Characterizing Web Archive Content

Andrew Boyko
Library of Congress
Washington, DC, USA

Abstract

The Library of Congress has been collecting web content since 2000, first through its MINERVA project[6] and, since 2004, as part of a broader Internet capture project. In addition to providing access to some collected content, we have begun to develop tools and techniques to better understand and preserve what we are collecting. When compared with other digital collections, content from the Web has some unique characteristics, such as naming issues and the varying types of relationships between items; nevertheless, when considered at the level of individual items, existing digital preservation approaches are entirely applicable.

In this article, we describe some initial results from examining some selected content from this perspective, including the tools used in our analysis of the Library's Web collections, the approaches taken, and directions for further analysis. We intend that this information will be useful for guiding future web harvest and preservation efforts both within and outside the Library. Our goals include:

- Identifying and measuring the content types in the collection;
- Assessing the variation in file types and validity of "wild" Internet content; and
- Determining typical attributes of various file types, to generate predictors for future web harvests.

We describe web collections as a specific case of a collection of heterogeneous digital content, focusing on the content as received. We will not address issues relating to acquiring the content, such as retrieval problems and link detection during the web crawl, as these issues have been addressed in detail elsewhere[2,8] and are ultimately orthogonal to preservation issues.

Content

The Library of Congress has acquired web collections over the past five years by a combination of activities, including contracting with the Internet Archive for web crawls to the Library's specification, donations, and crawls performed within the Library. The Library's web collections total roughly twenty terabytes, and are growing at a rate of a terabyte or more per month. These numbers are imprecise for two reasons: difficulty in accurately characterizing collection size and variability in the rate of acquisition.

Our collected content is stored by the web crawl tool in the ARC container format,[3] which aggregates content from a single web crawl in multiple arbitrarily ordered units of 100 megabytes. The ARC format, similar in concept to UNIX *tar* tape archive files or ZIP archives, abstracts platform issues from the naming and storage of the content. Content stored in ARC containers may be straightforwardly extracted to other storage, such as individual files or a digital repository; we distinguish ARC from the format of the actual content, and do not consider it inherent to the collection and preservation processes.

A single item in a web collection, corresponding to a resource obtained from an HTTP request for a single URL, may be generally be uniquely identified by the URL and the time of its retrieval. Within the ARC container, any single item's record comprises the complete HTTP response from the server that delivered the item, along with metadata provided by the capture tool. Other item metadata extracted or synthesized after acquisition must be written elsewhere, as the ARC is never re-written or amended. At present, we use a relational database to store this auxiliary metadata, as well as content indexes, but the sole instances of the actual content remain in the ARC container.

Collection Size

While intuitively it would seem that the size of a web collection would be readily apparent on inspection, selecting a single, unambiguous measure for reporting collection size becomes challenging. The most obvious metric would appear to be the amount of storage media space taken up by the content; it is straightforwardly measured and easily understood. However, common practice, based on Internet Archive's precedent, is to store each item, within the ARC container format, compressed with the *gzip* algorithm. The amount of compression varies with the type of content, and thus adds variability to the measurement – given a quantity of data retrieved from the Web, that quantity will be the upper bound for the required storage on disk, but the actual amount used will not be known until the compression is performed. For common web formats, this will average to a 25-50% reduction from the size of the content as received. Note that this compression is distinct from the compression inherent in most media formats (image, video, and audio); the distinction is between the compressed form of the resource as delivered and an additional layer of compression applied to the content on receipt.

The ambiguity added to the measurement by this compression can be demonstrated by considering the hypothetical decision to use an improved compression algorithm, which would generate smaller archived files when applied to re-compress collected content. Doing so would alter the reported size of the collection without adding or removing content. An obvious alternative metric would therefore be to tally the size of the content as acquired in a web crawl. While this avoids the ambiguity described above, and is more accurate in the abstract, it suffers the failing of being more complex to report, and less practically applicable. We have taken the approach of reporting both the uncompressed and compressed quantities. The uncompressed numbers accurately reflect the actual amount of content as originally acquired from the Internet, supporting bandwidth planning, and the compressed numbers reflect the content as found on transfer and archival media, supporting storage planning.

The issue of duplication in repeated web harvests drives another uncertainty into the matter of measurement. We have chosen not to perform any sort of post-processing step on our collections to remove duplication, having considered the risk of losing content through error in the process, and the value of ensuring that any given single web crawl is straightforwardly extractable, intact, from the entire collection. Given the constantly plunging cost of disk storage, we have remained confident in this practice. Nevertheless, in a particular weekly crawl of a topically related set of URLs, a meaningful portion (in one collection harvested monthly, 20-40%) of the content remains entirely constant, at the same URL, from week to week. Storing each copy of these static resources is entirely redundant, in a way that our compression approach does not address. We intend to investigate different approaches to duplicate identification and change detection as part of a general study of the use of digital repository software for managing web archive content.

Collection Rate

Our collecting has been on a more modest scale than massive crawls like those performed by Google or the Internet Archive. We have focused on specific events and thematic collecting rather than snapshotting the breadth of the Internet. Broad, essentially unscoped, web-crawling at Google's scale should be able to product content at a fairly constant rate, given a fixed supply of available bandwidth, computing power, and storage. At our scale, our crawls are bounded by a strict scope, and so have an ending point at which all accessible content has been acquired. Because of constraints in the scope of the crawl, which follow from our need to obtain copyright holder permission, our crawls retrieve content from the sites initially identified by the selection officials, and their immediate supporting peer sites, but go no further into the wider Internet.

With this narrower focus comes a more hand-crafted approach to selecting crawl seeds, and consequently a crawl for a particular collection, repeated every week or month, will vary significantly in size from instance to instance because of added and removed seeds, as well as sites that have grown or disappeared. Even in the case of a single crawl, the ultimate count of items and the storage required cannot be accurately predicted within even an order of magnitude, knowing only the crawl seed URLs and the scope; a preliminary crawl must be performed to gauge the scale of the content.

Content Challenges

Considering the content in a web collection as a heterogeneous mass of digital objects, such a collection displays some of the most challenging characteristics for preservation. Among the challenges inherent in this sort of collection, we find that content in a collection is:

- Multiply sourced
- Multi-format
- Of unknown pedigree
- Targeted at intentionally error-tolerant viewers
- Provided without manifest
- Named inconsistently

Any one of these complaints might arguably give a digital curator pause, and when faced with them in combination, the curator takes some solace in the admittedly low expectations of a user of the low-fidelity medium of the Web. We describe each of these in more detail.

Multiply Sourced

A typical collection at the Library comprises dozens to hundreds of starting "seed" URLs, each of which leads to an unknown number (not uncommonly, as many as hundreds of thousands) of linked documents referenced directly or indirectly by the seed. Each of the documents in turn further requires potentially dozens of supporting files to be acquired, in order to have captured the item with fidelity. Though the seeds for our collections are typically related by an overarching topic or theme, the sites in the collection and their creators share in general only the single fact of having provided to us their permission to be crawled. In any other respect, they vary widely, from governmental sites to individual sites, from non-profit organization to corporate, national or foreign. With the variety of creators comes an uncountable variety of approaches to content production approaches, naming schemes, and creation tools.

Multi-Format

At the broadest level, the content types found across the Web are fairly well understood and predictable. We expect to see HTML documents compose the majority of the items (typically 70-80%), followed by JPEG and GIF format images and PDF documents (each under 10%). Over seven years of collecting, the National Library of Sweden reports[10] that those four types account for 96% of their 260 million files; in a few typical collections, we measured 90-95%. The remaining several percent typically comprises another 30-40 content types, including audio, video, and office document formats, but including errors and anomalies, several hundred reported types may be found in any large collection.

The consistency of these types and proportions is misleading, though, when the variance of actual implementations of each of these formats is considered. For example, documents characterized by their server of origin as "HTML" display a dizzying array of variety, from strictly validating machine-generated XHTML to pure plain text with no sign of HTML tags within. More to the point, the actual type of a given item in the collection is not as clear as the above discussion suggests, for there exists no single entirely trustworthy source of a true type for the item. Web content is sent with a declared MIME content type,[5] based on a decision made by the web server or the web application. Another guess at a valid content type for a URL may be made by considering the suffix of the URL as a file extension, often but by no means always a valid assumption. Still another guess at the content type may be made by applying tools such as the common UNIX *file* utility[11] that attempt to guess the content type from common patterns found in various well known data formats.

It should not be surprising that these varying approaches will not consistently agree. An analysis by the International Internet Preservation Consortium,[4] performed on both Danish web sites and US Congressional sites, indicated that the declared content type, when correlated with the type reported by *file*, was reliable for image formats and PDF. Reporting was less consistent for textual formats, which are more difficult to unambiguously declare. Overall, 90-95% of the items in the examined collections were consistent.

In extending this analysis to correlate all three of the above sources of content type information, we discovered that a typical data set resulted in 72% of the content reporting agreement in all three content types, 11% featuring one disagreeing value, and 17% completely disagreeing.

These disparities speak more to the limitations of our toolset than of any fundamental flaw in the content itself. We must develop a heuristic for determining the most likely type of a piece of content, but in the near term will continue to collect type information from multiple sources and test its validity.

Unknown Pedigree

Of the top four formats in our collections (HTML, JPEG, GIF, and PDF), the possible means to create and manipulate any of them are virtually uncountable, from commercial tools to open-source libraries to ad-hoc or manual processes. This adds not only the sort of variability of adherence to the format specifications discussed above, but also more general concerns regarding the provenance of any given item in the collection. For example, given a digital photograph found on a web site, it is impossible to state with any confidence what processing or manipulation may have affected it, nor what the original source of the photo might have been, nor what copyright issues might affect it. While some sites might make assertions about some or all of this information, the general problem remains inherent in the medium.

Targeted at Intentionally Error-Tolerant Viewers

The actual curation of this content must treat the actual file types as far more fluid and ill defined than the ideals described by the various format specifications. Considering HTML, our most common type of interest, Beckett[1] reported in 1997 that only 6.5% of the UK web validated against an appropriate HTML DTD; at the time, that may have seemed strikingly low, but our own sampling with the "HTML Tidy" tool jibes with a 2003 report[9] that well under 1% of HTML documents are strictly valid. Nevertheless, it would be meaningless to argue that the documents we hold should not be considered HTML because they do not validate against any known specification for HTML.

Clearly, this problem is not new, if what is described above is in fact a problem of curation. Web browsers have been tolerant of malformed HTML from the web's earliest days; modern browsers such as Internet Explorer 6 and Mozilla Firefox use entirely different code to render HTML depending on whether the document signals a likelihood of being standard-compliant, or instead is presumed to be the equivalent of ungrammatical but intelligible text.

In all, the variability in content is much less a problem for the short-term curation of the content than it is for long-term preservation, in the event that format migration becomes appropriate. Our short-term mission is to ensure that we can reproduce the sites we harvest with fidelity to how they were presented at harvest time; malformed content acquired and stored precisely as served requires no immediate action.

Provided without Manifests

The nature of HTTP, the protocol by which web content is typically delivered, precludes a web crawler from requesting or receiving a complete list of available resources at a given site (such as a "directory listing"), or even necessarily what servers might be available within a given Internet domain. The only general means available for identifying content is following links from other documents. This provides an inherently imperfect view of the targeted site. Ignoring the broader issue of fidelity at the site level, we are unable at the level of individual content to known with confidence that we acquired all of a given set of content, or that we received the best available versions. Without available checksums or other objective descriptions of the individual items, we place our trust in the crawler tool and its ability to locate references to new content matching our scope of interest, and to capture the delivered stream of data without modification.

Named Inconsistently

Not only must we worry about the internal attributes of an item in our collection, but also how to preserve the ability to store and retrieve it by the name by which it was known on the Web. The URL that identifies a given piece of content is guaranteed to convey only the information about its original source, and not necessarily any of the other metadata commonly found in file names for digital content produced by other means. For example, the convention of naming files

with a suffix that suggests the item's content type is far from universal for Web content, for which the URL is expected to describe the logical content rather than its physical form. A more general problem is that file naming and the URL namespace have incompatible characteristic; typical filesystems (e.g. Windows NTFS, Linux ext2) will prohibit varying subsets of characters commonly found in URLs.

Content Challenges: Summary

The list of problems discussed above is not intended to be complete, but to describe aspects of the general class of problems inherent with web content. Strategies for dealing with these problems help us not only with web content, but also with digital content of other types we may acquire that shares some or all of the same attributes. None of the issues above is coupled to web-specific characteristics, such as linking relationships or the challenges of providing web browser-based access to the content.

Tools and Techniques

In advance of building a formalized repository for web archive content, we have built a simple crawl database to manage information about our web harvests, including item-level metadata retrieved in a crawl. In addition to providing a convenient means for access to the metadata delivered with the content, the database is useful for storing the results of other metadata extractions we may perform during post-processing.

Our basic need was for a structure within which we could simply visit each item in a set of content and, based on its existing metadata, decide whether to further subject the item's content to post-processing and analysis. To construct this structure required tools that address the following areas:

Archive Management

The ARC format provides independence from any particular filesystem, but requires tools and external indexing to locate and extract the content stored in the ARC files.

File Type Identification

Items in the collection incorporate the MIME content type announced by the web server. In addition to deriving a second view of the type information from the URL's file extension, we supplement the stated type information with our own type audit, using tools such as the Unix *file* command. Comparing these results, we can determine the most likely true file type for each object.

File Type Validation

Given the likely file type of an object, we would like to determine whether, and how well, the object complies with the specification of its format. The Library has developed a framework for applying particular validators to many types, such as the JHOVE toolset,[7] However, many content types lack standard validators.

Our toolset combines open-source and in-house tools, all of which run on the Linux operating system. These tools are implemented in a variety of programming languages, which provides some challenges because of the varied environments they expect. The Python programming language, and in particular its Java implementation, Jython, serves effectively as "glue" to integrate these disparate tools and system services.

Metrics

After processing each item in the archive, we are able to generate a variety of statistics about the data in aggregate. These include:

- Content types identified by tools
- Comparison with content types declared by the web server
- Comparison with URL file extensions
- Format validations
- Format version identifications

We can measure these characteristics for sub-sets of the collections, such as:

- Content from a time-frame (e.g., *2001, 2002 1st quarter*)
- Content from a domain or sub-domain (e.g., *.edu*, *loc.gov*)
- Content from a site type (e.g., blogs, public forums, corporate sites)

While we are developing our ability to generate the metrics above, we have begun to analyze content type information on sampled subsets of our various collections, and report some results here.

As discussed in relation to the error-tolerance of viewers, the bulk of the HTML documents harvested, approaching 99%, cannot be strictly validated against the letter of the appropriate HTML specification. More than saying anything about the content, it suggests that the measurement is without clear value. A finer-grained assessment of validity, which highlights troublesome areas in the document rather than providing a simple binary indication of validity, will help to guide some actual action.

Our examination of PDF documents in the collection indicated, unsurprisingly, that PDFs are vastly more likely than HTML documents to be valid by a strict definition of the format. At the same time, the nature of the format means that an invalid PDF is less likely than an invalid HTML document to provide satisfactory results when rendered, based on the expectations of the format. PDF intends to provide consistency of appearance across output devices, while HTML is merely intended to approximate the same appearance, and renderers are expected to degrade gracefully and tolerate malformation.

Using the PDF validator from the JHOVE toolkit, we determined 87% of a sample of PDFs to be "well-formed and valid"; another 11% were "well-formed but not valid" (signifying a semantic but not syntactic error), and 2.5% were not well-formed (i.e. genuinely corrupt). With a growing number of alternative routes to creating PDFs programmatically, particularly by ad-hoc scripts, we can

expect the percentage of technically invalid PDFs to increase in our collections.

Examining JPEG images with JHOVE's validator uncovered, by comparison, more consistency. Only 1% of the JPEGs in a tested collection were invalid. Of these, closer examination revealed several to be other image formats (such as PNG) that had been misidentified on their server of origin. Web browsers will typically ignore the stated type of images when choosing the appropriate renderer, so the problem affects only the preservationist, highlighting the need for such examinations.

Directions

By analyzing these measurements, we can better characterize our collection content and scope, improving the quality of future crawls. We can also make better estimates for continuing storage needs.

These statistics guide our preservation strategies as well. By correlating the identified content types with format registries and definitive lists of viewer tools, we can identify opportunities for migration or emulation. Trends over time also indicate the impending obsolescence of a format and highlight a preservation need. It is not yet clear how, or whether, to apply digital preservation approaches such as format migration to web archive content; we must investigate whether transformed content may be integrated into a collection without changing the nature of the content in context.

Our highest priority is to extend the depth of our exploratory analyses to the scope of the entire collection, and to systematize this analysis as part of general curatorial examination. By doing so, we will be able to more definitively identify areas of preservation risk in the collection; at present, because of our thematic collecting and the narrowness of our focus, it is difficult to extrapolate with any confidence from the sampling measurements made on individual collections to the general state of web collecting.

While a great deal of work is being done to describe best practices for archival content production, such as the use of the PDF/A archival standard or strongly validating XHTML, the benefit of this work to the curation of this web archive content is indirect. Our first responsibility is to maintain the content with fidelity to how it was originally presented, though future access requirements may well warrant format migration to archival standards.

As we proceed to make the extraction of content type-related metadata more systematic, the need for a more formal way of coupling the metadata to the content will become more pronounced. Open questions remain about an appropriate METS profile for web archive content, with its distinct characteristics of repetition in time, and the number and variety of strong relationships between items.

References

1. D. Beckett, "30% Accessible - A Survey of The UK Wide Web", *http://www.ilrt.bristol.ac.uk/people/cmdjb/papers/www6/paper.html,* April 1997
2. A. Boyko, Test Bed Taxonomy for Crawler, *http://www.netpreserve.org/publications/reports.php?id=001*, July 2004
3. M. Burner and B. Kahle, "Arc File Format", *http://www.archive.org/web/researcher/ArcFileFormat.php,* 1996
4. S. S. Christensen and A. Boyko, "MIME Type Analysis", IIPC draft, unpublished, 2004
5. N. Freed and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", IETF RFC 2046, November 1996
6. A. Grotke, Creating Access Points to Thematic Web Collections, *http://www.imaging.org/store/epub.cfm?abstrid=30280,* 2004
7. JHOVE – JSTOR Format-Specific Digital Object Validation, *http://hul.harvard.edu/jhove/jhove.html*, 2005
8. J. Marill et al, Web Harvesting Survey, *http://www.netpreserve.org/publications/reports.php?id=001,* 2004
9. C. Marincu and B. McMullin, "Improving Access to Online Information via Valid HTML Mark Up", *http://eaccess.rince.ie/white-papers/2003/warp-2003-01/* September 2003
10. K. Persson, "Kulturarw^3 statistics", *http://www.kb.se/kw3/ENG/Statistics. htm*
11. C. Zoulas, *ftp://ftp.astron.com/pub/file/*, 2004

Biography

Andrew Boyko is the technical lead for the Web Capture team at the Library of Congress, working on developing tools and processes for the long-term acquisition, access, and preservation of digital content. His background is in software development and system deployment, in the areas of web application, content management, and network protocols. He has a BS in computer engineering from Virginia Tech.