Change Detection and Classification of Digital Collections

Sampath Jayarathna
Department of Computer Science
California State Polytechnic University
Pomona, CA 91768
ukjayarathna@cpp.edu

Faryaneh Poursardar
Computer Science and Engineering
Texas A&M University
College Station, TX 77843
faryaneh@tamu.edu

***Abstract*—People develop personal information collections consisting of distributed web resources as both reminders that resources exist and to provide rapid access to these resources. Managing such collections is necessary to preserve their value. Unexpected changes within distributed collections can cause them to become outdated, requiring revisions to or removal of no-longer-appropriate resources and replacements for lost resources. In an effort to alleviate this problem, this paper presents a categorization and classification framework including a tool that supports the management and active curation of distributed collections of Web-based resources. We assess the need for such a system and analyze how current tools affect the management of personal collections with survey of 106 participants from online and offline communities. Results of the survey show that personal collections are common and collection management is an issue for ~20% of respondents. Additionally we examine and categorize the various degrees of change that digital documents endure within the boundaries of a distributed collection. Consequently, this paper will focus on two research questions. First, what facets of the change detection process can be automated? And second, looking at this problem from a user standpoint where each document contributes towards the overall meaning of the collection, what strategies can be used to effectively detect the consequences of the various types of change found in document collections?**

Keywords—Personal Digital Collections; Change Detection; Classificaiton

I. INTRODUCTION

Bush's Memex [6] and its associative trails offered a vision of how digital collections could take form. However, *As We May Think* failed to anticipate some of the challenges and difficulties associated with preserving and curating digital collections nowadays. For example, curating or maintaining a digital collection is not easy: selecting, organizing and contextualizing the resources in a collection are tasks that require significant effort from a curator. Moreover, curators' efforts do not cease once the resources have been added to a collection: it is often the case that a curator must also keep track of the resources to ensure that the collection remains valuable over time.

To make matters worse, there is a specific type of digital collection where looking after the consistency in its documents has a more crucial role. These collections are known as *distributed*, which means the administrative control of information related to a topic may be spread across other digital collections maintained by multiple scholars in multiple institutions. This administrative decentralization leads to changes that are unexpected by the maintainer of a collection. While most digital collections have some form of change via creation and deletion of resources, *distributed digital collections* made up of resources that are distributed across the Internet undergo additional kinds of change. These collections are brought together via hyperlinking, and there is no central curation of the collections. Also these distributed collections may bring together resources that are expected to remain as is (e.g. a description of different types of clouds) with resources that are expected to change as time goes on (e.g. a weather forecast.)

In addition to expected changes in content, unexpected changes in content and accessibility can be caused by different factors or circumstances. Changes can manifest because of deliberate actions on part of the resource creator/manager– for example, reorganization of the structure of the content, switching to a different content management system, or changing jobs and institutions. Change might also be due to unexpected events – earthquakes, power outages, disk failures, – or may be due to other uncontrollable factors –death, seizure of computers by law enforcement, or termination of the services from an Internet Service Provider.

Therefore, our work has been motivated to mitigate the impact of fluidity of web pages [5] that leads to collections becoming stale and requiring revisions and updates. This paper describes the software structure that we have developed to cope with these challenges and how we can categorize and use automatic classification in the framework. To understand these challenges, we first conducted a survey of potential users to elicit whether they create such collections and, if so, what technologies/tools are used to create and maintain their collections. We then developed the software infrastructure for managing distributed digital collections and change detection.

978-1-4673-9005-7/16/$31.00 ©2016 IEEE

Taking into account previous work, there are two questions that remain need to be addressed. First, what facets of the change detection process can be automated? This point becomes increasingly relevant when taking into account that the resources found in digital collections are often curated and maintained by experts with affiliations to professionally managed institutions. And second, what strategies can be used to effectively detect the consequences that the various amounts of change introduce into a digital library (DL) environment? From a user standpoint, this question has great relevance when considering that each document in a collection contributes towards its overall meaning and that a document that has undergone unexpected change can potentially interrupt the flow of a collection making it semantically incomplete.

We will address these questions in the following sections of this paper: Section 2 describes the related work; Section 3 presents the system architecture; Section4 and 5 presents the categorization of degree of change and how the web resource features used for resource classification; Section 6 describes the dataset and analyzes classification and survey results. Section 7 discusses lessons, implications, and conclusion.

II. RELATED WORK

Bookmarks have long been used as "personal web information spaces" to help users to remember web resources and retrieve interesting documents [25]. Li et al.[19] found that web users would like to build, organize and revisit a larger collection of bookmarks for future references than they can reasonably maintain now. Despite well-set guidelines for creating web resources [4], missing or misplaced web pages remain when dealing with references to these external resources. External resources on the Web are highly volatile and prone to be affected by unexpected change that can manifest as cases of "broken links" [15] or "link-rot" [30]. Web documents are not static resources and a certain degree of change is expected from them. However, as a member of the collection, these documents are expected to either change little over time or mutate harmoniously and accordingly with the other documents in order to preserve the semantic meaning and systematic order of the collection.

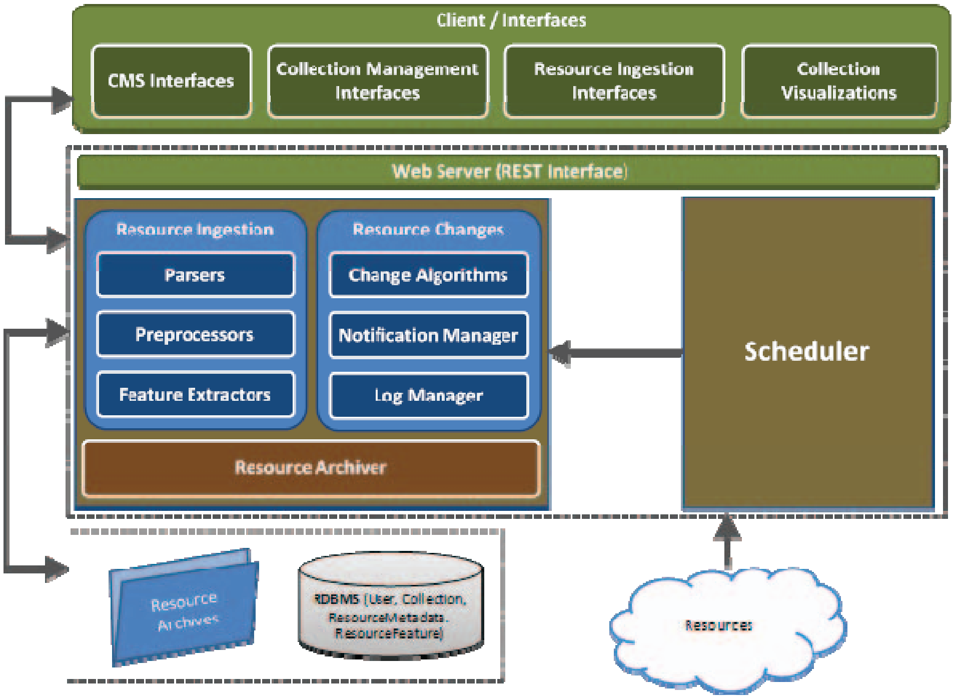

Fig. 1. Digital Collection Manager Architecture and Components including Resource Changes

Previous work on finding missing resources is based around the premise that documents and information are not lost but simply misplaced [2] as a consequence of the lack of integrity in the Web [1, 8]. Other studies have also focused on finding the longevity of documents in the Web [13] and in distributed collections [17, 29]. Phelps and Wilensky pioneered the use of lexical signatures to locate missing content in the Web [28]. They claimed that if a Web request returned a 404 error, querying a search engine with a five–term lexical signature could retrieve the missing content. Park et al. used Phelps and Wilensky's previous research to perform an evaluation of nine lexical signature generators that incorporate term frequency measures [27]. Additionally, Klein and Nelson have extracted lexical signatures from titles and backlinks to find missing Web resources [14].

Dalal et al. used a different method to find appropriate replacements for missing resources from the Web that belonged to a collection in Walden's Paths [7]. Their approach was based on a two–step process. First, metadata was extracted when the path was created thus preserving the author's intent and vision. Second, the extracted metadata was used to find pages when they cannot be retrieved. In the specific case of collections such as Walden's Paths, each node in a path is destined to make a contribution towards the overall concept and the continuity in the narration. Therefore, finding replacements becomes a critical factor to maintain the integrity of the collections and preserve their semantic meaning.

On the other hand, previous work on link persistence has focused on characterizing the availability of resources over time. Nelson and Allen measured the persistence and availability of documents in a digital library [24]. Koehler found that specialized document collections – such as legal, educational and some scientific citations – tend to stabilize over time [16]. However, citations in some domains have higher rates of failure [10]. McCown et al. also explored other factors that might cause a resource to fail by examining its age, path depth, top-level domain and file extension [21].

More so, the framework that we will describe in this paper builds upon these solutions but has some key differences: most notably, the coupling of natural language processing methods with a user interface that can handle large and highly dynamic collections.

III. SYSTEM ARCHITECTURE

Fig. 1. illustrates the architecture which comprises three main layers: the application layer, the service layer, and the storage layer. The application layer is home to all the user interfaces to the system. This includes interfaces for importing resources into collections, examining the status of a collection, and visualizing collections. All these interfaces interact with the service layer. The service layer is a REST web service that encapsulates the modules of system engine. The resource ingestion module handles the persistence of the resources submitted either by the user or the scheduler. First, each resource is parsed into metadata (e.g. location, associated collection) which is stored in a database. Then the module processes the resource by downloading its content and parsing it. Next, the content is sent to the feature extractor which computes and stores in the database all the features that will be used to compare different versions of the resource. The content is also sent to the resource archiver that ensures a persistent copy of each retrieved version.

Once the resource information has been stored, the scheduler module regularly polls the resource for a new version. Each poll checks whether each resource is alive, i.e. still available. If the connection is successful, the scheduler invokes the ingestion module to build and store a new version of the resource. Then the scheduler also invokes the resource change module to compute the differences between the new version of the resource and previous versions. Thus, the resource change manager in the service layer computes and records the differences between two versions of a resource. The differences are based on either the content or the features extracted from the resource (see section 4 and 5 for further discussion on categorization of changes and classification). Finally, changes to resources are reported to collection managers depending on the configuration of the resource. If the user has labeled the resource dynamic (i.e. he expects the content to be constantly changing) then notification occurs if no significant change is found or vice versa if the resource was labeled static.

IV. CATEGORIZATION OF CHANGES IN COLLECTIONS

To conduct our experiment on change detection algorithms, we needed a document corpus. For this purpose, we harvested the conference proceedings found in the Association for Computing Machinery Digital Library. While the ACM Digital Library stores and maintains the “Full text of every article ever published by ACM and bibliographic citations from major publishers in computing, it includes the links to the actual conference sites as distributed resources hosted externally and therefore more prone to be affected by unexpected change.

We then proceeded to inspect and categorize the 1492 pages that were retrieved with a 200 HTTP response code. We categorized these pages into three categories by evaluating the relationship between the anchor text and the corresponding retrieved page. As a result of this categorization, we found that 917 pages were “clearly correct” and 531 were incorrect. Additionally, we were unable to evaluate 44 pages because their contents didn’t provide us enough information to make an accurate assessment. These pages could have been placed into the “incorrect” category, but we decided to use an additional category to make our experiment as transparent as possible. Fig. 2. shows this classification.

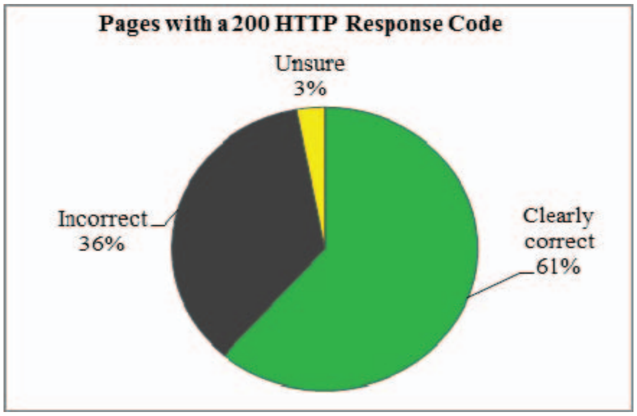

Fig. 2. Distribution of the pages that were retrieved with a 200 (OK) HTTP response code

Then we proceeded to classify the 531 incorrect pages in an effort to understand how conference sites degrade over time. Nine categories were used to classify the “incorrect” pages, which we list in order of severity. These nine groups provide insight regarding the different stages that conference pages go through until they are ultimately abandoned:

1. **Kind of correct:** (197 entries) Pages that contain related content, but they do not fully match the semantic concept encapsulated in the anchor text. When taking into account conference proceedings, these pages often link to a different year in the conference series. For example: Anchor text “Conference X 2006” references the Conference X 2009 site.
2. **Blank pages:** (141 entries) pages that returned no content.
3. **Pages in a different language:** (32 entries) Pages that didn’t match the language found in the anchor text. Most of these pages were in a language different than English.
4. **Failed redirects:** (30 entries)
5. **Directory listings pages:** (18 entries) Pages displaying a listing of files or a “Hello World” page. Probably caused by an error in the server configuration.

6. **University/institution pages:** (36 entries) This case that surfaces when a site has been taken down, but the server configuration redirects the user to its parent institution. In cases dealing with conference sites, servers would usually redirect the user to the website of the University that hosted the conference or to a related professional organization.
7. **Domain for sale pages:** (17 entries) Pages that indicated that the domain name registration has lapsed and it is being sold by a registrar, or taken over by a third party in order to profit from the sale.
8. **Error pages:** (17 entries) Pages that specifically state that an error has occurred.
9. **Deceiving pages:** (43 entries) Pages that have been taken over by a third party. The content displayed in these pages is totally unrelated to the original purpose of the site. We believe that these pages were not created to deceive users, but as an attempt to manipulate the PageRank algorithm [26].

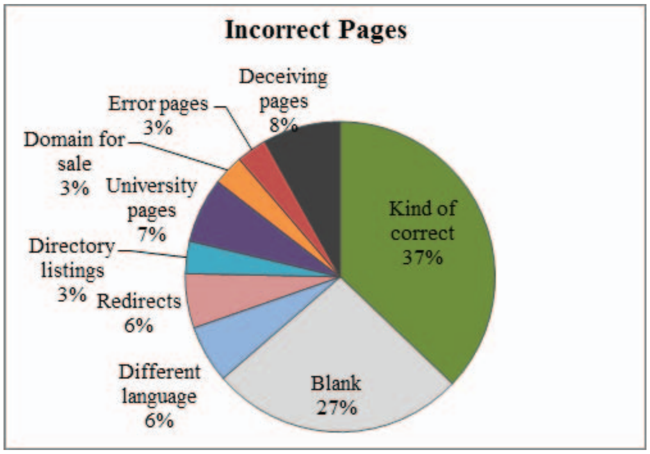

Fig. 3. Distribution of the incorrect pages

Fig. 3. shows the overall distribution of the incorrect pages. Many of these links still lead to information related to the original purpose but clearly not to the originally intended materials. There are a number of categories that result when no content is available depending on how the servers are configured – blank pages, failed redirects, directory listings, error pages, and university/institutional pages. The remaining pages are perhaps the most problematic, when the web address has been taken over and is for sale or being used for other purposes.

We next look at classification of these features and to enable tools that focus collection manager attention on the actions by auto classifying the identified categories.

V. Classification of Features

To develop classifiers for the types of problems, we extracted link-based features from the out-links (i.e., links on the page that was returned) and content-based features from individual pages. We analyzed the features from the links, the content from the page containing the links and the pointing page. In this qualitative analysis of the categories of pages in our corpus, we mainly attempt to find discriminative features derived from a combined approach based on link and content analysis to detect apparent categories. For this task, we applied information retrieval techniques that provide us with a set of features about the links and also about their contents. However, we must point out that this qualitative analysis does not focus on the study of the network topology or the link characteristics in a web page.

A. *Link-based Features*

Most of the link-based features were computed for the base-node and are based on the number of out-links in that page. In addition we calculated some of the features for child-nodes that are the valid out-links in these base-nodes.

Degree-related measures. We computed measures related to the *in-degree* and *out-degree* of the base-node. In addition, we also considered the number of internal-links, which is the number of out-links in the base-node pointing towards same host as the base-node, and the external-links.

Link-type: We believe that broken links (error pages) can provide useful information regarding the nature of the base-node. We also extracted MIME links, which are basically links featuring sound, video or image links. We calculated the valid links from the out-degree reducing the number of broken links and MIME links. Furthermore, we also collected information about other types of MIME links from the base-node such as CSS, text/plain and text/richtext. We defined these types of MIME links as "Import links" and they serve the purpose of linking external files attached to the base-node in order to modify the content of the base-node or provide redirects to external pages.

Anchor Text: When a page links to another, the anchor text shows the relevant information of the target page or summarizes this information in a way to persuade a user to visit this link. Therefore, a number of out-links with irrelevant anchor text shows a clear evidence of disagreement between this text and the target page.

Child-node related measures: We also computed the total number of out-links as the sum of the number of out-links from each child-node. In addition we calculated total number of import links from these child-nodes.

Thus, we have in total 13 features from each base-node relevant to the link-based features.

B. *Content-based Features*

Number of images: We counted the number of Images in both the base-node and in the child-nodes.

Child-Node Meta tags: We collected the description, keywords and title from the base-node and from all the child-nodes. We aggregated the metadata related to the page content from all the child-nodes into each relevant content feature.

KL-divergence: We define the following set of KL-divergence similarity features based on the header information from the meta tags and the textual content from the base-node and the child-nodes.

Meta tags: Meta tags provide structural metadata about a particular web page. We used the “description”, “keywords” and “title” from these tags to build a set of content-based features. The combination of these content-based features can be used to compute the divergence between base-node and child-nodes. We have combined the following set of features to create 6 content-based features to calculate divergence using Latent Dirichlet Allocation (LDA) and KL-divergence similarity measure. We combined the resulting content from the “description”, “keywords” and “title” into a single content-feature called “header”.

- Base-node_child-nodes_KLD_similarity
- Base-node_base-node-header_KLD_similarity
- Base-node_child-nodes-header_KLD_similarity
- Base-node-header_child-nodes_KLD_similarity
- Base-node-header_child_nodes-header_KLD_similarity
- Child-nodes_child-nodes-header_KLD_similarity

We have applied LDA to measure the probability distributions of topics of two or more particular content-based features. We then use KL-divergence to compute the divergence between these probability distributions of content-based features.

C. *Dataset for Change Detection Classifiers*

Previous research has shown (specially in web spam detection) that our problem can be modeled as a "binary" classification where the two classes involved are *correct* and *not-correct*. In these binary classification problems a model is built and evaluated in two phases: the training phase and testing phase. In addition we consider our problem as multi-label classification problem by focusing on the incorrect categories. We define this as "category" classification.

To improve the reliability of our classifiers, each evaluation of the learning schemas was performed by a stratified ten-fold cross-validation [18]. For each evaluation, the dataset is divided into ten equal folds and is trained ten times. Each fold is evaluated with a classifier that was trained with the other nine folds.

The kind of severe imbalance in a dataset shown in Fig. 4. will lead to poor classification results without any data rebalancing [9, 12]. Under sampling of the majority category is preferred compared to over sampling of minority categories because over sampling leads to over fitting [9]. However, under sampling has the drawback of under fitting for the majority category (correct category) due to possible loss of valuable information. This is not a serious problem in our case as our priority is to identify the pages in the incorrect categories more accurately. To train the classifiers, random under sampling was used to select a number of data instances of the majority class to balance the dataset.

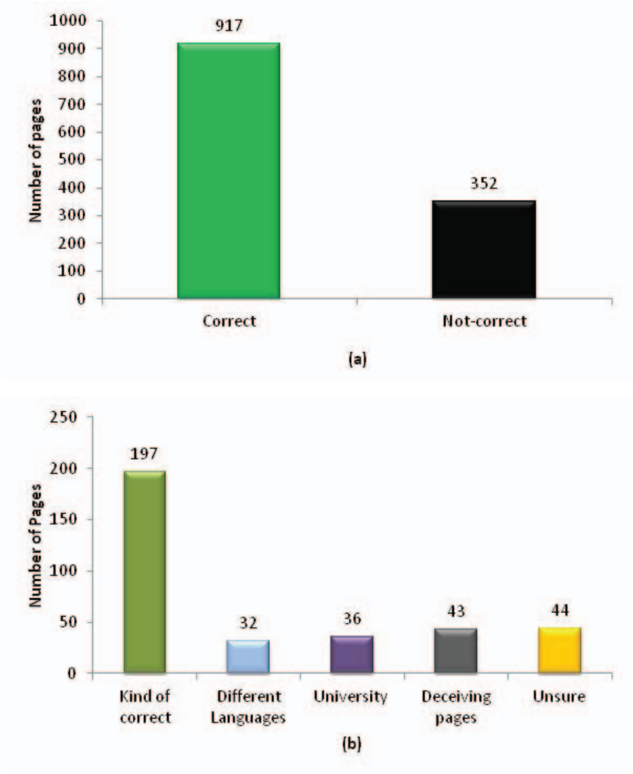

Fig. 4. Data imbalance (a) Binary classification of the “clearly correct” category with the “incorrect” and “unsure” categories combined to "Not-correct" (b) Category classification of "Not-correct" combining the “incorrect” and “unsure” categories.

We choose precision, recall and f-measure as the evaluation measures for our work. Prior studies [20, 23] have already proven that these measures are independent of category distributions provided that precision and recall are measured at the same time. Intuitively, precision measures exactness of the system (i.e., out of all predicted data instances for a specific category label how many are predicted correctly) while recall indicates the completeness of the system (i.e., out of all labeled data for a specific a category label how many are predicted correctly). F value measures the balance between precision and recall in a single value. In our tables with results assessing classifiers, precision and recall refers to their weighted average values. However, the precision and recall values for each category are explicitly given in the cases involving binary classification.

VI. RESULTS

We first provide results from the user survey to assess the need for a system to manage change and analyze how current tools affect the management of personal collections with survey of 106 participants from online and offline communities. Then, we present results from our classification based on the categorization approach presented earlier in section 4 and features described in section 5.

A. *Survey Concerning Collection Practices*

The survey focused on the types and purposes of personals collections, usage, and management techniques. When we asked if they had collections of web pages, 91% reported having a collection of web pages they maintain. Collections could be as simple as browser bookmarks, social bookmarks (Delicious, Pinterest, CiteULike etc.), or just a list of web sites. About 33% reported sharing their collections with others. 39% of respondents who shared their collections shared them with family and 44% with friends and work groups. About 80% responded that their collections were related to their work or were for academic purposes. Interestingly, only about 65% of respondents find their collection as important (Fig. 5.(a)) and about 25% moderately important. When we asked about the types of the sites in their collections, news sites were the most common, followed by weather, sports, social networks, blogs and video web sites. When asked whether they would use a system for maintaining personal collections, 63% responded positively to the idea of having a distributed collection manager (Fig. 5.(b))

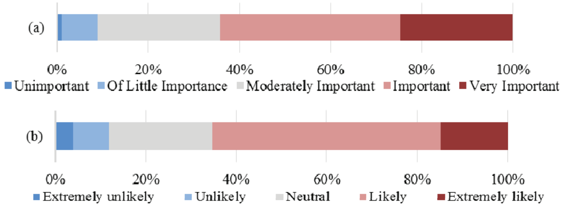

Fig. 5. (a) How important are your collections to you, (b) Likelihood of using a system for maintaining collections

We next asked what tools people used to maintain collections. Browser bookmarks were common (84% reported use) and 44% reported using combinations of bookmarks, cloud platforms, and online services to keep track of collections. Among browser users, 21% used browser bookmarks as well as browser tools like speed-dials, and online tools like OneTab and Pocket to keep track of and maintain web pages.

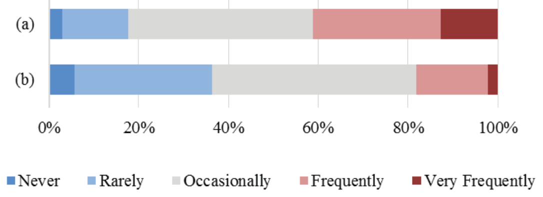

Fig. 6. Relative Frequency in which users (a) want to check for updates in their collections and (b) lose track of their collections

Fig. 6.(a) shows the relative frequency in which users wants to check for updates in their collections. Most (82%) of the respondents wanted at least occasional updates regarding their collections. We also asked how often they lose track of the web sites in their collections. Interestingly, about 5% of respondents felt like they lose track of their collections very frequently, 16% frequently and about 50% occasionally.

When changes happen to their collections, about 72% respondents find these changes as at least moderately dramatic, about 63% find it moderately difficult to determine if a change is important to them and about 65% respondents find it is moderately time consuming to determine if the change is important (see Fig. 7.(a), (b) and (c)).

We also asked what types of changes were likely to be of interest. Visual change, at 69% was of most interest. This is surprisingly in contrast to the previous findings in a similar study [4] of content change (89%) as more important compared to only 5.1% "visual". We suspect that when respondents said "visual", they included imagery, video and much social media content.

Finally, when asked what features they were interested in a tool to maintain personal collections, respondents indicated easy access, platform independence, easily synchronization, search across mobile devices as features they are mostly interested.

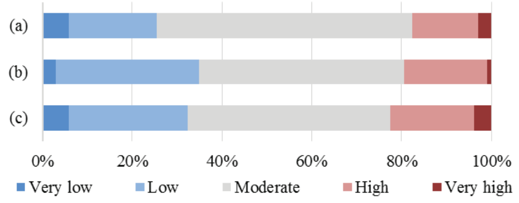

Fig. 7. When change happens in collections (a) How dramatic are these changes? (b) How difficult is it to determine if a change is important? (c) How time consuming is it to determine if a change is important?

B. *Classification Results*

We performed our binary and category classification with 71 algorithms that are implemented in the Weka toolkit [31]. We report the best classification results based on the F-measures from the following classifiers: K*, Decorate, Random Committee, Rotation Forest, Bagging, Boosting (e.g., LogitBoost) and decisions trees (e.g., Random Forest). The algorithmic details of these classifiers are beyond the scope of this paper and interested readers are referred to [11, 31].

Our first experiments explored the impact that the number of topics had on the effectiveness of our classifiers when assigning documents to different categories. As part of these experiments we varied the number of topics K between 5 and 25. After training and testing the category classification data and performing this evaluation, we found that the majority of our classifiers exhibit their best performance with 5 topics (K=5). Therefore, we used 5 topics for the remainder of our experiments involving the training and testing datasets and the analysis of the classifier results. Fig. 8. also shows the F-measure for the best classifiers. The best classifier in most of the feature sets in this category is Rotation Forest followed by Decorate and Random Forest.

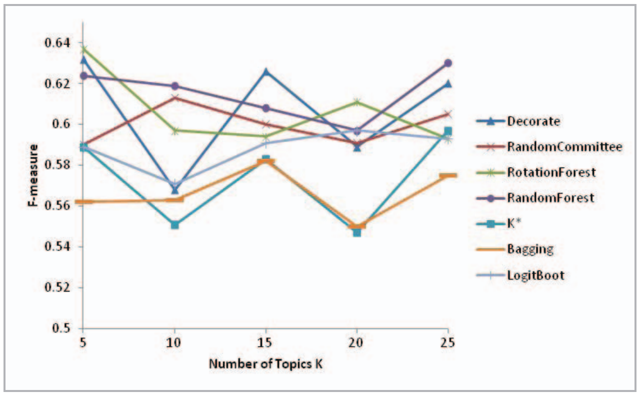

Fig. 8. Evolution of F-measure obtained by applying different number of topics in the classification of categories

The results of our experiments for the "clearly correct", "incorrect" and "unsure" categories as a binary classification problem, and the performance metrics for the 7 most effective classifiers from our evaluation are presented in Table 1. As a baseline for our experiments, we combined the "incorrect" and "unsure" categories into a single group that we called "not correct" and compared it with the "correct" category. Table 1 clearly shows that the majority of our classifiers consistently perform at 63% accuracy; Random Forest was the best performer for the binary classification. Decorate and Random Committee both exhibits a slightly higher F-measure for "correct" category, but Random Forest offers a substantially better F-measure for the "not correct" category.

To further investigate the performance of our classifiers in the "not correct" category, we divided the category classification using the same set of classifiers that we applied in the binary classification problem. As Table 2 shows, Random Forest, Rotation Forest and Decorate all perform in the range of 67% accuracy. Since we are more concerned with categorizing these "incorrect" categories, Random Forest offers the best overall performance and we will rely on it for future evaluations.

Table 3. illustrates the comparison of category classification using only Link-based and Content-based features. This result shows that Content-based features (Random Forest 0.48) are not as efficient on their own when compared to Link-based features (Random Forrest 0.613). On the other hand, this result suggests that when we combine the Content-based features with Link-based features, we get several significant improvements (Random Forest 0.624, Rotation Forest 0.637).

We further analyzed the “incorrect” category by first removing the "pages in a different language" and then removing the "unsure" category from the category classification problem. Although we originally grouped these two categories into the same "not correct" group, it is possible that some of these pages might contain valid or "correct" pages. The results displayed in Table 4 and Table 5 validates this hypothesis by showing that the performance of the Random Forest classifier increases 6% from 0.624 to 0.69 and 16% from 0.624 to 0.793.

TABLE 1. Binary classification (correct, not correct) combining all incorrect categories into single category

	Accuracy	MAE	TP		FP		Precision		Recall		F-Measure	
			C	N	C	N	C	N	C	N	C	N
Decorate	63.62%	0.4784	0.641	0.632	0.368	0.359	0.635	0.637	0.641	0.632	0.638	0.635
RandomCommitt	61.74%	0.4944	0.672	0.562	0.438	0.328	0.606	0.632	0.672	0.562	0.637	0.595
RotationForest	62.75%	0.4552	0.632	0.623	0.377	0.368	0.626	0.629	0.632	0.623	0.629	0.626
RandomForest	64.78%	0.2957	0.664	0.632	0.368	0.336	0.643	0.653	0.664	0.632	0.635	**0.642**
K*	63.19%	0.3689	0.629	0.635	0.365	0.371	0.633	0.631	0.629	0.635	0.631	0.633
Bagging	63.04%	0.4385	0.641	0.62	0.38	0.359	0.628	0.633	0.641	0.62	0.634	0.627
LogitBoost	61.30%	0.4521	0.629	0.597	0.403	0.371	0.61	0.617	0.629	0.597	0.619	0.607

TABLE 2. Classification of only the “incorrect” categories

	Accuracy	MAE	TP	FP	Precision	Recall	F-
Decorate	66.38%	0.1904	0.664	0.262	0.633	0.664	0.632
RandomCommittee	62.03%	0.1825	0.62	0.271	0.576	0.62	0.59
RotationForest	67.25%	0.1851	0.672	0.281	0.641	0.672	**0.637**
RandomForest	67.25%	0.1916	0.672	0.299	0.662	0.672	0.624
K*	61.45%	0.1555	0.614	0.275	0.583	0.614	0.589
Bagging	62.61%	0.2093	0.626	0.357	0.575	0.626	0.562
LogitBoost	63.19%	0.2008	0.632	0.302	0.58	0.632	0.589

TABLE 3. Link-based and Content-based features performance comparison

	F-Measure	
	Link-based	Content-
Decorate	0.600	0.471
RandomCommittee	0.604	0.471
RotationForest	0.577	0.476
RandomForest	0.613	0.48
K*	0.603	0.511
Bagging	0.566	0.457
LogitBoost	0.583	0.475

TABLE 4. Classification of only the “incorrect” categories by removing the “pages in a different language” category

	Accuracy	MAE	TP	FP	Precision	Recall	F-
Decorate	70.70%	0.2116	0.707	0.33	0.684	0.707	0.677
RandomCommittee	70.70%	0.1946	0.707	0.335	0.685	0.707	0.68
RotationForest	71.02%	0.2144	0.71	0.347	0.668	0.71	0.668
RandomForest	72.29%	0.2167	0.723	0.35	0.714	0.723	**0.69**
K*	64.97%	0.1756	0.65	0.334	0.637	0.65	0.632
Bagging	68.79%	0.2338	0.688	0.395	0.664	0.688	0.636
LogitBoost	67.83%	0.2213	0.678	0.383	0.639	0.678	0.637

TABLE 5. Classification of only the “incorrect” categories by removing the “pages in a different language” and “unsure” categories

	Accuracy	MAE	TP	FP	Precision	Recall	F-
Decorate	83.33%	0.1832	0.833	0.27	0.824	0.833	**0.824**
RandomCommittee	78.15%	0.1849	0.781	0.329	0.759	0.781	0.765
RotationForest	82.59%	0.1850	0.826	0.321	0.82	0.826	0.813
RandomForest	80.74%	0.1916	0.807	0.341	0.798	0.807	0.793
K*	78.52%	0.1497	0.785	0.344	0.767	0.785	0.767
Bagging	78.15%	0.2153	0.781	0.42	0.778	0.781	0.758
LogitBoost	80.00%	0.1872	0.8	0.367	0.79	0.8	0.784

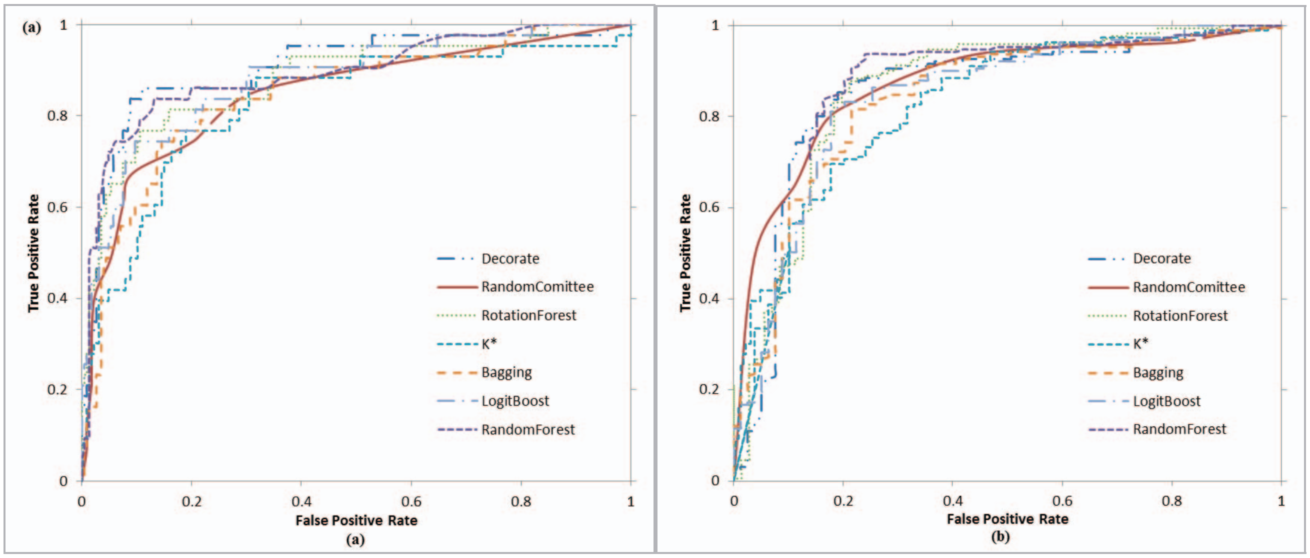

Fig. 9. ROC Curves for the Top 7 Classifiers, (a) "Deceiving Pages" category, (b) "Kind of Correct Pages" category

To further investigate the performance metrics for the 7 most effective classifiers in the “deceiving pages” and “kind of correct” categories, we generated a Receiver Operating Characteristics (ROC) graph. The ROC graphs display the relative tradeoff between benefits (true positive) rates on the Y axis and the costs (false positive) rate on the X axis. Fig. 9. show the ROC graph for “deceiving pages” and “kind of correct” categories. As the graphs show, the Rotation Forest, Random Forest, and Decorate offers the best tradeoff between true positive and false positive performance in both categories.

VII. Discussion and Conclusion

We must also point out that some categories were purposely left out from the classification algorithms as these cases can be handled by previous work. More specifically, detecting “blank pages”, “failed redirects”, “directory listings”, “domain for sale” and “error pages” become trivial cases when they are handled with previous work on identifying Soft 404 error pages [3, 22]. However, we believe that the big contribution of our research is detecting the instances when documents change unexpectedly and fall into more problematic categories such as “kind of correct” and “deceiving pages”.

This last point lead us to investigate and inquiry: Why are the documents in the “deceiving pages” category created? Although the pages in this category are very diverse in content and presentation, they do share two characteristics. First, the number of links that point to other pages within the site is much greater than the number of out-links. On average, pages in the “deceiving” category had 66 links, which is more than twice the average in the “correct” and “kind of correct” categories (20 and 27 links respectively). And second, the domain names that host these pages once belonged to a reputable institution for number of years (i.e., a conference series) before being abandoned. Consequently, these abandoned domain names have a very high value – not monetarily but in their possible uses. We could hypothesize that these pages are created to manipulate pageRank scores by utilizing a large number of links from a page that once had a high PageRank, but have been taken over by a third party. This problem becomes increasingly interesting when we consider that the cost of creating a web page is very little and that some search engines (most notably Google) do not share the overall rankings for their indexed sites, which can lead to some parties to abuse these malicious techniques.

We must also highlight out our research is not focused on detecting spam, but on investigating alternative curation methods to detect unexpected changes in web documents within a collection. However, the degree of change that we are focusing on falls within a specific range: not as subtle as a few terms substitutions in the body of a Web page and not as dramatic, causing servers to report errors explicitly. Our analysis focused on the instances that fall between these two extreme cases, which makes their detection more difficult and require the assistance of a classification system and detection framework such as the one that we have described in this paper.

Distributed collections containing documents from the web are known to change unexpectedly over time. In this paper we have described an approach that studies and categorizes the various degrees of change that digital documents endure within the boundaries of an institutionally managed repository. Documents can change unexpectedly and can introduce uncertainty when viewed as parts of a collection. Our work on identifying these resources helps to reduce this uncertainty by locating documents likely to be problematic and requiring the attention of collection managers.

Our tools provide the ability to create and manage distributed digital collections. Survey responses show that people create personal collections and many find these collections important. The responses also indicate that users have difficulty in keeping track of their collections. This finding validates the need for tools that can help monitor and manage distributed collections.

References

[1] Ashman, H., "Electronic document addressing: dealing with change," *ACM Computing Surveys,* vol. 32, pp. 201-212, 2000.

[2] Baeza-Yates, R., Pereira, I., and Ziviani, N., "Genealogical trees on the web: a search engine user perspective," in *Proceedings of the 17th international conference on World Wide Web*, Beijing, China, 2008.

[3] Bar-Yossef, Z., Broder, A. Z., Kumar, R., and Tomkins, A., "Sic transit gloria telae: towards an understanding of the web's decay," in *Proceedings of the 13th international conference on World Wide Web*, New York, NY, USA, 2004.

[4] Berners-Lee, T., "Cool {URIs} don\'t change," 1998.

[5] Bogen, I., Logasa, P., Shipman, F., and Furuta, R., "Distributed Collections of Web Pages in the Wild," *arXiv preprint arXiv:1101.0613,* 2011.

[6] Bush, V. and Think, A. W. M., "The atlantic monthly," *As we may think,* vol. 176, pp. 101-108, 1945.

[7] Dalal, Z., Dash, S., Dave, P., Francisco-Revilla, L., Furuta, R., Karadkar, U.*, et al.*, "Managing distributed collections: evaluating web page changes, movement, and replacement," in *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*, Tuscon, AZ, USA, 2004, pp. 160-168.

[8] Davis, H. C., "Hypertext link integrity," *ACM Computing Surveys,* vol. 31, p. 28, 1999.

[9] Drummond, C. and Holte, R. C., "Severe class imbalance: Why better algorithms aren’t the answer," in *Machine Learning: ECML 2005*, ed: Springer, 2005, pp. 539-546.

[10] Goh, D. H. L. and Ng, P. K., "Link decay in leading information science journals," *Journal of the American Society for Information Science and Technology,* vol. 58, pp. 15-24, 2007.

[11] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H., "The WEKA data mining software: an update," *ACM SIGKDD explorations newsletter,* vol. 11, pp. 10-18, 2009.

[12] He, H. and Garcia, E. A., "Learning from imbalanced data," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 21, pp. 1263-1284, 2009.

[13] Kahle, B., "Preserving the Internet," *Scientific American,* vol. 276, pp. 82-83, March 1997 1997.

[14] Klein, M., Ware, J., and Nelson, M. L., "Rediscovering missing web pages using link neighborhood lexical signatures," in *Proceedings of the 11th annual international ACM/IEEE Joint Conference on Digital libraries*, Ottawa, Ontario, Canada, 2011.

[15] Kobayashi, M. and Takeda, K., "Information retrieval on the web," *ACM Computing Surveys (CSUR),* vol. 32, pp. 144-173, 2000.

[16] Koehler, W., "A longitudinal study of Web pages continued: a consideration of document persistence," *Information Research,* vol. 9, 2004.

[17] Koehler, W., "Web page change and persistence---a four-year longitudinal study," *Journal of the American Society for Information Science and Technology,* vol. 53, pp. 162-171, 2002.

[18] Kohavi, R., "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *IJCAI*, 1995, pp. 1137-1145.

[19] Li, W.-S., Vu, Q., Agrawal, D., Hara, Y., and Takano, H., "PowerBookmarks: a system for personalizable Web information organization, sharing, and management," *Computer Networks,* vol. 31, pp. 1375-1389, 1999.

[20] Manevitz, L. and Yousef, M., "One-class document classification via neural networks," *Neurocomputing,* vol. 70, pp. 1466-1481, 2007.

[21] McCown, F., Chan, S., Nelson, M. L., and Bollen, J., "The availability and persistence of web references in D-Lib Magazine," *arXiv preprint cs/0511077,* 2005.

[22] Meneses, L., Furuta, R., and Shipman, F. M., "Identifying "Soft 404" Error Pages: Analyzing the Lexical Signatures of Documents in Distributed Collections," in *Proceedings of Theory and Practice of Digital Libraries 2012*, Paphos, Cyprus, 2012.

[23] Monard, M. C. and Batista, G., "Learning with skewed class distributions," *Advances in Logic, Artificial Intelligence and Robotics,* pp. 173-180, 2002.

[24] Nelson, M. and Allen, D. (2002, January 2002) Object Persistence and Availability in Digital Libraries. *D-Lib Magazine*.

[25] Niwa, S., Doi, T., and Honiden, S., "Web page recommender system based on folksonomy mining for ITNG'06 submissions," in *Third International Conference on Information Technology: New Generations (ITNG'06)*, 2006, pp. 388-393.

[26] Page, L., Brin, S., Motwani, R., and Winograd, T., "The PageRank citation ranking: Bringing order to the web," Stanford University1999.

[27] Park, S.-T., Pennock, D. M., Giles, C. L., and Krovetz, R., "Analysis of lexical signatures for improving information persistence on the World Wide Web," *Transactions on Information Systems,* vol. 22, pp. 540-572, 2004.

[28] Phelps, T. A. and Wilensky, R., "Robust Hyperlinks Cost Just Five Words Each," University of California at Berkeley2000.

[29] Spinellis, D., "The decay and failures of web references," *Communications of the ACM,* vol. 46, pp. 71-77, 2003.

[30] Taylor, M. K. and Hudson, D., "" Linkrot" and the usefulness of Web site bibliographies," *Reference & User Services Quarterly,* pp. 273-277, 2000.

[31] Witten, I. H. and Frank, E., *Data Mining: Practical machine learning tools and techniques*: Morgan Kaufmann, 2005.