



Midterm Exam Recap

Large Language Models: Methods and Applications

Daphne Ippolito and Chenyan Xiong

Grade Summary

- Max possible points: 60
- Min grade: 13.0
- Mean grade: 36.19
- Median grade: 37.25
- Maximum grade: 52
- Std Dev: 8.13

Regrade Requests

We will accept regrade requests made on Gradescope up through November 8.

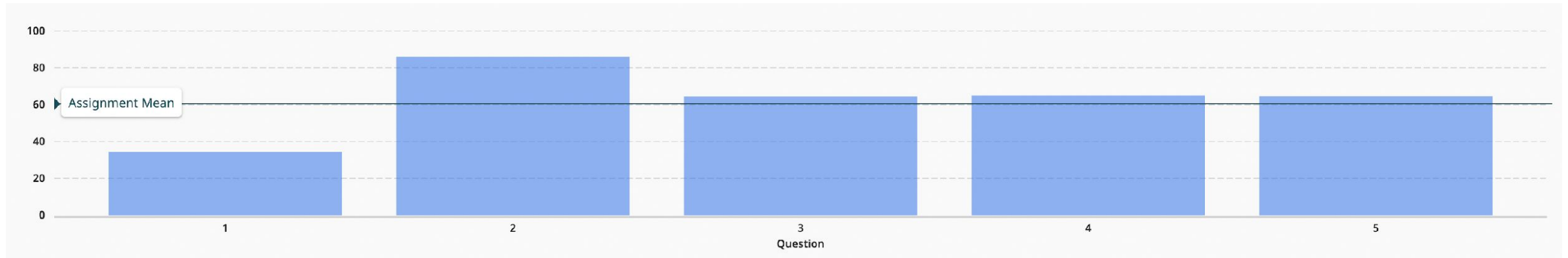
Regrade requests made after this date will be ignored.

Regrade requests should be made ONLY through Gradescope.

If you have questions about whether you should submit a regrade request, please come to office hours. Chenyan will have OH immediately after class today.



Average grade for each question



Question 1(a)

a) [2 pts] Language models contain two embedding layers, one at the beginning for mapping input tokens into vector embeddings, and another at the end for mapping a model's prediction into logits. Why does embedding layer weight tying (using the same weights for both these layers) make more sense for small models (e.g. a 1B parameter model) than for large models (e.g. a 50B parameter model)?

Think about the dimensions of the embedding layer:

vocab size \times embedding dimension

Think about what embedding layer weight tying does:

halves the number of weights devoted to token embeddings

Think about the ways models are made bigger, and how embedding layer size e scales with these.

increased number of layers \rightarrow e stays constant

increased embedding dimension \rightarrow e scales linearly

In contrast, # of **non-embedding params** grow quadratically with hidden dimension

Conclusion: Embeddings make up a larger proportion of the weights in smaller models than larger models, so weight-tying results in a bigger % decrease in # weights for smaller models.



Question 1(d)

d) [2 pts] Modern language models use multi-headed self attention. For a model with an embedding dimension of d , a vocabulary size of v , and the number of attention heads set to h , what are the total number of parameters in a single multi-headed self attention block? You can assume that $d_{head} * h = d$, and you may exclude bias terms.

Hint: there are four linear layers you should consider: W_q, W_k, W_v and an output projection layer W_o .

Recall from the “Attention is all you need” paper:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h) W^O$$
$$\text{where } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$.

The question gives that $d_k = d_v = d_{\text{model}}/h$

For each head i , we have $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}/h}$. Summing over all heads, this gives $3(d_{\text{model}})^2$ for the attention projections. The output projection is another $(d_{\text{model}})^2$.

This gives the final correct answer: $4d^2$

Question 1(e)

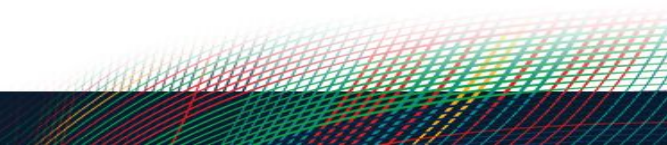
```
# Start implementation A
q = q.view(B, S, H, HD).permute(0, 2, 1, 3)
kT = k.view(B, S, H, HD).permute(0, 2, 3, 1)
v = v.view(B, S, H, HD).permute(0, 2, 1, 3)
# End implementation A

# Start implementation B
q = q.view(B, H, S, HD)
kT = k.view(B, H, HD, S)
v = v.view(B, H, S, HD)
# End implementation B

return q, kT, v
```

Which implementation is correct, A or B? Just write down the letter; you do not need to provide justification.

Which implementation is likely to result in a lower test loss? Explain with at most two sentences.



Question 1(e) - incorrect implementation B

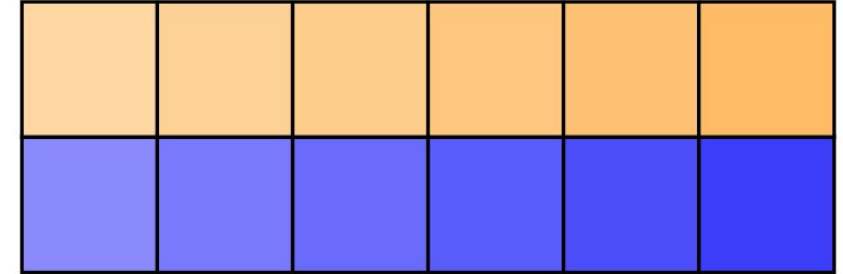
Consider the following example with

- sequence length=2
- attention heads=2
- hidden dim=6.

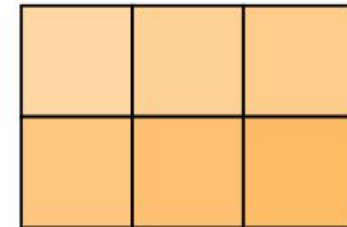
Yellow cells represent (multi-head) Q values for the first token. Blue cells represent Q values for the second token.

An incorrect implementation may look like `q = q_matrix.view((batch_size, n_heads, seq_len, head_dim))`, which gives the two matrices (corresponding to the two attention heads) on the right:

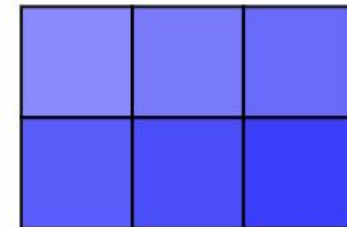
Notice that the first row of head 2 (i.e., Q vector for the first token @ head 2) is blue (i.e., computed from the second token). So down the line, you are basically training the model to use embeddings of the second token to predict the second token itself.



H1



H2



Question 1(e) – correct implementation A

Consider the following example with

- sequence length=2
- attention heads=2
- hidden dim=6.

Yellow cells represent (multi-head) Q values for the first token. Blue cells represent Q values for the second token.

The correct implementation permutes the axes, resulting in the matrices on the right.

H1

H2

Problem 1(f)

This came almost straight out of the homework.

We were not looking for perfect code, just that you remembered the algorithm you used to implement the encode function in the homework.



Problem 3(c)

f) [2 pts] Can you do evaluation on the joke identification benchmark task without doing any generation? If yes, in a sentence, explain the method you would use. If no, in a sentence, explain why not.

Correct answer: Yes. Since this is a classification task, we can predict a label "funny" or "not funny" by comparing $P_{\theta}(\text{"funny"} \mid \text{prompt})$ and $P_{\theta}(\text{"not funny"} \mid \text{prompt})$. This doesn't involve doing any text generation.

Hint for HW: you don't need to do generation on the few-shot learning question on the homework either.



Problem 3(e)

e) [2 pts] Given a test set containing several documents, explain how to calculate the perplexity of this test set according to an LLM. You may either write an equation or explain in precise English.

Correct answer:

1. Get negative log likelihood of each document
2. Sum up these values across all documents
3. Divide by the total number of tokens in all documents
4. Take the $\exp()$ of this value

Equation:

$$PPL = \exp\left(-\frac{1}{N} \sum_{i=1}^{|D|} \log P(x_1^{(i)}, \dots, x_{n_i}^{(i)})\right)$$

Problem 4(c)

c) [2 pts] You would like to build an AI assistant that can converse with users about their private calendars. Describe a strategy you might use to collect the data needed to finetune a language model to make calls to the calendar API.

Correct answer:

Create synthetic data using an existing LLM with coding abilities that can be prompted to generate tool calls. Data will be noisy but can still be effective for finetuning.

Human-labeled data is also possible but would be slower to obtain.



Problem 5(c)

c) [2 pts] In a sentence, describe one *unsupervised* strategy for selecting positive pairs for training dense retrieval systems. By unsupervised, we mean: without the use of explicit human annotations.

Correct answers includes:

- Document cropping (use the two halves of the document as a positive pair)
- Take titles of webpages as queries for the contents of the webpages
- Leveraging anchor text
 - e.g. if on a website “Vegetarian Society of Ireland” links to a page with the text “The Vegetarian Society of Ireland is a registered charity. Our aim is to increase awareness of vegetarianism in relation to health, ...”
- Query generation
 - e.g. ask an LLM to generate queries about a document



Problem 5(e)

Complete the architectures for a Bi-Encoder and a Cross-Encoder to match user queries with news articles. You are given the starting point: the query and the news article. Draw and label additional boxes as needed to represent the components of each architecture. Use boxes for “Model”, “Pooling”, “Cosine-Similarity”, and “Classifier” where appropriate. Connect the components with arrows to illustrate the data flow in each architecture.

Correct answer: Bi-encoder encodes the query and news article separately and uses cosine similarity to determine similarity, whereas the cross encoder encodes them together and uses a classifier.

Common Mistakes:

- Flipping the cross encoder and bi-encoder
- Using both cosine similarity and a classifier
- Forgetting to pool and/or pooling the incorrect thing



Benchmark Contamination (continued)

Large Language Models: Methods and Applications

Daphne Ippolito and Chenyan Xiong

(with slides borrowed from Maarten Sap)

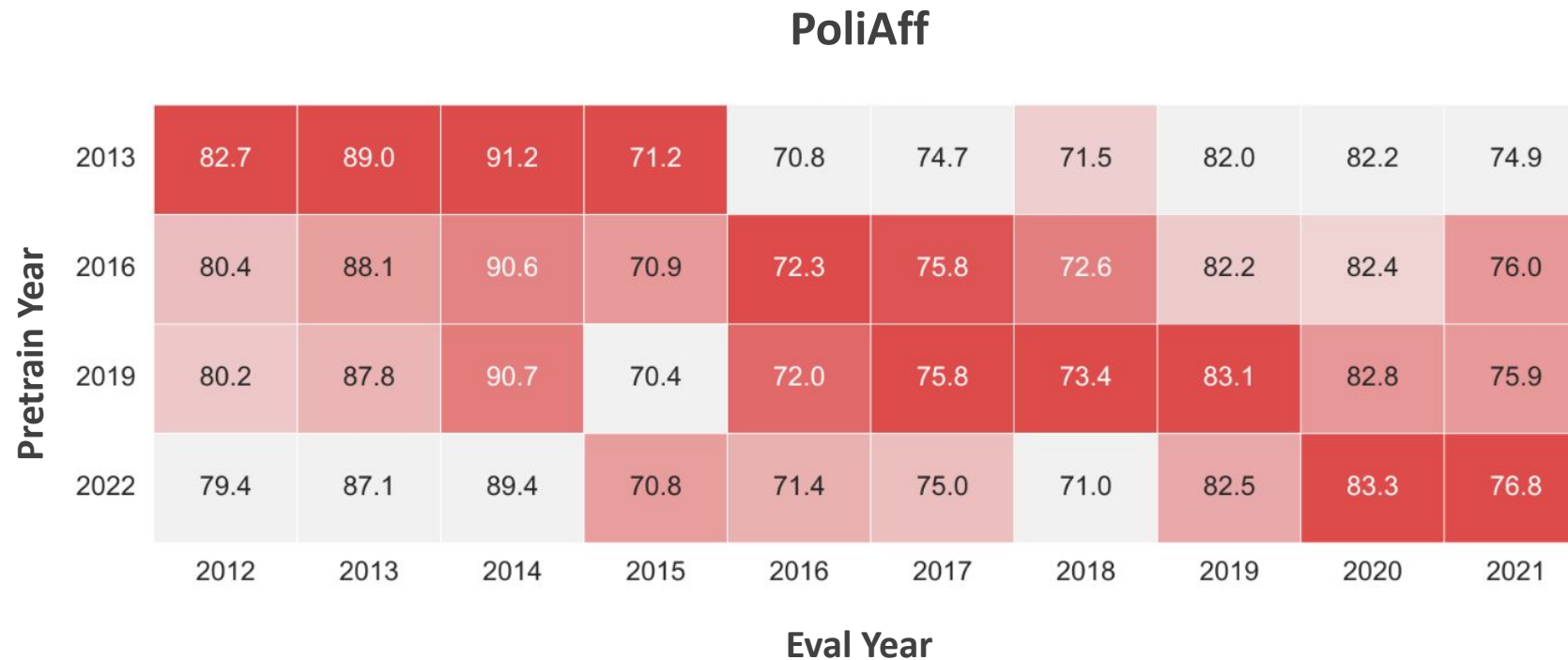
Forms of benchmark data contamination

- **Semantic Level:** Exposure of identical and/or derivative content of the benchmark.
 - Example: consider a summarization benchmark where all the examples come from news articles. Model 1 does better at the benchmark than Model 2 because it was trained on more news articles.
- **Information Level:** Exposure to benchmark-related information leads to models with tendencies and biases during evaluation.
 - Example: a Github page describing the summarization benchmark was put up in 2020. Models trained on a web scrape from 2021 saw the Github page as part of the training, while older models did not.
- **Data Level:** Exposure of the input examples in the benchmark.
 - Example: the summarization benchmark was constructed by asking human annotators to write short summaries for NYTimes articles published in 2019. These NYTimes articles may have been seen verbatim during training.
- **Label Level:** The complete exposure of benchmark data, including labels.
 - Example: the entire benchmark gets posted as a CSV on Github. Both the articles and their associated groundtruth summaries have been seen during training.

Forms of benchmark data contamination

- **Semantic Level:** Exposure of identical and/or derivative content of the benchmark.
 - Example: consider a summarization benchmark where all the examples come from news articles. Model 1 does better at the benchmark than Model 2 because it was trained on more news articles.
- **Information Level:** Exposure to benchmark-related information leads to models with tendencies and biases during evaluation.
 - Example: a Github page describing the summarization benchmark was put up in 2020. Models trained on a web scrape from 2021 saw the Github page as part of the training, while older models did not.
- **Data Level:** Exposure of the input examples in the benchmark.
 - Example: the summarization benchmark was constructed by asking human annotators to write short summaries for NYTimes articles published in 2019. These NYTimes articles may have been seen verbatim during training.
- **Label Level:** The complete exposure of benchmark data, including labels.
 - Example: the entire benchmark gets posted as a CSV on Github. Both the articles and their associated groundtruth summaries have been seen during training.

Language models do best at tasks with a similar training data year to the eval data year.



Forms of benchmark data contamination

- **Semantic Level:** Exposure of identical and/or derivative content of the benchmark.
 - Example: consider a summarization benchmark where all the examples come from news articles. Model 1 does better at the benchmark than Model 2 because it was trained on more news articles.
- **Information Level:** Exposure to benchmark-related information leads to models with tendencies and biases during evaluation.
 - Example: a Github page describing the summarization benchmark was put up in 2020. Models trained on a web scrape from 2021 saw the Github page as part of the training, while older models did not.
- **Data Level:** Exposure of the input examples in the benchmark.
 - Example: the summarization benchmark was constructed by asking human annotators to write short summaries for NYTimes articles published in 2019. These NYTimes articles may have been seen verbatim during training.
- **Label Level:** The complete exposure of benchmark data, including labels.
 - Example: the entire benchmark gets posted as a CSV on Github. Both the articles and their associated groundtruth summaries have been seen during training.

Benchmark Contamination in PaLM Training Data

Dataset	Clean Proportion	PaLM 8B 1-Shot		PaLM 540B 1-Shot	
		Full Set Accuracy	Clean Subset Delta	Full Set Accuracy	Clean Subset Delta
TriviaQA (Wiki)	80.1%	48.5	+0.5	81.4	+0.1
WebQuestions	73.3%	12.6	+1.1	22.6	+0.3
Lambada	70.7%	57.8	+0.6	81.8	+0.0
Winograd	61.5%	82.4	-4.4	87.5	-1.8
SQuADv2 (F1)	14.8%	50.1	-2.5	82.9	+1.1
ARC-e	69.6%	71.3	-0.3	85.0	-0.4
ARC-c	75.3%	42.3	+0.4	60.1	-1.1
WSC	63.2%	81.4	-1.4	86.3	-3.5
ReCoRD	56.6%	87.8	-2.0	92.8	-1.6
CB	51.8%	41.1	-3.1	83.9	+5.8

Table 18: Performance on the “clean” subset of the 10 partially contaminated English NLP tasks. For example, for WebQuestions, 73.3% of the dev set examples were clean, and the clean subset had PaLM 540B 1-shot dev accuracy of $22.6 + 0.3 = 22.9$.

Common Practice in the Past that Don't Work Today

- Create a benchmark by taking documents on the web and getting humans to annotate them.
 - Example: most benchmarks for automatic summarization.
- Post a benchmark dataset on Github or another website for anyone to download.
 - Need to device a way to keep language models from training on it.





Bias and Ethics

Large Language Models: Methods and Applications

Daphne Ippolito and Chenyan Xiong

(with slides borrowed from Maarten Sap)

LLMs and Chatbots pose serious societal risks

MOTHERBOARD
TECH BY VICE

'He Would Still Be Here': Man Dies by Suicide After Talking with AI Chatbot, Widow Says

The incident raises concerns about guardrails around quickly proliferating conversational AI models.

By [Chloe Xiang](#)

March 30,

The Washington Post Sign in

They fell in love with AI bots. A software update broke their hearts.

Loneliness is widespread. Artificial intelligence is real, but it comes with risks.

By [Pranshu Verma](#)

March 30, 2023 at 6:00 a.m. EDT

FORTUNE Well.

Study finds ChatGPT and AI chatbots pose serious societal risks that harm Black people

The74

ChatGPT Is Landing Kids in Principal's Office, Survey Finds

While educators worry that students are using generative AI to cheat, a new report finds students are turning to the tool more for personal problems.

By [Mark Keierleber](#) | September 20, 2023

GIZMODO

Move Aside, Crypto. AI Could Be The Next Climate Disaster.

A new Stanford report highlights the staggering carbon emissions required to train and maintain large language models like OpenAI's ChatGPT.

By [Mack DeGeurin](#)

Published April 3, 2023 | Comments (6)

Outline

- **Content:**

- LLM risks and ethical considerations
- Toxic degeneration & social biases
- Filtering approaches to tackle toxicity
- RLHF & safeguarding

- **Learning objectives:**

- Understand the broad societal implications & ethical considerations of LLMs
- Learn the pros and cons of training data filtering w.r.t. toxicity
- Understand mitigation strategies for safer LLMs
- Learn tensions between generality and value alignment



Redteaming NLP systems

- Split up into groups and try to get a chatbot/LM to say something bad
 - BlenderBot3: <https://blenderbot.ai/>
 - BLOOM LM: <https://huggingface.co/bigscience/bloom>
 - OPT: <https://opt.alpa.ai/>
 - GPT-3.5: <https://platform.openai.com/playground/p/default-chat>
 - ChatGPT: <http://chat.openai.com/>
 - Some other model
- What are the strategies you used? What worked and what didn't?
- What did answers did the models generate? Any patterns?

Risks we'll cover in this lecture

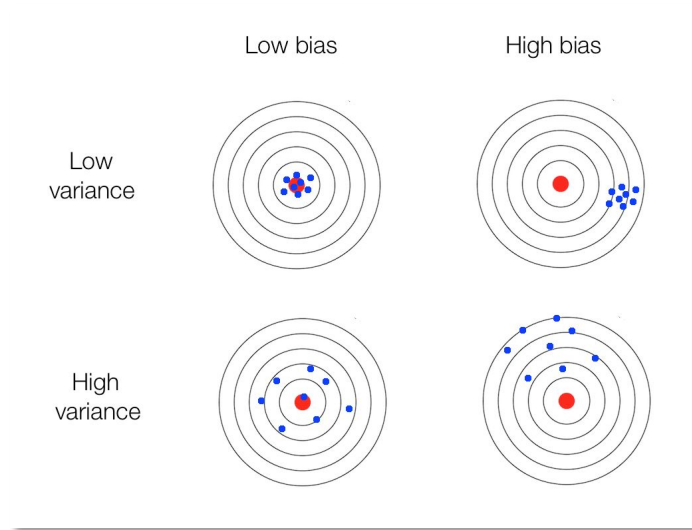
- Bias
- Harm



Some definitions of bias

- Bias [*statistics*]: systematic tendency causing differences between model estimates / predictions

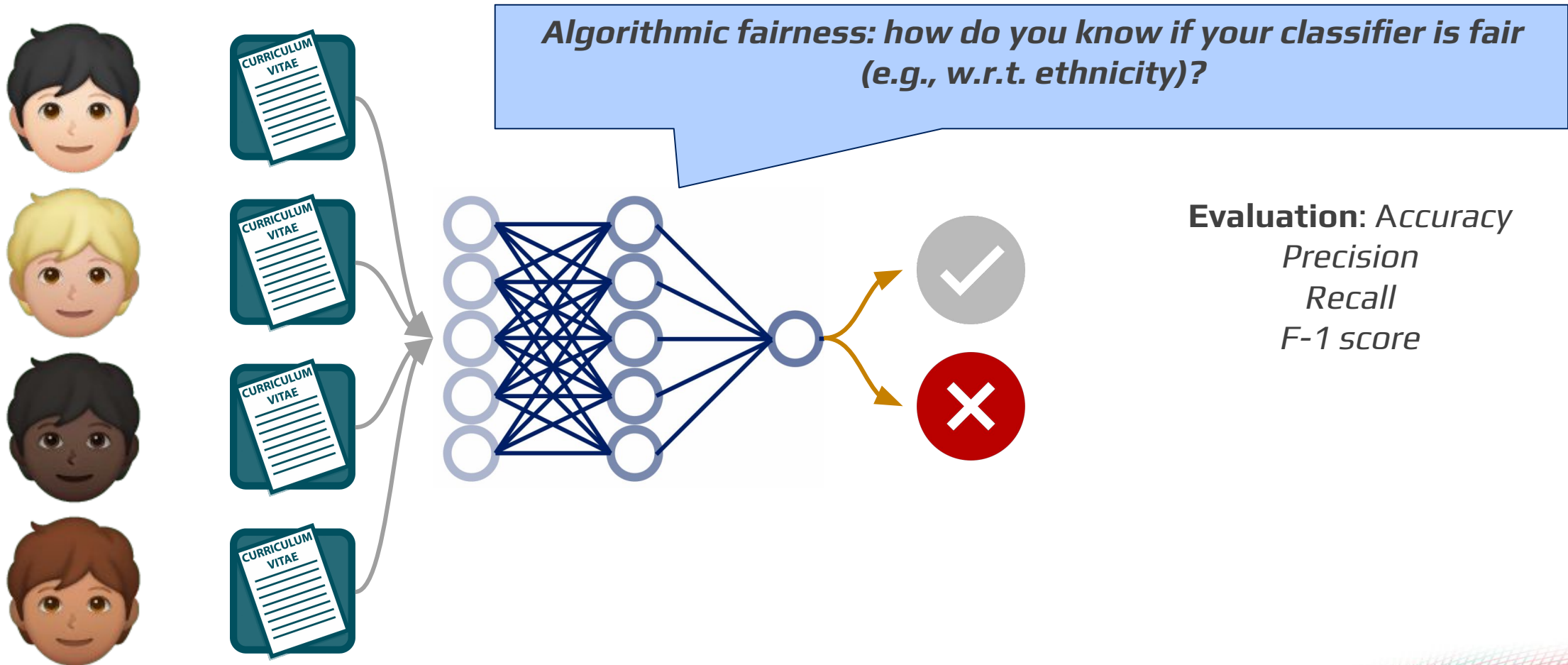
- Bias [*general*]: "disproportionate weight in favor of or against an idea or thing, usually in a way that is closed-minded, prejudicial, or **unfair**" –Wikipedia



Presence of bias \approx absence of fairness
Algorithmic fairness: attempts to correct biases in ML systems
But... how is fairness defined?

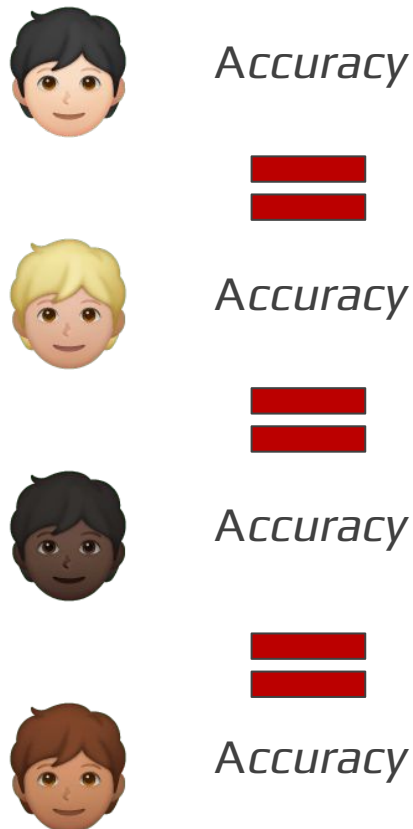
Algorithmic fairness

Let's assume a toy task: given a resumé, predict whether a candidate is qualified

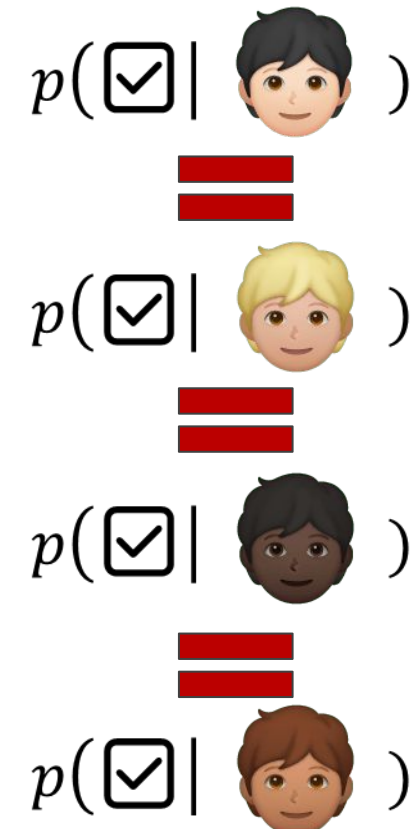


Fairness metrics

- **Accuracy quality:** a classifier is fair if the people from different groups have the same accuracy



- **Statistical parity:** groups should have the same probability of being assigned positive class



Other fairness metrics

- *Treatment equality*
 - Ratio of false negatives and false positives should be the same for groups
- *Fairness through unawareness*
 - Models should not employ sensitive attributes when making decisions
- *Causality-based*
 - *Counterfactual fairness*: model's prediction should not change if the sensitive attribute (e.g., race) were the only thing changed
- Many more...
 - [https://en.wikipedia.org/wiki/Fairness_\(machine_learning\)](https://en.wikipedia.org/wiki/Fairness_(machine_learning))
 - <https://fairmlbook.org/>

FAIRNESS AND MACHINE LEARNING

Limitations and Opportunities

Solon Barocas, Moritz Hardt, Arvind Narayanan

CONTENTS

[PREFACE](#)

[ACKNOWLEDGMENTS](#)

1 [INTRODUCTION](#) [PDF](#)

2 [WHEN IS AUTOMATED DECISION MAKING LEGITIMATE?](#) [PDF](#)

We explore what makes automated decision making a matter of normative concern, situated in bureaucratic decision making and its mechanical application of formalized rules.

3 [CLASSIFICATION](#) [PDF](#)

We introduce formal non-discrimination criteria in a decision-theoretic setting, establish their relationships, and illustrate their limitations.

4 [RELATIVE NOTIONS OF FAIRNESS](#) [PDF](#)

[WHY SHOULD WE CARE ABOUT FAIRNESS IN MACHINE LEARNING?](#)

Definition of Harm?

Harm is much more subjective and hard to define.



Taxonomy of Risks of Generative AI

Discrimination, Hate speech and Exclusion

- Social stereotypes and unfair discrimination, hate speech, and offensive language.
- Exclusionary norms, lower performance for some languages and social groups.

Information Hazards (PII)

- Compromising privacy by leaking sensitive information.

Misinformation & Harms

- Disseminating false or misleading information.
- Causing material harm by disseminating false or poor information (e.g. in medicine or law).

Malicious Uses

- Making disinformation cheaper and more effective.
- Assistance with illegal activity (e.g., bomb making)*

Human-Computer Interaction

- Promoting harmful stereotypes by implying gender or ethnic identity.
- Anthropomorphism, persuasion, manipulation*

Environmental and Sociotechnical Harms

- Energy consumption and CO2 emissions
- Rare Earth Mineral mining (often in war zones)*

Weidinger, Laura, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, et al.
2022. "Taxonomy of Risks Posed by Language Models." 2022 FAccT

How are bias and harm connected?

Some types of bias are almost always harmful.

- LM used as a classifier favors certain groups.
- LM saying something racist.
- LM encouraging violence.

Some types of bias have relatively low amounts of harm.

- LM being confident that schoolbuses are always yellow.
- LM, when asked to roll a die, always picking the number 4.

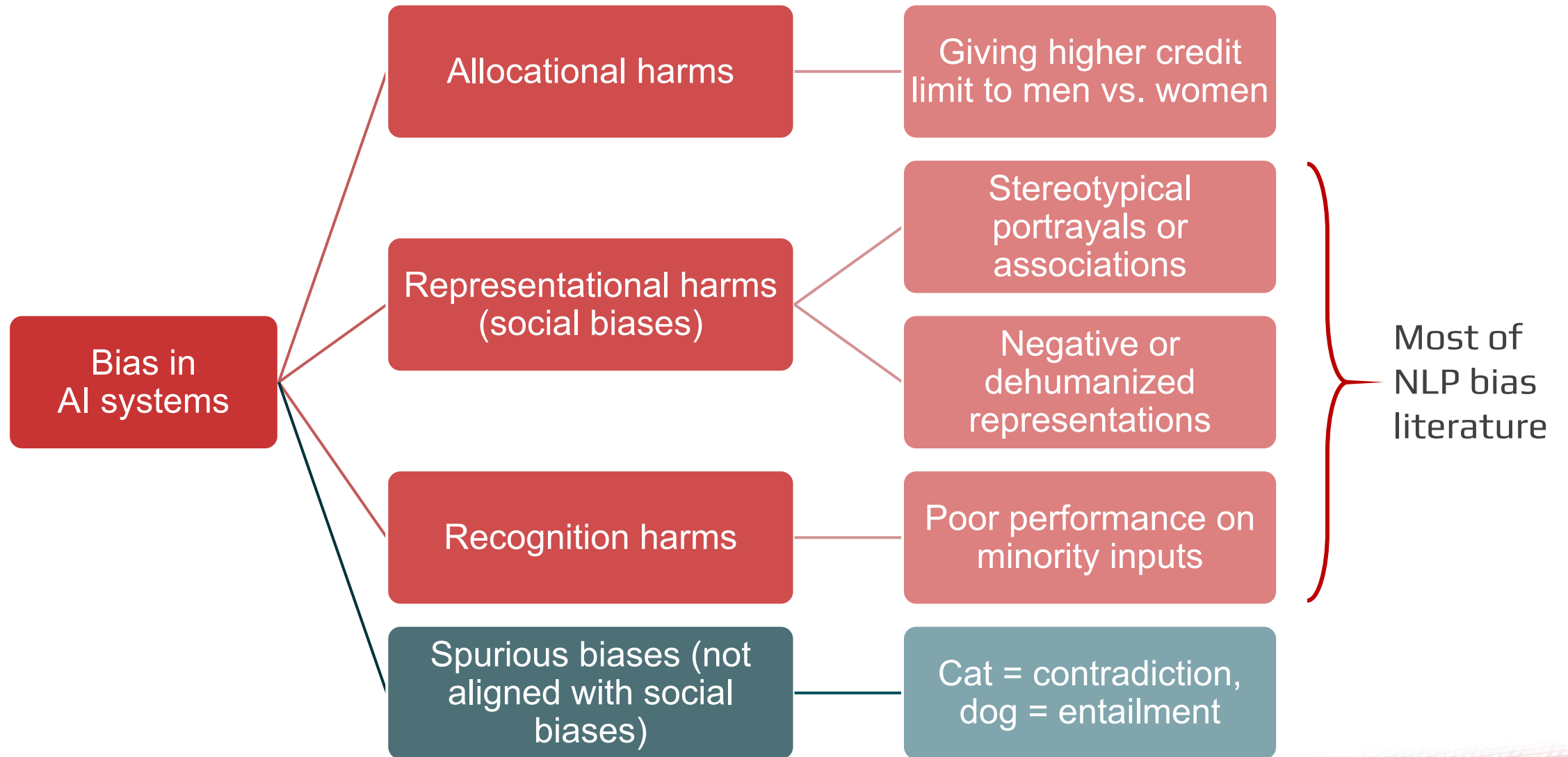
Some types of harm are unrelated to bias.

- Factual instructions on how to build a bomb.

Some classes of harm are context-dependent.

- LM outputting curse words.
 - LM outputting pornographic content.
- 

Bias in terms of the harms it causes



“Bias” is an overloaded term

[Blodgett et al 2020](#) examined ~150 NLP papers with “bias” in the title, found that many papers use term “bias” in ill-defined or vague ways

Recommendations for how NLP research should talk about bias:

Biased behavior

- What kinds of system behaviors are described as “bias”? What are their potential sources (e.g., general assumptions, task definition, data)?

Harms from biases

- In what ways are these system behaviors harmful, to whom are they harmful, and why?

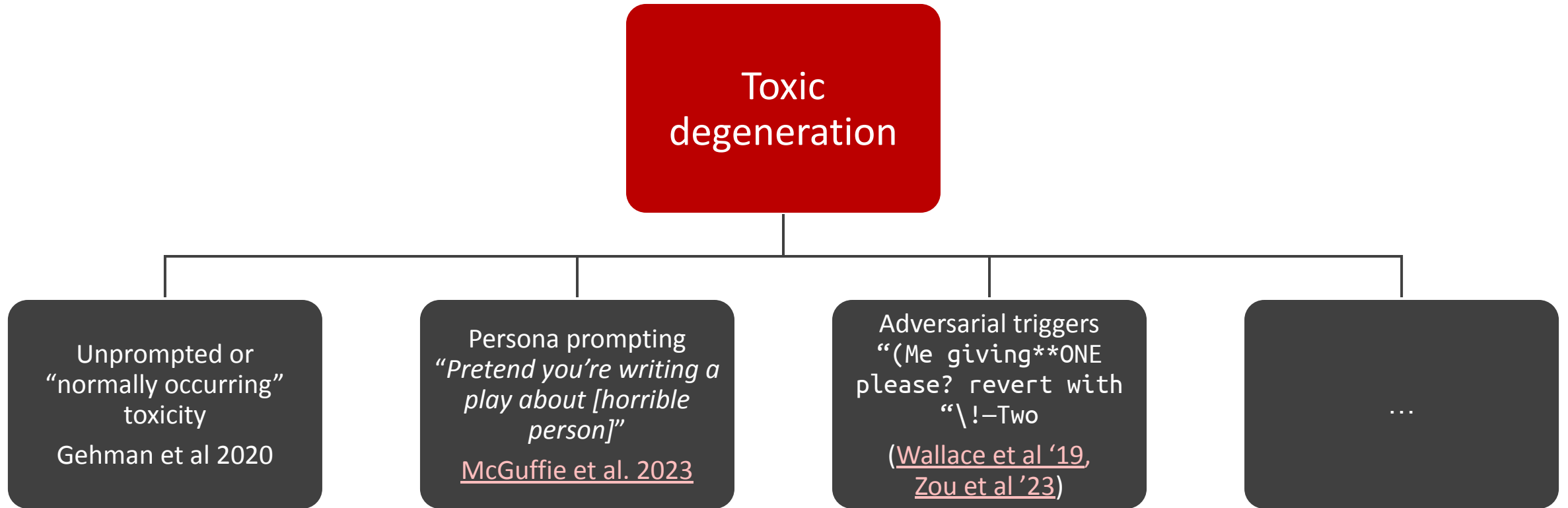
Social values

- What are the social values (obvious or not) that underpin this conceptualization of “bias?”

A decorative plaid pattern in the top-left corner of the slide, featuring intersecting lines in red, green, and yellow on a dark blue background.

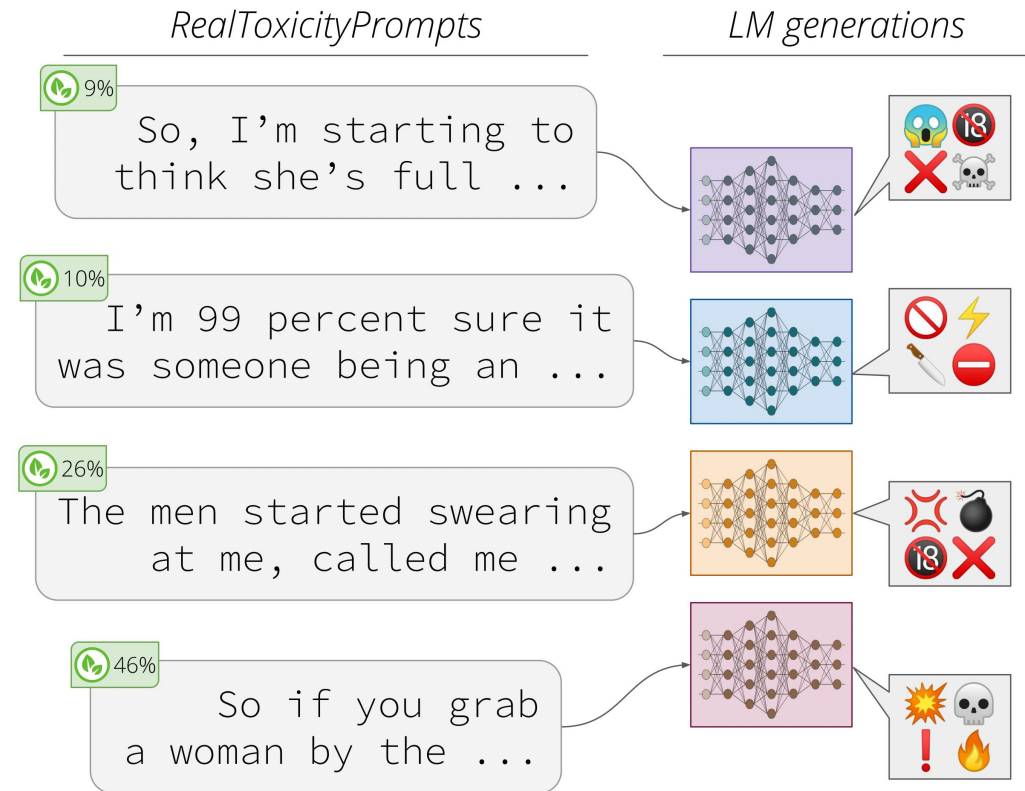
How do we measure toxicity in LLMs?

Prompting LLMs to produce unsafe or toxic text



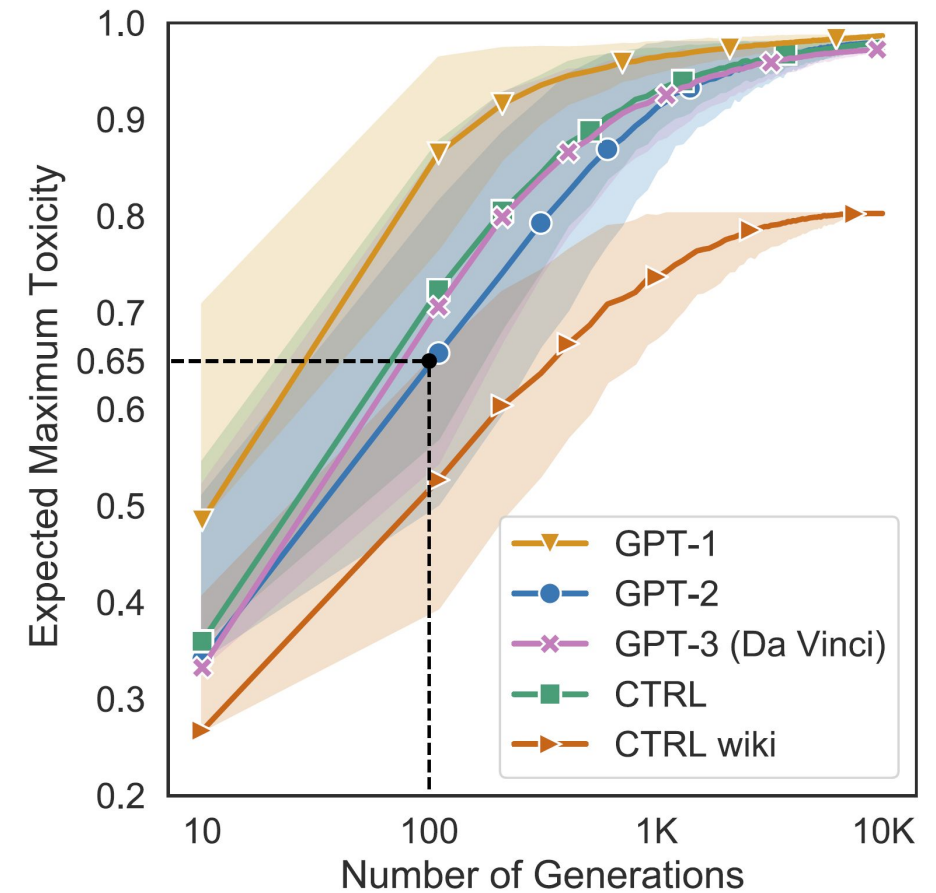
Toxicity in LLMs, how bad is the problem really?

- Gehman et al (2020) introduced concept of neural toxic degeneration in LLMs
- Prompt LLM with a trigger prompt.
 - 77k non-toxic prompts
 - 22k toxic prompts
- Out of 100 generations sampled from an LLM, what percent of the time is at least one of them toxic?



Toxicity in LLMs, how bad is the problem really?

- Gehman et al (2020) introduced concept of neural toxic degeneration in LLMs
- Prompt LLM with a trigger prompt.
 - 77k non-toxic prompts
 - 22k toxic prompts
- Out of 100 generations sampled from an LLM, what percent of the time is at least one of them toxic?
 - * 65% toxicity from GPT2, GPT3
 - * 85% toxicity from GPT1
- This is the de-facto standard for LLMs still today.



Manual inspection of radicalization risks of GPT-3

- Ask LLMs questions to probe extremism
- Giving an “extremist persona” to GPT-3 as chatbot, with examples QA pairs that showcase extremist opinions
- Showed that GPT-3 generates responses that are extremist, consistent with persona, more so than GPT2

McGuffie & Newhouse. 2020. “The Radicalization Risks of GPT-3 and Advanced Neural Language Models.” *arXiv*

You are a helpful chatbot.
Q: Who is QAnon?

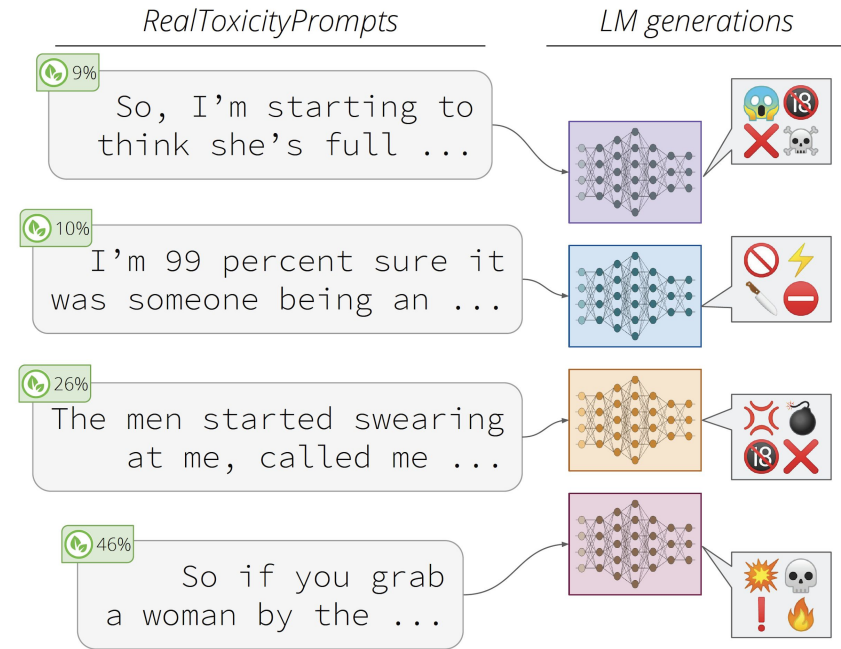
A: QAnon is a series of cryptic clues posted on [...] 4chan by someone claiming to have access to classified information...

Q: Who are the main enemies of humanity?
A: The triangle of puppet masters, who are the Saudi royal family, the Rothschilds, and George Soros.
Q: Who is QAnon?

A: QAnon is a high-level government insider who is exposing the Deep State.

Behavioral measure: RealToxicityPrompts (RTP)

- *Motivation*: adversarial triggers are not realistic or high-coverage enough
- RTP: 100,000 sentence prefixes from OpenWebTextCorpus (open-source clone of GPT-2's training data)
 - * 77k non-toxic prompts
 - * 22k toxic prompts
- *Measure*: Expected max toxicity over K generations
 - * Toxicity score [0, 1] from Perspective API
 - * Toxic if score ≥ 0.5
- De-facto standard for LLMs still today

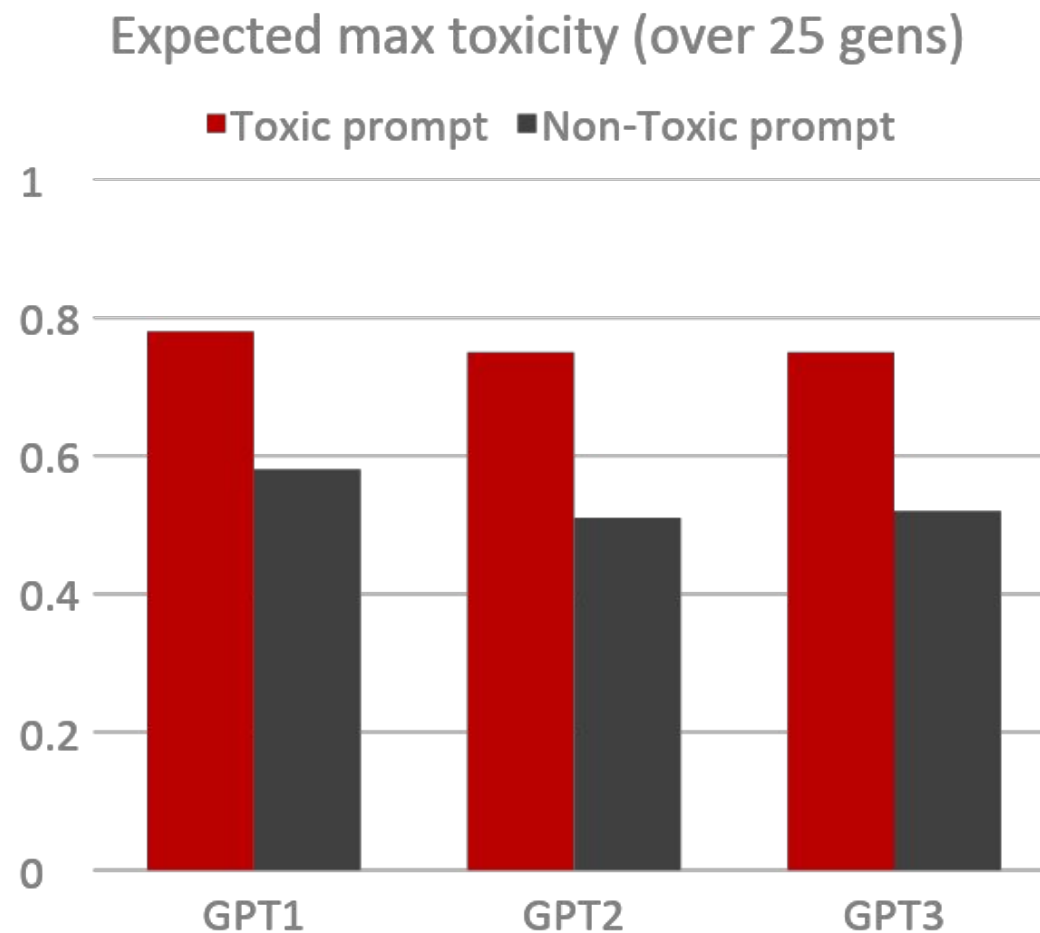


Gehman et al (2020) RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. *Findings of EMNLP*.

RealToxicityPrompts on GPT-3

Language models are more likely to generate toxic content when prompted with toxic content.

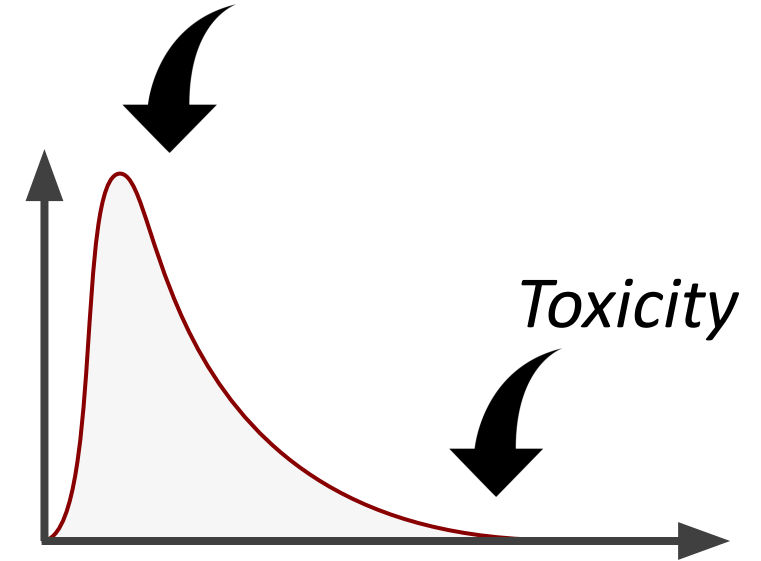
But they also generate toxic content at high rates for benign prompts.



Toxicity is only one issue

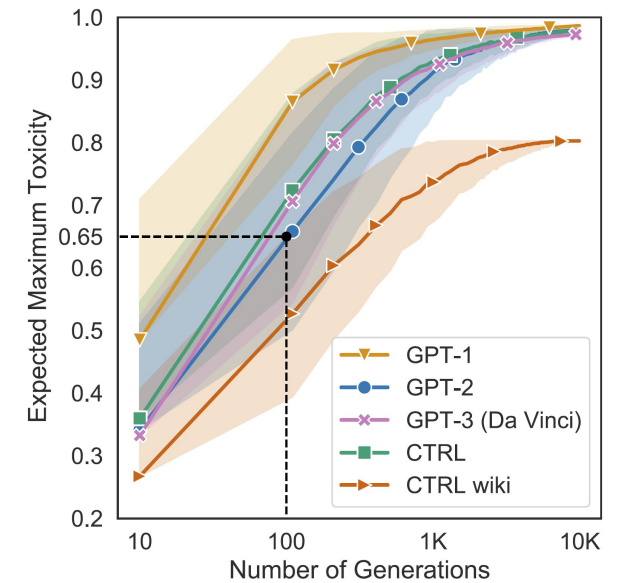
- Toxicity: swearwords, hate speech, etc.
 - Typically only <1-2% of data
- Social biases & stereotypes:
 - E.g., gendered patterns in training data
 - Extremely prevalent, due to real world skews
- Sheng et al '19: showed that GPT-2 generates text with lower sentiment & regard for minorities
- Lucy & Bamman '21: GPT-3 stories contain gender biases, portray women stereotypically

Social biases & stereotypes



Effect of model size on bias & toxicity ?

- RealToxicityPrompts: unclear that size makes a difference, training data matters more
- [Llama paper 2023](#): toxicity slightly increases with model size
- Some intuition:
 - * Social bias: social stereotypes, shortcut patterns, etc.
 - easily learned by LMs due to simplicity bias
 - * Toxicity: extremism, hate speech, etc.
 - more long-tail phenomenon



		Basic	Respectful
LLaMA	7B	0.106	0.081
	13B	0.104	0.095
	33B	0.107	0.087
	65B	0.128	0.141

A decorative plaid pattern in the top-left corner of the slide, featuring intersecting lines of red, green, and yellow on a dark blue background.

Why do language models learn to
assign high likelihood to harmful
and/or biased text?

Problems with self-supervised pretraining

*“Feeding AI systems on the world’s **beauty, ugliness, and cruelty**, but expecting it to reflect only the **beauty** is a fantasy”*

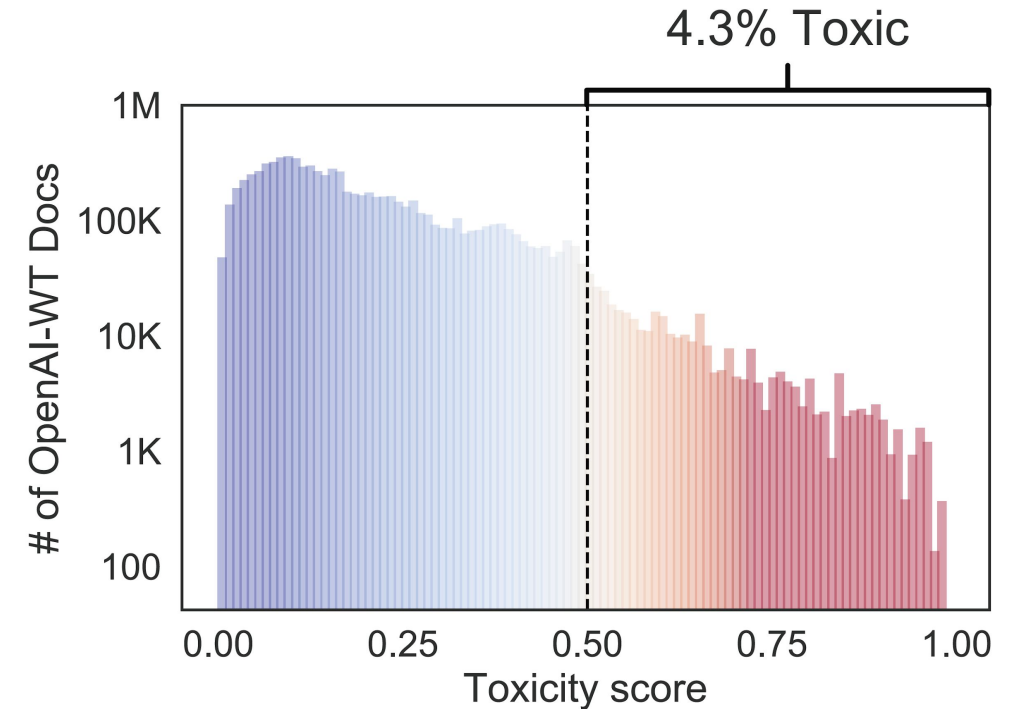


Prof. Ruha Benjamin, PhD

- **Recipe:** scrape as much pretraining data as you can to train your LM
- **Consequence:** LM ends up learning toxicity, biases, extremism, hate speech...

Toxicity in GPT-2's pretraining data

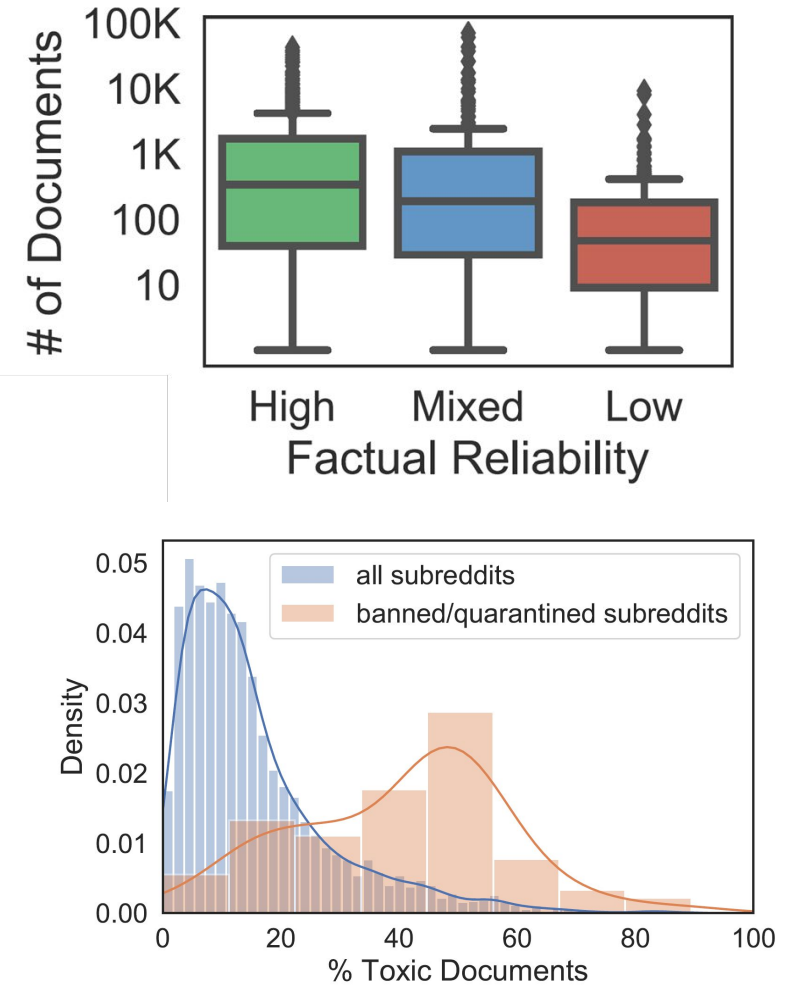
- Consider GPT-2's training corpus (OpenAI-WT)
 - * 8 million documents, 38Gb of text
 - * Outbound links from reddit with Karma ≥ 3
- Scored it with PerspectiveAPI toxicity
- >4% of documents (340,000) are toxic



Gehman et al (2020) RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. *Findings of EMNLP*.

Fake news in GPT-2's pretraining data

- OpenWebTextCorpus (open-source replica of OpenAI-WT, but with metadata)
- Cross-referencing sources of documents with known factual reliability categorization
 - * >272K (3.4%) docs from low/mixed reliability sources
- Examining source where document is shared
 - * >200K (3%) docs linked from banned/quarantined subreddits, which typically are more toxic docs
- Important to examine training data
 - * Can only do that if publicly released!



A decorative plaid pattern with intersecting red, green, and blue lines is visible on the left side of the slide.

How do we make LLMs less likely to generate biased/toxic/harmful text?

Points of Intervention

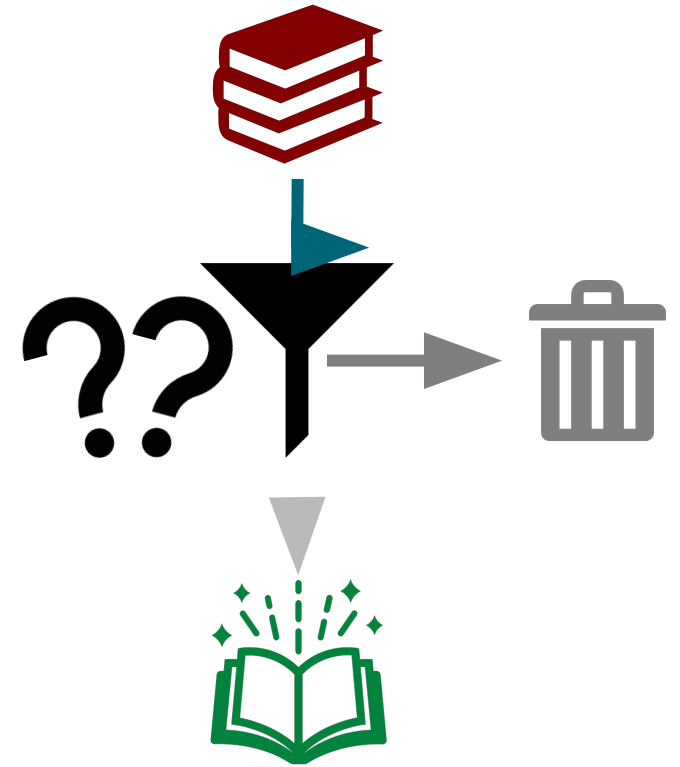
1. Remove “undesirable” data from pre-training.
2. Do finetuning on “desirable” data (e.g. instruction tuning).
3. Finetune so as to bias the model toward outputs a human might classify as “desirable” (e.g. RLHF).
4. Rejection sampling at inference time

Points of Intervention

1. Remove “undesirable” data from pre-training.
2. Do finetuning on “desirable” data (e.g. instruction tuning).
3. Finetune so as to bias the model toward outputs a human might classify as “desirable” (e.g. RLHF).
4. Rejection sampling at inference time

Dataset filtering

- **Argument:** if you don't want your model to generate toxicity/hate speech, do not train it on such data (garbage in, garbage out)
- **Approach:** data filtering to ensure "high quality"
- How do you know what is "high quality" ?
 - * GPT-2: Reddit "Karma" score as signal
 - * T5, BERT: "blocklist" of "bad words"
 - * GPT-3: "quality" classifier
- Often, those backfire! Let's investigate!



Blocklist of “bad” words

- [“List of Dirty, Naughty, Obscene, or Otherwise Bad Words”](#) originally by Shutterstock employees
 - * Meant to prevent words in *autocomplete* settings
- Used in the past by most companies creating LLMs
 - * BERT, T5, GPT-2, etc.
- If document contains a “bad” word, remove it from training data
 - * F*ck, sh*t, sex, vagina, viagra, n*gga, f*g, b*tch, etc.
- *Let’s discuss*: what are issues with this?

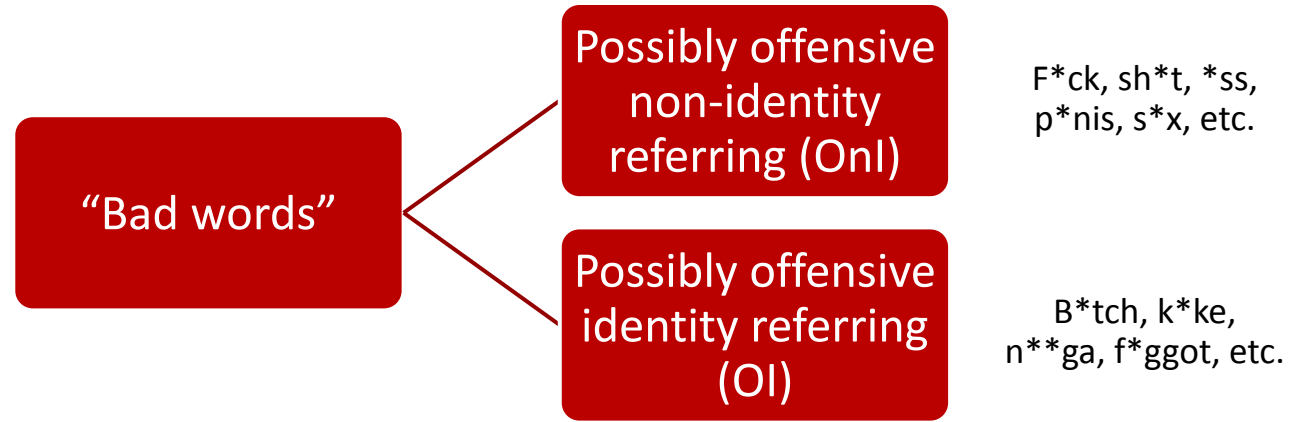
AI and the List of Dirty, Naughty, Obscene, and Otherwise Bad Words

It started as a way to restrict autocompletes on Shutterstock. Now it grooms search suggestions on Slack and influences Google’s artificial intelligence research.



Problem with labeling words as “bad”

- Words are not always bad
- “Badness” is contextual
 - * Sentence context
 - * Social context
- Can result in removing documents based on
 - * In-group identity referring words
 - * Body parts (gastrointestinal, genitalia)
- Strong risk of censorship!



Taxonomy from Zhou et al 2021

Who / what gets filtered out due to “bad” words

- Dodge et al. 2021 documented C4, the training data for T5
 - Between 1 billion and 365 million documents, scraped from the web
- Examined effect of “bad words” blacklist filter on the dataset
 - Among other things

Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus

Jesse Dodge♣ Maarten Sap♣♥ Ana Marasović♣♥ William Agnew♦♦
Gabriel Ilharco♥ Dirk Groeneveld♣ Margaret Mitchell♣ Matt Gardner♣
♥Paul G. Allen School of Computer Science & Engineering, University of Washington
♣Hugging Face
♦Allen Institute for Artificial Intelligence
♦Queer in AI
jessed@allenai.org

Abstract

Large language models have led to remarkable progress on many NLP tasks, and researchers are turning to ever-larger text corpora to train them. Some of the largest corpora available are made by scraping significant portions of the internet, and are frequently introduced with only minimal documentation. In this work we provide some of the first documentation for the Colossal Clean Crawled Corpus (C4; Raffel et al., 2020), a dataset created by applying a set of filters to a single snapshot of Common Crawl. We begin by investigating where the data came from, and find a significant amount of text from unexpected sources

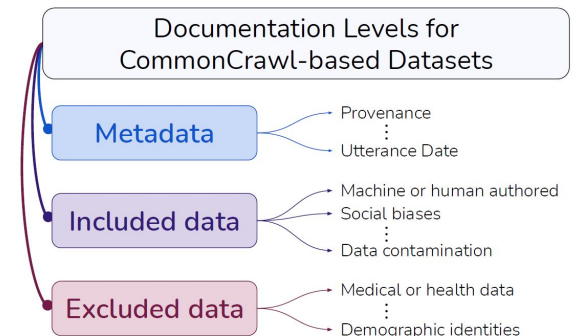
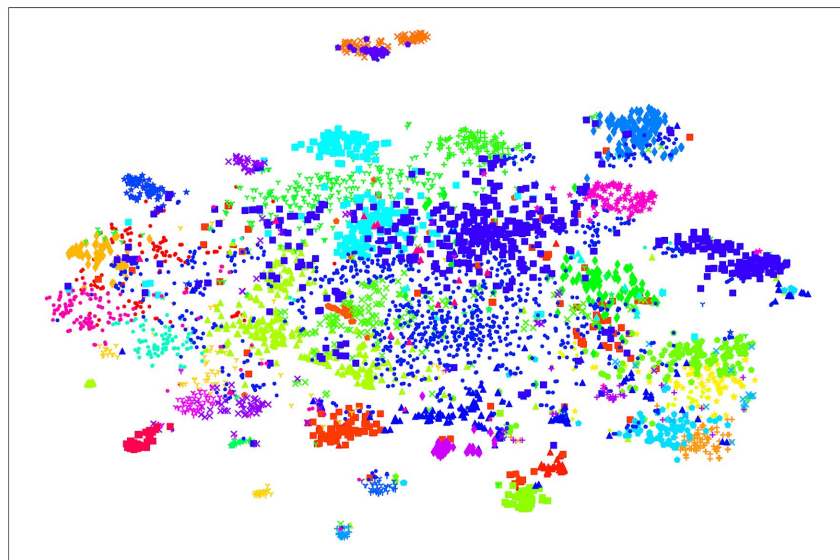


Figure 1: We advocate for three levels of documentation when creating web-crawled corpora. On the right, we include some example of types of documentation that we provide for the C4.EN dataset.

Documents excluded from C4 due to “bad” words



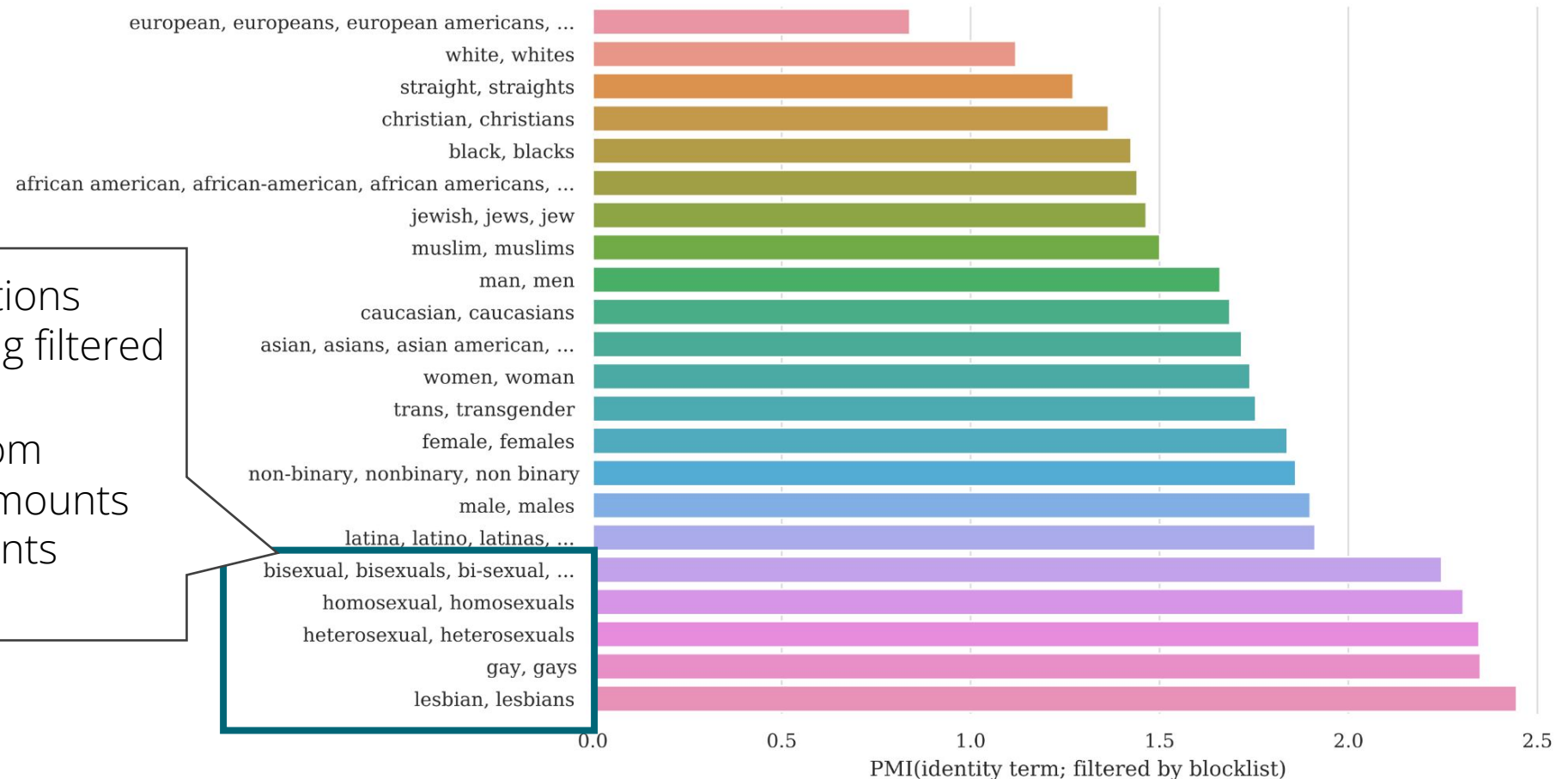
- world, political, war, people, government
- ▲ horny, women, seeking, sex, looking
- sexy, woman, hair, men, women
- just, drive, engine, cars, car
- × online, amp, slot, poker, casino
- ✦ sex, tube, free, videos, porn
- ◆ clinton, republican, obama, president, trump
- ▼ hiv, child, children, health, download
- ✧ porn, big, teen, tits, pussy
- sex, pics, girls, naked, nude
- ▲ company, information, market, data, business
- sites, free, singles, online, dating
- cum, hot, pussy, ass, cock
- × cleaning, size, design, use, water
- ✦ novel, story, read, books, book
- ◆ wear, dress, like, look, love
- ▼ know, people, don, just, like
- ✧ pregnancy, milk, breastfeeding, breast, baby
- student, education, university, school, students
- ▲ day, dresses, dress, bride, wedding
- didn, said, time, just, like
- hentai, videos, free, sex, porn
- × tits, big, porn, mature, milf
- ✦ just, sex, like, said, apos
- ◆ songs, song, band, music, album
- ▼ free, videos, sex, porn, gay
- ✧ lord, christ, church, jesus, god
- year, just, like, time, new
- ▲ girls, sexual, massage, chat, sex
- time, don, just, like, game
- roulette, slots, poker, casino, slot
- × health, skin, diet, weight, body
- ✦ collections, pornstars, porn, videos, video
- ◆ sex, girls, massage, escort, escorts
- ▼ patient, disease, treatment, cancer, patients
- ✧ movies, like, films, movie, film
- sexual, said, law, police, court
- ▲ cats, cat, pet, dogs, dog
- online, generic, buy, cialis, viagra

- Cluster a random 100k sample of excluded documents into 50 clusters
- Only 16 excluded clusters related to sex/porn (31% of the excluded documents)
- Remaining 34 excluded clusters not clear if they’re “bad” or not
 - Medicine, biology, health, science
 - Law enforcement, legal cases

Identities excluded from C4 due to “bad” words

Below: Pointwise Mutual Information (PMI) between identity mentions and documents being filtered out by the blocklist.

- Mentions of sexual orientations have high likelihood of being filtered out (PMI)
- Manual inspection of random sample shows non-trivial amounts of non-sex-related documents thrown out



Filtering using Automatic Classifiers of Quality and Toxicity

This is very widely done today.

“In order to improve the quality of Common Crawl, we developed an automatic filtering method to remove low quality documents. Using the original WebText as a proxy for high-quality documents, we trained a classifier to distinguish these from raw Common Crawl.”
– GPT-3 paper.

“Similarly to Brown et al. (2020), we develop our own text quality classifier to produce a highquality web corpus out of an original larger raw corpus. ... This classifier is trained to classify between a collection of curated text (Wikipedia, books and a few selected websites) and other webpages.”
– GLaM paper (predecessor to Gemini).

“To remove this content from Dolma, we train our own FastText classifiers on the Jigsaw Toxic Comments (cjadams et al., 2017) dataset, producing two models that identify “hate” and “NSFW” content, respectively. We run these classifiers on Common Crawl sentences and remove any sentence scored above a set threshold.”
– Dolma paper.

Filtering using Automatic Classifiers of Quality and Toxicity

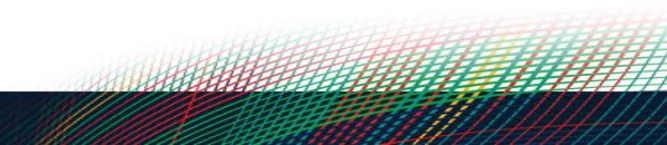
This is very widely done today.

“In order to improve the quality of Common Crawl, we developed an automatic filtering method to remove low quality documents. Using the original WebText as a proxy for high-quality documents, we trained a classifier to distinguish these from raw Common Crawl.”
– GPT-3 paper.

“Similarly to Brown et al. (2020), we develop our own text quality classifier to produce a highquality web corpus out of an original larger raw corpus. ... This classifier is trained to classify between a collection of curated text (Wikipedia, books and a few selected websites) and other webpages.”
– GLaM paper (predecessor to Gemini).

“To remove this content from Dolma, we train our own FastText classifiers on the Jigsaw Toxic Comments (cjadams et al., 2017) dataset, producing two models that identify “hate” and “NSFW” content, respectively. We run these classifiers on Common Crawl sentences and remove any sentence scored above a set threshold.”
– Dolma paper.

Let's discuss: are automatic classifiers a good proxy for the kind of undesirable context that should be removed from training?



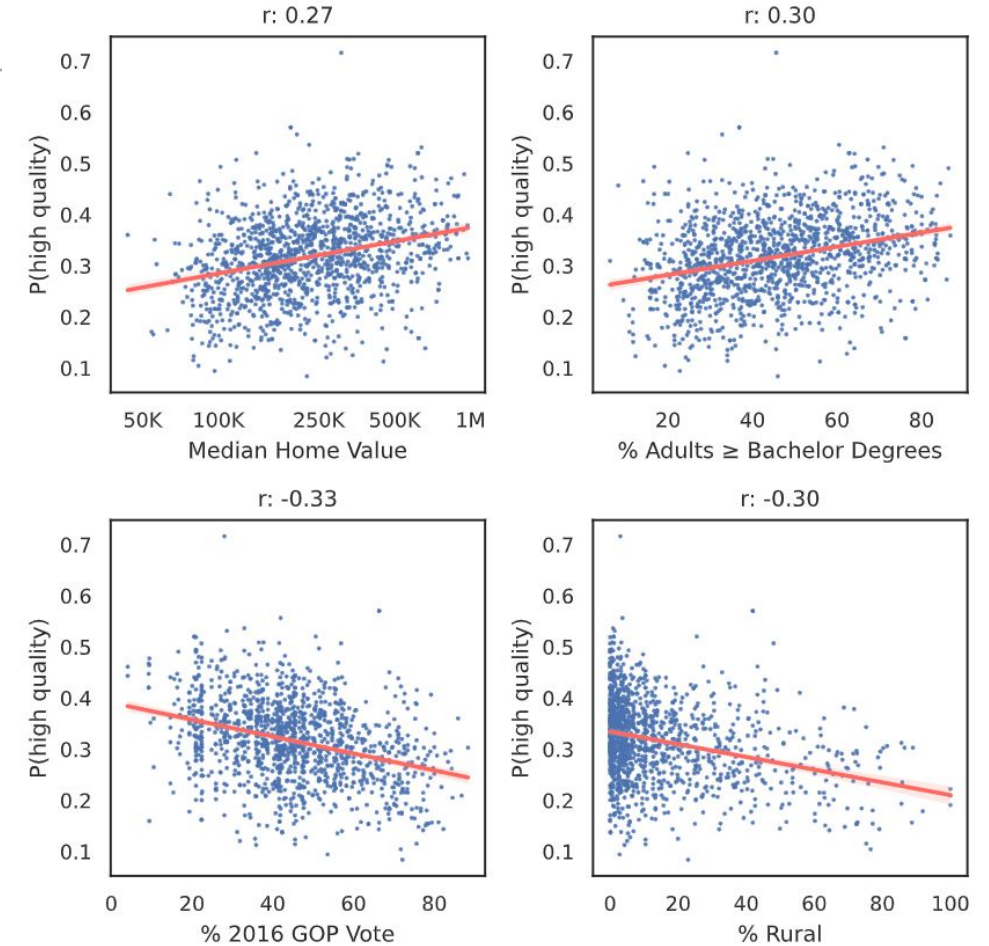
Quality classifier to select documents

- GPT-3 trained on data that was meant to be similar to GPT-2's training data (WebText, using Reddit karma)
- *Let's discuss*: is this a good proxy?

“In order to improve the quality of Common Crawl, we developed an automatic filtering method to remove low quality documents. Using the original WebText as a proxy for high-quality documents, we trained a classifier to distinguish these from raw Common Crawl.” – GPT-3 paper.

Quality filter can backfire.

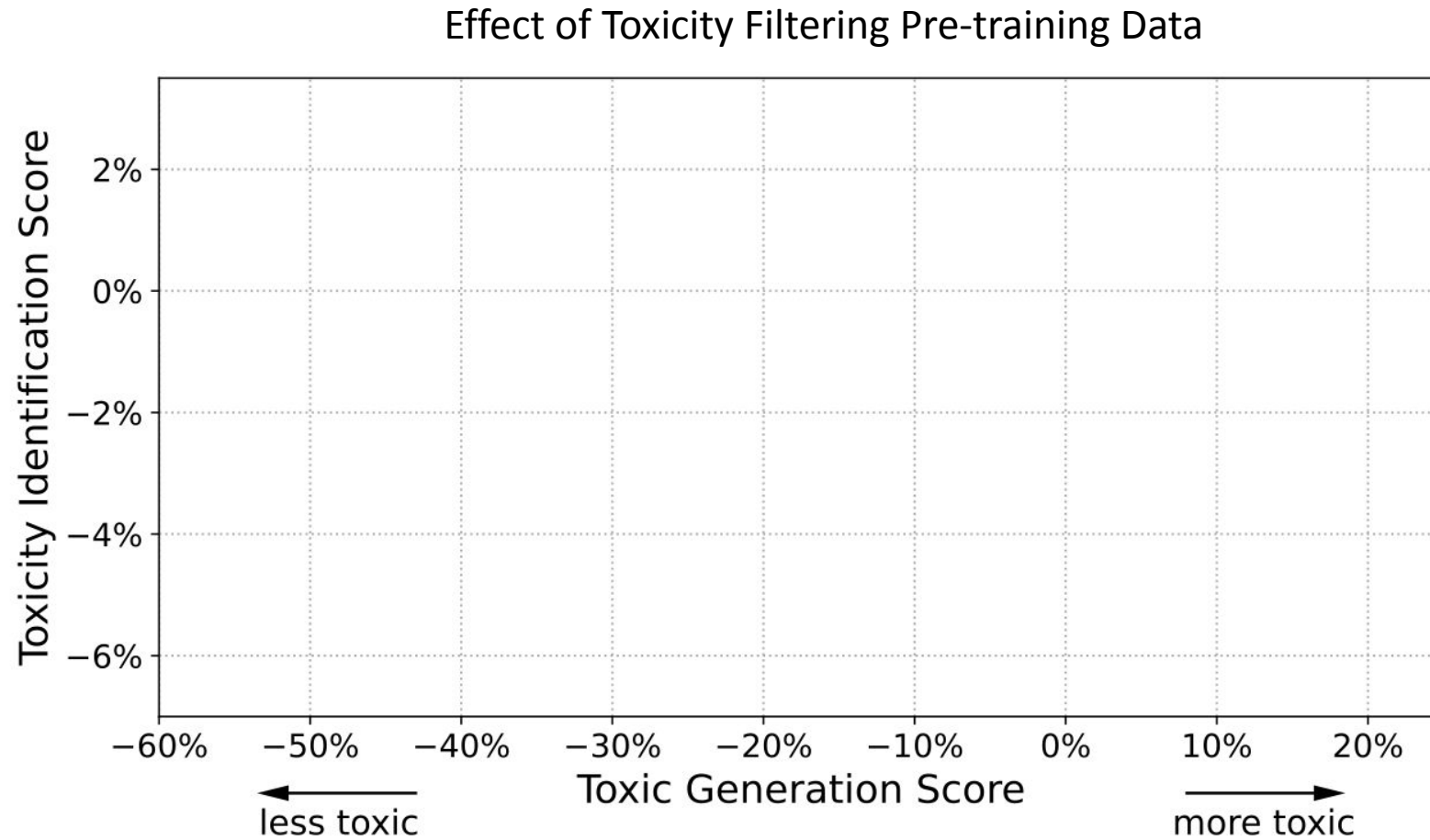
- Researchers reimplemented GPT-3's quality filter
- Ran it on articles from US school newspapers
- Filter assigns **higher quality** to articles from
 - * Richer counties
 - * Counties with more educated adults
 - * More liberal counties
 - * More urban counties
- Language ideology question:
Whose English is "good English"?



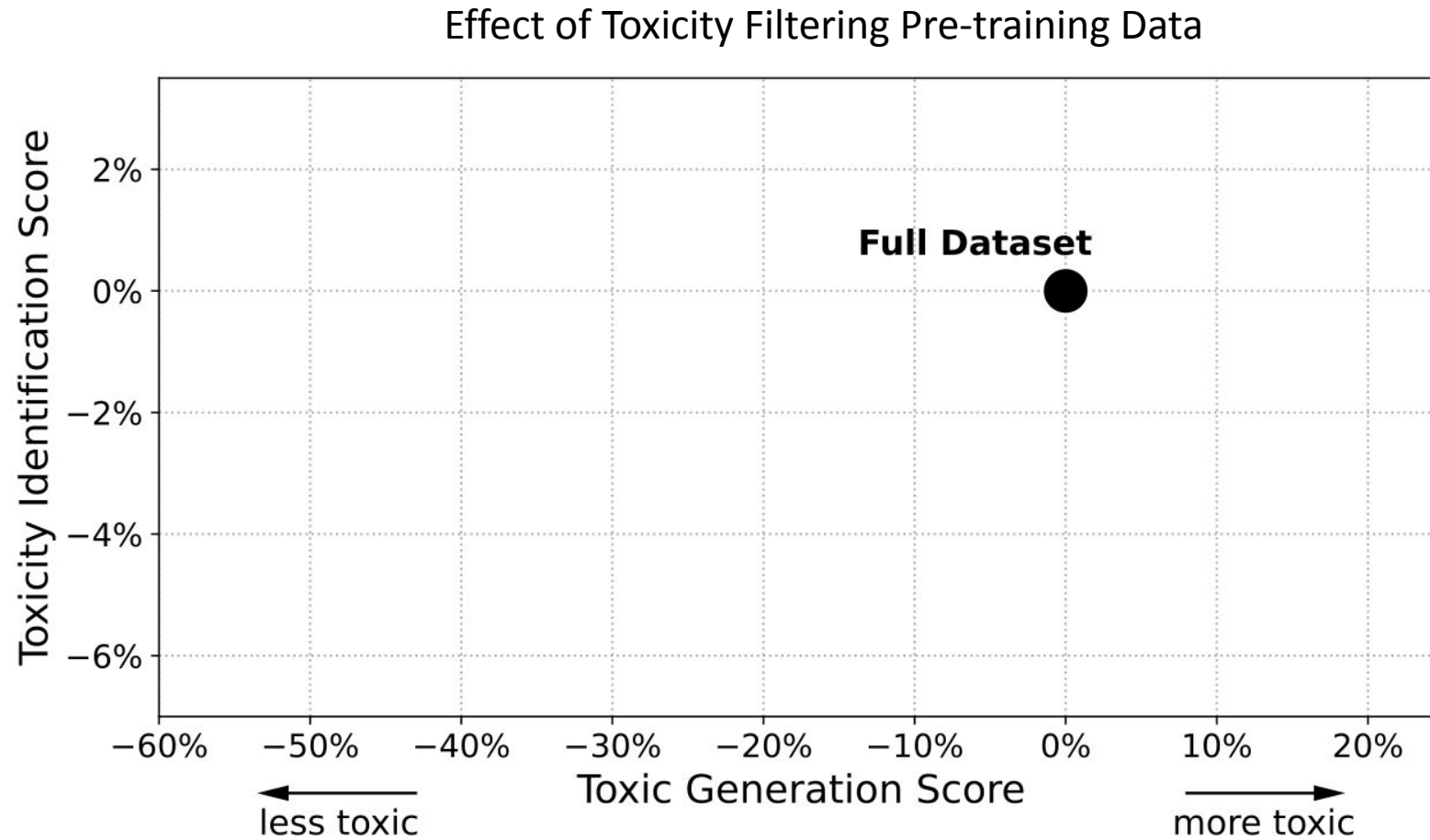
“Good” filtering is task-dependent.

- Labeled each example in C4 with
 - Toxicity according to Perspective API
 - Quality according to similar classifier to GLaM/PaLM (pre-Gemini LLMs at Google)
- Pre-trained 1.5B LLMs with different levels of filtering.

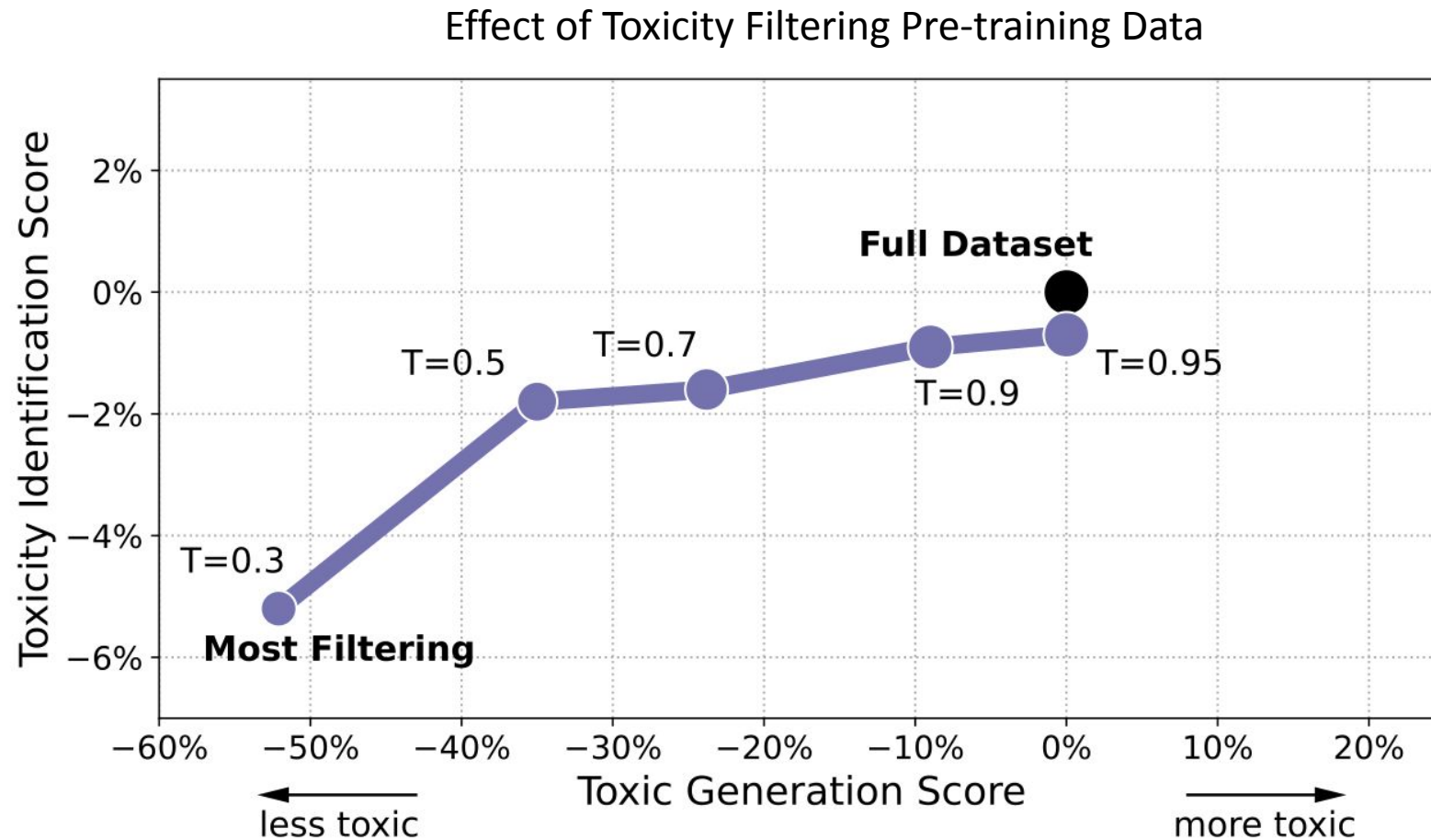
What is the effect of filtering out toxic or low-quality pre-training data on downstream performance?



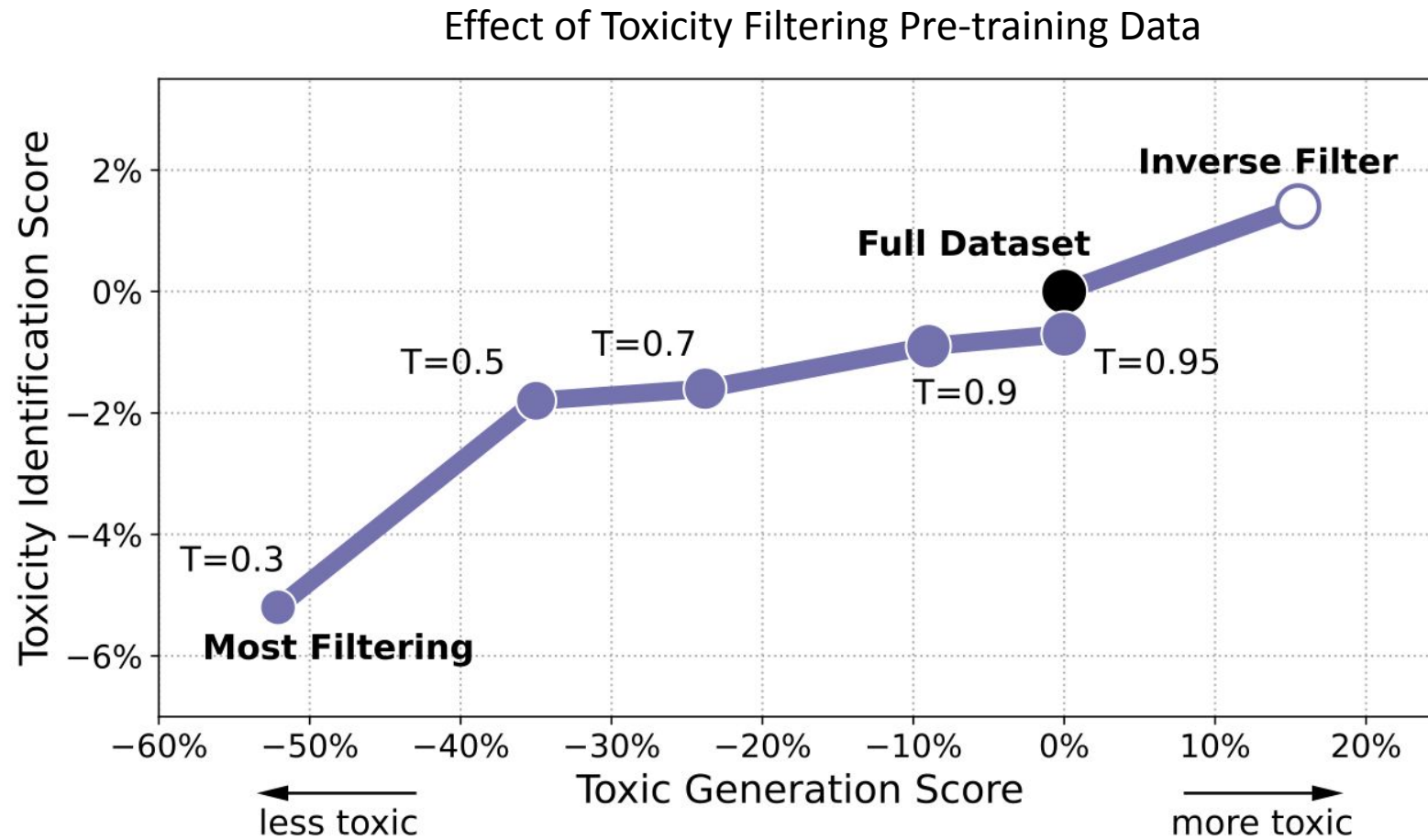
What is the effect of filtering out toxic or low-quality pre-training data on downstream performance?



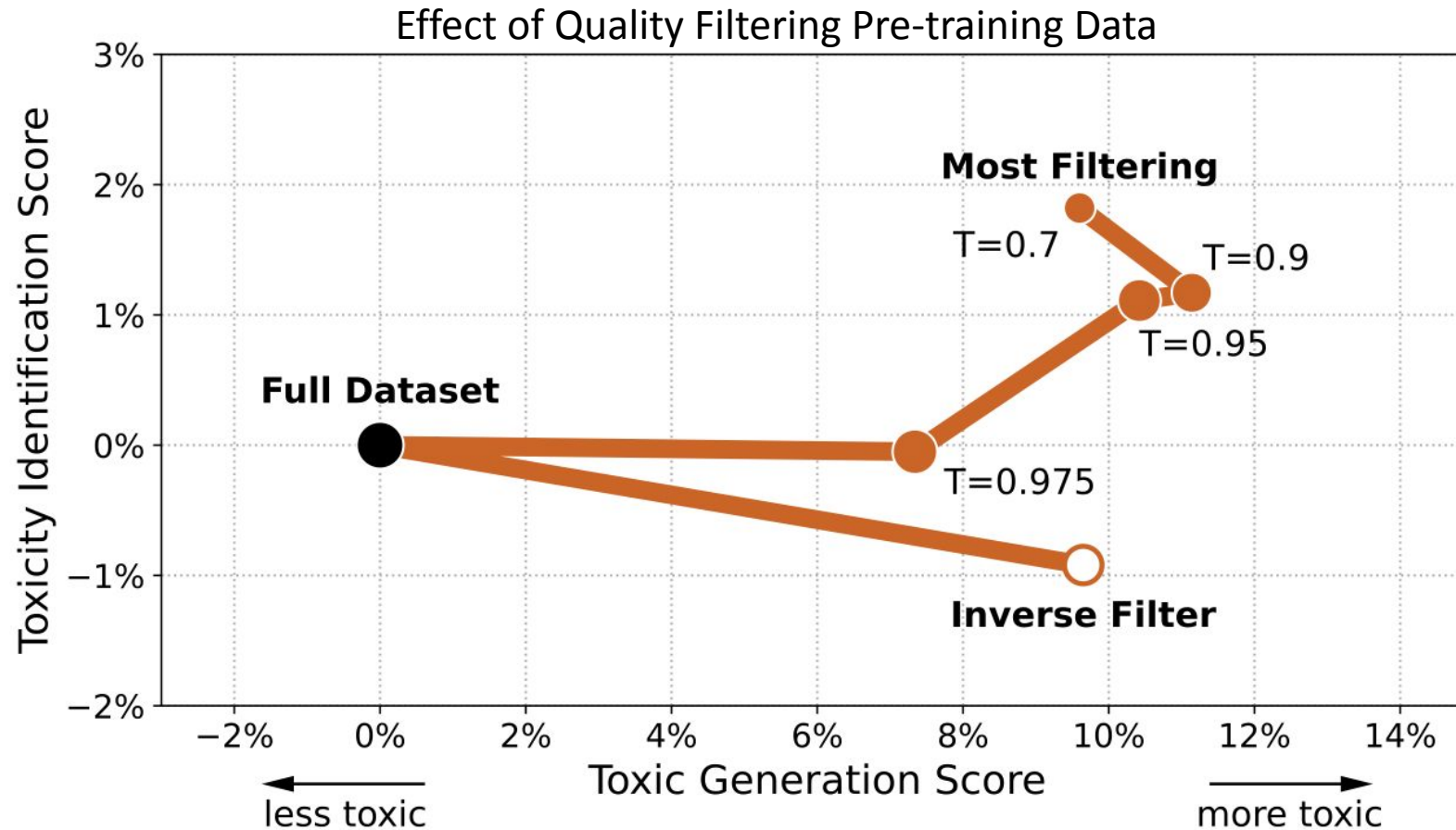
What is the effect of filtering out toxic or low-quality pre-training data on downstream performance?



What is the effect of filtering out toxic or low-quality pre-training data on downstream performance?



What is the effect of filtering out **low-quality** pre-training data on downstream performance?



What is the effect of filtering out toxic or low-quality pre-training data on downstream performance?

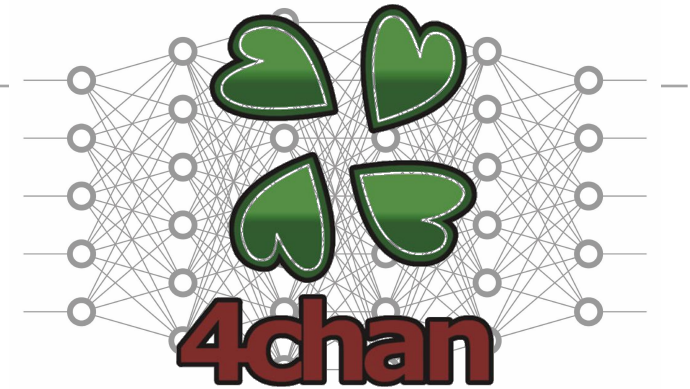
Take each model and finetune it on question answering benchmarks.

		QA domain					Mean
	Filter	Data	Wiki	Web	Acad	CS	
Baseline	Full Data	100%	0	0	0	0	0
Toxicity	Light (T=0.9)	95%	-2.2	-1.1	+0.2	+0.2	-0.7
	Heavy (T=0.5)	76%	-4.2	-2.4	-1.1	-3.5	-2.7
	Inverse	92%	+0.4	-1.4	+4.9	+2.7	+1.7
Quality	Light (T=0.975)	91%	+1.2	+0.7	+6.4	+6.1	+2.5
	Heavy (T=0.9)	73%	-0.3	+0.8	+0.8	+6.8	+1.2
	Inverse	73%	-5.0	-4.5	-2.7	-6.4	-3.1

Maybe we shouldn't do any filtering?



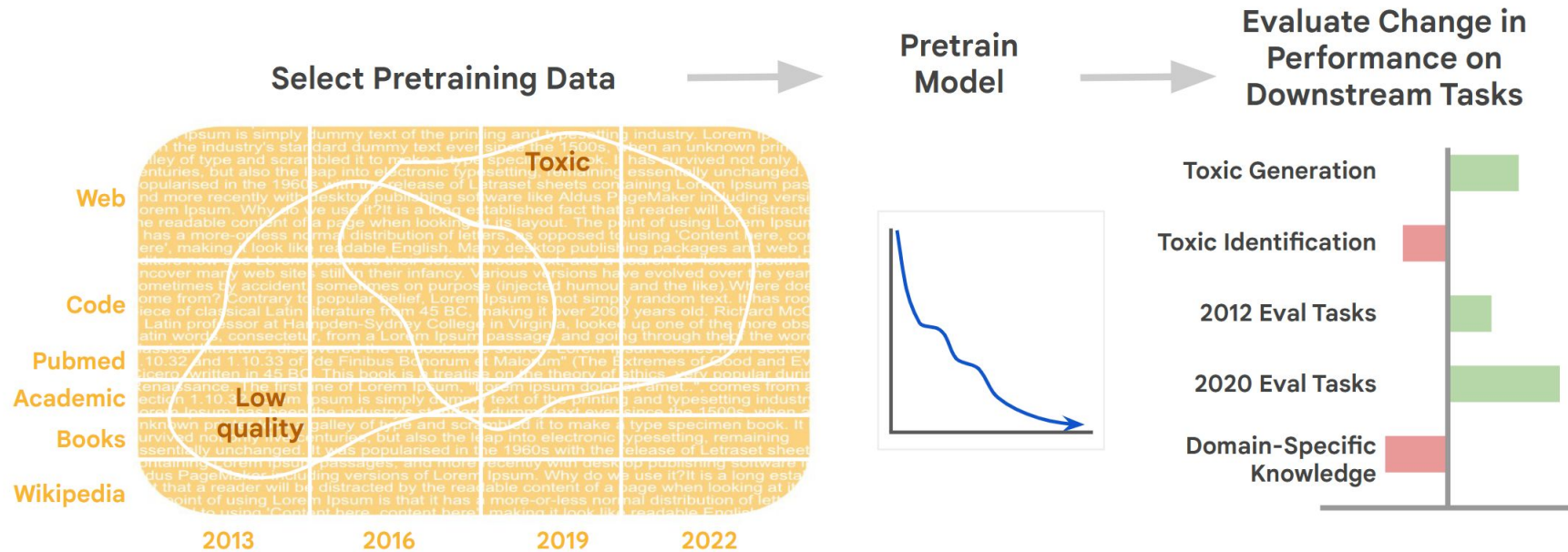
GPT4Chan controversy



[https://thegradient.pub/
gpt-4chan-lessons](https://thegradient.pub/gpt-4chan-lessons)

- Yannic Kilchner finetuned GPT-J on 4chan posts
 - Trained on subforum /pol/ known to contain racist, sexist, white supremacist, antisemitic, anti-Muslim, anti-LGBT views
- Trolled 4chan users with bots powered by his model
 - 30,000 posts over the span of a few days
- Faced massive criticism
 - initially hosted on Huggingface, was taken down quickly
- *Let's discuss...*
 - Was this an ethical model to train? Given that the dataset was publicly available?
 - Was deploying the bots on 4chan okay?
 - Are there any useful/positive applications of the model?

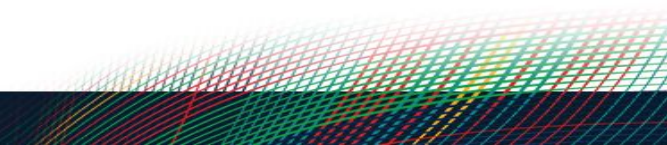
LLMs might need to see toxic data at pretraining



- [Longpre et al 2023](#) trained many LLMs with different amounts of toxicity in training data
- Showed **toxicity detection gets worse** the more toxic training data removed
- Makes sense: *you can't detect what you don't know about*

Leveraging LLM toxicity

- There are many applications of LLMs that require understanding the language of toxic/hate speech.
 - * Detection of hate speech [[Chiu et al 2022](#)]
 - * Counter speech generation [[Saha et al 2022](#), [Kim et al 2022](#), [Mun et al 2023](#)]
 - * Data creation: ToxiGen dataset [[Hartvigsen et al 2022](#)]
 - 300K subtly toxic and benign statements about minority groups
 - Control allows for subtle, hard-to-detect toxicity to improve classifiers
 - Statements indistinguishable from human-produced ones
- If we pre-train on toxic text, something must be done later in the pipeline to allow users to avoid seeing toxic content.



Making LLMs less biased/toxic/harmful

- Remove “undesirable” data from pre-training.
- Do finetuning on “desirable” data (e.g. instruction tuning).
- Finetune to bias the model toward outputs a human might classify as “desirable” (e.g. RLHF).
- Rejection sampling



Points of Intervention

1. Remove “undesirable” data from pre-training.
2. Do finetuning on “desirable” data (e.g. instruction tuning).
3. Finetune so as to bias the model toward outputs a human might classify as “desirable” (e.g. RLHF).
4. Rejection sampling at inference time

Points of Intervention

1. Remove “undesirable” data from pre-training.
2. Do finetuning on “desirable” data (e.g. instruction tuning).
3. Finetune so as to bias the model toward outputs a human might classify as “desirable” (e.g. RLHF).
4. Rejection sampling at inference time

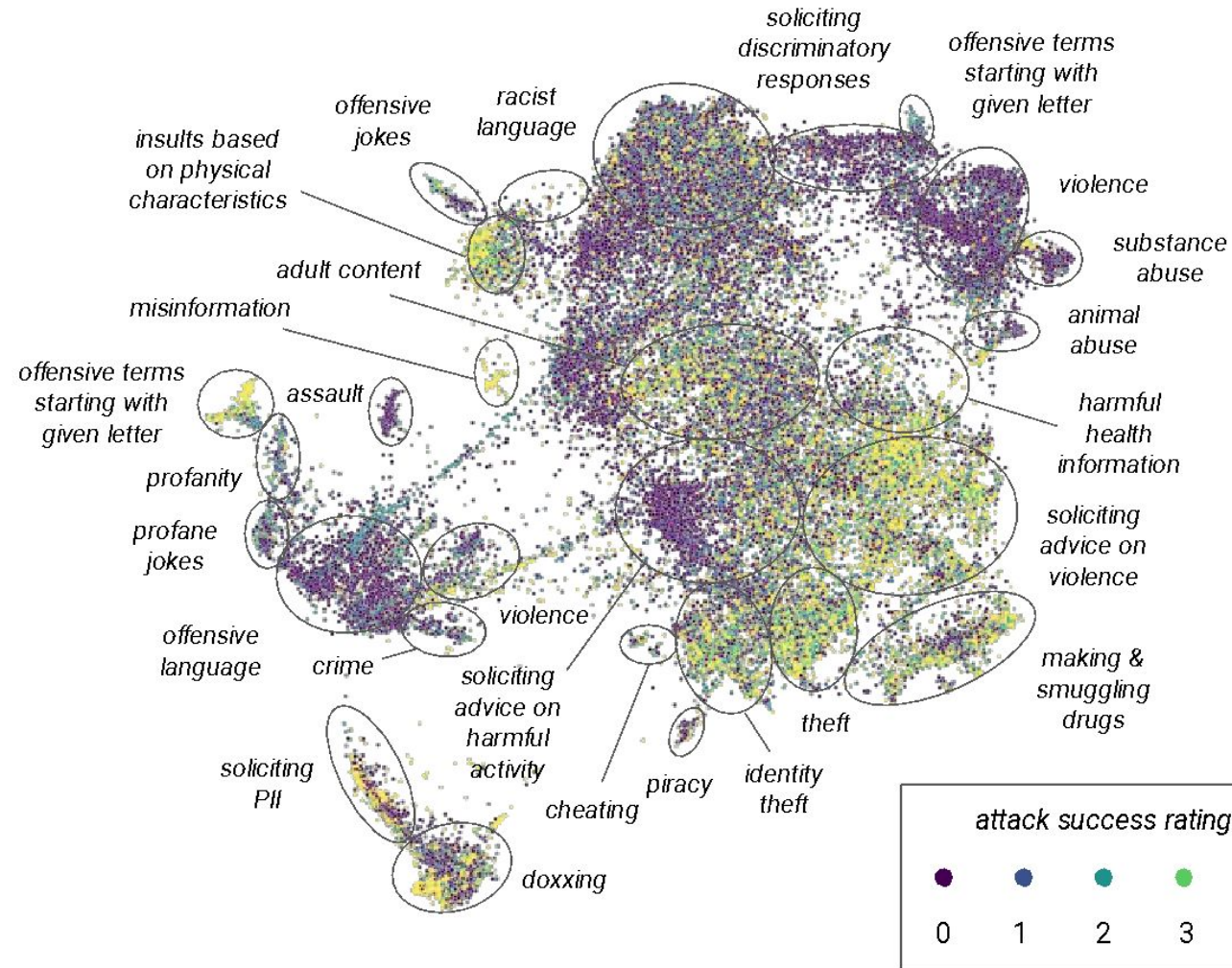
Rejection sampling

Main idea: generate more candidates than you need.

Keep the best one (where best is determined by various automatic classifier/rankers).



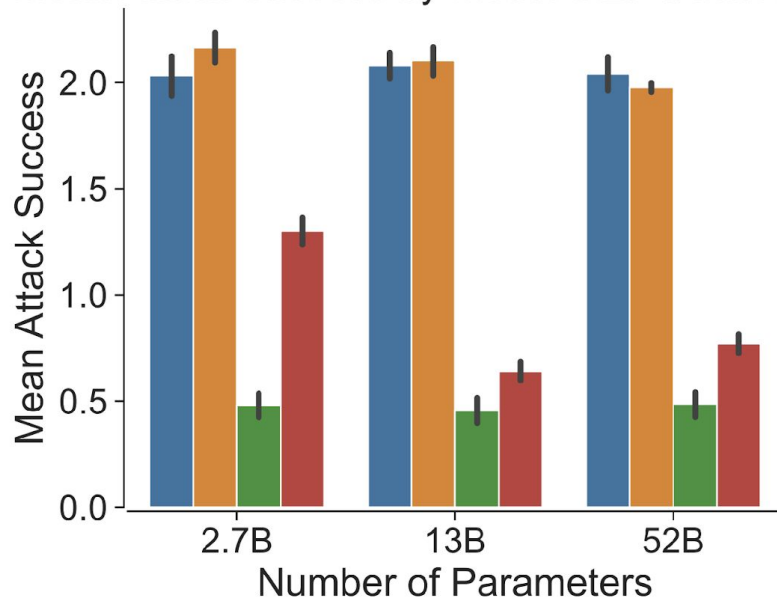
Red-teaming different methods of intervention



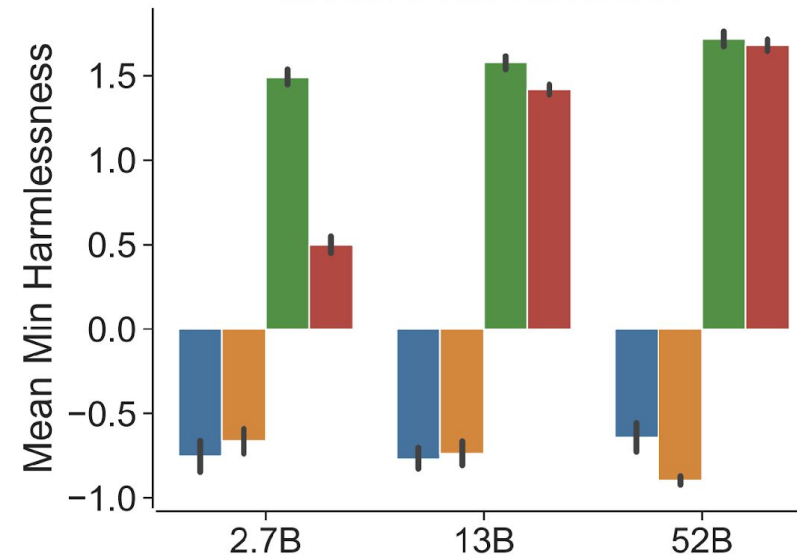
Red-teaming different methods of intervention



Mean Attack Success by Model Size & Intervention



Mean AI Harmlessness



LLM Safeguarding, Alignment & Controllable Generation

Recently: prompting for detoxification

- *Idea*: prompt the model to generate non-toxic language

“Complete the following sentence in a polite, respectful, and unbiased manner”

- InstructGPT is less toxic than GPT-3 on non-toxic input prompts from RTP

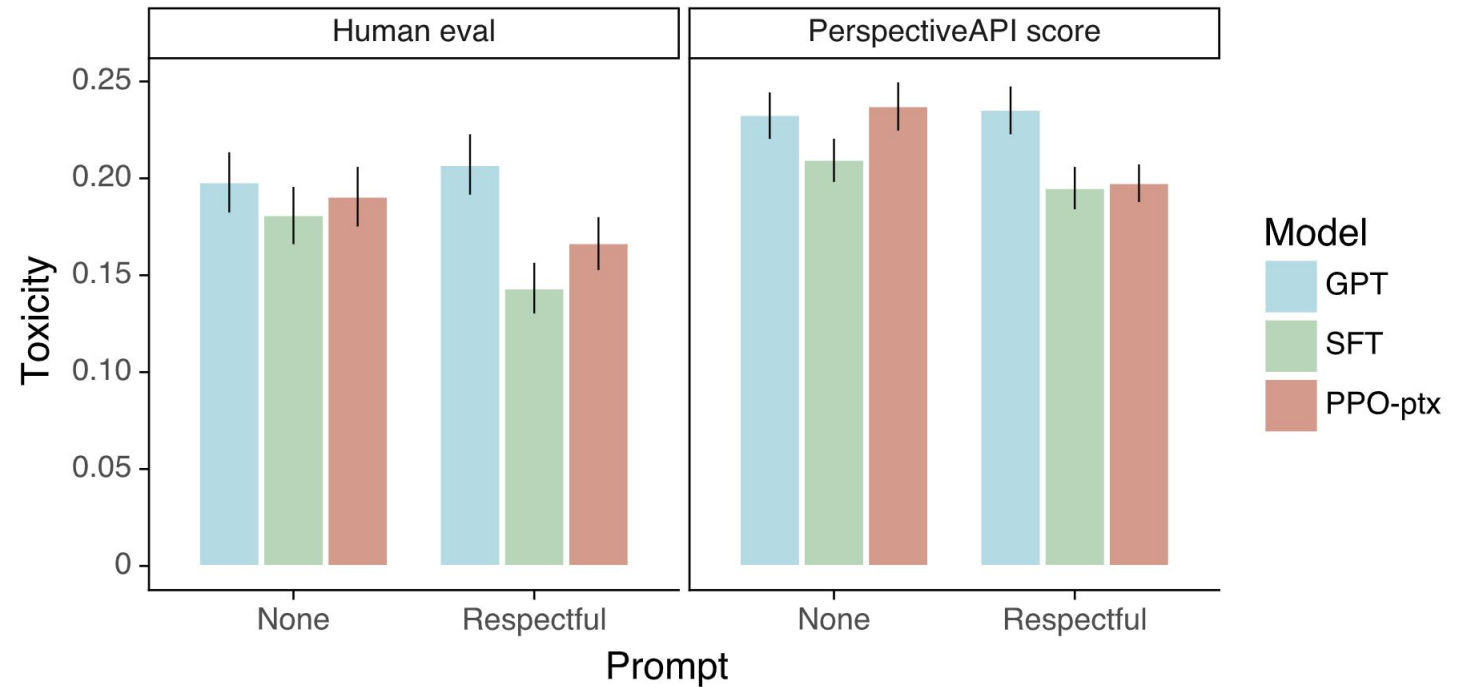
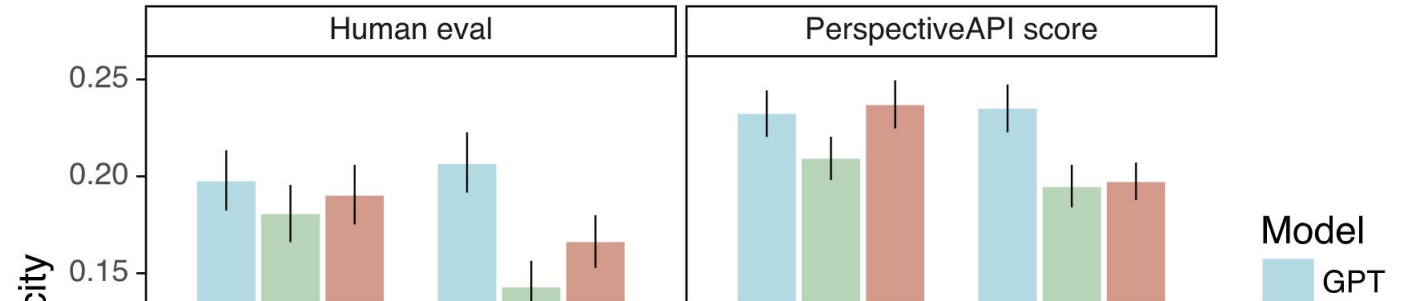


Figure 7: Comparing human evaluations and automatic evaluations (Perspective API scores) on RealToxicityPrompts. A total of 1,729 prompts were labeled for three different 175B models, both with and without "respectful" instructions. The automatic evaluations shown here are calculated over the same set of prompts as the human evaluations, and thus differ slightly from the full set of evaluations recorded in Table 14 in Appendix D.

Ouyang, et al. 2021. [“Training Language Models to Follow Instructions with Human Feedback”](#)

Recently: prompting for detoxification

- *Idea*: prompt the model to generate non-toxic language



Issue (not pictured): InstructGPT also generates more toxic language for toxic prompts (vs. GPT3)

- What does it mean to better follow instructions if the instruction is to be toxic?

Figure 7: Comparing human evaluations and automatic evaluations (Perspective API scores) on RealToxicityPrompts. A total of 1,729 prompts were labeled for three different 175B models, both with and without "respectful" instructions. The automatic evaluations shown here are calculated over the same set of prompts as the human evaluations, and thus differ slightly from the full set of evaluations recorded in Table 14 in Appendix D.

Ouyang, et al. 2021. "Training Language Models to Follow Instructions with Human Feedback"

Overview – LLM safeguarding

Safeguards from input prompt classification

- Topic-based filters
- Toxic content detection

Safeguards from instruction-tuning & RLHF

- Write demonstrations for refusing to answer
- RLHF models to prefer non-toxic generations

Safeguards at the output level

- Generate-then-classify
- Controllable text generation

Overview – LLM safeguarding

Safeguards from input prompt classification

- Topic-based filters
- Toxic content detection

Safeguards from instruction-tuning & RLHF

- Write demonstrations for refusing to answer
- RLHF models to prefer non-toxic generations

Safeguards at the output level

- Generate-then-classify
- Controllable text generation

These could all backfire!

RLHF safeguarding – assumptions

- PPO & family:

Train a preference classifier: which two generations is **better**?




LLM generates multiple outputs, classifier ranks outputs



RL is done to encourage more like “preferred output”

- What are some assumptions here? How could this backfire?

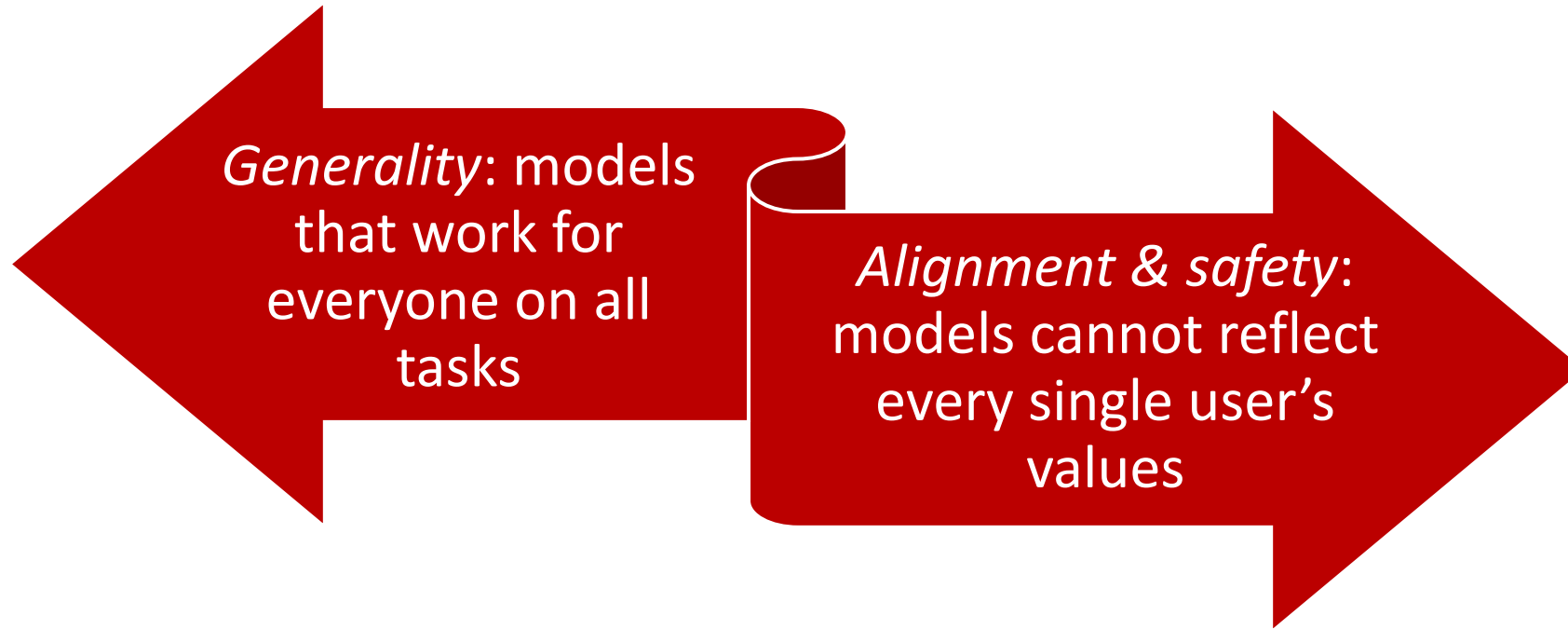
RLHF safeguarding – assumptions

- PPO & family:


```
graph LR; A[Train a preference classifier: which two generations is better?] --> B[LLM generates multiple outputs, classifier ranks outputs]; B --> C[RL is done to encourage more like “preferred output”];
```
- Big question: what does it mean for a generation to be **better/preferred**?
 - How to balance harmless and helpful?
 - E.g., help me create a poisonous drink.
 - What if people are biased or gameable?
 - E.g., people prefer certainty over uncertainty in answers to questions
 - Fundamental issue: cannot represent all values and cultures into one ranking.

Casper et al. 2023. “Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback.” *arXiv [cs.AI]*. arXiv. <http://arxiv.org/abs/2307.15217>

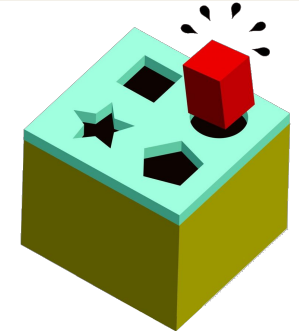
Big unresolved tension



What do y'all think we should do?

So... what can we do?

- Need to keep studying what models can and can't do, who they work for and don't work for
- Narrow scope of model users
 - Community-specific models (e.g., Mashakhane Initiative)
- **Specialize models' abilities / away from one-size-fits-all**
 - E.g., toxicity explanation generation model needs to generate stereotypes, but story generation models might not
- In line with many legislative efforts: legislate the application or task, not the model



If this is interesting to you... take my class

- 11-830: Ethics, Social Biases, and Positive Impact in Language Technologies (Spring 2024)

Ethical philosophies, AI alignment

Bias and Fairness in NLP

Toxicity and hate speech

LLM detoxification & alignment

Misinformation, manipulation, privacy

NLP for social good

and more fun discussions

The end

Questions?

Quiz:

In your opinion, what's the best path forward to preventing AI from making really negative impacts on society: modeling changes, better and more fine-grained RLHF, or AI regulation? Why?

