

# Algorithmic Awareness - Action Handbook

## OVERVIEW

This is a practice-oriented action handbook that provides background, resources, and best practices to guide libraries in implementing *Algorithmic Awareness* as a digital literacy.

Our Montana State University team, with [grant funding from the Institute of Museum and Library Services \(IMLS\)](#), has conducted research in order to develop an **Action Handbook**, a web-based teaching tool, and a standard curriculum to support the teaching of "*Algorithmic Awareness*": an understanding around the rules that govern our software and shape our digital experiences. We pursued this research because our technological experiences are increasingly mediated by algorithms - the code and computational processes embedded in our software. Recent work by scholars, such as Dr. Safiya Umoja Noble, has shown [how algorithms exhibit implicit biases and reify societal prejudices](#). Moreover, the technical nature of algorithms and the lack of transparency surrounding them can be a challenge for novices. We, and our patrons, routinely engage in systems that predict, recommend, and speculate about our interests based on the digital fingerprint we provide with our link clicks and "likes", but we all struggle understanding how and why those systems work as they do. In seeking to understand common systems, like the Facebook news feed or the Google search engine results page, we view this research as an opportunity to discover the scope and reach of algorithms and how they might be taught. We are addressing a gap in our field: a lack of an understanding of the rules that govern our software and shape our digital experiences. Librarians and information professionals have a long history of teaching digital literacy; however, while our [ACRL Framework for Information Literacy](#) considers how authority is constructed, our instruction programs have not kept up with how algorithms in software construct our experience. We are calling for a new competency that we have termed "*Algorithmic Awareness*". Librarians and information professionals can apply this new competency to enrich their instruction programs and extend their relevancy by defining an emerging form of digital literacy for their patrons. At its core, this research is about introducing new expertise into our profession and providing a new teaching moment for the field.

This **Action Handbook** consists of two parts with an introduction describing the in-depth definition of an algorithm. The first part will focus on technical aspects of teaching *Algorithmic Awareness* including background and implementation of teaching modules, an overview of the teaching tool, and the auditing computational software. The second part will cover the material

consequences of algorithms, provide a literature review, discuss the survey of the field, explain *Algorithmic Awareness* as a form of digital literacy, and provide ways to advocate algorithmic awareness. All of the [research project deliverables are available as Open Educational Resources \(OERs\) in our public GitHub repository](#). While there are arguably many ramifications surrounding gender and racial bias, these aspects of the paper will be limited in order to maintain a restricted scope.

This guide contains two main parts, followed by an appendix.

- Part 1— **Technical**: defining algorithm as a term, describing the teaching tool, auditing computational software, explaining the first principles of algorithms, noting the modules and methods for teaching *Algorithmic Awareness*.
- Part 2— **Social**: introducing the problem, reviewing the literature, discussing the survey of the field, explaining *Algorithmic Awareness* as a form of digital literacy, advocating for *Algorithmic Awareness*.

## TECHNICAL — Implementation Strategies for Algorithmic Awareness

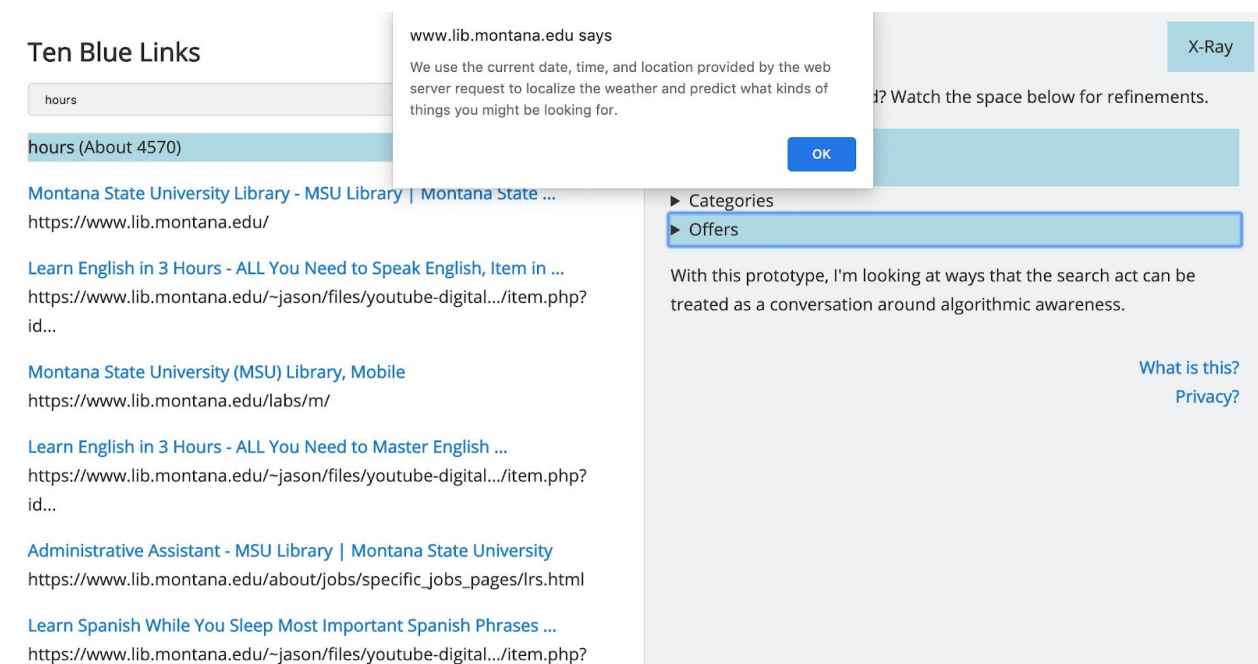
The definition of “algorithm” has evolved over time and depending upon a given context. The common definition according to Wikipedia is “an unambiguous specification of how to solve a class of problems” in relation to mathematics and computer science.” Like a math problem, an algorithm is a set of rules that dictates how an input  $x$  results in the output  $y$ . Another definition derived by Kevin Slavin, an assistant professor and founder of the Playful Systems group at the MIT Media Lab, is that algorithms, “the math that computers use to decide stuff... they repeat over and over again, and they ossify and calcify,” becoming a real force in the world ([Kevin Slavin: TED Talk, How Algorithms Shape Our World](#)). Matthew Reidsma, a researcher focusing on design ethics and user experience, provides another way to look at the definition of an algorithm. Reidsma claims a simple interpretation of a complex idea perpetuates misunderstanding. Instead, he claims, algorithms should be viewed as a large set of binary choices that create a "decision tree" based upon many variables that are not necessarily apparent to the user.

For our purposes, we are defining algorithms in terms of how they are applied within computer programs. In this context, an algorithm is an “instance of logic written in software by software developers to be effective for the intended "target" computer(s) to produce output from given (perhaps null) input” ([Wikipedia article on Computer Algorithms](#), 2017). Our focus is on how these logical rules are explicated and can be brought into the library’s pedagogical environment or a

classroom.

## Teaching Tool

As a frame of reference for algorithms, our research looked at how search algorithms work in practice. Even today, search remains a key component of the user experience. What is often misunderstood are the digital machinations and the ghosts in the [search] machine that produce search results for our users. To this end, our team created a teaching tool to demonstrate transparent software design. (Code is available in [our Montana State University Library GitHub](#) repository.) The [teaching tool is a search engine application that provides users the opportunity to see an “x-ray” of the variables affecting their query](#). For example, depending upon weather and location data, the search for the Montana State University Library may recommend a hot beverage at the Brewed Awakening coffee stand.



[Figure 1: The algorithmic teaching tool annotating the algorithms in use to create the search results view. The annotations are tooltips that one activates with the X-Ray button (top right).]

The goal here is to allow teachers to demonstrate common search algorithms, but give them and their learners a chance to slow down the process and highlight what is happening when a search facet is applied or a natural language query is interpreted by the machine. Some of the algorithms we demonstrate include: PageRank, merge sort and heap sort, Dijkstra’s algorithm, link analysis, and TF-IDF (Term Frequency-Inverse Document Frequency). users are able to interact with these algorithmic variables to facilitate understanding usually hidden in the search algorithmic process. In the process, students gain a perspective on how algorithmic code

becomes an interaction and we are able to establish how theory leads into practice. The tool provides a glimpse of algorithms in action and can be an essential learning aid when a teacher needs an example of algorithms within a common experience, like searching online.

### **Auditing Computational Software**

Part of the challenge of algorithmic literacy is the hidden and proprietary rules that create our digital experiences are not transparent. In an effort to reverse engineer these rules and the strategies for discerning the hidden components of a software experiences, the team conducted research and created a series of auditing strategies and scripts to look at these hidden processes. With this work, our team developed algorithm auditing practices using code, web scraping methods, and structured data analysis to uncover proprietary algorithms. (Code is available in [our Montana State University Library GitHub](#) repository.) Our case study for this work was the YouTube Video Recommendation Algorithm which has come under [criticism for its tactics in drawing parents' and childrens' attention to their videos](#). Our goal was to show the generic patterns, data points, and scripts one can use to analyze algorithms and to demonstrate how code can be used to introduce transparency and digital literacies to our students. In the end, teachers who apply these routines will realize actionable steps for teaching about algorithms and gain a sense of the first steps one can take to programmatically audit these systems with code.

### **First Principles of Algorithms**

Teaching about the often obfuscated processes of algorithms requires that you establish a context or lived experience with algorithms in action. Common online experiences, like social media or search engines, can provide this context. We [created a reference worksheet explaining algorithms in action within common interfaces](#). Note that many of these algorithms are proprietary, but they have a foundation in common algorithms that we are calling “first-principle” algorithms. A teacher can use the simple explanations in this reference document to begin the technical conversations around *Algorithmic Awareness*.

### **Teaching Modules and Methods**

Our team has created three teaching modules to distribute to professionals as a means of enhancing and expanding the *Algorithmic Awareness* concept into a series of workshops. We have also [created a draft curriculum and syllabus for a semester-length \*Algorithmic Awareness\* course](#). Each module can be presented individually, collectively, or built upon to create a robust understanding of this digital literacy in conglomeration with a variety of academic interests. Listed below is an outline and a general overview of each module.

The purpose of **Module One** is to introduce the concept of *Algorithmic Awareness* while building a comprehensive overview of the contemporary concept through real-world examples such as Amazon, Netflix, and Facebook. Additionally, students will gain a brief synopsis of coding during the break out activity, pushing many outside their comfort zone in order to facilitate a discussion surrounding the bias of programming. Finally, students will be able to interact with the Teaching Tools and identify key user features of transparency, connecting all parts of the presentation together.

**Module One: Algorithmic Awareness**

- a. An introduction to the term and general overview of material consequences.  
This module delves into common encounters with algorithms such as E-commerce, entertainment, and social media
- b. Break out activity: Program a library
- c. Demonstrate the Teaching Tool

**Module Two** delves into many contemporary implications of biased algorithms from ad targeting to digital redlining to ground the importance of this research and *Algorithmic Awareness*. In the break out sessions, students will explore comparisons between various web browsers, queries, and digital profiles. Moreover, through a comprehensive activity, students will be empowered to download their Google or Facebook data profiles and discuss the nuanced discrepancies of their online profiles.

**Module Two: Data Profiles, Transparency, Personalization, and Algorithms**

- a. Discusses ad targeting and digital redlining
- b. Break out activities: anonymous and personal profile comparison
- c. Break out activity: data download

Shifting gears a bit, **Module Three** offers an exploration of the promising directions and positive changes such as GDPR and the software design codes of ethics. Furthermore, the Algorithmic Auditing tool shows students a specific example in Youtube of our efforts to mediate the black box of search recommendations. Finally, students will step into the shoes of software designs and collaborate in small groups to take established or hypothetical software and implement *Algorithmic Awareness* into its design, bringing their ideas to the plenary discussion.

**Module Three: Promising Directions in Critical Algorithm Studies**

- a. Introduces positive changes made: GDPR and codes of ethics
- b. Demonstrate the Algorithmic Auditing tool

- c. Break out activity: sprinting to create transparency in software and elevator pitch

### **Four-Point Plan for Teaching Algorithmic Awareness**

At times, algorithms have a way of defying explanation. Over the course of our teaching and workshop activity, we were able to understand a series of teaching principles that helped facilitate learning in multiple contexts. Refer to these principles as potential guidelines when you are wanting to introduce an algorithm for learners.

**Teaching Principle 1: Frame the algorithm with a material consequence.**

Example: Show how Google search results might impact a person's reputation.

**Teaching Principle 2: Introduce “First Principle” of the algorithm, not the math formula.** Example: Frame what the primary intention of the algorithm is.

**Teaching Principle 3: Demonstrate the algorithm in action.** Example:

Demonstrate the algorithm in a piece of software or online experience, like the Twitter timeline or Netflix recommendations.

**Teaching Principle 4: Leave space for the group to discuss.** Example: Set a discussion prompt asking how the learners experience the impact of a particular algorithm.

## **SOCIAL — Communication Strategies for Algorithmic Awareness**

In addition to the technical aspects and methodologies of *Algorithmic Awareness*, developing robust understanding requires engaging in dialogue with a variety of stakeholders. This section presents some strategies for handling some of the questions and (perhaps) objections that may emerge.

While this algorithmic work focuses primarily on instructional academics, it is important to ground this research in material consequences. Although contemporary issues are integrated into all of the instructions, Module Two focuses on data profiles, digging into deeper issues of targeted advertisements, and digital redlining. Cathey O'Neil's book, *Weapons of Math Destruction*, provides a comprehensive overview of online, for-profit university advertisements paired with high-interest loan options heavily advertised in traditionally low-income communities. An expensive degree with little value on the job market combined with extortionate accrued interest is a prime example of the inequalities algorithms can create. Digital Redlining, defined by Wikipedia as the “practice of creating and perpetuating inequalities

between racial, cultural, and class groups specifically through the use of digital technologies, digital content, and the internet,” and can be seen when big data analytics take the form of policing, housing discrimination, and service exclusion to name a few. These contemporary issues demonstrate how social inequality is easily codified, oscillated, and perpetuated.

### **Literature Review**

In our literature review, we noted increasing calls to consider the algorithm as an object of study. What was striking in this review was that most of the earlier studies looking at the efficacy or construction of algorithms. This focus on the practice of building algorithms is seen in many studies featuring the benchmarking or comparison of algorithms in practice (dos Santos et. al 2016, Kibekbaev et. al. 2017). However more recently, the research takes a turn towards the social impacts of algorithms. Safiya Umoja Noble was leading this discussion when she asked: “What kinds of results do Google’s search engine provide about black girls when keyword searching?” (Noble 2013). These questions referencing the deleterious effects of common algorithms are picked up again by Michelle Willson when she considers the place of algorithms in our everyday online experiences (Willson 2017). Even further, these questions are given a vocabulary when Taina Bucher introduces her concept of “the algorithmic imaginary - ways of thinking about what algorithms are, what they should be and how they function” in looking at people’s experience of the algorithms present in Facebook (Bucher 2017). Bucher’s concept of the “algorithmic imaginary” maps directly into our ideas around an “*Algorithmic Awareness*” competency for librarians. And finally, we were particularly interested in Rob Kitchin’s study where he notes the “three main challenges that hinder research about algorithms (gaining access to their formulation; they are heterogeneous and embedded in wider systems; their work unfolds contextually and contingently), which require practical and epistemological attention.” We agree with Kitchin’s formulation and see our project as an answer to his call for “practical and epistemological attention”. We also see the design of our project with our emphasis on algorithms in a prototype search application where we can “gain access to their formulation” as a means to bring some transparency to how the algorithms work in practice.

### **Survey of the Field**

To gain a better understanding of interest in further learning and teaching about algorithms, we surveyed a group of people present for our Associate of College & Research Libraries (ACRL) Digital Scholarship Section’s Open Research and Digital Collections Discussions webinar. Although this small group already demonstrates a strong interest in learning about *Algorithmic Awareness*, the survey questions delve into the understanding of algorithms, personal anecdotes, access to further instruction, and institutional infrastructure.

The results of this survey conveyed a general understanding of what an algorithm is, most respondents noting “mathematics,” “instructional” processes, and computational decision making. Respondents noted interacting with algorithms “everywhere,” many keying in on e-commerce, social media, and search queries in particular. About ninety-five percent of all respondents had an interest in learning more about algorithms, however only about fifteen percent had access to continued instruction on algorithmic digital literacy and of those, many noted minimal incorporation or assistance. Respondents had varying thoughts on when to integrate algorithmics learning into educational curriculums but all reported a need to teach about algorithms, regardless of the academic year.

Although a boutique audience, the purpose and result of this survey were to gauge the general interest within the field and offer novel insight for future research and implementation of *Algorithmic Awareness*.

### **Digital Literacy**

ACRL has a history of supporting and building the framework for institutes of higher education to design instructional sessions, courses, curricula, and assignments. This *Framework* envisions information literacy as, “the set of integrated abilities encompassing the reflective discovery of information, the understanding of how information is produced and valued, and the use of information in creating new knowledge and participating ethically in communities of learning” ([ACRL Framework for Information Literacy for Higher Education](#), Adopted January 2016). The six concepts, all of which appear in alphabetical order, that anchor the frames are *Authority is Constructed and Contextual*, *Information Creation as a Process*, *Information Has Value*, *Research as Inquiry*, *Scholarship as Conversation*, and *Searching as Strategic Exploration*.

Authority is Constructed and Contextual embodies many nuances surrounding *Algorithmic Awareness* and qualifies this research as a form of digital literacy. One overarching learning objective of the coursework is to empower the layman to question the ‘how’ and ‘why’ behind a digital black box. Module One’s breakout session to “program the library” and subsequent discussion are indicative of constructed and contextual authority. This exercise provides a theorized example of the workings of an algorithm. In addition, it gives an introduction to coding with a balance between technical and comprehensive learning. While many struggled with even pseudo coding, it became clear each individual had a different background, interest, and goals about how an anonymous individual might want to interact with the library. Stepping into the shoes of a programmer fostered conversations about the problem of viewing algorithms as a mathematical, logical, and unbiased authority.

### **Advocate Algorithmic Awareness**



In order to ground this research in contemporary actions, the final module features “Promising Directions,” ways to advocate for *Algorithmic Awareness*, and other work supporting transparency. The European Union’s General Data Protection Regulation (GDPR) is a positive example of a legal framework setting up guidelines not only for protocol on handling personal information, but additionally impacting transparency, user experience, and software design. Another progressive example is the recently developed code of ethics of the Association for Computing Machinery. Articles 2.7 and 3.2, in particular, emphasize “foster[ing] awareness” and “understanding” of computer systems and their limitations, including “transparency” when keeping social responsibilities in mind. In hopes of celebrating and continuing contemporary work, advocating for *Algorithmic Awareness* balances the acknowledgment of the many benefits of algorithms while empowering patrons and educators to seek a certain standard of such systems. With an awareness of the limitations of algorithms, patrons can have an educated say in digital policies, integration of algorithms into various societal fields, and the future of user experience.

## WORKS CONSULTED

Beer, David. 2017. "The social power of algorithms." *Information, Communication & Society* 20, no. 1: 1-13. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

Bucher, Taina. "The algorithmic imaginary: exploring the ordinary affects of Facebook algorithms." *Information, Communication & Society* 20, no. 1 (January 2017): 30-44. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

dos Santos, Henrique Lemos, Cristian Cechinel, and Ricardo Matsumura Araújo. 2017. "A comparison among approaches for recommending learning objects through collaborative filtering algorithms." *Program* 51, no. 1: 35-51. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

Kibekbaev, Azamat, and Ekrem Duman. 2016. "Benchmarking regression algorithms for income prediction modeling." *Information Systems* 61, 40-52. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

Kitchin, Rob. "Thinking critically about and researching algorithms." *Information, Communication & Society* 20, no. 1 (January 2017): 14-29. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

Noble, Safiya Umoja. "Google Search: Hyper-visibility as a Means of Rendering Black Women and Girls Invisible." *Invisible Culture: An Electronic Journal for Visual Studies*. Issue 19, Fall 2013. Visual & Cultural Studies Program, University of Rochester

Willson, Michele. "Algorithms (and the) everyday." *Information, Communication & Society* 20, no. 1 (January 2017): 137-150. Library, Information Science & Technology Abstracts, EBSCOhost (accessed June 9, 2017).

## APPENDIX

### Technical (tools and deliverables)

- Teaching Tool [\[Demo\]](#) and [\[Code\]](#)
- Auditing Software [\[Code\]](#)
- [Modules](#)
- [Syllabi](#)
- [Breakout Activities and Exercises](#)

### Reading List

Algorithmic Justice League. (n.d.). <https://www.ajlunited.org/>.

Anderson, J., & Rainie, L. (2017). [Code-dependent: Pros and cons of the algorithm age](#). Pew Research Center.

Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). [Machine Bias](#). *ProPublica*.

Benjamin, R. (2019). [Race After Technology: Abolitionist Tools for the New Jim Code](#). Polity.

Broussard, M. (2018). [Artificial Unintelligence: How Computers Misunderstand the World](#). Cambridge, Massachusetts: The MIT Press.

Crawford, K., & Paglen, T. (2019). Excavating AI. <https://www.excavating.ai>.

Eubanks, V. (2018). *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. St. Martin's Publishing Group.

Haraway, Donna J. (1986). ["A Cyborg Manifesto"](#).

Lynch, C. (2017). ["Stewardship in the "Age of Algorithms"](#). *First Monday*, 22(12).

Noble, S. (2018). *Algorithms of Oppression*. New York University Press.

Posner, M. (2018). [See No Evil](#). *Logic Magazine*.

Seaver, N. (2017). Algorithms as culture: Some tactics for the ethnography of algorithmic systems. *Big Data & Society*, 4(2). <https://doi.org/10.1177/2053951717738104>

## **AUTHORSHIP and ACKNOWLEDGEMENTS**

This *Action Handbook* is co-authored by [Jason A. Clark](#) and Julian Kaptanian. This publication is a part of [“RE:Search” - Unpacking the Algorithms That Shape Our UX](#) grant and is supported by the [Institute of Museum and Library Services \(IMLS\)](#). The structure of this handbook was provided by Scott W.H. Young.