

An Initial Evaluation of Automated Organization for Digital Library Browsing*

Aaron Krowne and Martin Halbert
Woodruff Library — Library Systems
Emory University
Atlanta, GA
{akrowne,mhalber}@emory.edu

ABSTRACT

In this article we present an evaluation of text clustering and classification methods for creating digital library browse interfaces, focusing on the particular case of collections made up of heterogeneous metadata records. This situation is common in “portal” style digital libraries, which are built by harvesting content from many disparate sources, typically using the Open Archives Protocol for Metadata Harvesting (OAI-PMH). By studying the activity of users in an experimental system, we find that taxonomies built or populated using machine-learning (or “AI”) techniques provide a potentially useful avenue for browsing in this digital library scenario.

Categories and Subject Descriptors

H.3.7 [Information Systems]: Information Storage and Retrieval—*Digital Libraries*

General Terms

Experimentation, Human Factors, Measurement

Keywords

clustering, NMF, classification, categorization, browsing, taxonomies, portals, digital libraries, harvesting, portals

1. INTRODUCTION

The rapid acceptance and deployment of the Open Archives Protocol for Metadata Harvesting (OAI-PMH) has enabled the rise of digital library “portals”. These portals usually specialize within some particular subject domain, in which case we refer to them as “subject portals”. The Open Archives initiative has fostered the rise of these systems by

*We would like to thank the Andrew W. Mellon foundation for supporting this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

JCDL'05, June 7–11, 2005, Denver, Colorado, USA.

Copyright 2005 ACM 1-58113-876-8/05/0006 ...\$5.00.

making the metadata for digital library records extremely portable, and the underlying content databases transparent. In specific, OAI-PMH standardizes on a protocol, a data representation format (Dublin Core), and a design which puts low demands on the providers of content. Thus, digital libraries with special or otherwise sought-after collections can make their metadata available for re-use by relatively unaffiliated peers, allowing the sharing of content and addition of value within the digital library community in a *loosely-coupled* fashion.

A major example of a digital library portal built this way is the National Science Digital Library (NSDL¹). The NSDL’s constituent digital libraries are usually examples of subject portals (such as CITIDEL² for computing) or producers of original content (such as PlanetMath³ for mathematics). These portal-style digital libraries, subject or general, share as a common basis an aggregated collection of metadata records harvested from remote providers.

While OAI-PMH has been a great advance in that it has made it easy for such DLs to arise, the picture is not so rosy when it comes to building digital library services that are as unified as the underlying database itself. Although it is good that Dublin Core is a very flexible standard, this tends to lead to varying conventions and interpretations in its use at various digital libraries. In addition, often the best practices for field semantics are not followed. Yet, this does not “break” any of the technical elements of harvest and aggregation systems, which makes the resulting heterogeneity problem a latent one. Thus, when it comes time to build services on the aggregated collection, the system architect finds that the lack of a uniform semantic basis is a major impediment to functionality.

The MetaCombine project seeks to address the problem of building services over heterogeneous metadata and other aspects of digital library integration. This article describes one phase of the project, for which we have focused on a *browsing* service.

2. BROWSING

We define digital library *browsing* as an exploration or retrieval of a resource or resources through a navigable taxonomy. The taxonomy, which can be hierarchic, may also be referred to as a *subject hierarchy*, *classification scheme*, or

¹<http://www.nsd1.org/>

²<http://www.citidel.org/>

³<http://planetmath.org/>

various hybridizations of these terms. Typically browsing through such an interface is used when the end user does not know how to express the resource they are looking for in terms of keywords, or when they do not have a specific resource in mind. In addition, the intellectual coverage and hierarchic structure of a taxonomy is in itself a learning and exploration tool, a fact exploited by end users even if they do not consciously realize it. Thus, taxonomic browsing is a distinct and valuable digital library service.

The browse taxonomy (or *browse scheme* as we will interchangeably call them) may be presented in a variety of ways. Typically one encounters a directory-style interface (as in Yahoo or Open Directory), with levels in the hierarchy displayed as clickable category names, and DL items in that category shown below. Alternatively, the categories can be rendered graphically. We plan to study such interfaces in a later phase of the MetaCombine project. However, for this phase, we used a simple text-based directory-style listing.

The problem of metadata heterogeneity in the subject portal DL setting manifests itself in full force in the case of browsing services. In order to place resources into an taxonomy (browse or otherwise), some sort of *classification* must be a part of each resource's metadata. Further, the classification metadata must uniformly correspond to the taxonomy used for the browse interface. In the loosely-coupled harvesting scenario, however, there are a number of problems which undermine this assumption:

- Specialized DLs tend to use a variety of taxonomies, standard or ad hoc. They are different and incompatible, but have the same coverage, as far as the subject domain is concerned.
- Some DLs will use no classifications at all, forgoing an important class of services.
- Some DLs will not propagate this portion of the metadata, perhaps because it is extra work to encode it, or there is simply no standard way to do so.
- Some DLs might employ classification schemes that are too coarse for a specialized domain (such as UDC or LCCS⁴), rendering them of minimal use for information organization within the scope of the subject portal DL.
- The subject domain covered by the DL may be completely novel, and no prior scheme may exist.

Without taxonomic browsing, users of subject portals may be reduced to browsing via separation into *subcollections*, with each "folder" representing a source Open Archives (or other form of) content provider. This kind of organization is only useful in rare instances where the user cares more about the digital library of origin than the topic of the actual content. Common topics which appear in many subcollections may end up being fragmented in this case, violating one of the basic principles of taxonomy design and the foundation of their utility. This was the initial situation with AmericanSouth.org⁵, the digital library we are using as the basis of our study.

⁴These are Universal Dewey Classification and Library of Congress Classification Scheme, respectively.

⁵<http://www.americansouth.org/>

To solve this browse taxonomy problem, a few things need to be done:

1. Taxonomies need to be created (or discovered) where they are lacking.
2. Resources must be assigned category labels corresponding to the browse taxonomies desired.
3. The previous two must be done automatically, or mostly-automatic, due to the quantity of content which must be labelled or relabelled (for example, around 40,000 items for AmericanSouth.org).

We investigate meeting these goals with both *text clustering* and *text classification* methods. It is our hypothesis that clustering and/or classification can provide functional browse schemes for subject portal DLs which exceed the functionality of provider-based subcollections.

Some readers may ask why we cannot simply rely on manual, expert-generated taxonomies and classifications. We concede that such a vehicle of producing content organization is currently the ideal and will likely remain so for quite a while. However, expert, manual construction of taxonomies is known to be very expensive [12]. In addition to the construction of the taxonomy, resources are optimally placed into it by manual classification, which in turn dwarfs taxonomy-construction in cost, and clearly grows without bound in the size of the collection (hence the reason for condition #3). Thus, we take it for granted that manual browse schemes would outperform any of the "automatic" schemes we are studying here, but must point out that the economics of the situation well-motivates the consideration of automated methods.

3. BACKGROUND

In this section we give an overview of text classification and clustering in general, as well as some essentials of which specific algorithms and techniques we employed in our test system. The key difference between text clustering and classification is that classification assumes the existence of a classification scheme (or taxonomy *qua* classification scheme) and places items within it. By contrast, clustering takes a set of items *ab initio* and arranges them into subsets in such a way as to reveal "latent topics." From this, a classification scheme can usually be derived with a relatively small amount of work. Depending on whether a browse taxonomy already exists or whether one is desired, both clustering and automatic classification may potentially be useful to digital library portals for browse organization.

3.1 Classification

Text classification is a well-studied field, with extensive applications to digital libraries already. In the past 10-15 years, there have been a number of advances that have accelerated increases in effectiveness of text classification [11].

Text classification refers to a method, however, and not a technique. There are a number of algorithms and techniques based on them which all perform text classification. The particular technique one should use often varies depending on the situation. Some of the most prominent techniques are Naive Bayes, Support Vector Machines (SVMs) [4], k-Nearest Neighbor (kNN), and on-line linear (tf-idf vector space).

To be more formal, text classification assigns to a document d a category label c from the set of all categories, C . A text classification algorithm may in fact produce a ranked list of all $c \in C$ for each d , with numeric ranks typically of the form $0 < r < 1$. Text classification requires a *training set*, that is, a set of already-labelled (categorized) documents. From this set, a model is built which allows the classification of novel documents, not in the training set.

Text classification is actually just an instance of the classification problem in general, where *objects* having *attributes* (or *features*) are placed into *categories* based on the values of these attributes. To fold *text* classification into this family of algorithms, document words are interpreted as the features, and entire documents are interpreted as numerical vectors. The indices of these vectors correspond to features, and the values correspond to the influence or importance of that feature.

However, what separates text classification from traditional classification tasks is the *high-dimensionality* of the problem; the number of features (or dimensions) is proportional to the *dictionary size* of the corpus being classified. To help mitigate this problem, some preprocessing techniques can be applied, such as stemming and stopping [10], and dimensionality reduction [6]. However, dimensionality usually remains at the thousands to tens of thousands of features level. This has in turn affected which algorithms are typically selected for text classification [6].

In our case, we elected to use an on-line linear, Rocchio-weighted classifier. We used the BOW text mining package for our classifier [9], and in our tests of a number of leading algorithms, found that the Rocchio classifier performed the best by a significant margin. To discover this, we used ten-fold cross-validation on our training set, the Encyclopedia of Southern Culture articles, and compared SVM, Naive Bayes, Rocchio, kNN, and others⁶.

3.2 Clustering

Text clustering shares many commonalities with text classification. However, there is a major philosophical and practical difference: clustering is *preclassificatory*. With clustering, one seeks to classify documents when no prior classification scheme exists. Thus, clustering both generates a taxonomy and places documents within it.

Similar to classification, clustering is actually a general method, applied to text documents via a similar interpretation of their human-readable content. There are a number of techniques based on a variety of algorithms to perform clustering. Clustering has been studied for decades, largely in the social sciences [5]. However, there has recently been a blossoming of new techniques, as clustering moves into computing and information sciences. The dimensionality problem has been an even greater impetus for the discovery of new techniques here.

Clustering techniques break down into two families: agglomerative (bottom-up) and partition-based. Agglomerative clustering iteratively groups individual data points (documents) into clusters based on their similarity, producing

⁶This result conflicts with many reports that find SVMs perform best, as well as different work we have done with the WEKA machine learning system [3]. We are exploring this unexpected result, but do not believe the particular classifier used changes the nature of the results presented here.

a tree structure. However, the order of complexity of agglomerative algorithms is $O(n^2 \log n)$ for n items, which is prohibitively expensive for large collections [14].

Because of this performance limit, we focus on partition-based techniques. Some well-known clustering techniques by partitioning are k-means [13], Naive Bayes [1], the Gaussian mixture model [8], Latent Semantic Indexing (LSI) [2], and graph-based techniques (which turn out to be basically equivalent to techniques like LSI, see [14]). For our study, we employ a new method called Non-negative Matrix Factorization (NMF) [7]. NMF is a relatively new technique, applied to clustering in a similar manner to LSI, with NMF replacing the singular value decomposition (SVD) employed by LSI, and eliminating the need for a secondary cluster demarcation step. Recent work shows that NMF tends to match or outperform LSI and other techniques for document clustering [14]. Thus, we selected NMF as the basis for our clustering system.

We implemented NMF⁷ as described in [14]. The iteration typically converges after a small number of steps (10-20). The output can be thought of as weighted mappings of terms to clusters and documents to clusters. From the former, we can derive labels from the clusters. From the latter, we can derive classifications, by using the cluster with the maximum mapping weight for each document.

4. EXPERIMENT

Our goal was to compare the browse efficacy of a variety of AI-utilizing browse schemes to the trivial scheme (separation of resources into subcollections based on their digital library of origin). Thus, the trivial scheme serves as the control for this experiment. The four schemes we compared were:

1. **Subcollections** (*21 categories*) - Separation of resources into subcollections. These corresponded to their Open Archives site of origin.
2. **Top-level clusters** (*25 categories*) - A one-level scheme consisting of clusters formed over the entire digital library corpus (the domain of the history and culture of the American south).
3. **ESC-classified** (*24 categories*) - A one-level scheme consisting of pre-existing categories derived from the chapters of the Encyclopedia of Southern Culture. The articles of the encyclopedia were used as the training set for the classifier.
4. **Subcollection clusters** (*21 top-level categories, 232 total*) - In order to test whether subcollections could be useful in combination with the implicit organization of subcollections, we performed a separate clustering within each subcollection. This resulted in a two-level scheme, with subcollections at the top, and specialized categories derived from clustering within each.

The text used for clustering and classification was derived from the metadata of AmericanSouth.org records. We represented each record by extracting the Dublin Core **title**,

⁷Our implementation is available at <http://www.metacombine.org/>.

description, and subject fields⁸, concatenating them together, and performing some basic text cleaning. Thus, all of these metadata fields were equally weighted.

To test browse efficacy, we reasoned that it would make the most sense to have actual users engage in browse tasks using the four schemes above. To execute the study, record results, and automatically control for bias, we built an online experimental system accessible through the web. The system required users to perform a series of known-item retrievals through various browse schemes, and recorded their progress.

The details of the system are as follows. Users entered through a front page, and logged-in with their email address. Each user was first assigned a random browse scheme. Within this browse scheme, they were assigned a random resource for the current browse task, which was guaranteed to be categorized within the current scheme. The screen was divided vertically into two panels, with the current resource summarized in the left-hand panel. This summary consisted of selected portions of the metadata printed tabularly (title, description, subject descriptors, ID). In the right-hand panel was a hierarchical navigator for the current scheme. Categories were displayed as links, and clicking a category would yield either another page of category links, or a list of resources in the category if it was a leaf node (summarized as titles, descriptions, and IDs). Unique integer identifiers were used for each resource, replacing Open Archives string identifiers, to eliminate hints as to the collection of origin of each resource.

As the users navigated through the right-hand panel, their movements were recorded in a clickstream log. In addition to this, a record was kept of wall-clock time elapsed during each browse task. When they finally thought they found the resource for the current browse task, they could click on a “found it!” link. If it was not the correct resource, they would get an error message, and could go back to continue. Otherwise, they were congratulated on a successful find, the browse clock was stopped, and were assigned a new resource. They could also give up on the current browse task after a certain number of clicks spent searching, at which point they’d be assigned a new task (with a new resource). This process proceeded until four resources had been found within the current scheme, at which point the user would get a text box to enter in free-form comments about the scheme. After this, a new scheme would be randomly picked, and four browse tasks would need to be successfully completed within this scheme as well.

This continued until all four schemes were completed, at which point the first phase of the experiment was done. In the second phase, the user was asked to review each step of their navigation through the browse schemes, in the context of which resource they were looking for. They were asked to select the reason for taking that step in the navigation,

⁸For those that have high hopes for making use of the Dublin Core `subject` field for organization, we discovered that in practice, this field was highly inconsistent. It contained a mix of controlled and uncontrolled text, as well as widely varying ideas of granularity. When the content of `subject` was controlled, often the source vocabulary was unclear. Also varying were whether multiclassification was employed, and whether the source taxonomy was flat or hierarchical. Because of all this, we concluded that we would have to treat the field as uncontrolled text to make use of the information in it.

from a radio selection of five possible reasons (discussed in Section 5.1). Finally, they could enter in a few free-form comments about the study in general.

A few experimental design aspects are important to note. Firstly, the schemes were presented in a random order to each user, and resources were randomly selected within each scheme. This enabled us to capture learning effects *within* each scheme as the user progressed with it. However, because we could not bring the users into a controlled environment, we could not compel them to finish the study. The randomized presentation order of the schemes addressed this problem, ensuring that browse coverage of the schemes would be unbiased. In addition, the random selection of resources ensured that no particular items from the collection received any emphasis. Further, this random selection was balanced across subcollections, so that there was no bias towards any one topic or particular style of metadata encoding. Finally, the ability to give up on tasks was counter-balanced by the requirement that the user still successfully complete four tasks within the current scheme, ensuring we would have enough completed task data while simultaneously providing for an important aspect of the real-world setting.

For the study, we had 144 participants from Emory University. Two were faculty (1.4%), 16 were grad students (11.1%), 28 were staff (mostly library, 19.4%), 96 were undergrads (66.6%), and two did not specify their user class. Only 36 users completed all 16 browse tasks (25%). Only 54 completed at least 4 tasks (37.5%). However, we were able to glean useful data from all participants, regardless of how much they did, as discussed above.

5. RESULTS

In this section we present the results of our experiment. We believe that the real-world efficacy of a browse taxonomy cannot be summed up in a single metric, so we’ve identified a number of “dimensions” of performance and summarized the data for each.

5.1 Classification Accuracy

Classification accuracy is a well known kind of metric, used to measure how well an automated system places resources into a classification scheme, relative to some notion of their “correct” placement. This notion typically stems from expert or specialist judgments, as embodied in manual category labels applied to a test set of items. In our case, we lacked a substantial test set of this sort, as well as the expert time to create one. Although we had a training set for our classifier, none of the training items were actual resources from the digital library.

Instead, we formed our classification accuracy test set *post hoc*. As mentioned before, in the second phase of our experiment, we recorded user judgments of *why* each navigation click was performed. This became the basis of our evaluation set using the following methodology:

1. Whenever a user selected the first reason (“I thought the category was about this resource”), this was interpreted as a category label for the (scheme, category, resource) triplet. *Note that there could be many of these labels for the same resource within a scheme, with the same or varying categories.*

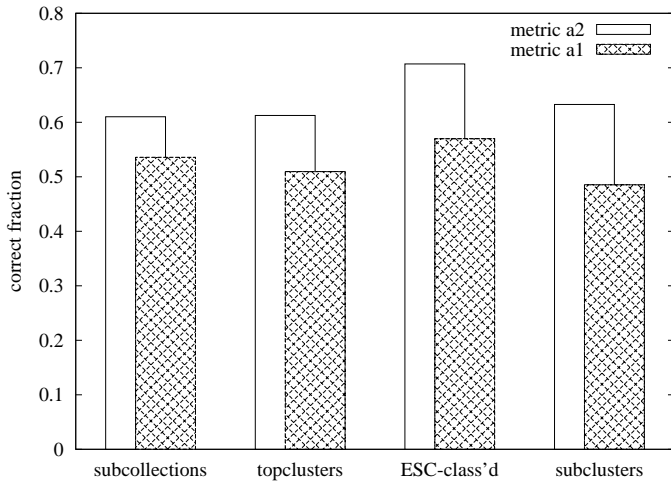


Figure 1: Scheme classification accuracy

- For each scheme, the set of resources included in such judgments was iterated over.
- The system’s placement of the resource for that scheme was compared to these judgments. For metric $A^{(1)}$, the fraction of judgments matching the system classification was used. For metric $A^{(2)}$, we used a value of 1 if *any* category in a label specifying the same scheme and resource matched, and 0 otherwise (the motivation for this relaxed criterion is that we cannot objectively say which user’s judgment is the correct one).
- The above values were averaged over the number of distinct resources specified in user judgments for that scheme.

The classification accuracy using metric $A^{(2)}$ was quite decent for all of the browse schemes (see Figure 1), with subcollections coming in last by a slight amount. Subcollection clusters broke out of a 3-way tie by a slight amount. The classification-based scheme performed the best by a noticeable margin of almost ten percent over the other schemes.

In the stricter metric $A^{(1)}$, nearly the same pattern holds, though the clustered schemes do slightly worse than the trivial scheme. Note that we can think of the divergence between $A^{(1)}$ and $A^{(2)}$ as a kind of “confusion” metric, which indicates how much disagreement there was in user intuitive judgments of classification. Proportionally speaking, this confusion is slightly greater for the AI schemes. It also seems to be the cause of the reduced performance of the clustered schemes in metric $A^{(1)}$.

These results show that only the ESC-classified scheme was significantly better than the trivial scheme in the classification sense. They also show that disagreement on placement is more of an issue for the AI schemes, which we would expect from the fact that they are the only subject-based schemes.

However, classification accuracy is a metric which doesn’t entirely capture the real-world aspects of elapsed time and total number of clicks, as well as other aspects we discuss below. In addition, the theoretical weakness of this metric is exposed by the simple observation that different users will select different categories for the same resources. There are many “right” answers in a specific instance; thus it probably

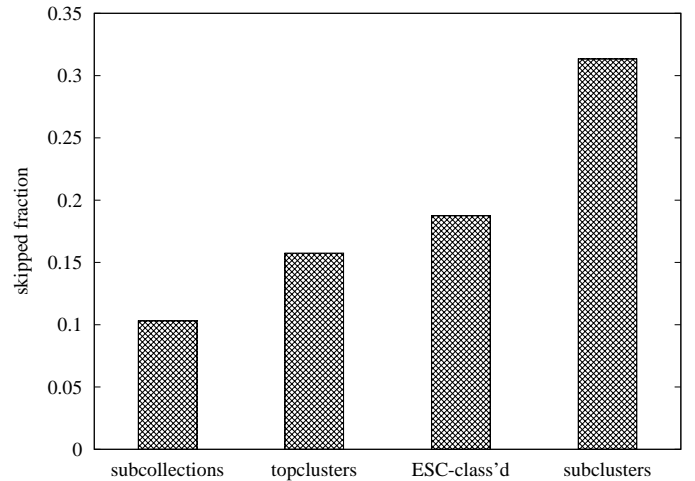


Figure 2: Proportion of browse failures per scheme.

makes less sense to look at *classification instances* than it does to look at the performance of *browse tasks* over the whole scheme.

5.2 Browse Failures

In this section we look at browse failures for each scheme, a simple metric which is based on the proportion of failed browse tasks compared to total browse tasks. A browse task was considered as “failed” when the user skipped it, giving up on trying to find the designated resource.

A chart of the of failed browse tasks is shown in Figure 2. Surprisingly, the trivial scheme suffers the least from browse failures, at about 10%. The clustering and classification schemes are significantly worse in this respect, though both come in under 20%. Finally, the two-level subcollection clusters does quite poorly, with about a 30% failure rate.

However, there is more to overall quality and user satisfaction than the fraction of failed browse attempts. This ignores the aspect of how well the browse experience went for the majority of *successful* browse tasks. We look at this in a variety of ways in the following sections.

5.3 Elapsed Time

Elapsed time is simply the wall-clock time spanning the browse task: from when the user was assigned the resource to when they found it (or gave up on the task). In our view, elapsed time is one of the most important metrics, if not the most important performance metric, as it corresponds to the kind of efficiency that often matters most (i.e., “time is money”).

One general caveat to keep in mind with this elapsed-time analysis is that our investigation employed the browse schemes in a way that didn’t test the free exploration mode of use of a taxonomy. In this mode, a user taking more time might actually indicate the scheme is doing a better job.

In Figure 3 we present a chart of average elapsed time for browse tasks, broken down by scheme. The chart is further broken down by all, completed, and skipped browse tasks. There are a few interesting things about this chart. Firstly, all of the automatically-derived schemes do worse than the trivial scheme, for all counts except skipped tasks. Here,

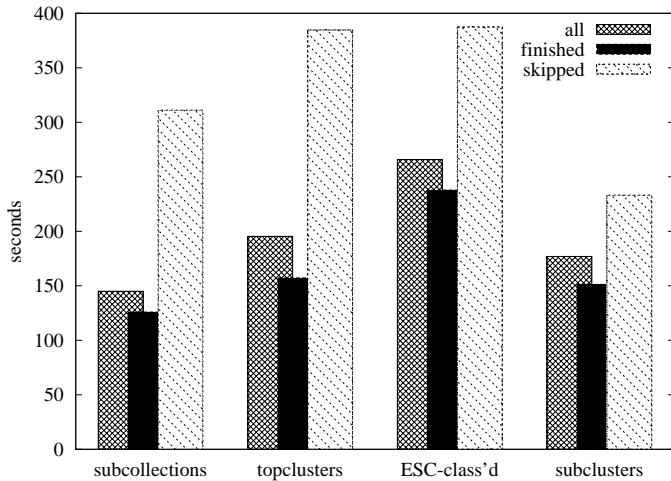


Figure 3: Average time (in seconds) to conclude browse tasks.

subcollection clusters had a lower elapsed time, most likely because the two-levels of the scheme caused users to give up more quickly.

Interestingly, subcollection clusters did as good or better than both of the other automatically-derived schemes for finished and all browse tasks, indicating users found what they were looking for quicker, when they did not give up. Finally, top-level clusters basically tied ESC-classified browse tasks for elapsed time in skipped tasks, but beat it by a significant margin for all and finished tasks.

5.3.1 Learning Effects

A legitimate observation is that, while a taxonomy may be difficult to use and inefficient when it is first encountered, it may eventually become very efficient as the user learns it. In fact, this effect should be present to some extent in every situation where a taxonomy is employed. Thus, averages of efficiency metrics like elapsed time may not tell the whole story — we also want to see how efficiency progresses as users get more familiar with a scheme.

Recall that we organized the experiment so that each user would receive a random sequence of schemes, but within each scheme, would have a series of consecutive browse tasks. This allowed for learning effects *within* each scheme, but prevented any bias towards learning effects *between* schemes. In Figure 4, we give a graph of average task completion elapsed time by task ordinal index. In other words, the first location on the horizontal axis is the first task, the second is the second task, and so forth⁹. In Table ??, we use this data to produce a summary “learning rate” score.

We can see from this information that learning effects were indeed present. Some schemes, such as ESC-classified, started out quite poorly, but users learned to compensate fairly quickly. The other schemes had a more gradual de-

⁹Note that these plots extend past four tasks. This is because skipped tasks are counted, and users received more browse tasks until they successfully completed four of them. Thus, schemes which had a higher prevalence of failed browse tasks will have plots that extend further toward the right. We count skipped tasks because the browsing that took place leading up to them still influences learning.

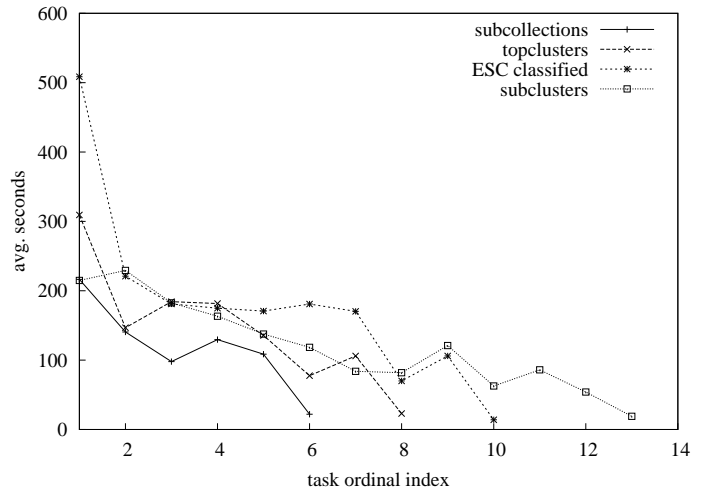


Figure 4: Average time to complete browse tasks (seconds), by task ordinal value. Linear regression through these curves yields learning rates (in seconds per index) of 29.61 for subcollections, 30.64 for top-level clusters, 35.23 for ESC-classified (best), and 15.44 for subcollection clusters.

cline. It could also be said that ESC-classified and subcollections had something of a “plateau” after initial learning, which could indicate a non-intuitive structure. Top-level clusters and subcollection clusters had the most steadily improving learning effects profile.

5.4 Clickstream Volume

Another interesting way to look at the economy of information retrieval through a browse scheme is to look at the number of clicks it took to find a resource (or give up). In Figure 5, we give a chart of the average number of browse clicks per task. For each scheme, averages are shown for all tasks, completed tasks, and uncompleted (failed) tasks, allowing some insight to clickstream volume differences for successful and failure browsing.

This chart indicates that all of the one-level schemes seem to be running close to each other, with ESC-classified doing slightly better for finished tasks. The only two-level scheme, subcollection clusters, does predictably worse, as one must click a minimum of two categories to get to a resource. However, it does not do twice as bad, suggesting the two levels and additional detail are relatively helpful.

Note that in all cases, the overall clickstream volume most closely parallels the volume for failed tasks. The explanation for this is that by far, most clicks occur during browse tasks that the user eventually gives up on. This does not mean that most tasks fail to be completed successfully. As we saw earlier, this is not the case, as browse failures are in the minority for all schemes.

5.5 Navigation Reasons

Recall that our experiment was divided into two parts: browsing, and retrospective. In the retrospective portion, users provided feedback about *why* they performed each navigation event, or click. They were shown the record for the browse task, followed by a list of categories and a multiple-choice selector for each. The five choices available were:

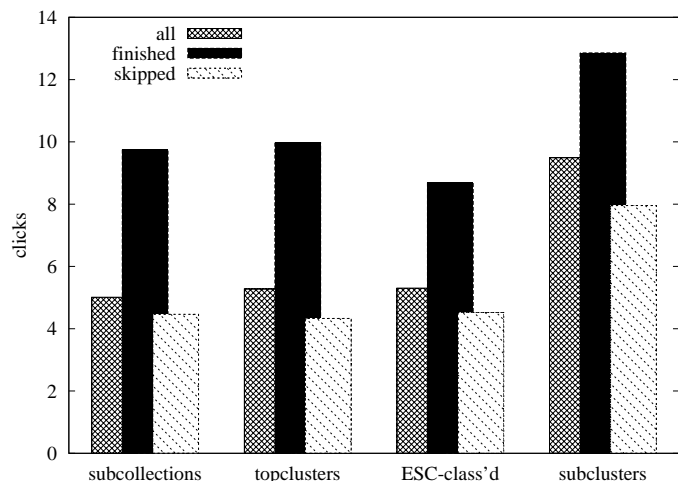


Figure 5: Average number of navigation clicks to conclude browse tasks.

1. “I thought the resource was about that topic.”
2. “I guessed the computer would put the resource in that category.”
3. “The resource had words in common with category.”
4. “Process of elimination/random/didn’t know where to look.”
5. “Mislicked/didn’t mean to click on this category.”

According to our preliminary investigations, these five reasons provide nearly complete coverage of clickstream motivations. The only vagary was that sometimes users thought multiple answers would be appropriate simultaneously; thus we instructed them to pick the *best* reason for simplicity.

With the exception of reason #5, these answers are not value-neutral. We can interpret the reason #1 as the most positive, as it indicates correct classification (both intuition and anticipation). Reason #2 indicates a lack of intuitive placement, but the ability to anticipate it. In other words, there was some understanding of the behavior of the classification system. If it was not intuitive, it was at least logical. Reason #3 is even weaker, indicating a guess by a simple word-occurrence heuristic, which is not necessarily the correct basis of automatic placement. Finally, reason #4 is quite negative, indicating a complete failing of intuition, no ability to anticipate placement, and not even the ability to apply a content-based heuristic to find the item.

In Figure 6, we plot the fraction of each of these types of answers for each scheme. A few patterns clearly emerge. The first is that guessing was clearly the dominant navigation type. We are unable to comment on this in an absolute sense. However, a clear sub-pattern of less guessing within the automatically-derived schemes is visible. In addition, both appropriate placement and anticipation of the system (reasons #1 and #2) rise for the automatic schemes. Finally, the simple heuristic navigation method (reason #3) falls for these schemes.

These results suggest that the automatically-derived taxonomies are more useful as browsing schemes, due to an increased prevalence of positive modes of navigation, and decreased prevalence of more negative ones.

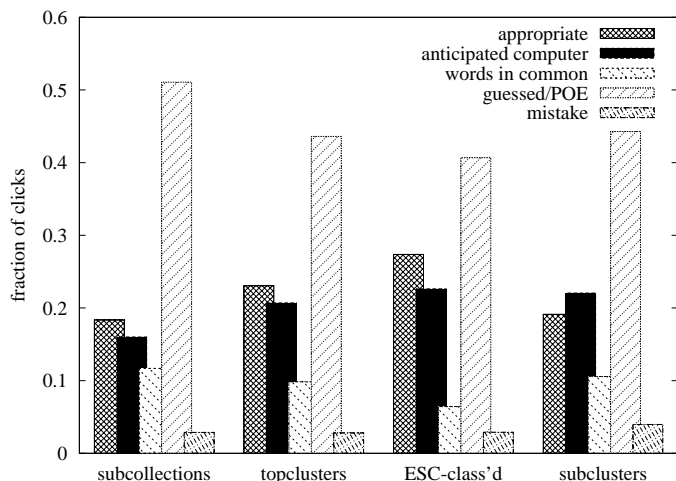


Figure 6: Navigation clickstream reasons

scheme	ratio
Subcollections	.5
Topclusters	.36
ESC-Class'd	.59
Subclusters	.35

Table 1: Ratio of positive to negative free response comments for each scheme. Higher is better.

5.6 Free Response

At the end of each scheme, users were given a chance to enter in a free-form response, giving their impressions (both good and bad) of how easy the scheme was to use. This is exactly what many users did, and we received 30-40 non-null responses for each scheme. Some of the responses were purely emotional (such as “I liked it,” “I hated it,” “It was great,” etc.), some were highly technical in the way they were complementary or critical, and some were basically irrelevant (commenting on the interface or aspects common to all schemes).

To get a picture of the overall gist of each scheme, we went through all of the comments and counted each one as either negative, positive, both, or neither¹⁰. We kept a running tab of negative and positive comments for each scheme (with a single comment sometimes appearing in both or neither of the columns), and in Figure 7, we graph these counts, normalized by the total number of non-null responses for each scheme. In Table 1, we give a table of the ratio of positive to negative comments for each scheme.

From this, it is clear that the ESC-classification scheme was most-loved as a browse taxonomy; it contained both the highest number of positive comments and the lowest number of negative comments. It is also notable that subcollections do so well in this sense; better than the two clustering-based schemes.

To give a taste of individual user reactions to the schemes, in the following subsections we present excerpts from se-

¹⁰Such a classification is necessarily very rough, however, we have no reason to believe our classifications were uniquely biased.

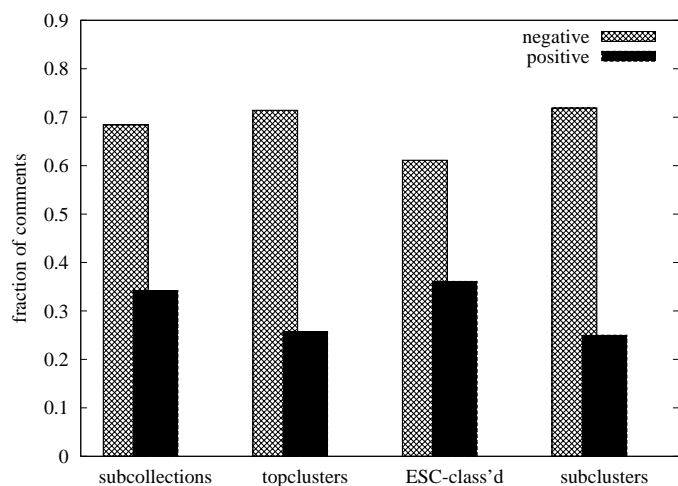


Figure 7: Overall user sentiment for each scheme, as a proportion of non-null feedback free responses.

lected comments. These comments highlight many important issues and problems with the schemes. We have tried to select comments that were less ambiguous for presentation here; however, some might express both positive and negative sentiment.

5.6.1 Subcollections

Positive comments:

- “[Categories] were more precise [than ESC] but more overlapping. ...”
- “The easiest set [of all of the schemes] (not counting misclicks), probably because of the inclusion of similar access points in both the description of the item to be found and in the various browsing categories.”
- “Easier than [ESC], but still not ideal. Some of these collections, like Atlanta History Center, have lots of eclectic stuff that isn’t hinted at by the collection name.”

Negative comments:

- “Browsing by collection was the most difficult and least intuitive method [of all of the schemes].”
- “It was easier [than the rest of the schemes], but only because the topics this time happened to have state names in them. If there is no reference to a state, it is almost impossible to find something unless you have other background info.”
- “When assoc w/a location, easy.”

The pattern which surfaces here seems to be that our subcollections, which had a high geographical bias, were quite easy to use for browsing when the geographical component of a record was known. However, in other cases, retrieval was much more difficult. The collections (like Atlanta History Center) which lacked a geographical bias were difficult to use through this taxonomy. The dominant positive impact seems to stem from the preponderance of geographically-slanted records in combination with geographically-segregated subcollections.

5.6.2 Top-Level Clusters

Positive comments:

- “Found most of the items in this section on the first try, but it was still slow-going. The categories were vague.”
- “This [browse] scheme was much simpler than [ESC], as the subjects around which it was grouped were more specific to the material. There were still many subjects which could have been consolidated probably ...”
- “Most of them were fairly easy to find, but documents that could be in multiple categories took a long time to find.”

Negative comments:

- “It’s still difficult to know where to start the search, because the category that seems most logical is often not the correct one. ... Sometimes the topic you want is buried under a category title that doesn’t seem to have anything to do with it. And items that are similar seem to be split up into different categories for reasons that are not really clear.”
- “Overall, the categories are not so appropriately named which makes it frustrating to find what you’re looking for.”
- “The trick seems to lie in dealing with overlapping subject categories.”

There is evidence that the organization by inherent subject of the top-level clustering scheme, rather than source collection, is appreciated by users. However, a few problems are clear. One is that the cluster names only roughly correspond to their content; an effect which we propose is due more to the weak coupling of some records to the clusters than any weak coupling of descriptive terms to the clusters. In other words, classification confidence is relatively low for the vast majority of resources in a cluster, despite the fact that it is higher for that cluster than any of the others. This creates the dual effect of users feeling the cluster is both “too vague” and also poorly-named; two sides of the same coin.

Yet at the same time, users experience clusters as “too specific,” focusing on extremely narrow topics and aspects of topics, such that many of them could be combined. For example, we had many top-level clusters related to the civil war, yet many other topics remained “buried” throughout the other clusters.

The presence of both “too vague” and “too specific” complaints reflects the tug-of-war we experienced in selecting the optimal number of clusters for a non-hierarchical set. Since there was no way to automatically balance the detail of the clusters, in some places detail was lost, while in others, it was too refined.

5.6.3 ESC-Classified Taxonomy

Positive comments:

- “The subject [categories] are wonderful, and make so much more sense than the [subcollections]. At least at the initial search the information is so much more important than the originating institution. I had almost no trouble at all.”
- “These category lists are much more straightforward than the [subcollections], though it’s still not entirely intuitive which category to search for the item.”
- “Was not intuitive, but went quicker than [subcollection clusters].”

Negative comments:

- “This final scheme was helpful when the placement of the item was evident, but generally, these categories were far too broad, and it was conceivable to place the item in several categories.”

- “Some of these are VERY difficult to locate. I’m not sure it is because I’m not very familiar with southern or civil war history, or if it is because I’m not making the correct connections with the descriptor phrases and categories.”
- “Compared to [subcollections], this one was very difficult. Arranging the databases according to subject rather than according to collection was difficult because the articles for which one looks could plausibly be categorized under many of the subjects.”

As we showed earlier, sentiment was the most positive for the ESC-classified scheme. The categories were considered much clearer, though still not entirely intuitive for the subject area. This is a result of the fact that the ESC “classification scheme” really was never intended to be such, it is instead an *a posteriori* grouping of articles received for an encyclopedia. Hence, much of the improvement over clusters can likely be attributed to its relatively uniform granularity, lessened duplication, and improved classification to within the categories. However, the negative comments show that the ESC scheme suffers from the same sort of problems as the clustered scheme.

5.6.4 Subcollection Clusters

Positive comments:

- “Overall ... I think the [sub-categories] provided this time around was more helpful than not.”
- “This scheme, with its second tier of categories, was the most helpful search tool, [better than topclusters and subcollections].”
- “This was the easiest scheme [of all]. The descriptions on the items gave enough clues to find the right category.”

Negative comments:

- “The browsing categories at the start are too scattered and unorganized.”
- “This one was arranged more simply than [topclusters], though the fragmentation of information that could have been put together was frustrating...”
- “This was the worst [cluster-based] scheme by far. ... Who in the world would organize [this way]?”

This scheme seems to have “broken even”, relative to both the clustered scheme and the trivial scheme. In some cases, it was an improvement, due to good clustering within subcollections. However, when the user did not know which subcollection (top-level category) to select, their retrieval task got harder by an order of magnitude. The main lesson to learn here seems to be that, in a hierarchical scheme, it is extremely important for the scheme’s accuracy to be near-perfect at non-leaf levels, lest the effect of a failure become compounded.

5.7 Summarized Results

In this section we give a side-by-side comparison of the examined browse schemes, considering each investigation as corresponding to a different performance metric. To produce an actual metric from each of these dimensions of quality, we scored each scheme in the most natural way (i.e., produced a single numerical value from the experimental data), and then ranked them all from one to four. Table 2 gives the results of this in tabular form. We urge the reader to take this table with a very large grain of salt, as the rankings mask the actual *extent* of the differences between schemes. We have included an “average” column for completeness;

scheme	metric						
	<i>c</i>	<i>b</i>	<i>t</i>	<i>l</i>	<i>r</i>	<i>s</i>	avg.
Subcollections	2	1	1	3	4	2	2.2
Topclusters	3	2	3	2	2	3	2.5
ESC-class’d	1	3	4	1	1	1	1.8
Subclusters	4	4	2	4	3	4	3.5

Table 2: Summary of scheme results, as ranks for the different performance metrics. The metrics are given by single-letter codes which are *c* for classification accuracy, *b* for browse failures, *t* for time elapsed, *l* for learning effects, *r* for navigation reasons, and *s* for free-response sentiment.

however, its practical use is limited by its lack of weighting given the needs of a particular scenario. Still, the table does suggest that AI-enriched schemes can be at least as useful as subcollection-based organization, both in particular respects and overall.

6. DISCUSSION

The various perspectives presented above on the efficacy of the four browse schemes each tell a slightly different story. Each scheme seems to take its turn at performing well in one of the metrics. Some of the differences in results likely stem from the qualities of the different resources assigned to users, in combination with the schemes they randomly fell under, further in combination with the character, mood, and experience of the user. Indeed, it is clear that there is no one right answer to the question of which scheme is “best”. Rather, different schemes appear to be better for different modes of use, which emphasize the different aspects we examined.

Our AI-enriched schemes “held their own” against the trivial scheme, with the ESC classification-based scheme even placing as “best overall” in our rough rankings. However, the bias in our collection towards the geography of the American South ensured that subcollections remained unusually useful as an organizational tool, which in turn boosted the performance of the trivial scheme. We do not suspect that this will necessarily be the case for other subject portal digital libraries, so we would expect to see a subcollection-based scheme do worse there as compared to clustering or automatic classification-based schemes.

While our clustering-based schemes did fairly well under some metrics, we think that that some of the performance shortfall was due to their nascent form. More could likely be done to improve our clustered taxonomies greatly, such as noun-phrase parsing, the addition of “miscellaneous” clusters to gather “unclassifiable” items, better tuning and thresholding, and perhaps most importantly, true hierarchical clustering. We plan to investigate all of these improvements in future work.

There is also another way to improve the automatic taxonomies: improved metadata. In the course of our study, we found that poorly-encoded records were a major source of semantic confusion. Without clear field semantics, content-based organization techniques have much of their foundations undermined. Although all of our records were Dublin Core, a major source of confusion lay in the interpretation of certain fields, especially **description**. We would like

to see **description** used purely for the natural-language articulation of the *topic* of an artifact, rather than meta-commentary about its presentation, location, provenance, or kind¹¹.

To get to the level of metadata quality we attained for our AI-enriched schemes, we had to perform a number of relatively labor-intensive clean-ups of content, such as producing lengthy stoplists and special text-cleaning routines to remove the most frequent boilerplate segments. Cleaner metadata would reduce this kind of preparation and speed up the deployment of automated browse taxonomy methods, as well as improve the quality of the output.

Finally, we must emphasize that our study should only be taken as a first step toward user study-based evaluation of AI browse taxonomies, because we examined known-item retrieval using browsing rather than searching (and did not have a search service available at all). In an actual digital library, we would encourage users to use search engines when they think they know the item they want, in addition to when they are becoming frustrated with the browse taxonomy and require another method. However, we do not think this undermines the usefulness of our results, because known-item search via browsing is “stricter” than free exploration. In other words, if a scheme performs well for known-item retrieval, it should perform as well or better for free exploration.

7. CONCLUSION

We have found that automatically-derived browse schemes at least match, and in many senses surpass collection-of-origin (subcollection)-based organization of browse resources in portal-style digital libraries. Due in part to variation in user characteristics, information needs, and collection characteristics, we believe that these schemes could already make a productive addition to real-world digital library systems.

We also think that it is likely machine learning techniques will continue to improve in the near future, hence improving the information organization possible in clustering and classification-based browse schemes.

Even without such progress, the utility of AI-enriched schemes will surely diverge upward from subcollection-based organization: as the number of collections integrated into a given portal grow, the topics latent within them will reach a point of saturation, whereas the quantity of categories (subcollections) by definition grows without limit. This will lead to a high degree of topical scatter, with subcollections becoming less and less meaningful for organization. AI-enriched schemes, on the other hand, are not subject to the same scalability problem.

We also think that the need for AI-enriched schemes will increase by the simple fact that more collections are becoming federated, and the sharing of free digital content and metadata catalogs is becoming more prevalent. We think the results presented here provide an encouraging first look at a new option for these kinds of efforts, showing that machine learning methods can give them a way to provide use-

ful browse services over heterogeneous content without the traditionally large amount of manual effort.

8. REFERENCES

- [1] Baker, L., McCallum, A. Distribution clustering of words for text classification. In *Proceedings of ACM SIGIR*, 1998.
- [2] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., Harshman, R. A. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391-407, 1990.
- [3] Garner, S. R. WEKA: The Waikato environment for knowledge analysis. In *Proceedings of the New Zealand Computer Science Research Students Conference*, pp. 57-64, 1995.
- [4] Joachims, T. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of The 10th European Conference on Machine Learning*, pp. 137-142, 1998.
- [5] Kaufman, L., Rousseeuw, P.J. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.
- [6] Koller, D., Sahami, M. Hierarchically Classifying Documents Using Very Few Words. In *Proceedings of the International Conference on Machine Learning*, pp. 170-178, 1997
- [7] Lee, D., Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, vol. 401, October 1999.
- [8] Liu, A., Gong, Y. Document clustering with cluster refinement and model selection capabilities. In *Proceedings of ACM SIGIR 2002*, Tampere, Finland, Aug. 2002.
- [9] McCallum, A. K. *Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering*, 1996.
<http://www.cs.cmu.edu/~mccallum/bow>.
- [10] Salton, G., McGill, M.J. *An Introduction to Modern Information Retrieval*, McGraw-Hill, 1998.
- [11] Sebastiani, Fabrizio. Machine learning in automated text categorization, *ACM Computing Surveys (CSUR)*, 34:1, 2002.
- [12] Soergel, Dagobert. Thesauri and ontologies in digital libraries. *JCDL 2004 Tutorials*, Tucson, AZ, 2004.
http://www.dsoergel.com/cv/B63_rome.html.
- [13] Willett, P. Document clustering using an inverted file approach. *Journal of Information Science*, 2:223-231, 1990.
- [14] Xu, Wei, et al. Document Clustering Based on Non-Negative Matrix Factorization. In *Proceedings of SIGIR 2003*, pp. 267-273, 2003.

¹¹For example, the **description** field for a portrait photograph should describe who is in it and perhaps where they are, while leaving photographic medium, method, and creator to other fields. Sometimes the “pollution” of the **description** field was unusually bad, containing boilerplate text from the originating institution or electronic access information.