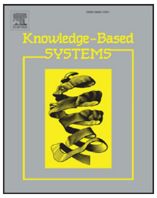
Knowledge-Based Systems 000 (2016) 1–11

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



Fast and reliable inference of semantic clusters

Nicolas Fiorini[a,*], Sébastien Harispe[a], Sylvie Ranwez[a], Jacky Montmain[a], Vincent Ranwez[b]

[a] LGI2P research center from the École des mines d'Alès, Site de Nîmes, Parc scientifique G. Besse, 30035 Nîmes cedex 1, France
[b] UMR AGAP, Montpellier SupAgro/CIRAD/INRA, 2 place P. Viala., 34060 Montpellier cedex 1, France

ARTICLE INFO

Article history:
Received 18 March 2016
Revised 30 June 2016
Accepted 6 August 2016
Available online xxx

Keywords:
Clustering
Cluster labeling
Semantic indexing
Neighbor joining
Complexity analysis

ABSTRACT

Document Indexing is but not limited to summarizing document contents with a small set of keywords or concepts of a knowledge base. Such a compact representation of document contents eases their use in numerous processes such as content-based information retrieval, corpus-mining and classification. An important effort has been devoted in recent years to (partly) automate semantic indexing, i.e. associating concepts to documents, leading to the availability of large corpora of semantically indexed documents. In this paper we introduce a method that hierarchically clusters documents based on their semantic indices while providing the proposed clusters with semantic labels. Our approach follows a neighbor joining strategy. Starting from a distance matrix reflecting the semantic similarity of documents, it iteratively selects the two closest clusters to merge them in a larger one. The similarity matrix is then updated. This is usually done by combining similarity of the two merged clusters, e.g. using the average similarity. We propose in this paper an alternative approach where the new cluster is first semantically annotated and the similarity matrix is then updated using the semantic similarity of this new annotation with those of the remaining clusters. The hierarchical clustering so obtained is a binary tree with branch lengths that convey semantic distances of clusters. It is then post-processed by using the branch lengths to keep only the most relevant clusters. Such a tool has numerous practical applications as it automates the organization of documents in meaningful clusters (e.g. papers indexed by MeSH terms, bookmarks or pictures indexed by WordNet) which is a tedious everyday task for many people. We assess the quality of the proposed methods using a specific benchmark of annotated clusters of bookmarks that were built manually. Each dataset of this benchmark has been clustered independently by several users. Remarkably, the clusters automatically built by our method are congruent with the clusters proposed by experts. All resources of this work, including source code, jar file, benchmark files and results are available at this address: http://sc.nicolasfiorini.info.

1. Introduction

In recent years, there has been a great evolution in document analysis applications. Clustering is the lion's share of automated document processing since it provides relevant organization of documents that synthesizes and underlines their properties and meanings. In our everyday life, organizing documents in (sub)folders is such a recurrent and crucial task that we tend to forget how tedious and time consuming it is. However, at some point we all complain about the emails piling up, the unsorted holiday pictures, the web pages that we saved on our bookmark list but that we cannot find among the hundred other ones. As more and more documents are now semantically tagged, it should be possible to automate their semantic organization, providing tools that would, for instance, organize your local library of scientific papers based on their keywords and MeSH annotations, structure your bookmarks based on the metadata of the corresponding web pages or sort your emails based on the tags you associate to them while accounting for the fact that "conference" and "workshop" tags are both refinement of a "scientific meeting" tag.

Conceptual annotation, also called semantic indexing, is proposed as a generic way of representing the content of documents. It aims at summarizing document contents with a small set of keywords or concepts of a knowledge representation such as an ontology. Such a compact representation of document contents eases their use in numerous automated processing tasks: content-based information retrieval, corpus-mining or documents classification. Semantic indexing is as tedious as complex: a synthetic and relevant semantic annotation requires a good understanding of the

[*] Corresponding author at: NCBI: National Center for Biotechnology Information, National Library of Medicine, Bldg. 38A rm 8N811B, 8600 Rockville Pike, Bethesda, Maryland 20894, USA.
E-mail address: nicolas.fiorini@nih.gov (N. Fiorini).

http://dx.doi.org/10.1016/j.knosys.2016.08.008
0950-7051/© 2016 Elsevier B.V. All rights reserved.

subject area the documents refer to, as well as a deep familiarity with the chosen knowledge representation. In recent years, an important effort has been devoted to automate semantic indexing, leading to the availability of large corpora of semantically indexed documents.

In this context, this paper introduces a hierarchical agglomerative clustering method that is based on the conceptual annotations of documents. The first originality of this approach lies in the use of a groupwise semantic similarity measure as the metric for document similarity: two documents are said to be close when the concept sets annotating them are similar in the sense of the semantic measure. Its second originality is that the similarity of two clusters is estimated as those of two documents rather than by some sort of initial document similarities aggregation as usually done. In our approach, when agglomerating two clusters, the conceptual index of the resulting larger cluster is automatically computed and then used to determine its similarity to others clusters consistently with the proposed cluster hierarchy. The final originality of our method lies in this tide imbrication of the clustering and labeling tasks which guarantees their consistence. The semantic similarity among clusters can hence be represented as tree branch lengths in the tree representation of the clustering thus underlining semantic properties of those clusters that can be used to identify the most meaningful clusters. To this aim, we propose a post-processing of the cluster hierarchy that takes advantage of branch lengths heterogeneity in the tree to only keep the most meaningful clusters.

The paper is organized as follows. The next section gives the context and our positioning with respect to the literature. Section 3 details the method and provides its space and time complexities. Section 4 presents the evaluation protocol. Section 5 provides the results obtained on a benchmark and their comparison with end-users' clusters; it discusses them and opens some perspectives. Finally, conclusions are drawn in Section 6.

2. Related work

Clustering is central to many applications in very different fields where people want to analyze and compare documents in the light of the domain knowledge they belong to: genes indexed by the Gene Ontology, scientific papers indexed by MeSH terms, bookmarks or pictures indexed by WordNet to cite a few. Organizing these documents manually within a hierarchy of named clusters/folders is a tedious everyday task.

As most operating systems use a hierarchical folder structure to organize electronic documents, people are now very familiar with this kind of document/folder organization. This organization provides a hierarchical representation of documents as they are grouped within imbricated named folders that can be seen as labeled clusters (each folder being a cluster labeled by its name). We all have faced the limit of such an approach. Especially when dealing with new documents that do not properly fit in the current hierarchy and would thus require to completely reorganize it, or when dealing with documents that could indifferently be placed in different folders. It has thus been suggested to replace the hierarchical folder approach by a document tagging system. An extensive comparison of those two approaches is provided in [1]. It concludes that there is no clear winner and that both strategies have pros and cons. For instance the folder strategy allows to declutter mailbox whereas multiple tag approaches facilitate later document search and allows to reveal unexpected or forgotten document connections. Tag approaches have been popularized by websites such as Twitter and its well known hashtag system. Numerous softwares have recently evolved to let users easily add multiple tags to documents and it is now possible to tag most document types using everyday life software applications. However recent work seems to indicate that, for most tasks, end-users continue to favor folder-based organization over tag-based one [2]. One can argue that this may be due to the force of the habit but this does not change the fact that end-users tend to favor hierarchical organization of their documents despites the advantages of the tag approach in certain cases. Here we propose to give the user the benefit of both approaches, while removing the tedious task of (re)constructing the first draft of their folder hierarchy by automatically building this hierarchy based on the document tags.

The whole point of clustering is to find groups of similar items that are different from other groups. Clustering methods are numerous [3] and depend on the type of data processed as well as clustering requirements and objectives that are fixed. In this paper we only consider hierarchical clustering methods, which produce a hierarchical representation of items instead of a flat partition of those items. The standard bottom-up strategy for this task processes by considering initial documents as singleton clusters and repeatedly grouping the two closest clusters into a new one, hence reducing by one the number of current clusters, until only one cluster remains.

The two main features of hierarchical agglomerative clustering are (i) the distance measure used to compare singletons and (ii) the approach used to distinguish which clusters are the closest ones. The distance measures implicitly determine the features used to cluster the documents. However, note that several distance measures can rely on the same features, but using them differently and hence providing a great diversity of output clusters. In this work, we focus on clustering approaches that rely on document semantic metadata. More precisely, we consider that each document is annotated by a set of concepts that are organized in hierarchical structures (e.g. ontologies, thesaurus) and that the clustering relies solely on these semantic annotations. Many semantic measures have been proposed to estimate the semantic similarity of two concepts [4], some relying only on the underlying hierarchical structuture (e.g. the path length between the compare concepts, *Information Content (IC)* of the compared concepts, etc.) whereas others use additional information such as color spectrum for image or word count for text documents. Note that the IC of a concept could also be defined solely based on the position of the concept within the ontology (roughtly speaking the closer the concept is to the ontology root the more generic it is and hence the lower is its IC) or it could be defined using additional external information such as the frequency of the concept in a representative corpus.

2.1. Semantic clustering

The literature presents some methods called semantic or mixing clustering with ontologies or metadata, however, there is no proper consensus on a field called semantic clustering the same way we define it, which is clustering documents by using their semantic descriptions. Kuhn et al. [5] for example introduced the concept of semantic clustering as the fact of grouping documents containing the same vocabulary. This approach is called semantic as they try to capture the meaning of the documents to cluster them, which is quite different from our objectives of clustering documents based on their semantic indices. Clerkin et al. [6] propose to use clustering in order to discover and create ontologies — and not using ontologies to cluster documents.

Some other studies are nevertheless closer to the scope of our work. Some researchers have for instance studied the impact of integrating knowledge base information in clustering algorithms [7]. To the best of our knowledge, Hotho et al. [8–10] have been the first to consider this kind of approach. Their work consists in enriching document annotations with background knowledge — in the most recent part, WordNet. Everything starts with the association of each document with a vector of term frequencies, further referred to as term vector. After being altered based on