

CS 4530 & CS 5500

Software Engineering

Lecture 11.2: Ethics in Software Engineering

Jonathan Bell, John Boyland, Mitch Wand
Khoury College of Computer Sciences
© 2021, released under [CC BY-SA](#)

Learning Objectives for this Lesson

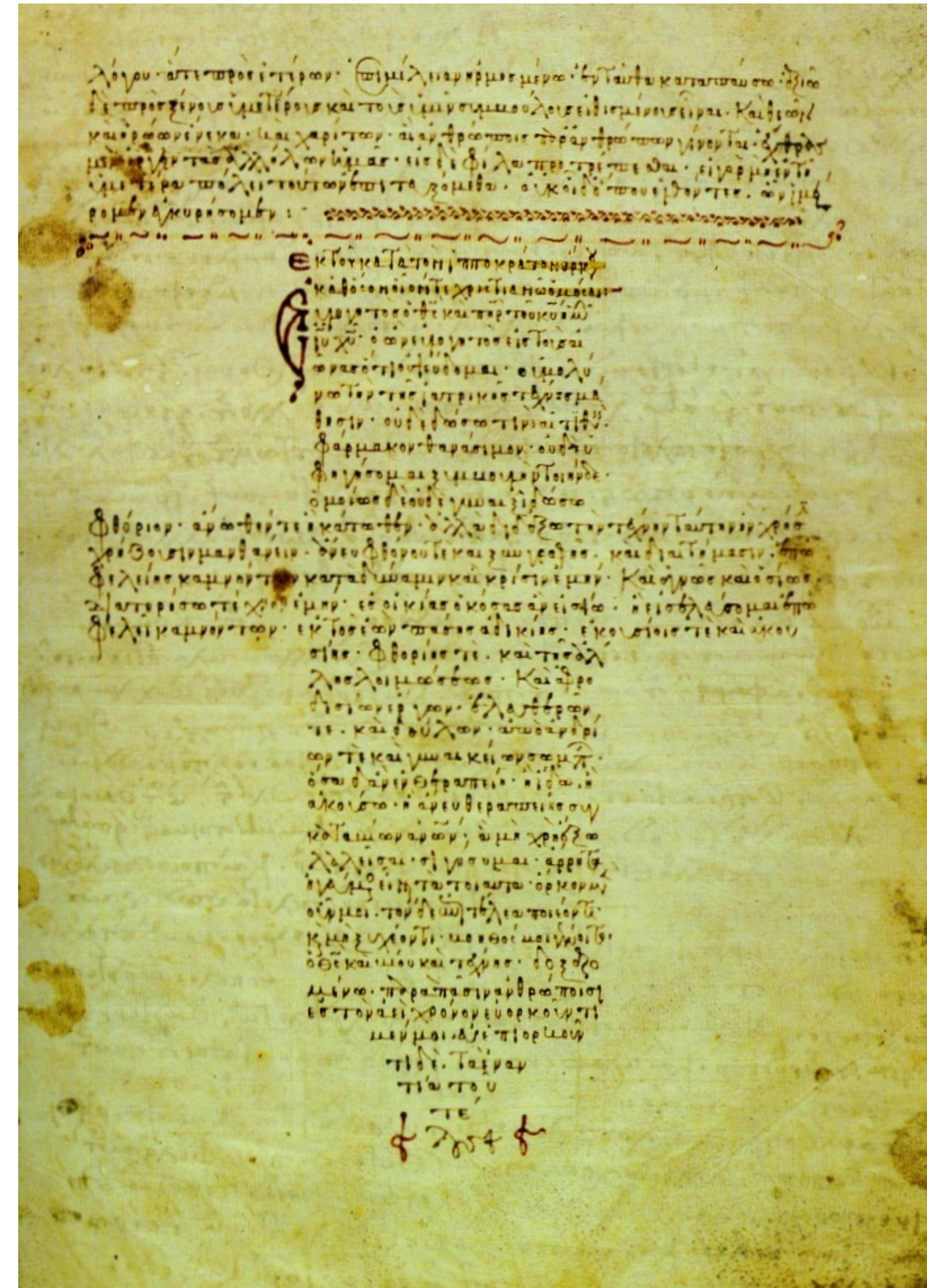
By the end of this lesson, you should be able to...

- Articulate the ethical responsibilities of professional software engineers
- List a set of questions that you can ask to evaluate the public impact of software

Professional Ethics

Professional standards

- By 1675, standards established for: divinity, law, medicine
- Professionals exercise specialist knowledge or skill - professional ethics governs how this knowledge should be governed



12th-century Byzantine manuscript of the Hippocratic Oath

Code of Ethics

Professional Engineers

Engineers, in the fulfillment of their professional duties, shall:

1. Hold paramount the safety, health, and welfare of the public.
2. Perform services only in areas of their competence.
3. Issue public statements only in an objective and truthful manner.
4. Act for each employer or client as faithful agents or trustees.
5. Avoid deceptive acts.
6. Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.



Code of Ethics

Professional Engineers: Citigroup Center

- Design met building code, but did *not* account for all failure modes
- Last-minute changes to construction increased odds of failure
- Fixed before disaster could strike, but kept a secret for 20 years



Code of Ethics

Software Engineers: Therac-25 (1985-1987)

- Bug in software caused 100x greater exposure to radiation than intended
- At least 6 died
- Likely far more suffered: deaths occurred over a period of 2 years!
- Weak accountability in manufacturer's organization



“Therac-25” by Catalina Márquez, Wikimedia commons, CC BY-SA 4.0

Code of Ethics

ACM's Code of Ethics Software Engineers

1. PUBLIC – Software engineers shall act consistently with the public interest.
2. CLIENT AND EMPLOYER – Software engineers shall act in a manner that is in the best interests of their client and employer.
3. PUBLIC – Software engineers shall act consistently with the public interest.
4. JUDGMENT – Software engineers shall maintain integrity and independence in their professional judgment.
5. MANAGEMENT – Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
6. PROFESSION – Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
7. COLLEAGUES – Software engineers shall be fair to and supportive of their colleagues.
8. SELF – Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

Does ACM's Code of Ethics Change Ethical Decision Making in Software Development?

Andrew McNamara
North Carolina State University
Raleigh, North Carolina, USA
ajmcnama@ncsu.edu

Justin Smith
North Carolina State University
Raleigh, North Carolina, USA
jssmit11@ncsu.edu

tldr: "No"

ABSTRACT

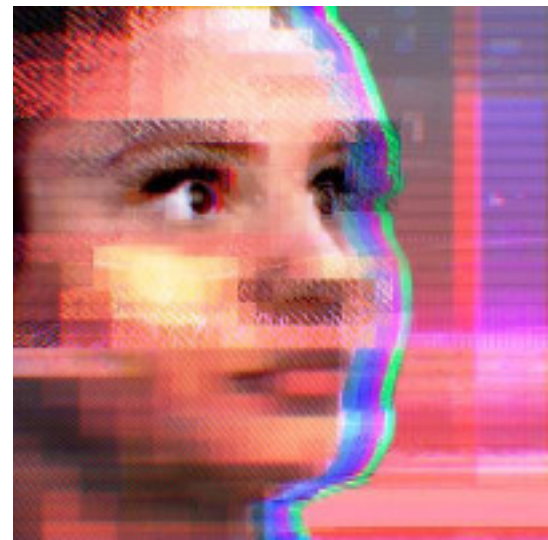
Ethical decisions in software development can substantially impact end-users, organizations, and our environment, as is evidenced by recent ethics scandals in the news. Organizations, like the ACM, publish codes of ethics to guide software-related ethical decisions. In fact, the ACM has recently demonstrated renewed interest in its code of ethics and made updates for the first time since 1992. To better understand how the ACM code of ethics changes software-

The first example is the Uber versus Waymo dispute [26], in which a software engineer at Waymo took self-driving car code to his home. Shortly thereafter, the engineer left Waymo to work for a competing company with a self-driving car business, Uber. When Waymo realized that their own code had been taken by their former employee, Waymo sued Uber. Even though the code was not apparently used for Uber's competitive advantage, the two companies settled the lawsuit for \$245 million dollars.

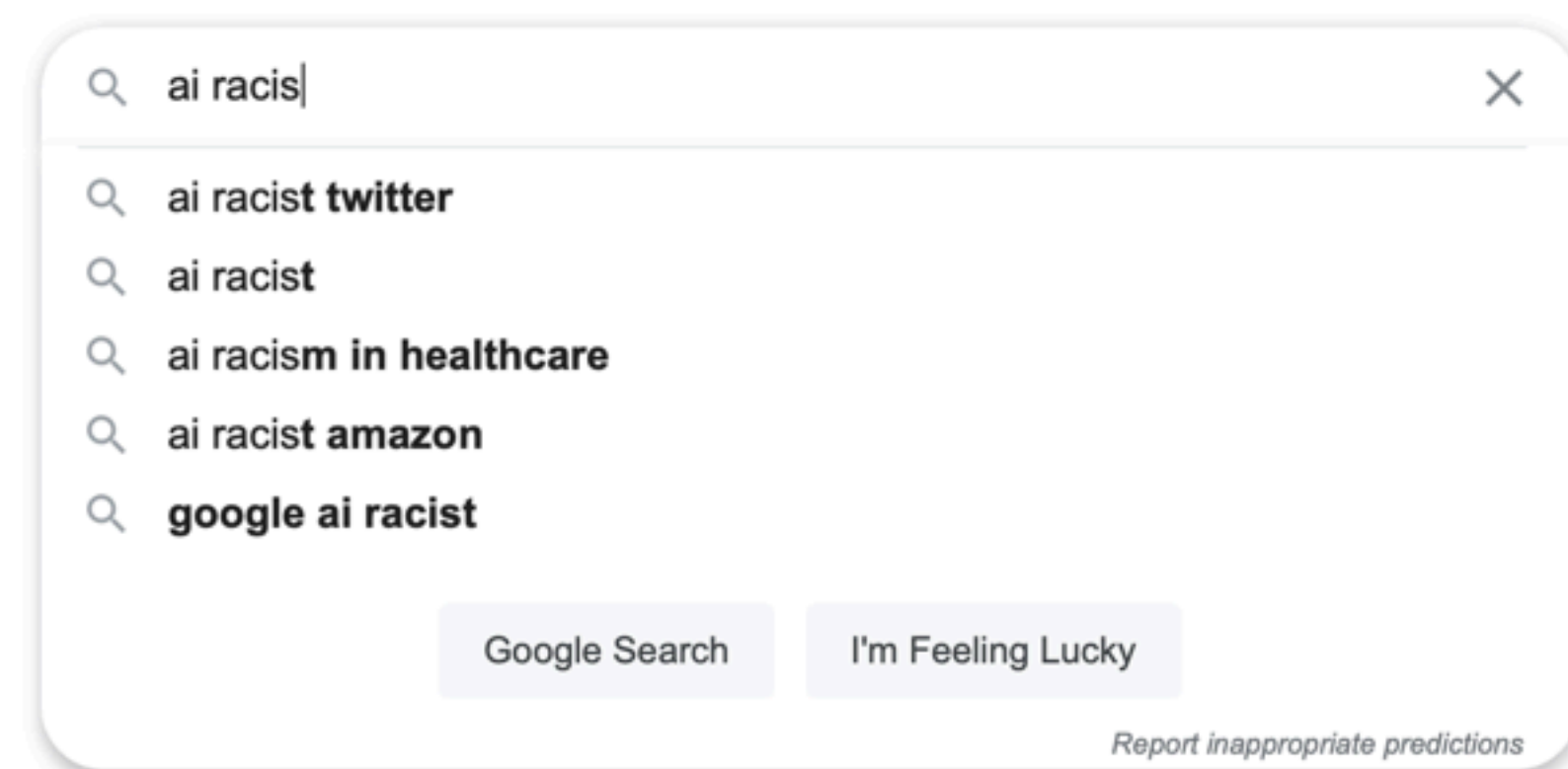
Ethical decisions are often hard decisions

Example: Text Recommendation

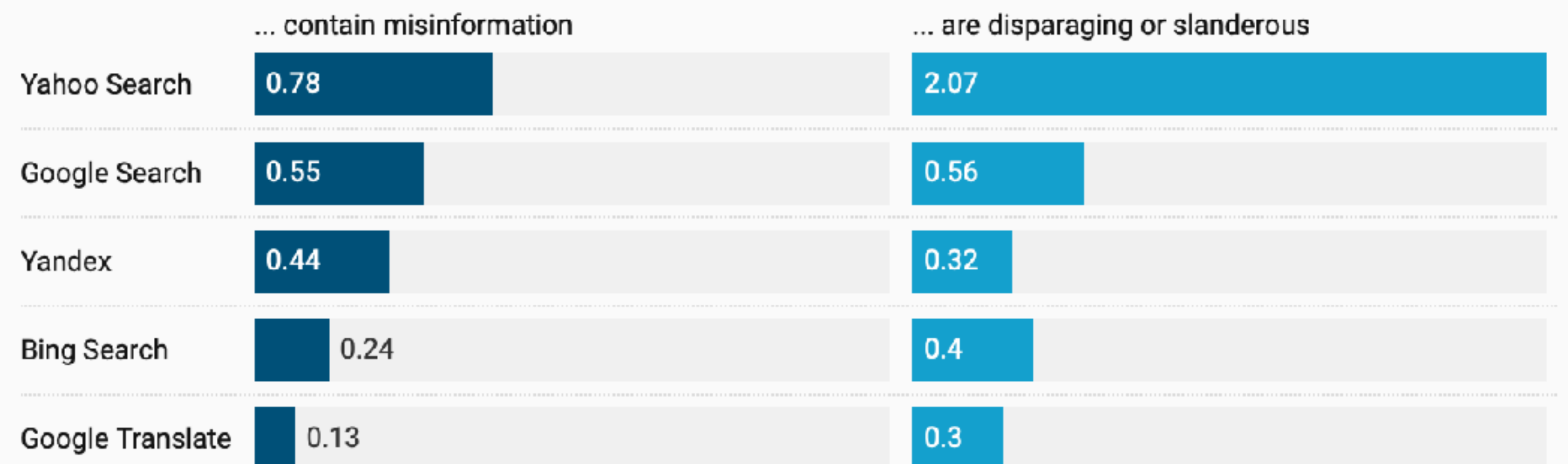
- Interactive AI systems can be useful
- Interactive AI systems can also amplify hate
- Ethical and moral frameworks provide a lens to make rational decisions



Microsoft's "Tay"



For a query on a given service, average number of suggestions that...



Based on 1,117 queries made between May 25 and May 27, 2020.

Source: [AlgorithmWatch](#) • [Get the data](#) • Created with [Datawrapper](#)

"Ten years on, search auto-complete still suggests slander and disinformation"

Accessibility should not be a hard decision

Example: Domino's + ADA

Domino's Would Rather Go to the Supreme Court Than Make Its Website Accessible to the Blind

Rather than developing technology to support users with disabilities, the pizza chain is taking its fight to the top

by Brenna Houck | @EaterDetroit | Jul 25, 2019, 6:00pm EDT

f t SHARE

A screenshot of a web browser displaying the Table of Contents for the WCAG 2.0 Specification. The browser's address bar shows "w3.org". A vertical blue bar on the left side of the page contains the text "WC3 Recommendation". The Table of Contents is organized as follows:

- Table of Contents
 - Introduction
 - [WCAG 2.0 Layers of Guidance](#)
 - [WCAG 2.0 Supporting Documents](#)
 - [Important Terms in WCAG 2.0](#)
 - WCAG 2.0 Guidelines
 - 1 Perceivable
 - 1.1 [Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.](#)
 - 1.2 [Provide alternatives for time-based media.](#)
 - 1.3 [Create content that can be presented in different ways \(for example simpler layout\) without losing information or structure.](#)
 - 1.4 [Make it easier for users to see and hear content including separating foreground from background.](#)
 - 2 Operable
 - 2.1 [Make all functionality available from a keyboard.](#)
 - 2.2 [Provide users enough time to read and use content.](#)
 - 2.3 [Do not design content in a way that is known to cause seizures.](#)
 - 2.4 [Provide ways to help users navigate, find content, and determine where they are.](#)
 - 3 Understandable
 - 3.1 [Make text content readable and understandable.](#)
 - 3.2 [Make Web pages appear and operate in predictable ways.](#)
 - 3.3 [Help users avoid and correct mistakes.](#)
 - 4 Robust
 - 4.1 [Maximize compatibility with current and future user agents, including assistive technologies.](#)
 - Conformance
 - [Conformance Requirements](#)
 - [Conformance Claims \(Optional\)](#)
 - [Statement of Partial Conformance - Third Party Content](#)
 - [Statement of Partial Conformance - Language](#)
 - Appendices
 - Appendix A: [Glossary](#) (Normative)
 - Appendix B: [Acknowledgments](#)
 - Appendix C: [References](#)

Safety-Critical Software Regulations

Standards for Software Development

- International bodies define standard processes that are designed to protect the public
- By (correctly) following such a standard, you can reduce the chance of harm to users, as well as your ethical (and legal) liability

INTERNATIONAL
STANDARD

IEC
62304

First edition
2006-05

Medical device software –
Software life cycle processes

This English-language version is derived from the original bilingual publication by leaving out all French-language pages. Missing page numbers correspond to the French-language pages.



Reference number
IEC 62304:2006(E)

Value Sensitive Design

Consider how our software impacts users' universal values [partial list]

- Human rights - Inalienable, fundamental rights to which all people are entitled
- Accessibility - Making all people successful users of the technology
- Justice - Procedural justice (process is fair) + distributive justice (outcomes are fair)
- Privacy - An individual's agency in determining what information about them is shared
- Human welfare - Physical, material and psychological well-being



Unpacking “Public Interest”

Do no harm: how can our software cause harm?

- How can my software fail? What are the implications of that failure?
- Who will use my software, and how might different users use it differently?
- How will my software impact those who do not use it directly?
- Will my software amplify negative behavior for users and society at large?

Unpacking “Public Interest”

How can our software make a positive contribution?

- Will my software make people’s jobs easier?
- Will my software make people happier?
- Will my software amplify positive behavior for users and society at large?

Where does this leave us?

So that we can sleep at night

- Above all: try to do good - consider the impacts of your software on others
- Follow best-practices, and actively push to improve them
- Next lesson: How do we understand our users, so that we can make the best software, and cause the least unintended harm?

This work is licensed under a Creative Commons Attribution-ShareAlike license

- This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/>
- You are free to:
 - Share — copy and redistribute the material in any medium or format
 - Adapt — remix, transform, and build upon the material
 - for any purpose, even commercially.
- Under the following terms:
 - Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
 - ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
 - No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.