Word Sense Disambiguation for Automatic Taxonomy Construction from Text-Based Web Corpora

Jeroen de Knijff, Kevin Meijer,
Flavius Frasincar, and Frederik Hogenboom

Erasmus University Rotterdam
PO Box 1738, NL-3000 DR
Rotterdam, The Netherlands

{312470jk, 312177km}@student.eur.nl
{frasincar, fhogenboom}@ese.eur.nl

Abstract. In this paper, we propose the Automatic Taxonomy Construction from Text (ATCT) framework for building taxonomies from text-based Web corpora. The framework is composed of multiple processing steps. Firstly, domain terms are extracted using a filtering method. Subsequently, Word Sense Disambiguation (WSD) is optionally applied in order to determine the senses of these terms. Then, by means of a subsumption technique, the resulting concepts are arranged in a hierarchy. We construct taxonomies with and without WSD and we investigate the effect of WSD on the quality of concept type-of relations using an evaluation framework that uses a golden taxonomy. We find that WSD improves the quality of the built taxonomy in terms of the taxonomic F-Measure.

1 Introduction

Nowadays, an ever increasing amount of documents is digitally stored and readily available on the Web. A common issue with managing these documents is that many of these are organized in an unstructured manner. Organizing these documents in a structured way by creating a taxonomy representation to clarify documents can be beneficial, as it enhances the overview of available documents. A taxonomy is defined as a specific form of an ontology, which is a formal, explicit specification of a shared conceptualization [5] that provides users with insight into the type relations between (domain) concepts.

Currently, many taxonomies are manually created. While manually constructing taxonomies is usually more accurate because of the involvement of domain experts, automatically generating taxonomies is often less costly and time consuming. Taxonomy construction on the Web is particularly of interest, as it enables inter-operability between Web sites, tools, etc., due to the knowledge aggregation into shared taxonomies. The Web fosters an incredible large

amount of information and knowledge that up until know is mostly not linked and even remains virtually invisible because of the lack of structure. Within the last decade, there has been a trend of connecting related data that has not been previously linked before. The well-know Linked Open Data cloud diagram[1], which aims to show the inter-connectivity of today's Web sites and their knowledge bases, grows by the year. However, this cloud could grow at an even higher rate and become increasingly more complex when widely applying automatic taxonomy construction. Hence, due to the need for structured information, as well as the complexity and considerable effort for manually creating taxonomies, automatic taxonomy construction is an interesting field to explore.

Although there is a substantial body of literature on automatic taxonomy construction, the amount of literature focussing on applying Word Sense Disambiguation (WSD) is limited, even though WSD is proven to be able to improve the results of clustering. Hence, we evaluate the influence of a sophisticated WSD algorithm on an existing concept subsumption method. We propose a framework for Automatic Taxonomy Construction from Text (ATCT), which we use for the evaluation of the added value of WSD in taxonomy construction.

The main contribution of this paper is four-fold. Firstly, we analyze the influence of WSD on taxonomy construction. Secondly, we investigate the optimal parameters for the methods that comprise the ATCT framework. Thirdly, we modify the subsumption algorithm introduced in [11] to take the position of the ancestors with respect to the current node into account. Finally, we present an implementation of this approach for the domain of economics and management, as well as the medical domain. To our knowledge, taxonomy construction has been applied to other domains, e.g., finance and tourism [3], but not to these domains before.

The organization of our paper is as follows. Section 2 includes a review of related work in the area of taxonomy construction. Section 3 describes in detail the ATCT framework that generates the taxonomies. Finally, the framework and its implementation are evaluated in Sect. 4 and conclusions are drawn in Sect. 5.

2 Related work

Extracting terms from text corpora can be done by means of linguistic methods, statistical methods, and hybrid methods. Linguistic methods generally use Natural Language Processing (NLP) techniques, such as Part-Of-Speech (POS) tagging, morphological analysis, and lexico-syntactic patterns [6]. Linguistic methods are very well capable of defining the function of a word in a sentence, but they do not take into consideration the importance of a term. Statistical methods only use statistical techniques to extract terms from text. A problem with statistical methods is that they can filter out less frequently occurring important terms, as linguistic functions are ignored. This resulted in the development of hybrid methods, using chi-square measures, term lengths, etc. [12, 13].

[1] The diagram is based on data from the W3C SWEO Linking Open Data Community Project and is regularly updated at `http://richard.cyganiak.de/2007/10/lod/`.

Several similarity measures exist for Word Sense Disambiguation (WSD), such as the fast (yet possibly inaccurate) Resnik's similarity [2], which calculates similarity values between terms by analyzing the degree of information they share. Jiang and Conrath's similarity measure [7] is more accurate, as it takes into account the information content of the lowest common subsumer as well as of the terms themselves.

Multiple techniques can be applied to construct hierarchical relations between terms. For example, one could employ (hierarchical) clustering techniques using various similarity measures, e.g., window-methods and co-occurrences. Labeling clusters can be done by selecting the centroid of the cluster or the lowest hypernym of terms in a cluster as a label [3]. Another approach to taxonomy construction is the usage of a classification method. One could combine a domain corpus, a general corpus, and a named-entity tagger to extract and arrange terms in a taxonomy using additional sources that provide more information about these terms in a tree-ascending or tree-descending way [13]. Classification methods can provide accurate results, but require a large training set, making this method difficult to use when a large training set is not available. A final approach to hierarchical relation creation is the usage of lexical-syntactic patterns, by creating taxonomic relations through pattern matching [6]. In the subsumption method, based on co-occurrences, a term subsumes another term if they co-occur frequently. This method is simple and it does not have any labeling issues, but it is weak in arranging terms that do not occur frequently in documents [11].

3 ATCT Framework

In our four-step ATCT framework for automatically constructing a domain-specific taxonomy, first the terms are extracted from the documents, which subsequently get processed in our term filtering step. Here, terms are filtered on lexical cohesion and domain pertinence, after which the most relevant terms are selected on the basis of a score, which is determined by domain pertinence, domain consensus, and structural relevance. The selected terms are processed into concepts as concept labels. The next optional step is *Word Sense Disambiguation* (WSD), which is used to derive the sense of a concept term and to find synonyms of the disambiguated term. Lastly, a *concept hierarchy* is created by constructing the type relations between concepts. The resulting hierarchy is represented in SKOS [1], a commonly used domain taxonomy representation format.

3.1 Term Extraction

Within the ATCT framework, terms are extracted from a set of documents by tagging all the words that appear in these documents and extracting those terms that are tagged as a noun. The choice for nouns is motivated by the fact that concepts are usually represented by nouns rather than other types of words, which is also the case for our utilized golden taxonomies.

3.2 Term Filtering

Extracted terms are filtered on multiple criteria, i.e., their domain pertinence and their lexical cohesion value. The most relevant terms are selected by calculating a score, which is based on the domain pertinence, domain consensus and structural relevance of the term [12].

Domain pertinence measures whether a term is relevant for the target domain. The more frequently a term appears in the domain corpus and the less frequently the term appears in the contrastive corpus, the higher the domain pertinence. Lexical cohesion measures cohesion among words in a term and is only used for compound nouns. Based on initial experiments, we find that for optimal performance for both measures the 30% terms with the lowest values should be filtered out. Domain consensus checks whether a term is important, i.e., it appears in several documents, and structural relevance is used for measuring importance of terms in relation to their document position appearance (title or body). Finally, the filters are combined and a weighted score is calculated. The terms with the highest scores are selected as concept labels that appear in the constructed domain taxonomy.

3.3 Word Sense Disambiguation

The optional WSD procedure is used for deriving the sense of a concept term and for finding the synonyms of the concept term. In order to find the sense of a term, we follow an approach that is based on the SSI algorithm [10]. First, we retrieve the possible senses that are associated to the term. For this, we employ a semantic lexicon, which is a large lexical database that contains many words with their synsets (collections of synonyms). When a term is recognized as a noun in the semantic lexicon, we retrieve the possible synsets for that particular term. One of those synsets is selected as the sense for a term by following a number of steps. The best suited sense is determined as follows:

$$sense_t = \max_{s_i \in S_t} \sum_{c_j \in C_t} sim(s_i, c_j) \tag{1}$$

where S_t is the set of possible senses for term t, C_t is the set of context senses, and $sim(s_i, c_j)$ is the similarity between possible term sense s_i and context sense c_j. The similarity measure we use is the one proposed by Jiang and Conrath [7], as this has proven to be an accurate and fast similarity measure [2].

3.4 Concept Hierarchy Creation

In order to build the actual concept hierarchy, the subsumption algorithm discussed in [11] is employed. In order to improve the quality of this algorithm we take the position of the ancestors with respect to the current node into account. For each concept, the potential parent concepts (subsumers) are determined. If $P(x \mid y) \geq t, P(y \mid x) < t$, where t is a co-occurrence threshold, x potentially

subsumes y. If x appears in at least the proportion t of all documents in which y appears and if y appears in less than the proportion t of all documents in which x appears, x is a potential parent of y. When the potential parents are found, the parent is determined by calculating a score for each potential parent:

$$score(p, x) = P(p \mid x) + \sum_{a \in A_p} w(a, x) \cdot P(a \mid x) \ , \tag{2}$$

where p is the potential parent node of x, A_p represents the list of ancestors of p, and $w(a, x)$ denotes a weight value with which each co-occurrence probability of ancestor a and x is multiplied. This weight is influenced by the amount of layers between ancestor a and node x:

$$w(a, x) = \frac{1}{d(a, x)} \ , \tag{3}$$

where $d(a, x)$ is the length of the path between node x and ancestor a. When the scores for all the potential parent concepts are calculated, the potential parent with the highest score is chosen as the parent of x.

3.5 ATCT Implementation

We implemented the ATCT framework as a Java-based tool, which parses nouns from texts by means of the Stanford [8] parser. For presenting RDF representations we employed the Jena framework [9]. Our domain taxonomies are exported as RDF files using a SKOS vocabulary [1], so that we can compare our taxonomy with related SKOS taxonomies.

4 Evaluation

For evaluation purposes, we compare two taxonomies generated with and without applying WSD by our implementation, each consisting of 2,000 distinct concepts. The used corpus contains 25,000 abstracts extracted from RePub (`http://repub.eur.nl/`) and RePEc (`http://repec.org/`). The golden taxonomy is preprocessed (translated and pruned) version of STW Thesaurus for Economics (`http://zbw.eu/stw/`), which is a manually created taxonomy in the field of economics and business economics, containing thousands of terms more than our automatically generated taxonomy. Additionally, we construct two taxonomies for the domain of medicine and health, using the large MeSH ontology (`http://onto.eva.mpg.de/obo/mesh.owl`) for arranging medical subject headings as a reference. The built taxonomy consists of 1,000 concepts and is constructed from a total of 10,000 RePub abstracts.

4.1 Experimental Setup

We evaluate the built taxonomies on two levels, using measures from literature [4]. We use the lexical precision (LP) and lexical recall (LR) to evaluate

to what degree the concepts of our constructed taxonomy are lexically shared with the golden taxonomy. We also measure the quality of the type-of relations by using the common semantic cotopy (*csc*), which is the collection of a concept and the concept's sub- and super-concepts that are shared between a core ontology and a reference ontology. Two measures that apply the *csc* to measure the quality of type-of relations of an ontology are the global taxonomic precision (TP) and the global taxonomic recall (TR), where TP reflects how similar the relations in the intersection of both ontologies are with respect to the core ontology, while TR reflects how similar the relations in the intersection of the ontologies are with respect to the reference ontology. Last, we apply the taxonomic F-Measure (TF), which is the harmonic mean of TP and TR, to compute the overall quality of the concept type-of relations.

4.2 Experimental Results

When comparing the generated taxonomies with the golden standards, we obtain relatively low lexical precision and recall. For the domain of economics and management, lexical precision and recall are merely 0.1769 and 0.0926, respectively. This can be explained by analyzing the reference taxonomy we used. The STW taxonomy mainly uses the categories (abstract) in the economics and management domain, while our taxonomy uses specific terms in this domain. For the domain of medicine and health, lexical precision is somewhat higher, i.e., 0.2388, while the lexical recall of 0.0156 is very low. The latter low value is explained by the fact that the MeSH ontology consists of 15,337 concepts, which is much higher than the size of our built taxonomy (1,000 concepts).

In order to be able to properly evaluate the quality of taxonomies that are built by applying WSD we use the semantically shared concepts of the core ontologies and reference ontologies rather than the lexically shared concepts to retrieve the quality of the type-of relations as concepts are now disambiguated.

We have applied a WSD approach specific for existing taxonomies on the golden taxonomies to disambiguate its concepts in order to be able to retrieve the semantically shared concepts, which is similar to the one we used for disambiguating concepts from text corpora. The difference is that we now use the surrounding taxonomy concepts (concept neighborhood) of a concept to disambiguate a concept rather than using text surroundings. The concept neighborhood consists of the concepts that are ancestors of a concept, and the concepts that are descendants of a concept. In case none of the concepts surrounding a taxonomy concept can be disambiguated, the most common sense is selected.

We investigated what percentage of concepts were disambiguated correctly for the ontologies for different amounts of ancestor and descendant layers. We found that for disambiguating a concept the best results are obtained when using a concept neighborhood that consists of two layers of ancestor concepts and two layers of descendant concepts. Further increasing the size of the concept neighborhood does not improve the results, while decreasing the concept neighborhood size lowers the percentage of concepts disambiguated correctly.

Table 1. Quality measurements for resulting taxonomies with and without the use of WSD, within the domain of economics and management (E&M), as well as within the domain of medicine and health (M&H).

Domain	Taxonomy	TP	TR	TF
E&M	T_1 (without WSD)	0.7382	0.5082	0.6023
	T_2 (with WSD)	0.8056	0.5813	0.6753
M&H	T_1 (without WSD)	0.5681	0.6051	0.5860
	T_2 (with WSD)	0.5907	0.6016	0.5961

Table 1 shows several quality measurements for the type-of relations. We distinguish between the two domains and between applying the framework with and without WSD. The table shows the TP, TR, and TF. Here, T_1 denotes the taxonomy without WSD and T_2 denotes the taxonomy with WSD. For the domain of economics and management, both taxonomies have high TF values, implying that the taxonomic relations between concepts in both taxonomies are good. The taxonomy with WSD seems to perform better than the taxonomy without WSD on all three quality measures. WSD improves TF by approximately 12.12% for the economics and management domain. WSD thus improves the quality of the concept type-of relations in our experiment. For the medicine and health domain, TF is again higher with WSD, but the increase is not as high as for economics and management. Nevertheless, for all built taxonomies WSD had a positive effect on the overall quality of the type-of relations.

5 Conclusions

We have presented the ATCT framework for the automatic generation of a domain taxonomy from text. The framework extracts potential taxonomy terms from a large corpus, resulting in a number of the most relevant terms, after having filtered the potential terms for domain pertinence, domain consensus, lexical cohesion, and structural relevance. When disambiguating term senses, a sense and alternative labels (synonyms) are added to the concept. Concepts with the same senses are removed from the concept list. Subsequently, taxonomic relations are created by means of a subsumption method, which arranges concepts in a taxonomy according to their co-occurrence in documents. After implementation, we found in our experiments that the usage of WSD in automatic taxonomy construction improves the performance measured in terms of the TF-value by 12.12% with respect to the method without WSD.

For future research it would be interesting to benchmark our method against other taxonomy creation methods, such as hierarchical clustering or classification methods. Furthermore, we would like to investigate the impact of WSD on these other methods. Exploring other term extraction methods such as lexico-syntactic patterns in combination with our framework is also a research direction that we would like to pursue. Finally, we would like to investigate the application of our approach to other domains, e.g., law, chemistry, physics, or history.

Acknowledgements

The authors are partially sponsored by the Dutch Organization for Scientific Research (NWO) Physical Sciences Free Competition project 612.001.009: Financial Events Recognition in News for Algorithmic Trading (FERNAT).

References

[1] Bechhofer, S., Miles, A.: SKOS Simple Knowledge Organization System Reference - W3C Recommendation 18 August 2009 (2009), Available at: http://www.w3.org/TR/2009/REC-skos-reference-20090818/

[2] Budanitsky, A., Hirst, G.: Semantic Distance in WordNet: An Experimental, Application-Oriented Evaluation of Five Measures. In: Workshop on WordNet and Other Lexical Resources, 2nd Meeting of the North American Chapter of the Assocation for Computational Linguistics (NAACL 2001). pp. 29–34. Association for Computational Linguistics (2001)

[3] Cimiano, P., Hotho, A., Staab, S.: Learning Concept Hierarchies from Text Corpora Using Formal Concept Analysis. Journal of Artificial Intelligence Research 24(1), 305–339 (2005)

[4] Dellschaft, K., Staab, S.: On How to Perform a Gold Standard Based Evaluation of Ontology Learning. In: 5th Int. Semantic Web Conf. (ISWC 2006). Lecture Notes in Computer Science, vol. 4273, pp. 228–241. Springer (2006)

[5] Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5(2), 199–221 (1993)

[6] Hearst, M.A.: Automatic Acquisition of Hyponyms from Large Text Corpora. In: 14th Conf. on Computational Linguistics (COLING 1992). vol. 2, pp. 539–545 (1992)

[7] Jian, J.J., Conrath, D.W.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: 10th Republic of China Computational Linguistics Conf. on Research in Computational Linguistics (ROCLING 1997). pp. 19–33. The Association for Compuational Linguistics and Chinese Language Processing (1997)

[8] Klein, D., Manning, C.D.: Fast Exact Inference with a Factored Model for Natural Language Processing. In: 16th Annual Conf. on Neural Information Processing Systems (NIPS 2002). Advances in Neural Information Processing Systems, vol. 15, pp. 3–10. MIT Press (2002)

[9] McBride, B.: Jena: Semantic Web Toolkit. IEEE Internet Computing 6(6), 55–59 (2002)

[10] Navigli, R., Lapata, M.: Graph Connectivity Measures for Unsupervised Word Sense Disambiguation. In: Veloso, M.M. (ed.) 20th Int. Joint Conf. on Artificial Intelligence (IJCAI 2007). pp. 1683–1688. AAAI Press (2007)

[11] Sanderson, M., Croft, B.: Deriving Concept Hierarchies from Text. In: 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR 1999). pp. 206–213. ACM (1999)

[12] Sclano, F., Velardi, P.: TermExtractor: a Web Application to Learn the Shared Terminology of Emergent Web Communities. In: 7th Conf. on Terminology and Artificial Intelligence (TIA 2007). Presses Universitaires de Grenoble (2007)

[13] Weber, N., Buitelaar, P.: Web-based Ontology Learning with ISOLDE. In: Workshop on Web Content Mining with Human Language at the 5th Int. Semantic Web Conf. (ISWC 2006) (2006), Available at: http://www.dfki.de/dfkibib/publications/docs/ISWC06.WebContentMining.pdf