Renato Domínguez García, Sebastian Schmidt, Christoph Rensing, Ralf Steinmetz: Automatic Taxonomy Extraction in Different Languages using Wikipedia and Minimal Language-specific Information. In: Alexander Gelbukh: Computational Linguistics and Intelligent Text Processing, Proceedings of the 13th International Conference on Intelligent Text Processing and Computaional Linguistics, CICLing 2012, LNCS 7181, pp. 42-53, March 2012. ISBN: 978-3642286032

Automatic Taxonomy Extraction in Different Languages using Wikipedia and minimal language-specific Information

Renato Domínguez García, Sebastian Schmidt, Christoph Rensing, and Ralf Steinmetz

Multimedia Communications Lab - Technische Universität Darmstadt
64283 Darmstadt - Germany
{renato.dominguez.garcia,sebastian.schmidt,christoph.rensing,ralf.steinmetz}@kom.tu-darmstadt.de
http://www.kom.tu-darmstadt.de

Abstract. Knowledge bases extracted from Wikipedia are particularly useful for various NLP and Semantic Web applications due to their coverage, actuality and multilingualism. This has led to many approaches for automatic knowledge base extraction from Wikipedia. Most of these approaches rely on the English Wikipedia as it is the largest Wikipedia version. However, each Wikipedia version contains socio-cultural knowledge, i.e. knowledge with relevance for a specific culture or language. In this work, we describe a method for extracting a large set of hyponymy relations from the Wikipedia category system that can be used to acquire taxonomies in multiple languages. More specifically, we describe a set of 20 features that can be used for for Hyponymy Detection without using additional language-specific corpora. Finally, we evaluate our approach on Wikipedia in five different languages and compare the results with the WordNet taxonomy and a multilingual approach based on interwiki links of the Wikipedia.

Keywords: Hyponymy Detection, Multilingual large-scale taxonomies, Wikipedia Mining, NLP

1 Introduction

Natural language processing (NLP) covers all steps of processing natural language from the syntactical representation (or audio representation) to the discourse. While the first steps aim at breaking down and analyzing the structure of the text, the latter steps cope with/handle reassembling and understanding. All those steps require human knowledge in machine processable form to be executable. Whereas the knowledge required for the first steps is of very local scope, which means the processing of the single tokens is only minimally dependent on neighbouring tokens and only requires a small number of rules, subsequent steps require more and more context. This holds for both, the context of the text in

the document itself and for the general knowledge required to derive the understanding of the text. For the latter, a structured knowledge base on the specific topic enables a machine to derive knowledge and put it into an abstract context. One of those knowledge bases is a taxonomy. Within a taxonomy, relations of the type *is-a* are contained, creating a tree-like structure of real world concepts. One example of such a *is-a* relation is the tuple *(juice, beverage)* because juice is a beverage.
In some fields of knowledge, like biology, elaborated taxonomies already exist. But there are still many domains without such explicit taxonomies. Additionally, these taxonomies are usually only defined in one language. Therefore, although a variety of taxonomies are existing in English, other languages lack these. This impedes the application of taxonomies in several languages and fields of knowledge impossible. Within this paper we present an approach to create taxonomies in different languages automatically from the category and article structure of Wikipedia. Our approach uses structural properties of Wikipedia and syntactical structure of single categories and articles, and requires only minimal language-specific information.
After giving an overview of existing work on automatic taxonomy creation (section 2), we will present our machine learning approach in section 3. Its evaluation is shown in chapter 4 and concluded in the last section.

2 Related Work

As one of the key challenges for NLP applications is to allow the extraction of machine-usable knowledge from written language, a lot of research on the topic has been conducted. In this section, we will give a short overview on approaches using for this purpose. We focus on approaches based on Wikipedia and exclude approaches based on Text Mining as these approaches rely on lexical patterns. Lexical patterns can be applied on texts in different languages in order to obtain taxonomies from scratch. but they are strongly language-dependent.
WordNet [8] is a knowledge base consisting of English words with short definitions and both lexical and semantic relations between those words. Semantic relations comprise hypernomy, hyponymy and synonymy among others. This enables the direct extraction of a taxonomy from WordNet. Within the Universal WordNetProject [2] based on wordnets in different languages and other information sources first an initial graph was built which is afterwards enriched by adding missing links and then iteratively refined by making use of machine learning techniques. The result is a multilingual lexical database of terms in combination with their meanings, containing relations between the terms. However, WordNet is a manually built resource and has to be catered for by linguistic experts. Thus, its growth is slow and novel, domain-specific or trending topics are usually not covered. The same holds for other manually created knowledge bases. Therefore, in scenarios that depend on the availability of a domain-generic set of concepts and up-to-date knowledge, these approaches do not suffice.

Sumida and Torisawa [16] make use of the structure of Wikipedia articles to extract *hyponomy* relations. Often, Wikipedia articles are structured in a way, that subsections in Wikipedia articles describe a *hyponym* of the wrapping section (e.g. the article *Sense* has the section *Senses* with the subsections *Sight*, *Taste* etc.). Sumida and Torisawa use this for discovering *hyponomy* relations by applying language dependent pattern matching and use of machine learning techniques to differentiate between *hyponymy* and *non-hyponymy* relations.
Ponzetto und Strube [13] use of this structure and aim to identify *hypernymy* and *hyponomy* between Wikipedia categories. Wikipedia categories build a large network containing links of different types. In many cases there is a subtype relation between two categories (e.g. the category *Juice* is connected with the category *Non-alcoholic beverages*), but in general it can be any kind of semantic relation (e.g. the category *Titles* is connected with *Sociolinguistics*). Therefore, they identify that can disambiguate taxonomic relations from others. For applying the approach to other languages the algorithm itself has to be modified because most of the features are strongly dependent on the used language. Further, the Tipster corpus [6] which is used for one step of *hyponomy* relation detection is not on hand in other languages than English. Kassner et al. [7] adapt the approach to be used with the German Wikipedia. In addition to the smaller size of the German Wikipedia they had to face the challenge of a more complicate word composition in German compared to English.
Navigli and Ponzetto present BabelNet [11], a multilingual semantic network created by the aggregation of WordNet, Wikipedia and SemCor. Additionally to those resources, the Google Translator API[1] is used to translate article names of the Wikipedia which do not have a correspondent in other languages. We see three problems with this approach. First, it relies on Statistical Machine Translation and its ability to translate to other languages. Second, the Google Translator has strong usage restrictions, which makes it not suitable for a publicly available resource. Finally, BabelNet use *interwiki links*[2] to build the multilingual semantic network. However, *interwiki links* do not exist for many articles in Wikipedia.
Another approach currently developed for creating a taxonomy from the English Wikipedia is WikiNet[10]. By analyzing categories and articles of the English Wikipedia a monolingual concept network is created. Afterwards, for all included concepts the *interwiki links* are examined and a multilingual concept network is created by adding all articles being interlinked by those links. The authors describe the portability of this approach to other languages. However, the impact of combining category systems in different languages is not clear.
A common disadvantage of the previously presented approaches is the loss of socio-cultural knowledge which is available in Wikipedia. Some artifacts of knowledge are only relevant for a single region with a single spoken language. Those artifacts are often only covered in the Wikipedia of the respective language. When

[1] `http://code.google.com/apis/language/` - retrieved 28.10.2011
[2] Links from a Wikipedia article in one language to an article in another language describing a similar concept

creating taxonomies only based on the English Wikipedia, all socio-cultural knowledge described in other Wikipedia versions can not be transferred into the taxonomy.
MENTA [3] addresses this issue by providing a multilingual taxonomy consisting of entities in various languages. To this end, all similiar entities from different Wikipedia versions are merged and afterwards, both syntactical and structural properties of Wikipedia and WordNet are used to determine taxonomic relations between the entities. This approach is not fully automatic but some linguistic exceptions for syntactical rules need to be specified manually.
As presented, there are several approaches to create taxonomies in different languages (semi-)automatically. They all show to have advantages, but none of them is at the same time easily adaptable to different languages, accurate and dynamic in terms of trending topics. With our approach, we aim at targeting those challenges.

3 Language-independent Acquisition of Hierarchical Relations

Two preliminary studies are the basis of our approach: In the first study [4] we analyzed the feasibility of Hyponymy detection in different languages using simple heuristics and in the second study [5], we described our application scenario for taxonomies in different languages and performed first experiments with a machine learning approach. In the following subsections we present our approach which involves of a set of 20 features to recognize *is-a* relations from Wikipedia categories. These features are described in this section.

3.1 The Feature Set for Recognizing *is-a* relations

We take pairs of categories (in following denoted as c_1 and c_2) from the Wikipedia category graph and apply our features to each of these pairs, called *links*. The returned values are used to build the *feature vector*, which is used by a classifier to determine if there is an *is-a* relation between both categories. In table 1, we present an overview of the used features. In the following, these features are described in detail.

Preprocessing Features These features are used to detect links containing nodes to administrative or refinement categories and evaluate to true, if the category belongs to one of those. Administration categories contain prefixes like `Wikipedia` or `User`. Refinement links are used in Wikipedia to organize multiple categories using the pattern `X by Y` (e.g. "Companies by country"). Their purpose is to structure and simplify the category graph. The prefixes of administration categories and the preposition used in the refinement links are used in all Wikipedia versions independently of the language. However, they have to be adapted to the respective language. For instance, the prefix `category` is translated to `Kategorie` in German and `Categoría` in Spanish.

Table 1. Overview of the used features

	Name	Value type	Feature type
1	`adminCatFeature`	binary	preprocessing
2	`refinementLinkFeature`	binary	preprocessing
3	`positionOfHeadFeature`	$\{2, 1, 0, -1\}$	syntactic
4	`coocurrenceOfWordsFeature`	$\mathbb{N}$	syntactic
5	`coocurrenceArticleFeature`	binary	structural
6	`commonArticleFeature`	$\{1, 0, -1\}$	structural
7	`c1c2IncomingLinksFeature`	$\{1, 0, -1\}$	structural
8	`c1c2OutgoingLinksLinksFeature`	$\{1, 0, -1\}$	structural
9	`c1distanceCommonAncestorFeature`	$\mathbb{N}$	structural
10	`c2distanceToCommonAncestorFeature`	$\mathbb{N}$	structural
11	`c1NumberOfSubcategoriesFeature`	$\mathbb{N}$	structural
12	`c1NumberOfSuperCategoriesFeature`	$\mathbb{N}$	structural
13	`c2NumberOfSubcategoriesFeature`	$\mathbb{N}$	structural
14	`c2NumberOfSuperCategoriesFeature`	$\mathbb{N}$	structural
15	`CommonWikilinksFeature`	$\mathbb{N}$	structural
16	`firstSentenceFeature`	binary	article
17	`RedirectFeature`	$\{1, 0, -1\}$	article
18	`c2Inc1Feature`	$\mathbb{N}$	article
19	`c1ArticleFeature`	binary	article
20	`c2ArticleFeature`	binary	article

Syntactic Features Syntactic features use string matching of syntactic components to differentiate between *is-a* and *not-is-a* links. We distinguish between two different syntactic features. The `positionOfHeadFeature` uses the fact that the lexical head of two category names is a very effective method for labeling *is-a* links [13]. This feature returns a value $[-1, 2]$ for pairs of categories c_1 and c_2 representing the position of the lexical head of the superordinate category. We differentiate between the following cases:

$$f_3(c_1, c_2) = \begin{cases} 2 & \text{if lexical head of } c_2 \text{ is at the end of } c_1 \\ 1 & \text{if lexical head of } c_2 \text{ is in the middle of } c_1 \\ 0 & \text{if lexical head of } c_2 \text{ is at the beginning of } c_1 \\ -1 & \text{else, i.e. no occurrence} \end{cases}$$

For instance, the value of this feature for c_1 ="French Revolution" and c_2 = "Revolution" is 2. In the English Wikipedia, the lexical head of a category is usually the last word. However, there are some exceptions, for example for categories containing prepositions e.g. "Campaign for nuclear disarmament" or containing refinement brackets e.g. "Sport (Ireland)". We cope with this issue by recognizing prepositions heuristically, i.e. matching preposition and using the term before the preposition. This heuristic works for other languages as well: In Arabic, for instance, where the lexical head is at the beginning of a category name or in German, where the lexical head is "hidden" inside noun compounds where the multiple noun modifies the meaning given by the last one, e.g. "Baumhaus" (Eng. tree house). If the position of the head is not the head position in a given language then we assume that there is a *not-is-a* relation between c_1 and c_2. `cooccurrenceOfWords` represents cooccurrences of words in both category names. This feature should match cases, in which two category labels have more words in common than the lexical head.

Structural Features These features exploit the structure of the category graph and the wikilink graph[3]. `coocurrenceFeature` returns `true` for pairs of categories which have at least one article in common [13]. Further, `commonArticleFeature` returns the number of articles in common between both categories. `c1c2IncomingLinksFeature` and `c1c2OutgoingLinksFeature` measure the strength of the relation between both categories. For this purpose, the number of articles in c_1 is counted, which have at least one incoming or outgoing wikilink to any article in c_2 [1]. The features `c1distanceCommonAncestorFeatu- re` and `c2distanceCommonAncestorFeature` calculate the distance between of given categories c_1 and c_2 to the first common ancestor c_A of both categories. Both distances are calculated separately, i.e. `c1distanceCommonAncestorFeatu- re` calculates the distance of c_1 to c_A and feature `c2distanceCommonAncestorFeature` the distance of c_2 to c_A. If $c_i = c_A$ then the distance for c_i is 0. `c1NumberOfSubcategories`, `c1NumberOfSupercategories`, `c2NumberOfSubcategories` and `c2NumberOfSupercategories` just counts the number of sub- and supercategories of c_1 and c_2. Categories having a huge number of subcategories usually represent more abstract concepts which can be referenced by many other categories, e.g. "Science". Finally, `CommonWikilinksFeature` counts the number of common wikilinks between c_1 and c_2.

Article Features This set of features is applied to the content of articles. The first sentence of an article has a special meaning for taxonomic applications as it usually contains a definition of the concept [12]. This fact is used by `definitionSentenceFeature` to recognize *is-a* relations in the first sentence. This means that if an article a belongs to a category c_1 with both having the same label, then we search for occurrences of lexical heads of c_2 in a in the first sentence of the article. For instance, if c_1 = "Mice" and c_2 = "Pet Rodent", we test if the first sentence of the article "Mice" contains the term "rodent". If the check is positive, then this feature returns `true`. An advantage of this method is that a language-dependent search of patterns is not needed, and thus it can be applied to different languages. Further, the feature `c2Inc1Feature` counts the number of occurrences of the lexical head of c_2 in the rest of the article of c_1. `c1ArticleFeature` and `c2ArticleFeature` match c_1 and c_2 to Wikipedia articles. If c_2 can be matched to an article then we assume that this category is an existing concept, otherwise it may be a category used to structure the category graph like e.g. lists. In this case `true` is returned. Finally, the feature `RedirectFeature` returns `true` if the article corresponding to c_1 is redirected in Wikipedia to the article corresponding to c_2. This represents a strong relation between both categories. This information is stored in Wikipedia in so called *redirect pages*.

[3] This graph is built by iterating over the Wikipedia articles and adding all links between two articles in the same language version of the Wikipedia

3.2 Language-specific Information needed by our Approach

This approach is applicable as such to very different languages without modifying the features with minimal language-specific information. Specifically, it can be used to derive taxonomies from different languages with little information about a language needed. The mandatory data is the following:

1. A list of prefixes of meta-categories that Wikipedia uses in this language, e.g. `wikipedia`, `user` or `articles`.
2. The preposition contained in refinement-links, e.g. `by` in English or `nach` in German.
3. A list of prepositions of a language in order to match the lexical head of categories containing prepositions heuristically, e.g. "Battalions of the Canadian Expeditionary Force".

The language-independency of our approach is restricted by the 281 existing Wikipedia versions (i.e. we can not acquire taxonomies from other languages) and by the input of the prefixes and the prepositions mentioned before. Using only this information it is possible to generate taxonomies in different Wikipedia languages as we show in the next section.

4 Evaluation

We evaluated our approach in four different languages: three European languages (English, German and Spanish) and two language with non-latin characters (Arabic and Russian). We used a manually labelled corpus for each language to obtain results by applying our approach for multiple languages.

4.1 Building the different Corpora

Our corpus consists of 1000 randomly selected Wikipedia articles and categories. We extracted the corpus using Wikipedia's export page[4] and following method:

1. Get random article a_i using the "Random page"-link[5] and add all links of a to all its categories in our corpus.
2. Choose a random category c of a and add all links of c to all its super categories $c_{s,i}$ in our corpus. As our corpus should contain 1000 articles, we filter out categories that have more than 100 super categories in order to have enough articles and categories from different domains.
3. Choose randomly a super category $c_{s,j}$ of $c_{s,i}$ and all links of $c_{s,j}$ and insert it into our corpus.
4. Repeat step 3. until the root category[6] or an already visisted category is reached moving to the top of the category graph.

[4] `http://en.wikipedia.org/wiki/Special:Export` - retrieved 28.10.2011
[5] `http://en.wikipedia.org/wiki/Special:Random` - retrieved 28.10.2011
[6] `http://en.wikipedia.org/wiki/Category:Contents` - retrieved 28.10.2011

5. Go to step 1, until corpus has 1000 articles.

After we built the corpus, we labelled it manually with the relevant relations (*is-a* and *not-is-a*). In Table 2 we summarize the size and distribution between *is-a* and *not-is-a* links in the different corpora.

Table 2. Summarized statistics of the different corpora

Language	English	Spanish	German	Arabic	Russian
Number of *is-a* links	1297 (29.9 %)	786 (36.1 %)	808 (33.6 %)	1135 (41.4 %)	2545 (40.4 %)
Number of *not-is-a* links	3048 (70.1 %)	1388 (63.9 %)	1597 (66.4 %)	1604 (58.6 %)	3752 (59.6 %)
Number of labeled links	4345	2174	2405	2739	6297

4.2 Evaluation results

In this section, we present the results of our evaluation. We used the Weka Machine Learning Toolkit [17] and chose J48 decision trees (Weka implementation of C 4.5 [14]) as a classifier. Decision trees are a commonly used classifier as they are fast to train and in classifying instances, their rules are simple to understand and they can be combined with other decision techniques in order to improve results. All classification results were subjected to ten-fold cross validation. Table 3 gives an overview of our results. It shows correctly and incorrectly classified instances. On average, 83.1 % of the links are labelled correctly and 16.9 % are labelled incorrectly.

Table 3. Summarized results of our approach by languages

Language	English	Spanish	German	Arabic	Russian
Correctly classified inst.	3583 (82.6 %)	1838 (84.6 %)	1963 (81.6 %)	2283 (83.4 %)	5067 (80.5 %)
Incorrectly classified inst.	753 (17.4 %)	336 (15.4 %)	442 (18.4 %)	456 (16.6 %)	1230 (19.5 %)
Total number of instances	4345	2174	2405	2739	6297

Table 4 summarizes the most common metrics of evaluation of categorization algorithms: Precision, Recall and F-Measure. For Precision, we obtained average results of 74.5 % and Recall was 77.2 % for *is-a* relations and for *not-is-a* relations Precision was 87 % and Recall 86.3 %. The English confusion matrix is additionally shown in Table 5.
Table 5 shows that the major source of misclassification is incorrectly classified *is-a* links. The reason is that a high number of *is-a* instances could not be recognized as *is-a* by single features, but by the combination of multiple feature values. Thus, these combinations can recognize more instances than simple heuristics, but they introduce some misclassified instances as they do not have a precision of 100 %. One possibility to improve the results presented here is to use cross-language links to integrate the results from different languages. Further,

Table 4. Detailed Accuracy by class and language

	Precision	Recall	F-Measure	Class
English	70.0 %	73.0 %	71.2 %	is-a
	88.3 %	86.7 %	87.5 %	not-is-a
Spanish	76.3 %	83.1 %	79.5 %	is-a
	89.9 %	85.4 %	87.6 %	not-is-a
German	71.9 %	74.4 %	73.1 %	is-a
	86.8 %	85.3 %	86.0 %	not-is-a
Arabic	79.7 %	80.4 %	80.0 %	is-a
	86.0 %	85.5 %	85.7 %	not-is-a
Russian	73.2 %	81.5 %	77.1 %	is-a
	86.4 %	79.8 %	83.0 %	not-is-a

Table 5. Confusion matrix English

a	b	← classified as
944	349	a = is-a
404	2639	b = not-is-a

we evaluated the effect of each type of features measured in Accuracy. We can see in Table 6 that single feature classes in most of the cases do not perform better than 70 %.

Table 6. Effect of feature classes by languages

	Preprocessing features	Syntactic features	Structural features	Article features
English	70.2 %	69.8 %	75.8 %	71.4
Spanish	63.8 %	69.9 %	73.6 %	70.5 %
German	66.4 %	68.1 %	74.1 %	67.2
Arabic	64.2 %	59.8 %	77.8 %	73.5
Russian	66.2 %	64.6 %	69.1 %	66.8

Finally, we rank all features by information gain [9], measuring how well a given feature separates the training instances according to their target classification. This is shown in Table 7. Features recognizing *not-is-a* links are ranked higher as the number of *not-is-a* links is higher than *is-a* links, i.e. they are used to categorize more links. In general, we can see that syntactic and structural features performed best. We observe, that the structural features are better for detecting *not-is-a* links and the syntactic features for *is-a* links.

Furthermore, we observed that the features `distanceC1ToCommonAncestor` and `distanceC2ToCommonAncestor` were not very distinctive. This can be explained by the method used to build our corpus. In our corpus, we collected only direct links between categories and articles. However, we believe that these features could be helpful in other scenarios. For instance, to train a classifier which does not only recognize direct links, but also indirect links, i.e. transitive links. Such a classifier could be used to recognize *is-a* relations independently of our taxonomy scenario. This is going to be part of future work.

4.3 Evaluation of our Approach using Existing Knowledge Bases

It is crucial to evaluate the obtained hierarchical relations in comparison with similar approaches. In this section, we compare the Accuracy of our approach against WordNet [8] and WikiNet [10] for English. WordNet is one of the most

Table 7. Ranking of the used features by languages

	English features	Spanish features	German features	Arabic features	Russian features
1.	c2Inc1Feature	c1c2IncomingLinks	c1c2IncomingLinks	c2Inc1Feature	PositionOfHead
2.	refinementLink	c2Inc1Feature	refinementLink	c1Article	c1c2IncomingLinks
3.	c2Article	PositionOfHead	c1Article	commonArticleFeature	c2Inc1Feature
4.	c1c2IncomingLinks	c2Article	c2Article	firstSentence	refinementLink
5.	c1c2OutgoingLinks	refinementLink	PositionOfHead	refinementLink	firstSentence

popular knowledge bases in English and WikiNet is a knowledge base which was acquired using interwiki links without additional external corpora.

First, we select those pairs of categories that overlap with WordNet and WikiNet. For each category pair, both categories have to be mapped to WordNet synsets and to WikiNet concepts. Our evaluation consists of a set of 15,483 instances which belong to the Wikipedia category graph, WordNet and WikiNet.

These pairs are evaluated by querying WordNet whether the concept denoted by the Wikipedia subcategory (c_1) is an instance or a subclass of the concept denoted by its category (c_2). The WordNet pairs c_1 and c_2 are looked up in direct relation as well as in indirect relation (i.e. c_1 *is-a* ... *is-a* c_2). We then take the result of the query as the actual (*is-a* or *not-is-a*) semantic relation for the category pair and use it to evaluate the results of our approach. The same procedure is done to evaluate the quality of WikiNet on this dataset. This way we are able to compute standard measures of Precision, Recall and F-Measure of our approach and WikiNet and compare the values, i.e. the information contained in WordNet is used a gold Standard.

Table 8 gives an overview of the results. It shows correctly classified instances by our approach and by WikiNet. 85.95 % of the labelled links were labelled by our approach correctly and 78.23 % by WikiNet. Table 9 shows detailed results

Table 8. Results of our approach and WikiNet compared with WordNet

	Our approach	WikiNet
Correctly classified	13307 (85.95%)	12113 (78.23%)
Incorrectly classified	2176 (14.05%)	3370 (21.77%)
Total number of inst.	15483	15483

of both approaches in our evaluation corpus. For F-Measure, we obtained results of 80.48 % for *is-a* relations and 89.02 % for *not-is-a* relations. These results were significantly better than the results provided by WikiNet.

Table 9. Detailed results of our approach and WikiNet compared with WordNet

	Precision	Recall	F-Measure	Class
Our approach	86.12 %	73.54 %	80.48 %	is-a
	85.86 %	92.42 %	89.02 %	not-is-a
WikiNet	69.14 %	78.29 %	73.37 %	is-a
	85.20 %	78.29 %	81.60 %	not-is-a

These results suggest that our approach performs better than approaches based on wikilinks for the English Wikipedia version. However, only 15 % of the whole instances could be evaluated using WordNet. All in all, we could perform an evaluation for 15,483 instances, but 85,938 instances remain unevaluated. There are two reasons for this: first, Wikipedia has a much larger coverage than WordNet and second, many categories in Wikipedia are semi-phrases (e.g. "People in fiction") that cannot be mapped to proper WordNet synsets.

In order to further evaluate the quality of our approach, we performed additional experiments, for example, applying our features not only to links between categories, but also to links between articles and categories. These results are however not presented in this paper, as they simply confirm the results already presented here.

5 Conclusion

Taxonomies are very useful for many NLP applications. However, automatic derivation of taxonomies relies in the most of cases on language-dependent methods or it is based on existing manually created knowledge bases like WordNet or GermaNet. In this work, we used the preliminary results of previous studies to develop a set of features and to train a binary classifier to automatically recognize taxonomic relations between pairs of Wikipedia categories extracted from the category graph.

We describe a robust language-independent Wikipedia-based approach which does not depend on further external sources of knowledge. Eventually, we evaluate the proposed features by measuring the accuracy of the classification of instances for each language. We compare the results with WordNet as ground truth and WikiNet as an approach using no additional external corpora other than Wikipedia. In future work, we plan the evaluation of our approach against approaches relying on additional external corpora other than Wikipedia like YAGO [15]. However, we expect that such approaches perform better than generic approaches as they are optimized with language-specific methods to work in one specific language, e.g. English.

Generally, our approach enables us to automatically derive a taxonomy from Wikipedia for different languages using syntactical and structural features and reducing the dependency on third parties. Further, the research presented here provides a foundation on which further applications and research (e.g. in the field of Wikipedia Mining or attaching semantics to web resources) can be based. This approach can also be used to automatically large-scale evaluation of knowledge bases in languages where manually created knowledge bases do not already exist.

Acknowledgments. This work was supported by funds from the German Federal Ministry of Education and Research under the mark 01 PF 08015 A and from the European Social Fund of the European Union (ESF). The responsibility for the contents of this publication lies with the authors.

References

1. S. Chernov, T. Iofciu, W. Nejdl, and X. Zhou. Extracting Semantics Relationships between Wikipedia Categories. In M. Völkel and S. Schaffert, editors, *Proceedings of the First Workshop on Semantic Wikis – From Wiki To Semantics*, Workshop on Semantic Wikis. ESWC2006, June 2006.
2. G. de Melo and G. Weikum. Towards a universal wordnet by learning from combined evidence. In *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 513–522, New York, NY, USA, 2009.
3. G. de Melo and G. Weikum. MENTA: Inducing Multilingual Taxonomies from Wikipedia. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management*, pages 1099–1108, 2010.
4. R. Domínguez García, C. Rensing, and R. Steinmetz. Automatic acquisition of taxonomies in different languages from multiple wikipedia versions. In *To be published in Proceedings of the 10th International Conference on Web-based Learning (ICWL 2011)*. Springer Link, 2011.
5. R. Domínguez García, P. Scholl, and R. Steinmetz. Supporting resource-based learning on the web using automatically extracted large-scale taxonomies from multiple wikipedia versions. In *Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies*, page 35. ACM, 2011.
6. D. Harman and M. Liberman. Tipster complete. *Corpus number LDC93T3A, Linguistic Data Consortium, Philadelphia*, 1993.
7. L. Kassner, V. Nastase, and M. Strube. Acquiring a Taxonomy from the German Wikipedia. In *Proceedings of the International Conference on Language Resources and Evaluation*. European Language Resources Association, 2008.
8. G. A. Miller. WordNet: A Lexical Database for English. *Communications of the ACM*, 38:39–41, 1995.
9. T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.
10. V. Nastase, M. Strube, B. Boerschinger, C. Zirn, and A. Elghafari. WikiNet: A Very Large Scale Multi-Lingual Concept Network. In *Proceedings of the International Conference on Language Resources and Evaluation*, 2010.
11. R. Navigli and S. P. Ponzetto. BabelNet: Building a Very Large Multilingual Semantic Network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225, 2010.
12. D. P. T. Nguyen, Y. Matsuo, and M. Ishizuka. Subtree Mining for Relation Extraction from Wikipedia. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 125–128, 2007.
13. S. P. Ponzetto and M. Strube. Deriving a Large-Scale Taxonomy from Wikipedia. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 1440–1445. AAAI Press, 2007.
14. J. R. Quinlan. *C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning)*. Morgan Kaufmann, 1 edition, January 1993.
15. F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A Core of Semantic Knowledge. In *16th international World Wide Web conference (WWW 2007)*, New York, NY, USA, 2007. ACM Press.
16. A. Sumida and K. Torisawa. Hacking Wikipedia for Hyponymy Relation Acquisition. In *Proceedings of the International Joint Conference on Natural Language Processing*, 2008.
17. I. H. Witten and E. Frank. *Data mining : practical machine learning tools and techniques*. Elsevier, Morgan Kaufman, Amsterdam [u.a.], 2. ed. edition, 2005.