Enabling Folksonomies for Knowledge Extraction: A Semantic Grounding Approach

Andrés García-Silva[1], Iván Cantador[2], and Oscar Corcho[1]

[1] Ontology Engineering Group,
Facultad de Informática, Universidad Politécnica de Madrid, Spain
{hgarcia,ocorcho}@fi.upm.es,
http://www.oeg-upm.net/
[2] Information Retrieval Group,
Escuela Politécnica Superior, Universidad Autónoma de Madrid, Spain
ivan.cantador@uam.es
http://ir.ii.uam.es/

Abstract. Folksonomies emerge as the result of the free tagging activity of a large number of users over a variety of resources. They can be considered as valuable sources from which it is possible to obtain emerging vocabularies that can be leveraged in knowledge extraction tasks. However, when it comes to understanding the meaning of tags in folksonomies, several problems mainly related to the appearance of synonymous and ambiguous tags arise, specifically in the context of multilinguality. We aim to turn folksonomies into knowledge structures where tag meanings are identified, and relations between them are asserted. For such purpose we use DBpedia as a general knowledge base from which we leverage its multilingual capabilities.

1 Introduction

Social tagging systems are popular Web 2.0 applications that let users to classify and exchange resources (*e.g.*, photos, products, and web pages) by means of manual annotations or tags. Folksonomies are the classification structures that emerge from the aggregation of individual annotations in social tagging systems. The fact that a large user community is annotating resources, often in collaborative environments, makes folksonomies an interesting source for acquiring knowledge. From these rich structures connecting users, tags and resources, it is possible to identify vocabularies that tend to stabilize over time around resources [16] and users [23]. Moreover, the underlying semantics elicited from folksonomies can be characterized by different similarity measures between tags [10, 22], which allow exploiting folksonomies in knowledge acquisition processes at large scale.

Despite such benefits, tags lack explicit semantics [1, 8, 32], and therefore their use as components of knowledge bases (*i.e.*, classes, instances, and data and object properties) is not straightforward. Synonyms, acronyms and spelling variations of a given concept must be identified so that they can be properly represented in a knowledge base, avoiding duplicity of information. Ambiguous tags have to be disambiguated so that they can be added to the knowledge base according to their intended meaning. Moreover, as it happens with the rest of user-generated content, tags are available in multiple languages, and in order to benefit from their multilingual information, a knowledge acquisition process should be aware of the meaning of a tag in its language, and should be able to establish correspondences between equivalent tags written in different languages.

Some approaches [2, 15, 26, 20, 6] tackle the lack of semantics associated with tags by clustering them, in the hope that obtained clusters expose the meanings of the tags. The clusters are created according to certain relations between tags, usually relying on the definition of tag similarity measures [10, 22]. Other approaches [1, 8, 32, 7], on the other hand, address this problem by relating tags to semantic entities in ontologies. Clustering-based approaches have the drawback that the meaning of the relations grouping the tags is not explicitly identified, which hampers the incorporation of the clusters into a knowledge base. Ontology-based approaches strongly depend on the ontology coverage of tags in the folksonomy. A low coverage limits the amount of knowledge that can be added to the knowledge base. Moreover, these approaches are limited to the language in which reference ontologies are written, and currently most of the ontologies are written in English.

Our approach aims to solve the lack of semantics in folksonomies by grounding tags to semantic entities in a knowledge base. We follow the method presented in [18], which addresses the grounding task, *i.e.*, figuring out the intrinsic (or intentional) meaning of symbols. The method associates symbols with taxonomies called "categorical representations", and these categories are used to identify and discriminate symbols. In the case of folksonomy tags, the considered taxonomies must be large enough so that tags can be related to entities in a large extent.

As the reference taxonomy we use DBpedia [4], a general-purpose knowledge base extracted from Wikipedia. The selection of DBpedia has been based on the following strengths: i) DBpedia represents a large source of knowledge, in constant evolution, and agreed by a worldwide community of editors; ii) DBpedia resources, which correspond to Wikipedia articles, can be used as concepts defining meanings of symbols; iii) DBpedia is multilingual and equivalent resources in different languages are related among them; iv) DBpedia is connected to a large number of datasets in the Open Linked Data cloud [3], and therefore we can benefit not only from the DBpedia ontology as a taxonomy but from the ontologies in the interlinked datasets.

In this paper we present Sem4Tags, an approach to perform the semantic grounding of tags to DBpedia resources. Sem4Tags benefits form DBpedia redirection links (*i.e.*, resources created from Wikipedia redirection pages) to deal with different morphological variations of tags referring to the same concept. In case of ambiguous tags, we conduct a disambiguation activity that uses i) the DBpedia disambiguation resources (*i.e.*, resources created from Wikipedia disambiguation pages) to benefit of the human knowledge about candidate meanings for a given tag, and ii) the textual descriptions of DBpedia resources, which are taken from the corresponding Wikipedia articles. More specifically, we transform the disambiguation problem in a retrieval task: an ambiguous tag and its semantic context define a query, and the DBpedia resource associated to the meaning of the tag has to be retrieved from a set of candidate resources. To represent the DBpedia resources we use a bag of words model that is created from textual descriptions of the resources. To implement the retrieval process (*i.e.*, the disambiguation process) we use the vector space retrieval model [29], which is a well-known method used in Information Retrieval to efficiently retrieve documents from large collections.

To evaluate the semantic grounding approach we conducted an experiment with a set of multilingual tags extracted from the online photo sharing site Flickr[3]. We run different versions of Sem4Tags, and ask evaluators to assess the associations between tags and DBpedia resources. We measured the reliability of the assessments using the Fleiss' Kappa statistic [13], and report the results using standard metrics, such as precision and recall.

The remainder of the paper is structured as follows. In Section 2 we present our process for the semantic grounding of multilingual tags. In Section 3 we describe the setup of the experiment we conducted to evaluate our approach. The obtained results as well as the conclusions of the experiment are presented in Section 4. In Section 5 we describe related work. Finally, in Section 6 we discuss future research lines, exemplifying how grounded tags can be leveraged in knowledge acquisition processes.

2 Semantic Grounding of Tags

As discussed in the introduction, our goal is to identify the meaning of a tag (in any natural language) in the context where it is used, by associating it with a resource in DBpedia. We understand by tag context the set of tags that co-occur in the annotation of a resource, even when they are written in different natural languages, and we consider that such context can be used to help on the selection of the correct sense of such tag.

Hence the Sem4Tags system takes as input a tag, its context, and, optionally, the language in which the tag is written, and outputs the corresponding semantic entity. The process followed by the Sem4Tags system, depicted in Figure 1, consists of four stages: *Preprocessing*, *Sense Retrieval*, *Active Context Selection*, and *Sense Disambiguation.* In the Preprocessing stages (see section 2.1) we turn tags into a normalized representation based on DBpedia resource names. We use DBpedia redirection resources to find the main concept a tag refers to. If we could not identify a DBpedia resource name for the tag, we modify it morphologically and use an existing spelling service to find alternative representations of the tag. Next, in the Sense Retrieval stage(see section 2.2) we query DBpedia for resources representing possible senses of the tag. In this activity we use DBpedia disambiguation resources to get the set of candidate resources that may represent

[3] http://www.flickr.com/

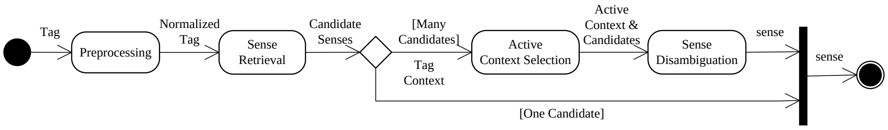


Fig. 1. Semantic grounding process of tags.

the meaning of a tag. If there is only one resource (*i.e.*, the tag is not ambiguous) we select it as the one representing the actual meaning of the tag. On the other hand, if there are more than one resource, we consider the tag as ambiguous. To deal with ambiguous tags we first process the tag context in an Active Context Selection process (see section 2.3), so that we identify the subset of tags in the context that are more related to the ambiguous tag. The authors of [17] have claimed that this subset helps on achieving better disambiguation results. Finally, in a Sense Disambiguation process (see section 2.4), from the obtained resource candidates we attempt to select the one that better describe the tag's meaning.

To deal with multilingual tags Sem4Tags relies on the DBpedia internalization datasets[4]. DBpedia provides different datasets for different languages. Datasets in each language are created from the Wikipedia version in that language, and thus resources are identified by different URIs, defined according to the Wikipedia version from which the resources were extracted. For instance, for New York City there is a New_York_City resource[5] in the English version of DBpedia, and an equivalent Nueva_York resource[6] in the Spanish version. Moreover, internalization datasets contain redirection links and disambiguation resources that are a key part of our approach. In addition, they have links that connect DBpedia resources with the Wikipedia articles from which they were extracted. Henceforth, when we mention DBpedia we refer to the DBpedia dataset corresponding to the language of the tag being processed. Similarly, when we mention DBpedia SPARQL endpoint we refer to the endpoint provided for that language[7].

2.1 Tag Preprocessing

Tags are written without any restriction by users, and thus several slightly modified tags (including misspellings) can refer to the same concept. For instance, *NYC*, *New york*, and *newyork* may refer to New York City. Therefore, our first activity is focused on finding a normalized form of each tag.

[4] See DBpedia internationalization datasets at http://wiki.dbpedia.org/Downloads
[5] http://dbpedia.org/resource/New_York_City
[6] http://es.dbpedia.org/resource/Nueva_York
[7] To see a list of the SPARQL endpoints available in other languages visit http://wiki.dbpedia.org/Internationalization/Chapters

```
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbpo:<http://dbpedia.org/ontology/>

SELECT ?resource WHERE {
  ?redirectResource rdfs:label ?label.
  ?redirectResource dbpo:wikiPageRedirects ?resource
  FILTER(?label="NYC"@en) }
```

Listing 1.1. SPARQL query for identifying the resource pointed by a redirect link.

We use DBpedia resource names as standard names of concepts to which tags can be transformed. First we check whether the tag corresponds to a redirection link in DBpedia, and follow the link to the corresponding resource. In listing 1.1 we show an example SPARQL query where we follow redirections links (dbpo:wikiPageRedirects) to identify the main resource associated with the label NYC. If we pose this query on the DBpedia SPARQL endpoint *http://dbpedia.org/sparql*, we obtain the *New_York_City* resource.

We also modify the tags to turn them into the standard notation of DBpedia resource names, which is based on the Wikipedia title capitalization style[8]. For instance, the *New york* tag is turn into New York. This capitalized version corresponds to the DBpedia resource New_York, which describes the state of New York, and therefore this resource is used as the normalized form of the tag.

Finally, if after the previous modifications we do not find a DBpedia resource name, we use the Yahoo! spelling service[9] to split concatenated words, and detect misspellings. Next, we transform spelling suggestions into valid DBpedia resource names, and check for their existence in DBpedia; If they exist, the tags are considered as normalized, otherwise they are discarded. For instance, for the tag *newyork* the spelling suggestion service splits the word into New York. This suggestion corresponds to the DBpedia resource New_York, and thus it is used as the normalization form of the tag.

In the set of tags that we handle in our evaluation, and which is described later in section 4, from the set of tags for which evaluators were able to identify their meaning and language, our approach was capable of associating 86.9% of tags in English, and 86.7% in Spanish to DBpedia resources. 76.4% of the tags in English and 76.6% in Spanish required modifications to find the corresponding DBPedia resources.

2.2 Sense Retrieval

To select the candidate DBpedia resources that may represent the meaning of a tag, we also query DBpedia through its SPARQL endpoint. Note that DBpedia encodes Wikipedia disambiguation pages, providing candidate senses for ambiguous tags. In this process, we use the normalized form of the tag to see

[8] http://en.wikipedia.org/wiki/Wikipedia:Article_titles

[9] http://developer.yahoo.com/search/boss/

```
PREFIX dbpr:<http://dbpedia.org/resource/>
PREFIX dbpo:<http://dbpedia.org/ontology/>

# Query 1.
ASK {?disamResource dbpo:wikiPageDisambiguates dbpr:New_York}

# Query 2.
SELECT ?candidate WHERE {
  ?disamResource dbpo:wikiPageDisambiguates dbpr:New_York .
  ?disamResource dbpo:wikiPageDisambiguates ?candidate. }
```

Listing 1.2. SPARQL query for retrieving candidate resources for New York.

if the corresponding DBpedia resource is related to a disambiguation resource. If this resource is not related to a disambiguation resource, it is returned as the one representing the meaning of the tag. In listing 1.2 (Query 1) we show the SPARQL query used to evaluate if the DBpedia resource representing the New York state is related or not with a disambiguation resource. We use the ASK operator, which answers true in case there is a triple in DBpedia linking a disambiguation resource (?disamResource) through the *wikiPageDisambiguates* relation with the New_York resource, and false otherwise. If we pose this query on the DBpedia SPARQL endpoint, the result is true. This indicates that New_York is related to a disambiguation resource.

In case the resource is related to a disambiguation resource, then the candidate resources are retrieved, and a disambiguation activity is performed. For instance, in listing 1.2 (Query 2) we show a SPARQL query with which we look for candidate DBpedia resources representing senses of the New York tag. When running this query on the DBpedia SPARQL endpoint we obtain 30 candidate resources.

2.3 Active Context Selection

Traditional disambiguation techniques in Computational Linguistics utilize a wide range of contextual features to address term ambiguities in well-formed sentences of whole texts. Some of these features are part-of-speech labels, collocation information, and surrounding words and sentences [27]. Unfortunately, these features are not available in certain Web-based systems where the context consists of limited, unstructured bags of words, such as those formed by social tags in folksonomies.

We are interested in a tag meaning within a particular annotation, and hence we define the context of a tag as the set of additional tags co-occurring in the annotation. However, there are tags that refer to subjective impressions of users (*e.g.*, *my favourite*, *amazing*) or technical details (*e.g.*, *Nikon*, *photo*), and can be useless (or even harmful) for disambiguation. Therefore, among all the tags in a context, we need to select those that help most on figuring out the target tag's meaning.

To conduct this selection we use a technique described in [17], which relies on the following hypothesis: the most suitable context words for disambiguation are the ones most highly semantically related to the ambiguous keyword. Based on this assumption, we use a simple mechanism to select the active context: After removing repeated words and stop words from the context, we compute the semantic relatedness between each context word and the word to disambiguate. The relatedness computation is performed by using a web-based relatedness measure, similar to the Normalized Google Distance [11], which takes into account the co-occurrence of words on web pages, according to frequency counts, and gives a value between 0 and 1, indicating the degree of semantic relatedness that holds between the compared words. Finally, we construct the active context set with the context words whose relatedness scores are above a certain threshold.

2.4 Sense Disambiguation

The goal of this activity is to select a sense representing the meaning of an ambiguous tag according to the context where it was used. The main idea is that the tag and its context can be compared against each one of the candidate DBpedia resources, by measuring the overlap of the terms in the context with the terms appearing in the textual descriptions of each candidate DBpedia resource. Note that since DBpedia resources correspond to Wikipedia articles, we can use the article text content to obtain the terms to be used in the disambiguation process.

We turn the tag disambiguation activity into a retrieval task that consists of retrieving the DBpedia resource that better represents the tag meaning from the set of candidate resources collected for the tag. Thus, the tag and its context are considered as an input query, and the candidate DBpedia resources as the documents to be retrieved. The result of a retrieval process is a ranked lists of documents. We require that the first document in the ranking represents the DBpedia resource that better describes the tag meaning. To represent the DBpedia resources we use the bag of words model, and to perform the retrieval process we use the well known Vector Space Model [29].

In a bag of word model each document is represented by a set of words, and thus DBpedia resources are represented by means of the words collected from their corresponding Wikipedia articles. To identify the Wikipedia article that describes a DBpedia resource we only need to replace in its URI the DBpedia prefix *http://dbpedia.org/resource* with the Wikpedia prefix *http://en.wikipedia.org/wiki*. We then process the textual content of the Wikipedia article to collect the above words. In this process we get rid of common words that add few value to the retrieval process, by using lists of stop words available for the different languages.

In the vector space model [29] each document is represented as a vector in a multidimensional space. This multidimensional space is defined by the set of words used to represent all the documents. In our case we represent both the tag and the candidate DBpedia resources as vectors. We then compare those vectors using the cosine of the angle as similarity function. The candidate resource whose vector is the most similar with the tag vector is selected as the one representing the meaning of the tag.

The values associated with each dimension in the vectors are calculated according to a weighting scheme. We use term frequency and inverse document frequency (TF-IDF). TF, the term frequency in a document, measures the importance of a term in a document, while IDF indicates whether a term is frequent or rare in the document collection. Note that we are not searching in the whole DBpedia resource collection, but in the more precise set of resources that are suggested by the disambiguation process. TF-IDF is calculated according to equation 1 for the i-th dimension of a document vector. In this equation *tf* is the the frequency of the corresponding word in the document, while *tfmax* is the frequency of the most frequent word in that document. N is the number of documents in the collection (*i.e.*, the DBpedia candidate resources), and n is the number of documents in that collection containing the word.

$$w_i = \frac{tf}{tfmax} * \log(\frac{N}{n}) \tag{1}$$

The vector space model is created as follows. First we create the *Vocabulary* set as the union of the top N frequent terms representing each of the candidate DBpedia resources. Next, for each candidate DBpedia resource we create a vector in $\Re^{|Vocabulary|}$ where each position corresponds to an element in an ordered version of the Vocabulary set. The value w_i associated with the i-th position in the vector is calculated using TF-IDF for the corresponding i-th term in the ordered set.

Similarly, we create a vector for the tag and its context. In this case, w_i takes as value 1 if the i-th term appears in the tag context, and 0 if not. We compare the tag vector and each of the sense vectors using the cosine function as similarity measure, and select the sense vector having the highest similarity with respect to the tag vector. Therefore, we return the resource associated to such sense as the semantic entity to assign to the tag.

Let us suppose we want to ground the tag *New York*, which has been used to annotate a particular photo together with the tags *Central Park, United States, Vacations, Summer, August.* The sense retrieval activity provides 30 candidate DBpedia resources to represent the meaning of that tag. The active context selection activity identifies that *Central Park*, and *United States* are the tags most related with New York, and therefore are considered as the tag context. For each of the 30 candidates we create the bag of words from the corresponding Wikipedia article. We then create the vector space model to represent each candidate as a vector. We also create a vector for the tag and its context. Next, we compare the vector of the tag with each of the vectors of the candidates using the cosine similarity. Some results are shown in Table 1. Note that the New_York_City result is the most similar to the tag vector, and thus it is chosen to represent the tag meaning.

Table 1. Cosine similarity of candidate DBpedia resources and the New York tag

DBpedia resource	Sim.	DBpedia resource	Sim.
New_York_City	0.185	New_York_County	0.117
New_York	0.175	New_York_metropolitan_area	0.032

3 Evaluation Setup

To evaluate our approach we used as test data a set of tagging activities obtained from Flickr. We queried the Flickr API for photos tagged with names of touristic places in Spain (*e.g.*, Barcelona, Canary Island, and Ibiza). We gathered a total of 764 photos uploaded to Flickr by 719 distinct users. On average these 764 photos were annotated using 12.4 tags, with a standard deviation of 7.85. Our data set consists of 9484 tag assignments, TAS (*i.e.*, triples $\langle user, tag, photo \rangle$), where 4153 distinct tags were used.

The evaluation focused on determining the precision of our semantic grounding approach, considering different decisions in the process. First, we were interested in evaluating how well Sem4Tags performs when the keywords representing each sense are the most frequent terms in the content of the Wikipedia articles related to each DBpedia resource, against a reduced set of terms extracted from article abstracts (*i.e.*, the first paragraph describing the article content). Our hypothesis was that large Wikipedia articles may contain as frequent keywords some terms that are not necessarily related to the main subject of an article. In contrast, abstracts could provide more concise information about the article's subject, and thus can lead to better disambiguation results.

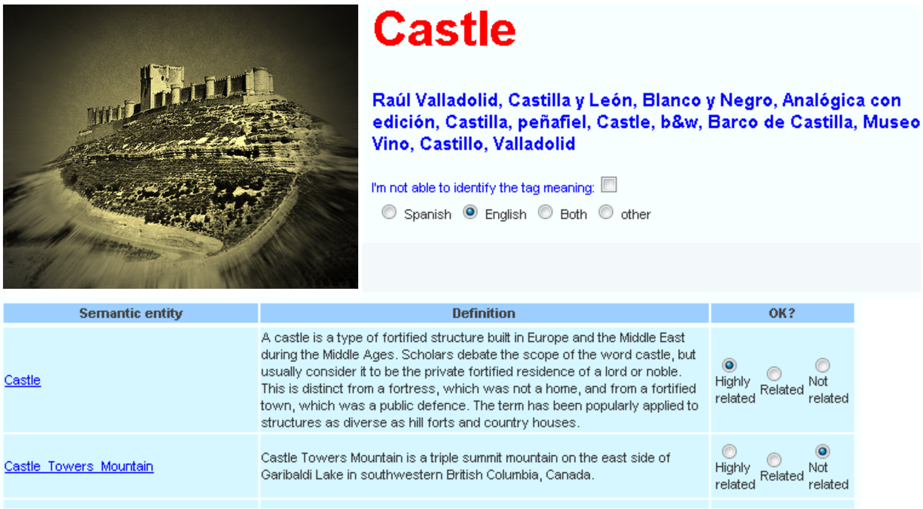
We also considered a baseline that directly relates tags with DBpedia resource names using exact string matching. Note that since DBpedia resource names are taken from the titles of Wikipedia articles, for a given tag, this baseline returns the default sense defined in Wikipedia, *i.e.*, the article that Wikipedia editors have chosen as the most likely meaning for such tag. For instance, in the case of the New York tag, the preferred meaning is the state *http://dbpedia.org/resource/New_York*.

To conclude this section we present a summary of the approaches evaluated for the semantic grounding of tags:

- **Baseline**: Selecting a sense without disambiguation nor preprocessing.
- **Sem4Tags**: Using the whole Wikipedia articles as sources of frequent terms of the senses.
- **Sem4TagsAC**: Conducting the same process as Sem4Tags, but including the selection of the Active Context.
- **Sem4TagsAbs**: Using only the first paragraph of the Wikipedia articles as sources of frequent terms of the senses.
- **Sem4TagsAbsAC**: Conducting the same process as Sem4TagsAC, but including the selection of the Active Context.

Evaluation Campaign We engaged 41 evaluators who had to assess a set of semantic associations[10] generated by each of the considered approaches. Evaluators were presented with 5 semantic entities produced by each approach. As context we provided the photo along with the other tags used to annotate it. We made sure that each semantic association was assessed by at least 3 evaluators, so that we could consider decisions taken by user majority. As we will show in Section 4, there was a significant agreement between the assessments given by the different evaluators about the semantic associations.

For each tagging activity evaluators decided whether they were able to identify the semantics of the tag. Then they had to identify the language of the tagging activity so that they evaluated the semantics associations accordingly. They were presented with the set of DBpedia resources (title and abstract) returned by all the approaches. Then, they were asked to state if each DBpedia resource associated with the tagging activity was highly related (HR), related (R), or not related (N). Note that the evaluation was blind since evaluators did not know from which approaches the semantic entities were coming from, and that the semantic entities were not presented in a predefined order. A screenshot of the evaluation application is shown in Figure 2. In the application the tag to ground is at the top, and the context tags are below in bold.



Castle

Raúl Valladolid, Castilla y León, Blanco y Negro, Analógica con edición, Castilla, peñafiel, Castle, b&w, Barco de Castilla, Museo Vino, Castillo, Valladolid

I'm not able to identify the tag meaning: □

○ Spanish ● English ○ Both ○ other

Semantic entity	Definition	OK?
Castle	A castle is a type of fortified structure built in Europe and the Middle East during the Middle Ages. Scholars debate the scope of the word castle, but usually consider it to be the private fortified residence of a lord or noble. This is distinct from a fortress, which was not a home, and from a fortified town, which was a public defence. The term has been popularly applied to structures as diverse as hill forts and country houses.	● Highly related ○ Related ○ Not related
Castle_Towers_Mountain	Castle Towers Mountain is a triple summit mountain on the east side of Garibaldi Lake in southwestern British Columbia, Canada.	○ Highly related ○ Related ● Not related

Fig. 2. Screenshot of the application in which the users evaluated the grounding of tags.

Metrics We used precision and recall as evaluation metrics. In the conducted experiment, evaluators identified which DBpedia resources were (highly) related

[10] Tuples of the form $\langle user, tag, photo, DBpedia\_resource, language \rangle$

to a given tag within the corresponding semantic context (*i.e.*, for the annotated photo), and the proposed approaches were supposed to retrieve such resources. As already mentioned, for a particular tag, an evaluator was presented with the DBpedia resources retrieved by all the approaches. The evaluator then assessed the resources as related or non related, contributing thus to build a ground-truth dataset. With this dataset, we computed precision and recall values for each of the approaches.

For a given approach and tag, precision is defined as the fraction of DBpedia resources retrieved by the approach that are actually related to the tag. Since our final goal is to provide a single related resource, we compute average precision values taking into account only the first results returned by each approach (*i.e.*, precision at one or $P@1$). For more exhaustive comparisons, we also compute $P@N$, with $N = 2, 3, 4, 5$. We note that in some applications it may be interesting to not only retrieve just one resource for a particular tag, but a (ranked) list of resources. In fact, as shown below, the evaluators stated there were tags with several relevant resources.

Furthermore, we compute the well known Mean Average Precision (MAP) metric, which considers the averages of the precision values at the points at which each relevant resource is retrieved, that is:

$$MAP = \frac{1}{|Tags|} \cdot \sum_{t \in Tags} AveP(t) = \frac{1}{|Tags|} \cdot \sum_{t \in Tags} \left(\sum_{N} \frac{P_t@N \cdot rel_t(N)}{|relevant(t)|} \right) \quad (2)$$

where $Tags$ is the set of evaluated tags, $relevant(t)$ is the set of relevant resources of tag t, $P_t@N$ is the precision at position N obtained by the evaluated approach for tag t, and $rel_t(N) = 1$ if the result at rank position N is a relevant resource for tag t, 0 otherwise.

In turn, recall is defined as the fraction of DBpedia resources related to the tag that are successfully retrieved by the approach. Similarly to precision, we also take into consideration recall at N or $R@N$, with $N = 1, 2, 3, 4, 5$. We note that for the top 5 results, not all the relevant resources for a particular tag may be retrieved. However, as will be shown in the next section, the obtained recall values were close to 100%, which means that taking into account a few top retrieved results, we would be considering almost all the existing relevant resources. Again, this could be exploited in applications where presenting several relevant DBpedia resources for a particular tag is valuable.

Finally, we compute the well known F metric, which is the weighted harmonic mean of precision and recall: $F = 2 \cdot precision \cdot recall/(precision + recall)$. This metric allows selecting a particular approach based on a required or desired balance between its precision and recall.

4 Evaluation and Discussion

We evaluated a total of 2260 tag assignments corresponding to 764 photos tagged with 1112 tags[11]. Evaluators were able to identify the semantics of 87% of the TAS. That is, in 87% of the assessments evaluators stated that they could identify the tag meaning. From this subset, evaluators stated that 62.6% were written in English and 87.7% in Spanish. Statistics about the evaluation are reported in Table 2.

Table 2. Description of the dataset.

	Users	Evaluations	Evaluations/ user	Photos	Tags	TAS	TAS/ photo
English tags	41	30400	741.46 (±206.51)	642	659	1232	1.92 (±0.79)
Spanish tags	41	49568	1208.98 (±152.10)	742	816	1727	2.33 (±0.74)

From the set of tags for which evaluators were able to identify their meaning and language, our process associated the 86.9% of tags in English and the 86.7% in Spanish to DBpedia resources. The preprocessing activity was useful to find DBpedia resource names for the 76.4% of the tags in English and 76.6% in Spanish.

4.1 Precision and Recall Analysis

Table 3 shows the results obtained by the different approaches on tags marked as English and Spanish. For a given tag (*i.e.*, a semantic association photo-tag-resource), based on the relevance assessments provided by three different evaluators, a semantic resource was considered relevant if at least two evaluators stated the resource was *highly related* (or *related/highly related*) to the tag, and non-relevant otherwise. There was a 'substantial' agreement among evaluators, in related and non-related assessments. Fleiss' kappa statistic [14] measuring the agreement among the evaluators' relevance assessments was $\kappa = 0.76$ (a value $\kappa = 1$ means complete agreement, and values higher than 0.60/0.80 are considered as of significant/strong agreement [21]) for the *highly related* case, and $\kappa = 0.71$ for the *related/highly related* case. In the reported results, the former case was used because of its higher agreement level. Similar average performance results were obtained with the latter case. Precision values were higher and recall values were lower. There were more relevant resources so it was easier to accurately retrieve a relevant entity, while it was more difficult to retrieve all relevant resources. We also measured the agreement when identifying the language. There was an 'almost perfect' agreement among users; Fleiss' kappa statistic was $\kappa = 0.83$.

[11] Dataset available in `www.oeg-upm.net/index.php/en/material-used-papers`

Table 3. Evaluation results achieved by the different approaches.

	MAP	P@1	P@2	P@3	P@4	P@5	R@1	R@2	R@3	R@4	R@5	F
English tags												
Baseline	0.78	0.88	-	-	-	-	0.78	-	-	-	-	0.28
Sem4Tags	$\underline{\mathbf{0.91}}$	0.89	0.53	0.37	0.29	0.23	0.81	0.91	0.93	0.95	0.96	$\underline{\mathbf{0.36}}$
Sem4TagsAC	$\underline{\mathbf{0.90}}$	0.90	0.52	0.36	0.28	0.23	***0.82****	0.90	0.92	0.93	0.94	$\underline{\mathbf{0.36}}$
Sem4TagsAbs	***0.84***†	0.82*	0.48	0.34	0.26	0.22	0.75*	0.85	0.89	0.90	0.92	**0.34**
Sem4TagsAbsAC	***0.86***†	0.86	0.48	0.34	0.26	0.22	0.79	0.86	0.89	0.90	0.92	**0.34**
Spanish tags												
Baseline	0.71	0.88	-	-	-	-	0.71	-	-	-	-	0.27
Sem4Tags	$\underline{\mathbf{0.93}}$	**0.93**	0.58	0.42	0.33	0.27	**0.79**	0.90	0.95	0.97	0.98	$\underline{\mathbf{0.41}}$
Sem4TagsAC	$\underline{\mathbf{0.93}}$	**0.94**	0.57	0.42	0.33	0.27	**0.80***	0.89	0.93	0.96	0.96	$\underline{\mathbf{0.40}}$
Sem4TagsAbs	$\underline{\mathbf{0.88}}^\ddagger$	0.90*	0.53	0.39	0.32	0.26	***0.76****	0.85	0.90	0.93	0.94	$\underline{\mathbf{0.39}}$
Sem4TagsAbsAC	$\underline{\mathbf{0.89}}^\ddagger$	0.91*	0.54	0.40	0.32	0.26	***0.77***	0.85	0.90	0.93	0.94	$\underline{\mathbf{0.39}}$

The results shown in the tables were obtained from those tagging activities where the associated semantic entities were known for the evaluators, and in which the corresponding tags were linked to DBpedia resources by at least one approach. Note that *recall* is computed assuming that the set of *all* tags relevant to a given tag is composed by the *relevant* (see definition above) entities retrieved by the investigated approaches. We cannot assure that we are able to retrieve all relevant entities in DBpedia but a strong representative sample of them.

Wilcoxon's statistical tests were performed for MAP, $P@1$, $R@1$ and F-measure to determine whether there were statistical significance differences between the metric values obtained with the baseline and the proposed approaches, and between the metric values obtained with Sem4Tags approach and its variants. The statistical tests were applied on those tagging activities where all approaches (including the baseline) were able to link at least one DBpedia resource. This lets us to present a more fair comparison among approaches, but implies a loss of information that hides a higher statistical evidence in the differences with metric values of approaches able to link DBpedia resources in a large number of cases. In Table 3 values in underline bold (p=0.01), bold (p=0.05), and italic bold (p=0.1) indicate a statistical significance difference with values achieved by the baseline approach. Values marked with ‡(p=0.01), †(p=0.05), and *(p=0.1) indicate a statistical significance difference with values achieved by Sem4Tags approach.

Finally, since the baseline retrieves a single semantic association for each tag, the metrics $P@N$ and $R@N$ with $N = 2, 3, 4, 5$ are not reported for that approach. Indeed, the coverage (recall) of the baseline is low in comparison to the proposed approaches, as shown in the tables. The following conclusions can be drawn from our study:

- In general, the baseline had a good performance with tags in both English and Spanish. This fact suggests that a high percentage of the analyzed tags was used in the sense directly found by the baseline, which corresponds to the Wikipedia default sense. Its high P@1 value is due to the fact that in the 90% of the TAS in English and 91% in Spanish, the correct sense corresponds with the default sense. Nevertheless, the coverage of the baseline, defined as

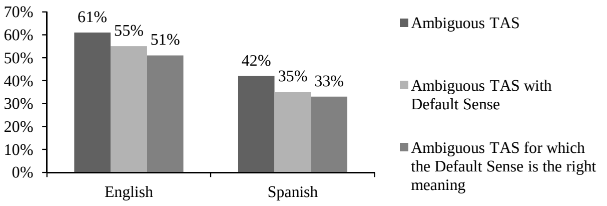


Fig. 3. Ambiguity of tags with relevant results produced by Sem4Tags.

the number of semantic associations produced by the baseline divided by the total number of TAS is extremely low: 27.7% in English and 19.4% in Spanish. This contrasts with the 79.1% of Sem4Tags coverage in English and 81.4% in Spanish. This difference in coverage is due to the preprocessing activity. Note that in the disambiguation of words in text documents the baseline, defined as the most frequent sense for a word, also achieves high precision [28].

– Sem4Tags and its variants perform better when dealing with Spanish tags. The amount of information in the Spanish Wikipedia compared with the English version is considerably lower[12], and this difference is reflected in the corresponding versions of DBpedia. Less articles in the Spanish version may indicate less ambiguity in the sense that not all the possible meanings of a word have been added to the Wikipedia. In fact, the average of senses was 23.3 for English and 10.35 for Spanish. As shown in Figure 3 there were less tags in Spanish considered ambiguous (42%) than in English (61%), and thus the grounding was straightforward for more tags in Spanish (58% of non ambiguous tags) than for tags in English (39% of non ambiguous tags).
– Sem4Tags and Sem4TagsAC were the approaches that obtained the best results both in terms of precision and recall. Almost all of these results present statistical significant differences with the results obtained by the baseline. Comparing Sem4Tags and Sem4TagsAC, we do not find a clear enhancement of semantic associations when exploiting the active context. In some cases, it seems that Sem4TagsAC obtains better P@1 and R@1 values, but the improvements are supported by no or low statistically evidence. This observation could be biased by the way in which statistical tests were conducted, as explained before.
– Sem4TagsAbs and Sem4TagsAbsAC are the worst approaches. Abstract terms do not provide enough information to properly disambiguate tag meanings. That is, the scarcity of terms in the abstract decreases the overlapping of these terms with tags in the context.

[12] As of April 2012, the English and Spanish Wikipedia have 3,921,259 and 882,859 articles respectively

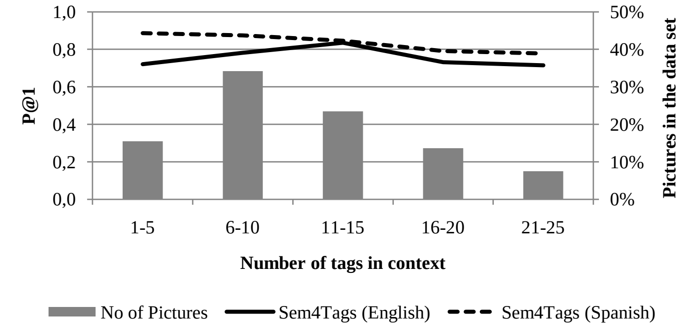


Fig. 4. Precision variation according to the context length.

- In the 17% and 20% of ambiguous tags in English and in Spanish respectively, the correct sense was different from the Wikipedia default sense. This evidences the need of performing a disambiguation activity.
- While Sem4Tags precision is related to the number of tags in the context, it presents different patterns for tags in English and Spanish (see Figure 4). In the case of tags in English, photos annotated with between 6 and 15 tags (representing 58% of the total) produce the highest P@1 reaching a peak for context containing between 11 and 15 tags. Short contexts with less than 5 tags, or long contexts with more than 16 tags produce, though satisfactory, lower P@1 values around 71%. Short contexts do not provide enough evidence (*i.e.*, words to measure the overlapping with words in the candidate senses) to select the right sense for a tag. In contrast, long contexts are noisy in the sense that some words in the context can indicate one possible meaning while other words point to other meaning. In case of tags in Spanish, short contexts do not affect the precision of the Sem4Tags approach. In fact, the highest precision is achieved when context length range between 0 an 10. Nevertheless, by starting from 15 context tags, precision decreases along with the context length until it stabilizes around 78%.

The exploitation of the active context of a tag seems to improve the performance of our approach. Nonetheless, we did not obtain statistically significant evidence to support that claim. Additional evaluations focused on measuring the importance of semantic context have to be done. Based on the satisfactory results achieved by Sem4Tags, and its simple extensions, we plan to conduct experiments with other languages so that we can analyze how distinct language characteristics affect our semantic grounding approach.

5 Related Work

The success of folksonomies as classification systems drew the attention of the research community. Early work showed that the aggregation of unrestricted individual annotations leads to the emergence of vocabularies around resources [16] and users [23], and thus the potential of folksonomies as sources of knowledge was confirmed. The range of advantages attributed to folksonomies includes the fact that the above vocabularies, *i.e.*, the most frequent tags, reflect the point of view of a large user community, while less frequent tags are left in the long tail.

Researchers have proposed different approaches to identify the underlying semantics of folkonomies. They can be classified into three types: 1) clustering approaches [2, 26, 20, 6], which aim to find clusters of tags relying on relatedness measures between them, 2) ontology-based approaches [1, 32, 7], whose goal is to associate tags with ontology entities, and 3) hybrid approaches [15], which use a combination of clustering techniques and ontologies.

The semantics that can be elicited from folksonomies depend on the similarity metric used to relate tags [10, 22]. Approaches that rely on these measures [2, 26, 20, 15] select the similarity metrics arbitrarily. Unlike these approaches, our research work aims at resolving the lack of semantics by grounding tags with semantic entities. Differently to approaches like [10, 1], which respectively use WordNet and ontologies retrieved by the Watson system[13], we exploit DBpedia, and correspondingly Wikipedia, as a multilingual general-purpose knowledge base. Furthermore, in contrast to [32], where the authors assume that users use each tag in a single sense, our approach supposes that a user can use a tag in several senses.

The exploitation of Wikipedia as a valuable source of semantic knowledge has been widely investigated in the literature. Medelyan et al. present in [24] an extensive and comprehensive survey of research work on extracting and making use of the concepts, relations, facts, and descriptions found in Wikipedia. The authors analyze the elements of Wikipedia that have been utilized to extract semantic knowledge, namely the article titles, text contents, categories, links and their anchor texts, infoboxes and templates, disambiguation pages, and redirection links.

In the context of extracting candidate senses or concepts in different forms by using Wikipedia elements, we can bring up the following representative contributions. In [30] Schonhofen presents a simple technique that relates terms in a document to Wikipedia entities based on the titles and categories of the entities' articles. Cantador et al. [7] also use the Wikipedia article titles to map social tags with entities. In this case, the YAGO ontology [31] is then used to assign the mapped entities to content- and context-based upper level classes. Bunescu and Pasca [5] focus on exploiting the Wikipedia's category taxonomy by a Machine Learning model for disambiguating named entities. Coursey et al. [12], on the other hand, collect and distinguish senses from Wikipedia links and their associated anchor texts in the articles. Ruiz et al. [9] analyze the Wikipedia text contents to identify lexical patterns that are used to extract new semantic relations between Wikipedia entities. The types of the entities related by each of the patterns are then used to disambiguate senses. Finally, Medelyan et al.

[13] http://watson.kmi.open.ac.uk/WatsonWUI/

[25] propose to take advantage of disambiguation pages and redirection links to select candidate senses and alternative labels respectively.

Our approach to ground tags to DBpedia resources taps into the textual description of the resources in Wikipedia. Similarly to Medelyan et al. [25] we use as candidate senses for an ambiguous term information taken from disambiguation pages as well as redirection links for alternative labels. However, in contrast to this approach ours is unsupervised. Furthermore, we have explored different variations of our approach where we change the amount of textual information consumed in the process as well as the number of information in the tag context.

6 Future Work and Discussion

In this paper we have described a system (Sem4Tags) that represents a step forward to our more general goal of implementing a knowledge extraction system that leverages tags and their semantic grounding, which we illustrate in the following scenario in the computer programming domain.

Let us assume that some domain experts have defined a list of web pages considered as prominent resources in the domain of computer programming, and have annotated them with tags like (*Web 2.0*, 1250), (*programmer*, 1173), (*todo*, 150), (*software engineer*, 900), (*mashups*, 700), etc. The number accompanying each tag is the annotation frequency. If we use tag co-occurrence when annotating resources as a tag similarity measure the previous tags could be considered as related.

We focus on two particular tags: *programmer* and *software engineer*. Programmer is an ambiguous tag according to DBpedia[14] since it may refer to a hardware programmer or to a computer programmer. Using as context the co-occurring tags our approach grounds the tag to the DBpedia entity *dbpr:Programmer*[15], which represents the computer programmer meaning. On the other hand, software engineer is not ambiguous and thus it is straightforward grounded to *dbpr:Software_engineer*. Despite the large coverage of the DBpedia ontology, which classifies around 1.83M instances, these two entities are not classified under any class. To avoid this limitation we can use the interlinked data sets to see if these entities make part of another ontology from which we can obtain ontological knowledge.

Figure 5 depicts linked data related to *dbpr:Programmer*. By browsing the *owl:sameAs* links we realize that *dbpr:Programmer* is equivalent to the classes *Development Program* and *Developer* in the OpenCyc ontology[16], and to the class *ComputerProgrammer* in the UMBEL ontology[17]. Development Program refers to software that is used to create other software, Developer refers to a computer programmer, and ComputerProgrammer refers to a person who develop

[14] http://dbpedia.org/page/Programmer_%28disambiguation%29

[15] The prefix dbpr stands for http://dbpedia.org/resource/

[16] OpenCyc homepage: http://opencyc.org/

[17] UMBEL homepage: http://umbel.org/

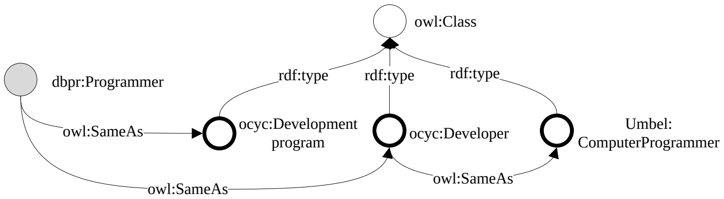


Fig. 5. Using ontologies published as linked data to gather additional semantic information

computer programs. Note that *owl:sameAs* relations in linked data are manually or automatically created, and therefore these links may be mistaken. This could be the case of the link established between Programmer and Development Program, since unless an instance of the latter creates a program automatically it should not be equivalent to a programmer. Nevertheless the three classes are relevant to the domain of study, and thus can be leveraged by the knowledge acquisition process.

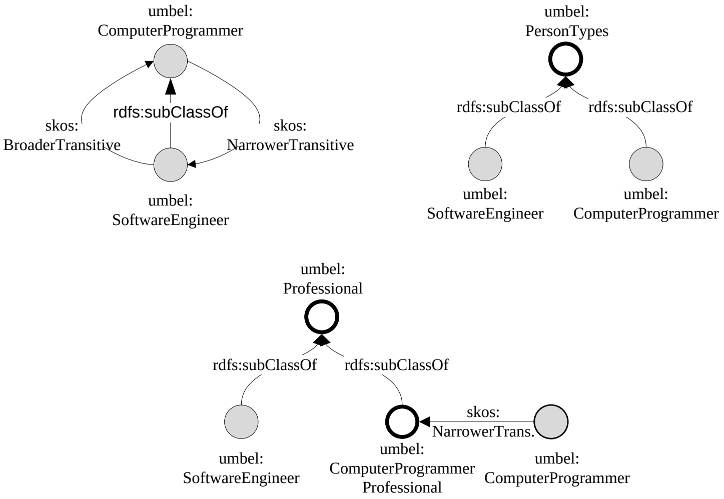


Fig. 6. Discovering relations in the linked data set. Bold edge nodes represent new classes discovered in the process.

On the other hand, the class *dbpr:Software_Engineer* is linked to the class *SoftwareEngineer* in the UMBEL ontology. Thus, we can tap into the UMBEL ontology to discover relations between the classes SoftwareEngineer and ComputerProgrammer. Figure 6 depicts distinct relations between these two classes that are formalized in the UMBEL ontology. Note that these relations can be identified by means of SPARQL queries, which browse the different paths that can be established between classes [19]. The UMBEL ontology states that: i) SoftwareEngineer is *rdfs:subClassOf* CumputerProgrammer, ii) both classes are *rdfs:subClassOf PersonType*, and iii) *ComputerProgrammerProfessional* and SoftwareEngineer are *rdfs:subClassOf Professional*, and ComputerProgrammer-Professional is a *narrowerTerm* of ComputerProgrammer. All these relations between the classes SoftwareEngineer and CumputerProgrammer, as well as the classes PersonType, Professional, and ComputerProgrammerProfessional, are domain relevant knowledge that may be captured in the knowledge acquisition process.

References

1. S. Angeletou, M. Sabou, and E. Motta. Semantically enriching folksonomies with flor. In *In Proceedings of the ESWC'09 Workshop on Collective Intelligence and the Semantic Web*, 2008.
2. G. Begelman, P. Keller, and F. Smadja. Automated tag clustering: Improving search and exploration in the tag space. In *Proceedings of the WWW'06 Collaborative Web Tagging Workshop*, Edinburgh, Scotland, 2006.
3. C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 2009.
4. C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. DBpedia - A crystallization point for the Web of Data. *Journal of Web Semantic*, 7(3):154–165, 2009.
5. R. Bunescu and M. Pasca. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006), Trento, Italy*, pages 9–16, 2006.
6. I. Cantador, A. Bellogín, I. Fernández-Tobías, and S. López-Hernández. Semantic contextualisation of social tag-based profiles and item recommendations. In *Proceedings of the 12th International Conference on E-Commerce and Web Technologies (EC-Web 2011), Tolouse, France*, pages 101–113, 2011.
7. I. Cantador, I. Konstas, and J. M. Jose. Categorising social tags to improve folksonomy-based recommendations. *Web Semantics*, 9(1):1–15, 2011.
8. I. Cantador, M. Szomszor, H. Alani, M. Fernández, and P. Castells. Enriching ontological user profiles with tagging history for multi-domain recommendations. In *Proceedings of the 1st International Workshop on Collective Semantics (CISWeb 2008)*, June 2008.
9. M. R. Casado, E. Alfonseca, and P. Castells. From wikipedia to semantic relationships: a semi-automated annotation approach. In *Proceedings of the 1st Workshop on Semantic Wikis: From Wiki to Semantics, Budva, Montenegro*, 2006.
10. C. Cattuto, D. Benz, A. Hotho, and G. Stumme. Semantic grounding of tag relatedness in social bookmarking systems. In *The Semantic Web - ISWC 2008*, volume 5318 of *Lecture Notes in Computer Science*, pages 615–631, 2008.
11. R. Cilibrasi and P. M. B. Vitanyi. Automatic meaning discovery using google, December 2004.
12. K. Coursey, R. Mihalcea, and W. Moen. Using encyclopedic knowledge for automatic topic identification. In *Proceedings of the 13th Conference on Computational*

Natural Language Learning (CoNLL 2009), Boulder, CO, USA, pages 210–218, 2009.

13. J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378 – 382, 1971.
14. J. L. Fleiss and J. Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619, 1973.
15. E. Giannakidou, V. Koutsonikola, A. Vakali, and Y. Kompatsiaris. Co-clustering tags and social data sources. In *Proceedings of the 9th International Conference on Web-Age Information Management (WAIM 2008)*, pages 317–324. IEEE Computer Society, 2008.
16. S. A. Golder and B. A. Huberman. Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, 2006.
17. J. Gracia and E. Mena. Multiontology Semantic Disambiguation in Unstructured Web Contexts. In *Proceedings of the the Workshop on Collective Knowledge Capturing and Representation (CKCaR 2009), Redondo Beach, CA, USA*, Sept. 2009.
18. S. Harnad. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3):335–346, June 1990.
19. P. Heim, S. Lohmann, and T. Stegemann. Interactive relationship discovery via the semantic web. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC 2010)*, volume 6088 of *LNCS*, pages 303–317, Berlin/Heidelberg, 2010. Springer.
20. R. Jaschke, A. Hotho, C. Schmitz, B. Ganter, and G. Stumme. Discovering shared conceptualizations in folksonomies. *Web Semantics Science Services and Agents on the World Wide Web*, 6(1):38–53, 2008.
21. J. Landis. Measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977.
22. B. Markines, C. Cattuto, F. Menczer, D. Benz, A. Hotho, and G. Stumme. Evaluating similarity measures for emergent semantics of social tagging. In *Proceedings of the 18th International World Wide Web Conference (WWW 2009)*, pages 641–641, April 2009.
23. C. Marlow, M. Naaman, D. Boyd, and M. Davis. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *Proceedings of the of the 17th Conference on Hypertext and Hypermedia (Hypertext 2006), Odense, Denmark*, pages 31–40, New York, NY, USA, 2006. ACM.
24. O. Medelyan, D. Milne, C. Legg, and I. H. Witten. Mining meaning from wikipedia. *International Journal of Human-Computer Studies*, 67(9):716–754, 2009.
25. O. Medelyan, I. H. Witten, and D. Milne. Topic indexing with wikipedia. In *In Proceedings of the Proceedings of the 1st AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI 2008), Chicago, IL, USA*, 2006.
26. P. Mika. Ontologies are us: A unified model of social networks and semantics. *Journal of Web Semantics*, 5(1):5–15, 2007.
27. R. Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):1–69, 2009.
28. R. Navigli, K. C. Litkowski, and O. Hargraves. Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval 2007)*, pages 30–35, Prague, Czech Republic, June 2007. Association for Computational Linguistics.
29. G. Salton and M. J. Mcgill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

30. P. Schonhofen. Identifying document topics using the wikipedia category network. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006), Hong Kong, China*, pages 456–462, 2006.
31. F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A large ontology from wikipedia and wordnet. *Web Semantics*, 6(3):203–217, 2008.
32. M. Tesconi, F. Ronzano, A. Marchetti, and S. Minutoli. Semantify del.icio.us: Automatically turn your tags into senses. In *Proceedings of the 1st Workshop on Social Data on the Web (SDoW 2008)*, 2008.