A Social Bookmarking System to Support Cluster Driven Archival Arrangement

Marc Bron
Dept. of Inf. and Comp. Sc.,
Utrecht University
m.m.bron@uu.nl

Shenghui Wang
OCLC Research
shenghui.wang@oclc.org

Titia van der Werf
OCLC Research
titia.vanderwerf@oclc.org

Maarten de Rijke
University of Amsterdam
derijke@uva.nl

ABSTRACT

Cultural heritage materials are increasingly being made available through standard search facilities. However, it is challenging to automatically organize these materials in a way that is well aligned with users' specific interests. We report on the development of a social bookmaking system to collect human annotations that are used to measure the performance of three different clustering algorithms. We find that there is a discrepancy between the latent structure present in the data and the clusters annotated by humans. However, it is difficult to detect such discrepancies explicitly.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering;
H.3.7 [**Digital Libraries**]: Collection

Keywords

Archives; clustering; social bookmarking

1. INTRODUCTION

Today's continuous digitization and storage of cultural heritage material provides scholars and enthusiasts with access to a wealth of information. Cultural heritage material, however, does not necessarily have a rich representation. Think of photographs, videos, and non-digital material. Therefore, archivists create finding aids, i.e., metadata documents, to describe collections and to facilitate locating materials. Indexes of these finding aids provide users with search functionality that is nowadays considered standard: keyword search and facets.

Users of cultural heritage material, however, experience difficulties in locating material related to their research topics for several reasons. Metadata is generally sparse due to limits on the amount of material archivists are able to annotate. Further, the vocabulary used by archivists to describe material does not necessarily align with that of the searcher [9]. These characteristics of the data present challenges for out-of-the-box retrieval models.

One way to improve discoverability of material is to provide users with curated lists for their information needs. The phenomenon of creating curated lists is also known as social bookmarking [2, 4]. An example of a social bookmarking system is delicio.us. Such curated lists (or clusters) of finding aids potentially improve discoverability of material.

Who then should create these lists? In an archival setting, reference archivists have always provided users with suggestions for relevant material. Further, explicit topic assignments are provided through the use of controlled access fields. Reference archivists, however, are able to handle only a limited number of requests, while controlled access fields may not match a user's interests.

In this paper we first introduce a social bookmarking tool to support archivists and lay users in creating clusters of finding aids centered around topics. We then analyze the clusters produced by expert and lay annotators and compare them to clusters produced by centroid-based, hierarchical, and density-based clustering algorithms. Specifically, we aim to answer the following research questions: (i) how do the topics (latent structures) in the data correspond to topics developed by human annotators; and (ii) which clustering structure measure is suitable as an optimization criterion to obtain automatic clusters corresponding to human annotations?

2. BOOKMARKING FINDING AIDS

A Social Bookmarking System. We developed a social bookmarking system that consists of four views: (a) a topic overview, (b) a create new topic view (topic description), (c) a search view, and (d) a judge view, see Figure 1. When logging into the system a user first arrives at the topic overview screen, see Figure 1 (a). Here, the user is able to select an existing topic or create a new topic.

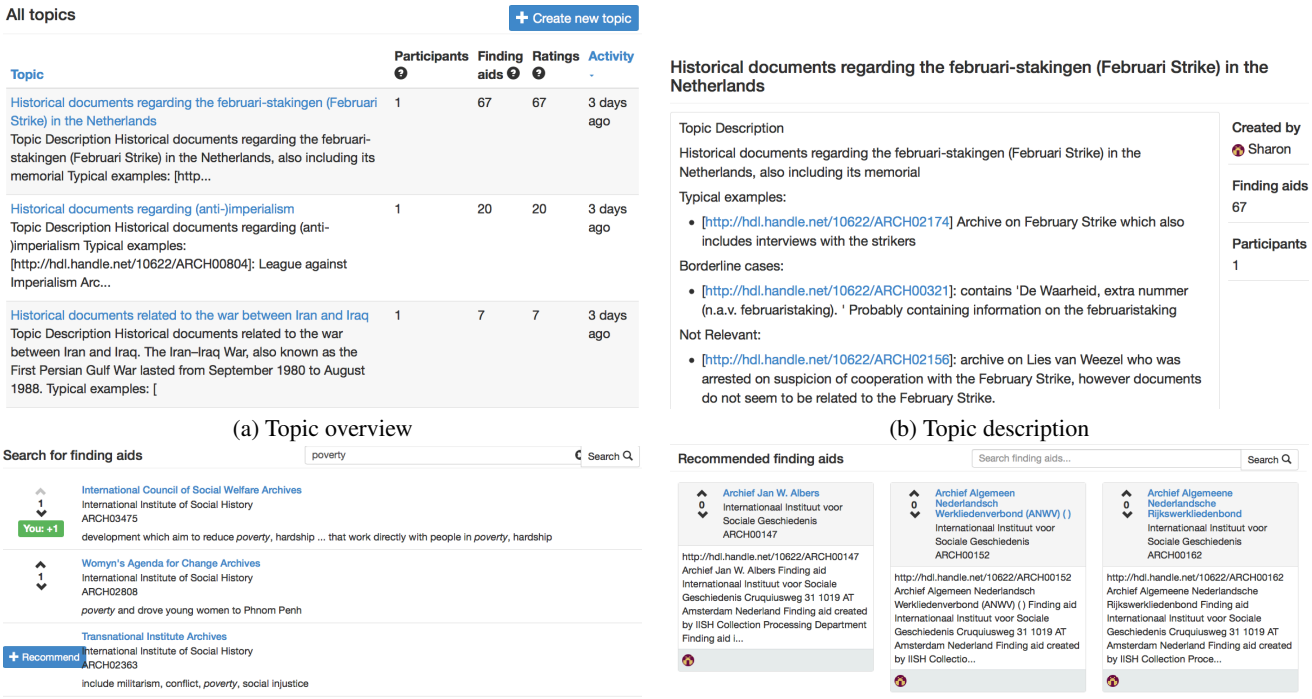
To create a new topic a user provides a topic title and a topic description, which can be edited later. The result is a topic description that is always visible at the top of the screen within a topic, see (b).

Within a topic there are two modes: search and judge, see (c) and (d) respectively. The search view allows a user to search a database of archival finding aids using keywords. If a search result has been judged before, its score is shown on the left, and the current user can vote for (by clicking the upward arrow) or against (by clicking the downward arrow). If a finding aid has not been judged before, a user can click the blue button "+ Recommend" to bookmark it.

In the judge view (d), the user is presented with all the finding aids that are judged related to this topic. The current user can go over them and make adjustments. Using this system, for each topic, we are able to collect positive and negative judgments on whether

IIiX '14 August 26 - 29 2014, Regensburg, Germany

http://dx.doi.org/10.1145/2637002.2637046.



(a) Topic overview

(b) Topic description

(c) Search view

(d) Judge view

Figure 1: The four views in our social bookmarking system.

a set of finding aids are relevant to this topic or not.

Data. To obtain a collection of archival finding aids we harvested 3400 finding aids from IISH[1] via their OAI-PMH api. Although a small collection by today's standards we believe that our analysis of this dataset provides relevant insights for two reasons: (i) searching in archival collections remains a challenging problem [5]; and (ii) clustering makes most sense within a single institute's data as archives focus on the curation of material related to different themes, e.g., social history or woman's archive [11].

Archivist annotations. The IISH archivists aim to annotate each of the finding aids in their collection with 1 to 4 themes (in case multiple are deemed appropriate). Themes are selected from a controlled vocabulary, such as "strikes" or "student movements." At this time 316 finding aids have been annotated with these themes.

Layman annotations. Two student assistants with a background in computer science created 42 topics and assigned 619 finding aids in total with our social bookmarking system. They were free to create any topic and were asked to record a typical example, some borderline cases, and the search terms they used while searching for candidates. The resulting annotations created by the archivists and students are available.[2]

3. CLUSTERING ANALYSIS

Clustering aims to group objects together that are similar to one another and different from objects in other groups. It does not use predefined labels to find rules for classifying labels to objects [12]. The structures discovered by a clustering method depend on the available features. In our case we have a finding aid which is a structured document consisting of several fields. We experiment with the following fields and consider all text within the field as well as their subfields as input to assess their value in grouping related material: *e*adheader (hdr), *e*adheader/unittitle (ttl), *a*rchdesc (ard), *a*rchdesc/controlaccess (ctl), *a*rchdesc/dsc (dsc); and *a*ll an aggregation of all finding aid fields.[3]

Bag Of Words. A standard approach to convert a document (here after we refer to finding aids or their fields as documents) into a feature vector is to consider each token as a feature. In order to tokenize our documents we remove all non-alphanumerical characters and split each document on white space. A stopword list is used and tokens that occur only once or are the 10% most frequent in the corpus are removed. The importance of tokens is further weighted by their tf.idf score [3].

Projection. Using tokens as features results in a high dimensional feature space in which the distances between vectors provide less discriminative power [8]. Various dimensionality reduction techniques have been proposed, such as latent semantic analysis (LSA). We use a manifold learning-based technique, i.e., isomap, that is able to detect non-linear structures in the data [10].

3.1 Cluster Structure Measures

Silhouette. The *silhouette* score is an internal characterization of the structure of cluster data. It ranges between -1 and 1, where closer to 1 means that points are tightly grouped and lie within their own clusters. A value closer to -1 indicates that points would be more appropriate in another cluster. The silhouette score sil is defined as: $sil(i) = \frac{\arg\min_{c,c\neq c_i} \text{avg\_dist\_nb}(i,c) - \text{avg\_dist\_nb}(i,c_i)}{\max\{\arg\min_{c,c\neq c_i} \text{avg\_dist\_nb}(i,c), \text{avg\_dist\_nb}(i,c_i)\}}$, where c_i is cluster of point i and $\text{avg\_dist\_nb}(i,c)$ is the average distance of a point i to all other points in a cluster c.

Adjusted Rand Index. The *rand index* measures the similarity between two partitions of a dataset. It is derived from the amount of pairs of data points that are in the same cluster (tp) in both partitions, the amount of pairs that are in different clusters in both par-

[1] International Institute for Social History, socialhistory.org/.

[2] ilps.science.uva.nl/resources/kiem-oclc-ugta

[3] See www.loc.gov/ead/tglib/elements for a description of each of the fields and Bron et al. [1] for their usage.

titions (tn), the amount of pairs that are in the same cluster in partition 1 but in different clusters in partition 2 (fp), and the amount of pairs in different clusters in partition 1 that are in the same cluster in partition 2 (fn). The rand index is given as: $R = \frac{tp+tn}{tp+tn+fp+fn}$. The *adjusted rand index* is a version of the rand index corrected for chance occurrences of pairs in appropriate clusters and is given by: $A = \frac{R-E[R]}{max(R)-E[R]}$, that is, the rand index adjusted by the expected rand index, normalized by the maximum achievable rand index [6].

V-measure. The *V-measure* is an external evaluation measure (using ground truth) based on the harmonic mean between the homogeneity and completeness scores of a clustering [7]. *Homogeneity* is between 0 and 1 and rewards a clustering that assigns only those data points that are members of the same class to a cluster. So homogeneity may be 1 when all data points are in a separate cluster or if clustering is perfect. *Completeness* is also between 0 and 1 and rewards a clustering that assigns *all* data points that are members of a single class to the same cluster. So completeness may be 1 when all data points are in a single cluster or if a clustering is perfect.

Intuitively, high homogeneity and low completeness indicates high fragmentation of the clusters, while low homogeneity and high completeness indicates data points are distributed over only a few clusters, i.e., it is hard to separate them into separate clusters.

3.2 Methods and Parameters

To investigate the latent structure in our data we use algorithms from three different families of clustering algorithms: density-based, hierarchical, and centroid-based clustering algorithms.

Density-Based. A density-based algorithm does not require the number of clusters to be specified up front. Instead DBSCAN relies on two parameters to determine the number of clusters: ϵ, which is the maximum distance at which a point is considered to be part of a cluster (reachable), and min_points, which is the minimum number of points that constitutes a dense region.

Hierarchical. Agglomerative hierarchical clustering is a connectivity-based clustering method. It uses a bottom up approach that starts with the assumption that all documents are in separate clusters and then merges clusters based on a cluster criterion and distance metric until a threshold is met. As such it is able to create clusters of different sizes depending on their distance. We use single linkage as the cluster criterion, i.e., the minimum distance between two points in different clusters. We set the threshold to produce no more then a certain number of clusters (max_cluster). With this parameter the algorithm greedily groups clusters together until the number of clusters specified by the threshold is reached.

Centroid-Based. Centroid-based clustering algorithms assign each data point to a centroid. The number of centroids needs to be specified and may be instantiated by selecting a number of data points at random. One of the properties of centroid-based methods is that they produce clusters that are similar in size. We use k-means clustering and experiment with varying the number of centroids (k).

3.3 Clustering Performance

Given the experimental settings outlined above we now look at clustering performance for different features, i.e., fields, and feature selection methods. Table 1 shows the performance for each field for the isomap feature selection method on both the student and IISH annotated topics. For each field the algorithm and parameter settings are shown that produced the maximum adjusted rand index score as well as the algorithm and settings that produced the maximum V-measure. We do not show the results for the bag of word features as these did not outperform the projected features. We experimented with projections on 50, 100, and 300 features but these produced comparable results. In the remainder of the paper, when reporting scores, we use the isomap projection on 50 features.

Table 1: Both the maximum adjusted rand index (A) and maximum V-measure (V) scores are shown together with the algorithm and parameter settings responsible for each of the fields. Result on both the student and IISH annotated topics are shown and the isomap feature selection method was used. This table uses the following abbreviations: hierarchical (h), DBSCAN (d), kmeans (k), homogeneity (H), completeness (C), field (fld), and parameters (prm).

	IISH					student				
fld	prm	A	V	H	C	prm	A	V	H	C
	isomap feature selection									
all	d 3, .3	**.29**	.27	.22	.36	k 100	**.19**	.54	.57	.51
all	d 1, .1	.00	**.52**	.93	.36	d 1, .1	.01	**.69**	.97	.54
dsc	k 1100	**.34**	**.52**	.72	.41	d 1, .1	**.03**	.41	.38	.45
dsc	k 1100	**.34**	**.52**	.72	.41	k 1100	.02	**.49**	.50	.48
ttl	h 200	**.41**	.39	.41	.37	k 100	**.10**	.45	.47	.43
ttl	k 1000	.05	**.49**	.80	.35	k 1100	.05	**.59**	.75	.49
ard	h 700	**.28**	.52	.74	.40	k 300	**.15**	.59	.71	.51
ard	h 1100	.26	**.54**	.82	.40	k 1100	.13	**.67**	.87	.54
ctl	h 300	**.24**	.40	.41	.38	k 100	**.10**	.47	.50	.45
ctl	h 1100	.06	**.49**	.84	.35	k 1100	.07	**.63**	.82	.52
hdr	h 100	**.14**	.39	.43	.35	k 100	**.15**	.50	.55	.47
hdr	d 1, .1	.01	**.50**	.87	.35	d 1, .1	.11	**.65**	.82	.54

The hierarchical and k-means algorithms produce clusterings that achieve maximum adjusted rand index and V-measure scores more often than DBSCAN does. Using the title field (ttl) hierarchical clustering gets closest to the IISH cluster annotations in terms of the adjusted rand index score. This is the case for the clustering produced by kmeans when compared to the student annotations. Regarding the V-measure we find that the archdesc field (ard) is most effective for the IISH clusters. There is no clear winner (algorithm or parameter setting) that performs well in terms of adjusted rand index and V-measure on both annotated topic sets.

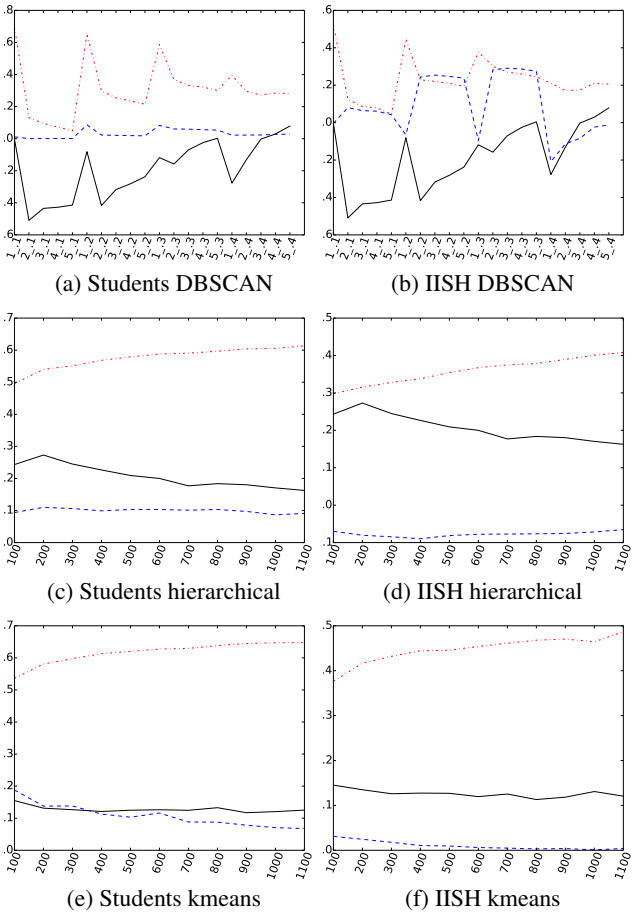
3.4 Analysis

To investigate the sensitivity of the algorithms, their parameter settings, and the resulting clusters, we consider the performance of the algorithms focused on the *all* field for various parameter settings. Figure 2(a) and (b) show the silhouette (black), V-measure (red, dot-dash), and adjusted rand index (blue, dash) scores for a range of parameter settings of DBSCAN on the IISH and student annotated topics. We observe a number of peaks in the V-measure indicating good clustering performance as determined by the ground truth. In these cases the minimum number of points and ϵ are small, e.g., the first peak has settings min_points $= 1$ and $\epsilon = 0.1$. With these settings DBSCAN has the tendency to create many small clusters. Table 1 shows that the peak in V-measure is caused by homogeneity score of .97 and a completeness score of .54, i.e., documents form the same class are in the same cluster but clusters are not complete. As the parameter settings increase, i.e., DBSCAN forms fewer clusters and the V-measure stabilizes at .28.

Regarding the adjusted rand index, DBSCAN produces better clusters on the IISH annotations than the student annotations. On those annotations a bias to singleton clusters (high homogeneity) results in dips in adjusted rand index score.

The silhouette score hovers between -0.5 and 0.1 indicating that documents are in different clusters or lie on the borders between clusters. This suggests that in the current feature space documents from different clusters are close and hard to separate.

Sub-figures (c) and (d) show the scores for different thresholds



(a) Students DBSCAN (b) IISH DBSCAN

(c) Students hierarchical (d) IISH hierarchical

(e) Students kmeans (f) IISH kmeans

Figure 2: Silhouette (black), V-measure (red, dot-dash), and adjusted rand index (blue, dashed) scores for a range of parameter settings for DBSCAN, hierarchical, and kmeans clustering for the student (left) and IISH (right) topic sets. Features consist of the *all* field with isomap feature selection as input.

for the hierarchical clustering algorithm. We see that the V-measure increases as the number of clusters increases. This is in line with our observations with the DBSCAN where the highest performance is with many small clusters. While the V-measure increases, the silhouette score decreases, i.e., as clusters better resemble the ground truth they start to deviate from the natural dense regions (topics) in the data. This suggests that the clusters as indicated by the ground truth do not correspond well to the structure in the data given the current features. The adjusted rand index is low and does not seem affected by the number of clusters.

Sub-figures (e) and (f) show the scores for the k-means algorithm that follow a similar pattern to those for the hierarchical clustering.

4. CONCLUSION

In this paper we introduced a social bookmarking system that allows archivists and lay users to produce curated lists of documents focused on a particular topic. We make two sets of topics available for studying user generated clusters of archival documents, one by archivists, and one by lay users. Regarding the potential of clustering algorithms to organize archival data around topics we found that there is a discrepancy between the latent structure present in the archival data and the clusters annotated by users. The hierarchical and k-means algorithm are able to optimize w.r.t. the V-measure by increasing the number of clusters. Optimizing performance in this way, however, is unhelpful for discovering topics as it biases towards singleton clusters. The density based algorithm (DBSCAN) is able to generate clusters closest to those proposed by human annotators as measured in terms of adjusted rand index without specifying the number of clusters. But the performance of DBSCAN is less predictable and only achieves good performance some of the times. Tuning the feature selection and clustering algorithm settings to optimize adjusted rand index seems promising as it strikes a balance between the two components of the V-measure (homogeneity and completeness). More experimentation is necessary to obtain effective features. In future work we plan to present users with the clusters produced by the various algorithms to evaluate their potential to seed new topics in a social bookmarking system.

Acknowledgments. This research was supported by OCLC Research, the European Community's Seventh Framework Program (FP7/2007-2013) under grant agreements nr 288024 (LiMoSINe) and nr 312827 (VOX-Pol), the Netherlands Organisation for Scientific Research (NWO) under project nrs 727.011.005, 612.001.116, HOR-11-10, 640.006.013, 650.001.005, the Center for Creation, Content and Technology (CCCT), the TimeCapsule project under number 314.99.111, the Dutch national program COMMIT, the ESF Research Network Program ELIAS, the Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW), the Netherlands eScience Center under project number 027.012.105, the Yahoo! Faculty Research and Engagement Program, the Microsoft Research PhD program, and the HPC Fund.

5. REFERENCES

[1] M. Bron, M. Proffitt, and B. Washburn. Thresholds for discovery: Ead tag analysis in archivegrid, and implications for discovery systems. *Code4Lib Journal*, 22, 2013.

[2] T. Hammond, T. Hannay, B. Lund, and J. Scott. Social bookmarking tools (i). *D-Lib Magazine*, 11(4), 2005.

[3] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge, 2008.

[4] M. Meder, T. Plumbaum, and F. Hopfgartner. Daiknow: A gamified enterprise bookmarking system. In *Advances in Information Retrieval*, pages 759–762. Springer, 2014.

[5] V. Petras, T. Bogers, E. Toms, M. Hall, J. Savoy, P. Malak, A. Pawłowski, N. Ferro, and I. Masiero. Cultural heritage in clef (chic) 2013. In *Information Access Evaluation*, pages 192–211. Springer, 2013.

[6] W. M. Rand. Objective criteria for the evaluation of clustering methods. *J. Amer. Statist. Assoc.*, 66(336): 846–850, 1971.

[7] A. Rosenberg and J. Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, volume 7, pages 410–420, 2007.

[8] M. Steinbach, L. Ertöz, and V. Kumar. The challenges of clustering high dimensional data. In *New Directions in Statistical Physics*, pages 273–309. Springer, 2004.

[9] M. Tahir, K. Mahmood, and F. Shafique. Use of electronic information resources and facilities by humanities scholars. *Electronic Library, The*, 28(1):122–136, 2010.

[10] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[11] S. Wang, A. Isaac, V. Charles, R. Koopman, A. Agoropoulou, and T. van der Werf. Hierarchical structuring of cultural heritage objects within large aggregations. In *TPDL'13*, pages 247–259, Valletta, 2013.

[12] S. M. Weiss, N. Indurkhya, T. Zhang, and F. Damerau. *Text mining: predictive methods for analyzing unstructured information*. Springer, 2010.