LINKING THESAURI TO THE LINKED OPEN DATA CLOUD FOR IMPROVED MEDIA RETRIEVAL

P. Debevere, D. Van Deursen,
E. Mannens, R. Van de Walle

Ghent University - IBBT
ELIS Dept. - Multimedia Lab
Gaston Crommenlaan 8 bus 201
B-9050 Ledeberg-Ghent, Belgium

K. Braeckman, R. De Sutter

VRT-medialab
Gaston Crommenlaan 10 bus 101
B-9050 Ledeberg-Ghent, Belgium

ABSTRACT

Efficient media search applications can highly improve productivity in various domains. This is also the case in a broadcast environment, where large amounts of media are generated and archived. In this paper, we show how connecting a keyword thesaurus, used to annotate archived media items at the Flemish public service broadcaster in Belgium, with the Linked Open Data (LOD) cloud can greatly improve search applications. First, an algorithm is described that is used to automatically link concepts defined in a thesaurus with DBpedia, an important linking hub in the LOD cloud. The evaluation of this algorithm gives an overall accuracy of 81.64%. This is followed by an overview of features that are useful in a search application that were enabled through establishing a connection with the LOD cloud.

1. INTRODUCTION

Media retrieval is often enabled through annotating archived media items. When searching for relevant media items in the archive, a user typically enters some keywords in a search form initiating a search operation on the annotation data. It is clear that the quality of the annotations therefore has a major impact on the efficiency of search applications.

The 'MediaLoep' project[1] investigates how archived audiovisual content in a media production environment can be retrieved in a more effective and efficient way. One of the goals of this research project is to increase the amount of quality metadata by capturing and structuring the available information generated during the media production process. This automatically retrieved metadata can then be used by archivists as a starting point for further annotation. A data model was developed in order to uniformly represent the information contained in the data sources [1]. The data model was implemented as an OWL ontology, enabling an unambiguous and machine processable representation of the information. The use of Semantic Web technologies also allows to connect with other data sources using the Linked Open Data (LOD) principles [2].

In this paper, we focus on the automatic enrichment of the keyword thesaurus used to annotate archived media items at the Flemish public service broadcaster in Belgium (i.e., Vlaamse Radio- en Televisieomroep (VRT)). We then illustrate additional functionality that is obtained through connecting thesaurus concepts with the Linked Open Data cloud.

The remainder of the paper is organized as follows. Section 2 describes the thesaurus used at VRT. Section 3 describes how we connect the thesaurus to the LOD cloud. Section 4 describes the developed algorithm, which is evaluated in Section 5. Some features that are enabled through connecting a keyword thesaurus with the LOD cloud are described in Section 6. Conclusions are drawn in Section 7.

[1] http://www.vrtmedialab.be/en/projects/medialoep/

2. BASIS THESAURUS

Basis is the main tool to annotate archived media items at VRT. Basis currently contains over 700 000 records, spanning a period of over 20 years. A Basis record contains metadata fields such as title, textual description, duration, keywords and identification number of the corresponding item in the Media Asset Management System (MAMS).

The keywords field can only contain controlled values, defined by a manually maintained thesaurus. This thesaurus currently contains over 300 000 terms. These terms not only represent named entities such as persons and locations, but also nouns as these are used to describe the scene of a media item.

Basis also allows the addition of relationships between terms. The defined relationships are 'narrower term' (NT), 'related' (RT), and 'alternative label' (indicated in Basis as 'use for', UF). However, the majority of terms (approximately 83%) in the thesaurus have no relationships with other terms.

A Basis keyword (lead term, LT) is represented as a capitalized string, with the absence of diacritical signs. Although most keywords are represented in Dutch, keywords also appear in other languages such as English and French. Unfortunately, keywords are not provided with a language tag. In addition, keywords are not categorized in terms of types, such as persons or organizations. Some example keywords taken from the thesaurus are illustrated in Listing 1.

Listing 1. Example thesaurus keywords.

```
LT = BIDEN JOSEPH
UF = BIDEN JOE
RT = JACOBS JILL

LT = WASHINGTON
UF = WASHINGTON DC
NT = PENTAGON
NT = LINCOLN THEATRE
NT = WITTE HUIS
```

3. CONNECTING TO THE LOD CLOUD

The LOD cloud[2] consists of data sets that have been published according to the LOD principles. In order to link concepts defined in the thesaurus to the LOD cloud, the thesaurus was formalized to RDF using SKOS [3]. Listing 2 illustrates the formalized concepts of Listing 1 in N3 notation.

Listing 2. Formalized thesaurus concepts (N3 notation).

```
@prefix the:   <http://medialoep.vrt.be/thesaurus#>.
@prefix skos: <http://www.w3.org/2004/02/skos/core#>.
the:BIDEN_JOSEPH skos:prefLabel "BIDEN JOSEPH",
          skos:altLabel "BIDEN JOE",
          skos:related the:JACOBS_JILL.
the:JACOBS_JILL  skos:prefLabel "JACOBS JILL".
the:WASHINGTON   skos:prefLabel "WASHINGTON",
          skos:altLabel "WASHINTON DC",
          skos:narrower the:WASHINGTON_DC,
          skos:narrower the:WITTE_HUIS.
the:WASHINGTON_DC skos:prefLabel "WASHINGTON_DC".
the:WITTE_HUIS skos:prefLabel "WITTE HUIS".
```

A valuable data set which is considered an important linking hub in the LOD cloud is DBpedia[3], a formalization of Wikipedia. DBpedia has links with many other data sets and many other data sets also connect with DBpedia. In addition, DBpedia has a broad coverage in different domains. For these reasons, DBpedia is an ideal data set to map a keyword thesaurus to.

As DBpedia is a formalization of Wikipedia, for each Wikipedia article a corresponding DBpedia resource is available. A resource is provided with a label (*rdfs:label*) which corresponds to the article title in Wikipedia. An abstract is generated (*dbpedia:abstract*) consisting of maximum 500 words. Articles related to the same topic but in different languages are all represented as the same resource. For each available language, a label and abstract is generated with the appropriate language tag. Articles in Wikipedia belong to different categories. In DBpedia, these categories are represented as a hierarchy using SKOS. Wikipedia contains redirect pages e.g. to identify synonym terms and disambiguation pages linking to different articles describing different meanings of homonyms. These concepts are also present in DBpedia (using the *dbpprop:redirect* and *dbprop:disambiguates* respectively). For a more detailed description of DBpedia, we refer to [4].

4. LINKING ALGORITHM

The main steps of the algorithm are illustrated in Algorithm 1. The following subsections discuss each step in more detail.

4.1. Context construction thesaurus concept

In order to link a keyword with the corresponding concept defined in an external data set, we must know what the keyword represents. For example, when only reading the keyword string OBAMA, it is unclear whether the keyword represents a location in Japan or the current presidential family of the United States of America or maybe something else. For this reason, it is necessary to construct a context. Related work [5, 6] focusing on the detection and disambiguation of named entities in texts typically use a window of words around the detected named entity as the context. When mapping a keyword thesaurus, this approach is not possible. The way we construct a context for thesaurus keywords is described next. A list is constructed

[2]A representation of the LOD cloud is available at http://richard.cyganiak.de/2007/10/lod/

[3]http://dbpedia.org

Algorithm 1 General algorithm for linking thesaurus concepts with another data set. T represents the thesaurus which is linked to the external data set D.

for all $t \in T$ **do**
  $context_t = constructContext(t, T)$
  list $C = selectCandidates(t, D)$
  list $S = initializeScoreList(C)$
  for all $c \in C$ **do**
    $context_c = constructContext(c, D)$
    $score_c = calculateSimilarity(context_c, context_t)$
    $addToScoreList(score_c)$
  end for
  $concept = selectBestMatch(C, S)$
  $categorizeConcept(concept)$
  $linkConcept(t, concept)$
end for

Table 1. Co-occurrence list for the keyword OBAMA MICHELLE

keyword	frequency
OBAMA BARACK	79
USA	71
TOESPRAAK	37
PRESIDENTSVERKIEZING	34
OBAMA MALIA ANN	27
OBAMA SASHA	27
KANDIDAAT	20
...	

containing all the keywords that co-occur with this keyword in at least one Basis record. This list is then ordered according to the frequency count of each keyword. For example, the first entries of the ordered list generated for the keyword OBAMA MICHELLE are illustrated in Table 1. Note that the entire list typically contains much more entries (98 entries for the current example). However the frequency count of a co-occurring keyword drops significantly as it is positioned lower in the list. Therefore, the list is truncated using a threshold based on the term frequency of the first item in the list. In our implementation, a threshold of 30% was used, which means that for the current example all co-occurring keywords having a frequency count smaller than 23 are omitted. The remaining keywords then form the initial context.

If the keyword has relationships defined with other keywords in the thesaurus, the initial context is extended. All keywords appearing in a *skos:related* relationship with the current keyword are added to the context. Keywords that are defined as narrower terms of the keyword (*skos:narrower*), are only added to the context if the number of narrower terms is below a certain threshold (set to 5 in our implementation). This was done because it was observed that when a keyword has many narrower terms, these do not contribute to the context. Finally, when a keyword is defined as a broader term of the current keywords, all its broader terms are recursively obtained and added to the context.

4.2. Selecting candidates

As already mentioned in Section 3, the external data set used in this paper is DBpedia. A list of candidate resources is constructed based on the values of the labels (*rdfs:label*) of the resources. Because it cannot be expected that the labels of the corresponding DBpedia resource exactly match with the label used in the thesaurus, re-

sources containing only the words from the thesaurus keyword label (*skos:prefLabel* or *skos:altLabel*), either with Dutch, English, German or French label tag (because of the absence of language tags in the thesaurus) are added to the list. Note that the word order does not need to be the same as the word order of the keyword label (e.g., for persons, in DBpedia the first name appears first in the label whereas in the Basis thesaurus, the surname appears first). In addition, some labels contain context information between parentheses. If this is the case, the content between parentheses is ignored. Because the labels of the keywords in the thesaurus are capitalized and contain no diacritics, diacritics and locations of capitalized characters are ignored in the DBpedia labels. Also, occurrences of characters such as '-' are ignored.

Because Wikipedia (and therefore DBpedia) is a multilingual effort, it is possible that some concept is described in e.g. English but does not have a corresponding description in Dutch. For this reason, keyword labels are also translated to English, French and German using an automatic translating tool. Resources matching with the translated label as described above (with the exception that the language tag now must match the language to which the label was translated) are then also added to the candidate list. Note that if the matching DBpedia resources represent a disambiguation or redirect page, the resources that are disambiguated or redirected to respectively are added to the candidate list instead.

4.3. Candidate context construction

For each candidate in the candidate list a context must be constructed. The context is based on the abstract (*dbpedia-owl:abstract*), category labels (*skos:subject*) and type labels (*rdf:type*). As already mentioned in Section 3, categories in DBpedia are organized into a large SKOS hierarchy. Because this hierarchy is not completely transitive, a limit is set on the number of broader hierarchy levels that is included. Note that a context can be constructed for different languages, as every string literal in DBpedia is provided with a language tag.

4.4. Calculating similarity

In order to calculate the similarity between a thesaurus keyword and a candidate concept, the Vector Space Model (VSM) was used, as described next. Every context can be represented as a string. For a thesaurus keyword, the context string consists of the concatenation of the labels of the selected context keywords. For the candidate resources from the external data set (DBpedia), the context string consists of the concatenation of the selected labels and the abstract. A stop list is applied to each context string, followed by the application of a stemming algorithm. Then a vocabulary list was constructed based on the tokens present in the combined context strings. Suppose the vocabulary list has size n. Now the thesaurus keyword and candidate concepts can be represented as a n-dimensional vector. Each component of the vector represents the term frequency-inverse document frequency (tf-idf) weight. This is a commonly used weighting scheme in information retrieval. In order to assign a score indicating the similarity between the thesaurus keyword (k) and a candidate concept (c), the cosine similarity measure is used. The cosine similarity is calculated as follows:

$$sim(k,c) = \frac{\vec{V}(k).\vec{V}(c)}{\left\|\vec{V}(k)\right\| \left\|\vec{V}(c)\right\|}$$

The effect of the denominator is length normalization of the context strings. In our implementation, the final score is the average of the cosine similarity values calculated for the Dutch, English, French and German vector representations. Note that a keyword is always assumed to be in Dutch. In order to generate a keyword context in another language, the selected keyword labels are translated to the target language. When a translation of a label was not possible (e.g. the label represents a name of a person, or the label is not Dutch), the original label is used. The generation of context strings for DBpedia candidate resources in other languages is straightforward.

Table 2. Evaluation data set statistics

category	Basis	DBpedia	accuracy (%)
PER	641	268	96.26
LOC	369	348	86.18
OTH	2373	2041	76.99
All	3383	2657	81.64

4.5. Selecting best match

VSM is a commonly used model in information retrieval systems, where it is desired to have the best matching concept as the top result, but it is often acceptable that the best matching concept appears within the top t (e.g. $t = 5$) results. However, this behavior is not acceptable when linking concepts between data sets, as the candidate with the highest score should always be the matching concept. Therefore, we implemented the following additional heuristic for choosing the best matching candidate concept. If only one resource in DBpedia has an exact label match with the thesaurus keyword label (Dutch), we select this resource as the matching resource if the resource belongs to the top t results, where $t = 0.2c$, and c represents the candidate list size.

4.6. Categorizing concept

Once the best matching concept is chosen, categorization is performed based on type (*rdf:type*) and category (*skos:category*) information of the selected DBpedia resource. Note that many resources in DBpedia are defined as instances of classes defined in external data sets. For example, the DBpedia resource representing Barack Obama is defined as having type *yago:LivingPeople*, which is a class defined in YAGO[4], another data set present in the LOD cloud. In our implementation, we use type information of DBpedia and YAGO.

5. RESULTS

In order to evaluate the proposed algorithm, we collected the keywords from the Basis records annotating news items created during December 2009. This resulted in a list of 3383 keywords. For this list, a ground truth was constructed, consisting of the corresponding DBpedia resource (if present) and a categorization into one of the categories 'person' (PER), 'location' (LOC), and 'other' (OTH). Table 2 gives an overview of the evaluation data set statistics. The first column (Basis) gives the number of keywords per category, the second column (DBpedia) represents the number of thesaurus keywords that have a corresponding mapping with DBpedia. The last column of Table 2 gives the accuracy values. It can be seen that the accuracy values for the 'person' and 'location' categories are significantly higher than for the 'other' category. This is mainly due to the larger number of candidates that are selected for keywords in this category. However, overall the algorithm gives acceptable results.

[4]http://www.mpi-inf.mpg.de/yago-naga/yago/

Table 3. Categorization results evaluation data set

correct category (%)	PER	LOC
	91.42	89.94

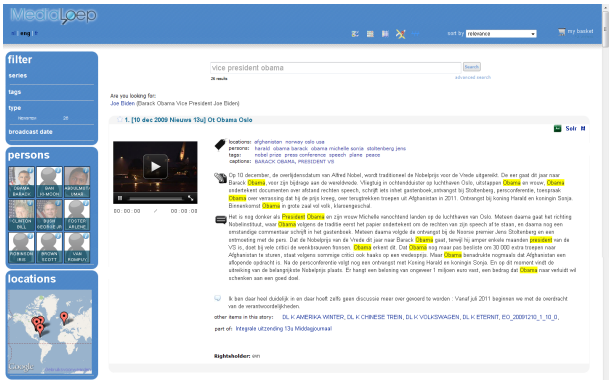

Fig. 1. Overview of the MediaLoep search application.

Table 3 gives an overview of the categorization results. As a keyword is by default categorized in the 'other' category, only results for the 'person' and 'location' categories are relevant. It can be seen that the categorization yields good results.

6. ADVANCED MEDIA SEARCH

Once a connection is established with the LOD cloud, many additional features can be implemented in a media search application. The features we implemented in our search application are:

- **Introduction of facets**: By applying categorization of the keywords based on information from the LOD cloud, the development of a faceted search application is enabled.
- **Internationalization**: After conversion using SKOS, each keyword is represented as a URI. Once linked with the corresponding DBpedia resource, additional labels in many languages are available, enabling media search in different languages.
- **Alternative visualizations**: Using geo-coordinate information, search results can be displayed on a map. This is obtained using the RDF version of GeoNames[5], another data set in the LOD cloud, which is connected to DBpedia.
- **Semantic query suggestion**: In addition to labels and abstracts, resources described in DBpedia contain many other formalized properties. These properties are used to enable semantic query suggestions in the search application as described in [1].

Fig. 1 illustrates the media search application. The left panel contains the introduced facets. Also, using Google Maps, a spatial representation of media items is obtained. The query panel located at the top shows an example of semantic query suggestion enabled through connecting the thesaurus with DBpedia. When a user for example enters the keywords "vice president obama", media items annotated with these query words are returned in the result list. In addition, the application suggests "Joe Biden" as a query the user might be interested in, because DBpedia contains a resource (dbpedia:Barack_Obama) which is linked to dbpedia:Joe_Biden via a property (dbpprop:vicepresident) with label "vicepresident". As the thesaurus concept representing Joe Biden is also linked with the corresponding concept in DBpedia, the retrieval of annotated media items with this keyword is enabled.

[5]http://geonames.org

7. CONCLUSIONS

This paper presented an algorithm for linking a keyword thesaurus that is used to annotate archived media items to the LOD cloud. Because the addition of keywords to media items is a frequently used method to enable media retrieval, the approach described in this paper is also applicable in more general use cases. Using this algorithm, concepts defined in the keyword thesaurus are automatically linked to DBpedia, an important linking hub in the LOD cloud. The evaluation illustrates that the algorithm gives acceptable results. The advantages of linking a keyword thesaurus to the LOD cloud are also demonstrated through the developed media search application. Using information present in the LOD cloud, this search application supports facets, semantic query suggestion, multilingual search and alternative visualizations.

8. ACKNOWLEDGEMENTS

The research activities as described in this paper were funded by Ghent University, VRT-medialab, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research Flanders (FWO-Flanders), and the European Union.

9. REFERENCES

[1] P. Debevere, D. Van Deursen, D. Van Rijsselbergen, E. Mannens, M. Matton, R. De Sutter, and R. Van de Walle, "Enabling Semantic Search in a News Production Environment," in *Proceedings of the 5th International Conference on Semantic and Digital Media Technologies*, Saarbrücken, Germany, December 2010.

[2] T. Berners-Lee, "Design Issues: Linked Data," 2006, http://www.w3.org/DesignIssues/LinkedData.html,.

[3] A. Miles, B. Matthews, M. Wilson, and D. Brickley, "SKOS core: Simple Knowledge Organisation for the Web," in *Proceedings of the 2005 international conference on Dublin Core and metadata applications*, Madrid, Spain, 2005, pp. 1–9, Dublin Core Metadata Initiative.

[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," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 3, pp. 154–165, September 2009.

[5] H. T. Nguyen and T. H. Cao, "Named entity disambiguation: A hybrid statistical and rule-based incremental approach," in *3rd Asian Semantic Web Conference (ASWC08)*. 2008, vol. 5367, pp. 420–433, Springer.

[6] R. Bunescu and M. Pasca, "Using Encyclopedic Knowledge for Named Entity Disambiguation," in *Proceesings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06)*, Trento, Italy, 2006, pp. 9–16.