Journal of King Saud University – Computer and Information Sciences 31 (2019) 15–21

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Bridging the Gap between the Social and Semantic Web: Extracting domain-specific ontology from folksonomy

Mohammed Alruqimi *, Noura Aknin

Information Technology and Modeling Systems Research Unit, Abdelmalek Essaadi University, Morocco

ARTICLE INFO

Article history:
Received 10 July 2017
Revised 27 September 2017
Accepted 19 October 2017
Available online 20 October 2017

ABSTRACT

Folksonomies have become very popular as means to organize large sets of resources shared over the Social Web. The bottom-up nature of folksonomies has proved to be an interesting alternative to the current effort at semantic web ontologies since folksonomies provide a rich terminology generated by large user-communities. Besides, ontologies extracted from folksonomies can represent the intelligence collective of social communities. Such ontologies also represent a core element of a new feature of the Web, the Internet of Things. Many research studies have captured semantics in folksonomies, some of which have developed ontologies from folksonomy. However, the formal specific-domain ontology consisting of domain-dependent relations has not been researched yet. This paper introduces an algorithm for deriving a domain-specific ontology from folksonomy tags. The proposed algorithm starts by collecting a domain-specific terminology; next, discovering a pre-defined set of conceptual relationships among the domain terminologies. The evaluation of the algorithm, using a dataset extracted from BibSonomy, demonstrated that the algorithm could effectively learn domain ontologies consisting of domain concepts linked by meaningful and high accurate relationships. Furthermore, the proposed algorithm can help reduce common issues related to tag ambiguity and synonymous tags.

1. Introduction

Social tagging websites (e.g. www.del.icio.us/, www.bibsonomy.org/, www.flickr.com/, etc.) offer users open platform to describe their content using their own vocabularies, forming the so-called folksonomies (Vander Wal, 2007). From a knowledge organization point of view, folksonomies have two main advantages: folksonomies provide a vast amount number of user-generated annotations and directly reflect users' vocabularies and interests; they are relatively cheap to develop and harvest as they emerge from end users' tagging (Hotho et al. 2006; Szomszor et al. 2007; Mathes 2004). These advantages have turned folksonomies into an interesting alternative to current efforts at semantic web ontologies (Al-Khalifa and Davis 2007; Szomszor et al. 2007; Gruber 2007; Shirky 2005; Lee, 2015). On the other hand, the fact that ontology users are not involved in ontology construction contributes significantly to the current dearth of satisfying coverage in ontologies (Van Damme et al., 2007). Therefore, automatically developing an initial version of an ontology from user-generated systems reduces the high cost associated with using ontology engineers to develop domain ontologies from scratch. Though many approaches to the explicit semantics behind social tags, as described in Section 2, have been proposed, the results remain limited. Most of these approaches focus on discovering related tags rather than building ontologies, much less domain-specific ones (García-Silva et al., 2012). In addition, these approaches either do not define the nature of relations between tags or derive only limited kinds of relations (often taxonomic or more general) without considering the kind of conceptual relationships that should be modeled in particular domains (Trabelsi, et al., 2010). This paper proposed an algorithm to elicit domain-specific ontology from folksonomy. Experimental results, on real word data available in BibSonomy, demonstrated that the proposed algorithm could effectively learn a domain terminology. As well, discover a set of pre-defined meaningful relationships among the domain terminologies. The remainder of the paper is organized as follows: In Section 2, a selection of related work is outlined. Next, in Section 3, the dataset used in the experiments described. Section 4, the proposed algorithm is presented, then, in Section 5, extensive

* Corresponding author.
E-mail addresses: m.alruqimi@uae.ma (M. Alruqimi), aknin@ieee.org (N. Aknin).
Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2017.10.005
1319-1578/

discussion and evaluation of the proposed algorithm. Section 6, conclusions.

2. Related work

Much work has been done to introduce semantics in folksonomy (García-Silva et al., 2012; Alruqimi and Aknin, 2015; Jabeen et al., 2016), and to investigate methods of deploying this semantics for tasks such as information retrieval (Uddin et al., 2013; Zubiaga et al., 2013; Tommasel and Godoy, 2015), recommender systems (Cantador et al., 2011; Ching Hsu, 2013; Font et al., 2015), and ontologies development (Hamdi et al., 2012; García-Silva et al., 2014; Wang et al., 2015). Generally, Semantics behind folksonomies studies fall into three broad categories: statistical analysis approaches, knowledge-based approaches and hybrid approaches. The early studies explored means of leveraging the co-occurrence statistics of tags and the tripartite structure of folksonomies to measure tag relatedness (e.g., Begelman et al., 2006; Schmitz, 2006; Heymann and Garcia-Molina, 2006; Mika, 2007). This category of approaches focused on grouping similar tags and building their hierarchies without providing methods for defining the exact meaning of tags and their relations. More recently researchers (e.g., García-Silva et al., 2014; Tesconi et al., 2008; Cantador et al., 2008; Angeletou, 2008) proposed to make tags semantics explicit by grounding them to corresponding entries in online knowledge bases, such as thesaurus and ontologies. In this context, WordNet and DBpedia are widely used as semantic knowledge sources for mapping tags. Although these approaches are more precision, but approaches heavily dependent on WordNet get poor recall due to the fact that many of the tags from folksonomies do not exist in WordNet. Lately, DBpedia has grabbed more attention in this respect. DBpedia has considerably larger coverage than WordNet. However, it provides more specific relationships (e.g., "*was born in*", "*lives in*", etc.) limited to entities with proper names. Another type of approaches on the topic (e.g., Specia and Motta, 2007) was proposed using a combination of the two abovementioned approaches. In general, there is a lack of methods that extract domain-specific ontologies from folksonomies. Besides existent approaches only identify limited kinds of relations that are either taxonomic relations (i.e. *has*, *partOf*, *subClassOf*) or more general relations (i.e. *sameAs*, *isa*). Based on the above evaluation, our algorithm produces baseline domain ontologies from tags in folksonomies. The proposed algorithm collects a domain terminology from tags relying on a set of domain keywords extracted automatically from Wikipedia pages titles. Then, it discovers pre-defined relations with more specific senses in a given domain.

3. Dataset description

The data selected for the experiments is a snapshot of BibSonomy (Benz et al., 2010), Knowledge & Data Engineering Group, 2008, which is available on the BibSonomy site in the form of "DAT" files (http://www.kde.cs.uni-assel.de/bibsonomy/dumps/#-datasets). BibSonomy is a web-based social bookmarking and collaborative tagging system which enables the storage, tagging, sharing, and retrieval of bookmarks and publications. It is online since 2006 and is actively used by several thousand users. In fact, users utilize folksonomies with various intentions. For instance, Delicious is used for general purpose whereas BibSonomy primarily serves academic and scientific interests. Compared to general folksonomy, academic folksonomy has a more complex nature in terms of semantics and sparsity of the data (Du et al., 2009; Lee, 2015; Dong et al., 2017). Therefore, they would be more useful for building ontologies (particularly, ontologies for scientific domains). However, several pre-processing steps have been conducted on the whole dataset before randomly selecting 20,000 resources assigned with 85,006 tags (11,865 unique tags), as shown in Table 1.

Table 1
Dataset.

Dataset	Resources	Tags	Unique Tags
BibSonomy	20,000	85,006	11,865

4. The proposed algorithm

The proposed algorithm takes a domain name as input and produces the corresponding domain ontology as output. This algorithm first represents folksonomy resources as an undirected weighted graph. Next, it collects a domain terminology through traversing the resources graph relying on a set of domain keywords extracted automatically from titles of Wikipedia entries. Finally, we extract semantics information about the collected domain terminology by linking it to corresponding Wikipedia entries. This includes identifying the meaning, attributes and synonyms of the domain terminology as well as discovering semantic relationships among the domain terminologies. The general method, as shown in Fig. 1, is performed through two main phases: domain terminology extraction; and concept/relation identification.

4.1. Domain terminology extraction

Typically, the process of building a domain ontology starts by collecting the domain terminology (domain-relevant terms). Therefore, the goal of this phase is to collect a domain terminology from tags. To this end, we developed a Java tool, which we called TermRank. This tool takes the name of a specific domain and a prepared folksonomy dataset as inputs and returns a list of domain-relevant terms. This phase is conducted in three sub-main activities: 1) pre-processing, 2) generating the resources graph and 3) collecting the domain terminology.

4.1.1. Pre-processing

In this activity, many proceedings were performed on the dataset to clean it. These processes include removing the "imported" tag which repeats in a large number of records, and deleting special characters, duplicated records and tags in the same record and prepositions. Furthermore, we used a lexical vector to exclude non-objective tags that caused noisy connections between the resources on the resources graph (Alruqimi and Aknin, 2015; Cantador et al., 2011).

4.1.2. Resources graph generation

A folksonomy can be seen as a tuple $A: = (U, T, R)$, where U, T, and R, are finite sets, whose elements are called users, tags and resources, respectively. Folksonomy can be represented as an undirected tri-partite hyper-graph $G = (V, E)$ where $V = U \cup T \cup R$, is the set of vertices and $E = \{(u, t, r) | (u, t, r) \in A\}$ is the set of edges; the tri-partite graph can be folded into two and one-mode graphs (Mika, 2007). In this work, we built the one-mode graph $G' = (V', E')$ in which V' represents the set of resources, and E' represents the set of weighted edges where two resources (r_i, r_j) will be connected by an edge if they have at least one common tag assigned to both resources (Fig. 2). Formally, $E' = \{(r_i, r_j) | \exists ((u, t_m, r_i) \in A \wedge (u, t_n, r_j) \in A \wedge t_m = t_n)\}$. The number of common tags between r_i and r_j represents the weight of the edge $w(r_i,r_j)$. Formally: $w(ri, rj) = |\{t \in T | (u, ri, t) \in A\} \cap \{t \in T | (u, rj, t) \in A\}|$ (García-Silva et al., 2014). Vertices of the graph have, as attributes, the list of their assigned tags. In the following section, we describe how to traverse this graph in order to collect the relevant domain terms.

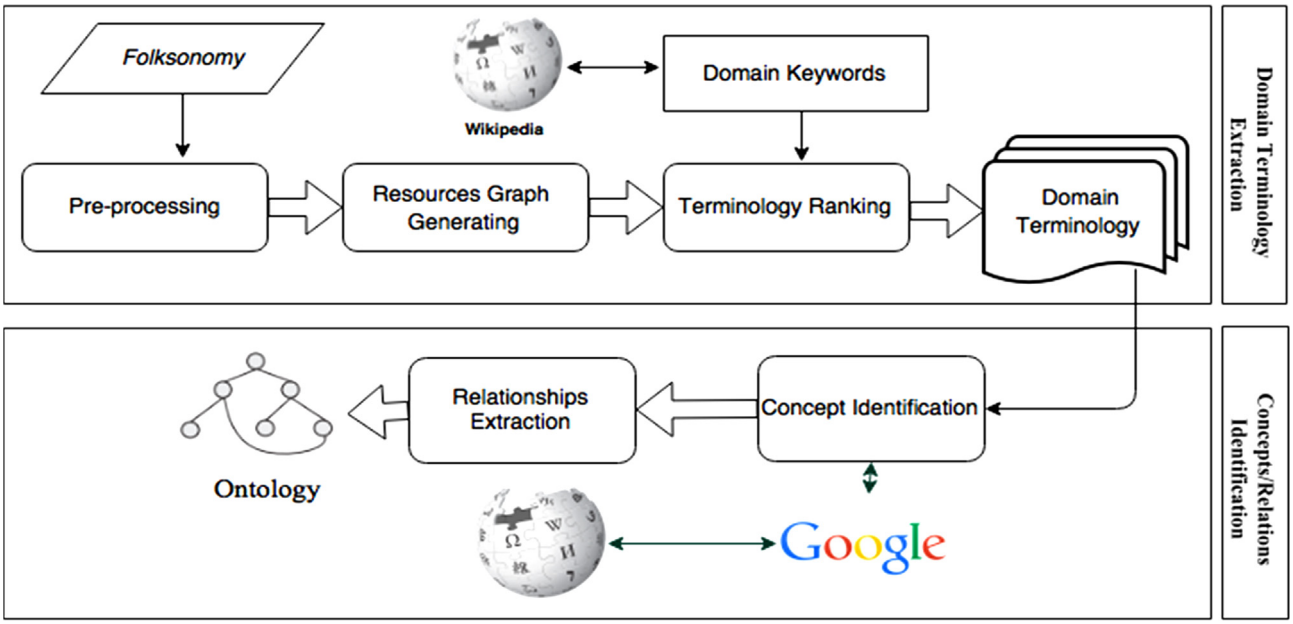

Fig. 1. The structure of the proposed algorithm.

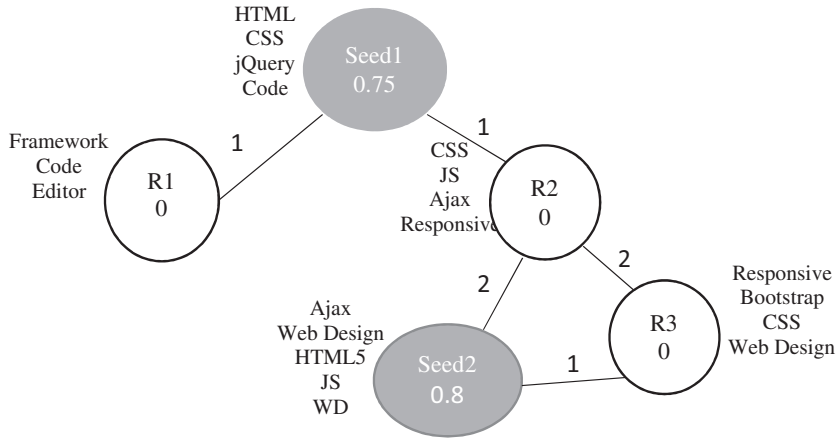

Fig. 2. An example of a resources graph.

To implement this phase, we use JGraphT library, which is a free Java class library that provides mathematical gra ph-theory objects and algorithms (http://jgrapht.org/).

4.1.3. Collecting domain terminology

By traversing the resources graph *G'*, we look for resources that are relevant to a given domain, and then collect the tags assigned to these resources as domain terminologies. Our method requires set of *domain keywords* as references to select set of resources as starting points (seeds) to traverse the graph *G'*. In more details, Implementation this activity goes throughout three sequential steps as follows: **Firstly**, we extracted a set of *Domain Keywords* from titles of Wikipedia articles and redirection pages contained in the main Wikipedia category corresponded to the given domain. **Secondly**, a set of resources, that are considered highly belong to the given domain, was selected as seeds. To select a resource as a seed, at least two-thirds of the tags assigned to this resource should be approximately matched to the D*omain Keywords.* See Eq. (1); Let us consider *K* is the set of domain keywords.

$$a(ri) = \frac{|\{t \in T|(u, ri, t) \in A\}| \cap |\{K\}|}{|\{t \in T|(u, ri, t) \in A\}|} \quad (1)$$

Finally, we traverse the graph *G'* starting from the selected *seeds*. Throughout the traversing process, we applied a ranking function over each visited vertex. The ranking function rates the relevance of a vertex to the given domain based on the number and weight of the paths coming from the different seeds to it (See Eq. (2) adapted from (García-Silva et al., 2014)). Resources that have a ranking value greater than a defined *h* threshold have been marked as domain-relevant resources, and hence all their associated tags have been gathered *rj* as domain-relevant terms. To traverse the graph, we use the breadth first search (BFS) method; once the graph being traversed starting from a particular seed, the traversing process stops whether reaching another seed or reaching a terminal vertex.

$$a(rj) = a'(rj) + \frac{|\{t \in T|(u, rj, t) \in A\} \cap \{t \in T|(u, ri, t) \in A\}|}{|\{t \in T|(u, rj, t) \in A\}|} * \frac{a(ri)}{d} \quad (2)$$

Let us consider ri is the previously visited vertex from which we reached, *d* is the distance between the current vertex and seed.

Procedure 1 *Generating domain keywords*

1: domainKeywords ← ExtractDomainKeywords(*Wikipedia*, *domain_name*). 2: for each vertex in G' do. 3: simValue ← compareTagsToKeywords(vertex, keyword). 4: if(simValue > threshold) Then. 5: setVertexSeed(vertex, "yes"). 6: addVertexToSeedList(*seedList*, *vertex*). 7: end if. 8: end for. 9: Function ExtractDomainKeywords (*Wikipedia*, *domain_name*). 10: catgoriesTitles ← getCategories (*Wikipedia*, *domain_name*). 11: articlesTitles ← getArticlesOfMainCat (*Wikipedia*, *domain_name*). 12: redirectPagesTitles ← getPageRedirects (*articles*). 13: domainKeywords ← catgoriesTitles ∪ articlesTitles ∪ redirectPagesTitles. 14: return domainKeywords. 15: End Function.

Procedure 2 *Collecting domain-relevant terms*

1: domainTerms ← array[ ]. 2: for each seed in seedList do. 3: vertexList ← BreadthFirstSearch (*G'*, *seed*). 4: add(domainTermList, vertexList). 5: end for. 6: domainTerms ← getUniqeTerms(*domainTermList*). 7: *Function* BreadthFirstSearch (*G'*, *seed*). 8: q ← traversedVerteices[ ]. 9: while(q is not empty and vertex is not seed) do. 10: v_j ← next(q). 11: vi ← previousVertx(). 12: rank ← doRank(vj, vi, *seed*). 13: if(rank > threshold) Then. 14: setState (vj, "true"). 15: add (RelevantTerms, vj). 16: end while. 17: return RelevantTerms. 18: End Function. 19: Function doRank (*vj*, *vi*, *seed*). 20: rank ← getRank(vi) +|w(vi ∩ vj)| / | vi | ∗ getRank(vj) / pathBetween(*seed*, *vj*). 21: return rank. 22: End Function.

4.2. Concepts and relations identification

This task is primarily referring to the process of defining the intended meaning of the extracted terms and discovering relations among them. It also includes disambiguating terms and extracting semantic information about them as well. To perform this task, we proposed to use Google as an intermediary to match the tags to their appropriate Wikipedia entries. By correlating tags to Wikipedia entries, we identify the exact meaning of terms and retrieve semantic information about each correlated term, including its category and its alternative names. Then, we discover relations among terms based on texts of the Wikipedia entries correlated to those terms. The advantage of using Wikipedia as a reference to map terms is that Wikipedia is a community-driven knowledge base, much like folksonomies are, so that it rapidly adapts to accommodate new terminology. Many of the popular tags occurring in folksonomies do not appear in grammar dictionaries, such as WordNet,

Fig. 4. Using Google as an intermediary to retrieve Wikipedia articles that represents terms meaning.

because they correspond to proper nouns, modern technical words, or are widely used acronyms. In addition, the redirect pages in Wikipedia provide synonyms and morphological variations for a concept. For example, when searching the tag '*nyc*' in Wikipedia, the entry for New York City is returned.

4.2.1. Concepts identification

By concepts identification, we mean to identify for each term the appropriate Wikipedia article that represents its intended meaning so that we can standardize names of the terms and enrich them by adding their categories and their possible synonyms as well. See the example depicted in Fig. 3. To perform this task, we used Google to retrieve the appropriate corresponding Wikipedia article for each term. As it shown in Fig. 4. Using Google as an intermediary to retrieve Wikipedia articles that represents terms meaning, we first passed to Google a term enclosing between the domain name (in this example: "Web Development") as a context and the word ("Wikipedia") to bring Wikipedia pages to the top. Then, we look for a morphological matching between the term and the titles of the top four retrieved Wikipedia pages. The simplest case occurs when a term can be matched directly to the first Google result. In other cases, a term could be matched directly to a page title, to a part of the title, or to one of the redirected pages. As well, terms could be matched to abbreviations that come with the Wikipedia entries' titles enclosed between parentheses. In some cases, matching to Wikipedia entries fails.

In fact, querying Wikipedia through Google allows taking advantage of techniques embedded in it, such as stemming and lemmatization, so that we have a high chance of finding the correct corresponding Wikipedia articles. In the example shown in Fig. 3, passing the term 'CSS' to Google resulted in retrieving the Wikipedia article entitled 'Cascading Style Sheets' since CSS represents a redirect page to this article in Wikipedia. In the case of disambiguated terms, (for instance the term "Ajax" that could refer to a programming language or a mythological Greek hero), the Wikipedia article that represents its intended meaning comes first in the Google results due to using the domain name as context. However, we use information available on the selected Wikipedia articles to enrich the terms. These includes redirect pages as alternative names, and Wikipedia categories containing that page that are listed on the bottom of each article.

4.2.2. Concepts relations extraction

This activity is aimed at discovering semantic relationships among the domain concepts**,** identified in the previous activity. Unlike most approaches in the literature that have re-used the

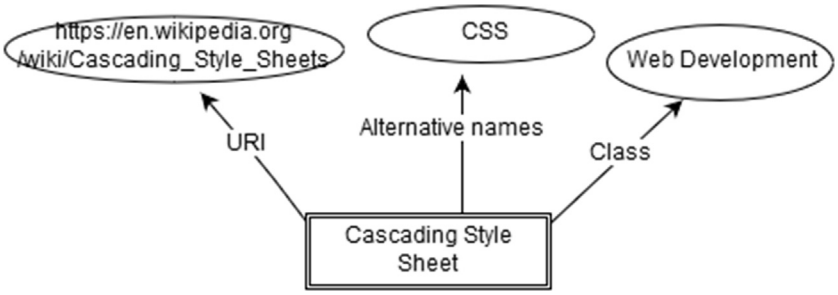

Fig. 3. An example of applying Concepts Identification process for the term "CSS".

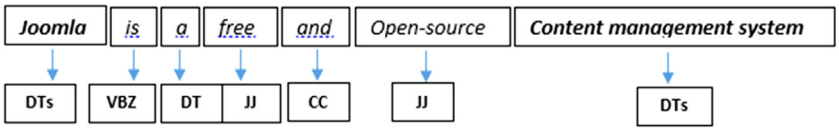

Fig. 5. Example of a generated string tag.

semantic relations defined in the existing ontologies such as WordNet and DBpedia, we extract semantic relations from the free-text content of Wikipedia articles, correlated to these domain concepts, since we aim to extract our own interesting relationships that are not provided by these ontologies; we are more interested in domain- dependent relationships that can be expressed by particular verbs of an area of interest. For example, in our case study domain, 'Web Development', techniques verbs such as *design*, *create*, *run*, *execute*, and *develop* etc., can form more interesting semantic relations. Table 3 shows the relationships of interest suggested by our method with examples; the domain concepts are in bold. In the following, we detail the two main activities in this phase: pre-processing and relations extraction.

4.2.3. Pre-processing

This task aims to extract sentences that contain pairs of domain concepts, generating a 'tag string' for each sentence. See Fig. 5. To perform this task, we first applied pre-processing to the whole text of the set of Wikipedia articles bound to the domain concepts in the Concepts Identification activity (Section 4.2.1). Pre-processing activity implies the elimination of non-alphabet sentences (such as codes and equations), semi-structured sentences (such as tables, info-boxes, and numeric menus), and the parentheses expressions. Then, we gathered sentences that contained pairs of domain concepts, finally, applying part-of-speech tagging to create a tag string for each sentence. The tag string is composed of the part-of-speech tags defined in OpenNLP[1] along with our own tags (DTs and PDR) that represent domain concepts, and the set of pre-defined verbs and keywords that represent our relations types respectively. Table 5 shows some of Keywords that represent the various types of relations of interest. To accomplish this activity, we reused some of the existing tools. First, to access Wikipedia data, we use Java Wikipedia Library (JWPL) (Zesch et al., 2008). Second, we use the Apache OpenNLP for part-of-speech tagging.

4.2.4. Relations extraction

To extract the relations of interest (see Table 3), we used a set of regular expressions (see Table 6) adapted from (Arnold and Rahm, 2014; Stoutenburg et al., 2009) to detect our defined linguistic patterns. More specifically, we matched the "tag string" for each sentence to a set of regular expressions to detect linguistic patterns that define relations of interest. As an example, Fig. 5 and Table 4 show a sentence with its created tag string by our method. Note in this example that '**Joomla**' and '**content management system**' were tagged by our own tag (**DTs**) as they match two domain concepts. Also, the combination of "*is*" and "*a*" has been tagged as (**PDR**) since it matches one of the typical pre-defined patterns that represent a relation kind. Table 5 shows some of the keywords that our approach used along with the regular expressions to identify the linguistic pattern for each relation of interest.

In fact, the position of the domain concepts in the sentences is important as our algorithm focuses on detection directed relations between pairs of concepts in the form (*Concept1*, **relation pattern**, *Concept2*).

[1] https://opennlp.apache.org/

5. Evaluation and discussion

In this section, we evaluate ontologies that were generated by our method from a folksonomy excerpt extracted from BibSonomy Dataset.

5.1. Terminology extraction

Terminology evaluation process (determining whether they are relevant to a given domain or not) is a difficult task due to the lack of evaluation frameworks and comparison methodologies, in addition to the lack/incomplete of electronic resources that can be used as a gold standard for the evaluation as well(Dellschaft and Staab, 2006; Vivaldi and Rodríguez, 2010; Alruqimi and Aknin, 2015). Furthermore, folksonomy tags are uncontrolled vocabularies that contain many slang words and abbreviations, while the electronic resources often use formal and compound terms. However, as we detailed in Section (3), our experiments were performed on a dataset composed of 20,000 resources annotated by 85,006 tags (11,865 unique tags). Two domains of computer science have been selected for the experiments: Semantic Web, and Web Development. To evaluate the obtained terminologies gained for the both two domains, we used majority voting of five researchers who we asked to make judgments of domain relevancy (whether a term is relevant to the given domain or not) for all the obtained terms by associating a label "*relevant*", "*irrelevant*", or "*uncertain*" with each term as follows: "*relevant*" for terms that represent topics within the given domain; "*irrelevant*" for terms that do not represent topics within the given domain *and* "*uncertain*" for unobvious terms. Tables 7 and 8 show results we obtained using three different thresholds *h*; where the "*Distinct Terms*" column shows all obtained terms after removing duplicated items, and the "*Relevant Terms*" column shows the terms marked as "*relevant*". Hence, the precision is calculated according to the equation: $on = \frac{|relevant|}{distincterms}$. Table 2 show sample of terminologies including several technical words and acronyms extracted for the two domains. In fact, many non-objective tags still appear in the obtained terminology causing lower precision.

5.2. Concepts and relations identification

After passing the domain terminology (93 terms related to the web development, obtained from extract domain terminology activity – with 0.7 threshold), to the procedure of concept identification described in Section (4.2.1), we found that only 55 of 93 terms can be matched successfully to Wikipedia entries. Hence,

Table 2
Samples of extracted terms for the web development and semantic web domains.

AJAX BLOG **ASP** BOOKMARKLETS BUILDER **DHTML** DWR **DOM** FIREFOX **GUI WYSIWYG LIGHTBOX HTTP W3C JS** OPEN_SOURCE **PHP** PHOTOSHOP	XML RELATIONSHIPS **RDF** QUERY PYTHON PHILOSOPHY **OWL P2P CYC** OVERVIEW **ONTOLOGY OPENDATA W3C** RSS TAXONOMY **SPARQL** DEL.ICIO.US
CSS WEBSERVICE MYSQL HTACCESS JSF JSON OSX TOREAD TEACHING	**SKOS** IDEA **WORDNET** BLOGENTRY **ONTOLOGY_DEVELOPMENT NEWNET MEMES N3** MEDIATION **KM LKL-KSS**
DATABASE COLOUR-SCHEMES **CMS** BUTTON **JAVASCRIPT** DEMO **MYSQL** FLASH **HTML JQUERY** FONT FORMS GIMP FRAMEWORK **LAYOUT JSP**	**JOSEKI ICAL** HUMOR GRID **FOLKRANK FOLKSONOMY** BENCHMARKING REASONING **SWAP SEMANTIC-WEB**

Table 3
Semantic concepts relations.

Relation type	Example	Extracted relationships
is a	Content management systems such as Joomla	<**Joomla**, *is a*, **CMS**>
has	PHP has a direct module interface called SAPI	<**PHP**, *has*, **SAPI**>
part of	JQuery is included with Visual Studio for use within ASP.NET	<**JQuery**, *part of*, **ASP.NET**>
use	ASP.NET is designed for web development	<**ASP.NET**, *used for*, **Web development**>
same as	Web development may refer to web design	<**Web development**, *same as*, **web design**>
Developed by	ASP.NET was developed by Microsoft. Joomla is written in PHP	<**ASP.NET**, *developed by*, **Microsoft** >

Table 4
POS-tagging definition with an example.

Word	POS tag	Word class	Notes
Joomla	DTs	Domain term	DTs (Domain terminology)
is	VBZ	Verb, 3rd person singular present	PDR (pre-defined relation)
a	DT	Determiner	
free	JJ	Adjective	
and	CC	Coordinating conjunction	
Open-source	JJ	Adjective	
Content management system	DTs	Domain term	DTs (Domain terminology)

Table 5
Keywords used for detecting pre-defined proposed relations.

Relationship	Typical patterns
Is a	is a/an, is a/an class of .describes a/an
Same as	Such as, refer to, known as, short for
Has/part of	Include, consist, contain, is part of
Created by	Developed by, designed by, created by, writing by, executed by
Used for/in	Used for, designed for, used in, developed to

Table 6
Regular expressions with examples.

The expression	Example sentence	The extracted relation
DTs VBZ DT * [PDR] DTs	***PHP*** *is a server-side scripting language designed for* ***web development***.	**PHP** *used for* **Web Development**
DTs VBZ DT * DTs	**ASP.NET** *is an* open-source server-side **web application framework** *designed for* **web development** to produce dynamic web pages	**ASP.NET** *is a* **Web Application Framework**

Table 7
Statistics of the results (Semantic Web Domain).

Threshold	Semantic web		
	Distinct terms	Relevant terms	Precision
0.5	243	150	61.73
0.6	190	126	66.32
0.7	116	77	66.38

Table 8
Statistics of the results (Web Development Domain).

Threshold	Web development		
	Distinct terms	Relevant terms	Precision
0.5	316	176	55.69
0.6	232	130	56.03
0.7	165	93	56.36

the textual content of these fifty-five Wikipedia articles will form the corpus which we base on to extract the concepts relations. By applying our method to extract relations between the domain concepts to this text corpus, we obtained 42 relations. Table 6 shows samples of the extracted relations. In fact, we did not obtain a high recall (42 relations between 93 concepts) as we used simple patterns with limited pre-defined types of relations in addition to the fact that 38 concepts cannot be matched to Wikipedia entries, but our patterns were in the most cases correctly detected, leading to a precision of 94%. However, we noticed empirically that mapping tags to Wikipedia entries can be used a good way for excluding non-objective tags since, normally, the non-objective tags do not have corresponding articles in Wikipedia. Another note is that some terms could not be correlated to a Wikipedia article due to missing completed context (e.g. "usability" term cannot be linked but "web usability" can be). In some cases, terms cannot be matched to Wikipedia articles due to the variant structures of compound terms. For instance, matching the term "DHML" to the article labelled "Dynamic Html" fails although they refer to the same concept. Nevertheless, using Google to querying Wikipedia increases the probability of positive matching terms to Wikipedia entities, as Google can recognise words morphology. However, for generating other domains ontologies by our algorithm, domain experts may be involved in selecting the suitable dataset for a given domain and suggesting keywords that form the relation of interest that are more suitable for the domain at hand. Social tagging websites have different orientations and audiences; users utilize them with various intentions. Finally, although our experiments show significant results with a precision of 94%, but with less recall as we use simple patterns and regular expressions. For simplicity purpose, this work identifies only a limit number of relations, but the relations patterns can be easily adapted to discover other types of relations. Furthermore, developing more complicated patterns based on NLP and deep machine learning techniques to capture unlimited types of relationships will be the goal of our future works. Besides developing a method that looks for corresponding entries on the different online knowledge sources for terms that cannot be mapped to Wikipedia.

6. Conclusion and future work

This work studied folksonomies as potential sources of knowledge that can be exploited for developing formal domain

ontologies. Despite many studies aiming at explicit semantic structures from folksonomies, the construction of a formal ontology consisting of concepts and well-defined domain-dependent relations among them still challenged not solved yet. Mainly, this paper focused on designing a new algorithm for deriving more interesting domain ontology from folksonomy. We have shown how those social tags can be used with the more formal knowledge available in Wikipedia to generate ontologies, which are more receptive to knowledge change, and are more representative to the online communities' collective intelligence (Mikroyannidis, 2007). However, enhancing the extraction techniques, as we stated early, and developing many ontologies using different data are issues that might be considered in future works. To examine our algorithm, we performed our experiments on a real dataset obtained from BibSonomy. The results of our experiments suggested that our algorithm can effectively learn domain terminology, as well as identify more meaningful relationships among the domain terminology compared to the other methods.

References

Al-Khalifa, H.S., Davis, H.C., 2007. Towards better understanding of folksonomic patterns. In: Proceedings of the 18th Conference on Hypertext and Hypermedia - HT '07. ACM Press, New York, New York, USA, p. 163. https://doi.org/10.1145/1286240.1286288.

Alruqimi, M., Aknin, N., 2015. Semantic emergence from social tagging systems. Int. J. Org. Collective Intelligence 5 (1), 16–31. https://doi.org/10.4018/IJOCI.2015010102.

Angeletou, S., 2008. Semantic Enrichment of Folksonomy Tagspaces. Springer, Berlin Heidelberg, pp. 889–894. Retrieved from http://link.springer.com/chapter/10.1007/978-3-540-88564-1_58.

Arnold, P., Rahm, E., 2014. Extracting Semantic Concept Relations from Wikipedia. In: Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14) - WIMS '14. ACM Press, New York, New York, USA, pp. 1–11. https://doi.org/10.1145/2611040.2611079.

Begelman, G., Keller, P., Smadja, F., 2006. Automated Tag Clustering: Improving Search and Exploration in the Tag Space. Retrieved from: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.120.5736>.

Benz, D., Hotho, A., Jäschke, R., Krause, B., Mitzlaff, F., Schmitz, C., Stumme, G., 2010. The social bookmark and publication management system bibsonomy. VLDB J. https://doi.org/10.1007/s00778-010-0208-4.

Cantador, I., Konstas, I., Jose, J.M., 2011. Categorising social tags to improve folksonomy-based recommendations. Web Semantics: Science, Services and Agents on the World Wide Web 9 (1), 1–15. https://doi.org/10.1016/j.websem.2010.10.001.

Cantador, I., Szomszor, M., Alani, H., Fernández, M., Castells, P., 2008. Enriching ontological user profiles with tagging history for multi-domain recommendations. In: 1st International Workshop on Collective Semantics: Collective Intelligence & the Semantic Web. Spain. Retrieved from: <https://eprints.soton.ac.uk/265451/>.

Ching Hsu, I., 2013. Integrating ontology technology with folksonomies for personalized social tag recommendation. Appl. Soft Comput. 13 (8), 3745–3750. https://doi.org/10.1016/J.ASOC.2013.03.004.

Damme, C. Van, Hepp, M., & Siorpaes, K. FolksOntology: An Integrated Approach for Turning Folksonomies into Ontologies, 2007.

Dellschaft, K., Staab, S., 2006. On How to Perform a Gold Standard Based Evaluation of Ontology Learning. Springer, Berlin, Heidelberg, pp. 228–241. https://doi.org/10.1007/11926078_17.

Dong, H., Wang, W., Coenen, F., 2017. Deriving dynamic knowledge from academic social tagging data: a novel research direction. Proceedings of iConference 2017, 661–666. iSchools https://doi.org/10.9776/17313.

Du, H., Chu, S. K. W., & Lam, F. T. Y. Social bookmarking and tagging behavior: an empirical analysis on Delicious and Connotea. In Proceedings of the 2009 International Conference on Knowledge Management. Retrieved from http://hdl.handle.net/10722/127067, 2009.

Font, F., Serrà, J., Serra, X., 2015. Analysis of the impact of a tag recommendation system in a real-world folksonomy. ACM Trans. Intelligent Syst. Technol. https://doi.org/10.1145/2743026.

García-Silva, A., Corcho, O., Alani, H., Gómez-Pérez, A., 2012. Review of the state of the art: discovering and associating semantics to tags in folksonomies. Knowledge Eng. Rev. 27 (1), 57–85. https://doi.org/10.1017/S026988891100018X.

García-Silva, A., García-Castro, L.J., García, A., Corcho, O., 2014. Social Tags and Linked Data for Ontology Development: A Case Study in the Financial Domain. ACM, New York, NY, USA, pp. 32:1–32:10. https://doi.org/10.1145/2611040.2611075.

Gruber, T., 2007. Ontology of folksonomy: a mash-up of apples and oranges. Int. J. Semantic Web Inf. Syst. 3 (1), 1–11. https://doi.org/10.4018/jswis.2007010101.

Hamdi, S., Lopes Gancarski, A., Bouzeghoub, A., Ben Yahia, S., 2012. Enriching ontologies from folksonomies for Elearning: DBpedia case. In: 2012 IEEE 12th International Conference on Advanced Learning Technologies. IEEE, pp. 293–297. https://doi.org/10.1109/ICALT.2012.197.

Heymann, P., & Garcia-Molina, H. Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems. Retrieved from /brokenurl#dbpubs. stanford.edu:8090/pub/2006-10, 2006.

Hotho, A., Jaschke, R., Schmitz, C., Stumme, G., 2006. Information retrieval in folksonomies: search and ranking. In: Proceedings of the 3rd European conference on The Semantic Web: research and applications. Springer-Verlag, pp. 411–426. https://doi.org/10.1007/11762256_31.

Jabeen, F., Khusro, S., Majid, A., Rauf, A., 2016. Semantics discovery in social tagging systems: A review. Multimed. Tools Appl. 75 (1), 573–605. https://doi.org/10.1007/s11042-014-2309-3.

Knowledge & Data Engineering Group, University of Kassel: Benchmark Folksonomy Data from BibSonomy, version of September 30st, 2008.

Lee, D.H., 2015. Comparative analysis of index terms and social tags Retrieved fromhttp://www.dbpia.co.kr/Article/NODE06335549 J. Korean Soc. Library Inf. Sci. 49, 291–311.

Mathes, A. (2004). Folksonomies - Cooperative Classification and Communication Through Shared Metadata. Retrieved July 2, from http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html, 2016.

Mika, P., 2007. ontologies are us: a unified model of social networks and semantics. Web Semant. 5 (1), 5–15. https://doi.org/10.1016/j.websem.2006.11.002.

Mikroyannidis, A., 2007. Toward a social semantic web. Computer 40 (11), 113–115. https://doi.org/10.1109/MC.2007.405.

Schmitz, P., 2006. Inducing Ontology from Flickr Tags. Edinburgh, Scotland.

Shirky, C. Ontology is Overrated – Categories, Links, and Tags. Retrieved December 29, 2016, http://www.shirky.com/writings/ontology_overrated.html, 2005.

Specia, L., Motta, E., 2007. Integrating Folksonomies with the Semantic Web. Springer-Verlag, Berlin, Heidelberg, pp. 624–639. https://doi.org/10.1007/978-3-540-72667-8_44.

Stoutenburg, S., Kalita, J., Hawthorne, S., 2009. Extracting semantic relationships between Wikipedia articles. Proc. 35th International Conference on Current Trends in Theory and Practice of Computer Science. Spindelruv Mlyn, Czech Republic.

Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., & Servedio, V. D. P. Folksonomies, the Semantic Web, and Movie Recommendation. In 4th European Semantic Web Conference, Bridging the Gap between Semantic Web and Web 2.0. Retrieved from http://eprints.ecs.soton.ac.uk/14007/, 2007.

Tesconi, M., Ronzano, F., Marchetti, A., & Minutoli, S. Semantify del.icio.us: automatically turn your tags into senses. In Social Data on the Web workshop (SDoW2008). Karlsruhe, 2008.

Tommasel, A., Godoy, D., 2015. Semantic grounding of social annotations for enhancing resource classification in folksonomies. J. Intelligent Inf. Syst. 44 (3), 415–446. https://doi.org/10.1007/s10844-014-0339-y.

Trabelsi, C., Ben Jrad, A., & Ben Yahia, S. (2010). Bridging Folksonomies and Domain Ontologies: Getting Out Non-taxonomic Relations (pp. 369–379). https://doi.org/10.1109/ICDMW.2010.72.

Uddin, M.N., Duong, T.H., Nguyen, N.T., Qi, X.M., Jo, G.S., 2013. Semantic similarity measures for enhancing information retrieval in folksonomies. Expert Syst. Appl. https://doi.org/10.1016/j.eswa.2012.09.006.

Vander Wal, T. Folksonomy Coinage and Definition. Retrieved from http://www.vanderwal.net/folksonomy.html, 2007.

Vivaldi, J., Rodríguez, H., 2010. Finding Domain Terms using Wikipedia. Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta, Malta. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/citations.

Wang, S., Wang, W., Zhuang, Y., Fei, X., 2015. An ontology evolution method based on folksonomy. J. Appl. Res. Technol. 13 (2), 177–187. https://doi.org/10.1016/J.JART.2015.06.015.

Zesch, T., Müller, C., Gurevych, I., 2008. Extracting lexical semantic knowledge from wikipedia and wiktionary. Proceedings of the Conference on Language Resources and Evaluation (LREC). Electronic Proceedings.

Zubiaga, A., Fresno, V., Martinez, R., Garcia-Plaza, A.P., 2013. Harnessing folksonomies to produce a social classification of resources. IEEE Trans. Knowledge Data Eng. 25 (8), 1801–1813. https://doi.org/10.1109/TKDE.2012.115.