Journal of Theoretical and Applied Information Technology
31st July 2013. Vol. 53 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CLUSTER LABELING WITH LINKED DATA

[1]MARTIN DOSTAL, [2]MICHAL NYKL, [3]KAREL JEŽEK

[1]NTIS, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic

[2,3]Department of Computer Science and Engineering, FAS, University of West Bohemia

E-mail: [1]madostal@ntis.zcu.cz, [2]nyklm@kiv.zcu.cz, [3]jezek_ka@kiv.zcu.cz

ABSTRACT

In this article, we would like to introduce our approach to cluster labeling with Linked Data. Clustering web pages into semantically related groups promises better performance in searching the Web. Nowadays, only special semantic search engines provide clustering of results. Other engines are doubtful as far as the quality of clusters and moreover a dependable system for labeling these clusters is lacking. Linked Data is a set of principles for publishing structured data in a machine readable way with regards to linking with other Web resources. This enables data from different sources to be connected and queried over the Internet. The information from Linked Data can be used for preliminary estimates of topics covered by a set of documents. Topics are represented as resources from Linked Data and are used for smooth human-readable labeling of clusters.

Keywords: *Cluster labeling, Linked Data, Clustering, Semantic web*

1. INTRODUCTION

The volume of available electronic documents is rapidly increasing, which leads to a requirement for their ingenious digital processing. One of the popular approaches to organizing textual data is the use of clustering algorithms, which divide documents into coherent clusters. The goal of clustering methods is to identify distinct groups in a dataset. The basic implementation of clustering puts together documents that share many terms. More generally, it puts together documents with similar features. The cluster hypothesis [1] says that documents in the same cluster behave similarly with respect to the relevance to information needs. The hypothesis states that if there is a document from a cluster that is relevant to a search request, then it is likely that other documents from the same cluster are also relevant.

The exploration of Internet resources is even more challenging than local text-document processing because of uncertainty as far as the quality of documents. Instead of making high quality web pages, some authors aim to make their pages rank highly by playing with the Web page features that search engine ranking algorithms are based on. This behavior is usually called search engine spam [2] [3]. Very often the unwanted pages have a higher ranking than the expected information sources. This is caused by the preparation of texts with regard to routine statistical analysis methods that should increase the position of an unwanted page in a search results. Search engine ranking systems were designed to prevent these search engine optimization (SEO) techniques. The most common ranking algorithms are PageRank [4] and HITS [5]. Those algorithms are based on the theory that the most important pages on the Internet are those pages with the most links leading to them. Links can be marked as votes. The importance of the page that contains the link and the number of outgoing links is also considered.

New kinds of spam aiming at links have appeared in the form of link farms. Building link farms is one technique that can deteriorate link-based ranking algorithms. Additional precautions have had to be taken by search engines in the form of identifying link farm spam pages [6].

Search engines are able to find related documents with regards to the content and quality of web pages, but the main problem remains untouched. It is hidden in the problem of a user's query. It is not difficult to imagine a situation in which it is hard, if not impossible, to formulate a query precisely. It is impossible to find related documents, if the topic of a requested document is not very common and the user is not familiar with the vocabulary appropriate for describing a topic of interest. Clustering can be the answer to this demand. There are many applications of clustering:

- Search result clustering,

- Scatter-Gather,
- Collection clustering,
- Cluster-based retrieval.

The principle behind search result clustering is to divide results into distinct clusters, so the user is able to choose a more specific subset of documents for listing. The search query is matched against clusters and the content of the best scoring clusters is returned as a result. This content can be divided into other clusters and the user can choose the appropriate cluster again. This approach is part of the Dogpile [7] and Yippy [8] search engines. The user can walk through clusters and use them as a filter for search results. This application is shown in Fig. 1.

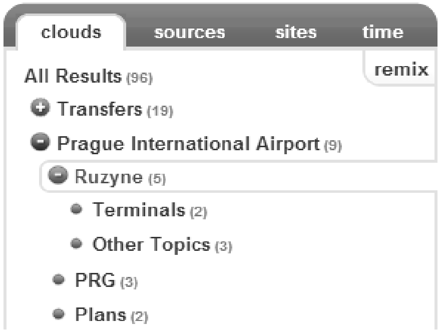

Figure 1: Yippy – clusters with labels

Scatter-Gather [9] is a technique for document browsing that employs document clustering as its primary operation for browsing large document collections. It is based on interactivity with the user. Initially, the system scatters the collection into a small number of clusters and presents them to the user. The user selects one or more of the groups for further browsing. The selected groups are gathered together to form a sub-collection and clustered again. Selection of the related clusters can be done based on the summaries or cluster labels.

Collection clustering is focused on dividing the documents in a set into smaller subsets for better performance of further information retrieval.

Cluster-based retrieval is based on the idea that a collection of documents can be divided into smaller subsets and a search can be done only by matching the query to one or more clusters. For example, when a query is set, the best document is found with common techniques and other related documents are chosen from the same cluster as the first one.

Fundamental clustering methods are based on the number of shared terms and other statistic-based approaches. Web 2.0 brings some additional possibilities on how we can improve clustering methods. Tags can be used as an additional feature for clustering methods [10] [11]. In [12] an approach was proposed where tags significantly increase the F-measure (1) for K-means from 0.139 to 0.225 in a test dataset built from ODP [13].

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \qquad (1)$$

Tags can increase the F-measure for the latent Dirichlet allocation (LDA) clustering algorithm from 0.260 to 0.307 in the same dataset. Many tagging services [14] [15] have been introduced with promising results. There are systems for assisting in tag selection [16] that help to solve some problems like synonyms and different levels of specificity. Tags can be used as keywords but rather in the form of Linked Data resources [17] with additional semantic information.

Basic concept of Linked Data is introduced in Section 3. Our approach to cluster labeling is discussed in Section 4. Evaluation of our method is realised with two popular data sets: 20 News Groups [18] and ODP [13]. Our results are discussed in Section 5.

2. PREVIOUS WORK

Nowadays, there exist four basic approaches to cluster labeling:

- Differential cluster labeling,
- cluster internal labeling,
- combination of inter-cluster and intra-cluster labeling,
- label extraction from taxonomies or corporas.

Differential cluster labeling compares clusters with each other and chooses terms or keywords that maximally distinguish the individual clusters. Common methods for feature selection can be used for this approach. Popular is mutual information, X^2-test or information gain (IG). IG [19] measures the amount of information that each argument contains about the other. For term *t* and category *c*, information gain is defined as:

$$IG(t,c) = \sum_{x\in\{t,\bar{t}\}} \sum_{y\in\{c,\bar{c}\}} P(x,y)\, log\, \frac{P(x,y)}{P(x)P(y)} \quad (2)$$

When *t* and *c* are independent, IG(t,c) = 0. The output of this labeling approach is a set of terms or keywords for each cluster. However, a list of significant keywords will many times fail to provide a meaningful readable label for a set of documents. In many cases, the suggested terms tend to represent different aspects of the topic underlying the cluster. In other cases, a good label may not occur directly in the text and it is not possible to extract it. User intervention can be required to choose a proper label from the suggested terms to successfully describe the cluster's topic.

Cluster internal labeling computes a label that depends on the cluster itself, not on other clusters. The common method is to label a cluster with the title of the document closest to the centroid based on the cosine similarity. Titles of documents are easier to read than a list of keywords. The title of a document can contain an important context or a topic that was not mentioned in the text directly or that was not chosen by statistical methods.

Another approach [20] uses a combination of intra-cluster and inter-cluster term extraction, based on a modified version of the information gain measure. This approach tries to capture the most significant and discriminative words for each cluster.

Other work investigates the contribution of external knowledge bases for cluster labeling. In [21] Wikipedia is used to enhance the quality of cluster labeling. A general framework for cluster labeling extracts candidate labels from Wikipedia in addition to important terms that are extracted directly from the text. The labeling quality of each candidate is then evaluated by several independent judges and the best evaluated candidates are recommended for labeling.

3. LINKED DATA

The concept of Linked Data [22] was first introduced by Tim Berners-Lee. He set up four rules for machine readable content on the Web:

- Use URIs as names for things.
- Use HTTP URIs so that people can look up those names.
- When someone looks up a URI, provide useful information using the standards (RDF*, SPARQL).
- Include links to other URIs so that they can discover more things.

More specific is the idea of Linked Open Data (LOD), which is based on the presumption of freely published data without restrictions in usage or additional fees.

The Linked Data initiative has given rise to an increasing number of RDF documents as well as other machine-readable sources, many of which are freely accessible online. These resources are often created as a result of database exports. That is the reason why we have to deal with duplicate information sources. There are two basic problems with duplicates resources: disambiguation and co-reference resolution. These problems were discussed in [23]. DBLP and DBpedia [24] are two of those common Linked Data resources often used for academic research.

4. APPROACH TO CLUSTER LABELING WITH LINKED DATA

In this section, we would like to discuss our approach to internal cluster labeling with Linked Data. First, we will explain the basic principles behind the Linked Data application, than we will look at a graph expansion and scoring illustration. The labeling algorithm will be presented by a pseudo code with additional comments.

4.1 Application of Linked Data

Linked Data contains information about a resource and moreover links to other related resources. The resources are applied as tags to documents. There are two basic types of links that we can directly use:

- Parent-child relation,
- links to synonyms.

These connections are bidirectional so a child can find his parent and a parent can find his children. Relations are described by ontology predicates. For example: *"dbpedia-owl:genre"*, *"skos:broader"*, *"dcterms:subject"*. The meaning of these predicates differs slightly, but we can use it in the same way. An example of these relations between resources is shown in Fig. 2.

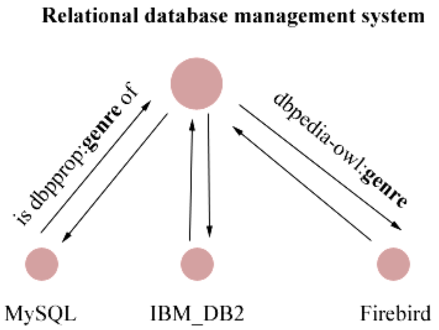

Figure 2: Scheme of hierarchical relations between nodes in LD

Synonyms are designated by the ontology relation: "*owl:sameAs*", which indicates true synonyms, and the relation "*skos:related*", which indicates related concepts.

4.2 Graph expansion and scoring

Each tag *t* is rated by a score. This score consists of two parts: the internal score and the gathered score (3).

$$t_i^{SC} = t_i^{SCI} + t_i^{SCG} \qquad (3)$$

First, we have to prepare and expand a graph of tags. Then, we set the tags' internal score in the preparatory phase. The tags' gathered score will be computed in the calculation phase. Graph expansion is implemented using the following procedure:

1. Tags are assigned to documents of a cluster.
2. The tags' internal score for each cluster is computed based on the number of assigned documents.
3. Relationships from Linked Data are used for graph expansion. Parents are added automatically, children are added in case they occur in the content of the documents. The internal score for those tags is computed as explained above.
4. Synonyms are replaced by one representative. This action removes cycles and a spanning tree is created.
5. The choice of the best representing tag that will be used for labeling is done in the following algorithm. In case more than just one tag is required, the terminate condition of *removeMinNodes* can be updated.

The gathered score of a tag will be determined by using equation 4:

$$t_i^{SCG} = \sum_{j=1}^{count(t_i^{NB})} \frac{t_j^{SC}}{t_j^{OE}} \qquad (4)$$

Where *count(*t^{NB}*)* is the number of neighbors of the tag, t_i^{SC} is the score of a tag and t_i^{OE} is the number of output edges from the tag.

The tag's internal score t_i^{SCI} can be computed in many different ways based on the statistical approach. The easiest way is to set the value of a tag's internal score as the number of documents that this tag is assigned.

4.3 Description of labeling method

Input:

```
    T = set of tags (synonyms are grouped)
    E = set of edges (used by tags)
    TS = set of scores (for tags)
```

Initialization:

```
  For each tag t in T do
      t.score = TS[t.id]; // tag is an object
```

Algorithm:

```
Function GetBestNode
  For step from 1 to N do // linear complexity
  norm = 0;
  For all tag t in T do
  For all tag.tn in t.incommingNeighbours do
      t.score += tn.score / tn.numberOutEdges;
      norm += square(t.score);
      norm = sqrt(norm);

      // normalization
      For all tag t in T do
          t.score = t.score / norm;
      norm = 0;

/* remove all nodes without input edges
   if there is more than one node */
  T.removeMinNodes(); // there are no cycles
```

Output:

The tag (node) with the highest score will be used for labeling:

$$t_{best} = \arg max_k \; t_k^{SC} \qquad (5)$$

Each tag contains a huge amount of semantic information about its name in different languages, a description in different languages and links to other related resources with additional semantic relations. If we only need a short label, we can use the title of a tag in an appropriate language. If we need the semantic meaning of a tag, we can use its description in machine or human readable form.

5. EVALUATION

A set of experiments was conducted to evaluate the correctness and reliability of our approach to the label selection problem. The data and evaluation measures will be described first, followed by descriptions of the experiments and their results (Table 1).

5.1 Data collections and evaluation measures

Three different data collections were used as a source of documents for clustering and labeling:

- Conferences – our collection consists of 15 000 calls for papers.
- News Groups – specifically 20 News Groups dataset [18].
- ODP – Open Directory Project [13], which is often use for the evaluation of text-mining problems.

Evaluation of the cluster labeling method is a difficult problem, for which no established methodology has gained wide acceptance. We have set up user evaluation of the cluster labeling as follows:

- Direct match (DM).
- Correct is close (CIC) – correct tag is at a distance of under two.
- HJ correct – human judge claims that label is completely correct.
- HJ wrong – human judge claims that label is completely wrong.
- HJ acceptable – human judge is able to accept this label as a relatively correct description of the document subset.

5.2 Experimental setup and results

The experiment consists of these steps:

1. Use one of the document sources: Conferences, News groups or ODP.
2. The human judge will create 10 clusters with the same number of documents. In our case it was 10 documents. This human will choose the best node from Linked Data for labeling these clusters.
3. Script takes those documents and applies one of the common clustering algorithms. K-means was used in our case.
4. Our labeling algorithm is applied to the clusters and evaluated as follows.

Table 1: Evaluation of our cluster labeling algorithm.

	Conf.	News Groups	ODP
DM	5	1	2
CIC	2	1	0
HJ correct	6	4	3
HJ wrong	2	3	3
HJ accept.	8	5	6

6. CONCLUSION AND FUTURE WORK

Our approach to cluster labeling is very intuitive and easy to use. Linked Data has great potential as an external source of free knowledge and promises better results with the growing number of electronic documents with free and open access. This information from Linked Data can be used directly in the form of labels or indirectly as ontology for information extraction. The labels from Linked Data are short and precise, which enables better utilization and understanding.

Further research will be focused on differential labeling with maximum effort placed on cluster distinction. We will try to find a measure for label distinction as the distance between tags in a graph may not be sufficient enough.

Another area of further research will be focused on existing graph algorithms such as PageRank and HITS. We would like to explore the possibilities of using these algorithms for the best choice of a tag that will be used as label.

ACKNOWLEDGMENTS

This work was supported by the grants GAČR P103/11/1489 “Methods of development and verification of component-based applications using natural language specifications”, European Regional Development Fund (ERDF), project "NTIS – New Technologies for the Information Society", European Centre of Excellence, CZ.1.05/1.1.00/02.0090 and SGS-2013-029 “Advanced computing and information systems”.

REFERENCES:

[1] Ch. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008.

[2] A. Perkins. Silverdisc. [Online]. http://www.silverdisc.co.uk/articles/spam-classification

[3] H. Garcia-Molina and Z. Gyongyi, "Web spam taxonomy," Stanford Digital Library Technologies Project, Stanford, Technical report 2004.

[4] P. Lawrence, B. Sergey, M. Rajeev, and W. and Terry, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford InfoLab, Technical Report.

[5] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," in *Journal of the ACM (JACM)*, vol. Volume 46 Issue 5, New York, 1999, pp. 604 - 632.

[6] B. Wu and B. D. Davison, "Identifying link farm spam pages," in *Proceedings WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, New York, 2005, pp. 820-829.

[7] Dogpile web search. [Online]. http://www.dogpile.com/

[8] Yippy. [Online]. http://yippy.com/

[9] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey, "Scatter/Gather: a cluster-based approach to browsing large document collections," in *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, 1992, pp. 318 - 329.

[10] G. Begelman, P. Keller, and F. and Smadja, "Automated Tag Clustering: Improving search and exploration in the tag space," in *Proceedings of the Fifteenth International World Wide Web Conference*, Edinburgh, 2006.

[11] C. H. Brooks and N. and Montanez, "Improved annotation of the blogosphere via autotagging and hierarchical clustering," in *Proceedings of the 15th International World Wide Web Conference*, Edinburgh, 2006, pp. 625-632.

[12] D. Ramage, P.l Heymann, Ch. D. Manning, and H. Garcia-Molina, "Clustering the tagged web," in *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, New York, 2009, pp. 54 - 63.

[13] (2012, June) ODP - Open Directory Project. [Online]. http://dmoz.org/

[14] (2012, June) Freebase knowledge base. [Online]. http://www.freebase.com

[15] (2012, June) Delicious. [Online]. http://www.delicious.com/

[16] S. Sood et al., "TagAssist: Automatic Tag Suggestion for Blog Posts," in *Proceedings of the International Conference on Weblogs and Social Media*, Colorado, 2007.

[17] M. Dostal and K. and Ježek, "Automatic tagging based on Linked Data," in *IEEE International Conference on Service-Oriented Computing and Applications*, Perth, 2010.

[18] 20 Newsgroups. [Online]. http://people.csail.mit.edu/jrennie/20Newsgroups/

[19] T.M., Thomas, J.A. Cover, *Elements of information theory*. New York, USA: John Wiley & Sons, 1991.

[20] F. Geraci, M. Pellegrini, M. Maggini, and F. Sebastiani, "Cluster Generation and Cluster Labelling for Web Snippets," in *Lecture Notes in Computer Science*, 2006, pp. 25-36.

[21] D. Carmel, H. Roitman, and N. Zwerdling, "Enhancing Cluster Labeling Using Wikipedia," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, vol. I, New York, 2009, pp. 139-146.

[22] T. B. Lee. (2009, června) Design Issues. [Online]. http://www.w3.org/DesignIssues/LinkedData.html

[23] A. Jaffri, H. Glaser, and I. Millard, "URI Disambiguation in the Context of Linked Data," in *LDOW 2008*, Beijing, China, 2008.

[24] (2012, June) DBPedia.org. [Online]. http://dbpedia.org