Folksonomy Enrichment and Search

Sofia Angeletou, Marta Sabou, and Enrico Motta

Knowledge Media Institute (KMi)
The Open University, Milton Keynes, United Kingdom
{S.Angeletou,R.M.Sabou,E.Motta}@open.ac.uk

Abstract. The Semantic Web community has expressed its interest on how the Semantic Web technology can be applied more efficiently in a manner that supports real world applications. Additionally, the popularity of social tagging systems has demonstrated a clear need for organisation and more flexible ways of querying the user contributed content. This work presents FLOR, a folksonomy enrichment algorithm, which exploits a variety of knowledge sources to apply structure on the user tagspaces. In addition, a query mechanism is presented demonstrating how the enriched folksonomies structures can be interrogated by transforming the user keyword queries on folksonomies to formal queries on semantic structures. The first prototype of the FLOR enrichment algorithm and a first instance of the query mechanism have been implemented and a demonstration is available online[1].

1 Introduction

The interaction between Web2.0 and Semantic Web has been given special attention by pioneers of both fields [2,4,5]. On the one hand, Semantic technology is expected to solve folksonomies problems such as *lack of structure, ambiguity, synonymy, basic level variation, syntactical variation* that may impede folksonomy search either by returning few results but mainly, by not supporting meaningful or representative result presentation. On the other hand, the socially derived tagspaces and their emerging semantics are claimed to potentially provide valuable information for overcoming certain impediments of the Semantic Web such as slow evolution of ontologies. The efforts to combine folksonomies and the Semantic Web have followed two main lines of research, statistical methods based on co-occurrence and explicit application of semantics on tags.

FLOR's novelty mainly lies on its entirely automated nature. FLOR selects relevant knowledge on the fly without pre-selection and user interference and automatically enriches the tags with all the appropriate semantic information available. In addition to the FLOR enrichment algorithm, we present a first approach on interrogating the semantically enriched folksonomies to test the Semantic Web usefulness in this context.

[1] http://flor.kmi.open.ac.uk

L. Aroyo et al. (Eds.): ESWC 2009, LNCS 5554, pp. 801–805, 2009.

2 Technique

In the following we describe the FLOR enrichment algorithm that leads to semantically structured folksonomies and the query mechanism to translate the user queries into semantic queries on these structures.

2.1 FLOR Folksonomy Enrichment

FLOR is an algorithm that automatically transforms tagspaces into semantic structures by making use of multiple knowledge sources. FLOR enrichment is performed in four phases as depicted in Fig. 2. An earlier version of the algorithm is explained in more detail in [1].

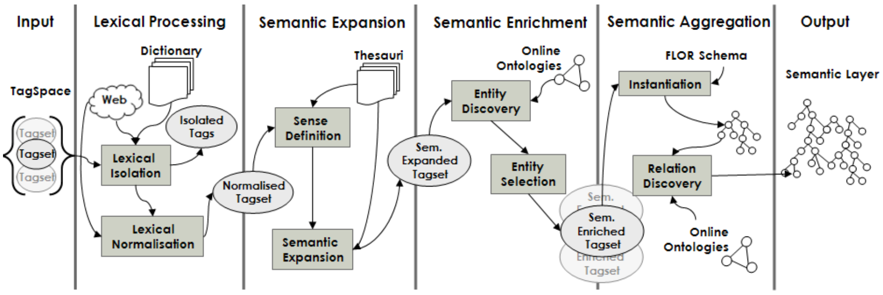


Fig. 1. FLOR Phases

PHASE 1: Lexical Processing. The first step of this phase, **Lexical Isolation)**, uses a set of heuristics to identify the tags not to be further processed by FLOR. Currently we isolate tags shorter than three characters, non-English tags and tags containing numbers or special characters. The **Lexical Normalisation** step produces a number of lexical representations for the tag, aiming to bridge the naming conventions used in folksonomies, ontologies and other knowledge resources.

PHASE 2: Sense Definition and Semantic Expansion. In the first step of Phase 2, **Sense Definition and Disambiguation**, sense disambiguation is performed by utilising WordNet based similarity techniques as explained in [1]. Another issue this phase tries to address is the Semantic Web sparseness. Expanding the tag with synonyms and lexical variations increases the possibility of finding the correct Semantic Web Entity in Phase 3. We perform **Semantic Expansion** for each tag, using WordNet.

PHASE 3: Semantic Enrichment. The third phase of FLOR identifies the Semantic Web Entities (SWEs) that are relevant for each tag taking as input the results of Phases 1 and 2. These SWEs are selected during the **Entity Discovery** step by querying a gateway to online ontologies [3]. Finally in the **Entity Selection** step we filter the SWEs in order to identify the ones that correspond to the relevant tags.

PHASE 4: Semantic Aggregation. The final phase returns the output of FLOR, a semantic structure with the definitions of the concepts corresponding to each tag and the relations of all the tags within the input tagspace. This is done by incrementally aggregating the enrichments of individual tagsets. The first step of Phase 4, the **Instantiation**, is executed after a tagset has been enriched and is the step where each individually enriched tagset is integrated into a final schema with which we represent the concepts of *Resource* `isTaggedWith` *Tag* `hasDefinition` *Sense* `isEnrichedWith` *Semantic (Web) Entity*. Finally, **Relation Discovery** is the last step of FLOR and returns the semantic layer linking together the tags. The actual relation discovery happens for the senses of the tags; among tags of a specific tagset but also cross-tagset, among tags of the overall tagspace (see Fig. 1). The Scarlet relation discovery algorithm ([6]) will be applied here. This step adds the statement: *Semantic (Web) Entity* `isRelatedto` *Semantic (Web) Entity* to the schema. The property `isRelatedto` can be any type of Object Property.

2.2 Querying Mechanism

At the moment the query mechanism supports only single keyword queries. As a next step we plan to support multiple keyword queries which will require a more complex query mechanism. The current demonstration performs search on a subset of Flickr resources (photos). The three different possible result sets for each query are Tags, Synonyms - Lexicals and Related.

The **Tags (A)** result set contains all the photos that have been tagged with the query keyword explicitly and show the same results as retrieved by a tag query in Flickr. The retrieval of this set is straightforward.

The **Synonyms - Lexicals (B)** set of results contains the photos that are tagged with synonyms or lexical variations of the query keyword. The notions of Synonym and Lexical are supported by our schema as Datatype Properties of a Sense. For example, *Sense_(Leaf)* `hasSynonym` "Foliage" and `hasLexical` "Leaves". The process of selecting this result set first identifies the senses that have the query keyword as a value in the `hasSynonym` or `hasLexical` properties. The mapping then to the tags and resources is straightforward.

The **Related (C)** set contains photos that are tagged with a sense that relates to keyword in the following ways.

- **SubClass**, tagged with subclasses of the query keyword.
- **SuperClass**, tagged with superclasses of the query keyword
- **Sibling**, tagged with senses sharing the same superclasses with the keyword
- **Generic Relation**, tagged with senses that relate with the keyword with all other possible relations (excluding disjointness) for example, meronymy.

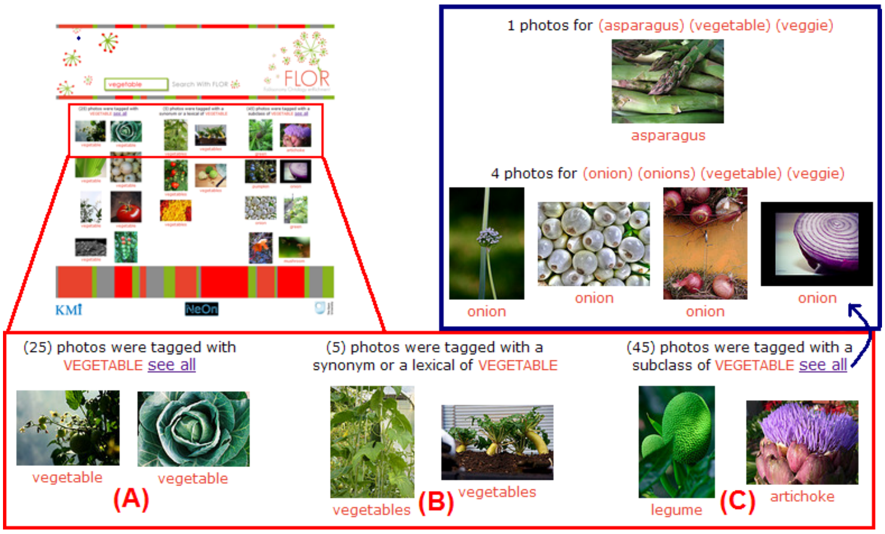
More specifically, the process of selecting the set Related (C) first maps the keyword query to a sense in the same manner as in Synonyms - Lexicals (B). Then, the related sense is discovered according to the above cases and the inverse process is followed to identify tags and photos. This means that we retrieve all the photos that are tagged with synonyms or lexicals of the previously discovered related sense.

3 Implementation

The first prototype has been implemented and evaluated on a dataset from Flickr. Currently we have partly implemented Phase 4 of FLOR (Semantic Aggregation) and only the SubClass case of Related (C) from the query mechanism. With regards to the data, we selected a Flickr subset, from the group **Plant [directory]**[2] (5.943 members and 63.454 photos on 24-07-2008). We then randomly selected 12233 photos with a total of 89446 tags. For the first evaluation of this setup 11 users asked the system queries related to plants (45 queries in total) and retrieved **36% additional results to the ones that are returned without the FLOR enrichment with a mean precision of 94%.**

4 Demonstration Plan

This demo will focus on showcasing the functionality of querying on a Flickr dataset enriched by FLOR as described. The enrichment algorithm will be explained and demonstrated for each case of user query and how the FLOR results are obtained in detail. The basic part of the demonstration will be performed on the Web application implementing the setup described in Section 3.



1 photos for (asparagus) (vegetable) (veggie)

4 photos for (onion) (onions) (vegetable) (veggie)

(25) photos were tagged with VEGETABLE see all

(5) photos were tagged with a synonym or a lexical of VEGETABLE

(45) photos were tagged with a subclass of VEGETABLE see all

Fig. 2. Demonstration Screens

The initial page of the application consists of a search box. The first result page is broken down into three columns as shown in the left part of Fig.2. Column (A) presents the photos that are explicitly tagged with the query keyword, e.g.,

[2] http://www.flickr.com/groups/plantdirectory/

`vegetable` and the results are exactly the same as the ones returned with the current tag-based search of folksonomies. Column(B) presents the results that are tagged with synonyms or different lexical representations of the query term e.g., `veggies, vegetables`. Finally, column (C) presents the photos that are tagged with subclasses of the query keyword e.g., `legume, artichoke`.

To avoid visual clutter, a maximum of ten photos are presented in each column. If a column contains more than ten responses, these can be accessed by clicking on "**see all**". For the first two columns this will simply lead to a "bag" display of photos. In the case of column (C), the "**see all**" page categorises the photos under each subclass of the query keyword as demonstrated in the top-right part of Fig.2.

Under each photo we present the tag which was mapped against the query term. For example, in column (C) of Fig. 2 the tags `legume` and `artichoke` were found as subclasses of `vegetable`. By clicking on a photo the user obtains a larger view of it with all its associated tags.

We expect the visitor to learn how the integration of Semantic Web with Web2.0 is realised and experience a real world application that is useful to the casual web user and user-friendly.

Acknowledgements

This work was funded by the NeOn project sponsored under EC grant number IST-FF6-027595.

References

1. Angeletou, S., Sabou, M., Motta, E.: Semantically enriching folksonomies with FLOR. In: Proc. of the 5th ESWC: CISWeb, Tenerife, Spain (2008)
2. Benjamins, R., Davies, J., Baeza-Yates, R., Mika, P., Zaragoza, H., Greaves, M., Gomez-Perez, J., Contreras, J., Domingue, J., Fensel, D.: Near-term prospects for semantic technologies. IEEE Intelligent Systems 23, 76–88 (2008)
3. d'Aquin, M., Sabou, M., Dzbor, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Motta, E.: Watson: A gateway for the semantic web. In: 4th ESWC, Innsbruck, Austria (2007)
4. Greaves, M.: Semantic Web 2.0. IEEE Intelligent Systems 22(2), 94–96 (2007)
5. Lassila, O., Hendler, J.: Embracing "Web 3.0". IEEE Internet Computing 11(3), 90–93 (2007)
6. Sabou, M., d'Aquin, M., Motta, E.: Exploring the semantic web as background knowledge for ontology matching. Journal of Data Semantics (2008)