Lecture 3: Reading Papers
Lecture Topics

• How do you read a research paper?
• How do you present a research paper?
Reading academic papers: Why?

• Important for literature reviews
  – Must think critically!
  – What's good? What's not? What would you have done differently?

• However... this is someone's work
  – Someone was proud of this
  – Often essential for career advancement

• Important for informing your work!
Reading academic papers: How

• It takes 3-5 reads to **really** understand an academic paper
  – Often spread over 2 weeks
  – “Reads” will vary in what you cover

• Set aside enough time when reviewing or reading a paper

• A single read will NOT give you enough information on a paper!
Reading academic papers: Method

• Read 1:
  – Title, authors, section headers
  – Ignore content!
  – You may read the abstract if desired

• Read 2 (next day):
  – Read the abstract, introduction, and conclusion
  – Look at the figures/tables
  – Review the references – did they cite canonical works?
  – Begin asking:
    • What do the authors want me to understand?
    • What are the main contributions?

• Read 3 (next day):
  – Read the entire paper
  – Do NOT make notes!
  – Focus on understanding the methodology and contributions

• Next reads (2-3 days later):
  – Re-read the entire paper
  – Make notes; what is good? Not good? What would you have done different?
  – Subsequent reads should focus on understanding clarity, quality, etc.

• Continue re-reading the paper until you can replicate the research!
This is a paper

- It happens to be one of mine
- JCDL2017
- ACM
- 2-column format
Step 1:

- Read the title, authors' names
- Ignore most of the content
- Read the section headings throughout the paper
4 DESCENDANT MODEL

Dreier et al. [1] present a model for caching in a client-side environment that uses a tree structure to represent the hierarchical relationships between resources. In this model, the tree is constructed using the Document Object Model (DOM) structure of the web page, with each node representing an element in the DOM that includes JavaScript code.

In this model, the root node represents the entire page, and each child node represents a child element of the parent node. The model allows for efficient caching of the page by caching the entire tree structure and reusing it for subsequent requests.

The model also includes a mechanism for handling resource requests, such as image requests, by using a caching mechanism that stores the responses to these requests in memory. This allows for faster response times and improved performance.

5 APPROACH

In this section, we present a new approach for collecting and analyzing data from web pages. Our approach involves using a combination of machine learning and web crawling techniques to collect data from various sources.

We begin by collecting a large dataset of web pages from a variety of sources, including news websites, social media platforms, and e-commerce sites. We then use natural language processing techniques to extract relevant information from the collected data.

Our approach also includes the use of machine learning algorithms to identify patterns and trends in the data. This allows us to make predictions about future trends and identify new opportunities for growth.

We then use this data to create a model that can be used to predict future trends and identify new opportunities for growth. This model can be used to inform decision-making and improve strategies for brand building and marketing.

6 STATE EQUVALENCY

In this section, we present a new method for verifying the equvalency of state descriptions in JavaScript. Our approach involves using a combination of static and dynamic analysis techniques to compare the state descriptions of different JavaScript programs.

We begin by collecting a large dataset of state descriptions from various sources, including web pages, browser extensions, and mobile applications. We then use static analysis techniques to extract relevant information from the state descriptions.

We then use dynamic analysis techniques to compare the state descriptions of different JavaScript programs. This allows us to verify the equvalency of state descriptions and identify potential errors in the code.

Our approach can be used to improve the reliability and security of JavaScript programs by identifying potential errors and preventing attacks.

7 FUTURE WORK

In this section, we discuss potential future work that could be done in this area. We believe that there is a need for further research into the equvalency of state descriptions in JavaScript and how these descriptions can be used to improve the reliability and security of JavaScript programs.

We also believe that there is a need for more research into the use of state descriptions in the development of web applications. This could include the use of state descriptions in the design and development of web applications, as well as the use of state descriptions in the testing and deployment of web applications.

We hope that our research will stimulate further interest in this area and lead to new and exciting developments in the field of JavaScript and web development.
Step 2:

- Read the abstract, introduction, conclusion
- Look at the figures
- Try to understand what we will learn
- What are the authors contributing?
Archival Crawlers and JavaScript: Discover More Stuff but Crawl More Slowly

Justin F. Brinelle
The MITRE Corporation
jbrinelle@mitre.org

Michele C. Weagle
Old Dominion University, Department of Computer Science
weagle@odu.edu

Michael L. Nelson
Old Dominion University, Department of Computer Science
mnelson@odu.edu

ABSTRACT
The web is today’s primary publication medium, making web archiving an important activity for historical and analytical purposes. Web pages are increasingly interactive, resulting in pages that are correspondingly difficult to archive. JavaScript enables interactions that can potentially change the client-side state of a representation. We refer to representations that had embedded resources via JavaScript as deferred representations. It is difficult to discover and crawl all of the resources in deferred representations and the result of archiving deferred representations is archived web pages that are either incomplete or erroneously omissioned resources from the live web. We propose a method of discovering and archiving deferred representations and their descendants (representation states) that are only reachable through client-side events. Our approach identified an average of 38.5 descendants per seed URI crawled, 70.9% of which were reachable through an event fired. This approach also added 15.6 times more embedded resources than Hittite to the crawl frontier, but at a crawl rate that was 38.9 times slower than simply using Hittite. If our method was applied to the July 2015 Common Crawl dataset, a web-scale archival crawler will discover an additional 7.17 PB (8.12 times more) of information per year. This illustrates the significant increase in resources necessary for more thorough archival crawls.

CCS CONCEPTS
- Information systems → Digital libraries and archives

KEYWORDS
Web Archiving, Digital Preservation; Moments, Web Crawling

1. INTRODUCTION
As the web grows as the primary medium for publication, communication, and other services, so grows the importance of preserving the web as evidenced by recent articles in PAB, Yahoo (20), and the [27]. Web resources are augmented, existing in the perspective of web, important historical events frequently disappear from the web without preservation. We may gain pages because we are not aware they should be saved or because the pages themselves hard to access. In 2014, a Ukrainian government announced a two wars social media[1], with video evidence, that they shut down military cargo plans in Ukrainian airspace. Evidence suggests the domain was the commercial Malaysian Airlines Flight 17 (MH17). The Ukrainian separatists moved from social media's claim of shooting down what was a non-military passenger plane. The Internet Archive [1], using the Hittite web crawler [19], was crawling and archiving the social media site twice daily and as such the claim was hard to access. It is evidence that Ukrainian separatists shut down MH17 in 2014. This among others[1] is an example of the importance of archiving web archiving to reveal history and establish evidence of information published on the web.

However, not all historical events are archived for the future as is the 17th century example. In an attempt to limit online privacy and theft of intellectual property, the U.S. Congress passed the widely unpopular Stop Online Privacy Act (SOPA) [17]. While the attempted passage of SOPA may be an important step in history, the overhanging cloud response is significant. On January 19, 2012, many prominent websites organized a world-wide blackout in protest of SOPA. Wikipedia blackout on January 18, 2012, led to a “splash page” that prevented access to Wikipedia content.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:

1 CONCLUSIONS
In this paper, we present a model for crawling deferred representations by identifying interactive portions of pages and discovering descendants. We adopt prior work by Biener et al. and present a FSM to discover ancestors and estimate storage requirements for the descendants. We show that the deferred representations from our 50 URI-R sample set has 38.5 descendants per URI-R, and are surprisingly shallow, only reaching a depth of two levels. This means that these deferred representations are shallower than originally anticipated (but also very broad) and therefore it is more feasible to completely archive deferred representations using automated methods than previously thought. Archives that do not execute JavaScript during archiving are incomplete, 90% of URI-Rs have descendants and 90% of the embedded resources in those descendants are not archived. Crawler all descendants is 38.5 times slower than crawler with only Hittite, and adds 15.6 times more URI-Rs to the crawl frontier than Hittite alone.

Crawling all descendants in the sample is 38.9 times slower than crawling with only Hittite, but adds 15.6 times more URI-Rs to the crawl frontier than Hittite alone. However, most of URI-R (newly added by traversing pages) was unreachable (96% unreachable, and assumed to be unreachable, at t=2 and 90% at t=3). However, 22.4% of the newly discovered URI-Rs using our model was one of the top 10 occurring URI-Rs, indicating a high amount of overlap between Hittite. Thus, these are ad servers and data-services like Google Analytics. In the future, we will work to incorporate PhantomJS into a web crawler to measure the actual benefits and increase archival coverage realized when crawling deferred representations. We will also work to develop an approach to solve our current edge cases (Sections 7), including a way to host applications like mapping applications using our automated approach along with an approach using “cached interactions”. Our goal is to understand how many executions of cached interactions are necessary to uncover an acceptable threshold of embedded resources (e.g. how many pages and zooms are needed to get all Google Maps tiles for all of Norfolk, VA, USA?). We will also investigate filling out forms similar to Rosenthal et al. [80].

This concludes this paper; provides measurements for a well-known phenomenon as it occurs in a small sample of URI-Rs, establishes an understanding of how much web archives and archiving is hampered by not executing JavaScript, and presents a process for better archiving deferred representations. We demonstrate that archiving deferred representation is a core daunting task with regards to...
Archival Crawlers and JavaScript: Discover More Stuff but Crawl More Slowly

Justin F. Brumelle
The MITRE Corporation
jfb@mitre.org

Michele C. Weagle
Old Dominion University,
Department of Computer Science
weagle@odu.edu

Michael L. Nelson
Old Dominion University,
Department of Computer Science
mnelson@odu.edu

ABSTRACT

The web is today's primary publication medium, making web archiving an important activity for historical and analytical purposes. Web pages are increasingly interesting, resulting in pages that are correspondingly difficult to archive. JavaScript enables interactions that can potentially change the client-side state of a representation. We refer to representations that had embedded resources via JavaScript as deferred representations. It is difficult to discover and crawl all of the resources in deferred representations and the result of archived deferred representations is archived web pages that are either incomplete or erroneously archived resources from the live web. We propose a method of discovering and archiving deferred representations and their descendents (representation state) that are only reachable through client-side events. Our approach identified an average of 38.5 descendents per seed URI crawled, 70.9% of which were reachable through on-click events. This approach also added 15.6 times more embedded resources than Heritrix to the crawl frontier, but at a crawl rate that was 58.9 times slower than simply using Heritrix. If our method was applied to the July 2015 Common Crawl dataset, a web-scale archival crawler will discover an additional 7.17 PB (5.12 times more) of information per year. This illustrates the significant increase in resources necessary for more thorough archival crawls.

CCS CONCEPTS
- Information systems → Digital libraries and archives

KEYWORDS

Web Archiving; Digital Preservation; Mentions; Web Crawling

1 INTRODUCTION

As the web grows as the primary medium for public information, communication, and other services, it is clear that the importance of preserving the web (as evidenced by recent articles in The New York Times [28] and The Atlantic [29]) is not diminishing. Web resources are ephemeral, existing in the perpetual moment; important historical events frequently disappear from the web without being preserved or recorded. We may rate pages because we are not aware they should be saved or because the pages themselves are hard to archive.

On July 17, 2015, Ukrainian separatists announced via social media, with video evidence, that they shot down a military cargo plane in Ukraine. Evidence suggests that the downing was the commercial Malaysian Airlines Flight 17 (MH17). The Ukrainian separatists removed from social media their claim of shooting down what was now known as the non-military passenger plane. The Internet Archive [32], using the Heritrix web crawler [33, 41], was crawling and archiving the social media site twice daily and archived the claimed credit for downing the aircraft, that is evidence that Ukrainian separatists shot down MH17 [7]. This (along with others [1]) is an example of the importance of historical web archiving to record history and establish evidence of information published on the web.

However, not all historical events are archived as fortuitously as the MH17 example. In an attempt to limit online piracy and theft of intellectual property, the U.S. Government proposed the widely unpopular Stop Online Piracy Act (SOPA) [37]. While the attempted passing of SOPA may be a more footnote in history, the overwhelming protest in response is significant. On January 18, 2012, many prominent websites organized a worldwide blackout in protest of SOPA. Wikipedia blocked out its site by using Inband to load a "splash page" that prevented access to Wikipedia's content.

11 CONCLUSIONS

In this paper, we present a model for crawling deferred representations by identifying interactive portions of pages and discovering descendents. We adapt prior work by Dinu et al. and present a FSM to describe descendents and estimate storage requirements for the descendents.

We show that the deferred representations from our 480 URI-R collections have 38.5 descendents per URI-R, are surprisingly shallow, only reaching a depth of two levels. This means that these deferred representations are shallower than originally anticipated (but also very broad) and therefore it is more feasible to fully archive all deferred representations using automated methods than previously thought. Archives that do not use JavaScript during archiving are incomplete; 69% of URI-Rs have descendents and 90% of the embedded resources in those descendents are not archived.

Crawling all descendents is 38.9 times slower than crawling with only Heritrix, but adds 13.50 times more URI-Rs to the crawl frontier than Heritrix alone. Crawling all descendents in a single URI-R is 38.9 times slower than crawling with only Heritrix, but adds 13.50 times more URI-Rs to the crawl frontier than Heritrix alone. However, most of the new URI-Rs are not reachable from the page being crawled (i.e., they have an unarchived, unarchived, or the top occurring URI-R, indicating a high amount of overlap within PDP, mostly those are ad servers and data-services like Google Analytics). In the future, we will incorporate deferred representations into a web crawler to measure the actual benefit and increased archival coverage realized when crawling deferred representations. We will work to develop an approach to solve the current edge cases (Section 7), including a way to handle applications using our automated approach along with an approach using "canned interactions". Our goal is to understand how many executions of each of these interactions are necessary to uncover an acceptable end of embodied resources (e.g., how many pages and boxes are needed to get all GeoMap data tiles for all of Northern VA, USA7). We will also investigate filtering out forms similar to Egelra et al. [10].

Our work presented in this paper provides measurement for performance characteristics that occur in a small sample of URI-Rs, establishes an understanding of how much web archival and crawlers are missing from accurately crawling deferred representations, and presents a process for better archival descendents. We demonstrate that archived deferred representation is a free daunting task with regards to
Step 3+:

- Read the entire paper
- Keep re-reading until you understand the work as well as the authors
Presenting academic papers: Why?

• Researchers must communicate their work
  – Important for career advancement
  – Important for networking
  – Important for citations (see Career Advancement…)

• Doing great things isn't enough – peers must know about and understand your work
Presenting academic papers: How?

- Tell 'em what you're gonna tell 'em
- Tell 'em
- Tell 'em what you told 'em
- Relies on storytelling, salesmanship, and clarity
Presenting academic papers: Method

- 1-2 slides -- Introductory bio
- 2-4 slides -- Introduce the topic area (includes common myths and associated truths)
- 1-2 slides -- Introduce the paper (including motivation)
- 1-2 slides -- Background and related works
- 4-5 slides -- Research methodology (include experimental design, metrics, data sets, etc.)
- 1-2 slides -- Findings & conclusions (include relevant graphs)
- 1-2 slides -- Issues and recommendations with the paper
- 1-2 slides -- Recommendations for research in the topic area
- Presentations should range from 15-20 slides and last for 30-45 minutes with 15-30 minutes for Q&A and discussion
Archival Crawlers and JavaScript: Discover More Stuff but Crawl More Slowly

Justin F. Brunelle (MITRE)
Michele C. Weigle (ODU)
Michael L. Nelson (ODU)

JCDL2017
Just enough background – Prove to the audience that there is a problem

Introduce the topic area
Introduce your solution/research

• Explain why you think your work is brilliant and so should your audience!

• Be clear; tie the solution back to the problem (preferably with examples)
Prove that your research is brilliant

- Graphs, stats, NUMBERS! Prove your evaluation out!
- Compare to existing frameworks/solutions if possible
- Use examples or demonstrations where appropriate
Recap your results

- Summarize/recap the problem
- Recap the important parts of your approach
- Remind the audience of the most relevant stats and findings

---

**Storage Impact of Two-Tiered Crawling**

- IIPC-proposed JSON metadata of interactions, resulting descendants
  - Potentially used to resolve URI-M collisions
  - 16.5KB WARC metadata
  - 143MB for total dataset
- 11.4 times larger for deferred vs nondeferred
- Totals 5.12 times more storage per URI-R for total dataset

Findings and conclusions

© 2017 The MITRE Corporation. All rights reserved.
End with the takeaways

- Tell 'em what you told'em
- Include future works – how should you or another researcher extend this work?

Conclusions and Future Work

- Archival Halting Problem: How much is enough?
  - Mapping Applications – How many pans and zooms gets all the Toronto Google map tiles?
  - How many CNN.com pages get all the Google Ads?
- Crawl more slowly but more thoroughly
- Defined descendants, showed that they are 2-levels deep
- Descendant embedded resources are largely unarchived
- Storage impact of crawling descendants and deferred representations

Inform crawl owners regarding depth versus coverage of deferred representations

Recommended research in the topic area
Things to keep in mind

• Audience:
  – Technical? Non-technical?
  – Domain experts? New to the space?

• Time limit
  – Leave time for questions
  – Questions are a compliment!

• Pictures!!
  – Power point is a presentation TOOL
  – Your soundtrack is the real star
More things to keep in mind

- Cite sources, particularly of figures
- Cheap clipart makes a cheap presentation
  - Use only useful graphics
  - Annotate as appropriate
- Use slide numbers
Let's see additional info

- https://www.slideshare.net/mweigle/how-to-prepare-and-give-and-academic-presentation
- Slide 17
Presentation Tips

• Practice, then practice more
• Don't read, don't memorize, don't script
• Have canned emphasis statements
  – For example, if you have a tagline that you want remembered, practice the delivery
• Face the audience
• Thank the audience and any contributors
• You're a performer!
If you're nervous...

- Power poses actually work
  - Some brain scientists proved it...
- Practice, then practice more
- Use ice breakers to energize yourself
  - Let's practice! Paper towel roll game
Tips for reviewing papers

• If you don't have the time, say no
  – Reviews should be several paragraphs long
  – Reviews require thoughtful and clear responses

• Recap the paper so that the author knows you understood it

• This is someone's work; be firm, gentle, and back up any criticism with examples
  – Don't be too harsh! For whatever reason, CS reviewers can be extremely critical
  – Reviewer 3...

• Provide recommendations for improvement if needed

• Given praise where appropriate