\{\mathcal{C}_S\}$
08.   For each label $\mathcal{C}_S$ of $\{\mathcal{C}_S\}$
09.     $\{\mathcal{X}\}$ = Find queries of $\mathcal{S}$ with the label $\mathcal{C}_S$ in $\{\mathcal{M}_E\}$
10.     $\{\mathcal{Y}\}$ = Find clusters of $\{\mathcal{S}\}$ containing some query
10.           with the label $\mathcal{C}_S$ in $\{\mathcal{M}_E\}$
11.     If $|\{\mathcal{X}\}| > \mathcal{J}\times|\{\mathcal{S}\}|$
12.       If $|\{\mathcal{Y}\}| < \mathcal{K}$
13.         For each query $\mathcal{X}$ of $\{\mathcal{X}\}$
14.           Insert pair $<\mathcal{C}_S,\mathcal{X}>$ into output pairs $\{<\mathcal{C},\mathcal{I}>\}$
15. Return pairs $\{<\mathcal{C},\mathcal{I}>\}$
```

Figure 2: Acquisition of labeled sets of class instances

Step 1 in Figure 2 applies lexico-syntactic patterns $\{\mathcal{E}\}$ that aim at extracting Is-A pairs of an instance (e.g., *Google*) and an associated class label (e.g., *Internet search engines*) from text. The two patterns, which are inspired by (Hearst, 1992) and have been the de-facto extraction technique in previous work on extracting conceptual hierarchies from text (cf. (Ponzetto and Strube, 2007; Snow et al., 2006)), can be summarized as:

$\langle$[..] $\mathcal{C}$ [such as|including] $\mathcal{I}$ [and|,|.]$\rangle$,

where $\mathcal{I}$ is a potential instance (e.g., *Venezuelan equine encephalitis*) and $\mathcal{C}$ is a potential class label for the instance (e.g., *zoonotic diseases*), for example in the sentence: *"The expansion of the farms increased the spread of zoonotic diseases such as Venezuelan equine encephalitis [..]"*.

During matching, all string comparisons are case-insensitive. In order for a pattern to match a sentence, two conditions must be met. First, the class label $\mathcal{C}$ from the sentence must be a non-recursive noun phrase whose last component is a plural-form noun (e.g., *zoonotic diseases* in the above sentence). Second, the instance $\mathcal{I}$ from the sentence must also occur as a complete query somewhere in the query logs $\{\mathcal{Q}\}$, that is, a query containing the instance and nothing else. This heuristic acknowledges the difficulty of pinpointing complex entities within documents (Downey et al., 2007), and embodies the hypothesis that, if an instance is prominent, Web search users will eventually ask about it.

In Steps 4 through 14 from Figure 2, each cluster is inspected by scanning all labels attached to one or more queries from the cluster. For each label $\mathcal{C}_S$, if a) $\{\mathcal{M}_E\}$ indicates that a large number of all queries from the cluster are attached to the label (as controlled by the parameter $\mathcal{J}$ in Step 12); and b) those queries are a significant portion of all queries from all clusters attached to the same label in $\{\mathcal{M}_E\}$ (as controlled by the parameter $\mathcal{K}$ in Step 13), then the label $\mathcal{C}_S$ and each query with that label are stored in the output pairs $\{<\mathcal{C},\mathcal{I}>\}$ (Steps 13 and 14). The parameters $\mathcal{J}$ and $\mathcal{K}$ can be used to emphasize precision (higher $\mathcal{J}$ and lower $\mathcal{K}$) or recall (lower $\mathcal{J}$ and higher $\mathcal{K}$). The resulting pairs of an instance and a class label are arranged into sets of class instances (e.g., {*rabies*, *west nile virus*, *leptospirosis*,...}), each associated with a class label (e.g., *zoonotic diseases*), and returned in Step 15.

2.2 Open-Domain Class Attributes

The labeled classes of instances collected automatically from Web documents are passed as input to phase (2) from Figure 1, which acquires class attributes by mining a collection of Web search queries. The attributes capture properties that are relevant to the class. The extraction of attributes exploits the set of class instances rather than the associated class label, and consists of four stages:

1) identification of a noisy pool of candidate attributes, as remainders of queries that also contain one of the class instances. In the case of the class *movies*, whose instances include *jay and silent bob strike back* and *kill bill*, the query *"cast jay and silent bob strike back"* produces the candidate attribute *cast*;

2) construction of internal search-signature vector representations for each candidate attribute, based on queries (e.g., *"cast selection for kill bill"*) that contain a candidate attribute (*cast*) and a class instance (*kill bill*). These vectors consist of counts tied to the frequency with which an attribute occurs with a given "templatized" query. The latter replaces specific attributes and instances from the query with common placeholders, e.g., *"X for Y"*;

3) construction of a reference internal search-signature vector representation for a small set of seed attributes provided as input. A reference vector is the normalized sum of the individual vectors corresponding to the seed attributes;

4) ranking of candidate attributes with respect to each class (e.g., *movies*), by computing similarity scores between their individual vector representations and the reference vector of the seed attributes.

The result of the four stages is a ranked list of attributes (e.g., [*opening song*, *cast*, *characters*,...]) for each class (e.g., *movies*).

In a departure from previous work, the instances of each input class are automatically generated as described earlier, rather than manually assembled. Furthermore, the amount of supervision is limited to seed attributes being provided for only one of the classes, whereas (Paşca, 2007) requires seed attributes for each class. To this effect, the extraction includes modifications such that only one reference vector is constructed internally from the seed attributes during the third stage, rather one such vector for each class in (Paşca, 2007); and similarity scores are computed cross-class by comparing vector representations of individual candidate attributes against the only reference vector available during the fourth stage, rather than with respect to the reference vector of each class in (Paşca, 2007).

3 Evaluation

3.1 Textual Data Sources

The acquisition of open-domain knowledge, in the form of class instances, labels and attributes, relies on unstructured text available within Web documents maintained by, and search queries submitted to, the Google search engine.

The collection of queries is a random sample of fully-anonymized queries in English submitted by Web users in 2006. The sample contains approximately 50 million unique queries. Each query is

Found in WordNet?	Count	Pct.	Examples
Yes (original)	1931	42.2%	baseball players, endangered species
Yes (removal)	2614	57.0%	caribbean countries, fundamental rights
No	38	0.8%	agrochemicals, celebs, handhelds, mangas

Table 1: Class labels found in WordNet in original form, or found in WordNet after removal of leading words, or not found in WordNet at all

accompanied by its frequency of occurrence in the logs. The document collection consists of approximately 100 million Web documents in English, as available in a Web repository snapshot from 2006. The textual portion of the documents is cleaned of HTML, tokenized, split into sentences and part-of-speech tagged using the TnT tagger (Brants, 2000).

3.2 Evaluation of Labeled Classes of Instances

Extraction Parameters: The set of instances that can be potentially acquired by the extraction algorithm described in Section 2.1 is heuristically limited to the top five million queries with the highest frequency within the input query logs. In the extracted data, a class label (e.g., *search engines*) is associated with one or more instances (e.g., *google*). Similarly, an instance (e.g., *google*) is associated with one or more class labels (e.g., *search engines* and *internet search engines*). The values chosen for the weighting parameters $\mathcal{J}$ and $\mathcal{K}$ from Section 2.1 are 0.01 and 30 respectively. After discarding classes with fewer than 25 instances, the extracted set of classes consists of 4,583 class labels, each of them associated with 25 to 7,967 instances, with an average of 189 instances per class.

Accuracy of Class Labels: Built over many years of manual construction efforts, lexical gold standards such as WordNet (Fellbaum, 1998) provide wide-coverage upper ontologies of the English language. Built-in morphological normalization routines make it straightforward to verify whether a class label (e.g., *faculty members*) exists as a concept in WordNet (e.g., *faculty member*). When an extracted label (e.g., *central nervous system disorders*) is not found in WordNet, it is looked up again after iteratively removing its leading words (e.g., *nervous system disorders*, *system disorders* and *disorders*).

Class Label={Set of Instances}	Parent in WordNet	C?
american composers={aaron copland, eric ewazen, george gershwin,...}	composers	Y
modern appliances={built-in oven, ceramic hob, tumble dryer,...}	appliances	S
area hospitals={carolinas medical center, nyack hospital,...}	hospitals	S
multiple languages={chuukese, ladino, mandarin, us english,...}	languages	N

Table 2: Correctness judgments for extracted classes whose class labels are found in WordNet only after removal of their leading words (C=Correctness, Y=correct, S=subjectively correct, N=incorrect)

As shown in Table 1, less than half of the 4,583 extracted class labels (e.g., *baseball players*) are found in their original forms in WordNet. The majority of the class labels (2,614 out of 4,583) can be found in WordNet only after removal of one or more leading words (e.g., *caribbean countries*), which suggests that many of the class labels correspond to finer-grained, automatically-extracted concepts that are not available in the manually-built WordNet. To test whether that is the case, a random sample of 200 class labels, out of the 2,614 labels found to be potentially-useful specific concepts, are manually annotated as correct, subjectively correct or incorrect, as shown in Table 2. A class label is: correct, if it captures a relevant concept although it could not be found in WordNet; subjectively correct, if it is relevant not in general but only in a particular context, either from a subjective viewpoint (e.g., *modern appliances*), or relative to a particular temporal anchor (e.g., *current players*), or in connection to a particular geographical area (e.g., *area hospitals*); or incorrect, if it does not capture any useful concept (e.g., *multiple languages*). The manual analysis of the sample of 200 class labels indicates that 154 (77%) are relevant concepts and 27 (13.5%) are subjectively relevant concepts, for a total of 181 (90.5%) relevant concepts, whereas 19 (9.5%) of the labels are incorrect. It is worth emphasizing the importance of automatically-collected classes judged as relevant and not present in WordNet: *caribbean countries*, *computer manufacturers*, *entertainment companies*, *market research firms* are arguably very useful and should probably be considered as part of

Class Label		Size of Instance Sets			Class Label		Size of Instance Sets		
M (Manual)	E (Extracted)	M	E	$\frac{M \cap E}{M}$	M (Manual)	E (Extracted)	M	E	$\frac{M \cap E}{M}$
Actor	actors	1500	696	23.73	Movie	movies	626	2201	30.83
AircraftModel	-	217	-	-	NationalPark	parks	59	296	0
Award	awards	200	283	13	NbaTeam	nba teams	30	66	86.66
BasicFood	foods	155	3484	61.93	Newspaper	newspapers	599	879	16.02
CarModel	car models	368	48	5.16	Painter	painters	1011	823	22.45
CartoonChar	cartoon characters	50	144	36	ProgLanguage	programming languages	101	153	26.73
CellPhoneModel	cell phones	204	49	0	Religion	religions	128	72	11.71
ChemicalElem	chemicals	118	487	1.69	River	river systems	167	118	15.56
City	cities	589	3642	50.08	SearchEngine	search engines	25	133	64
Company	companies	738	7036	26.01	SkyBody	constellations	97	37	1.03
Country	countries	197	677	91.37	Skyscraper	-	172	-	-
Currency	currencies	55	128	25.45	SoccerClub	football clubs	116	101	22.41
DigitalCamera	digital cameras	534	58	0.18	SportEvent	sports events	143	73	12.58
Disease	diseases	209	3566	65.55	Stadium	stadiums	190	92	6.31
Drug	drugs	345	1209	44.05	TerroristGroup	terrorist groups	74	134	33.78
Empire	empires	78	54	6.41	Treaty	treaties	202	200	7.42
Flower	flowers	59	642	25.42	University	universities	501	1127	21.55
Holiday	holidays	82	300	48.78	VideoGame	video games	450	282	17.33
Hurricane	-	74	-	-	Wine	wines	60	270	56.66
Mountain	mountains	245	49	7.75	WorldWarBattle	battles	127	135	9.44
					Total mapped: 37 out of 40 classes		-	-	**26.89**

Table 3: Comparison between manually-assembled instance sets of gold-standard classes (M) and instance sets of automatically-extracted classes (E). Each gold-standard class (M) was manually mapped into an extracted class (E), unless no relevant mapping was found. Ratios ($\frac{M \cap E}{M}$) are shown as percentages

any refinements to hand-built hierarchies, including any future extensions of WordNet.

Accuracy of Class Instances: The computation of the precision of the extracted instances (e.g., *fifth element* and *kill bill* for the class label *movies*) relies on manual inspection of all instances associated to a sample of the extracted class labels. Rather than inspecting a random sample of classes, the evaluation validates the results against a reference set of 40 gold-standard classes that were manually assembled as part of previous work (Paşca, 2007). A class from the gold standard consists of a manually-created class label (e.g., *AircraftModel*) associated with a manually-assembled, and therefore high-precision, set of representative instances of the class.

To evaluate the precision of the extracted instances, the manual label of each gold-standard class (e.g., *SearchEngine*) is mapped into a class label extracted from text (e.g., *search engines*). As shown in the first two columns of Table 3, the mapping into extracted class labels succeeds for 37 of the 40 gold-standard classes. 28 of the 37 mappings involve linking an abstract class label (e.g., *SearchEngine*) with the corresponding plural forms among the extracted class labels (e.g., *search engines*). The remaining 9 mappings link a manual class label with either an equivalent extracted class label (e.g., *SoccerClub* with *football clubs*), or a strongly-related class label (e.g., *NationalPark* with *parks*). No mapping is found for 3 out of the 40 classes, namely *AircraftModel*, *Hurricane* and *Skyscraper*, which are therefore removed from consideration.

The sizes of the instance sets available for each class in the gold standard are compared in the third through fifth columns of Table 3. In the table, M stands for manually-assembled instance sets, and E for automatically-extracted instance sets. For example, the gold-standard class *SearchEngine* contains 25 manually-collected instances, while the parallel class label *search engines* contains 133 automatically-extracted instances. The fifth column shows the percentage of manually-collected instances (M) that are also extracted automatically (E). In the case of the class *SearchEngine*, 16 of the 25 manually-collected instances are among the 133 automatically-extracted instances of the same class,

Label	Value	Examples of Attributes
vital	1.0	*investors*: investment strategies
okay	0.5	*religious leaders*: coat of arms
wrong	0.0	*designers*: stephanie

Table 4: Labels for assessing attribute correctness

which corresponds to a relative coverage of 64% of the manually-collected instance set. Some instances may occur within the manually-collected set but not the automatically-extracted set (e.g., *zoominfo* and *brainbost* for the class *SearchEngine*) or, more frequently, vice-versa (e.g., *surfwax*, *blinkx*, *entireweb*, *web wombat*, *exalead* etc.). Overall, the relative coverage of automatically-extracted instance sets with respect to manually-collected instance sets is 26.89%, as an average over the 37 gold-standard classes. More significantly, the size advantage of automatically-extracted instance sets is not the undesirable result of those sets containing many spurious instances. Indeed, the manual inspection of the automatically-extracted instances sets indicates an average accuracy of 79.3% over the 37 gold-standard classes retained in the experiments. To summarize, the method proposed in this paper acquires open-domain classes from unstructured text of arbitrary quality, without a-priori restrictions to specific domains of interest and with virtually no supervision (except for the ubiquitous Is-A extraction patterns), at accuracy levels of around 90% for class labels and 80% for class instances.

3.3 Evaluation of Class Attributes

Extraction Parameters: Given a target class specified as a set of instances and a set of five seed attributes for a class (e.g., $\{$*quality*, *speed*, *number of users*, *market share*, *reliability*$\}$ for *SearchEngine*), the method described in Section 2.2 extracts ranked lists of class attributes from the input query logs. Internally, the ranking uses Jensen-Shannon (Lee, 1999) to compute similarity scores between internal representations of seed attributes, on one hand, and each of the candidate attributes, on the other hand.

Evaluation Procedure: To remove any possible bias towards higher-ranked attributes during the assessment of class attributes, the ranked lists of attributes to be evaluated are sorted alphabetically into a merged list. Each attribute of the merged list is manually assigned a correctness label within its respective class. An attribute is *vital* if it must be present in an ideal list of attributes of the class; *okay* if it provides useful but non-essential information; and *wrong* if it is incorrect.

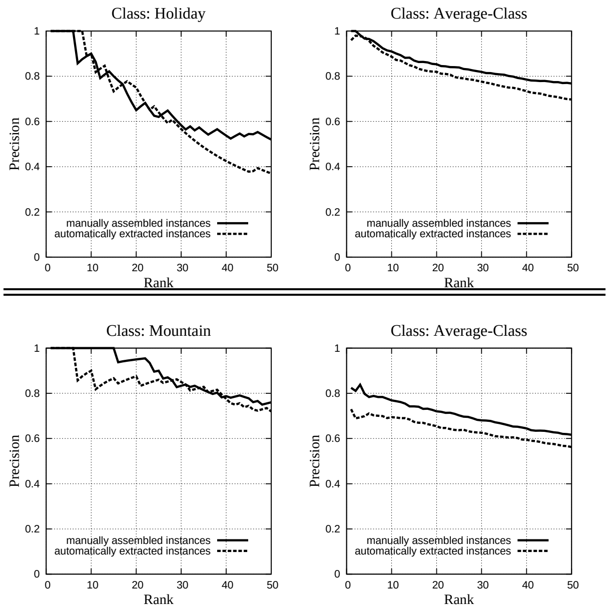


Figure 3: Accuracy of attributes extracted based on manually assembled, gold standard (M) vs. automatically extracted (E) instance sets, for a few target classes (leftmost graphs) and as an average over all (37) target classes (rightmost graphs). Seed attributes are provided as input for each target class (top graphs), or for only one target class (bottom graphs)

To compute the overall precision score over a ranked list of extracted attributes, the correctness labels are converted to numeric values as shown in Table 4. Precision at some rank N in the list is thus measured as the sum of the assigned values of the first N candidate attributes, divided by N.

Accuracy of Class Attributes: Figure 3 plots precision values for ranks 1 through 50 of the lists of attributes extracted through several runs over the 37 gold-standard classes described in the previous section. The runs correspond to different amounts of supervision, specified through a particular choice in the number of seed attributes, and in the source of instances passed as input to the system:

- number of input seed attributes: seed attributes are provided either for each of the 37 classes, for a total of 5×37=185 attributes (the graphs at the top of Figure 3); or only for one class (namely, *Country*),

Class		Precision				Top Ten Extracted Attributes
#	Class Label={Set of Instances}	@5	@10	@15	@20	
1	accounting systems={flexcube, myob, oracle financials, peachtree accounting, sybiz,...}	0.70	0.70	0.77	0.70	overview, architecture, interview questions, free downloads, canadian version, passwords, modules, crystal reports, property management, free trial
2	antimicrobials={azithromycin, chloramphenicol, fusidic acid, quinolones, sulfa drugs,...}	1.00	1.00	0.93	0.95	chemical formula, chemical structure, history, invention, inventor, definition, mechanism of action, side-effects, uses, shelf life
5	civilizations={ancient greece, chaldeans, etruscans, inca indians, roman republic,...}	1.00	1.00	0.93	0.90	social pyramid, climate, geography, flag, population, social structure, natural resources, family life, god, goddesses
9	farm animals={angora goats, burros, cattle, cows, donkeys, draft horses, mule, oxen,...}	1.00	0.80	0.83	0.80	digestive system, evolution, domestication, gestation period, scientific name, adaptations, coloring pages, p**, body parts, selective breeding
10	forages={alsike clover, rye grass, tall fescue, sericea lespedeza,...}	0.90	0.95	0.73	0.57	types, picture, weed control, planting, uses, information, herbicide, germination, care, fertilizer
	Average-Class (25 classes)	**0.75**	**0.70**	**0.68**	**0.67**	

Table 5: Precision of attributes extracted for a sample of 25 classes. Seed attributes are provided for only one class.

for a total of 5 attributes over all classes (the graphs at the bottom of Figure 3);

• source of input instance sets: the instance sets for each class are either manually collected (M from Table 3), or automatically extracted (E from Table 3). The choices correspond to the two curves plotted in each graph in Figure 3.

The graphs in Figure 3 show the precision over individual target classes (leftmost graphs), and as an average over all 37 classes (rightmost graphs). As expected, the precision of the extracted attributes as an average over all classes is best when the input instance sets are hand-picked (M), as opposed to automatically extracted (E). However, the loss of precision from M to E is small at all measured ranks.

Table 5 offers an alternative view on the quality of the attributes extracted for a random sample of 25 classes out of the larger set of 4,583 classes acquired from text. The 25 classes are passed as input for attribute extraction without modifications. In particular, the instance sets are not manually post-filtered or otherwise changed in any way. To keep the time required to judge the correctness of all extracted attributes within reasonable limits, the evaluation considers only the top 20 (rather than 50) attributes extracted per class. As shown in Table 5, the method proposed in this paper acquires attributes for automatically-extracted, open-domain classes, without a-priori restrictions to specific domains of interest and relying on only five seed attributes specified for only one class, at accuracy levels reaching 70% at rank 10, and 67% at rank 20.

4 Related Work

4.1 Acquisition of Classes of Instances

Although some researchers focus on re-organizing or extending classes of instances already available explicitly within manually-built resources such as Wikipedia (Ponzetto and Strube, 2007) or WordNet (Snow et al., 2006) or both (Suchanek et al., 2007), a large body of previous work focuses on compiling sets of instances, not necessarily labeled, from unstructured text. The extraction proceeds either iteratively by starting from a few seed extraction rules (Collins and Singer, 1999), or by mining named entities from comparable news articles (Shinyama and Sekine, 2004) or from multilingual corpora (Klementiev and Roth, 2006).

A bootstrapping method (Riloff and Jones, 1999) cautiously grows very small seed sets of five instances of the same class, to fewer than 300 items after 50 consecutive iterations, with a final precision varying between 46% and 76% depending on the type of semantic lexicon. Experimental results from (Feldman and Rosenfeld, 2006) indicate that named entity recognizers can boost the performance of weakly supervised extraction of class instances, but only for a few coarse-grained types such as *Person* and only if they are simpler to recognize in text (Feldman and Rosenfeld, 2006).

In (Cafarella et al., 2005), handcrafted extraction patterns are applied to a collection of 60 million Web documents to extract instances of the classes *Company* and *Country*. Based on the manual evaluation of samples of extracted instances, an estimated number of 1,116 instances of *Company* are extracted at a precision score of 90%. In comparison, the approach of this paper pursues a more aggressive goal, by extracting a larger and more diverse number of labeled classes, whose instances are often more difficult to extract than country names and most company names, at precision scores of almost 80%.

The task of extracting relevant labels to describe sets of documents, rather than sets of instances, is explored in (Treeratpituk and Callan, 2006). Given pre-existing sets of instances, (Pantel and Ravichandran, 2004) investigates the task of acquiring appropriate class labels to the sets from unstructured text. Various class labels are assigned to a total of 1,432 sets of instances. The accuracy of the class labels is computed over a sample of instances, by manually assessing the correctness of the top five labels returned by the system for each instance. The resulting mean reciprocal rank of 77% gives partial credit to labels of an evaluated instance, even if only the fourth or fifth assigned labels are correct. Our evaluation of the accuracy of class labels is stricter, as it considers only one class label of a given instance at a time, rather than a pool of the best candidate labels.

As a pre-requisite to extracting relations among pairs of classes, the method described in (Davidov et al., 2007) extracts class instances from unstructured Web documents, by submitting pairs of instances as queries and analyzing the contents of the top 1,000 documents returned by a Web search engine. For each target class, a small set of instances must be provided manually as seeds. As such, the method can be applied to the task of extracting a large set of open-domain classes only after manually enumerating through the entire set of target classes, and providing seed instances for each. Furthermore, no attempt is made to extract relevant class labels for the sets of instances. Comparatively, the open-domain classes extracted in our paper have an explicit label in addition to the sets of instances, and do not require identifying the range of the target classes in advance, or providing any seed instances as input. The evaluation methodology is also quite different, as the instance sets acquired based on the input seed instances in (Davidov et al., 2007) are only evaluated for three hand-picked classes, with precision scores of 90% for names of *countries*, 87% for *fish species* and 68% for instances of *constellations*. Our evaluation of the accuracy of class instances is again stricter, since the evaluation sample is larger, and includes more varied classes, whose instances are sometimes more difficult to identify in text.

4.2 Acquisition of Class Attributes

Previous work on the automatic acquisition of attributes for open-domain classes from text is less general than the extraction method and experiments presented in our paper. Indeed, previous evaluations were restricted to small sets of classes (forty classes in (Paşca, 2007)), whereas our evaluations also consider a random, more diverse sample of open-domain classes. More importantly, by dropping the requirement of manually providing a small set of seed attributes for each target class, and relying on only a few seed attributes specified for one reference class, we harvest class attributes without the need of first determining what the classes should be, what instances they should contain, and from which resources the instances should be collected.

5 Conclusion

In a departure from previous approaches to large-scale information extraction from unstructured text on the Web, this paper introduces a weakly-supervised extraction framework for mining useful knowledge from a combination of both documents and search query logs. In evaluations over labeled classes of instances extracted without a-priori restrictions to specific domains of interest and with very little supervision, the accuracy exceeds 90% for class labels, approaches 80% for class instances, and exceeds 70% (at rank 10) and 67% (at rank 20) for class attributes. Current work aims at expanding the number of instances within each class while retaining similar precision levels; extracting attributes with more consistent precision scores across classes from different domains; and introducing confidence scores in attribute extraction, allowing for the detection of classes for which it is unlikely to extract large numbers of useful attributes from text.

References

M. Banko, Michael J Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. 2007. Open information extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 2670–2676, Hyderabad, India.

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

M. Cafarella, D. Downey, S. Soderland, and O. Etzioni. 2005. KnowItNow: Fast, scalable information extraction from the Web. In *Proceedings of the Human Language Technology Conference (HLT-EMNLP-05)*, pages 563–570, Vancouver, Canada.

M. Collins and Y. Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of the 1999 Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*, pages 189–196, College Park, Maryland.

D. Davidov, A. Rappoport, and M. Koppel. 2007. Fully unsupervised discovery of concept-specific relationships by Web mining. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL-07)*, pages 232–239, Prague, Czech Republic.

D. Downey, M. Broadhead, and O. Etzioni. 2007. Locating complex named entities in Web text. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 2733–2739, Hyderabad, India.

R. Feldman and B. Rosenfeld. 2006. Boosting unsupervised relation extraction by using NER. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP-ACL-06)*, pages 473–481, Sydney, Australia.

C. Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database and Some of its Applications*. MIT Press.

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

A. Klementiev and D. Roth. 2006. Weakly supervised named entity transliteration and discovery from multilingual comparable corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL-06)*, pages 817–824, Sydney, Australia.

L. Lee. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics (ACL-99)*, pages 25–32, College Park, Maryland.

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL-98)*, pages 768–774, Montreal, Quebec.

R. Mooney and R. Bunescu. 2005. Mining knowledge from text using information extraction. *SIGKDD Explorations*, 7(1):3–10.

M. Paş ca, D. Lin, J. Bigham, A. Lifchits, and A. Jain. 2006. Organizing and searching the World Wide Web of facts - step one: the one-million fact extraction challenge. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-06)*, pages 1400–1405, Boston, Massachusetts.

M. Paş ca. 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)*, pages 101–110, Banff, Canada.

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

S. Ponzetto and M. Strube. 2007. Deriving a large scale taxonomy from Wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI-07)*, pages 1440–1447, Vancouver, British Columbia.

E. Riloff and R. Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI-99)*, pages 474–479, Orlando, Florida.

Y. Shinyama and S. Sekine. 2004. Named entity discovery using comparable news articles. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING-04)*, pages 848–853, Geneva, Switzerland.

R. Snow, D. Jurafsky, and A. Ng. 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)*, pages 801–808, Sydney, Australia.

F. Suchanek, G. Kasneci, and G. Weikum. 2007. Yago: a core of semantic knowledge unifying WordNet and Wikipedia. In *Proceedings of the 16th World Wide Web Conference (WWW-07)*, pages 697–706, Banff, Canada.

P. Treeratpituk and J. Callan. 2006. Automatically labeling hierarchical clusters. In *Proceedings of the 7th Annual Conference on Digital Government Research (DGO-06)*, pages 167–176, San Diego, California.