Hypercane: Toolkit for Summarizing Large Collections of Archived Webpages

SHAWN M. JONES
Los Alamos National Laboratory, Los Alamos, NM
Old Dominion University, Norfolk, VA
and
MICHELE C. WEIGLE AND MICHAEL L. NELSON
Old Dominion University, Norfolk, VA

In the Dark and Stormy Archives (DSA) project, we focus on *storytelling* techniques to summarize collections of archived web pages. Since collections can have hundreds or even thousands of *seeds* (initial URLs) and each seed can be recrawled many times, with each version separately maintained, techniques that include information about all members of the collection can be overwhelming. The premise of storytelling is to focus on sampling exemplar pages from the collection, and present them in a social media interface familiar to users. We present *Hypercane*, the tool in the DSA suite responsible for selecting exemplar pages. Hypercane offers eight action statements that can be combined in various ways to customize the sample that is produced. Because of its modular design, Hypercane can also be used to analyze large web archive collections outside of the DSA suite.

1. INTRODUCTION

The Web is an integral part of how we interact with our world, from learning about current events to accessing public government documents to communicating with the larger community. We know that the Web is not static – the content on webpages can change, the URLs of webpages can change, webpages can disappear altogether. Because of the dynamic nature of the Web, web archiving is essential for preserving our history for future study. The Internet Archive's Wayback Machine [Negulescu 2010] is the oldest and largest public web archive, allowing users to revisit and replay many webpages from the past 25 years.

Libraries, museums, universities, and government entities have been active participants in web archiving for many years. Many of these institutions are tasked with archiving their own online presence, while others collect and preserve webpages related to certain topics or events. Because preserving webpages can be a time and personnel intensive operation, many of these institutions subscribe to the Internet Archive's Archive-It [Archive-It 2014] service, which provides collection-based web archiving and replay services. Users can create web archive collections based around a specific topic and request Archive-It to archive

a set of *seed URIs* at specified intervals.

Archive-It has been in operation since 2006 [Archive-It 2014] and holds more than 15,000 collections from over 2,000 organizations. One of the largest collections holds almost 340,000 seeds with over 450,000 archived copies of webpages, or *mementos*. Understanding the content in such collections can be overwhelming. For several years, the Web Science and Digital Libraries (WS-DL) Research Group at Old Dominion University (ODU) has been investigating ways to help users make sense of large web archive collections. Our recent work has focused on automated methods to summarize web archive collections and present the summary in a user-friendly manner using social media storytelling techniques. An example of this is shown in Figure 1, showing the result of summarizing a large collection about COVID-19 (Archive-It collection 13529) into a set of articles visualized with social cards, a format that is familiar to users of social media.

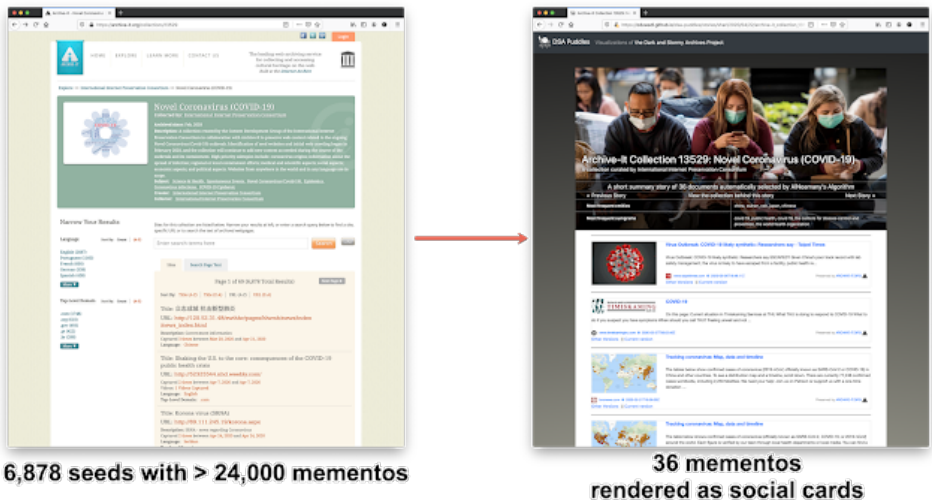

Fig. 1. We can summarize a web archive collection by selecting a small number of exemplars and then visualize them with social media storytelling techniques.

In this article, we describe Hypercane, a toolkit for summarizing large web archive collections. Our initial motivation for Hypercane was to facilitate storytelling with web archives, the larger goal of which is to make web archives more accessible to the general public. We describe the usage of Hypercane and present use cases, and hope that this will inspire other uses for Hypercane. More information about Hypercane can be found in a series of blog posts [Jones 2020a; 2020b; 2020c] and at `https://oduwsdl.github.io/hypercane/`.

2. BACKGROUND

2.1 Memento

Our work with web archives is facilitated by the Memento protocol [Van de Sompel et al. 2013], which provides a framework for HTTP content negotiation in the time dimension. Memento allows HTTP clients to provide the `Accept-Datetime` HTTP request header to indicate the preferred datetime for the requested web resource. If the resource is available, a Memento-compliant web server (usually connected to a web archive) would include the `Memento-Datetime` HTTP response header to indicate the capture datetime of the returned resource and include other Memento-related metadata in the `Link` HTTP header. Memento also provides terminology that makes it easier to talk about multiple versions of web resources:

—*URI-R* - the URI of the original resource, a web resource that exists or used to exist

—*URI-M* - the URI of an archived version of the URI-R at a particular datetime, also known as a *memento*

—*URI-T* - the URI of a list of mementos for a particular URI-R, also known as a *TimeMap*

2.2 Storytelling with Web Archives

Hypercane is an essential part of the Dark and Stormy Archives (DSA) Toolkit [Jones et al. 2020; AlNoamany et al. 2017], which summarizes a collection of archived pages (cf. Figure 1) by sampling only a small number (often about 28) of exemplar archived pages to produce a "story" in a familiar social media interface. Figure 2 provides an overview of the DSA Toolkit architecture.

Hypercane is the first step in the process and takes as input a collection of mementos and produces as output some smaller set of mementos, based on the operations and parameters given to the tool. Hypercane can use structural features of web archive collections, along with similarity metrics and Natural Language Processing, to select the best mementos from collections for our stories. The operations provided by Hypercane will be described further in this article and allow Hypercane to be used for purposes other than just the DSA Toolkit.

Hypercane leverages the Off-Topic Memento Toolkit (OTMT) [Jones et al. 2018; AlNoamany et al. 2016] as a library to identify archived pages that deviate from the topic of the original collection and uses AIU [Jones 2018] to extract information from web archive collections at Archive-It and collections in Australia's web archive through Trove [National Library of Australia b] and PANDORA [National Library of Australia a].

Raintale [Jones 2019] can take input as a set of mementos from Hypercane to produce a story. Raintale uses MementoEmbed [Jones et al. 2020] to produce a surrogate of each memento and then can then publish the story to an individual file, in a format like HTML, or a service, like Twitter. The goal is to allow a user to summarize and visualize the whole collection or a specific aspect of it.

This separation of concerns in the DSA Toolkit allows each component to focus on one problem area. Hypercane intelligently samples from collections. MementoEmbed extracts information about individual mementos. Raintale renders the information for many me-

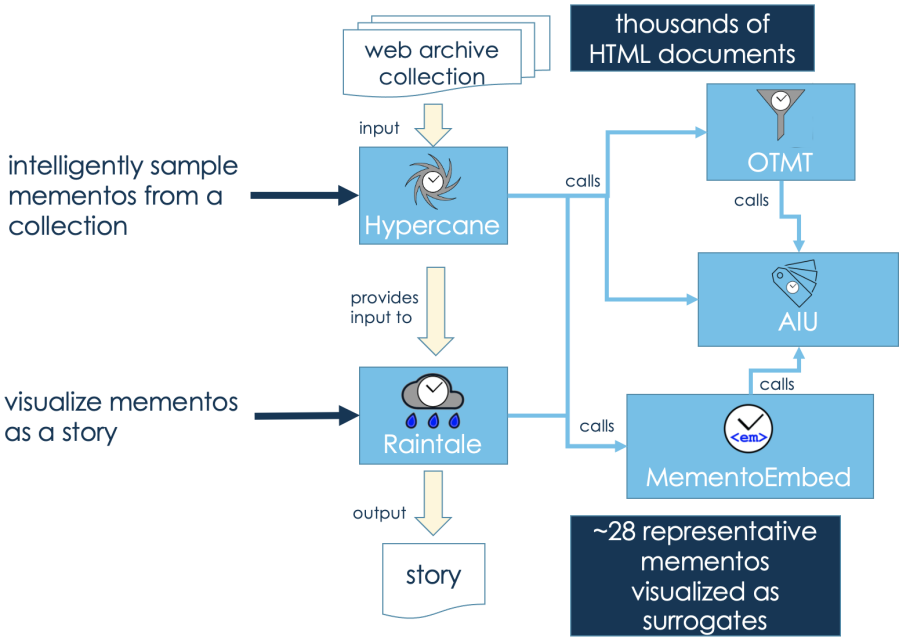

Fig. 2. The DSA Toolkit automated storytelling process relies upon Hypercane for its input and Raintale for its output, with additional data provided by OTMT, AIU, and MementoEmbed.

mentos. Raintale knows nothing of the Memento Protocol and relies upon MementoEmbed to give it the information it needs. Additionally, this separation of concerns allows us to use each component individually to solve other problems. For instance, Raintale can render stories based on human selection rather than from Hypercane input, and MementoEmbed can be used to generate a single social card of a memento for a blog post. We will discuss other ways that Hypercane can be used in Section 6.

3. RUNNING HYPERCANE

Hypercane, `hc`, is a command-line application that takes a collection of mementos as input and provides various *actions* that can be applied to the collection to produce a new set of mementos as output.

The basic template for a Hypercane command is

```
# hc action action_parameter -i input_type -a input_parameter -o filename
```

Hypercane supports several types of input across all of its commands. An input type is supplied with the `-i` argument. For each input type, the `-a` argument specifies the collection identifier or the file containing the input. Hypercane supports Archive-It collections with `-i archiveit -a collection_id`, Trove collections with `-i trove -a collection_id`, PANDORA collections with `-i pandora_collection -a collection_id`, and PANDORA subjects with `-i pandora_subject -a subject_id`. Hypercane also

allows users to manually specify a list of input resources. These can be a list of URI-Ms (`mementos`), URI-Rs (`original-resources`), or URI-Ts (`timemaps`).

Hypercane makes heavy use of the Memento Protocol to discover TimeMaps from original resources and mementos from TimeMaps. If original resources are provided as input but no mementos are available, meaning that the original resources have not yet been archived, Hypercane will use ArchiveNow [Aturban et al. 2018] to request that a set of public web archives capture those resources, producing new URI-Ms. In this way, a user could supply a list of URI-Rs (that may or may not have been previously archived) and get a list of URI-Ms (that have all now been archived).

The `-o` argument is used to specify the output filename. Most Hypercane commands will output a tab-separated file containing a list of mementos identified by URI-M.

4. HYPERCANE ACTIONS

Hypercane commands consist of one action with arguments that affect how that action is executed. Below we provide an overview of each of the provided actions. More details about how to specify each of these actions are available in the Hypercane documentation [Jones]. In addition, all Hypercane commands support the `--help` argument.

4.1 Sample

Hypercane's initial goal was to automatically produce an intelligent sample from a web archive collection, based on the DSA toolkit. To support this, the `sample` action takes a collection and performs a specified sampling operation on the input. The first sampling algorithm supported was `dsa1`, which implements AlNoamany's DSA algorithm [AlNoamany et al. 2017]. This sampling algorithm first excludes off-topic, near-duplicate, and non-English mementos. Then the remaining mementos are distributed equally by memento-datetime over about 28 slices[1], and the webpages in each slice are clustered using DBSCAN [Ester et al. 1996]. Then the "best" representative memento from each cluster is chosen, and the resulting mementos are put in chronological order. As will be explained, each of the steps in AlNoamany's DSA sampling algorithm have been broken into separate actions available in Hypercane. Thus, the `sample dsa1` action can be carried out with chain of other Hypercane commands (Section 5).

Hypercane also provides simpler sampling algorithms, including randomly selecting k mementos after filtering out only off-topic and near-duplicate mementos (`filtered-random`), randomly sampling k mementos from the input (`true-random`), and selecting every j^{th} memento from the input (`systematic`).

The `sample` action also provides sampling operations specifically tailored for output from its `cluster` action. The algorithms include choosing j random mementos from each cluster (`stratified-random`), choosing every j^{th} memento from each cluster (`stratified-systematic`), and selecting all mementos from j randomly-chosen clusters (`random-cluster`).

[1] In AlNoamany et al. [AlNoamany et al. 2017], the goal was to produce a final set of about 28 mementos.

4.2 Report

Hypercane can produce reports for use in storytelling and rudimentary collection analysis.

For collection content analysis, Hypercane provides the `terms` report that provides all terms discovered in the input, including their frequency, document frequency, probability, and corpus-wide TF-IDF as a tab-delimited file. This report can also use the sumgram tool [Nwala 2019] to compute the frequency of conjoined ngrams in the collection. The `entities` report provides a list of all entities discovered in the input using the spaCY library [Honnibal et al. 2020], including frequency, probability, and corpus-wide TF-IDF in a tab-delimited file.

Other reports are output in the form of a JSON file and have been used to support Raintale (`image-data`), analysis of HTML `META` tags in Jones et al. [Jones et al. 2021] (`html-metadata`), collection seed and growth statistics in Jones et al. [Jones et al. 2018] (`seed-statistics` and `growth`, respectively), and metadata analysis in Jones et al. [Jones et al. 2019] (`metadata-statistics`).

4.3 Identify

The `identify` action produces one type of Memento Protocol object from another. The standard Hypercane `-i` option is used to specify the input type, and the parameter to `identify` specifies the output type: `mementos`, `timemaps`, or `original-resources`.

With `identify`, a user can submit a list of URI-Ts and get the full list of URI-Ms or URI-Rs discovered in those TimeMaps. They can also submit a list of URI-Ms and produce the corresponding URI-Ts or URI-Rs. One can also supply a collection identifier (e.g., from Trove or Archive-It) and Hypercane will produce a list of URI-Ms, URI-Rs, or URI-Ts.

4.4 Filter

Filtering is key to building a sample from a larger collection. With `filter` an archivist can include or exclude mementos based on certain criteria. Each command outputs a list of URI-Ms identifying the mementos that meet that criteria. Hypercane's `filter` action accepts either `include-only` or `exclude` as subactions. These modifiers then apply to the criteria specified. Supported filter criteria include off-topic mementos (`on-topic`), duplicate pages (`non-duplicates`), content language (`languages`), text pattern in the content (`containing-pattern`), text pattern in the URI-R (`containing-url-pattern`), and cluster scoring (`highest-score-per-cluster`, `largest-clusters`, `score`).

4.5 Cluster

Clustering divides a collection into meaningful sub-collections for further processing, such as scoring and filtering to produce exemplars. Hypercane supports AlNoamany's time slice clustering [AlNoamany et al. 2017] (`time-slice`), AlNoamany's Simhash-based clustering, DBSCAN (`dbscan`), K-Means (`kmeans`), and topic modeling with Latent Dirichlet Allocation (LDA) (`lda`). Hypercane can also cluster a collection by the domain name of each URI-R (`domainname`) or by the URI-R itself (`original-resource`).

4.6 Score

Scoring is also an essential staple of summarization to help rank multiple candidates. The DSA toolkit uses scoring to choose mementos from the result of clustering. Hypercane supports AlNoamany's scoring function [AlNoamany et al. 2017] (`dsa1-scoring`), BM25 [Xapian Project] scoring (`bm25`), scoring by path depth as defined by McCown et al. [McCown et al. 2005] (`path-depth`), scoring based on web page categories established by Padia et al. [Padia et al. 2012] (`url-category-score`), scoring based on number of images (`image-count`), and scoring by how well it would be represented as a social card on Facebook and Twitter as applied by Jones et al. [Jones et al. 2021] (`simple-card-score`).

For web page category scoring, we have updated the list of web sites fitting into Padia et al.'s categories. Hypercane currently scores social media domains based on a list from Adobe [Adobe Inc. 2021], image sharing sites based on a list from Wikipedia [Wikipedia contributors 2020], video sharing sites based on another list from Wikipedia [Wikipedia contributors 2021], and news websites based on lists from Pew Research [Pew Research Center 2019] and W3Newspapers [w3newspapers.com 2021]. We intend to make these lists configurable in the future.

4.7 Order

Sorting content is critical for conveying meaning. Hypercane can order mementos based on the resulting `score` action. For storytelling, we often want articles to flow in chronological order, so Hypercane also supports time-based scoring (`pubdate-else-memento-datetime`, `memento-datetime`). Hypercane leverages the newspaper3k library [Ou-Yang 2013] to discover the publication date of a given memento. If newspaper3k cannot find a publication date, Hypercane falls back to the memento-datetime of the resource.

4.8 Synthesize

Hypercane's `synthesize` action allows users to generate output in different formats suitable for other tools. Hypercane can generate a set of files in a directory, where these files can be in WARC format [ISO 28500:2017 2017] (`warcs`), Raintale story JSON (`raintale-story`), webpage source (`files`), or webpage source with boilerplate removed (`bpfree-files`).

5. CONSTRUCTING ALGORITHMS WITH ACTIONS

Because all Hypercane actions accept as input the same file format as they output, most Hypercane commands can feed data into each other. For instance, as shown in Figure 3, we could execute a `sample` action using one algorithm and feed the result into another `sample` action using a different algorithm. The resulting mementos can then be fed into an `order` action before finally feeding the result into a `synthesize` action to generate input for a Raintale story containing only those mementos. Many such combinations of actions are possible. In these examples, we will be using Archive-It collection 13529, "Novel Coronavirus (COVID-19)" (see Figure 1).

In Listing 1 we show the series of Hypercane commands that implement each part of the

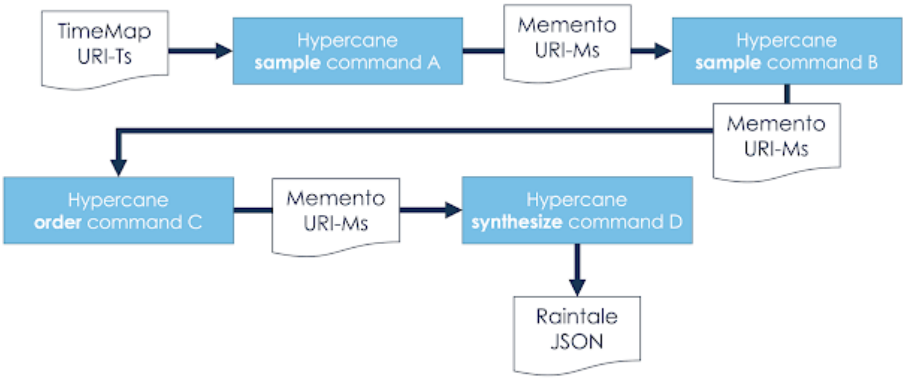

Fig. 3. A hypothetical Hypercane workflow shows a user providing a list of TimeMap URI-Ts as input to a sample command. The output list of Memento URI-Ms from that command can be used as input to subsequent commands who then feed others.

Listing 1. Hypercane commands for implementing AlNoamany's DSA1 sampling algorithm on Archive-It collection 13529

```
hc identify timemaps -i archiveit -a 13529 -o timemaps.tsv
hc filter include-only on-topic -i timemaps -a timemaps.tsv -o ontopic.tsv
hc filter exclude near-duplicates -i mementos -a ontopic.tsv \
    -o non-duplicates.tsv
hc filter include-only languages --lang en -i mementos \
    -a non-duplicates.tsv -o english-only.tsv
hc cluster time-slice -i mementos -a english-only.tsv -o sliced.tsv
hc cluster dbscan -i mementos -a sliced.tsv -o sliced-and-clustered.tsv
hc score dsa1-scoring -i mementos -a sliced-and-clustered.tsv -o scored.tsv
hc filter include-only highest-score-per-cluster -i mementos \
    -a scored.tsv -o highest-scored.tsv
hc order pubdate-else-memento-datetime -i mementos -a highest-scored.tsv \
    -o ordered.tsv
```

AlNoamany DSA1 sampling algorithm on Archive-It collection 13529. The output of each command feeds into the next. For example, the `english-only.tsv` file generated by `hc filter include-only languages` serves as input to `hc cluster time-slice`. To further reuse our existing work, when a user executes the `hc sample dsa1` command, they are asking Hypercane to execute this series of advanced actions as a shell script.

The goal of Hypercane was not merely to reimplement AlNoamany's algorithm, but also to give archivists a tool to customize summarizations. An archivist could easily only include Japanese documents, or skip the `hc cluster dbscan` step altogether. AlNoamany's algorithm gives us a guide of what is initially possible in this problem space, but it is by no means the only sampling solution.

We implemented a filtered random algorithm that borrows parts of AlNoamany's algorithm and then randomly chooses from the result by using the `hc sample true-random` command. The individual commands are shown in Listing 2. A user can also request this

Listing 2. Hypercane commands for implementing the filtered-random sampling algorithm

```
hc identify timemaps -i archiveit -a 13529 -o timemaps.tsv
hc filter include-only on-topic -i timemaps -a timemaps.tsv -o ontopic.tsv
hc filter exclude near-duplicates -i mementos -a ontopic.tsv \
    -o non-duplicates.tsv
hc sample true-random -i mementos -a non-duplicates.tsv \
    -o filtered-random-sample.tsv
```

Listing 3. Hypercane commands to built a SPFT story from Archive-It collection 13529

```
hc identify timemaps -i archiveit -a 13529 -o timemaps.tsv
hc filter include-only on-topic -i timemaps -a timemaps.tsv -o ontopic.tsv
hc filter exclude near-duplicates -i mementos -a ontopic.tsv \
     -o non-duplicates.tsv
hc filter include-only languages --lang en -i mementos \
     -a non-duplicates.tsv -o english-only.tsv
hc filter include-only near-datetime -i mementos -a english-only.tsv \
    --start-datetime 2020-03-19T00:00:00 \
    --end-datetime 2020-03-19T23:59:00 -o time-filtered.tsv
hc cluster time-slice -i mementos -a time-filtered.tsv -o sliced.tsv
hc cluster dbscan -i mementos -a sliced.tsv -o sliced-and-clustered.tsv
hc score dsa1-scoring -i mementos -a sliced-and-clustered.tsv -o scored.tsv
hc filter include-only highest-score-per-cluster -i mementos -a scored.tsv \
    -o highest-scored.tsv
hc order pubdate-else-memento-datetime -i mementos -a highest-scored.tsv \
    -o ordered.tsv
```

combination by executing `hc sample filtered-random`, which calls a shell script that executes these commands in the background.

AlNoamany et al. [AlNoamany et al. 2017] described four types of stories possible with web archive collections:

—Fixed Page Fixed Time (FPFT) - the sample consists of mementos representing different versions of the same URI-R rendered by different user agents (e.g., mobile vs. desktop)

—Fixed Page Sliding Time (FPST) - the sample consists of mementos from the same URI-R over time

—Sliding Page Fixed Time (SPFT) - the sample consists of mementos captured during a specific time range within the collection

—Sliding Page Sliding Time (SPST) - the sample consists of mementos drawn from any time and any URI-R in the collection

So far, we have focused on SPST stories. We can leverage other filter commands to tell other types of stories. In Listing 3, we use `hc filter include-only near-datetime` to leverage parts of AlNoamany's Algorithm to tell an SPFT story for collection 13529 that only includes mementos with Memento-Datetimes on March 19, 2020.

We can also use `hc filter include-only containing-url-pattern` to tell a story about the Pakistani government's changing response to COVID-19 (Listing 4).

Listing 4. Hypercane commands to produce a FPST story about Pakistan and COVID-19

```
hc identify timemaps -i archiveit -a 13529 -o timemaps.tsv
hc filter include-only on-topic -i timemaps -a timemaps.tsv -o ontopic.tsv
hc filter exclude near-duplicates -i mementos -a ontopic.tsv \
    -o non-duplicates.tsv
hc filter include-only languages --lang en -i mementos -a non-duplicates.tsv \
    -o english-only.tsv
hc filter include-only containing-url-pattern -i mementos -a english-only.tsv \
    --urir-pattern "http://covid.gov.pk/" -o urir-filtered.tsv
hc cluster time-slice -i mementos -a urir-filtered.tsv -o sliced.tsv
hc cluster dbscan -i mementos -a sliced.tsv -o sliced-and-clustered.tsv
hc score dsa1-scoring -i mementos -a sliced-and-clustered.tsv -o scored.tsv
hc filter include-only highest-score-per-cluster -i mementos -a scored.tsv \
    -o highest-scored.tsv
hc order pubdate-else-memento-datetime -i mementos -a highest-scored.tsv \
    -o ordered.tsv
```

6. HYPERCANE USE CASES

We can explore collections with Hypercane outside of the DSA Toolkit. In the following two examples we show using Hypercane to generate output that can be used in Gensim topic modelling [Řehůřek and Sojka 2010] and using Hypercane to generate output based on input from the StoryGraph visualization tool [Nwala et al. 2020].

6.1 Synthesizing Outputs for Gensim

Here we explore how to use Hypercane's `synthesize` outputs with Gensim. Most NLP tutorial examples assume that the user has access to plain boilerplate-free text files. Hypercane allows us to generate a directory containing these types of text files from a collection of archived web pages using the `synthesize bpfree-files` option. This produces a directory containing one file per memento with its boilerplate removed as well as a file containing a mapping of these files to their original URI-Ms.

From this set of files, we used LDA in Gensim to break the corpus into 10 topics. Then we invoke matplotlib and the WordCloud library to produce images for each. We show a sample of four of the images in Figure 4. This visualization gives us some idea of the different types of information that are present in Archive-It collection 13529. From these images, we can see that some topics focus on more on China while others appear to discuss US governor responses.

6.2 Visualizing the Day's Top News Stories

As part of the SHARI (StoryGraph Hypercane ArchiveNow Raintale Integration) process [Jones et al. 2020], Hypercane relies upon StoryGraph to provide a list of URI-Rs linking to news articles about the biggest news story for that date. Hypercane then converts this list into a set of URI-Ms with its `identify` action. With these URI-Ms, Hypercane applies the `report` action to generate reports on entities, terms, and images. It applies the `order` action to order the list of URI-Ms by publication date. It then consolidates all of these

Fig. 4. Four LDA topics from Archive-It collection 13529 as of April 2020, modeled as word clouds.

reports and the ordered list of URI-Ms through the `synthesize` action that generates JSON for Raintale. SHARI publishes a new story every day containing the articles that make up the biggest news story of the previous day and shares the story on Twitter from the `@StormyArchives` account. Figure 5 demonstrates the output of the SHARI process for May 16, 2021.

7. CONCLUSION

The Hypercane toolkit provides a set of composable actions for working with large collections of archived web pages and is an integral part of the Dark and Stormy Archives framework for storytelling with web archives. Hypercane leverages the Memento Protocol and allows the user to input an Archive-It collection number, Trove collection number, Pandora collection number, a list of mementos, a list of TimeMaps, or a list of original resources as the starting point for processing. Hypercane's principal focus is providing a small sample of documents from a larger collection. Hypercane provides preset algorithms for deriving the sample and also provides advanced actions, including filter, cluster, score, and order, that can be used to build user-defined algorithms.

In this article, we have described Hypercane's actions and demonstrated how they can be composed to construct new sampling algorithms. We have also shown how Hypercane can be used to synthesize output for use in other tools, such as Gensim topic modelling and the SHARI visualization of the day's top news stories.

New filters are on the horizon. We intend to expand the features available to the `k-means` and `dbscan` commands provided by the `cluster` action. We are currently researching how to score mementos based on their images and sentences, as well as ordering documents based on other criteria, such as CarbonDate [SalahEldeen and Nelson 2013]. We are working with the developer of StoryGraph to provide clustering via the StoryGraph algorithm. We will group Hypercane commands together into new intelligent sampling algorithms accessible through the `sample` action, labeling these *recipes* as DSA2, DSA3,

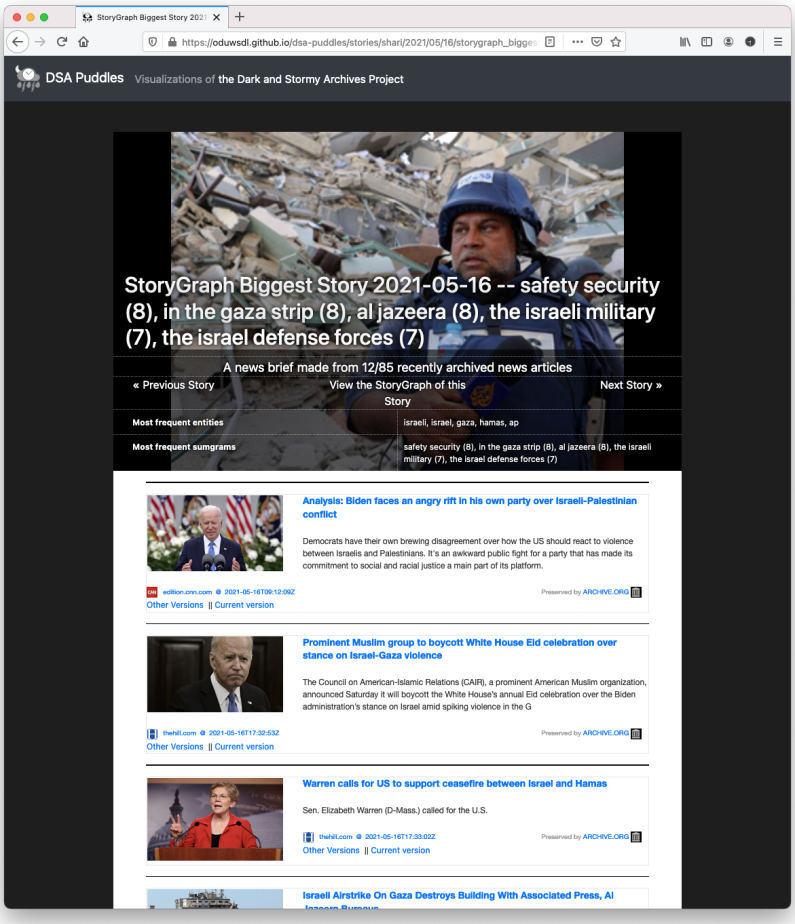

Fig. 5. A story generated via the SHARI process, providing an overview of the events on May 16, 2021.

and so on. We will also explore how to employ snowball sampling, possibly augmenting the list of mementos from the input through search engines, crawling, or other external services. As part of a recent IIPC grant [Nelson 2021], we are also developing a graphical user interface for Hypercane.

Hypercane is another tool for the web archivist's toolkit. We initially conceived it to summarize web archive collections for storytelling, but along the way realized that we might want to reuse these samples with other tools as well. We hope that this tool will enable others to further explore web archives and produce interesting samples and stories using resources from web archives.

ACKNOWLEDGMENTS

This work has been partially supported by an Institute of Museum and Library Services (IMLS) National Leadership Grant (LG-71-15-0077-15) and by the International Internet Preservation Consortium (IIPC) Discretionary Funding Program.

REFERENCES

ADOBE INC. 2021. Referrer type. https://experienceleague.adobe.com/docs/analytics/components/dimensions/referrer-type.html?lang=en.

ALNOAMANY, Y., WEIGLE, M. C., AND NELSON, M. L. 2016. Detecting off-topic pages within TimeMaps in web archives. *International Journal of Digital Libraries (IJDL) 17,* 3, 203–221. https://doi.org/10.1007/s00799-016-0183-5.

ALNOAMANY, Y., WEIGLE, M. C., AND NELSON, M. L. 2017. Generating stories from archived collections. In *Proceedings of the 9th International ACM Web Science Conference*. Troy, NY. https://doi.org/10.1145/3091478.3091508.

ARCHIVE-IT. 2014. About Archive-It. https://www.archive-it.org/blog/learn-more/.

ATURBAN, M., KELLY, M., ALAM, S., BERLIN, J., NELSON, M. L., AND WEIGLE, M. C. 2018. Archivenow: Simplified, extensible, multi-archive preservation (poster). In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL)*. Fort Worth, TX, 321–322. https://doi.org/10.1145/3197026.3203880.

ESTER, M., KRIEGEL, H.-P., SANDER, J., AND XU, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. 226–231. https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf.

HONNIBAL, M., MONTANI, I., VAN LANDEGHEM, S., AND BOYD, A. 2020. spaCy: Industrial-strength Natural Language Processing in Python. https://doi.org/10.5281/zenodo.1212303.

ISO 28500:2017 2017. Information and documentation — WARC file format. Standard, International Organization for Standardization.

JONES, S. 2018. Extracting metadata from Archive-It collections with Archive-It Utilities. https://ws-dl.blogspot.com/2018/07/2018-07-03-extracting-metadata-from.html.

JONES, S. 2019. Raintale – a storytelling tool for web archives. https://ws-dl.blogspot.com/2019/07/2019-07-11-raintale-storytelling-tool.html.

JONES, S. 2020a. Hypercane part 1: Intelligent sampling of web archive collections. https://ws-dl.blogspot.com/2020/06/2020-06-03-hypercane-part-1-intelligent.html.

JONES, S. 2020b. Hypercane part 2: Synthesizing output for other tools. https://ws-dl.blogspot.com/2020/06/2020-06-10-hypercane-part-2.html.

JONES, S. 2020c. Hypercane part 3: Building your own algorithms. https://ws-dl.blogspot.com/2020/06/2020-06-17-hypercane-part-3-building.html.

JONES, S., KLEIN, M., WEIGLE, M. C., AND NELSON, M. L. 2020. MementoEmbed and Raintale for web archive storytelling. Tech. Rep. 2008.00137, arXiv. Aug. Presented at the ACM/IEEE JCDL 2020 Workshop on Web Archiving and Digital Libraries (WADL), http://arxiv.org/abs/2008.00137.

JONES, S., NWALA, A., KLEIN, M., WEIGLE, M. C., AND NELSON, M. L. 2020. SHARI – An Integration of Tools to Visualize the Story of the Day . Tech. Rep. 2008.00139, arXiv. Aug. Presented at the ACM/IEEE JCDL 2020 Workshop on Web Archiving and Digital Libraries (WADL), http://arxiv.org/abs/2008.00139.

JONES, S. M. Hypercane: Intelligent sampling of web archive collections. https://hypercane.readthedocs.io/.

JONES, S. M., NEBLITT-JONES, V., WEIGLE, M. C., KLEIN, M., AND NELSON, M. L. 2021. It's all about the cards: Sharing on social media probably encouraged HTML metadata growth. Tech. Rep. abs/2104.04116, arXiv. https://arxiv.org/abs/2104.04116.

JONES, S. M., NWALA, A., WEIGLE, M. C., AND NELSON, M. L. 2018. The many shapes of Archive-It. In *Proceedings of the International Conference on Digital Preservation (iPres)*. Boston, MA. https://doi.org/10.17605/OSF.IO/EV42P.

JONES, S. M., WEIGLE, M. C., AND NELSON, M. L. 2018. The Off-Topic Memento Toolkit. In *Proceedings of the International Conference on Digital Preservation (iPres)*. Boston, MA. https://doi.org/10.17605/OSF.IO/UBW87.

JONES, S. M., WEIGLE, M. C., AND NELSON, M. L. 2019. Social Cards Probably Provide For Better Understanding Of Web Archive Collections. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*. 2023–2032. https://doi.org/10.1145/3357384.3358039.

MCCOWN, F., CHAN, S., NELSON, M. L., AND BOLLEN, J. 2005. The Availability and Persistence of Web References in D-Lib Magazine. Tech. Rep. cs/0511077, arXiv. Presented at the 5th International Web Archiving Workshop (IWAW05), https://arxiv.org/abs/cs/0511077.

NATIONAL LIBRARY OF AUSTRALIA. PANDORA. http://pandora.nla.gov.au.

NATIONAL LIBRARY OF AUSTRALIA. Trove. https://trove.nla.gov.au.

NEGULESCU, K. C. 2010. Web Archiving @ the Internet Archive. http://www.digitalpreservation.gov/meetings/documents/ndiipp10/NDIIPP072110FinalIA.ppt. Presentation at the 2010 Digital Preservation Partners Meeting.

NELSON, M. L. 2021. Improving the Dark and Stormy Archives Framework by Summarizing the Collections of the National Library of Australia. https://netpreserve.org/projects/dark-and-stormy-archives/.

NWALA, A. 2019. Introducing sumgram, a tool for generating the most frequent conjoined ngrams. https://ws-dl.blogspot.com/2019/09/2019-09-09-introducing-sumgram-tool-for.html.

NWALA, A. C., WEIGLE, M. C., AND NELSON, M. L. 2020. 365 Dots in 2019: Quantifying Attention of News Sources. Tech. Rep. abs/2003.09989, arXiv. https://arxiv.org/abs/2003.09989.

OU-YANG, L. 2013. Newspaper3k: Article scraping & curation. https://newspaper.readthedocs.io/en/latest/.

PADIA, K., ALNOAMANY, Y., AND WEIGLE, M. C. 2012. Visualizing Digital Collections at Archive-It. In *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*. 15–18. https://doi.org/10.1145/2232817.2232821.

PEW RESEARCH CENTER. 2019. State of the News Media Methodology. https://pewresearch-org-preprod.go-vip.co/journalism/2019/07/23/state-of-the-news-media-methodology/#digital-native-news-outlet-audit.

ŘEHŮŘEK, R. AND SOJKA, P. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, 45–50. http://www.lrec-conf.org/proceedings/lrec2010/workshops/W10.pdf.

SALAHELDEEN, H. M. AND NELSON, M. L. 2013. Carbon Dating the Web: Estimating the Age of Web Resources. In *Proceedings of the 22nd International Conference on World Wide Web*. Rio de Janeiro, Brazil, 1075–1082. https://doi.org/10.1145/2487788.2488121.

VAN DE SOMPEL, H., NELSON, M., AND SANDERSON, R. 2013. RFC 7089 - HTTP Framework for Time-Based Access to Resource States – Memento. https://tools.ietf.org/html/rfc7089.

W3NEWSPAPERS.COM. 2021. List of popular news websites. https://www.w3newspapers.com/newssites/.

WIKIPEDIA CONTRIBUTORS. 2020. List of image-sharing websites. https://en.wikipedia.org/wiki/List_of_image-sharing_websites.

WIKIPEDIA CONTRIBUTORS. 2021. List of online video platforms. https://en.wikipedia.org/wiki/List_of_video_hosting_services#Specifically_dedicated_video_hosting_websites.

XAPIAN PROJECT. The BM25 weighting scheme. https://xapian.org/docs/bm25.html.

Shawn M. Jones is a graduate research assistant working for the Prototyping Team at the Los Alamos National

Laboratory's Research Library. Shawn is also a PhD candidate at Old Dominion University and a member of the Web Science and Digital Libraries research group, led by Michael Nelson. He participates in projects and frameworks such as Memento, Signposting, and Robust Links. In addition, Shawn is the lead of the Dark and Stormy Archives project, an initiative to summarize web archive collections through visualization techniques common in social media.

Michele C. Weigle is a Professor of Computer Science at Old Dominion University. Her research interests include web science, digital preservation, social media, and information visualization. She has published over 100 articles in peer-reviewed conferences and journals and has been PI or Co-PI on over $2M in funding for research related to web archiving from NSF, NEH, IMLS, IIPC, and the Andrew W. Mellon Foundation. Dr. Weigle received her PhD in computer science from the University of North Carolina at Chapel Hill in 2003.

Michael L. Nelson is a professor at Old Dominion University and the Virginia Modeling, Analysis, and Simulation Center (VMASC). Prior to joining ODU, he worked at NASA Langley Research Center from 1991-2002. More information about Dr. Nelson and his research group can be found at: oduwsdl.github.io.