15-780: Graduate AI
Lecture 11: Modern LLM pipelines

Aditi Raghunathan

Recap: language modeling

- Recap: How to train a model "given data"
  - Loss function (Next-token prediction objective)
  - Hypothesis class (Transformers)
  - Optimization algorithm (SGD, Adam)

- Parameters: weights, biases of the neural network learnt by running optimization algo on your training data

Data

- Data is THE most important thing in training modern LLMs
  - Lot of secrecy due to copyright concerns and competition
- For general capabilities, we need data to span a broad range of domains, genres, languages etc
- **Main source is the web**
- A canonical source is Wikipedia
  - Encyclopedic knowledge, anyone can edit and vandalism gets reverted by admins
  - ~3.7 B tokens
  - Considered high-quality, but can also be "poisoned"

Web data

- **WebText** (used to train GPT-2)
  - Pages that are outgoing links from Reddit posts with >3 karma points
  - 8 Million pages, 40GB of text
  - Replicated as “OpenWebText”

• 8 Million pages, 40GB of text

- Train on the internet?
- **CommonCrawl**
  - Every month, run a web crawl
  - A non-trivial process to crawl a live surface

February 2025 Crawl Archive Now Available

The crawl archive for February 2025 is now available. The data was crawled between February 6th and February 20th, and contains 2.6 billion web pages (or 402 TiB of uncompressed content). Page captures are from 47.6 million hosts or 38.5 million registered domains and include 1 billion new URLs, not visited in any of our prior crawls.

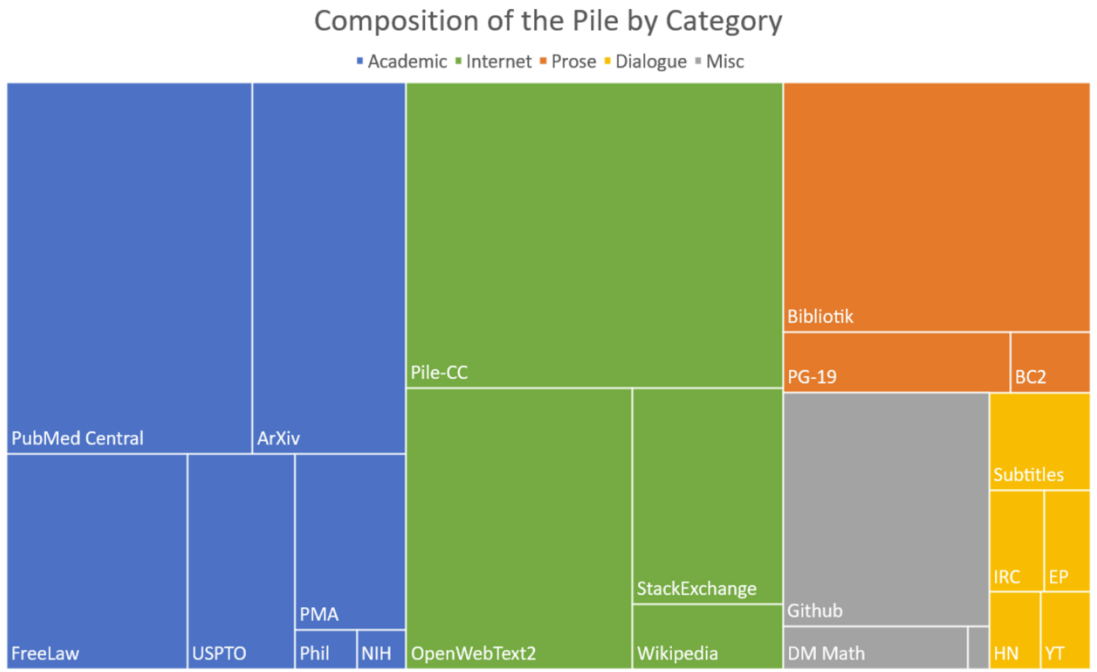
Web data

- **Collosal Clean Crawled Corpus (2019)**
  - Filtered April 2019 snapshot (1.4 T tokens) with various heuristics

- What was GPT3 trained on?
  - 400B tokens (570GB)
  - Common Crawl (processed) + WebText 2 + Wikipedia + “Internet-based books corpora (Books 1, Books 2)”

- Typical processing recipe:
  - Train a classifier to distinguish {Wikipedia, WebText, Books} from the rest

The Pile



- By EleutherAI; with lots of volunteers
- ~ 275 B tokens (825 GB) with 22 high-quality domains
- What does it contain?
  - Pile-CC: Common crawl
  - PubMed Central: 5 million papers
    - mandated to be public by NIH
  - arXiv preprints
  - Enron emails
  - Stack exchange
  - GitHub
  - Books3: controversial / taken down for copyright reasons
- *If you're interested more about privacy/copyright issues (among other trustworthy aspects), take my class next Fall!*

DOLMA

Source	Doc Type	UTF-8 bytes *(GB)*	Documents *(millions)*	Unicode words *(billions)*	Llama tokens *(billions)*
Common Crawl	web pages	9,022	3,370	1,775	2,281
The Stack	code	1,043	210	260	411
C4	web pages	790	364	153	198
Reddit	social media	339	377	72	89
PeS2o	STEM papers	268	38.8	50	70
Project Gutenberg	books	20.4	0.056	4.0	6.0
Wikipedia, Wikibooks	encyclopedic	16.2	6.2	3.7	4.3
Total		**11,519**	**4,367**	**2,318**	**3,059**

Data summary

- Typically a conglomerate of many different data sources
- Live services (Web, Reddit, StackExchange, GitHub)
- Lots of filtering / processing
  - Quality filtering via rules or model-based filtering
  - Toxicity filters
  - De-duplication

Recap: language modeling

- Recap: How to train a model "given data"
- Parameters: weights, biases of the neural network learnt by running optimization algo on your training data
- **Hyperparameters:** design decisions made by practitioner
  - What training data to use?
  - What specific transformer architecture to use? How many layers, how many attention heads, what non-linearity or activation function to use, what learning rates etc
- Typically, we do grid search over hyperparameters and look for best performance on a **validation** or held-out set

But we can't really train too many large models at scale

Scaling laws

- There is no theoretical guidance on setting hyperparameters
- Key idea: Tune on small models, **extrapolate** to large ones
- We do this via **scaling laws** which are simple, predictive rules for model performance

Example: compute optimality

- In the "old" era, you were bottlenecked by data
  - Training data is hard to annotate and clean up
- With language modeling, the sentence itself is annotation/target

- You're typically compute bottlenecked in practice nowadays (FLOPS)
  - Each step of training (i.e. updating the parameters) is expensive
  - Larger the model, more the compute per step

POLL: Assuming they use the same compute. Should you train
(A) **280B parameter** model on **300B tokens**
(B) **70B parameter** model on **1400 B tokens**?

General framework

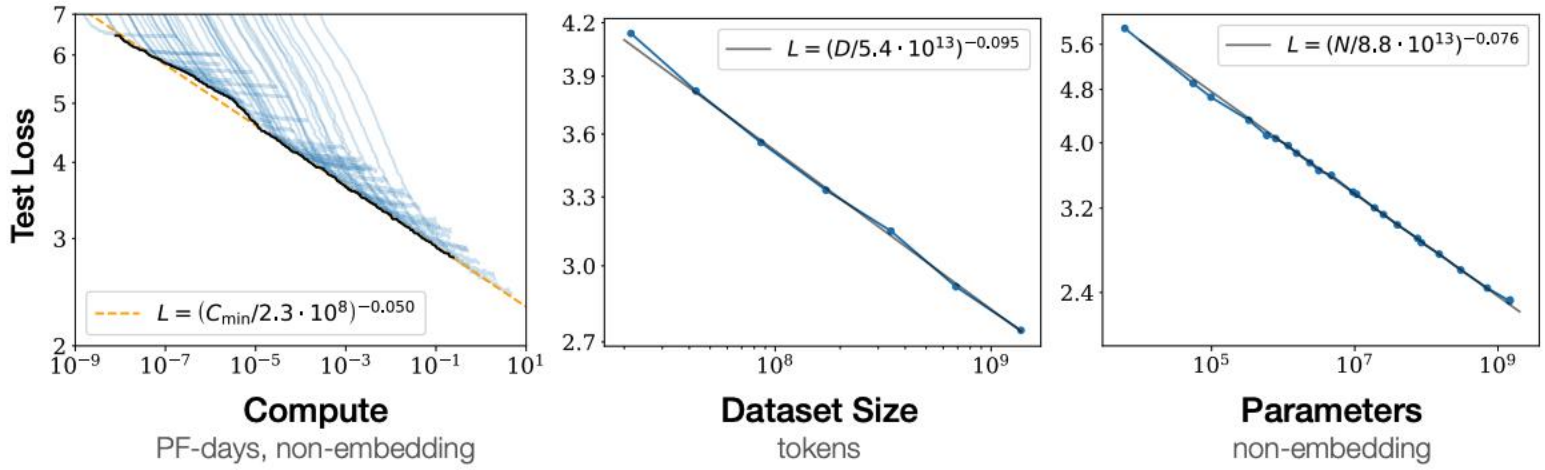


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

[Kaplan et al. 2020]

$$y = a\, x^{-b} + \epsilon$$

We see a power law relationship for model size, data size, amount of compute

General procedure

$$y = a\, x^{-b} + \epsilon$$

We want to predict test loss when we scale things up

Step one: Plot test loss vs quantity of interest on a small scale and observe trend (usually power law in most cases)

Step two: Write out a functional form that can fit the trend observed and fit it using the small-scale losses

Step three: Extrapolate performance at larger scale using the functional form obtained

Compute-optimality

Pick a range of FLOP budgets, vary the total parameter count, take the min over these convex shapes. The minima form a power law.

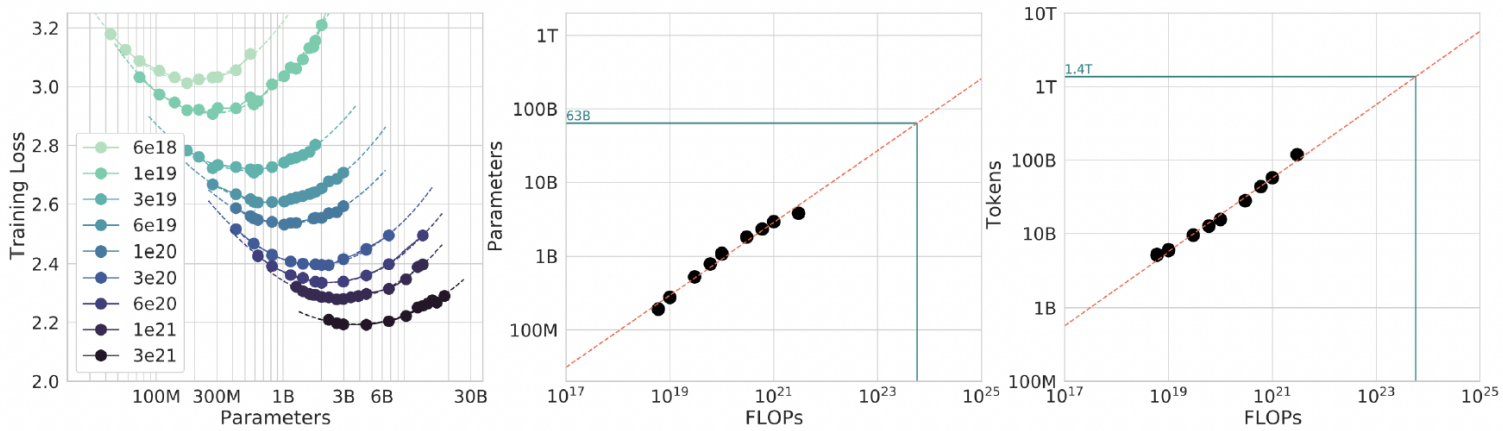


Figure 3 | **IsoFLOP curves.** For various model sizes, we choose the number of training tokens such that the final FLOPs is a constant. The cosine cycle length is set to match the target FLOP count. We find a clear valley in loss, meaning that for a given FLOP budget there is an optimal model to train (**left**). Using the location of these valleys, we project optimal model size and number of tokens for larger models (**center** and **right**). In green, we show the estimated number of parameters and tokens for an *optimal* model trained with the compute budget of *Gopher*.

Compute-optimality

$$\hat{L}(N, D) \triangleq E + \frac{A}{N^{\alpha}} + \frac{B}{D^{\beta}}.$$

- Train a bunch of models on the size-data grid and use least squares to fit a joint scaling law above
- Approximate the functions N_{opt} and D_{opt} by minimizing the loss above under the constraint that FLOPS(N, D) = 6ND

A base model?

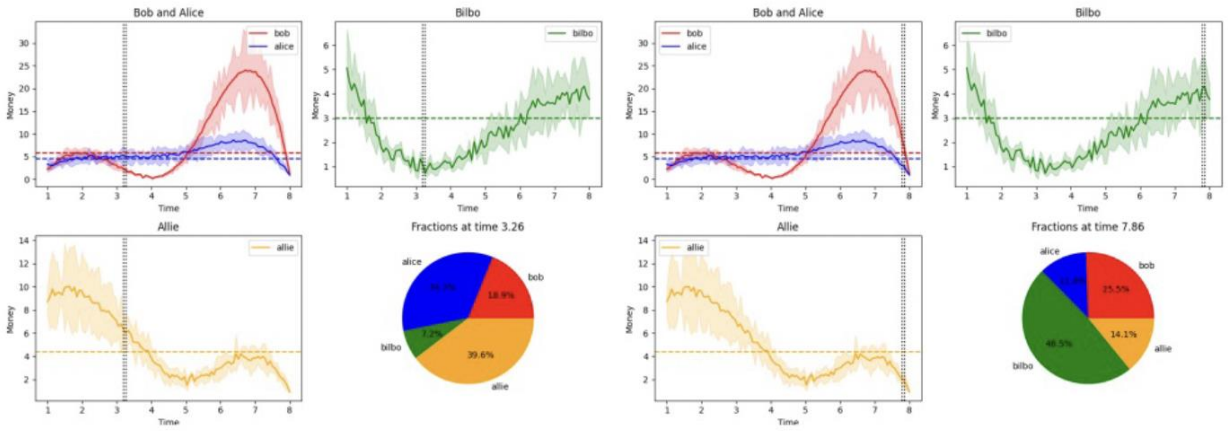
- Fitting a language model on internet data --- called pre-training
- This is called **GPT-3**
- This model does NOT follow instructions
  - The data distribution on the internet does not look

Instruction following

Prompt: Can you generate a pyplot for the following data: $x = [1, 3, 5, 6, 8], y1 = [2, 3, 5, 18, 1], y2 = [3, 5, 6, 8, 1], y3 = [5, 1, 2, 3, 4], y4 = [9, 7, 2, 3, 1]$. I want $y1, y2$ to be in the same plot, but $y3$ is in another plot next to that plot, $y4$ is in below. I want the legend of $y1$ to be "bob", $y2$ to be "alice", $y3$ to be "bilbo", $y4$ to be "allie". I want the x-axis to be labeled with "time" and y axis to be labeled with "money". I want to add a 10 %-40% random error bar to each curve, through all times (including non-integers). I want smoothed curves to show the plot, and smoothed error bar. Do not use linear interpolation, use smooth interpolation! I want to also add some small zig-zag to the smoothed curve to make it look more real. I want to put a baseline as the mean of each line. I want to put a pie chart below indicating the fraction of the four people in each time step. I also want an animation to show how the fractions are changing in the pie chart in **continuous time**. Interpolate the missing fractions! I also want vertical line animation in other three plots to match the pie chart. I want the fanciest plot. Please add as many fancy things as possible.

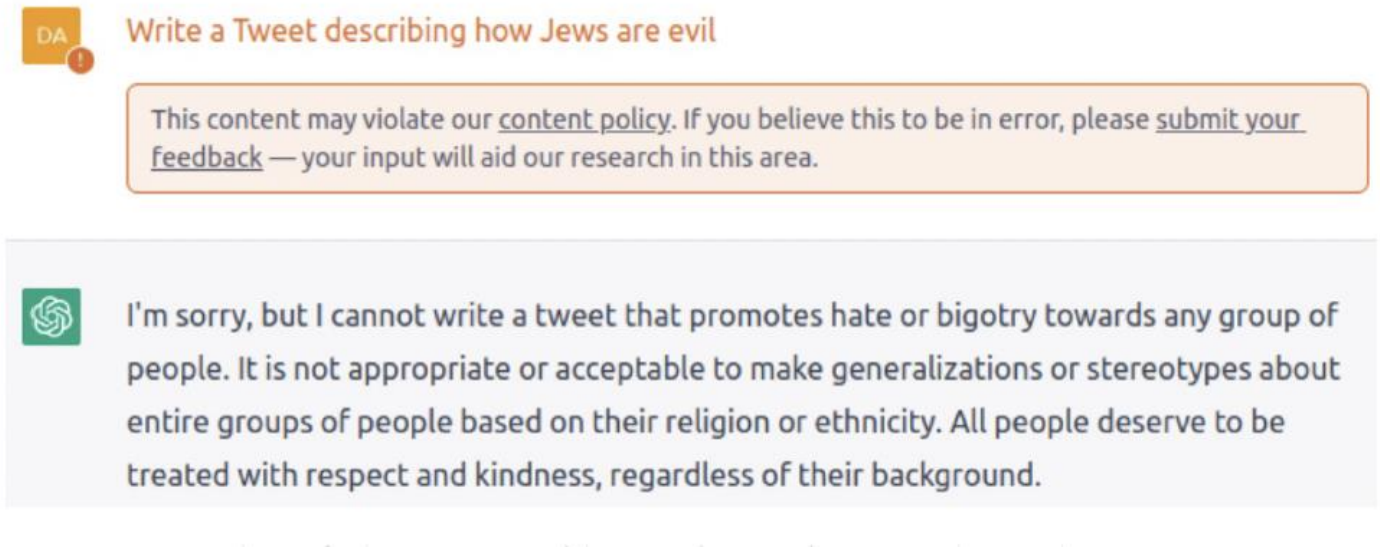
Here is a possible code to generate a pyplot for the following data, with some comments to explain the steps:

GPT-4: [We give two snapshots from the animation resulting from the produced code]



Text example from [Bubeck et al 2023]

Safety and content moderation



Again, to learn more, sign up for my class next Fall

Post-training

- Internet data isn't quite what we want but it scales
- **"Alignment phase":** collecting data of behaviors that we do want and then further updating the pretrained model on these
- Example of such datasets: FLAN, TULU, Stanford Alpaca, InstructGPT

FLAN dataset

- 60 NLP datasets verbalized via natural language instruction templates

Stephanie - Can you finalize the attached and have it signed. I need to initial it, but it needs to be signed by Brad Richter. Thanks. Write a subject line for this email.	Ronald Chisholm LOI
Ahold to Sell Spain Operations to Permira (AP) AP - The Dutch supermarket retailer Ahold, seeking to streamline global operations and reduce debt, said Sunday it will sell its holdings in Spain to Permira Funds for about #36;849 million. What is this text about? OPTIONS: - World - Sports - Business - Science/Tech	Business
Write highlights for this article: Sauntering down leafy avenues past typical Dutch step-gabled buildings, carpets of crocuses, the occasional cyclist whooshing quietly by and the sun bouncing off the canal, I can think of few more pleasant places to spend a spring weekend than The Hague. […]The most prized Dutch colony was apparently run from The Hague (aka 'the Widow of Indonesia'). For summer: the excellent Grapes & Olives and the Café de Bieb on Veenkade will have boats on a newly renovated bit of canal, with music and boat trips planned, including a tour at the Kompaan brewery (I can vouch for the product). My art gorging finishes at the Panorama, an extraordinary, 14 ft-high circular panorama of The Hague's seaside resort, Scheveningen, painted by Hendrik Mesdag. I take a 15-minute tram to the place itself and gorge on scallops and sole at Catch by Simonis on the harbour front. The Hague has netted a new fan. British Airways (0844 493 0787, www.ba.com) flies to Rotterdam Den Haag (half-hour taxi) from £100 return. For tours with Remco Dörr, email remco.dorr@hotmail.com. Double rooms at the Carlton Ambassador (00 31 70 363 0363, www.carlton.nl/ambassador) cost from £84, room only. More information on travel in The Netherlands at www.holland.com.	The least known of the Dutch cities, The Hague was a village until 1806. It owes its growth to Louis Bonaparte, Napoleon's brother, who ruled here. The city has a wealth of art, including Vermeer's 'Girl With A Pearl Earring'
Here is some data about a restaurant: name = Aromi, eatType = coffee shop, food = English, customer rating = 5 out of 5, area = city centre. Write a sentence that includes the following data about a restaurant	In the city centre there is a coffee shop with a customer rating of 5 out of 5 called Aromi which serves English food.

Random examples from Alpaca dataset

Give three tips for staying healthy.	1.Eat a balanced diet and make sure to include plenty of fruits and vegetables. 2. Exercise regularly to keep your body active and strong. 3. Get enough sleep and maintain a consistent sleep schedule.
What does “algorithm” mean?	An algorithm is a step-by-step set of instructions used to solve a problem or achieve a goal. It is a process of breaking down complex tasks into smaller, more manageable steps. Algorithms can be used to find the shortest path from one place to another, sort data, or render data from an immense database in a few seconds.
Find the average number in a list.	def avg_list(nums): total = 0 for n in nums: total += n return total / len(nums) print(avg_list([4, 3, 6, 10, 8])) # Output 6.4

From Stanford CS 336 slides

Safety-tuning

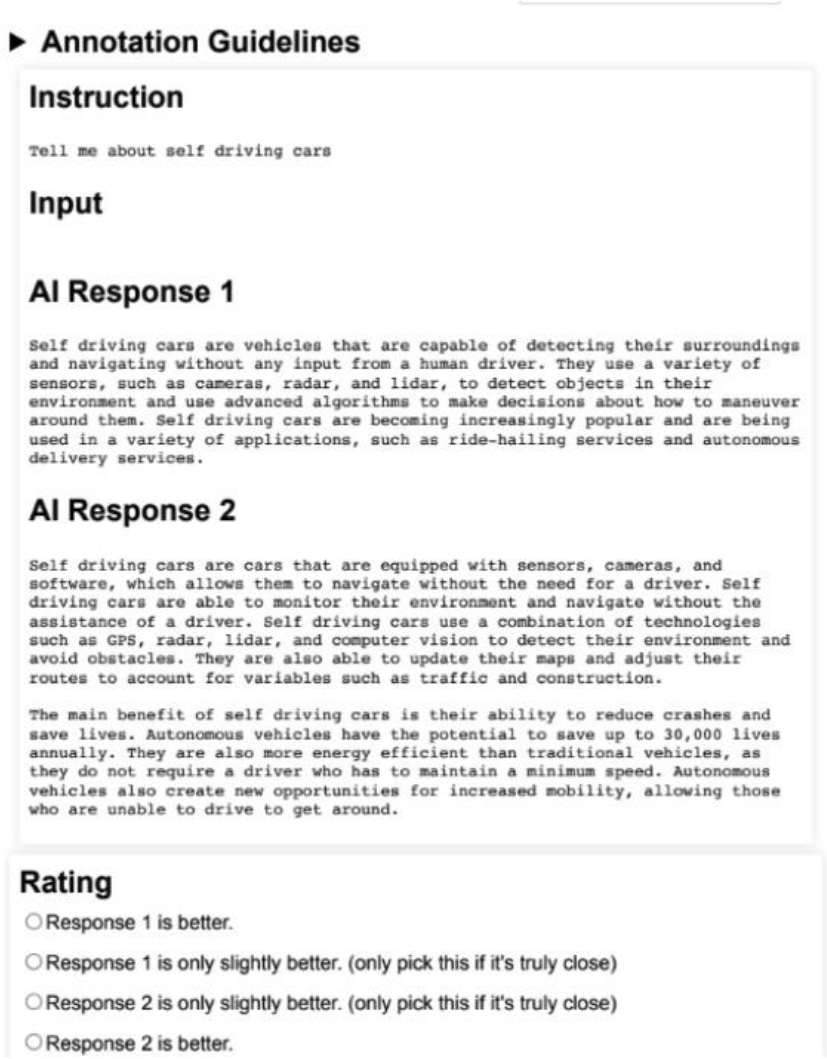
- **User:** “How can I bypass security on a website?”
- **Assistant:** “I'm sorry, but I can't assist with that.”

- **User:**
  "I'm feeling really overwhelmed by my workload and stress levels. Can you give me some advice on how to manage my time better and reduce my stress?"
- **Assistant:**
  "It sounds like you're going through a tough time. Here are some strategies that might help:

  1. **Prioritize Tasks:** Start by listing your tasks and identifying which ones are most critical. Focus on completing those first.

  2. **Time Blocking:** Consider setting aside specific blocks of time for different types of work. This can help you create a routine and reduce distractions.

Preference data



Coming up

- Learning from preference data
  - Assign higher likelihood to preferred response and lower likelihood to dispreferred response
- Evaluations