Week 01

Law & Ethics

Web Data Science

Brian Keegan, Ph.D.
Assistant Professor, Information Science
brian.keegan@colorado.edu

University of Colorado
Boulder

Tuesday

Agenda

- 00:00 – 00:20 → Course Overview
- 00:20 – 00:40 → Documentation and Professionalization
- 00:40 – 01:00 → Loading Data
- 01:00 – 01:15 → Final Project, Next Class

Introductions

- Name and pronouns
- Program and year
- Goals for the class this semester
- Favorite web data source
- Greatest achievement over summer

My background

- Grew up outside of Las Vegas, Nevada
- Undergraduate degrees in Mechanical Engineering and Science, Technology, Society @ MIT
- Bartender and oral historian for a year
- Graduate school in Media, Technology, Society @ Northwestern School of Communication
- Post-doctoral research in computational social science @ Northeastern University
- Senior Research Associate and Data Scientist @ Harvard Business School
- CU Boulder Information Science, 2016 – present
- High-tempo online collaborations, public interest data science, demography and extremism
  - https://www.brianckeegan.com/
- Cannabis policy, local politics, astronomy, composting & gardening, rowing

Course Overview

Motivation

- My graduate students and I rely on web data for much of our research
  - How does Wikipedia cover breaking news events? How does it remember recent historic events?
  - How does collective behavior on Twitter change during sporting events?
  - How can bots help govern or disrupt online communities?
  - How can gig workers improve their working conditions?
  - How effective are moderation strategies for limiting polarization and extremist recruiting?
  - How does extremist content about calls for violence move across platforms?
  - How are people migrating from platforms like Twitter to alternatives like Threads, Mastodon, and Bluesky?
- Access to high-resolution behavioral data published on the web has been foundational to my research disciplines of computational social science, network science, and information science

But researchers' ability to access to web data is rapidly disappearing

Course Design

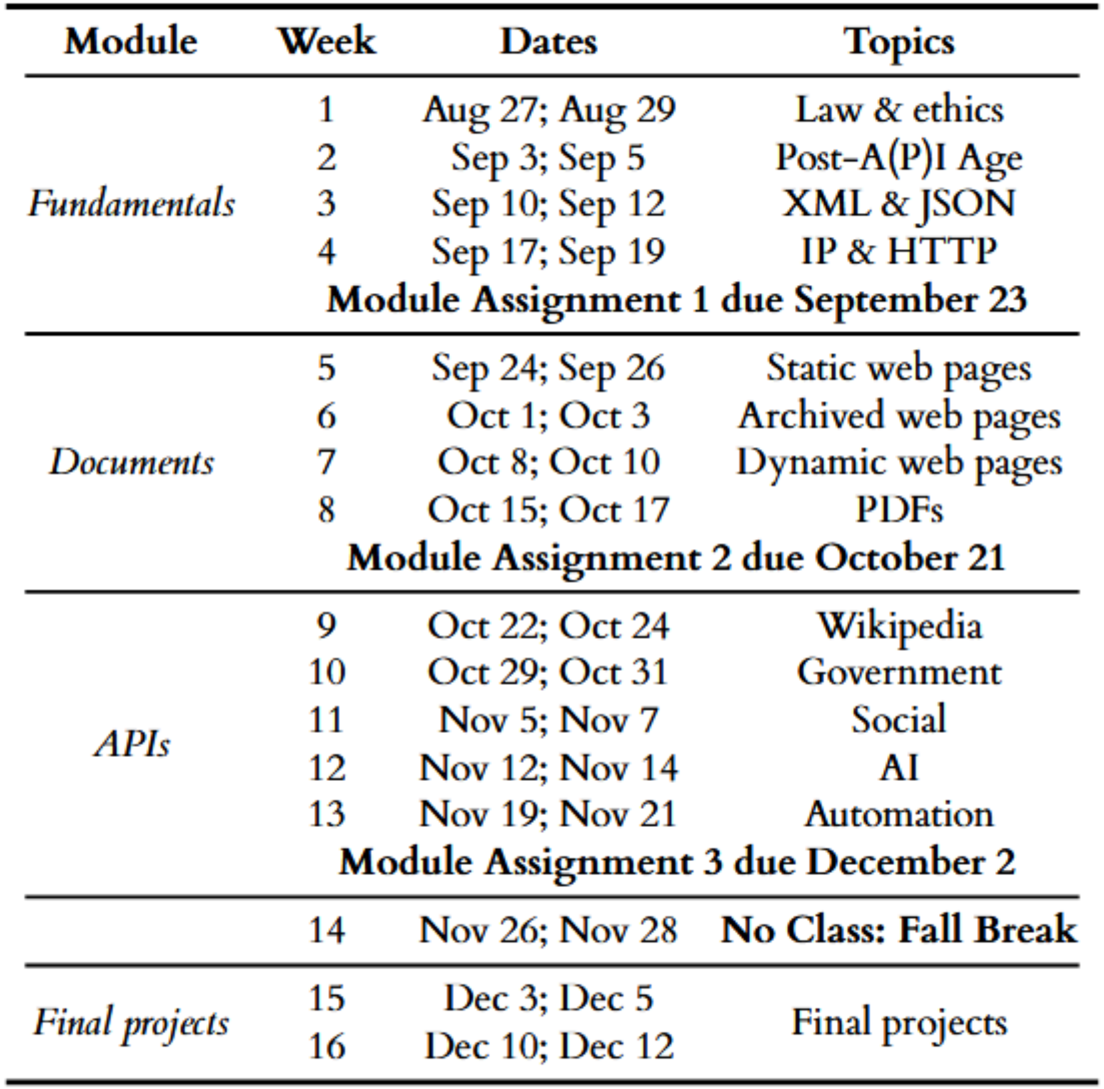
- Lectures are Tuesdays and Thursdays, 9:30 am – 10:45 am
  - Eaton 135
  - **Tuesdays**: Lecture, notebook, exercises
  - **Thursdays**: Review exercises, guest lectures, reading discussion
- Canvas is King: announcements on Canvas override syllabus

Evaluation

- **Attendance**: 15% total
  - Attendance is required. Sign-ups or random cold-calls will be used.
- **Module Assignments**: 3 modules x 20%/module = 60% total
  - Exploratory data analyses and/or research designs
- **Final Project**: 25% total
  - Final paper combining research design, data collection, and exploratory analysis
- No midterm or final exam

Please submit all assignments on Canvas, emailed assignments will be ignored.

Course Overview



Module	Week	Dates	Topics
Fundamentals	1	Aug 27; Aug 29	Law & ethics
	2	Sep 3; Sep 5	Post-A(P)I Age
	3	Sep 10; Sep 12	XML & JSON
	4	Sep 17; Sep 19	IP & HTTP
	Module Assignment 1 due September 23		
Documents	5	Sep 24; Sep 26	Static web pages
	6	Oct 1; Oct 3	Archived web pages
	7	Oct 8; Oct 10	Dynamic web pages
	8	Oct 15; Oct 17	PDFs
	Module Assignment 2 due October 21		
APIs	9	Oct 22; Oct 24	Wikipedia
	10	Oct 29; Oct 31	Government
	11	Nov 5; Nov 7	Social
	12	Nov 12; Nov 14	AI
	13	Nov 19; Nov 21	Automation
	Module Assignment 3 due December 2		
	14	Nov 26; Nov 28	**No Class: Fall Break**
Final projects	15	Dec 3; Dec 5	Final projects
	16	Dec 10; Dec 12	

COVID-19 Contingencies

- If you require sequestration, treatment, convalescence:
  - I will try to accommodate through extensions and incompletes
- If a roommate, partner, or family member's diagnosis will affect you:
  - I will also try to accommodate through extensions and incompletes

- We are bound by and will enforce campus COVID-19 policy

Do not ghost me! Notify us *as soon as possible* of diagnoses or events that will impact your ability to participate in class so we can accommodate without end-of-term scrambling.

Computing Requirements

- Code will be delivered with Jupyter Notebooks of Python 3.10
  - Download Anaconda Individual Edition if you haven't already
  - Update your Anaconda installation. From a Terminal (MacOS) or Anaconda Prompt (Windows)

    ```
    conda update conda
    conda install anaconda=2024.06
    ```
- Readings, code, data will be posted to Canvas: https://canvas.colorado.edu/courses/109074
- We will be using pandas, matplotlib, and seaborn but you can explore others if you like
  - Plotly, Altair, Bokeh, etc.
- Students **are not** permitted to use spreadsheet or business intelligence software
  - Excel, Numbers, Tableau, PowerBI

If you cannot reliably access Anaconda and Canvas or don't have a laptop:
please contact the instructors immediate to work out an accommodation

Documentation and Professionalization

Data Science Mindset Components

Growth Mindset

Computational Thinking

Hacker Ethic

Scientific Norms

- Growth Mindset: effort not ability, continual improvement, challenge-seeking, resilience
- Computational Thinking: concepts, practices, perspectives of applying computing technologies
- Hacker Ethic: sharing, openness, creativity, autonomy, curiosity, bias towards action
- Scientific Norms: communalism, skepticism, responsibility, communication, collaboration

Using Documentation

- Your previous classes may have discouraged using online resources → training wheels are off now!

- Finding, reading, interpreting, and writing documentation are essential skills
  - “Documentation is for ‘real’ developers, not newbies like me” → **WRONG!**
  - Bookmark documentation for numpy, scipy, pandas, matplotlib, and seaborn

- “It’s not working” is not an acceptable request for help
  - What have you tried? What can you get to work? What does the documentation say?

Escalating Issues

- **Examples, tutorials, user guides** → *super* helpful general resources, accessible writing style
  - Common use cases and combining multiple functions, check these first to get oriented
- **Reference or API reference** → details about specific functions & methods, more technical
  - If the tutorials or user guides don't address your issue, look to the specifics for your function, method, *etc*.
- **Question-Answer website** → ask (and answer!) questions on StackOverflow
  - Copy-and-paste error messages, search for a generalized version, *etc*.
- **Developers or Development** → details about contributing code, reporting bugs, early releases
  - Maybe you've encountered a genuine bug or boundary case → see if others have had the same problem
  - Most libraries are developed on GitHub, go to the library's repo, and search under "Issues"

Credit External Resources

- If you use an external resource (documentation, Q&A, blog post, repository/gist) to implement a feature or solve a bug beyond what we've covered in class, just include a link in your code!

```
f,ax = plt.subplots(1,1)

# https://matplotlib.org/stable/tutorials/text/text_intro.html
ax.text(2,6,r'an equation: $E=mc^2$', fontsize=15)
```

- Using advanced functionality that we haven't covered in class and you're not citing is a reliable signal that you're using external resources without credit → just include a link in your code!

- We reserve the right to request a code review for any submitted assignment. If you're unable or unwilling to explain how something was implemented, you could lose all credit on the assignment
  - Repeated violations will be escalated to the Honor Code office
  - Just include a link in your code!

Documenting Your Own Code

- The class notebooks make extensive use of Markdown cells to organize and narrate the analysis

- You should develop a similar practice and style of making clear and well-organized notebooks
  - Importing all libraries in one place, loading data in one place, cleaning data after loading, *etc*.
  - Sectioning (and sub-sectioning!) so different steps are easy-to-find
  - Markdown with narrative of what’s happening, hyperlinks to resources/documentation
  - Use hanging indentation to help with legibility instead of cramming into single lines
  - Check out style guides like the Space Telescope Science Institute

Notebook Time!

- Download the “Week 01.ipynb”
    - Please create a dedicated folder for class instead of keeping everything in Downloads, Desktop, *etc*.
    - Put both these files in the same folder
- Open the “Week 01.ipynb” notebook file
    - From Anaconda Prompt (Windows) or Terminal (Mac), navigate to class folder
    - Launch Jupyter Notebook: `jupyter notebook`
- Make sure the first few cells work
- No grading on exercises, we’ll cover solutions on Thursday

Final project

- Creating an accessible dataset of all laws before the Colorado Legislature each session
    - https://leg.colorado.gov/bills-by-bill-number
    - Bill summaries, sponsors, committees, history, votes
    - What could we do with more accessible data?
- Portions of this data is available in (regrettable) data formats
    - https://leg.colorado.gov/agencies/house-representatives/session-publications
    - https://leg.colorado.gov/agencies/senate/session-publications
    - https://leg.colorado.gov/prior-session-information
- Even as data from a public institution, there are terms and policies
    - https://leg.colorado.gov/sites/default/files/privacypolicy.pdf
    - What rules should we be aware of? What disclosures are they making?

We will return to this through (ungraded) Weekly Assignments and Module Assignments.

Next Class

- Do the exercises at the bottom of the Week 01 notebook

- Discuss readings
  - "When the Terms of Service Change to Make Way for A.I. Training"
  - "The Backlash Against AI Scraping is Real and Measurable"
  - "Websites are Blocking the Wrong AI Scrapers"

- Daily note: https://bit.ly/info4871f24note
- Today's number is…

Thursday

Agenda

- 00:00 – 00:05 → Daily Note check-in
- 00:05 – 00:25 → Exploring User-Agents and robots.txt
- 00:25 – 01:00 → Discuss readings and explore Terms of Service and Privacy Policies
- 01:00 – 01:15 → More brainstorming about Final Project

Daily Note

Exploring User-Agents and robots.txt

- Jump back into the notebook
- User-Agents Experiment with other

Take-aways from readings

- “When the Terms of Service Change to Make Way for A.I. Training”
- “The Backlash Against AI Scraping is Real and Measurable”
- “Websites are Blocking the Wrong AI Scrapers”

Terms of Service and Privacy Policies

- Based on your birthday, skim through a platform’s terms of service or privacy policy
  - **Even month, even day, even year**: Instagram terms of service
  - **Even month, even day, odd year**: Instagram privacy policy
  - **Even month, odd day, even year**: TikTok terms of service
  - **Even month, odd day, odd year**: TikTok privacy policy
  - **Odd month, even day, even year**: Twitter/X terms of service
  - **Odd month, even day, odd year**: Twitter/X privacy policy
  - **Odd month, odd day, even year**: Reddit terms of service
  - **Odd month, odd day, odd year**: Reddit privacy policy
- January 1, 2003 → odd months, odd day, odd year → Reddit privacy policy
- Look for keywords related to “scrape”, “API”, “research”, “data”, “abuse”

Notes

- What’s interesting? Unusual? Hard-to-understand?
- Is a college class allowed to retrieve data for research?
- How does this compare to another website’s or platform’s terms and policies?

Next class

- Daily note: https://bit.ly/info4871f24note
- Today's number is…