Extracting Definitions and Hypernym Relations relying on Syntactic Dependencies and Support Vector Machines

Guido Boella
University of Turin
Department of Computer Science
boella@di.unito.it

Luigi Di Caro
University of Turin
Department of Computer Science
dicaro@di.unito.it

Abstract

In this paper we present a technique to reveal definitions and hypernym relations from text. Instead of using pattern matching methods that rely on lexico-syntactic patterns, we propose a technique which only uses syntactic dependencies between terms extracted with a syntactic parser. The assumption is that syntactic information are more robust than patterns when coping with length and complexity of the sentences. Afterwards, we transform such syntactic contexts in abstract representations, that are then fed into a Support Vector Machine classifier. The results on an annotated dataset of definitional sentences demonstrate the validity of our approach overtaking current state-of-the-art techniques.

1 Introduction

Nowadays, there is a huge amount of textual data coming from different sources of information. Wikipedia[1], for example, is a free encyclopedia that currently contains 4,208,409 English articles[2]. Even Social Networks play a role in the construction of data that can be useful for Information Extraction tasks like Sentiment Analysis, Question Answering, and so forth.

From another point of view, there is the need of having more structured data in the forms of ontologies, in order to allow semantics-based retrieval and reasoning. Ontology Learning is a task that permits to automatically (or semi-automatically) extract structured knowledge from plain text. Manual construction of ontologies usually requires strong efforts from domain experts, and it thus needs an automatization in such sense.

[1] http://www.wikipedia.org/
[2] April 12, 2013.

In this paper, we focus on the extraction of hypernym relations. The first step of such task relies on the identification of what (Navigli and Velardi, 2010) called *definitional sentences*, i.e., sentences that contain at least one hypernym relation. This subtask is important by itself for many tasks like Question Answering (Cui et al., 2007), construction of glossaries (Klavans and Muresan, 2001), extraction of taxonomic and non-taxonomic relations (Navigli, 2009; Snow et al., 2004), enrichment of concepts (Gangemi et al., 2003; Cataldi et al., 2009), and so forth.

Hypernym relation extraction involves two aspects: linguistic knowlege, and model learning. Patterns collapse both of them, preventing to face them separately with the most suitable techniques. First, patterns have limited expressivity; then, linguistic knowledge inside patterns is learned from small corpora, so it is likely to have low coverage. Classification strictly depends on the learned patterns, so performance decreases, and the available classification techniques are restricted to those compatible with the pattern approach. Instead, we use a syntactic parser for the first aspect (with all its native and domain-independent knowledge on language expressivity), and a state-of-the-art approach to learn models with the use of Support Vector Machine classifiers.

Our assumption is that syntax is less dependent than learned patterns from the length and the complexity of textual expressions. In some way, patterns grasp syntactic relationships, but they actually do not use them as input knowledge.

2 Related Work

In this section we present the current state of the art concerning the automatic extraction of definitions and hypernym relations from plain text. We will use the term *definitional sentence* referring to the more general meaning given by (Navigli and Velardi, 2010): *A sentence that provides a for-*

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 532–537,
Sofia, Bulgaria, August 4-9 2013.

mal explanation for the term of interest, and more specifically as a sentence containing at least one hypernym relation.

So far, most of the proposed techniques rely on lexico-syntactic patterns, either manually or semi-automatically produced (Hovy et al., 2003; Zhang and Jiang, 2009; Westerhout, 2009). Such patterns are sequences of words like "*is a*" or "*refers to*", rather than more complex sequences including part-of-speech tags.

In the work of (Westerhout, 2009), after a manual identification of types of definitions and related patterns contained in a corpus, he successively applied Machine Learning techniques on syntactic and location features to improve the results.

A fully-automatic approach has been proposed by (Borg et al., 2009), where the authors applied genetic algorithms to the extraction of English definitions containing the keyword "*is*". In detail, they assign weights to a set of features for the classification of definitional sentences, reaching a precision of 62% and a recall of 52%.

Then, (Cui et al., 2007) proposed an approach based on *soft patterns*, i.e., probabilistic lexico-semantic patterns that are able to generalize over rigid patterns enabling partial matching by calculating a generative degree-of-match probability between a test instance and the set of training instances.

Similarly to our approach, (Fahmi and Bouma, 2006) used three different Machine Learning algorithms to distinguish actual definitions from other sentences also relying on syntactic features, reaching high accuracy levels.

The work of (Klavans and Muresan, 2001) relies on a rule-based system that makes use of "cue phrases" and structural indicators that frequently introduce definitions, reaching 87% of precision and 75% of recall on a small and domain-specific corpus.

As for the task of definition extraction, most of the existing approaches use symbolic methods that are based on lexico-syntactic patterns, which are manually crafted or deduced automatically. The seminal work of (Hearst, 1992) represents the main approach based on fixed patterns like "NP_x *is a/an* NP_y" and "NP_x *such as* NP_y", that usually imply $< x$ IS-A $y >$.

The main drawback of such technique is that it does not face the high variability of how a relation can be expressed in natural language. Still, it generally extracts single-word terms rather than well-formed and compound concepts. (Berland and Charniak, 1999) proposed similar lexico-syntactic patterns to extract *part-whole* relationships.

(Del Gaudio and Branco, 2007) proposed a rule-based approach to the extraction of hypernyms that, however, leads to very low accuracy values in terms of Precision.

(Ponzetto and Strube, 2007) proposed a technique to extract hypernym relations from Wikipedia by means of methods based on the connectivity of the network and classical lexico-syntactic patterns. (Yamada et al., 2009) extended their work by combining extracted Wikipedia entries with new terms contained in additional web documents, using a distributional similarity-based approach.

Finally, pure statistical approaches present techniques for the extraction of hierarchies of terms based on words frequency as well as co-occurrence values, relying on clustering procedures (Candan et al., 2008; Fortuna et al., 2006; Yang and Callan, 2008). The central hypothesis is that similar words tend to occur together in similar contexts (Harris, 1954). Despite this, they are defined by (Biemann, 2005) as *prototype-based ontologies* rather than formal terminological ontologies, and they usually suffer from the problem of data sparsity in case of small corpora.

3 Approach

In this section we present our approach to identify hypernym relations within plain text. Our methodology consists in relaxing the problem into two easier subtasks. Given a relation $rel(x, y)$ contained in a sentence, the task becomes to find 1) a possible x, and 2) a possible y. In case of more than one possible x or y, a further step is needed to associate the correct x to the right y.

By seeing the problem as two different classification problems, there is no need to create abstract patterns between the target terms. In addition to this, the general problem of identifying definitional sentences can be seen as to find at least one x and one y in a sentence.

3.1 Local Syntactic Information

Dependency parsing is a procedure that extracts syntactic dependencies among the terms contained in a sentence. The idea is that, given a hypernym relation, hyponyms and hypernyms may be

characterized by specific sets of syntactic contexts. According to this assumption, the task can be seen as a classification problem where each term in a sentence has to be classified as hyponym, hypernym, or neither of the two.

For each noun, we construct a textual representation containing its syntactic dependencies (i.e., its syntactic context). In particular, for each syntactic dependency $dep(a, b)$ (or $dep(b, a)$) of a target noun a, we create an abstract token[3] *dep-target-*$\hat{b}$ (or *dep-*$\hat{b}$*-target*), where $\hat{b}$ becomes the generic string "*noun*" in case it is another noun; otherwise it is equal to b. This way, the nouns are transformed into abstract strings; on the contrary, no abstraction is done for verbs.

For instance, let us consider the sentence "*The Albedo of an object is the extent to which it diffusely reflects light from the sun*". After the Part-Of-Speech annotation, the parser will extract a series of syntactic dependencies like "*det(Albedo, The)*", "*nsubj(extent, Albedo)*", "*prepof(Albedo, object)*", where *det* identifies a determiner, *nsubj* represents a noun phrase which is the syntactic subject of a clause, and so forth[4]. Then, such dependencies will be transformed in abstract terms like "*det-target-the*", "*nsubj-noun-target*", and "*prepof-target-noun*". These triples represent the feature space on which the Support Vector Machine classifiers will construct the models.

3.2 Learning phase

Our model assumes a transformation of the local syntactic information into labelled numeric vectors. More in detail, given a sentence S annotated with the terms linked by the hypernym relation, the system produces as many input instances as the number of nouns contained in S. For each noun n in S, the method produces two instances S^n_x and S^n_y, associated to the label *positive* or *negative* depending on their presence in the target relation (i.e., as x or y respectively). If a noun is not involved in a hypernym relation, both the two instances will have the label *negative*. At the end of this process, two training sets are built, i.e., one for each relation argument, namely the x-set and the y-set. All the instances of both the datasets are then transformed into numeric vectors according to the Vector Space Model (Salton et al., 1975), and are finally fed into a Support Vector Machine classifier[5] (Cortes and Vapnik, 1995). We refer to the two resulting models as the x-model and the y-model. These models are binary classifiers that, given the local syntactic information of a noun, estimate if it can be respectively an x or a y in a hypernym relation.

Once the x-model and the y-model are built, we can both classify definitional sentences and extract hypernym relations. In the next section we deepen our proposed strategy in that sense.

The whole set of instances of all the sentences are fed into two Support Vector Machine classifiers, one for each target label (i.e., x and y).

At this point, it is possible to classify each term as possible x or y by querying the respective classifiers with its local syntactic information.

4 Setting of the Tasks

In this section we present how our proposed technique is able to classify definitional sentences unraveling hypernym relations.

4.1 Classification of definitional sentences

As already mentioned in previous sections, we label as *definitional* all the sentences that contain at least one noun n classified as x, and one noun m classified as y (where $n \neq m$). In this phase, it is not further treated the case of having more than one x or y in one single sentence. Thus, given an input sentence:

1. we extract all the nouns (POS-tagging),
2. we extract all the syntactic dependencies of the nouns (dependency parsing),
3. we fed each noun (i.e., its instance) to the x-model and to the y model,
4. we check if there exist at least one noun classified as x and one noun classified as y: in this case, we classify the sentences as *definitional*.

4.2 Extraction of hypernym relations

Our method for extracting hypernym relations makes use of both the x-model and the y-model as for the the task of classifying definitional sentences. If exactly one x and one y are identified

[3] We make use of the term "abstract" to indicate that some words are replaced with more general entity identifiers.

[4] A complete overview of the Stanford dependencies is available at http://nlp.stanford.edu/software/dependencies_manual.pdf.

[5] We used the Sequential Minimal Optimization implementation of the Weka framework (Hall et al., 2009).

in the same sentence, they are directly connected and the relation is extracted. The only constraint is that x and y must be connected within the same parse tree.

Now, considering our target relation $hyp(x, y)$, in case the sentence contains more than one noun that is classified as x (or y), there are two possible scenarios:

1. there are actually more than one x (or y), or
2. the classifiers returned some false positive.

Up to now, we decided to keep all the possible combinations, without further filtering operations[6]. Finally, in case of multiple classifications of both x and y, i.e., if there are multiple x and multiple y at the same time, the problem becomes to select which x is linked to which y[7]. To do this, we simply calculate the distance between these terms in the parse tree (the closer the terms, the better the connection between the two). Nevertheless, in the used corpus, only around 1.4% of the sentences are classified with multiple x and y.

Finally, since our method is able to extract single nouns that can be involved in a hypernym relation, we included modifiers preceded by preposition "of", while the other modifiers are removed. For example, considering the sentence "An Archipelago is a chain of islands", the whole chunk "chain of islands" is extracted from the single triggered noun chain.

5 Evaluation

In this section we present the evaluation of our approach, that we carried out on an annotated dataset of definitional sentences (Navigli et al., 2010). The corpus contains 4,619 sentences extracted from Wikipedia, and only 1,908 are annotated as *definitional*. On a first instance, we test the classifiers on the extraction of hyponyms (x) and hypernyms (y) from the definitional sentences, independently. Then, we evaluate the classification of definitional sentences. Finally, we evaluate the ability of our technique when extracting whole hypernym relations. With the used dataset, the constructed training sets for the two classifiers (x-set and y-set) resulted to have approximately 1,500 features.

[6]We only used the constraint that x has to be different from y.

[7]Notice that this is different from the case in which a single noun is labeled as both x and y.

Alg.	P	R	F	Acc
WCL-3	98.8%	60.7%	75.2 %	83.4 %
Star P.	86.7%	66.1%	75.0 %	81.8 %
Bigrams	66.7%	**82.7%**	73.8 %	75.8 %
Our sys.	88.0%	76.0%	**81.6%**	**89.6%**

Table 1: Evaluation results for the classification of definitional sentences, in terms of Precision (P), Recall (R), F-Measure (F), and Accuracy (Acc), using 10-folds cross validation. For the WCL-3 approach and the Star Patterns see (Navigli and Velardi, 2010), and (Cui et al., 2007) for Bigrams.

Algorithm	P	R	F
WCL-3	78.58%	60.74% *	68.56%
Our system	**83.05%**	**68.64%**	**75.16%**

Table 2: Evaluation results for the hypernym relation extraction, in terms of Precision (P), Recall (R), and F-Measure (F). For the WCL-3 approach, see (Navigli and Velardi, 2010). These results are obtained using 10-folds cross validation (* Recall has been inherited from the definition classification task, since no indication has been reported in their contribution).

5.1 Results

In this section we present the evaluation of our technique on both the tasks of classifying definitional sentences and extracting hypernym relations. Notice that our approach is susceptible from the errors given by the POS-tagger[8] and the syntactic parser[9] . In spite of this, our approach demonstrates how syntax can be more robust for identifying semantic relations. Our approach does not make use of the full parse tree, and we are not dependent on a complete and correct result of the parser.

The goal of our evaluation is twofold: first, we evaluate the ability of classifying definitional sentences; finally, we measure the accuracy of the hypernym relation extraction.

A definitional sentences is extracted only if at least one x and one y are found in the same sentence. Table 1 shows the accuracy of the approach for this task. As can be seen, our proposed approach has a high Precision, with a high Recall. Although Precision is lower than the pat-

[8]http://nlp.stanford.edu/software/tagger.shtml

[9]http://www-nlp.stanford.edu/software/lex-parser.shtml

tern matching approach proposed by (Navigli and Velardi, 2010), our Recall is higher, leading to an higher overall F-Measure.

Table 2 shows the results of the extraction of the whole hypernym relations. Note that our approach has high levels of accuracy. In particular, even in this task, our system outperforms the pattern matching algorithm proposed by (Navigli and Velardi, 2010) in terms of Precision and Recall.

6 Conclusion and Future Work

We presented an approach to reveal definitions and extract underlying hypernym relations from plain text, making use of local syntactic information fed into a Support Vector Machine classifier. The aim of this work was to revisit these tasks as classical supervised learning problems that usually carry to high accuracy levels with high performance when faced with standard Machine Learning techniques. Our first results on this method highlight the validity of the approach by significantly improving current state-of-the-art techniques in the classification of definitional sentences as well as in the extraction of hypernym relations from text. In future works, we aim at using larger syntactic contexts. In fact, currently, the detection does not surpass the sentence level, while taxonomical information can be even contained in different sentences or paragraphs. We also aim at evaluating our approach on the construction of entire taxonomies starting from domain-specific text corpora, as in (Navigli et al., 2011; Velardi et al., 2012). Finally, the desired result of the task of extracting hypernym relations from text (as for any semantic relationships in general) depends on the domain and the specific later application. Thus, we think that a precise evaluation and comparison of any systems strictly depends on these factors. For instance, given a sentence like "In mathematics, computing, linguistics and related disciplines, an algorithm is a sequence of instructions" one could want to extract only "instructions" as hypernym (as done in the annotation), rather than the entire chunk "sequence of instructions" (as extracted by our technique). Both results can be valid, and a further discrimination can only be done if a specific application or use of this knowlege is taken into consideration.

References

M. Berland and E. Charniak. 1999. Finding parts in very large corpora. In *Annual Meeting Association for Computational Linguistics*, volume 37, pages 57–64. Association for Computational Linguistics.

C. Biemann. 2005. Ontology learning from text: A survey of methods. In *LDV forum*, volume 20, pages 75–93.

C. Borg, M. Rosner, and G. Pace. 2009. Evolutionary algorithms for definition extraction. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 26–32. Association for Computational Linguistics.

K.S. Candan, L. Di Caro, and M.L. Sapino. 2008. Creating tag hierarchies for effective navigation in social media. In *Proceedings of the 2008 ACM workshop on Search in social media*, pages 75–82. ACM.

Mario Cataldi, Claudio Schifanella, K Selçuk Candan, Maria Luisa Sapino, and Luigi Di Caro. 2009. Cosena: a context-based search and navigation system. In *Proceedings of the International Conference on Management of Emergent Digital EcoSystems*, page 33. ACM.

C. Cortes and V. Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Hang Cui, Min-Yen Kan, and Tat-Seng Chua. 2007. Soft pattern matching models for definitional question answering. *ACM Trans. Inf. Syst.*, 25(2), April.

R. Del Gaudio and A. Branco. 2007. Automatic extraction of definitions in portuguese: A rule-based approach. *Progress in Artificial Intelligence*, pages 659–670.

I. Fahmi and G. Bouma. 2006. Learning to identify definitions using syntactic features. In *Proceedings of the EACL 2006 workshop on Learning Structured Information in Natural Language Applications*, pages 64–71.

B. Fortuna, D. Mladenič, and M. Grobelnik. 2006. Semi-automatic construction of topic ontologies. *Semantics, Web and Mining*, pages 121–131.

Aldo Gangemi, Roberto Navigli, and Paola Velardi. 2003. The ontowordnet project: Extension and axiomatization of conceptual relations in wordnet. In Robert Meersman, Zahir Tari, and DouglasC. Schmidt, editors, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, volume 2888 of *Lecture Notes in Computer Science*, pages 820–838. Springer Berlin Heidelberg.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

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

E. Hovy, A. Philpot, J. Klavans, U. Germann, P. Davis, and S. Popper. 2003. Extending metadata definitions by automatically extracting and organizing glossary definitions. In *Proceedings of the 2003 annual national conference on Digital government research*, pages 1–6. Digital Government Society of North America.

J.L. Klavans and S. Muresan. 2001. Evaluation of the definder system for fully automatic glossary construction. In *Proceedings of the AMIA Symposium*, page 324. American Medical Informatics Association.

Roberto Navigli and Paola Velardi. 2010. Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden, July. Association for Computational Linguistics.

Roberto Navigli, Paola Velardi, and Juana Mara Ruiz-Martnez. 2010. An annotated dataset for extracting definitions and hypernyms from the web. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

R. Navigli, P. Velardi, and S. Faralli. 2011. A graph-based algorithm for inducing lexical taxonomies from scratch. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, pages 1872–1877. AAAI Press.

R. Navigli. 2009. Using cycles and quasi-cycles to disambiguate dictionary glosses. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 594–602. Association for Computational Linguistics.

S.P. Ponzetto and M. Strube. 2007. Deriving a large scale taxonomy from wikipedia. In *Proceedings of the national conference on artificial intelligence*, volume 22, page 1440. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November.

R. Snow, D. Jurafsky, and A.Y. Ng. 2004. Learning syntactic patterns for automatic hypernym discovery. *Advances in Neural Information Processing Systems 17*.

Paola Velardi, Stefano Faralli, and Roberto Navigli. 2012. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, pages 1–72.

Eline Westerhout. 2009. Definition extraction using linguistic and structural features. In *Proceedings of the 1st Workshop on Definition Extraction*, WDE '09, pages 61–67, Stroudsburg, PA, USA. Association for Computational Linguistics.

I. Yamada, K. Torisawa, J. Kazama, K. Kuroda, M. Murata, S. De Saeger, F. Bond, and A. Sumida. 2009. Hypernym discovery based on distributional similarity and hierarchical structures. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 929–937. Association for Computational Linguistics.

H. Yang and J. Callan. 2008. Ontology generation for large email collections. In *Proceedings of the 2008 international conference on Digital government research*, pages 254–261. Digital Government Society of North America.

Chunxia Zhang and Peng Jiang. 2009. Automatic extraction of definitions. In *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pages 364 –368, aug.