Instance-Based Ontology Population Exploiting Named-Entity Substitution

Claudio Giuliano
Fondazione Bruno Kessler
Trento, Italy
giuliano@fbk.eu

Alfio Gliozzo
Laboratory for Applied Ontology
Italian National Research Council
Rome, Italy
alfio.gliozzo@cnr.istc.it

Abstract

We present an approach to ontology population based on a lexical substitution technique. It consists in estimating the plausibility of sentences where the named entity to be classified is substituted with the ones contained in the training data, in our case, a partially populated ontology. Plausibility is estimated by using Web data, while the classification algorithm is instance-based. We evaluated our method on two different ontology population tasks. Experiments show that our solution is effective, outperforming existing methods, and it can be applied to practical ontology population problems.

1 Introduction

Semantic Web and knowledge management applications require to populate the concepts of their domain ontologies with individuals and find their relationships from various data sources, including databases and natural language texts. As the extensional part of an ontology (the ABox) is often manually populated, this activity can be very time-consuming, requiring considerable human effort. The development of automatic techniques for ontology population is then a crucial research area. Natural language processing techniques are natural candidates to solve this problem as most of the data contained in the Web and in the companies' intranets is free text. Information extraction (IE) is commonly employed to (semi-) automate such a task.

Current state-of-the-art IE systems are mostly based on general purpose supervised machine learning techniques (e.g., kernel methods). However, supervised systems achieve acceptable accuracy only if they are supplied with a sufficiently large amount of training data, usually consisting of manually annotated texts. Consequently, they can be only used to populate top-level concepts of ontologies (e.g., people, locations, organizations). In fact, when the number of subclasses increases the number of annotated documents required to find sufficient positive examples for all subclasses becomes too large to be practical. As domain ontologies usually contain hundreds of concepts arranged in deep class/subclass hierarchies, alternative techniques have to be found to recognize fine-grained distinctions (e.g., to categorize people as scientists and scientists as physicists, mathematicians, biologists, etc.).

In this paper, we present an approach to the classification of named entities into fine-grained ontological categories based on a method successfully employed in lexical substitution.[1] In particular, we predict the fine-grained category of a named entity, previously recognized, by simply estimating the plausibility of sentences where the entity to be classified is substituted with the ones contained in the training data, in our case, a partially populated ontology.

In most of the cases, ontologies are partially populated during the development phase and after that the annotation cost is practically negligible, making this method highly attractive in many applicative domains. This allows us to define an instance-based learning approach for fine-grained

[1] Lexical substitution consists in identifying the most likely alternatives (substitutes) of a target word given its context (McCarthy, 2002).

Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 265–272
Manchester, August 2008

entity categorization that exploits the Web to collect evidence of the new entities and does not require any labeled text for supervision, only a partially populated ontology. Therefore, it can be used in different domains and languages to enrich an existing ontology with new entities extracted from texts by a named-entity recognition system and/or databases.

We evaluated our method on the benchmark proposed by Tanev and Magnini (2006) to provide a fair comparison with other approaches, and on a general purpose ontology of people derived from WordNet (Fellbaum, 1998) to perform a more extensive evaluation. Specifically, the experiments were designed to investigate the effectiveness of our approach at different levels of generality and with different amounts of training data. The results show that it significantly outperforms the baseline methods and, where a comparison is possible, other approaches and achieves a good performance with a small number of examples per category. Error analysis shows that most of the misclassification errors are due to the finer-grained distinctions between instances of the same super-class.

2 Lexical Substitutability and Ontology Population

Our approach is based on the assumption that entities that occur in similar contexts belong to the same concept(s). This can be seen as a special case of the distributional hypothesis, that is, terms that occur in the same contexts tend to have similar meanings (Harris, 1954).

If our assumption is correct, then given an instance in different contexts one can substitute it with another of the same ontological type (i.e., of the same category) and probably generate true statements. In fact, most of the predicates that can be asserted for an instance of a particular category can also be asserted for other instances of the same category. For instance, the sentence "Ayrton Senna is a F1 Legend" preserves its truthfulness when Ayrton Senna is replaced with Michael Schumacher, while it is false when Ayrton Senna is replaced with the MotoGP champion Valentino Rossi.

For our purposes, the Web provides a simple and effective solution to the problem of determining whether a statement is true or false. Due to the high redundancy of the Web, the high frequency of a statement generated by a substitution usually provides sufficient evidence for its truth, allowing us to easily implement an automatic method for fine-grained entity classification. Following this intuition, we developed an ontology population technique adopting pre-classified entities as training data (i.e., a partially populated ontology) to classify new ones.

When a new instance has to be classified, we first collect snippets containing it from the Web. Then, for each snippet, we substitute the new instance with each of the training instances. The snippets play a crucial role in our approach because we expect that they provide the features that characterize the category to which the entity belongs. Thus, it is important to collect a sufficiently large number of snippets to capture the features that allow a fine-grained classification.

To estimate the correctness of each substitution, we calculate a plausibility score using a modified version of the lexical substitution algorithm introduced in Giuliano et al. (2007), that assigns higher scores to the substitutions that generate highly frequent sentences on the Web. In particular, this technique ranks a given list of synonyms according to a similarity metric based on the occurrences in the Web 1T 5-gram corpus,[2] which specify n-grams frequencies in a large Web sample. This technique achieved the state-of-the-art performance on the English Lexical Substitution task at SemEval 2007 (McCarthy and Navigli, 2007).

Finally, on the basis of these plausibility scores, the algorithm assigns the new instance to the category whose individuals show a closer linguistic behavior (i.e., they can be substituted generating plausible statements).

3 The IBOP algorithm

In this section, we describe the algorithmic and mathematical details of our approach. The instance-based ontology population (IBOP) algorithm is an instance-based supervised machine learning approach.[3] The proposed algorithm is summarized as follows:

Step 1 For each candidate instance i, we collect the first N snippets containing i from the Web. For instance, 3 snippets for the candidate instance Ayrton Senna are "The death of Ayrton Senna at

[2] `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13`.

[3] An analogy between instance-based learning methods and our approach is left to future work.

the 1994 San Marino GP", "triple world champion Ayrton Senna", and "about F1 legend Ayrton Senna".

Step 2 Then, for each retrieved snippet q_k ($1 \leqslant k \leqslant N$), we derive a list of hypothesis phrases by replacing i with each training instance j from the given ontology. For instance, from the snippet "about F1 legend Ayrton Senna", we derive "about F1 legend Michael Schumacher" and "about F1 legend Valentino Rossi", assuming to have the former classified as F1 driver and the latter as MotoGP driver.

Step 3 For each hypothesis phrase h_j, we calculate the plausibility score s_j using a variant of the scoring procedure defined in Giuliano et al. (2007). In our case, s_j is given by the sum of the pointwise mutual information (PMI) of all the n-grams ($1 < n \leqslant 5$) that contain j divided by the self-information of the right and left contexts.[4] Dividing by the self-information allows us to penalize the hypotheses that have contexts with a low information content, such as sequences of stop words. The frequency of the n-grams is estimated from the Web 1T 5-gram corpus. For instance, from the hypothesis phrase "about F1 legend Michael Schumacher", we generate and score the following n-grams: "legend Michael Schumacher", "F1 legend Michael Schumacher", and "about F1 legend Michael Schumacher".

Step 4 To obtain an overall score s_c for the category c, we sum the scores obtained from each training instance of category c for all snippets, as defined in Equation 1.

$$s_c = \sum_{k=1}^{N} \sum_{l=1}^{M} s_l, \tag{1}$$

where M is the number of training instances for the category c.[5]

Step 5 Finally, the instance i is categorized with that concept having the maximum score:

$$c^* = \begin{cases} \underset{c}{\text{argmax}}\ s_c & \text{if } s_c \geqslant \theta; \\ \emptyset & \text{otherwise.} \end{cases} \tag{2}$$

[4]The pointwise mutual information is defined as the log of the deviation between the observed frequency of a n-gram and the probability of that n-gram if it were independent and the self-information is a measure of the information content of a n-gram ($-\log p$, where p is the probability of the n-gram).

[5]Experiments using the sum of average or argmax score yield worst results.

Where a higher value of the parameter θ increases precision but degrades recall.

4 Benchmarks

For evaluating the proposed algorithm and comparing it with other algorithms, we adopted the two benchmarks described below.

4.1 Tanev and Magnini Benchmark

Tanev and Magnini (2006) proposed a benchmark ontology that consists of two high-level named entity categories (i.e., person and location) both having five fine-grained subclasses (i.e., mountain, lake, river, city, and country as subtypes of location; statesman, writer, athlete, actor, and inventor are subtypes of person). WordNet and Wikipedia were used as primary data sources for populating the evaluation ontology. In total, the ontology is populated with 280 instances which were not ambiguous (with respect to the ontology). We extracted the training set from WordNet, collecting 20 examples per sub-category, of course, not already contained in the test set.

4.2 People Ontology

The benchmark described in the previous section is clearly a toy problem, and it does not allow us to evaluate the effectiveness of our method, in particular the ability to perform fine-grained classifications. To address this problem, we developed a larger ontology of people (called People Ontology), characterized by a complex taxonomy having multiple layers and containing thousands of instances. This ontology has been extracted from WordNet, that we adapted to our purpose after a re-engineering phase. In fact, we need a formal specification of the conceptualizations that are expressed by means of WordNet's synsets, and, in particular, we need a clear distinction between individuals and categories, as well as a robust categorization mechanism to assign individuals to general concepts.

This result can be achieved by following the directives defined by Gangemi et al. (2003) for OntoWordNet, in which the informal WordNet semantics is re-engineered in terms of a description logic. We follow an analogous approach. Firstly, any possible instance in WordNet 1.7.1 has been identified by looking for all those synsets containing at least one word starting with a capital letter. The result is a set of instances I. All the remaining

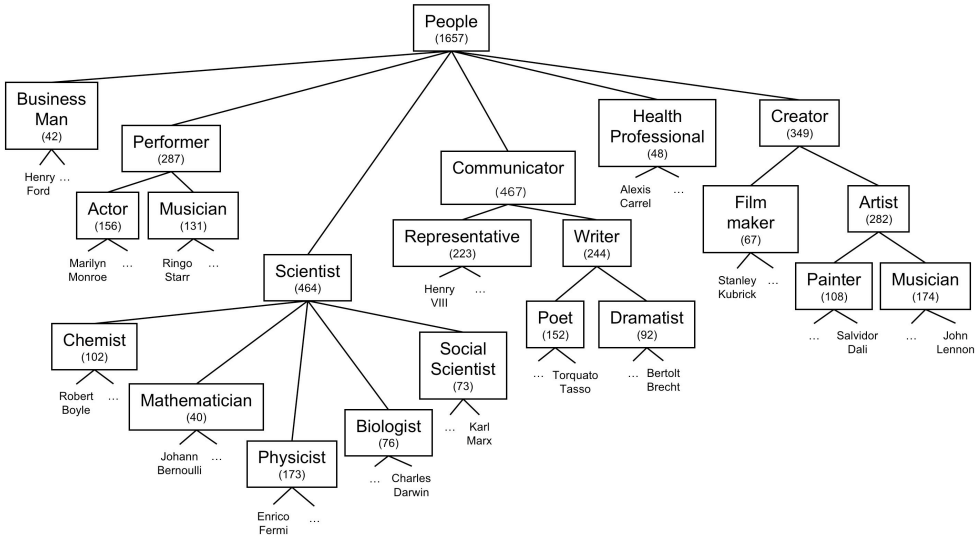

Figure 1: The taxonomy of the People Ontology extracted from WordNet 1.7.1. Numbers in brackets are the total numbers of individuals per category. Concepts that have less than 40 instances were removed.

synsets are then regarded as concepts, collected in the set C. Then, is_a relations between synsets are converted into one of the following standard OWL-DL constructs:

X subclass_of Y if X is_a Y and $X \in C$ and $Y \in C$

X instance_of Y if X is_a Y and $X \in I$ and $Y \in C$

The formal semantics of both $subclass\_of$ and $instance\_of$ is formally defined in OWL-DL. $subclass\_of$ is a transitive relation (i.e., $X subclass\_of Y$ and $Y subclass\_of Z$ implies $X subclass\_of Z$) and the $instance\_of$ relation has the following property: $X instance\_of Y$ and $Y subclass\_of Z$ implies $X instance\_of Z$.

To define the People Ontology, we selected the sub-hierarchy of WordNet representing people, identifying the corresponding top-level synset $X = \{$*person, individual, someone, somebody, mortal, soul*$\}$, and collecting all the classes Y such that Y is a subclass of X and all the instances I such that I is an instance of Y. We discovered that many concepts in the derived hierarchy were empty or scarcely populated. As we need a sufficient amount data to obtain statistically significant results, we eliminated the classes that contain less than 40 instances from the ontology. The derived ontology contains 1627 instances structured in 21 sub-categories (Figure 1). Finally, we randomly split its individuals into two equally sized subsets. The results reported in the following section were evaluated using two-fold cross-validation on these two subsets.

5 Evaluation

In this section, we present the performance of the IBOP algorithm on the evaluation benchmarks described in the previous section.

5.1 Experimental Setting

For each individual, we collected 100 entity mentions in their context by querying Google™. As most of them are names of celebrities, the Web provided sufficient data.[6]

We approached the population task as a standard categorization problem, trying to assign new instances to the most specific category. We measured standard precision/recall figures. In addition, we evaluated the classifier accuracy at the most abstract level, by inheriting the predictions from sub-concepts to super-concepts. For example, when an instance is assigned to a specific category (e.g., Musician), it is also (implicitly) assigned to all its super-classes (e.g., Artist and Creator). This operation is performed according to the extensional semantics of the description logic, as described in the previous section. Following this approach, we are able to evaluate the effectiveness of our algorithm at any level of generality. The micro- and macro-averaged F_1 have been evaluated by taking into account both specific and generic classes at the same time. In this way, we tend to penalize the

[6] A study of how the number of snippets N would impact the performance of the IBOP algorithm has been deferred to future work.

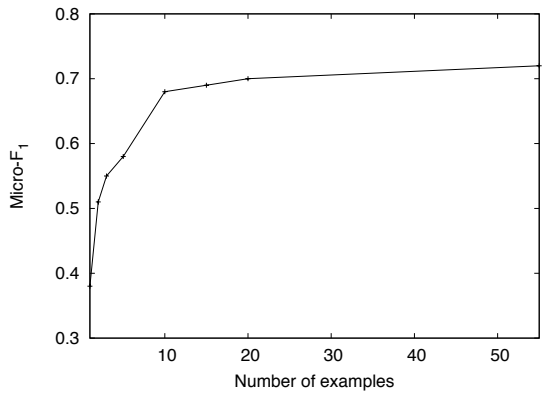

Figure 2: Learning curve on the People Ontology.

gross misclassification errors (e.g., Biologist vs. Poet), while minor errors (e.g., Poet vs. Dramatist) are less relevant. This approach is similar to the one proposed by Melamed and Resnik (2000) for a similar hierarchical categorization task.

5.2 Accuracy

Table 1 shows micro- and macro-averaged results of the proposed method obtained on the Tanev and Magnini (2006) benchmark and compares them with the class-example (Tanev and Magnini, 2006), IBLE (Giuliano and Gliozzo, 2007), and class-word (Cimiano and Völker, 2005) methods, respectively. Table 2 shows micro- and macro-averaged results of the proposed method obtained on the People Ontology and compares them with the random and most frequent baseline methods.[7] In both experiments, the IBOP algorithm was trained on 20 examples per category and setting the parameter $\theta = 0$ in Equation 2.

For the People Ontology, we performed a disaggregated evaluation, whose results are shown in Table 3, while Figure 2 shows the learning curve. The experiment was conducted setting the parameter $\theta = 0$.

System	Micro-F_1	Macro-F_1
IBOP	73	71
Class-Example	68	62
IBLE	57	47
Class-Word	42	33

Table 1: Comparison of different ontology population techniques on the Tanev and Magnini (2006) benchmark.

[7]The most frequent category has been estimated on the training data.

System	Micro-F_1	Macro-F_1
IBOP	70.1	62.3
Random	15.4	15.5
Most Frequent	20.7	3.3

Table 2: Comparison between the IBOP algorithm and the baseline methods on the People Ontology.

Class	Prec	Recall	F_1
Scientist	84.4	73.3	78.4
Physicist	63.0	39.3	48.4
Mathematician	25.0	67.5	36.5
Chemist	44.2	52.0	47.7
Biologist	62.5	13.2	21.7
Social scientist	43.1	30.1	35.5
Performer	76.5	66.9	71.4
Actor	67.5	67.9	67.7
Musician	68.1	48.9	56.9
Creator	70.6	84.5	76.9
Film Maker	52.9	68.7	59.7
Artist	72.8	85.5	78.6
Painter	74.4	86.1	79.8
Musician	68.9	81.6	74.7
Comunicator	76.4	83.1	79.6
Writer	78.6	76.6	77.6
Poet	67.4	61.2	64.1
Dramatist	65.0	70.7	67.7
Representative	84.8	76.7	80.6
Business man	47.2	40.5	43.6
Health professional	29.3	25.0	27.0
micro	69.6	70.7	70.1
macro	62.3	70.7	62.3

Table 3: Results for each category of the People Ontology.

5.3 Confusion Matrix

Table 4 shows the confusion matrix for the People Ontology task, in which the rows are ground truth classes and the columns are predictions. The experiment was conducted using 20 training examples per category and setting the parameter $\theta = 0$. The matrix has been calculated for the finer-grained categories and, then, grouped according to their top-level concepts.

5.4 Precision/Recall Tradeoff

Figure 3 shows the precision/recall curve for the People Ontology task obtained varying the parameter θ in Equation 2. The experiment was conducted using 20 training examples per category.

5.5 Discussion

The results obtained are undoubtedly satisfactory. Table 1 shows that our approach outperforms the other three methods on the Tanev and Magnini (2006) benchmark. Note that the Class-Example approach has been trained on 1194 named enti-

	Scientist					Performer		Creator			Communicator			Business	Health
	Phy	Mat	Che	Bio	Soc	Act	Mus	Fil	Pai	Mus	Poe	Dra	Rep	man	prof
Phy	68	40	25	3	11	2	0	0	3	1	7	1	7	2	3
Mat	3	27	1	0	0	0	0	1	0	0	4	0	2	1	1
Che	12	10	53	2	7	3	1	2	2	0	1	0	4	4	1
Bio	4	12	13	10	3	3	0	1	5	2	4	1	11	2	5
Soc	6	3	4	1	22	4	0	2	2	3	4	1	12	0	9
Act	3	1	2	0	0	106	6	20	0	3	2	4	7	1	1
Mus	1	1	2	0	0	16	64	5	2	28	2	2	7	0	1
Fil	0	0	0	0	0	7	0	46	0	4	1	1	4	3	1
Pai	2	1	0	0	1	1	1	2	93	3	1	0	2	1	0
Mus	1	0	0	0	0	1	16	2	3	142	1	3	2	1	2
Poe	1	2	1	0	1	2	3	3	6	12	93	20	6	1	1
Dra	0	2	1	0	0	3	0	2	2	3	9	65	1	2	2
Rep	0	6	7	0	3	6	1	0	3	2	5	0	189	1	0
Bus	3	3	6	0	0	0	1	1	0	1	2	0	6	17	2
Hea	4	0	5	0	3	3	1	0	4	2	2	2	10	0	12

Table 4: Confusion matrix for the finer-grained categories grouped according to their top-level concepts of the People Ontology.

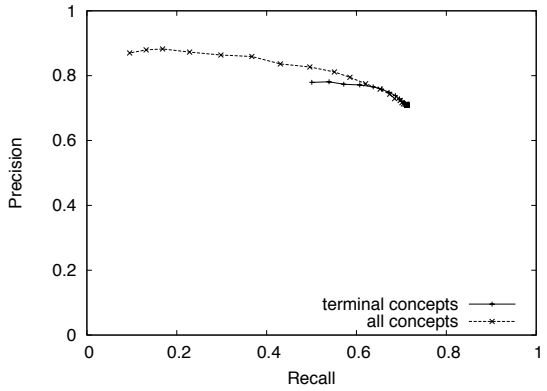

Figure 3: Precision/recall curve on the People Ontology.

ties, almost 60 examples per category, whereas we achieved the same result with around 10 examples per category. On the other hand, as Table 2 shows, the IBOP algorithm is effective in populating a more complex ontology and significantly outperforms the random and most frequent baselines.

An important characteristic of the algorithm is the small number of examples required per category. This affects both the prediction accuracy and the computation time (this is generally a common property of instance-based algorithms). Therefore, finding an optimal tradeoff between the training size and the performance is crucial. The learning curve (Figure 2) shows that the algorithm outperforms the baselines with only 1 example per category and achieves a good accuracy ($F_1 \approx 67\%$) with only 10 examples per category, while it reaches a plateau around 20 examples ($F_1 \approx 70\%$), but leaving a little room for improvement.

Table 4 shows that misclassification errors are largely distributed among categories belonging to the same super-class (i.e., the blocks on the main diagonal are more densely populated than others). As expected, the algorithm is much more accurate for the top-level concepts (i.e., Scientist, Communicator, etc.), where the category distinctions are clearer, while a further fine-grained classification, in some cases, is even difficult for human annotators. In particular, results are higher for fine-grained categories densely populated and with a small number of sibling categories (i.e., Painter and Musician). We have observed that the results on sparse categories can be made more precise by increasing the training size, generally at the expense of a lower recall.

We tried to maximize precision by varying the parameter θ in Equation 2, that is, avoiding all assignments where the plausibility score is lower than a given threshold. Figure 3 shows that the precision can be significantly enhanced ($\sim 90\%$) at the expense of poor recall ($\sim 20\%$), while the algorithm achieves 80% precision at around 50% recall.

Finally, we performed some preliminary error analysis, investigating the misclassifications in the categories Scientists and Musicians. Several errors are due to lack of information in WordNet, For example, Leonhard Euler was a mathematician and physicist, however, in WordNet, he is classified as physicist, and our system classifies him as mathematician. On the other hand, for simplicity, the algorithm returns a single category per instance,

however, the test set contains many entities that are classified in more than one category. For instance, Bertolt Brecht is both poet and dramatist and the system classified him as dramatist. Another interesting case is the presence of two categories Musician, one is subclass of Performer and the other of Artist, in which, for instance, Ringo Starr is a performer while John Lennon is an artist, while the system classified both as performers.

6 Related work

Brin (1998) defined a methodology to extract information from the Web starting from a small set of seed examples, then alternately learning extraction patterns from seeds, and further seeds from patterns. Despite the fact that the evaluation was on relation extraction the method is general and might be applied to entity extraction and categorization. The approach was further extended by Agichtein and Gravano (2000). Our approach differs from theirs in that we do not learn patterns. Thus, we do not require ad hoc strategies for generating patterns and estimating their reliability, a crucial issue in these approaches as "bad" patterns may extract wrong seeds instances that in turn may generate even more inaccurate patterns in the following iteration.

Fleischman and Hovy (2002) approached the ontology population problem as a supervised classification task. They compare different machine learning algorithms, providing instances in their context as training examples as well as more global semantic information derived from topic signature and WordNet.

Alfonseca and Manandhar (2002) and Cimiano and Völker (2005) present similar approaches relying on the Harris' distributional hypothesis and the vector-space model. They assign a particular instance represented by a certain context vector to the concept corresponding to the most similar vector. Contexts are represented using lexical-syntactic features.

KnowItAll (Etzioni et al., 2005) uses a search engine and semantic patterns (similar to those defined by Hearst (1992)) to classify named entities on the Web. The approach uses simple techniques from the ontology learning field to perform extraction and then annotation. It also is able to perform very simple pattern induction, consisting of looking at n words before and n words after the occurrence of an example in the document. With pattern learning, KnowItAll becomes a bootstrapped learning system, where rules are used to learn new seeds, which in turn are used to learn new rules. A similar approach is used in C-PANKOW (Cimiano et al., 2005). Compared to KnowItAll and C-PANKOW, our approach does not need handcrafted patterns as input. They are implicitly found by substituting the training instances in the contexts of the input entities. Another key difference is that concepts in the ontology do not need to be lexicalized.

Tanev and Magnini (2006) proposed a weakly-supervised method that requires as training data a list of terms without context for each category under consideration. Given a generic syntactically parsed corpus containing at least each training entity twice, the algorithm learns, for each category, a feature vector describing the contexts where those entities occur. Then, it compares the new (unknown) entity with the so obtained feature vectors, assigning it to the most similar category. Even though we used a significantly smaller number of training instances, we obtained better results on their benchmark.

More recently, Giuliano and Gliozzo (2007) proposed an unsupervised approach based on lexical entailment, consisting in assigning an entity to the category whose lexicalization can be replaced with its occurrences in a corpus preserving the meaning. A disadvantage is that the concepts in the ontology have to be lexicalized, as they are used as training examples. Our approach is based on a similar idea, but with the main difference that an instance is substituted with other instances rather than with their category names. Considering that, in most of the cases, ontologies are partially populated during the development phase, and hence the annotation cost is marginal, our approach is a realistic alternative for practical ontology population problems.

7 Conclusions and Future Work

We have described an instance-based algorithm for automatic fine-grained categorization of named entities, previously identified by an entity recognition system or already present in a database. This method is meant to provide an effective solution to the ontology population problem. It exploits the Web or a domain corpus to collect evidence of the new instances and does not require labeled texts for supervision, but a partially populated ontology.

The experimental results show that, where a comparison is possible, our method outperforms previous methods and it can be applied to different domains and languages to (semi-) automatically enrich an existing ontology.

Future work will address the definition of a hierarchical categorization strategy where instances are classified in a top-down manner, in order to efficiently populate very large ontologies, since we plan to apply this method to extract structured information from Wikipedia. Furthermore, we will investigate how co-reference resolution might well benefit from our ontology classification. Finally, we plan to exploit the IBOP algorithm for ontology mapping and multilingual alignment of lexical resources.

Acknowledgments

Claudio Giuliano is supported by the X-Media project (`http://www.x-media-project.org`), sponsored by the European Commission as part of the Information Society Technologies (IST) program under EC grant number IST-FP6-026978.

References

Agichtein, Eugene and Luis Gravano. 2000. Snowball: extracting relations from large plain-text collections. In *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94, New York, NY, USA. ACM.

Alfonseca, Enrique and Suresh Manandhar. 2002. Extending a lexical ontology by a combination of distributional semantics signatures. In *EKAW '02: Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, pages 1–7, London, UK. Springer-Verlag.

Brin, Sergey. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98*.

Cimiano, Philipp and Johanna Völker. 2005. Towards large-scale, open-domain and ontology-based named entity classification. In *Proceedings of RANLP'05*, pages 66– 166–172, Borovets, Bulgaria.

Cimiano, Philipp, Günter Ladwig, and Steffen Staab. 2005. Gimme the context: Context-driven automatic semantic annotation with C-PANKOW. In Ellis, Allan and Tatsuya Hagino, editors, *Proceedings of the 14th World Wide Web Conference*, pages 332 – 341, Chiba, Japan, MAY. ACM Press.

Etzioni, Oren, Michael Cafarella, Doug Downey, Ana M. Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):191–134.

Fellbaum, Christiane. 1998. *WordNet. An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Fleischman, Michael and Eduard Hovy. 2002. Fine grained classification of named entities. In *Proceedings of the 19th International Conference on Computational Linguistics*, Taipei, Taiwan.

Gangemi, Aldo, Roberto Navigli, and Paola Velardi. 2003. Axiomatizing WordNet glosses in the OntoWordNet project. In *Proocedings of the Workshop on Human Language Technology for the Semantic Web and Web Services at ISWC 2003*, Sanibel Island, Florida.

Giuliano, Claudio and Alfio Gliozzo. 2007. Instance based lexical entailment for ontology population. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 248–256.

Giuliano, Claudio, Alfio Gliozzo, and Carlo Strapparava. 2007. Fbk-irst: Lexical substitution task exploiting domain and syntagmatic coherence. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 145–148, Prague, Czech Republic, June.

Harris, Zellig. 1954. Distributional structure. *WORD*, 10:146–162.

Hearst, Marti A. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes, France, July.

McCarthy, Diana and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 48–53, Prague, Czech Republic, June.

McCarthy, Diana. 2002. Lexical substitution as a task for WSD evaluation. In *Proceedings of the ACL-02 workshop on Word Sense Disambiguation*, pages 109–115, Morristown, NJ, USA.

Melamed, I. Dan and Philip Resnik. 2000. Tagger evaluation given hierarchical tag sets. *Computers and the Humanities*, pages 79–84.

Tanev, Hristo and Bernardo Magnini. 2006. Weakly supervised approaches for ontology population. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics (EACL-2006)*, Trento, Italy.