

Behind the Secrets of Large Language Models

Lecture 5

Collecting LLM training data

Michael Färber, Simon Razniewski

Acknowledgment

- Adapted from slides by Malte Ostendorff, IDESSAI 2024 summer school

Tales from the road



<https://iswc2025.semanticweb.org/>

**Knowledge Base Construction from
Pre-Trained Language Models**

Workshop @ 24th International Semantic Web
Conference (ISWC 2025)

<https://lm-kbc.github.io/workshop2025/>

[Introducing GPTKB to the Semantic Web](#)



Recap

Date	Lecture	
13.10.	Intro (Razniewski)	I. Foundations
20.10.	Word representation (Razniewski)	
27.10.	Neural networks (Färber)	
03.11.	Deep Learning+Attention (Färber)	
10.11.	Training data (Razniewski)	II. Building and Training LLMs
17.11.	Architectures (Färber)	
24.11.	Training (Razniewski)	
01.12.	Transfer learning (Färber)	
8.12.	Evaluation (Razniewski)	III. Using and extending LLMs
15.12.	Applications (Färber)	
05.01.	KGs/RAG (Razniewski)	
12.01.	Vision LMs (Haase)	
19.01.	Agents I (Haase)	
26.01.	Agents II (TBD)	III. Using and extending LLMs
2.2.	Ethics and safety (Razniewski)	

- LLMs are statistical models that **learn** a probability distribution over a sequence of words and that predict the next token for a given input.

- What do we learn from?

→ **Data!**

Outline

- 1. Importance of data (10')**
2. Pre-training (50')
 1. Crawling
 2. Scraping and Cleaning
 3. Datasets
3. Instruction tuning (5')
4. Alignment (5')
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Does data matter?

We find that data quality is critical to a highly-performing model

(Google Gemini technical report, 2023)

The innovation lies entirely in our dataset for training

(Phi-3 Technical Report, 2024)

A major factor in the recent success of large language models is the use of enormous and ever-growing text datasets for unsupervised pre-training.

(A Survey on Data Selection for Language Models, 2024)

Garbage in, garbage out

(Apocryphal)

→ Quality

→ Diversity

→ Size (see Lec. 7)

Training data

Stage 1: Unsupervised Pretraining



- Large amounts of plain text (Llama3: 15 trillion tokens)
- Diverse sources and topics (scientific literature, news, forums)
- Most prominent source: Web crawled text (CommonCrawl)

Stage 2: Supervised Instruction Fine-tuning



- Supervised text pairs (input-output, question-answer, text-summary)
- Task-oriented data (diverse tasks are needed to generalize to unseen tasks)

Stage 3: Alignment / preference-tuning



- Human (or AI) feedback data on preferred output (style and format)
- Pairwise feedback (good vs bad)
- Listwise feedback (ranking)

Less public sources

Outline

1. Importance of data (10')
- 2. Pre-training (50')**
 - 1. Crawling**
 2. Scraping and cleaning
 3. Datasets
3. Instruction tuning (5')
4. Alignment (5')
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Recap: LLM

- A language model is a probability distribution over sequences of words

Next word Given words

- $P(X_{t+1} = x_{t+1} | x_1, \dots, x_t)$

- *Training data:*

$X_{11}, X_{12}, X_{13}, X_{14}, \dots$

$X_{21}, X_{22}, X_{23}, X_{24}, \dots$

$X_{31}, X_{32}, X_{33}, X_{34}, \dots$

Billion to trillion times

How to get this?

Crawling, scraping, filtering!

Crawling: Task

- **Given:** One or several source URLs
- **Return:** Document corpus obtained by transitive hyperlink closure (bounded)

Donald John Trump (born June 14, 1946) is the 45th President of the [United States](#).

click

The United States unites lots of states:
Some of the cooler ones are
[California](#) and [New York](#).

click

click

Crawler: Basic function

1. Start with queue of important URLs

http://... http://... http://...

2. Download



`<a href=XYZ></a>`

4. Find URLs in page, enqueue them

3. If page is "good", add it to corpus



5. repeat the process until you covered all pages

- within a certain depth
- in a certain domain
- with certain topics
- .

How to find new URLs?

- In an **HTML page**
 - Hyperlinks `<a href="...">`
 - Media ``, `<audio src="...">`, `<video src="...">`, `<source src="...">`
 - Frames `<iframe src="...">`
 - Page text by regular expressions
- In **other kinds of files** (PDFs...).
- In **sitemaps** provided specifically to crawlers

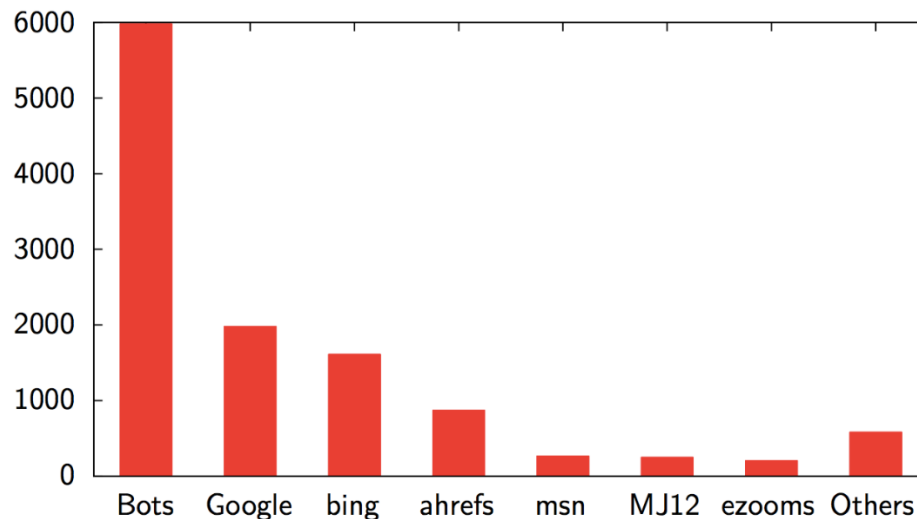
How much traffic do crawlers cause?

[Cloudflare blog, 2025]

“Around 30% of global web traffic today comes from bots”

Almost half of all internet traffic comes from bots. For the fifth consecutive year in a row this figure has been on the rise.

[2024 Imperva Bad Bot Report]

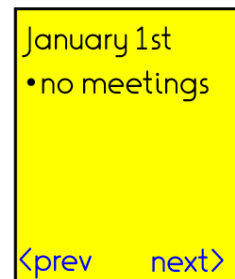
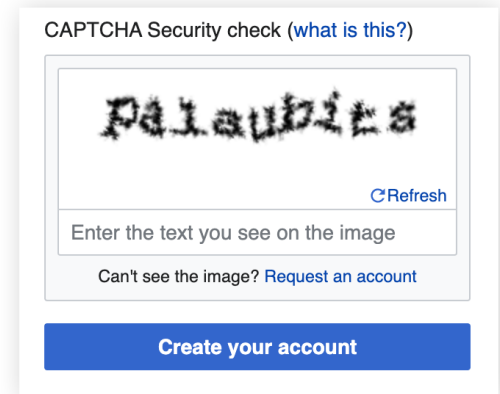


How to avoid being blocked?

- **Parallel requests** can easily overload web server capacities
- Serious consequences (crashes) are rare, but commercial servers will quickly block you
- Precaution:
 - **Wait a minimal delay between requests to the same server.**
 - ⇒ Depends on the server (wikipedia.org vs your laptop).
 - ⇒ Depends on the resource (large files...).
 - ⇒ Generally, waiting at least one second is preferable.
- Requests to **different servers** can be **parallelized**.

Who likes crawlers?

- Writing crawlers is super easy!
 - 30 lines in Python, 10 minutes → 10k+ websites
 - Not everyone likes crawlers
- Honor-based exclusion: [robots.txt](#)
 - <http://tu-dresden.de/robots.txt>
- Hard exclusion: [Captchas](#)
 - De-facto solved by ... GenAI
- [Traps](#) for naive crawlers



Outline

1. Importance of data (10')
- 2. Pre-training (50')**
 1. Crawling
 - 2. Scraping and cleaning**
 3. Datasets
3. Instruction tuning (5')
4. Alignment (5')
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Scraping and cleaning

- Raw web data is **ugly**

```
<div class="wp-block-columns container pt-8 is-layout-flex wp-container-core-columns-is-lay
<div class="wp-block-column is-layout-flow wp-block-column-is-layout-flow">
<h2 class="wp-block-heading">ScaDS.AI Dresden/Leipzig</h2>

<p class="pr-8">ScaDS.AI (Center for Scalable Data Analytics and Artificial Intelligence) and B
<div class="colored-link"><a href="https://scads.ai/about-us" class="default">Learn more</
<div class="wp-block-column is-layout-flow wp-block-column-is-layout-flow">
<figure class="wp-block-image"><img loading="lazy" decoding="async" width="1440" ds/GA
</div>
</div>
```

- Extensive cleaning needed
 - Remove code, retain content (+structure?)

Regex to the rescue

- **Regular expression** (regex) are sequences of characters that specifies a match pattern in text
- Usually used by string-searching algorithms for **"find"** or **"find and replace"** operations on strings
- Rooted in theoretical computer science and formal language theory

Regex cheatsheet

Character	Meaning	Example
*	Match zero, one or more of the previous	Ah* matches "Ahhhh" or "A"
?	Match zero or one of the previous	Ah? matches "A" or "Ah"
+	Match one or more of the previous	Ah+ matches "Ah" or "Ahhh" but not "A"
\	Used to escape a special character	Hungry\? matches "Hungry?"
.	Wildcard character, matches any character	do.* matches "dog", "door", "dot", etc.
()	Group characters	See example for
[]	Matches a range of characters	[cbf]ar matches "car", "bar", or "far" [0-9]+ matches any positive integer [a-zA-Z] matches ascii letters a-z (uppercase and lower case) [^0-9] matches any character not 0-9.
 	Matche previous OR next character/group	(Mon) (Tues)day matches "Monday" or "Tuesday"
{ }	Matches a specified number of occurrences of the previous	[0-9]{3} matches "315" but not "31" [0-9]{2,4} matches "12", "123", and "1234" [0-9]{2,} matches "1234567..."
^	Beginning of a string. Or within a character range [] negation.	^http matches strings that begin with http, such as a url. [^0-9] matches any character not 0-9.
\$	End of a string.	ing\$ matches "exciting" but not "ingenious"

Metacharacter	Purpose
\w	[a-zA-Z0-9_] word characters
\s	whitespace characters
\d	[0-9] digit characters
\W	[^a-zA-Z0-9_] non-word characters
\S	non-whitespace characters
\D	[^0-9] non-digit characters
.	any character
\n	newline characters

Regex tasks

1. Write a regex to find all HTML tags (<....>)
 - `<[>]*>`
2. Write a regex to match all URLs
 - `https?:/(?:www\.)?[a-zA-Z0-9-]+\.[a-zA-Z]{2,}(?:/[^\s]*)?`
3. Write a regex to find all hyperlink texts (content between <a href ...> XYZ)
 - `<a\s+href="["']*">(.*?)</a>`

BeautifulSoup + Selenium

- **BeautifulSoup**

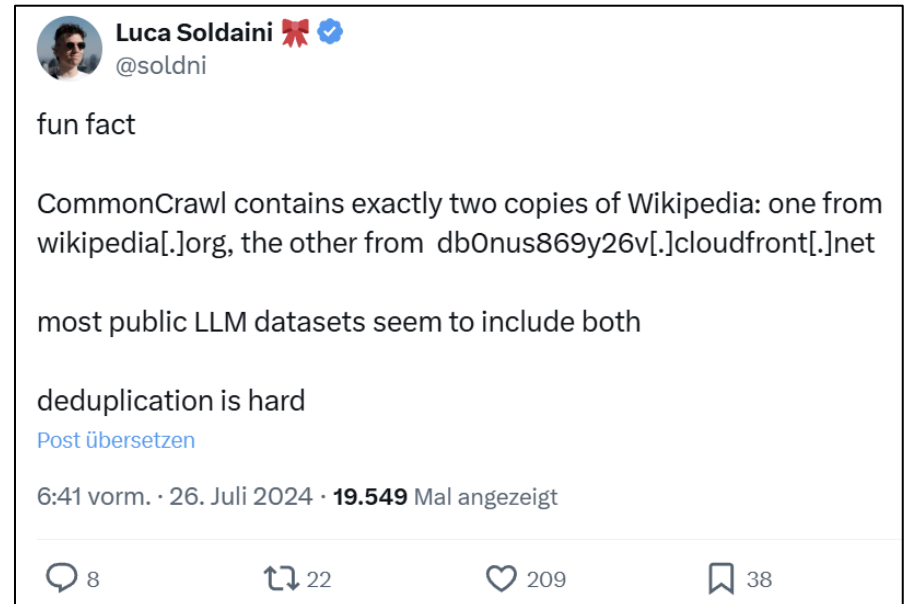
- **Python HTML/XML Parsing Library:** Great for static pages; works with requests
- **Lightweight & Fast:** Quickly extracts tags, attributes, and text

- **Selenium**

- **Browser Automation:** Controls real browsers to interact with dynamic pages
- **Handles JavaScript:** Ideal for sites with interactive elements
- **Slower but Versatile:** Suitable for complex websites, supports running without rendering (headless mode)

Deduplication

- Everyone copies from everyone
- Common tactic of **commercial spammers** (content farms)
- Even good-faith often no exact duplicates (dates, quotes, corrections, ...)



- **Deduplication at scale challenging**
 - Locality-sensitive hashing (LSH)
 - MinHash (Andrei Broder, 1997)

Data cleaning in C4

- We only retained lines that ended in a terminal punctuation mark (i.e. a period, exclamation mark, question mark, or end quotation mark).
- We discarded any page with fewer than 3 sentences and only retained lines that contained at least 5 words.
- We removed any page that contained any word on the “List of Dirty, Naughty, Obscene or Otherwise Bad Words”.⁶
- Many of the scraped pages contained warnings stating that Javascript should be enabled so we removed any line with the word Javascript.
- Some pages had placeholder “lorem ipsum” text; we removed any page where the phrase “lorem ipsum” appeared.
- Some pages inadvertently contained code. Since the curly bracket “{” appears in many programming languages (such as Javascript, widely used on the web) but not in natural text, we removed any pages that contained a curly bracket.
- Since some of the scraped pages were sourced from Wikipedia and had citation markers (e.g. [1], [citation needed], etc.), we removed any such markers.
- Many pages had boilerplate policy notices, so we removed any lines containing the strings “terms of use”, “privacy policy”, “cookie policy”, “uses cookies”, “use of cookies”, or “use cookies”.
- To deduplicate the data set, we discarded all but one of any three-sentence span occurring more than once in the data set.

OSCAR: Multilingual Web Data



- OSCAR project aims to provide web-based multilingual resources and datasets
- Web data filtering is challenging!
 - Large scale of Web data (+100 TBs and more)
 - All filtering methods need to be highly-efficient and scalable.
- Simple heuristics ([OSCAR documentation](#)):
 - tiny: The document has a low (<5) number of lines.
 - short_sentences: The document has a high number (>50%) of short lines (<400 bytes)
 - noisy: The document has a high percentage of punctuation (>50%)
 - and other filters ...
- Filter Web content based on URL using block lists ([Blacklist UT1](#))
- Adult Content Filtering: Perplexity from a KenLM model trained on known adult content.
- Deduplication: Remove duplicated content with Locality Sensitive Hashing.

Filters remove
~90% of the data

Deduplication removes
another ~50%

Filtering: Fine print

[Dodge, Jesse et al. “Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus.” EMNLP’21]

“We provide some of the first documentation for the C4 corpus. We begin by investigating where the data came from, and find a significant amount of text from [unexpected sources like patents and US military websites](#).

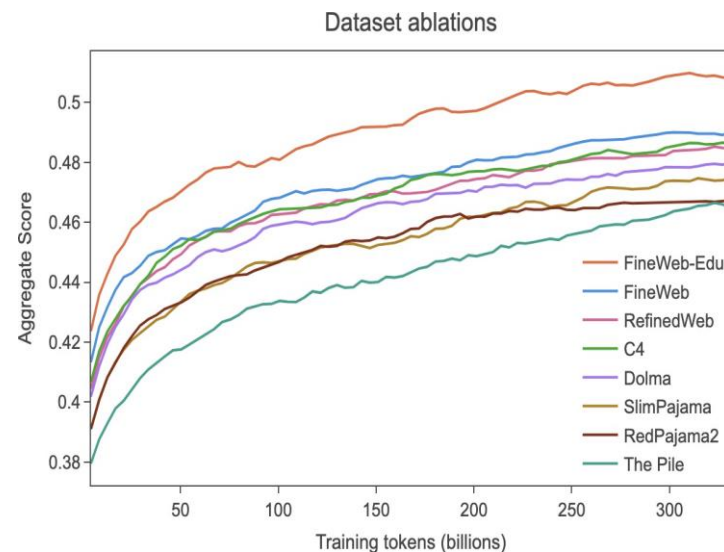
To understand the impact of the filters applied to create this dataset, we evaluate the text that was removed, and show that [blocklist filtering disproportionately removes text from and about minority individuals](#).”

What data do we want to retain?

- The term “**high quality**” is not well-defined!
 - Statistical models such as LLMs are defined by their training data
 - High quality data = high quality model

→ The only solution: **Run experiments!**

- [E.g., Penedo et al. \(2024\)](#) conducted data quality studies for language model pretraining.
 - More than **60k H100 GPU hours*** for training ablation models of 1.8B param for ~ 28B tokens
 - **FineWeb-Edu**: Most performance gains are achieved by using a Transformer-based model to filter for educational content.



* \$4.9 per [H100-GPU hour on AWS](#) (\$ 300,000 in total)

Outline

1. Importance of data (10')
- 2. Pre-training (50')**
 1. Crawling
 2. Scraping and cleaning
 - 3. Datasets**
3. Instruction tuning (5')
4. Alignment (5')
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Existing open datasets

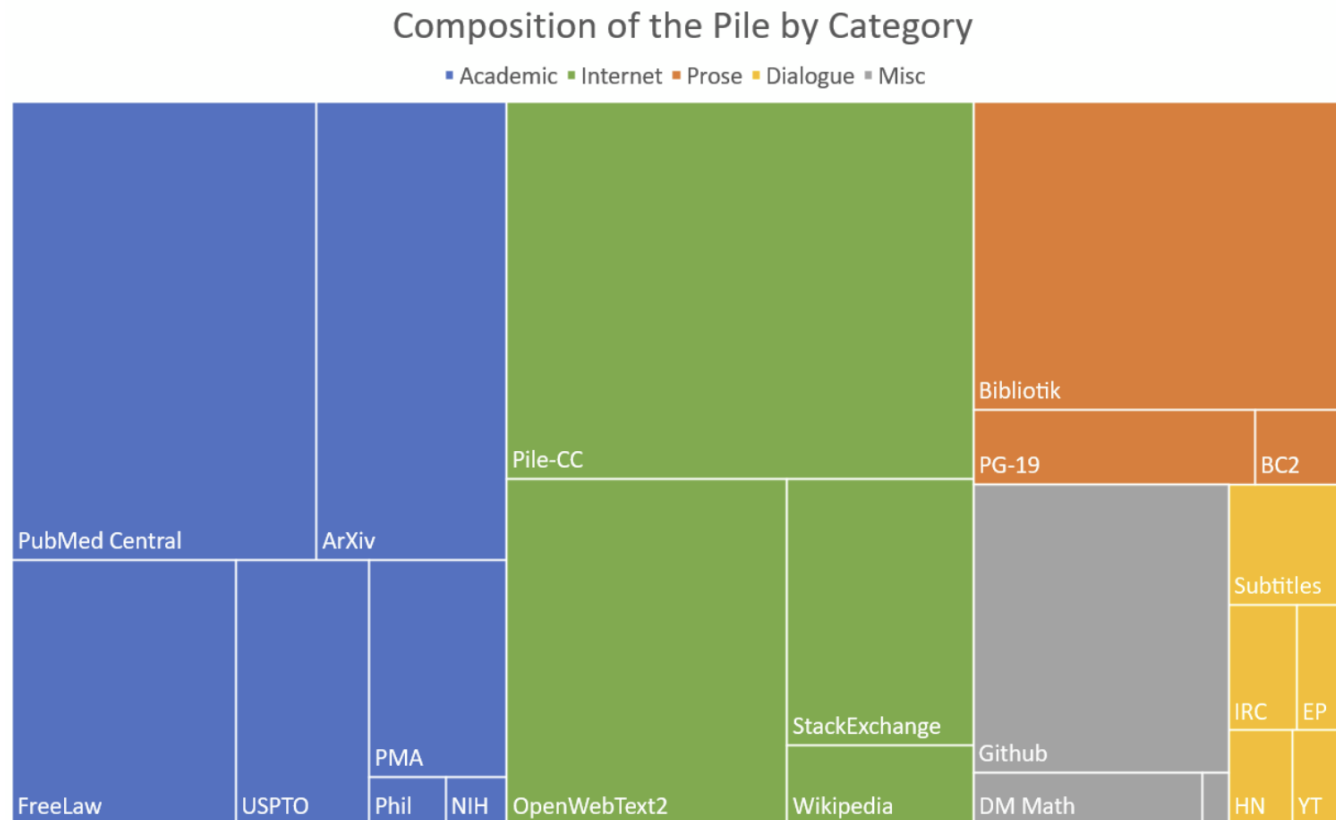
	pages	size
ClueWeb	1b	25 TB
CommonCrawl	6b	100TB
Internet Archive	2b	80TB
C4 corpus		750 GB
The Pile		825 GB
enWikipedia	5m	30 GB
Dresden web table corpus	125m	
Twitter dumps 2016 US election	280m	
Reddit dumps		...
Wikia dumps		...



What do known LLMs use?

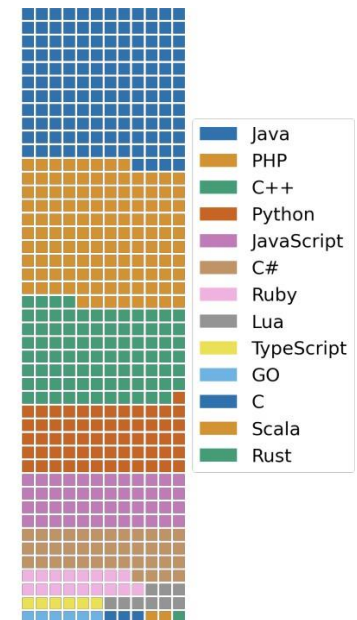
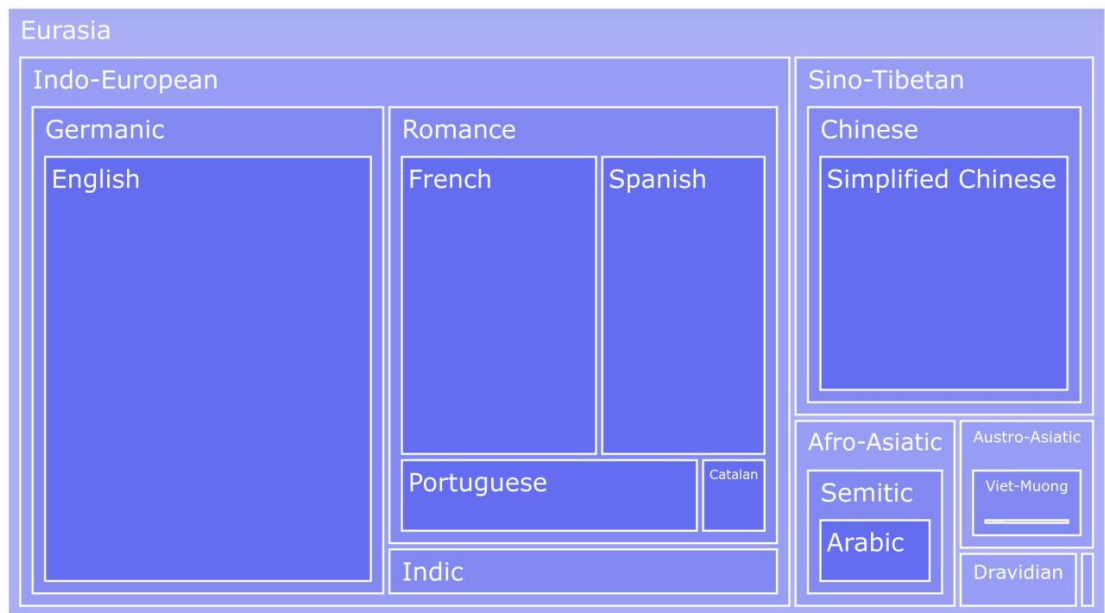
- **GPT2** (2019): 40 GB of Web text (from outbound Reddit links)
- **BERT** (2019): BooksCorpus (800M words) and Wikipedia (2,500M words)
- **GPT3** (2020): 570GB of Web text after filtering, roughly equivalent to 400 billion byte-pair-encoded tokens, Books1, Books2, Wikipedia
- **GPT-Neox / The Pile** (2020): 800GB text from Web, academic text, news, Wikipedia, subtitles, books, laws, programming code.
- **Bloom /Roots** (2022): 1.6TB of text from various sources (mostly Web text)
- Today's frontier model LLM providers say very little about their training data.
 - Meta: *"Llama 3 was pretrained on 15T tokens ...from publicly available sources"*
 - Training data as main differentiator between models. Crucial assets!
 - Legal issues such as copyright
(see ongoing lawsuit: [OpenAI vs New York Times](#))

GPT-NeoX /The Pile (2020)

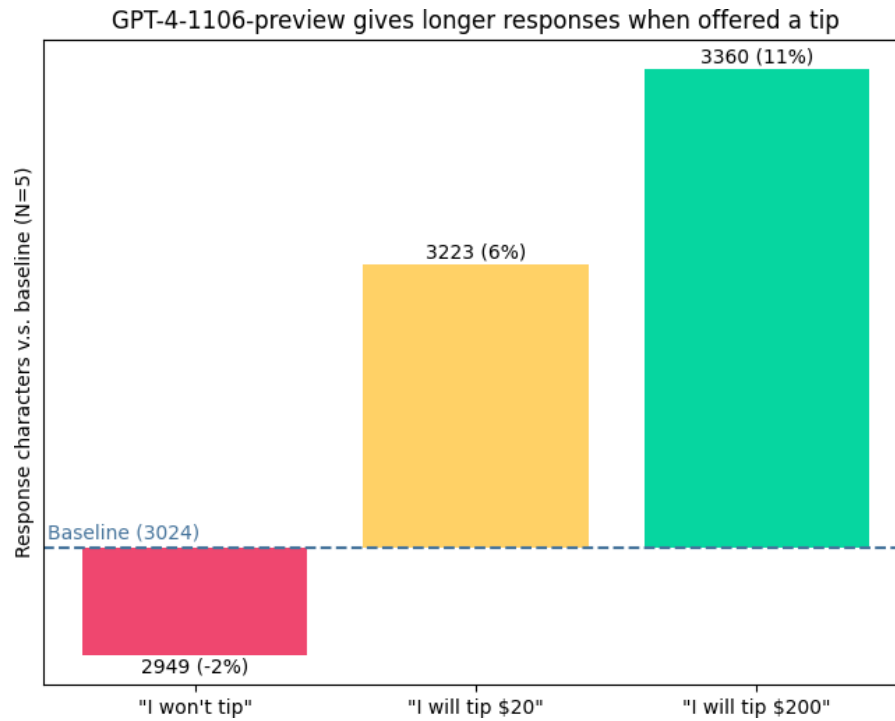


Oversampling – WP 3x, Gutenberg 2.5x, Arxiv 2x

ROOTS: Data for Training of Bloom (2022)



ChatGPT is American.



Source: <https://x.com/voooooogel/status/1730726744314069190> (@voooooogel on X)

Outline

1. Importance of data (10')
2. Pre-training (50')
 1. Crawling
 2. Scraping and cleaning
 3. Datasets
- 3. Instruction tuning (5')**
4. Alignment (5')
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Supervised Instruction Fine-tuning

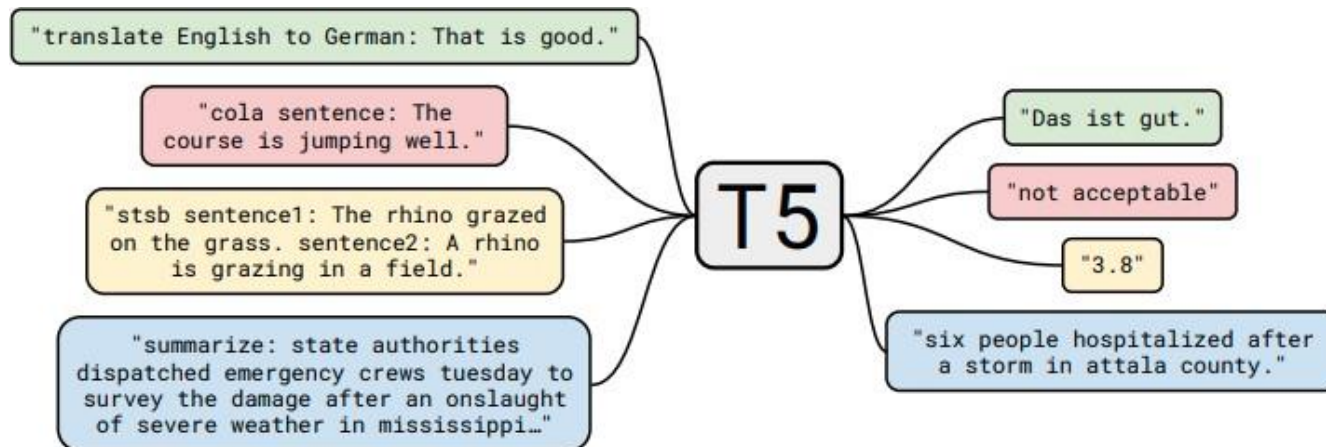
- LLMs predict the **next token with the highest probability** according to their training data
- A base language model is **not a chatbot** but a completion engine for Web content and other text seen during training
- Chatbots (language understanding, world knowledge, ...) and other applications are built on top of LLMs
- LLMs can be **optimized for specific tasks and applications** (e.g., chatbots, instruction following, ...) via fine-tuning (comparable to classical supervised learning)
- Supervised fine-tuning requires structured/annotated datasets (input- output pairs for multiple tasks):

Vanilla LLM (Base model)	
Input	Where is the Eiffel Tower located?
Output	How tall is the Eiffel Tower?

QA-Fine-tuned LLM	
Input	Where is the Eiffel Tower located?
Output	Paris

Multi-Task Learning

- LLMs are intended to be general-purpose models:
Many different tasks (unseen during training)
- Multi-task learning is a common approach in various machine learning domains
- Raffel et al. (2019) popularized multi-task learning with their T5 language model



cola = Corpus of Linguistic Acceptability = grammaticality
stsb = Semantic Textual Similarity Benchmark = semantic similarity

Multi-task datasets over time

Release	Collection	Model Details				Data Collection & Training Details			
		Model	Base	Size	Public?	Prompt Types	Tasks in Flan	# Exs	Methods
2020 05	UnifiedQA	UnifiedQA	RoBerta	110-340M	P	ZS	46 / 46	750k	
2021 04	CrossFit	BART-CrossFit	BART	140M	NP	FS	115 / 159	71M	
2021 04	Natural Inst v1.0	Gen. BART	BART	140M	NP	ZS / FS	61 / 61	620k	+ Detailed k-shot Prompts
2021 09	Flan 2021	Flan-LaMDA	LaMDA	137B	NP	ZS / FS	62 / 62	4.4M	+ Template Variety
2021 10	P3	T0, T0+, T0++	T5-LM	3-11B	P	ZS	62 / 62	12M	+ Template Variety + Input Inversion
2021 10	MetaICL	MetaICL	GPT-2	770M	P	FS	100 / 142	3.5M	+ Input Inversion + Noisy Channel Opt
2021 11	ExMix	ExT5	T5	220M-11B	NP	ZS	72 / 107	500k	+ With Pretraining
2022 04	Super-Natural Inst.	Tk-Instruct	T5-LM, mT5	11-13B	P	ZS / FS	1556 / 1613	5M	+ Detailed k-shot Prompts + Multilingual
2022 10	GLM	GLM-130B	GLM	130B	P	FS	65 / 77	12M	+ With Pretraining + Bilingual (en, zh-cn)
2022 11	xP3	BLOOMz, mT0	BLOOM, mT5	13-176B	P	ZS	53 / 71	81M	+ Massively Multilingual
2022 12	Unnatural Inst.†	T5-LM-Unnat. Inst.	T5-LM	11B	NP	ZS	~20 / 117	64k	+ Synthetic Data
2022 12	Self-Instruct†	GPT-3 Self Inst.	GPT-3	175B	NP	ZS	Unknown	82k	+ Synthetic Data + Knowledge Distillation
2022 12	OPT-IML Bench†	OPT-IML	OPT	30-175B	P	ZS + FS CoT	~2067 / 2207	18M	+ Template Variety + Input Inversion + Multilingual
2022 10	Flan 2022 (ours)	Flan-T5, Flan-PaLM	T5-LM, PaLM	10M-540B	P NP	ZS + FS CoT	1836	15M	+ Template Variety + Input Inversion + Multilingual

Scaling:

from 46 to
~2000 tasks

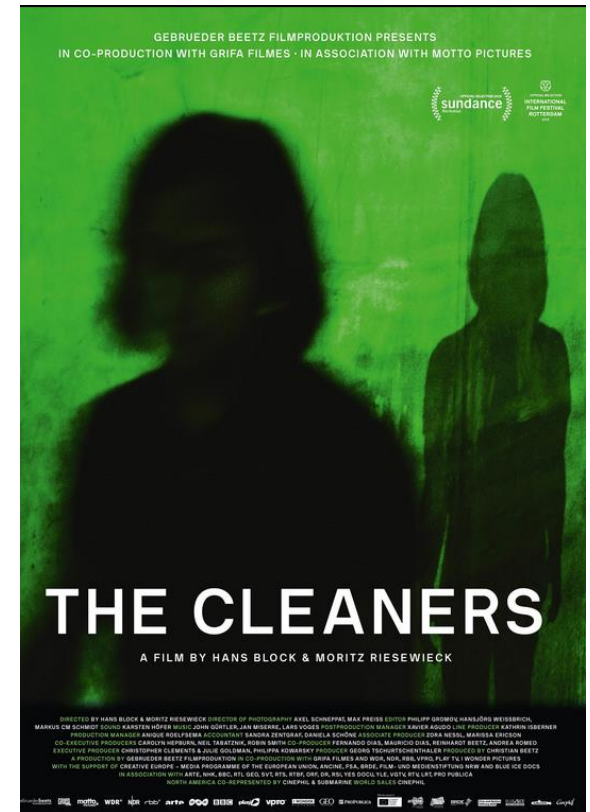
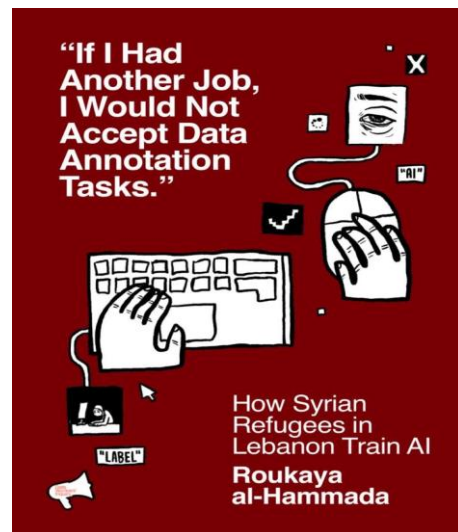
from 750k
to 15M
examples

Source: Longpre et al. (2023) [“TheFlan Collection: Designing Data and Methods for Effective Instruction Tuning”](#)

Where do these datasets come from?

- Instruction data not easily available on the web
→ Much **human generation** involved
- **Data annotation long tradition** in NLP
 - Traditional expert annotations guidelines sometimes >100 pages
- **Crowdsourcing (“clickwork”)**
 - Prominent platforms: Amazon Mechanical Turk, Prolific, ...
 - Complex or open-ended annotation tasks difficult
→ Task design as art, divide and conquer
 - Automated quality checks via agreement/honeypot questions
 - Often minimum-wage workers at fringes of society
- Crowdsourcing for instructions
 - Expenses likely in 100,000s of \$ for commercial providers

The dark side of clickwork



Outline

1. Importance of data (10')
2. Pre-training (50')
 1. Crawling
 2. Scraping and cleaning
 3. Datasets
3. Instruction tuning (5')
- 4. Alignment (5')**
5. Synthetic data (10')
6. Discussion (10')
 - Group work

Alignment & Preference Fine-Tuning

- OpenAI popularized “reinforcement learning from human feedback” (RLHF) as a technique to align language model outputs more with human preferences ([Stiennon et al., 2020](#))
- There are different ML algorithms for preference tuning (PPO, DPO, KTO, ...)
- Each of the methods expects training data in a different format

Chosen	Reject
The Eiffel Tower is located in Paris, France....	Paris

Response	Label
Paris	0
The Eiffel Tower is located in Paris, France....	1

Base LLM	
Input	Where is the Eiffel Tower located?
Output	How tall is the Eiffel Tower?

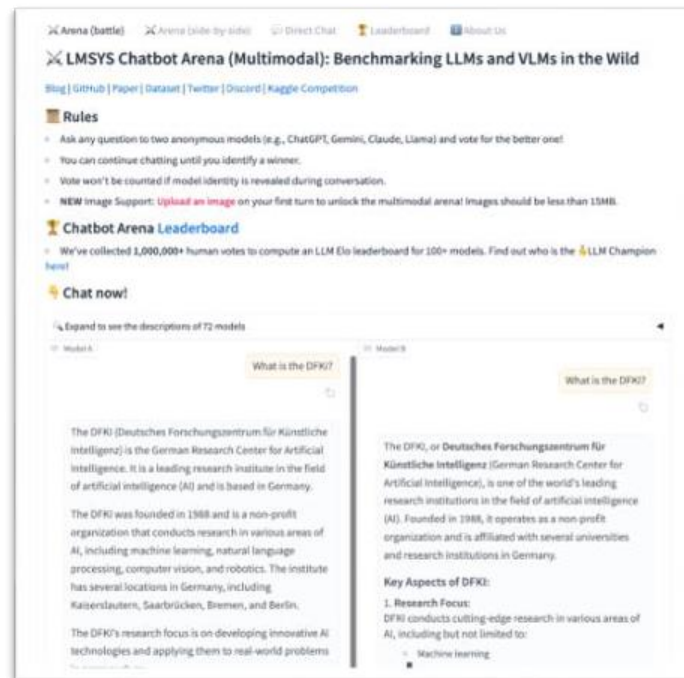


QA-Fine-tuned LLM	
Input	Where is the Eiffel Tower located?
Output	Paris



Preference-fine-tuned LLM	
Input	Where is the Eiffel Tower located?
Output	The Eiffel Tower is located in Paris, France. Specifically, it is situated on the Champ de Mars, a public park in the 7th arrondissement (district) of Paris, near the Seine River. The exact address is: 5 Avenue Anatole France, 75007 Paris, France. [...]

Chatbot Arena



<https://lmarena.ai/>

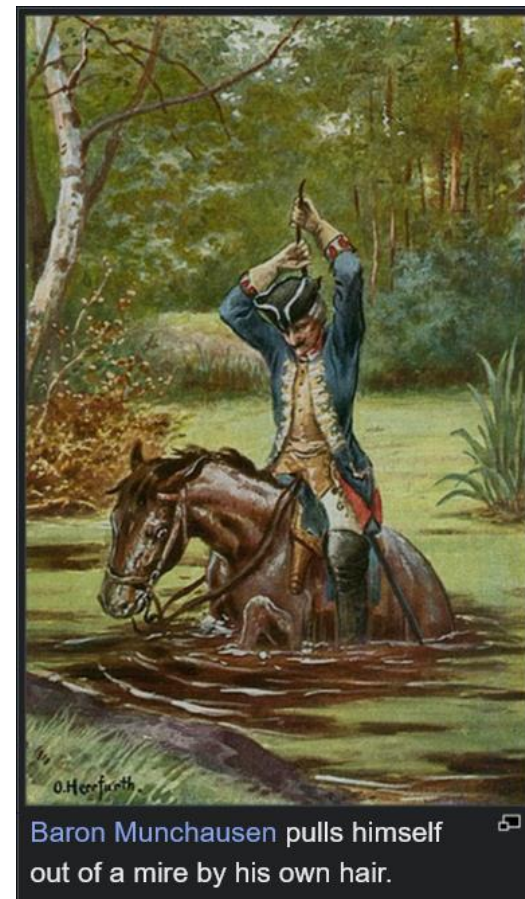
Let's try!

Outline

1. Importance of data (10')
2. Pre-training (50')
 1. Crawling
 2. Scraping and cleaning
 3. Datasets
3. Instruction tuning (5')
4. Alignment (5')
- 5. Synthetic data (10')**
6. Discussion (10')
 - Group work

Can we use GenAI to generate training data?

- This sounds odd?
- This is odd!
- Intuition 1: Do not add content, but **increase density/downstream utility**
- Intuition 2: **Exploit stronger model to train weaker one**
- Intuition 3: Ranking/scoring:
Evaluation is easier than generation



Intuition 1.1:

Cleaning/rephrasing data

```
<div class="wp-block-columns container pt-8 is-layout-flex wp-container-core-columns-is-lay
<div class="wp-block-column is-layout-flow wp-block-column-is-layout-flow">
<h2 class="wp-block-heading">ScaDS.AI Dresden/Leipzig</h2>

<p class="pr-8">ScaDS.AI (Center for Scalable Data Analytics and Artificial Intelligence) and B
<div class="colored-link"><a href="https://scads.ai/about-us" class="default">Learn more</
<div class="wp-block-column is-layout-flow wp-block-column-is-layout-flow">
<figure class="wp-block-image"><img loading="lazy" decoding="async" width="1440" ds/GA
</div>
</div>
```

Original

The stock rose \$2.11, or about 11 percent, to close Friday at \$21.51 on the New York Stock Exchange.

Revenue in the first quarter of the year dropped 15 percent from the same period a year earlier.

Medium Style

The stock experienced an increase of approximately 11 percent, closing at \$21.51 on the New York Stock Exchange on Friday, with a rise of \$2.11.

During the initial three months of the current year, there was a 15 percent decrease in revenue compared to the corresponding quarter of the previous year.

- Does not add information
- Removes irrelevant content
- Brings the data in a format closer to what is required downstream

Intuition 1.2:

Extract instructions from the web

Can obtain instructions from general web text



Raw Docs

Unformatted Text, Site Information, Ads

Topics Science\nAnatomy&Physiology\nAstronomy\nAstrophysics
\nBiology\nChemistry \n...Socratic Meta...Featured Answers
How do you simplify $\frac{(u^4 v^3)}{(u^2 v^{-1})^4}^0$ and write it using only positive exponents?

Answer by NickTheTurtle (Apr 1, 2017)

Explanation:\nAnything raised to the 0^{th} power is simply 1.
\n\nRelated Questions\nWhat is the quotient of powers property?
\n\n\nHow do you simplify expressions using the quotient rule?...
Impact of this question\n1274 views around the world
#Apps\niOS\nAndroid\nLinks\n[Privacy](#)\n[Terms](#)\n[Help](#)



Extracted QA

Formatted QA but lacking detailed solutions

Question: How do you simplify $(u^4 v^3 / (u^2 v^{-1})^4)^0$ and write it using only positive exponents?

Answer: Explanation: Anything to the 0th power is just simply 1.



Rewritten QA

Formatted QA augmented with detailed solutions

Question:

How do you simplify $(u^4 v^3 / (u^2 v^{-1})^4)^0$ and write it using only positive exponents?

Answer:

To simplify the expression $(u^4 v^3 / (u^2 v^{-1})^4)^0$ and rewrite it using only positive exponents, we start by evaluating the expression from the innermost operation outward.

First, consider the exponent of zero on the entire fraction:

Any expression raised to the power of zero is equal to 1 (provided the base is not zero). Therefore, $(u^4 v^3 / (u^2 v^{-1})^4)^0 = 1$

This simplification makes the other calculations unnecessary because raising any non-zero expression to the power of zero will always result in 1.

So, the simplified expression is simply: 1

Intuition 1.3:

Turn existing data into other tasks

TU Dresden (for German: Technische Universität Dresden, abbreviated as TUD), also as the Dresden University of Technology,[a] is a public research university in Dresden, Germany. It is the largest institute of higher education in the city of Dresden, the largest university in Saxony and one of the 10 largest universities in Germany with 32,389 students as of 2018.[3]

The name Technische Universität Dresden has only been used since 1961; the history of the university, however, goes back nearly 200 years to 1828. This makes it one of the oldest colleges of technology in Germany, and one of the country's oldest universities, which in German today refers to institutes of higher education that cover the entire curriculum. The university is a member of TU9, a consortium of the nine leading German Institutes of Technology. The university is one of eleven German universities which succeeded in the Excellence Initiative in 2012, thus getting the title of a "University of Excellence". The TU Dresden succeeded in all three rounds of the German Universities Excellence Initiative (Future Concept, Graduate Schools, Clusters of Excellence).

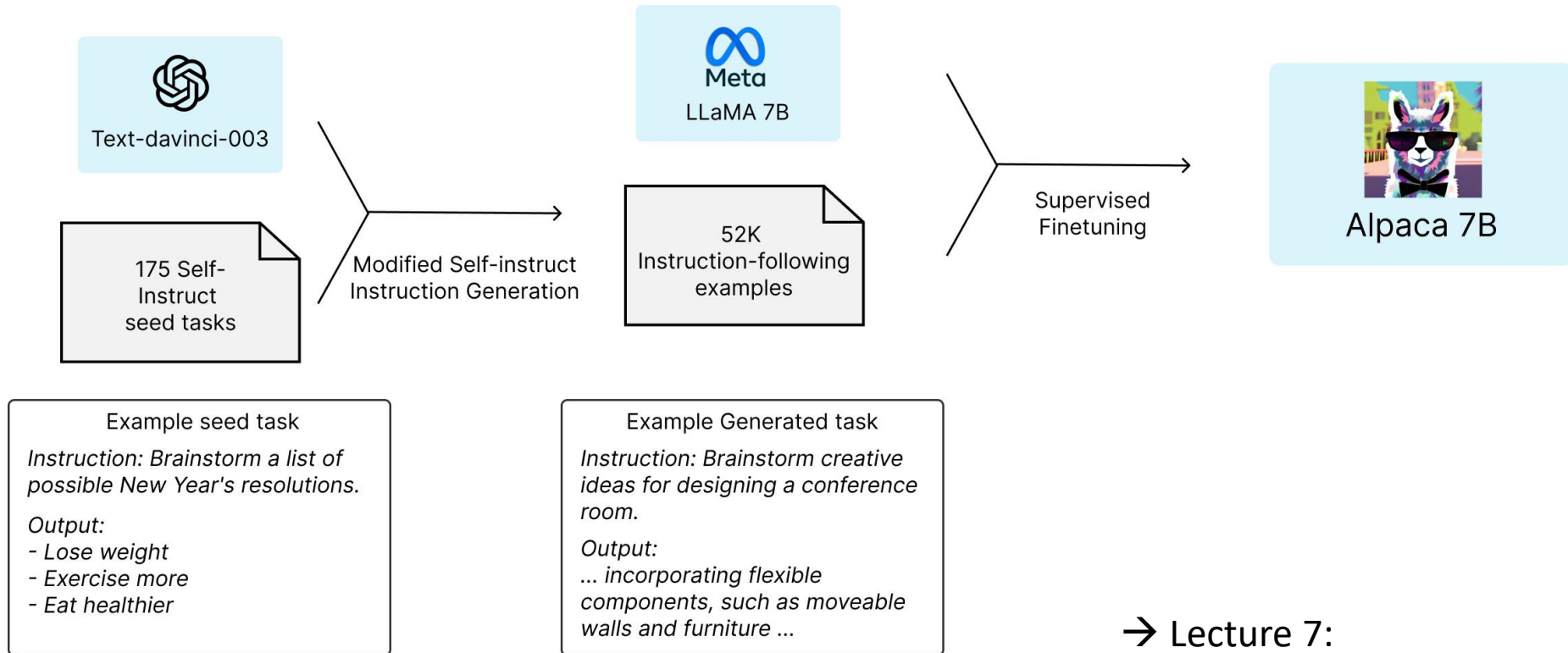
Based on the above text, generate a set of short-form QA-pairs.

→ Let's try

* Works only for selected (asymmetric?) tasks, e.g., QA and spelling correction, but not for summarization, translation, etc.

Intuition 2:

Stronger model → Weaker model



→ Lecture 7:
Knowledge
distillation

You are asked to come up with a set of 20 diverse task instructions. These task instructions will be given to a GPT model and we will evaluate the GPT model for completing the instructions.

Here are the requirements:

1. Try not to repeat the verb for each instruction to maximize diversity.
2. The language used for the instruction also should be diverse. For example, you should combine questions with imperative instructions.
3. The type of instructions should be diverse. The list should include diverse types of tasks like open-ended generation, classification, editing, etc.
2. A GPT language model should be able to complete the instruction. For example, do not ask the assistant to create any visual or audio output. For another example, do not ask the assistant to wake you up at 5pm or set a reminder because it cannot perform any action.
3. The instructions should be in English.
4. The instructions should be 1 to 2 sentences long. Either an imperative sentence or a question is permitted.
5. You should generate an appropriate input to the instruction. The input field should contain a specific example provided for the instruction. It should involve realistic data and should not contain simple placeholders. The input should provide substantial content to make the instruction challenging but should ideally not exceed 100 words.
6. Not all instructions require input. For example, when a instruction asks about some general information, "what is the highest peak in the world", it is not necessary to provide a specific context. In this case, we simply put "<noinput>" in the input field.
7. The output should be an appropriate response to the instruction and the input. Make sure the output is less than 100 words.

List of 20 tasks:

https://github.com/tatsu-lab/stanford_alpaca/blob/main/prompt.txt

Intuition 3: Scoring easier than generation

- Fundamental asymmetry in many knowledge tasks
 - Analogy to complexity class NP!
- Try it yourself:
Write a Haiku about flying toast slices with jam.
- Which one is better?

Option 1	Option 2	Option 3
The light of a candle is transferred to another candle spring twilight	Golden bread takes flight, jam gleams like dawn on its wings breakfast meets the sky.	Toast lifts from the plate, jam gleams red in morning light, birds chase crumbs across the wind, laughing.

- LLMs can evaluate/compare
- Stylistic features like verbosity, formality, grammaticality
 - Semantic features like topical match
 - Factuality relative to a reference passage

Outline

1. Importance of data (10')
2. Pre-training (50')
 1. Crawling
 2. Scraping and cleaning
 3. Datasets
3. Instruction tuning (5')
4. Alignment (5')
5. Synthetic data (10')
6. **Discussion (10')**
 - Group work

Discussion: Benchmark leakage

- Are newer LLMs really better?
- Or do they just **train on the test data**?

NLP Evaluation in trouble: On the Need to Measure LLM Data Contamination for each Benchmark

Oscar Sainz¹ Jon Ander Campos² Iker García-Ferrero¹ Julen Etxaniz¹
Oier Lopez de Lacalle¹ Eneko Agirre¹

¹ HiTZ Center - Ixa, University of the Basque Country UPV/EHU
{oscar.sainz,iker.graciaf,julen.etxaniz}@ehu.eus
{oier.lopezdelacalle,e.agirre}@ehu.eus

² Cohere
jonander@cohere.com

Abstract

In this position paper, we argue that the classical evaluation on Natural Language Processing (NLP) tasks using annotated benchmarks is in trouble. The worst kind of data contamination happens when a Large Language Model (LLM) is trained on the test split of a benchmark, and then evaluated in the same benchmark. The extent of the problem is unknown, as it is not straightforward to measure. Contamination causes an overestimation of the performance of a contaminated model in a target benchmark and associated task with respect to their non-

et al., 2020) the need for data has been solved by crawling the internet, reaching trillions of tokens (Touvron et al., 2023a), and making it **very hard to know whether a specific benchmark was used to train the LLM**. This is applicable to all models, even if they document the source of the data at a high level, but especially for closed models with no or insufficient documentation.

Data contamination has two consequences. The first one is that the performance of an LLM when evaluated on a benchmark it already processed during pre-training will be overestimated, causing it

→ Lecture
“Evaluation”

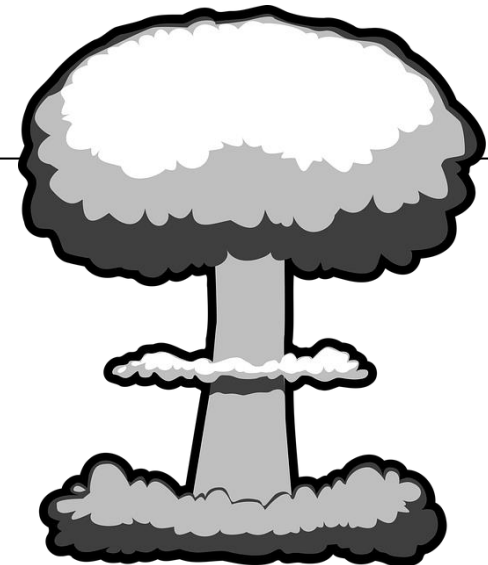
Will LLM-generated content destroy their own training sources?

AI models collapse when trained on recursively generated data

[Ilia Shumailov](#) ✉, [Zakhar Shumaylov](#) ✉, [Yiren Zhao](#), [Nicolas Papernot](#), [Ross Anderson](#) & [Yarin Gal](#) ✉

[Nature](#) **631**, 755–759 (2024) | [Cite this article](#)

564k Accesses | **353** Citations | **3460** Altmetric | [Metrics](#)



Group work

- Discuss the following questions with your neighbor:
 1. Which **three kinds** of training data do LLMs need?
 2. **Where** can we get each **from**?
What can we **synthesize**?
 3. What are **important filters** on web data?

Take home – Lecture 5

1. **Data** is one of the **main differentiators** between LLMs
 - Major commercial secret
 2. LLM training broadly requires **three types of data**
 - Unsupervised pre-training data, instruction fine-tuning data, answer alignment data
 3. **Pre-training data** is typically **web-crawled**
 - Filtering important
 - Definition of “good” data is not clear
 4. **Instruction** and alignment data often **crowdsourced**
 - Synthesizing helps
- Lab today:
 - TBD