A Pretrainer's Guide to Training Data

Measuring the Effects of Data Age, Domain Coverage, Quality, & Toxicity

Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee,
Adam Roberts, Barret Zoph, Denny Zhou, Jason Wei, Kevin Robinson,
David Mimno, Daphne Ippolito

Today's Talk: *A Pretrainer's Guide*

1) **Introduction**
    - Data curation is everywhere
    - Experimental Setup
2) **Effects of Data Age**
3) **Effects of Quality & Toxicity Filters**
4) **Effects of Data Composition**
5) **Key Takeaways**

Pretraining data curation is *everywhere*

Curation Decisions	*Frequently Disclosed*	*Guided by Intuition*	*Meaningful Impact*
• Training data selection	❌	✅	✅
• Scrape timestamp	~	✅	✅✅
• Data cleaning	❌	✅	✅✅
• Language filtering	~	✅	✅✅
• PII removal	❌	❌	✅
• Deduplication	❌	❌	✅✅
• Toxicity / SafeURL filtering	❌	✅	✅
• Quality filtering	❌	✅	✅
• Sampling strategy	❌	~~✅~~ ❌	✅✅

Pretraining data curation is *everywhere*

Model	Represented Domains (%)									Filters		Data	
	Wiki	Web	Books	Dialog	Code	Acad	Pile	C4	M-L	Tox	Qual	Pub	Year
BERT	76		24				✗	✗			H	Part	2018
GPT-2		100					✗	✗			H	Part	2019
RoBERTa	7	90	3				✗	✔			H	Part	2019
XLNet	8	89	3				✗	✔			H	Part	2019
T5	<1	99					✗	✔		H	H	✔	2019
GPT-3	3	82	16				✗	✔	7%		C	✗	2021
GPT-J/Neo	1.5	38	15	4.5	13	28	✔	Part			C	✔	2020
GLaM	6	46	20	28			✗	✔			C	✗	2021
LaMDA	13	24		50	13		✔	✔	10%	C	C	✗	2021
AlphaCode					100		✗	✗			H	✗	2021
CodeGen	1	24	10	3	40	22	✔	Part			H	Part	2020
Chinchilla	1	65	10		4		✔	✔		H	C	✗	2021
Minerva	<1	1.5	<1	2.5	<1	95	✔	✔	<1%		C	✗	2022
BLOOM	5	60	10	5	10	10	✔	✔	71%	H	C	Part	2021
PaLM	4	28	13	50	5		✗	✔	22%		C	✗	2021
Galactica	1	7	1		7	84	✔	Part			H	Part	2022
LLAMA	4.5	82	4.5	2	4.5	2.5	Part	✔	4%		C	Part	2020

Data Choices & their Consequences

Largest empirical analysis of effects of pretraining data choice
(28x 1.5B pretraining runs)

Top 50 models are ~70% of downloads.

*Decisions w/o empiricism are **expensive**.*

Empirically quantify, validate, & challenge intuitions

Experimental setup

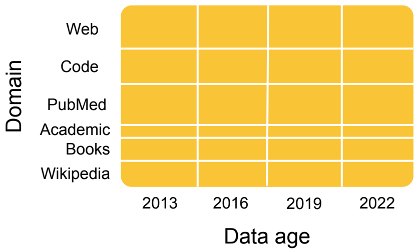
1. Full pretraining dataset

Experimental setup

1. Full pretraining dataset

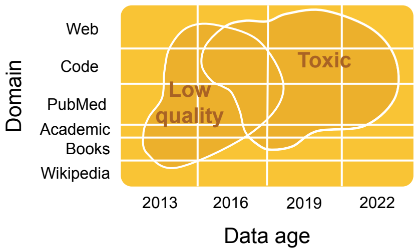
Experimental setup

1. Full pretraining dataset



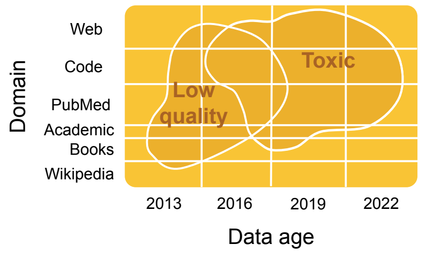
Experimental setup

1. Full pretraining dataset

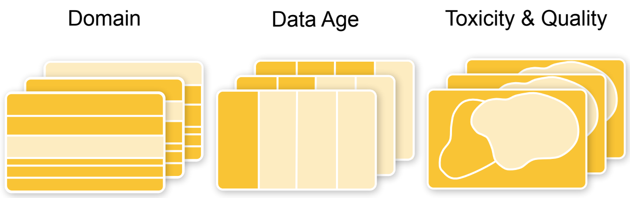


Experimental setup

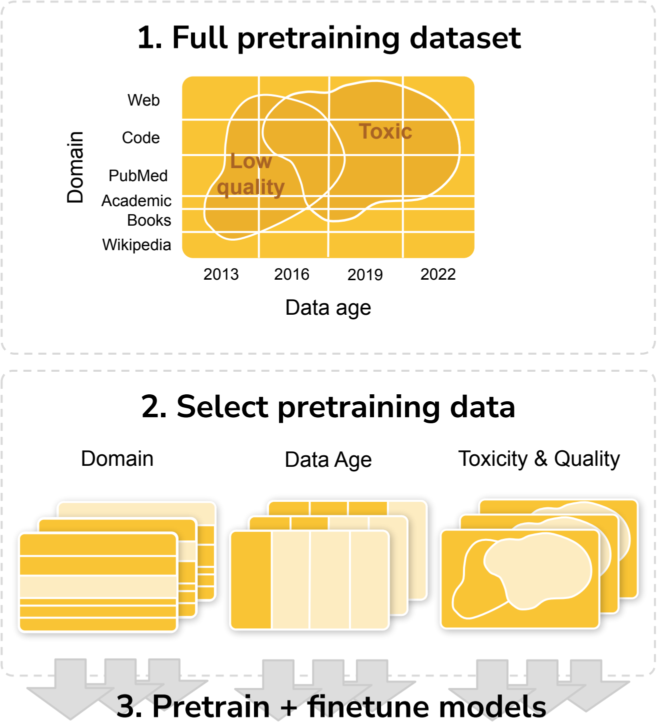
1. Full pretraining dataset



2. Select pretraining data

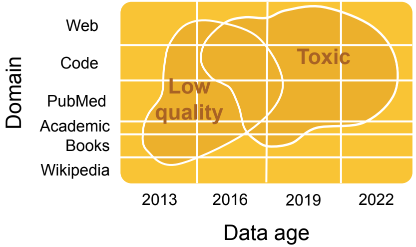


Experimental setup



Experimental setup

1. Full pretraining dataset

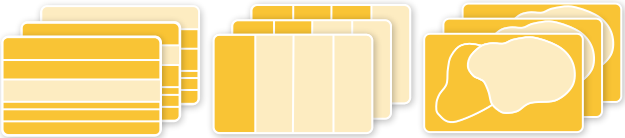


2. Select pretraining data

Domain

Data Age

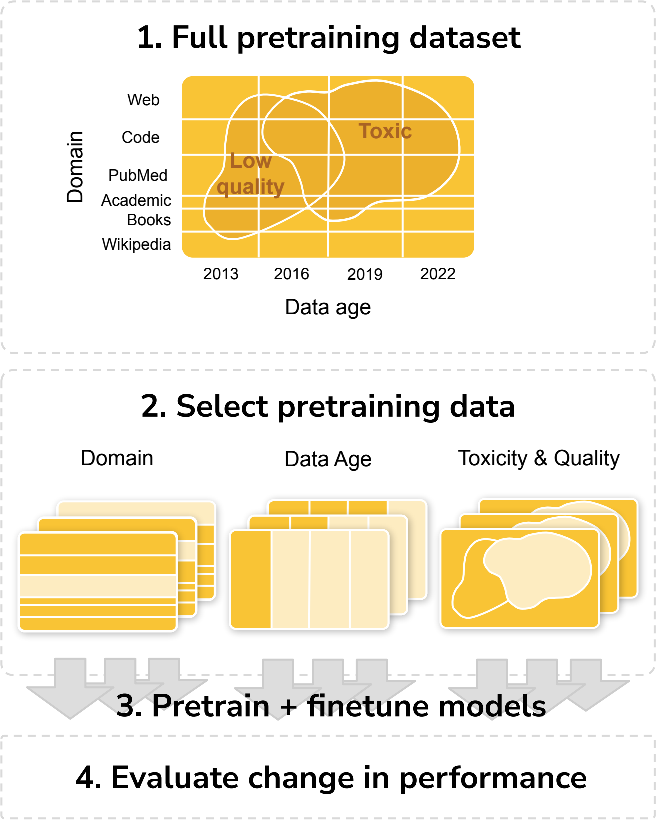
Toxicity & Quality



3. Pretrain + finetune models

4. Evaluate change in performance

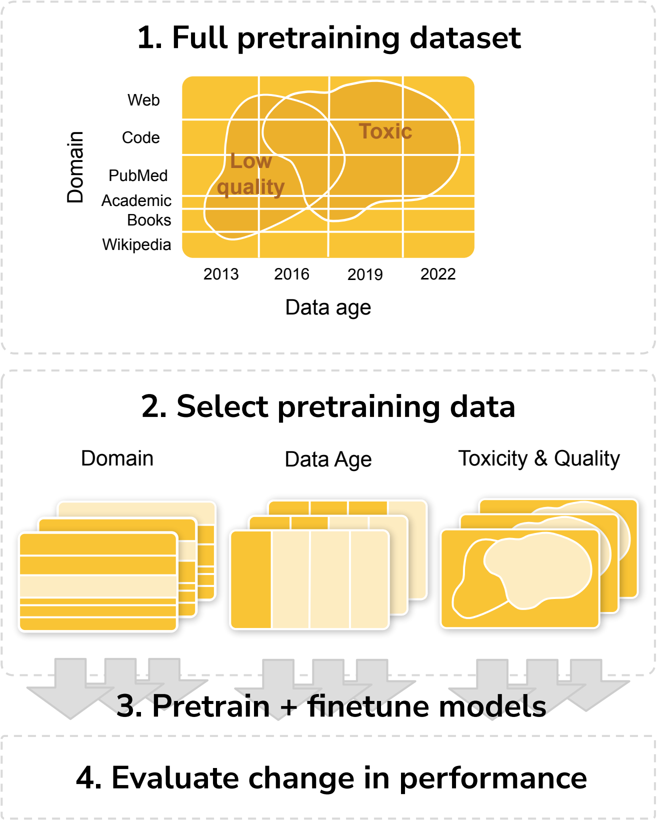
Experimental setup



Datasets:

- an unfiltered version of C4 (Raffel & al., 2020; Dodge & al., 2021)
- The Pile (Gao & al., 2020)

Experimental setup



Datasets:

- an unfiltered version of C4 (Raffel & al., 2020; Dodge & al., 2021)
- The Pile (Gao & al., 2020)

Model architectures: decoder-only autoregressive LM

- 1.5B-parameter
- 20M-parameter

Today’s Talk: *A Pretrainer’s Guide*

1) **Introduction**
   - Data curation is everywhere
   - Experimental Setup
2) **Effects of Data Age**
3) **Effects of Quality & Toxicity Filters**
4) **Effects of Data Composition**
5) **Key Takeaways**

Data age

Finetuning setting: *Temporal misalignment* between finetuning and evaluation datasets causes performance degradation.

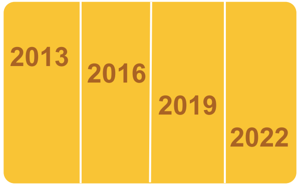
(Luu & al., 2021; Lazaridou & al., 2021, many others)

Question: Does mismatch in data age between pretraining and evaluation data cause performance degradation?

YES

Data age

1. Full pretraining dataset



2. Choose data age

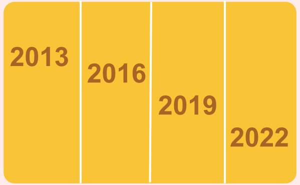
3. Pretrain models

4. Evaluate

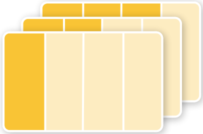
- temporal degradation on various tasks

Data age

1. Full pretraining dataset



2. Choose data age



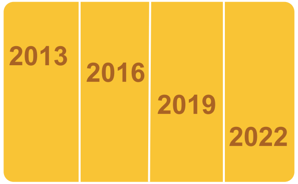
3. Pretrain models

4. Evaluate

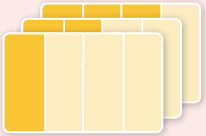
- temporal degradation on various tasks

Data age

1. Full pretraining dataset

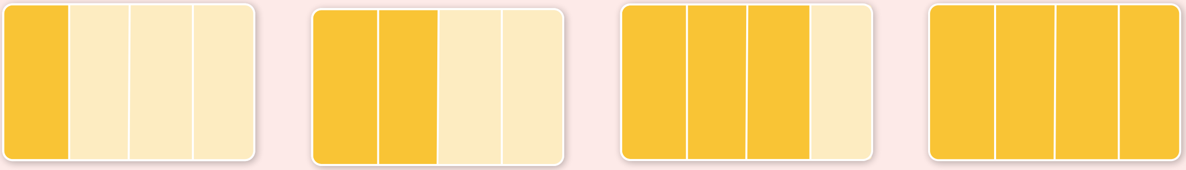


2. Choose data age



Four scrapes of the Common Crawl from different years:

2013 **2016** **2019** **2022**



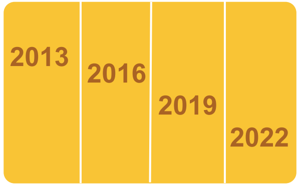
3. Pretrain models

4. Evaluate

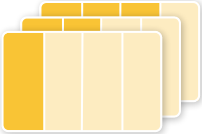
- temporal degradation on various tasks

Data age

1. Full pretraining dataset



2. Choose data age



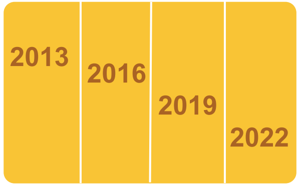
3. Pretrain models

4. Evaluate

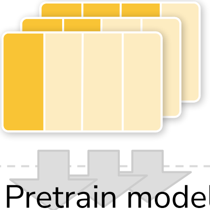
- temporal degradation on various tasks

Data age

1. Full pretraining dataset



2. Choose data age



3. Pretrain models

4. Evaluate

- temporal degradation on various tasks

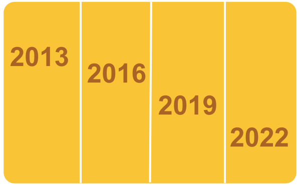
Temporal degradation (Luu & al., 2021)

Expected decrease in performance from one year of difference between pretraining and evaluation data

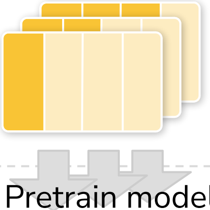
(averaged over evaluation years)

Data age

1. Full pretraining dataset



2. Choose data age



3. Pretrain models

4. Evaluate

- temporal degradation on various tasks

Temporal degradation (Luu & al., 2021)

Expected decrease in performance from one year of difference between pretraining and evaluation data

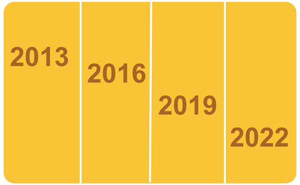
(averaged over evaluation years)

Datasets with year metadata:

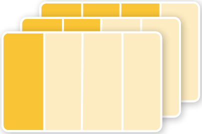
- **News:** PubCLS , NewSum
- **Twitter:** PoliAff, TwiERC
- **Science:** AIC

Data age

1. Full pretraining dataset



2. Choose data age



3. Pretrain models

4. Evaluate

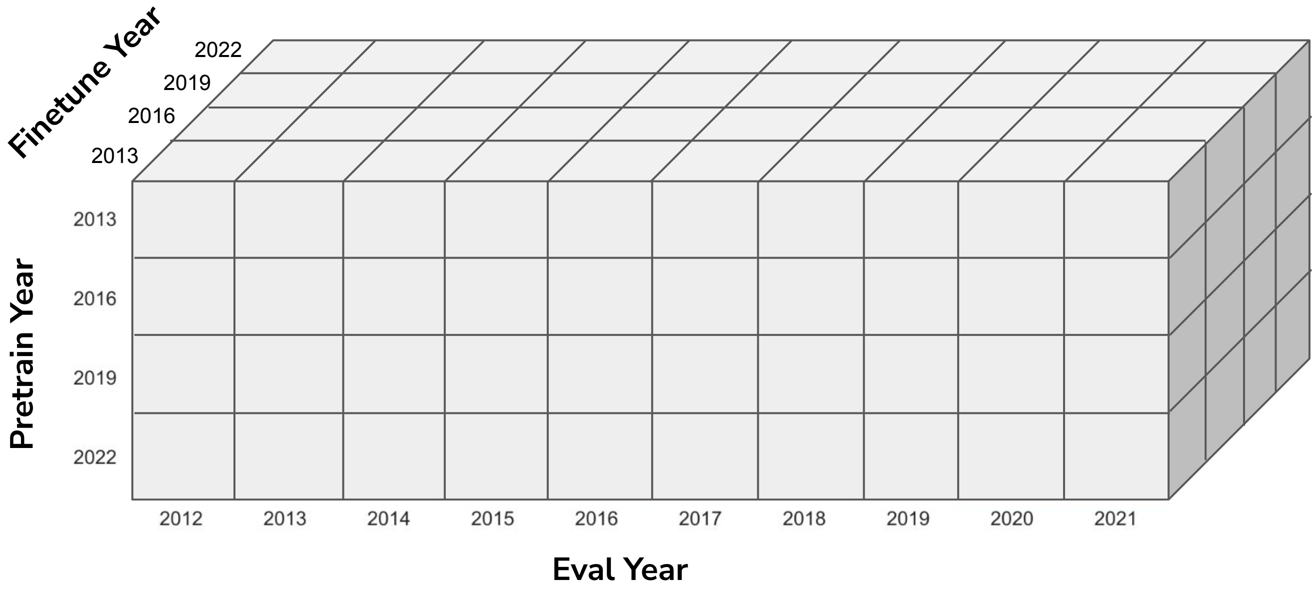
- temporal degradation on various tasks

1. Pretrain a model on each dataset
2. Finetune each model on downstream tasks (separately, by year)

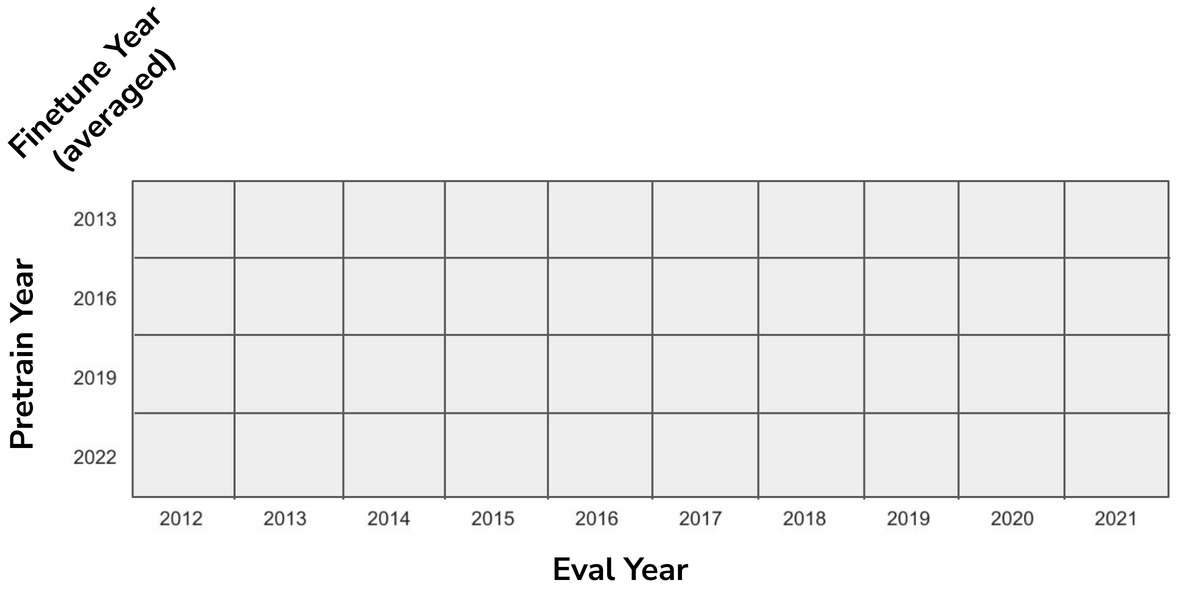
one model for every combination of pretraining year and finetuning year

Data age: illustrative dataset

Data age: illustrative dataset



Data age: illustrative dataset

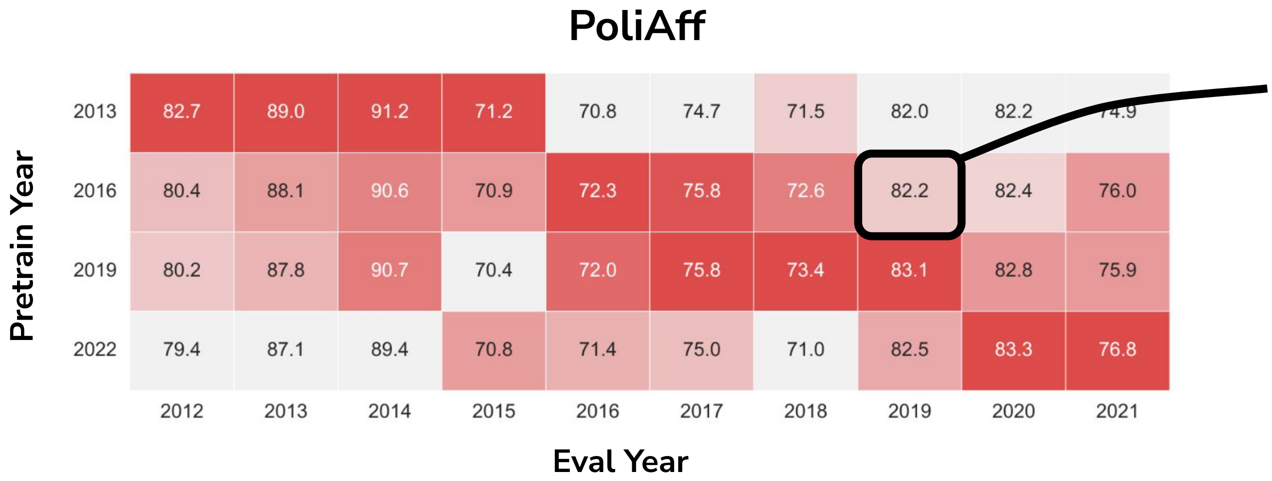


Data age: illustrative dataset

PoliAff

Pretrain Year \ Eval Year	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
2013	82.7	89.0	91.2	71.2	70.8	74.7	71.5	82.0	82.2	74.9
2016	80.4	88.1	90.6	70.9	72.3	75.8	72.6	82.2	82.4	76.0
2019	80.2	87.8	90.7	70.4	72.0	75.8	73.4	83.1	82.8	75.9
2022	79.4	87.1	89.4	70.8	71.4	75.0	71.0	82.5	83.3	76.8

Data age: illustrative dataset



accuracy after pretraining on 2016 data, evaluating on 2019 data

(averaged across finetuning years)

Data age: illustrative dataset

PoliAff

Pretrain Year \ Eval Year	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021
2013	82.7	89.0	91.2	71.2	70.8	74.7	71.5	82.0	82.2	74.9
2016	80.4	88.1	90.6	70.9	72.3	75.8	72.6	82.2	82.4	76.0
2019	80.2	87.8	90.7	70.4	72.0	75.8	73.4	83.1	82.8	75.9
2022	79.4	87.1	89.4	70.8	71.4	75.0	71.0	82.5	83.3	76.8

Takeaway:

1. Accuracy is higher when pretraining and eval year are closer in time *(even after finetuning)*

Full results in the paper!

Data age: one year difference between training/eval

Domain	Task	Finetuning TD
News	PubCLS	5.63
	NewSum	2.91
Twitter	PoliAff	4.93
	TwiERC	0.53
Science	AIC	0.24
	Mean	**2.84**

We first reproduce temporal misalignment between finetuning and eval datasets, as in Luu & al., 2021.

Data age: one year difference between training/eval

Domain	Task	Finetuning TD	Pretraining TD
News	PubCLS	5.63	0.59
	NewSum	2.91	0.73
Twitter	PoliAff	4.93	0.28
	TwiERC	0.53	0.23
Science	AIC	0.24	0.23
	Mean	**2.84**	**0.41**

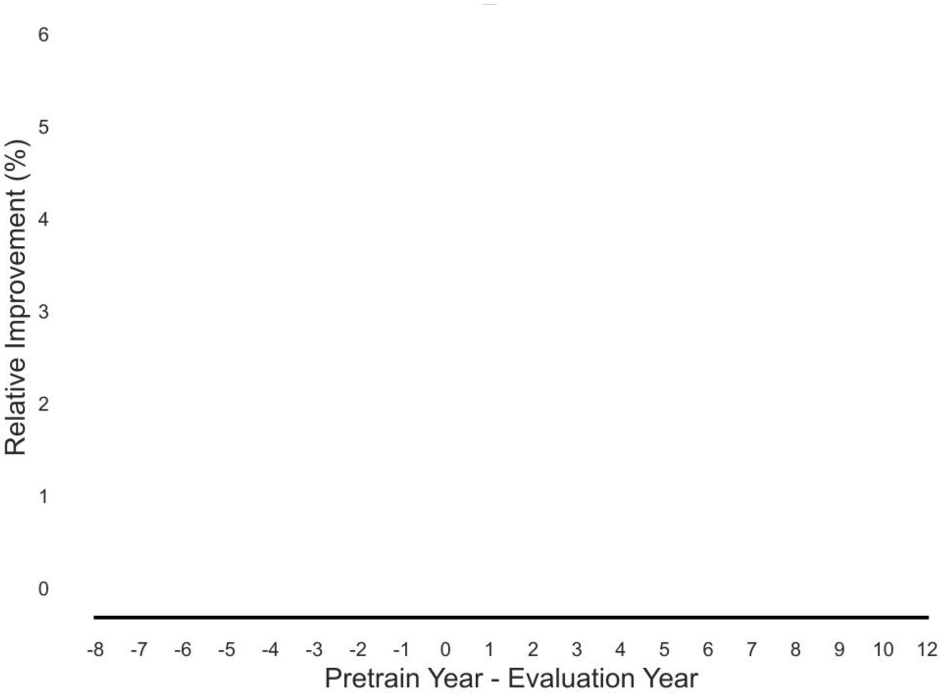
Data age: one year difference between training/eval

Domain	Task	Finetuning TD	Pretraining TD
News	PubCLS	5.63	0.59
	NewSum	2.91	0.73
Twitter	PoliAff	4.93	0.28
	TwiERC	0.53	0.23
Science	AIC	0.24	0.23
	Mean	**2.84**	**0.41**

Takeaway:

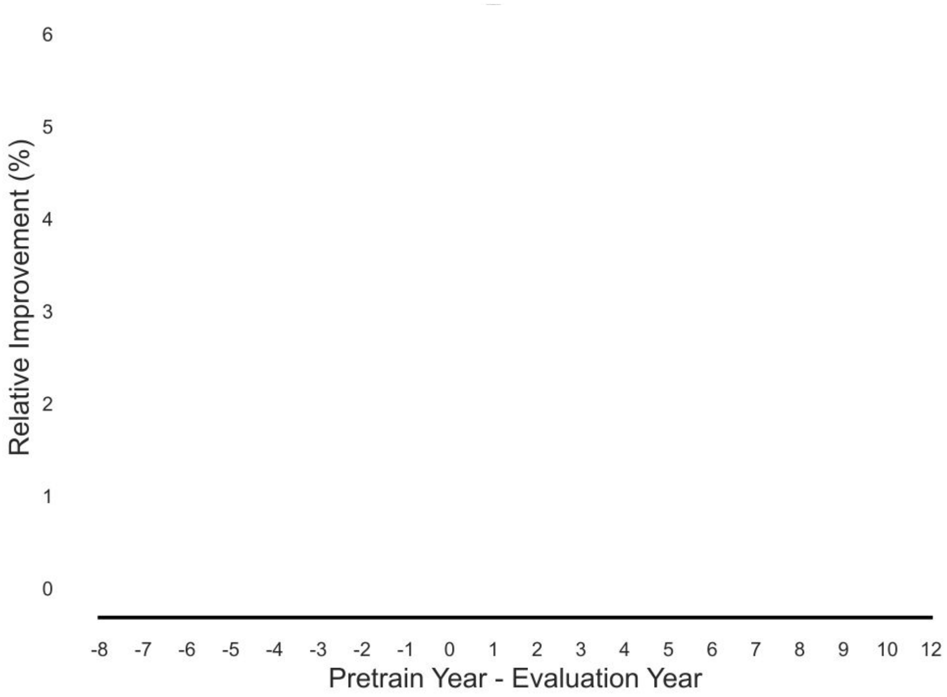
Temporal degradation due to pretraining is significant and persistent across domains.

Data age



6
5
4
3
2
1
0
Relative Improvement (%)
-8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 12
Pretrain Year - Evaluation Year

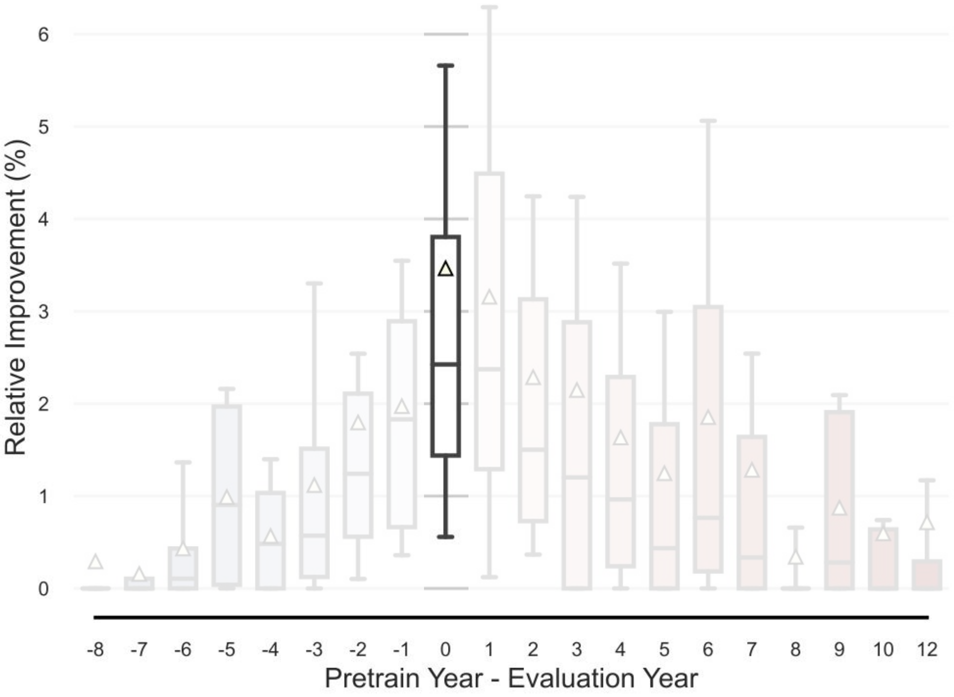
Data age



Each result is associated with:

- dataset
- pretraining year
- finetuning year
- evaluation year

Data age

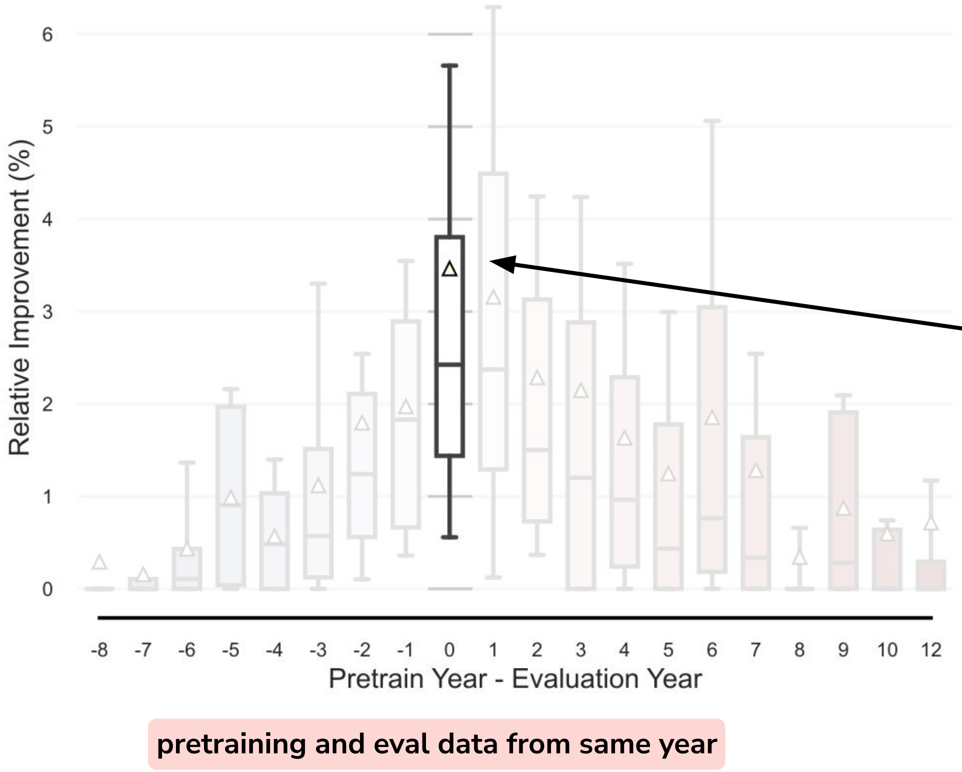


pretraining and eval data from same year

Each result is associated with:

- dataset
- pretraining year
- finetuning year
- evaluation year

Data age



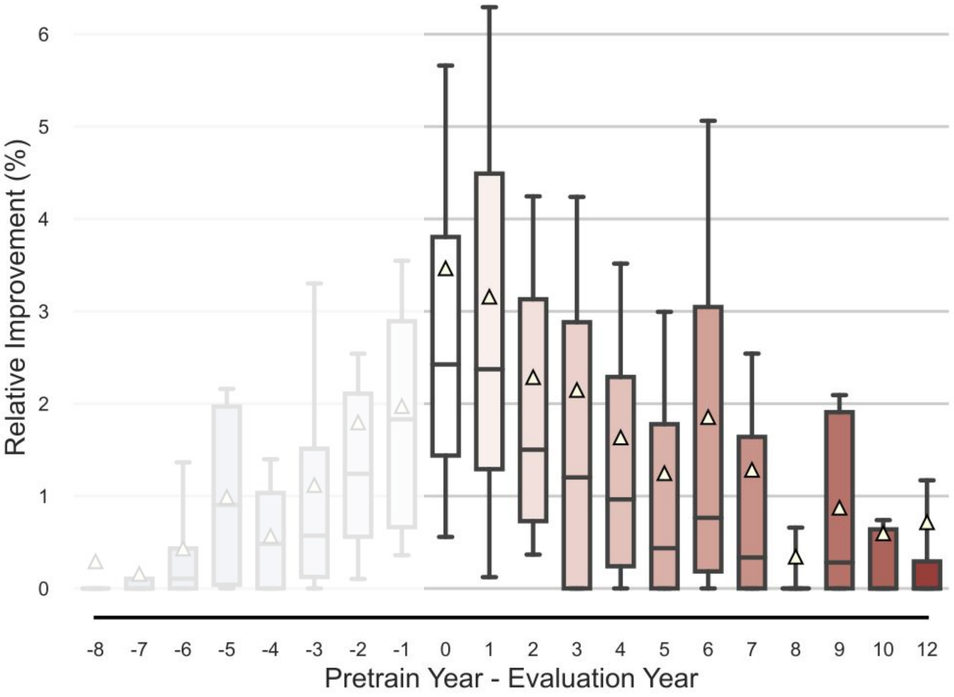
Each result is associated with:

- dataset
- pretraining year
- finetuning year
- evaluation year

1. Result included if pretraining data and eval data are from the same year

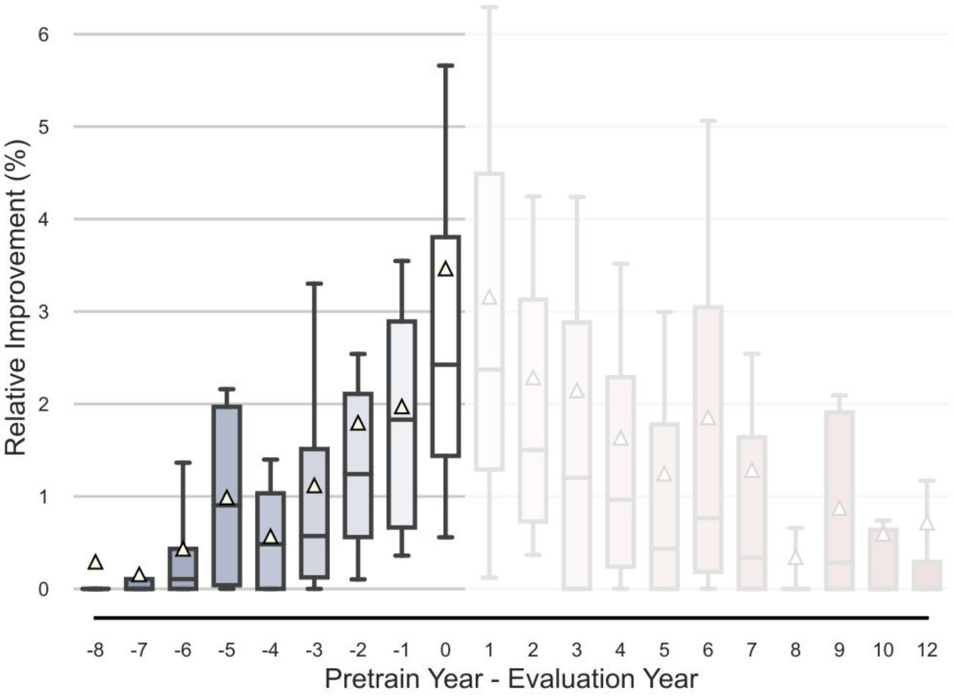
2. Each result is compared to worst performance on that dataset's eval year

Data age



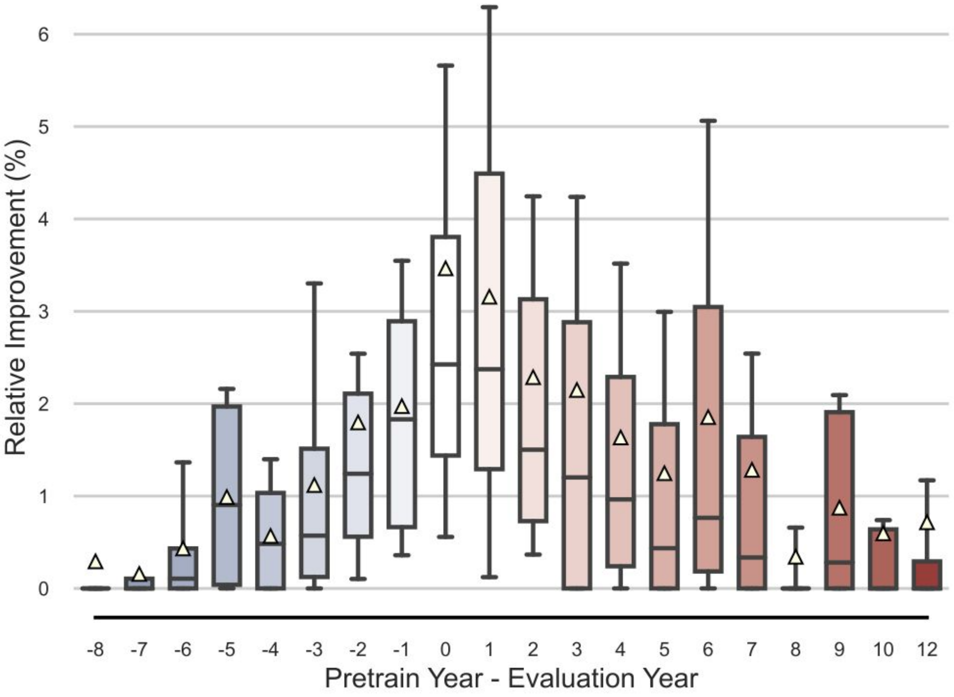
pretraining data newer than eval data

Data age



eval data newer than pretraining data

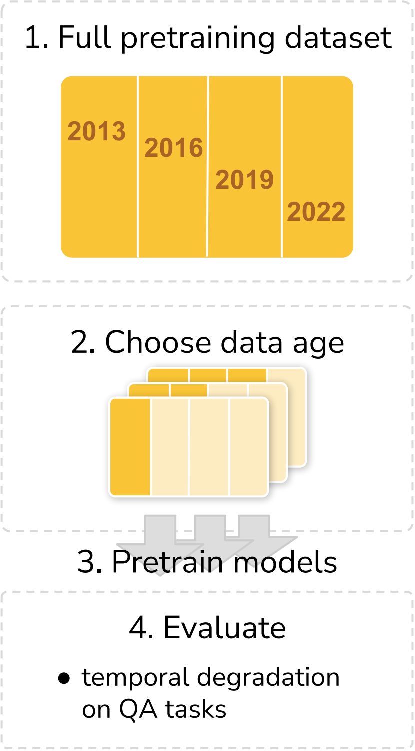
Data age



Takeaways:

1. Models and datasets become stale.
2. Temporal degradation persists even after finetuning.
3. Temporal degradation happens faster when evaluating old models on new benchmarks.

Data age: **recommendations**



1. Release age distributions for pretraining data.
   Stale pretraining data is not overcome by finetuning.

2. **In the paper:** the effects of pretraining temporal misalignment are stronger for larger models than smaller models.

Content filtering: toxicity and quality

Broad goals:
- Best downstream performance across tasks
- Prevent models from generating toxic text
- Identify toxic text

Quality filters in practice: Almost all models filter for some notion of quality

Toxicity filters in practice: T5, LaMDA, Chinchilla remove pretraining documents that might be toxic. Most models don't filter or don't disclose filtering.

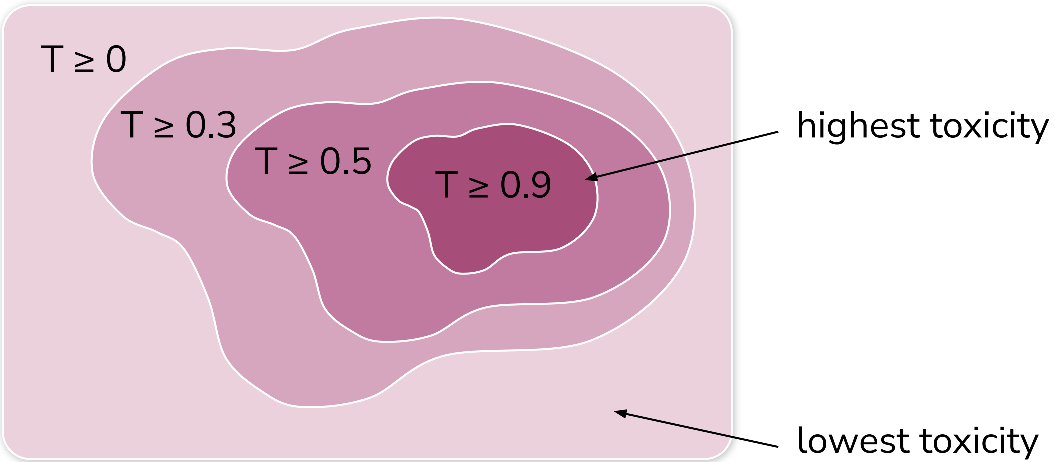
Question: How does filtering pretraining documents based on toxicity and quality actually affect downstream tasks?

Toxicity

Toxicity: How do we measure toxicity?

Perspective API: every document gets a score from 0 (nontoxic) to 1 (toxic)

This is just one possible operationalization, with many downsides.



https://perspectiveapi.com

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

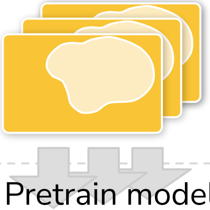
Toxicity

1. Full pretraining dataset

Baseline: no toxicity filtering

Toxicity threshold ≤ 1.0

2. Vary filter threshold



3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

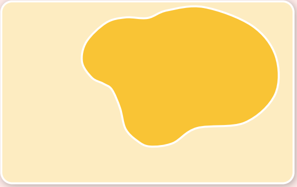
3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

Light filtering (toxicity threshold ≤ 0.9)
Filter out documents with highest toxicity

Heavy filtering (toxicity threshold ≤ 0.3)
Filter out documents with at least some toxicity



Inverse toxicity filter
Filter out *least* toxic documents

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

1. Pretrain a model on each dataset
2. Finetune each model on downstream tasks (separately)

4. Evaluate

- toxic generation
- toxicity identification

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

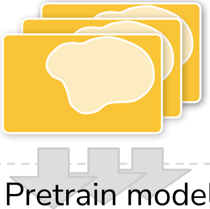
4. Evaluate

- toxic generation
- toxicity identification

Toxicity

1. Full pretraining dataset

2. Vary filter threshold



3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

Toxic generation: Is generated text considered toxic?

Datasets:

- RealToxicityPrompts (Gehman & al., 2020)
- RepBias (Chowdhery & al., 2022)

Toxicity

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

Toxic generation: Is generated text considered toxic?

Datasets:

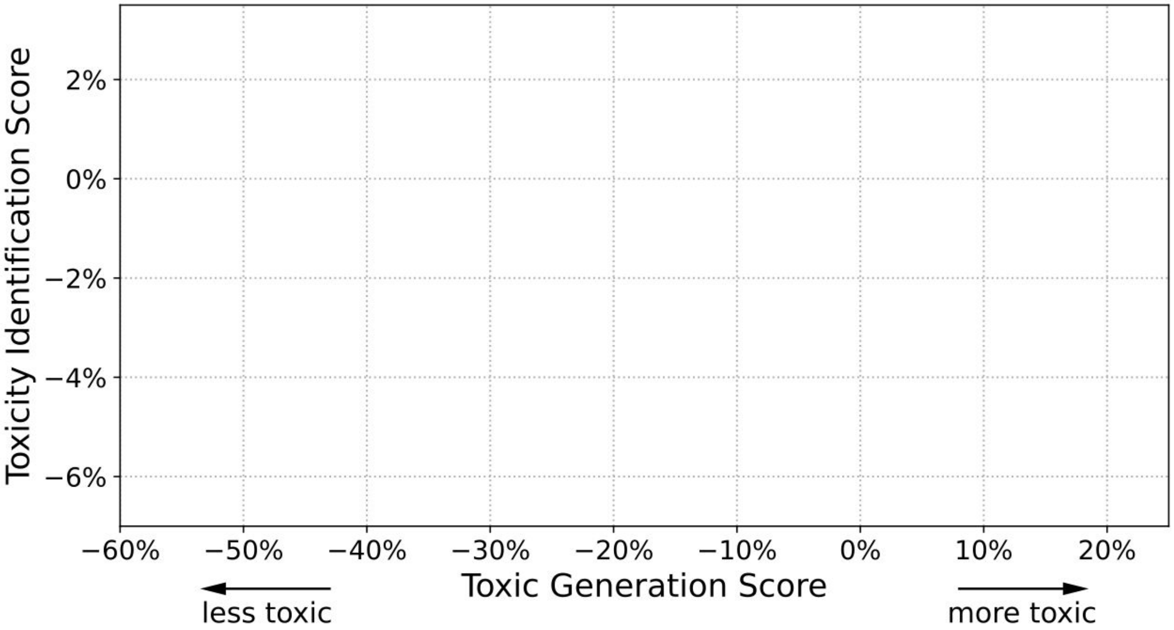
- RealToxicityPrompts (Gehman & al., 2020)
- RepBias (Chowdhery & al., 2022)

Toxicity identification: Can the model classify text as toxic?

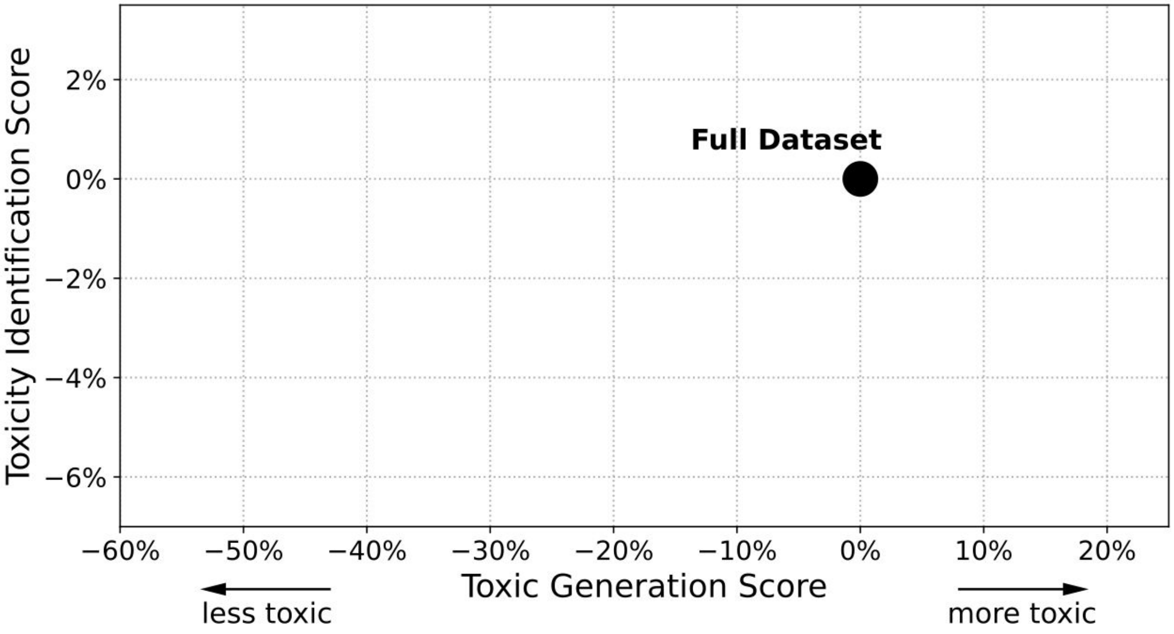
Datasets:

- Social Bias Frames (Sap & al., 2020)
- DynaHate (Vidgen & al., 2021)
- Toxigen (Hartvigsen & al., 2022)

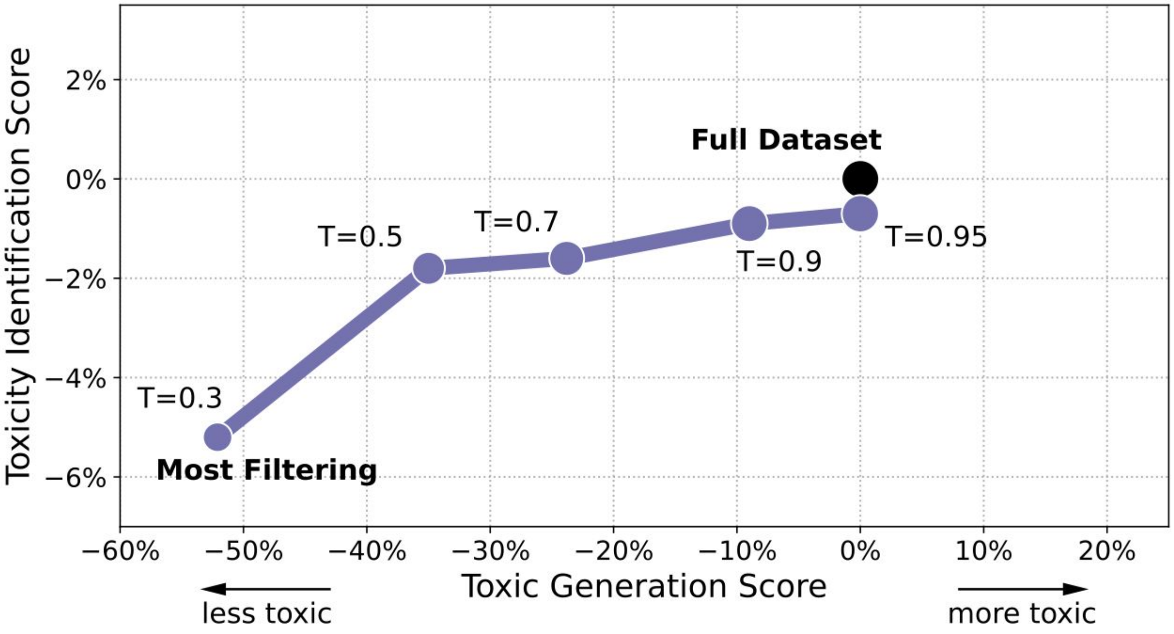
Toxicity: tradeoff between identification and generation



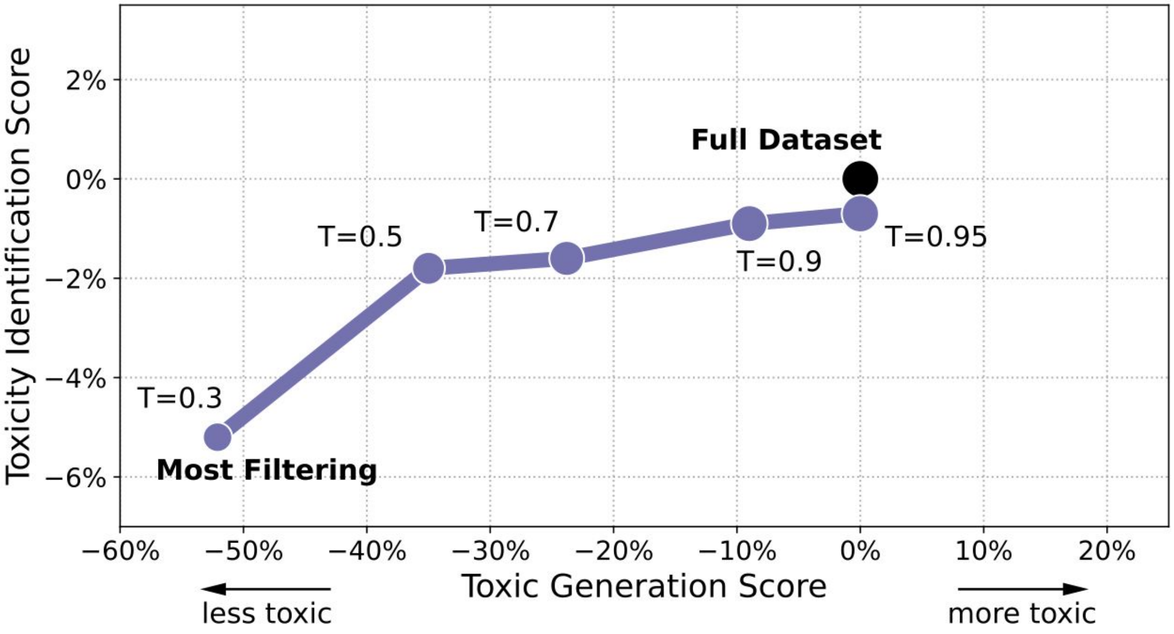
Toxicity: tradeoff between identification and generation



Toxicity: tradeoff between identification and generation



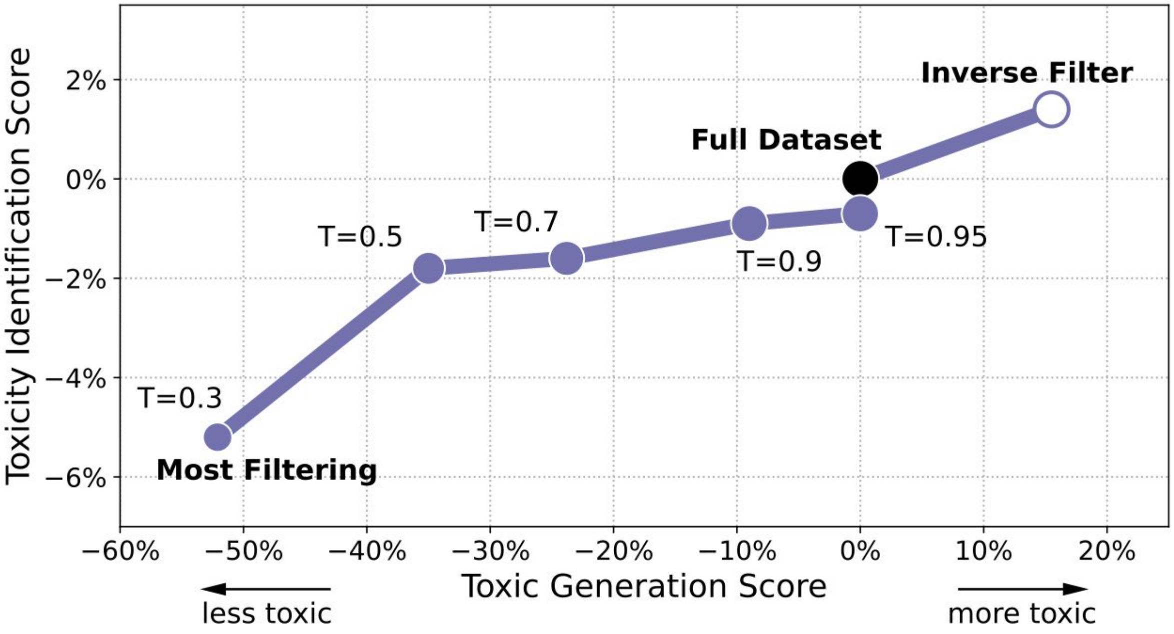
Toxicity: tradeoff between identification and generation



Takeaways:

1. Toxicity filtering reduces toxic generation at the cost of decreased identification.

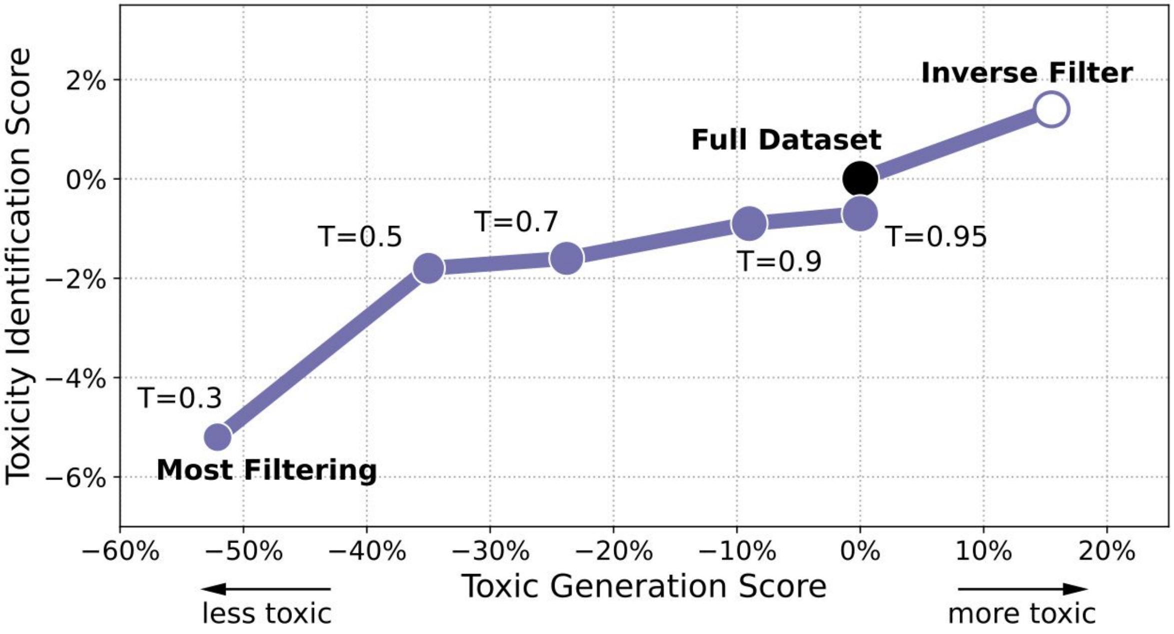
Toxicity: tradeoff between identification and generation



Takeaways:

1. Toxicity filtering reduces toxic generation at the cost of decreased identification.

Toxicity: tradeoff between identification and generation



Takeaways:

1. Toxicity filtering reduces toxic generation at the cost of decreased identification.

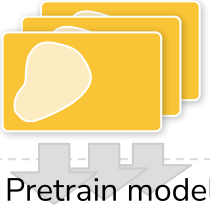
2. If the goal is to identify toxic text, then training on toxic data is more effective.

What about quality filtering?

Quality filtering’s effect on toxicity evals

1. Full pretraining dataset

2. Vary filter threshold

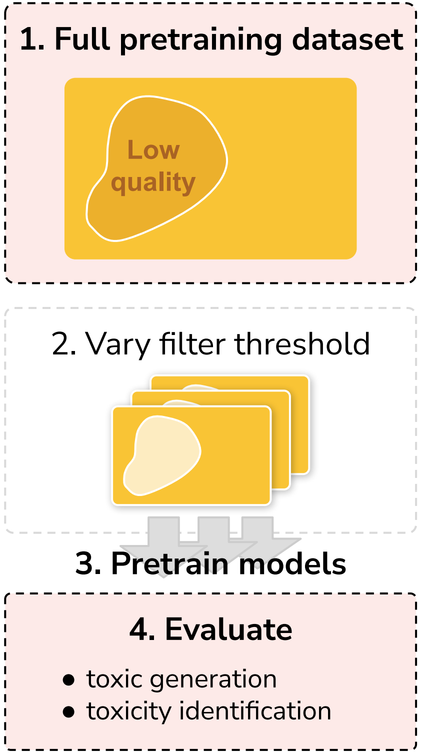


3. Pretrain models

4. Evaluate

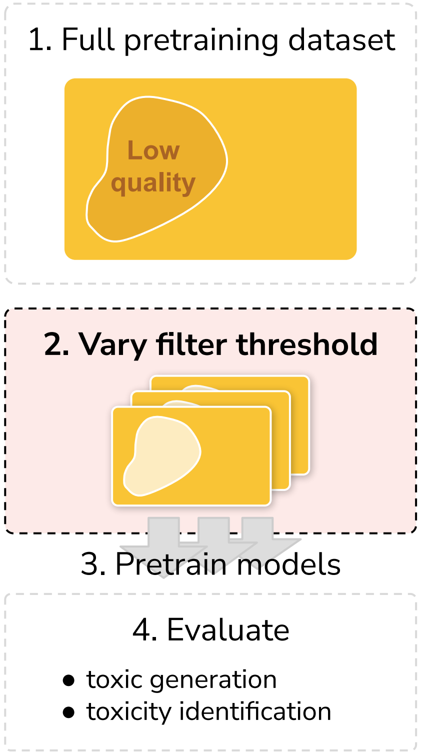
- toxic generation
- toxicity identification

Quality filtering’s effect on toxicity evals



Same setup, baseline, and evals as toxicity filtering

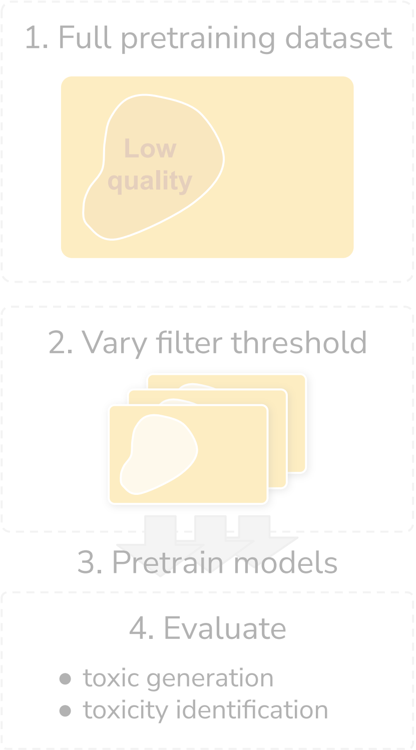
Quality filtering's effect on toxicity evals



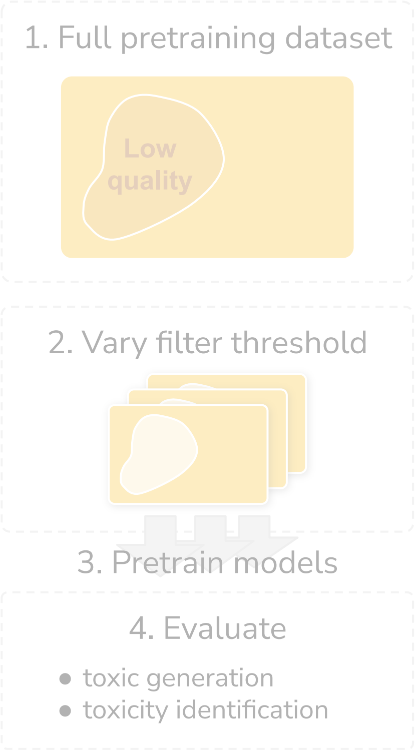
Same setup, baseline, and evals as toxicity filtering

Filter by quality instead of toxicity

Quality: How do we measure “quality”?



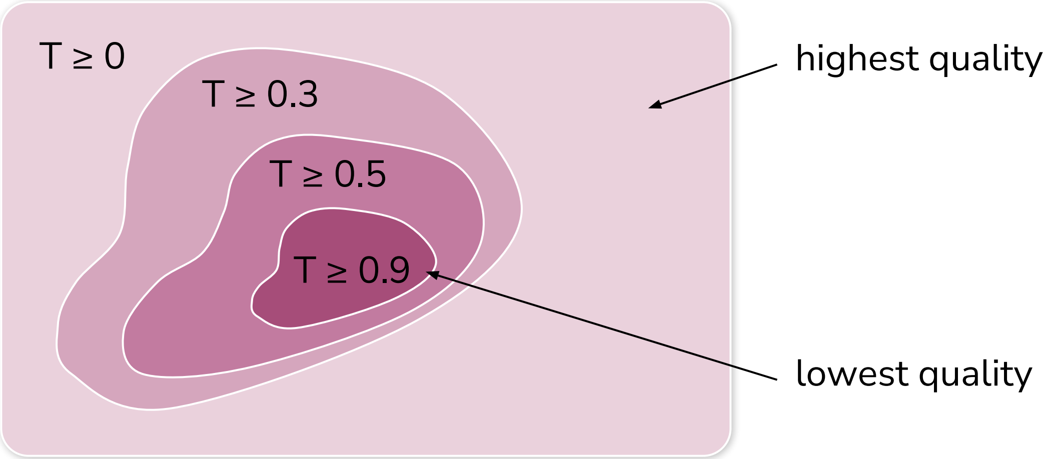
Quality: How do we measure "quality"?



GLaM/PaLM classifier:

(Du & al., 2022)
(Chowdhery & al., 2022)
GPT-3, probably GPT-4

- Wikipedia + books are high quality
- every document gets a score from 0 (high quality) to 1 (low quality)

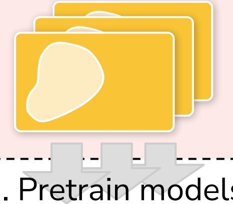


This is an existing operationalization, with many downsides.

Quality filtering's effect on toxicity evals

1. Full pretraining dataset

2. Vary filter threshold



3. Pretrain models

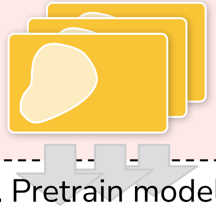
4. Evaluate

- toxic generation
- toxicity identification

Quality filtering's effect on toxicity evals

1. Full pretraining dataset

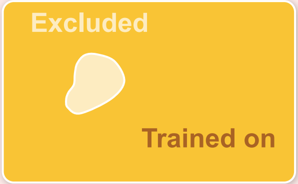
2. Vary filter threshold



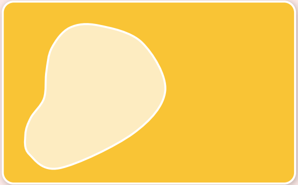
3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification



Light filtering (quality threshold ≤ 0.95)
Filter out documents with lowest quality



Heavy filtering (quality threshold ≤ 0.7)
Filter out all but the highest-quality documents

Inverse quality filter
Filter out *highest* quality documents

Quality filtering’s effect on toxicity evals

1. Full pretraining dataset

2. Vary filter threshold

3. Pretrain models

4. Evaluate

- toxic generation
- toxicity identification

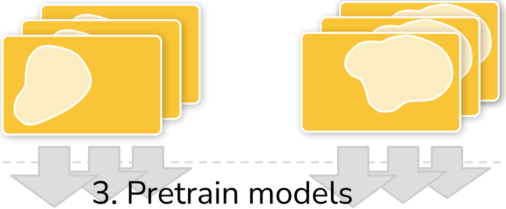
In the paper: quality filtering improves toxicity identification

Toxicity and quality filters: **downstream performance**

1. Full pretraining dataset

2. Quality filtering

Toxicity filtering

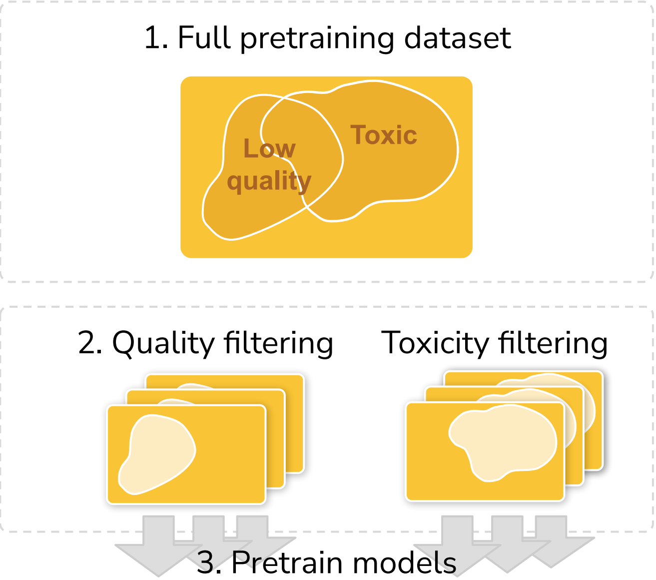


3. Pretrain models

4. Evaluate

- QA tasks, grouped by domain

Toxicity and quality filters: **downstream performance**



4. Evaluate

- QA tasks, grouped by domain

Question answering:

27 QA tasks from:
- MRQA (Fisch & al., 2019)
- UnifiedQA (Khashabi & al., 2020)

Categorized by domain:
- Wiki
- Web
- Academic
- Commonsense

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
Baseline	Full Data	100%	0	0	0	0	0

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
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

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
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

Takeaways:

1. Toxicity filtering hurts performance across domains.

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
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

Takeaways:

1. Toxicity filtering hurts performance across domains.

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
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

Takeaways:

1. Toxicity filtering hurts performance across domains.

Toxicity and quality filters: **downstream performance**

			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
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

Takeaways:

1. Toxicity filtering hurts performance across domains.
2. Quality filtering improves performance across most domains, despite removing data.

Toxicity and quality filters: **downstream performance**

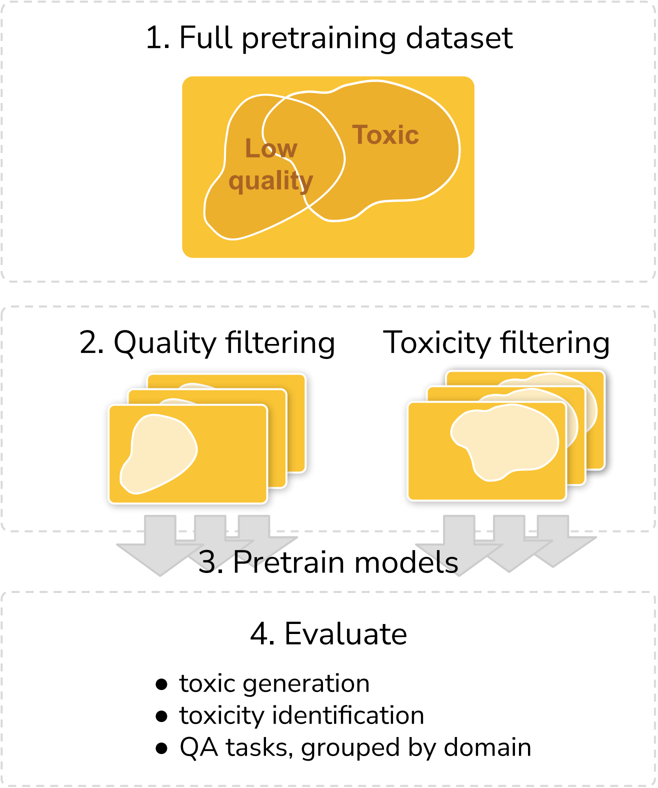
			QA domain				
	Filter	**Data**	Wiki	Web	Acad	CS	**Mean**
Baseline	Full Data	100%	0	0	0	0	0
Toxicity	Light (T=0.9)	95%	-2.2	-1.1	+0.2	+0.2	-0.7
	Heavy (T=0.5)	76%	-4.2	-2.4	-1.1	-3.5	-2.7
	Inverse	92%	+0.4	-1.4	+4.9	+2.7	+1.7
Quality	Light (T=0.975)	91%	+1.2	+0.7	+6.4	+6.1	+2.5
	Heavy (T=0.9)	73%	-0.3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

Takeaways:

1. Toxicity filtering hurts performance across domains.

2. Quality filtering improves performance across most domains, despite removing data.

Toxicity and quality filters: **recommendations**



1. If the goal is to identify toxic text, then don't use toxicity filters.

2. Use quality filters: quality filters improve performance, despite removing training data

3. Investigate other kinds of quality filtering, not just similarity to books and Wikipedia

Domain composition

1. Full pretraining dataset

2. Ablate domain

3. Pretrain models

4. Evaluate

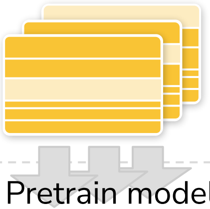
- QA tasks, grouped by domain

Domain composition

1. Full pretraining dataset

Baseline: all domains

2. Ablate domain

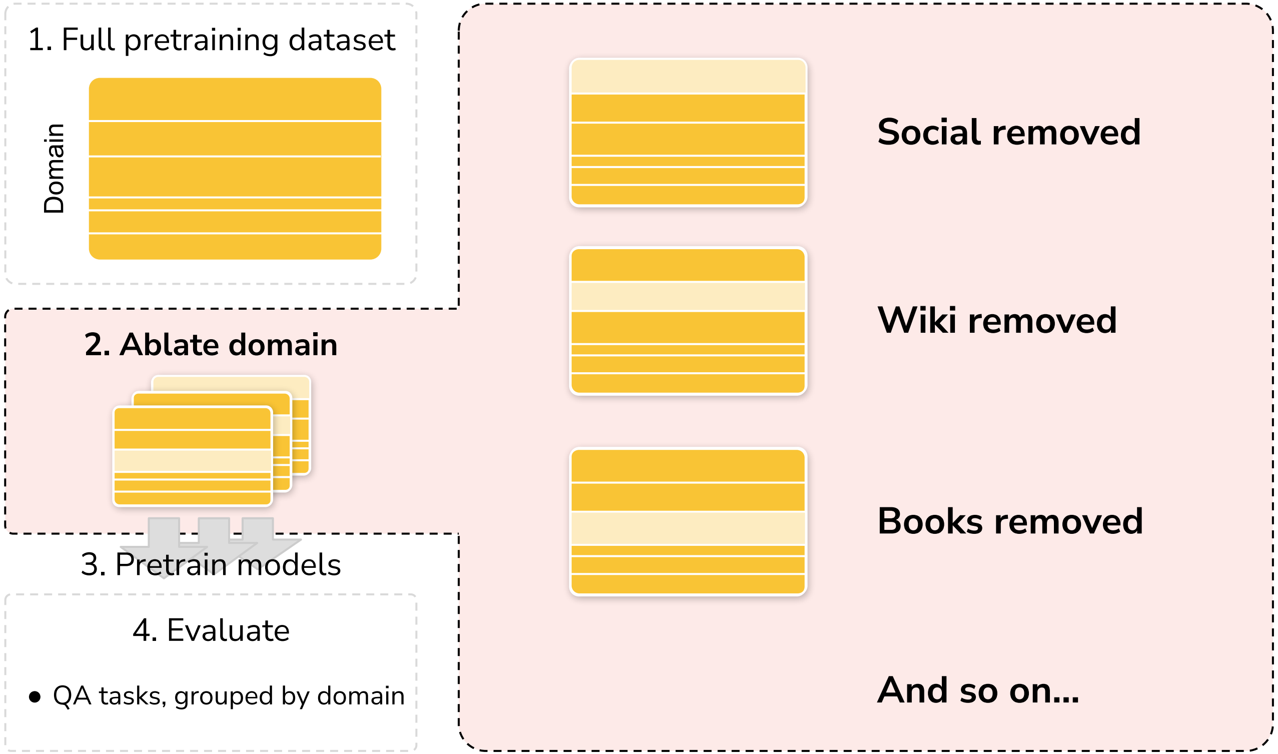


3. Pretrain models

4. Evaluate

- QA tasks, grouped by domain

Domain composition



Domain composition

1. Full pretraining dataset

Domain

2. Ablate domain

3. Pretrain models

1. Pretrain a model on each dataset
2. Finetune each model on downstream tasks separately

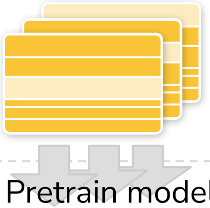
4. Evaluate

- QA tasks, grouped by domain

Domain composition

1. Full pretraining dataset

2. Ablate domain



3. Pretrain models

4. Evaluate

- QA tasks, grouped by domain

Same as previous domain setup

Domain composition

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Domain composition

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Domain composition

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Domain composition

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Domain composition

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Takeaways:

1. Removing books and Common Crawl domains hurt downstream performance the most.

Domain composition

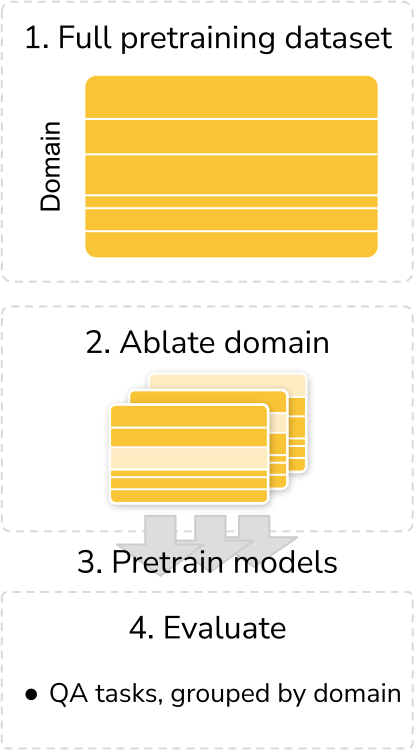
	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Takeaways:

1. Removing books and Common Crawl domains hurt downstream performance the most.

2. Targeted data helps for targeted evaluations.

Domain composition: **recommendations**



1. Train on as much data as possible.
   Quantity matters more than domain composition.

2. Prioritize heterogeneous data sources.

Today's Talk: *A Pretrainer's Guide*

1) **Introduction**
    - Data curation is everywhere
    - Experimental Setup
2) **Effects of Data Age**
3) **Effects of Quality & Toxicity Filters**
4) **Effects of Data Composition**
5) **Key Takeaways**

Key Takeaways

- ➔ Data is largely undocumented & unknown. Practitioners are *guided by intuition*.
- ➔ Stale pretraining data matters & is not overcome by finetuning!
- ➔ Temporal misalignment effects grow with model size.
- ➔ “Quality” filters boost performance, even while reducing training data.
- ➔ Toxicity filters hurt. Inverse toxicity filters can help a lot for some tasks.
- ➔ Data heterogeneity and quantity matter most, especially web and books data.

Key Limitations

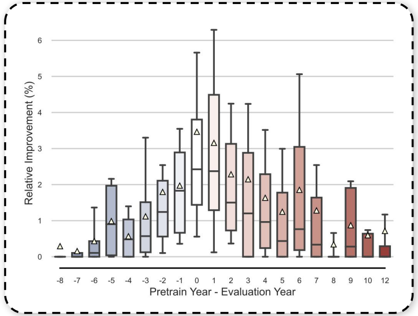
- “Quality” is ill-defined & deserves more attention.
- Compute is expensive! But so is dark data & documentation debt.
- Blackbox APIs have limitations.

A Pretrainer’s Guide to Training Data

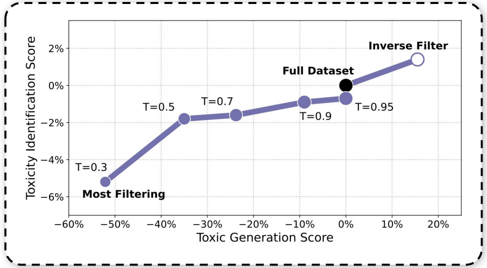
Data curation is everywhere.

Model	Represented Domains (%)									Filters		Data	
	Wiki	Web	Books	Dialog	Code	Acad	Pile	C4	M-L	Tox	Qual	Pub	Year
BERT	76		24				✗	✗			H	Part	2018
GPT-2		100					✗	✗			H	Part	2019
RoBERTa	7	90	3				✗	✔			H	Part	2019
XLNet	8	89	3				✗	✔			H	Part	2019
T5	<1	99					✗	✔		H	H	✔	2019
GPT-3	3	82	16				✗	✔	7%		C	✗	2021
GPT-J/Neo	1.5	38	15	4.5	13	28	✔	Part			C	✔	2020
GLaM	6	46	20	28			✗	✔			C	✗	2021
LaMDA	13	24		50	13		✗	✔	10%	C	C	✗	2021
AlphaCode					100		✗	✗			H	✗	2021
CodeGen	1	24	10	3	40	22	✔	Part			H	Part	2020
Chinchilla	1	65	10		4		✗	✔		H	C	✗	2021
Minerva	<1	1.5	<1	2.5	<1	95	✗	✔	<1%		C	✗	2022
BLOOM	5	60	10	5	10	10	✔	✔	71%		C	Part	2021
PaLM	4	28	13	50	5		✗	✔	22%		C	✗	2021
Galactica	1	7	1		7	84	✗	Part			H	Part	2022
LLAMA	4.5	82	4.5	2	4.5	2.5	Part	✔	4%		C	Part	2020

Data age: pretrained models become stale



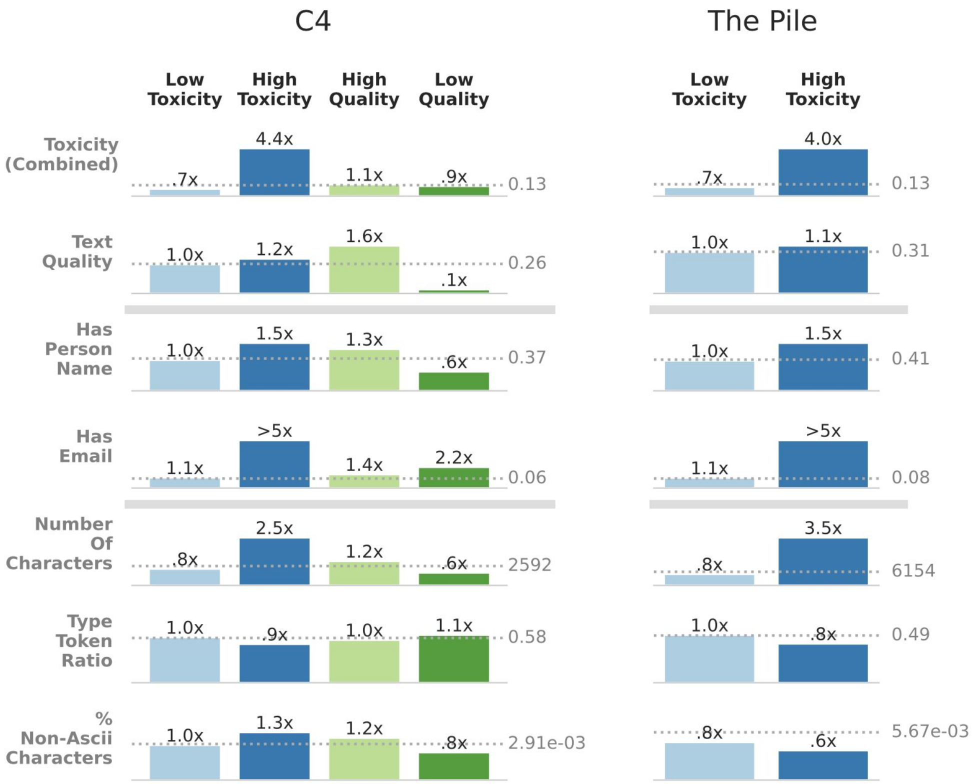
Quality filters improve performance. Toxicity filters hurt.



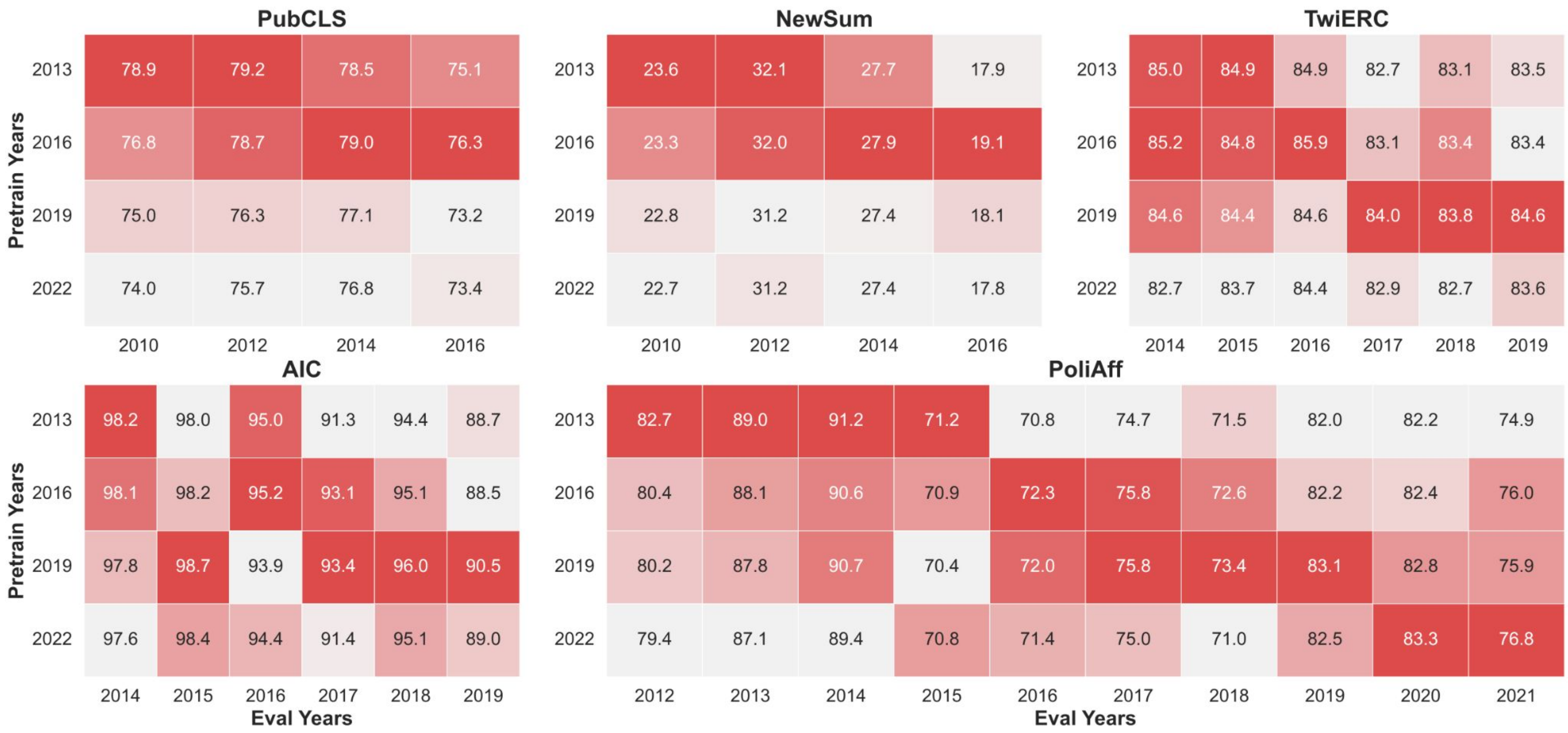
Domain composition: heterogeneity and quantity improve performance

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
No Social (99%)	-0.8	-3.7	0.1	3.5	-3.5	3.5	0.3
No Wiki (98%)	-1.3	-5.3	0.2	0.9	-4.4	7.2	-0.4
No Books (93%)	-3.5	-6.3	0.0	-1.6	-6.5	-4.4	-2.8
No OpenWeb (93%)	-2.0	-4.1	-1.0	0.6	-5.8	-2.9	-1.4
No Legal (91%)	-2.7	-2.9	0.4	0.8	-2.6	-0.4	-0.7
No Academic (87%)	-0.3	-2.5	-0.9	2.2	-1.1	4.3	0.2
No Pubmed (85%)	-0.3	-3.0	-5.8	-1.5	-5.9	3.9	-1.4
No Code (81%)	-0.5	-3.1	-1.2	1.2	-5.8	4.4	-0.2
No CC (73%)	-3.2	-6.2	-4.6	-5.9	-8.0	-5.2	-4.9

Impact of data curation on data attributes



Data age


PubCLS
2013 78.9 79.2 78.5 75.1
2016 76.8 78.7 79.0 76.3
2019 75.0 76.3 77.1 73.2
2022 74.0 75.7 76.8 73.4
2010 2012 2014 2016
NewSum
2013 23.6 32.1 27.7 17.9
2016 23.3 32.0 27.9 19.1
2019 22.8 31.2 27.4 18.1
2022 22.7 31.2 27.4 17.8
2010 2012 2014 2016
TwiERC
2013 85.0 84.9 84.9 82.7 83.1 83.5
2016 85.2 84.8 85.9 83.1 83.4 83.4
2019 84.6 84.4 84.6 84.0 83.8 84.6
2022 82.7 83.7 84.4 82.9 82.7 83.6
2014 2015 2016 2017 2018 2019
AIC
2013 98.2 98.0 95.0 91.3 94.4 88.7
2016 98.1 98.2 95.2 93.1 95.1 88.5
2019 97.8 98.7 93.9 93.4 96.0 90.5
2022 97.6 98.4 94.4 91.4 95.1 89.0
2014 2015 2016 2017 2018 2019
PoliAff
2013 82.7 89.0 91.2 71.2 70.8 74.7 71.5 82.0 82.2 74.9
2016 80.4 88.1 90.6 70.9 72.3 75.8 72.6 82.2 82.4 76.0
2019 80.2 87.8 90.7 70.4 72.0 75.8 73.4 83.1 82.8 75.9
2022 79.4 87.1 89.4 70.8 71.4 75.0 71.0 82.5 83.3 76.8
2012 2013 2014 2015 2016 2017 2018 2019 2020 2021
Pretrain Years
Eval Years

Toxicity filters hurt downstream performance across domains

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.06 (92%)	0.4	-1.4	0.7	4.9	4.1	2.7	1.6
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.95 (98%)	-1.0	-0.4	-0.5	0.6	1.7	1.3	0.2
T=0.9 (95%)	-2.2	-1.1	-3.0	0.2	2.9	0.2	-0.7
T=0.7 (86%)	-2.1	-1.4	-2.9	0.1	-0.9	-0.2	-1.3
T=0.5 (76%)	-4.2	-2.4	-3.3	-1.1	-0.3	-0.1	-2.1
T=0.3 (61%)	-3.8	-4.4	-2.5	-0.3	-1.3	-3.5	-2.7

Toxicity filters hurt downstream performance across domains

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.06 (92%)	0.4	-1.4	0.7	4.9	4.1	2.7	1.6
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.95 (98%)	-1.0	-0.4	-0.5	0.6	1.7	1.3	0.2
T=0.9 (95%)	-2.2	-1.1	-3.0	0.2	2.9	0.2	-0.7
T=0.7 (86%)	-2.1	-1.4	-2.9	0.1	-0.9	-0.2	-1.3
T=0.5 (76%)	-4.2	-2.4	-3.3	-1.1	-0.3	-0.1	-2.1
T=0.3 (61%)	-3.8	-4.4	-2.5	-0.3	-1.3	-3.5	-2.7

Takeaways:

1. Significant toxicity filtering hurts performance across domains.

Toxicity filters hurt downstream performance across domains

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.06 (92%)	0.4	-1.4	0.7	4.9	4.1	2.7	1.6
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.95 (98%)	-1.0	-0.4	-0.5	0.6	1.7	1.3	0.2
T=0.9 (95%)	-2.2	-1.1	-3.0	0.2	2.9	0.2	-0.7
T=0.7 (86%)	-2.1	-1.4	-2.9	0.1	-0.9	-0.2	-1.3
T=0.5 (76%)	-4.2	-2.4	-3.3	-1.1	-0.3	-0.1	-2.1
T=0.3 (61%)	-3.8	-4.4	-2.5	-0.3	-1.3	-3.5	-2.7

Takeaways:

1. Significant toxicity filtering hurts performance across domains.
2. Discarding least toxic data actually improves most downstream performance.

Quality filters improve downstream performance across domains

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.5 (73%)	-5.0	-4.5	-2.2	-2.7	1.2	-6.4	-3.3
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.975 (91%)	1.2	0.7	6.1	6.4	4.7	6.1	2.7
T=0.95 (84%)	-1.2	1.0	3.7	-0.3	3.2	4.9	1.1
T=0.9 (73%)	-0.3	0.8	1.8	1.0	1.9	6.8	1.4
T=0.7 (46%)	-1.2	0.8	1.7	0.8	2.0	4.2	0.9

Quality filters improve downstream performance across domains

	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.5 (73%)	-5.0	-4.5	-2.2	-2.7	1.2	-6.4	-3.3
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.975 (91%)	1.2	0.7	6.1	6.4	4.7	6.1	2.7
T=0.95 (84%)	-1.2	1.0	3.7	-0.3	3.2	4.9	1.1
T=0.9 (73%)	-0.3	0.8	1.8	1.0	1.9	6.8	1.4
T=0.7 (46%)	-1.2	0.8	1.7	0.8	2.0	4.2	0.9

Takeaways:

1. Quality filtering improves performance across most domains, despite removing a lot of data.

Quality filters improve downstream performance across domains

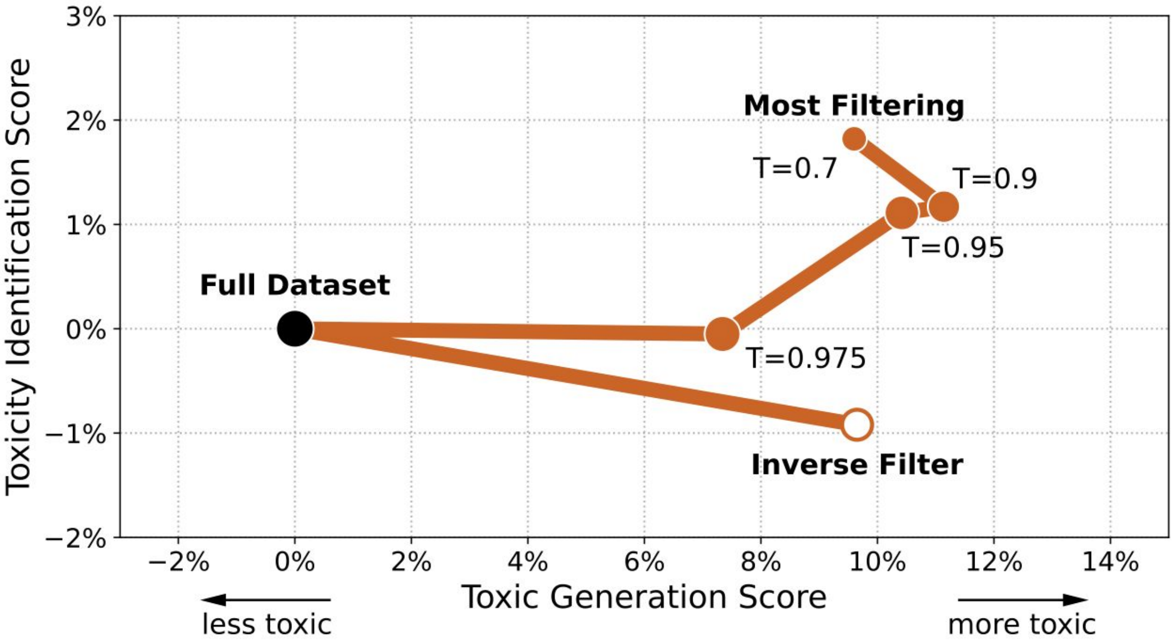
	Wiki	Web	Biomed	Academic	Common Sense	Contrast Sets	Average
Inverse T=0.5 (73%)	-5.0	-4.5	-2.2	-2.7	1.2	-6.4	-3.3
Full Dataset (100%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T=0.975 (91%)	1.2	0.7	6.1	6.4	4.7	6.1	2.7
T=0.95 (84%)	-1.2	1.0	3.7	-0.3	3.2	4.9	1.1
T=0.9 (73%)	-0.3	0.8	1.8	1.0	1.9	6.8	1.4
T=0.7 (46%)	-1.2	0.8	1.7	0.8	2.0	4.2	0.9

Takeaways:

1. Quality filtering improves performance across most domains, despite removing a lot of data.

2. Training on lowest quality data decreases performance.

Quality: mixed effect on toxicity evals



Takeaways:

1. Quality and toxicity are measuring very different things

2. Quality filtering improves toxicity identification