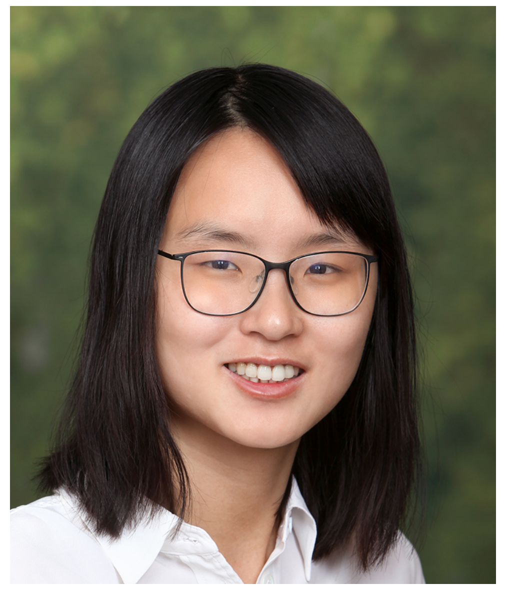
COS 597G: Understanding Large Language Models

PRINCETON UNIVERSITY

Fall 2022

Logistics



- **Instructor**: Danqi Chen
- **Teaching assistant**: Alexander Wettig

- **Location**: Sherrerd Hall 101
- **Meetings**: Monday/Wednesday 10:30-11:50am

- **Office hours:**
  - Danqi's office hour: Monday 2:30-3:30pm (appointment-based, 15 minutes each)
  - Alex's office hour: Wednesday 3-4pm (friend center student space)

Logistics

Website: https://www.cs.princeton.edu/courses/archive/fall22/cos597G/

Instructor	Danqi Chen (danqic AT cs.princeton.edu)
Teaching assistant	Alexander Wettig (awettig AT cs.princeton.edu)
Lectures	Monday/Wednesday 10:30-11:50am
Location	Sherrerd Hall 101
Pre-lecture feedback meetings	Wednesday/Friday, 4:45pm-5:15pm, COS 412
Office hours	Danqi's office hour: Monday 3-4pm, COS 412 (by appointment) Alex's office hour: Wednesday 3-4pm, Friend Center (student space lobby)
Feedback form	https://forms.gle/rkJVxY8fvn7pchv89

anonymous feedback form

OH link

Schedule

Date	Topic/papers	Recommended reading	Pre-lecture questions	Presenters	Feedback providers
Sep 7 (Wed)	**Introduction**	1. Human Language Understanding & Reasoning 2. Attention Is All You Need (Transformers) 3. Blog Post: The Illustrated Transformer 4. HuggingFace's course on Transformers	-	Danqi Chen	-
		What are LLMs?			
Sep 12 (Mon)	**BERT/RoBERTa (encoder-only models)** 1. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	1. RoBERTa: A Robustly Optimized BERT Pretraining Approach 2. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators		Danqi Chen	
Sep 14 (Wed)	**T5/BART (encoder-decoder models)** 1. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5)	1. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension			

Note: We will maintain the website for schedule, lecture slides etc, not Canvas!

Logistics

- We will use Slack as the primary mode of communication (**no Ed!**). For important announcements (e.g., deadlines), I will still write emails.
- We will send an invitation to all the enrolled students later today.
- Why Slack?
  - We prefer Slack messages over emails for all logistical questions. Please just DM me (@danqi) and/or Alex (@Alexander Wettig) instead of emails!
  - We will use Slack to provide feedback on your lectures (more on this later)
  - We will use Slack to clarify lecture-related questions and share more links and papers!
  - If you like, you can also use this Slack team for your project communications
  - More importantly, we strongly encourage you ask questions, share random musings, highlight interesting papers, brag about cool findings, even send stable diffusion pics or GPT-3 examples (use #random!).

Course structure

- This is **an advanced graduate course** and we will be teaching and discussing state-of-the-art papers about large language models
- The course is mostly presentation- and discussion-based and all the students are expected to come to the class regularly and participate in discussion
- Prerequisites: COS484/584 or similar
  - Familiarity with neural networks and Transformer models (encoder, decoder, encoder-decoder)
  - Familiarity with basic NLP tasks, including understanding (text classification, textual entailment, question answering) and generation (translation, summarization) tasks

Recommended reading

1. Human Language Understanding & Reasoning
2. Attention Is All You Need (Transformers)
3. Blog Post: The Illustrated Transformer
4. HuggingFace's course on Transformers

Learn about Transformers yourself!

Course structure

- 20 lectures in total + 2 guest lectures + 1 in-class presentation (see a draft schedule on the website)

Required reading: everyone needs to read them before the class and answer pre-lecture questions

Each lecture has 2 presenters and 3-4 feedback providers

Date	Topic/papers	Recommended reading	Pre-lecture questions	Presenters	Feedback providers
Oct 24 (Mon)	**Scaling** 1. Training Compute-Optimal Large Language Models	1. Scaling Laws for Neural Language Models 2. Emergent Abilities of Large Language Models			
Oct 26 (Wed)	**Privacy** 1. Extracting Training Data from Large Language Models	1. Quantifying Memorization Across Neural Language Models 2. Deduplicating Training Data Mitigates Privacy Risks in Language Models			
Oct 31 (Mon)	**Bias & Toxity** 1. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models 2. OPT paper, Section 4	1. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 2. Whose Language Counts as High Quality? Measuring Language Ideologies in Text Data Selection			

Recommended if you are interested in researching more on the topic!

We also encourage the presenter to consider incorporating some “recommended reading” into their lecture

Components and grading

- 25% class participation
  - Read paper(s) before the class and answer 2+1 pre-lecture questions
  - Come to the class and participate in discussion
- 30% presentation
  - 1-2 times throughout the semester
- 5% lecture feedback
  - 2-3 times throughout the semester
- 40% final project

Class presentation

- **Two students** work together and deliver a 60-minute presentation
- You should cover *at least* the required paper(s) and your goal is to educate others in the class about the **topic** of the class
    - Add background and context!
    - Add more discussion and related work ("recommended reading") when you see a fit
- It is your job to decide how to best cover the material and how to divide the work
- Lecture preparation meetings (COS 412):
    - **Monday** 3:30-4 for lectures on **Wednesday**
    - **Friday** 4:45-5:15 for lectures next **Monday**
    - Send me your draft slides on Slack before 11:59pm the night before
    - Google slides are encouraged (easier for comments); Keynote/Powerpoint are fine too
    - Hint: add slide numbers for comments and feedback!
- We are always happy to provide comments and give suggestions on Slack (just DM us!)

What is a good presentation?

- The oldest paper we are going to read is 4 years old ☺ Most papers were published in the past 1-2 years (if not a few months).
- They represent the state-of-the-art of the field and our *current understanding* of LLMs
- All research builds on previous research: putting things in context when you read and present a paper!! You are expected to read more if you want to fully understand papers
- When you present a paper:
  - Highlight the biggest take-aways
  - Think about why this paper is important
  - Pay attention to technical details too
  - Choose and present experimental results properly
  - (Bonus) what can be done in the future?

Pre-lecture questions

- Questions posted on the website (a Google form link)
- Due at 11:59pm the night before the lecture

You should be able to answer these questions after you read the paper(s)

- Q1. What is the biggest improvement of BERT compared to GPT?
- Q2. Why does BERT keep the masked tokens unchanged for 10%?

- Q3. If we scale up BERT by 1000x, would it be still better than unidirectional models? Why do you think the largest models to date are always unidirectional?

A more open-ended question: we want to collect your thoughts before the class too and leave time for discussion

Peer feedback

For each lecture, we will have 3-4 students to provide feedback on the lectures

- Send the feedback (again, Slack!) to me within a day of the presentation
- Comments on clarity, structure, completeness, slides ..
- Offer constructive criticism but also suggestions
- No need for complete sentences, bullet points are fine

Meeting format

- 60-minute lecture
  - Be prepared for lots of questions (and we encourage questions in the class)
  - Please control your time (rehearsal is very helpful)!

- 20-minute reviewing and discussing pre-lecture questions
  - Divide the class into groups of 4-5 students (depending on seating and enrollment)
  - 10 minutes: Discuss the open-ended question
  - 10 minutes: Decide on a leader in each group and present a 1-minute summary

Final project

- Students complete a research project in teams of 2-3
- Proposal deadline: Oct 14 11:59pm
- In-class presentation: Dec 5th - we are likely to extend the class
- Final paper deadline: Dec 16th (dean's date)
- Two typical types of projects:
    - #1: Train or fine-tune a medium-sized language model (e.g., BERT/RoBERTa, T5) yourself for any problem of your interest. Check out HuggingFace's model hub!

      https://huggingface.co/models

      Hugging Face

    - #2: Evaluate one of the largest language models (e.g., GPT-3, Codex) and understand their capabilities, limitations and risks.

      https://openai.com/api/

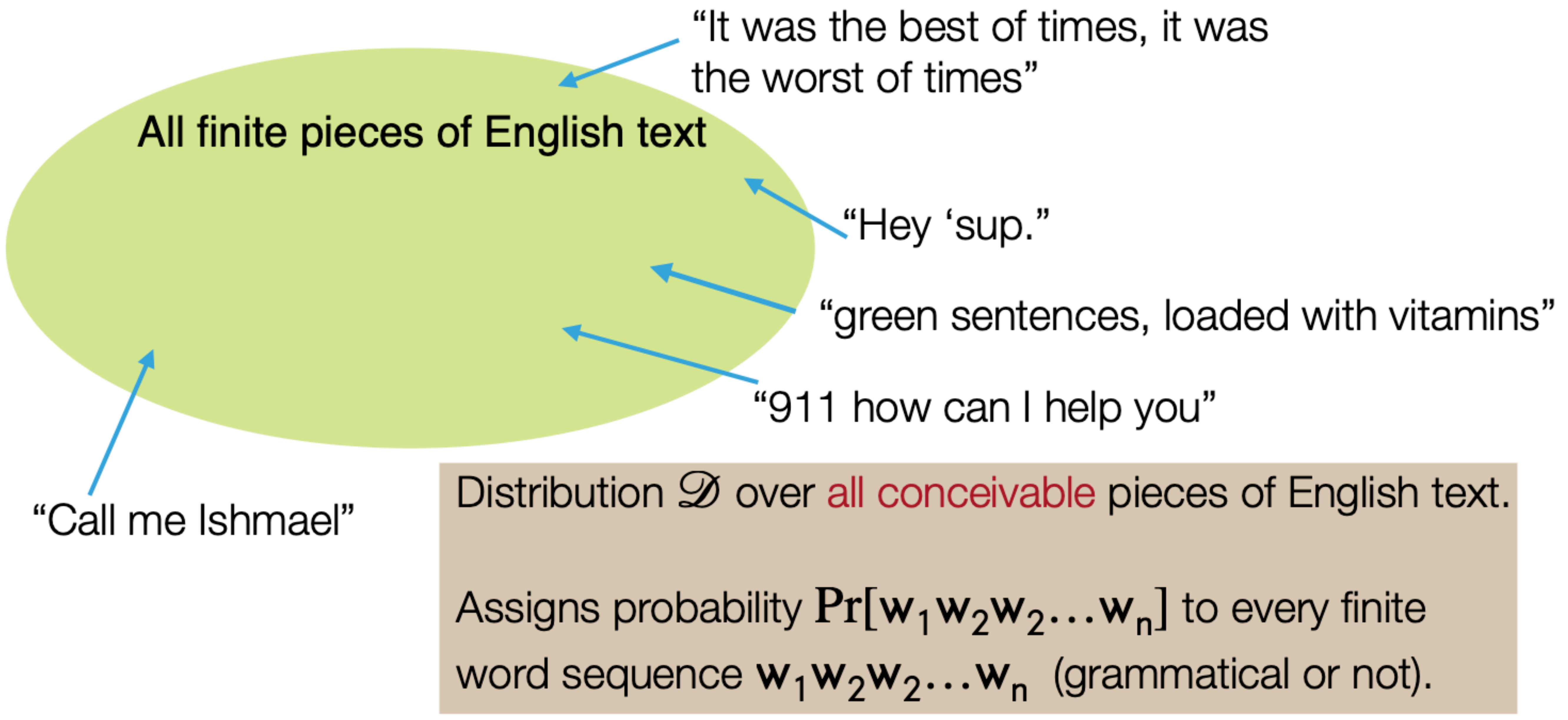
      https://opt.alpa.ai

      Note: we will provide certain budget for compute needs or access to LLMs. More coming soon!

What are large language models (LLMs)?

Language models: narrow sense

- A probabilistic model that assigns a probability $P[w_1, w_2, \ldots, w_n]$ to every finite sequence $w_1, \ldots, w_n$ (grammatical or not)



Source: COS 324

Language models: narrow sense

Conditional probability

$$p(w_1, w_2, w_3, \ldots, w_N) =$$
$$p(w_1)\ p(w_2|w_1)\ p(w_3|w_1, w_2) \times \ldots \times p(w_N|w_1, w_2, \ldots w_{N-1})$$

Sentence: "the cat sat on the mat"

$$P(\text{the cat sat on the mat}) = P(\text{the}) * P(\text{cat}|\text{the}) * P(\text{sat}|\text{the cat})$$
$$*P(\text{on}|\text{the cat sat}) * P(\text{the}|\text{the cat sat on})$$
$$*P(\text{mat}|\text{the cat sat on the})$$

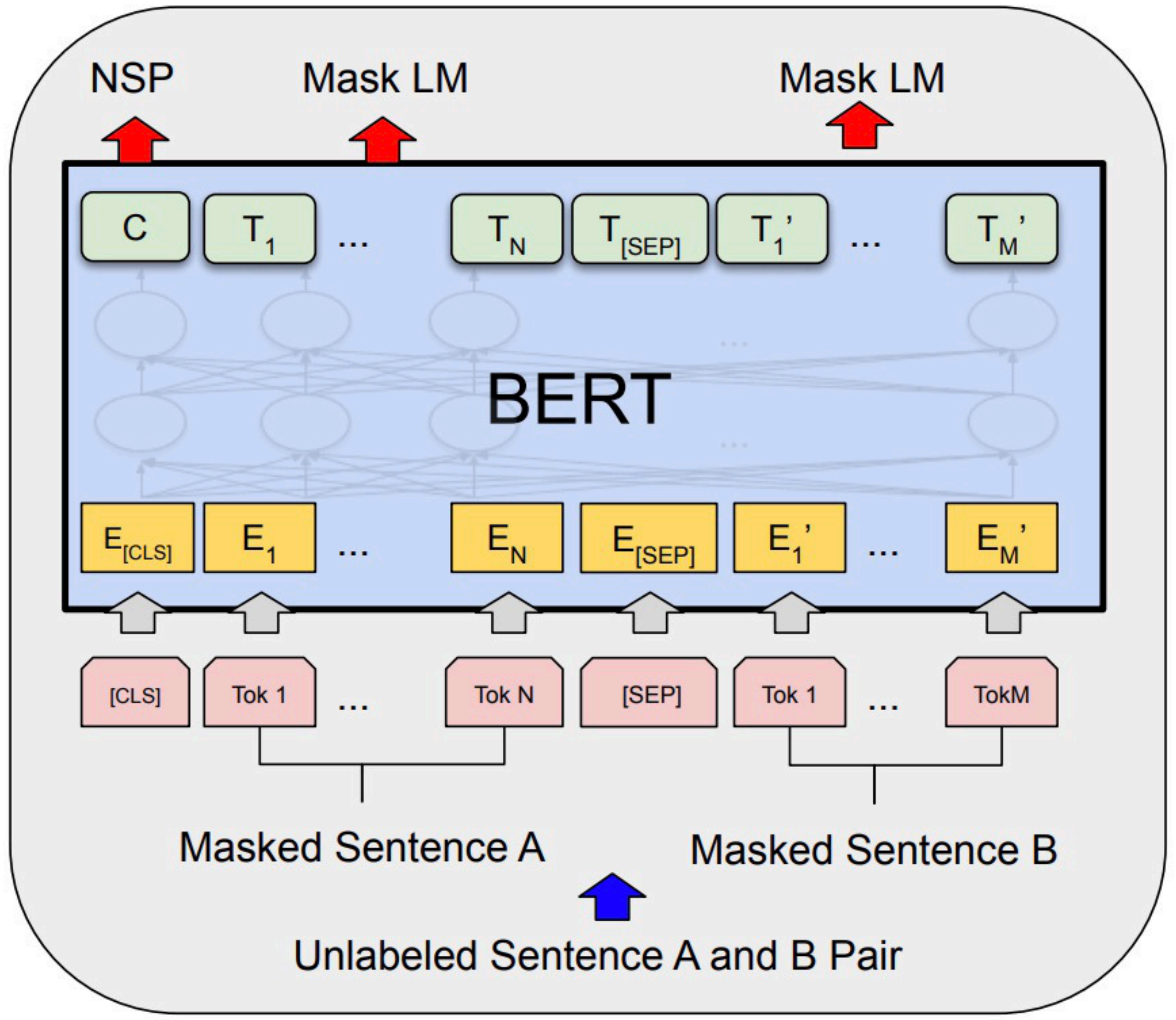
Implicit order

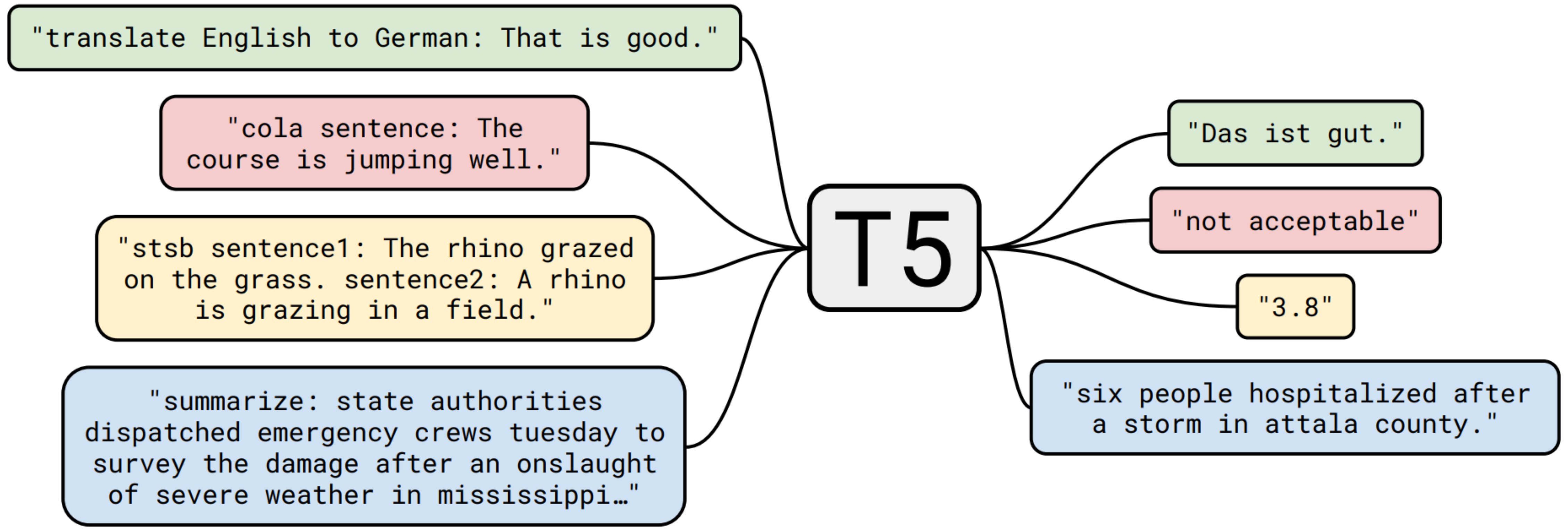
Source: COS 484

GPT-3 still acts in this way but the model is implemented as a very large neural network of 175-billion parameters!

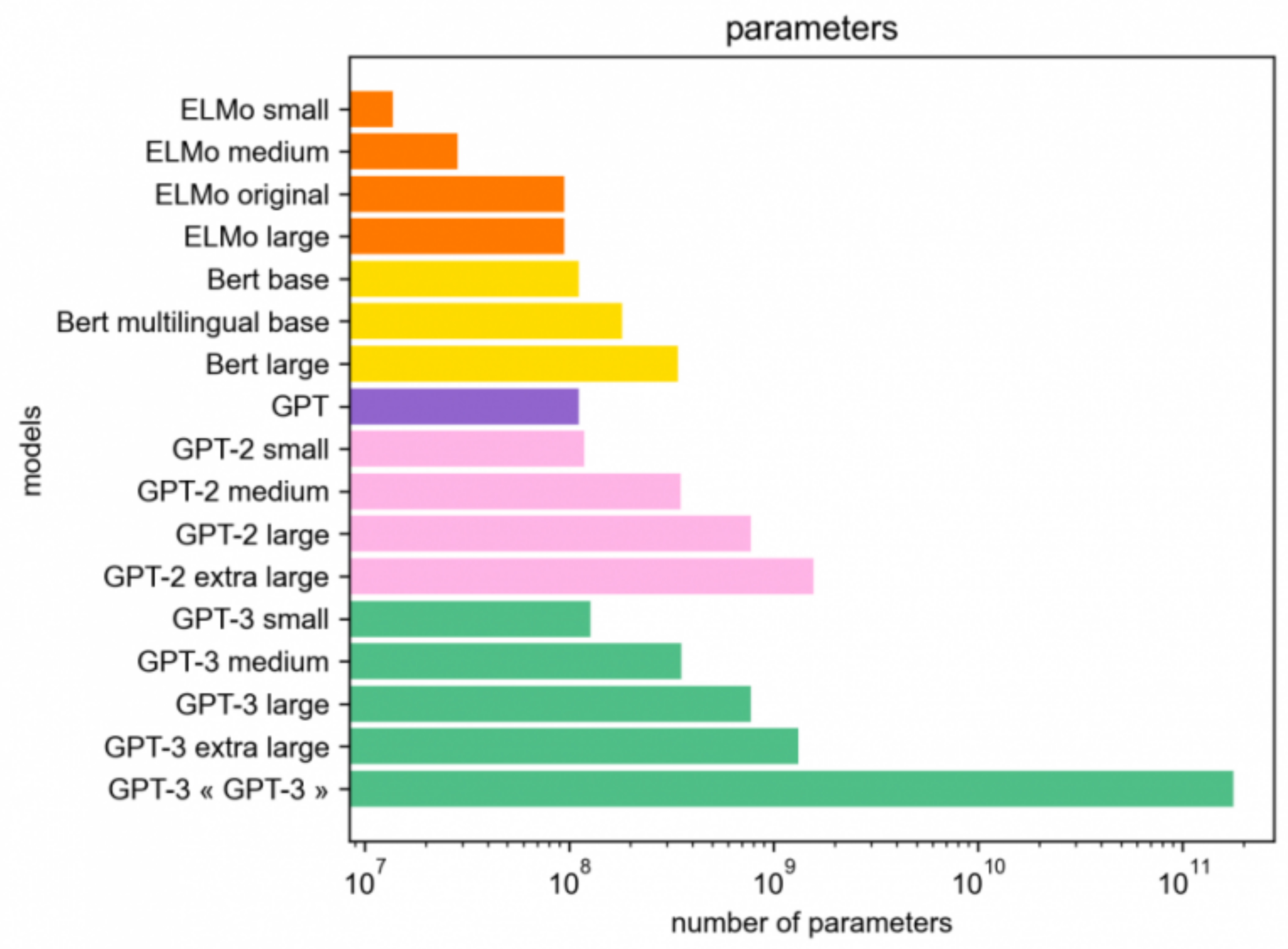
Language models: broad sense

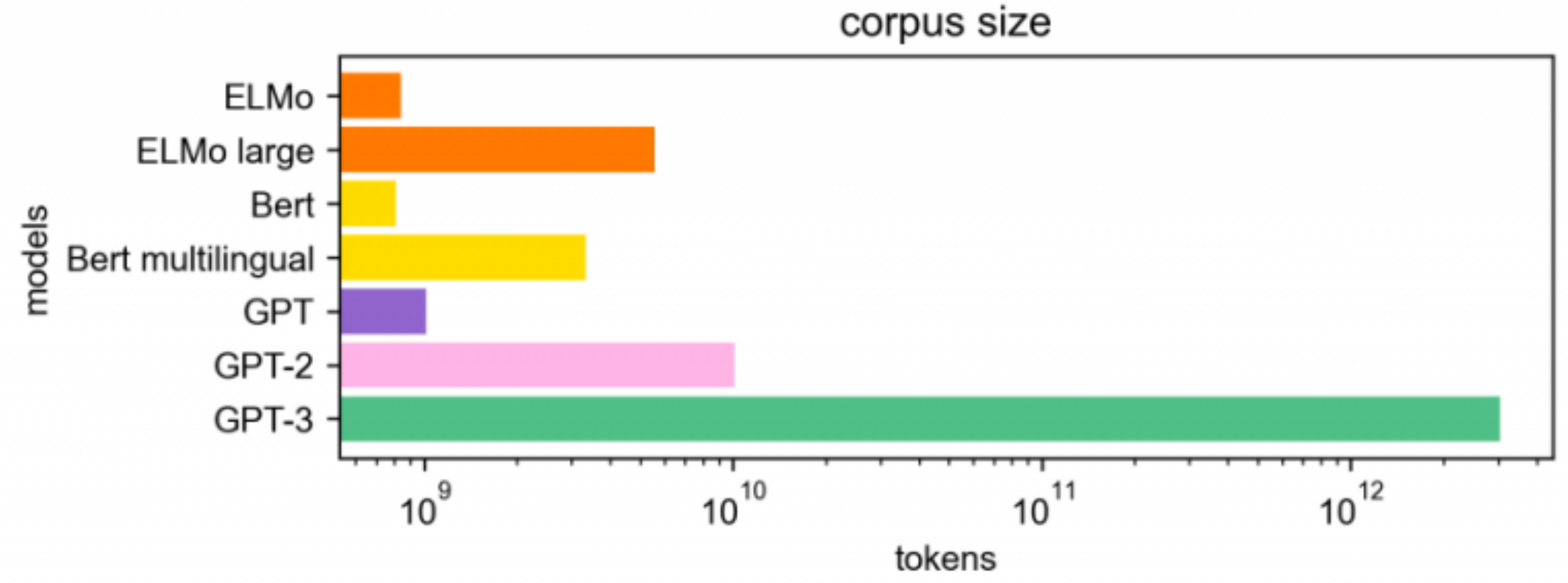
- Decoder-only models (GPT-x models)
- Encoder-only models (BERT, RoBERTa, ELECTRA)
- Encoder-decoder models (T5, BART)





How large are "large" LMs?





More recent models: PaLM (540B), OPT (175B), BLOOM (176B)…

Image source: https://hellofuture.orange.com/en/the-gpt-3-language-model-revolution-or-evolution/

How large are "large" LMs?

- Today, we mostly talk about two camps of models:
  - Medium-sized models: BERT/RoBERTa models (100M or 300M), T5 models (220M, 770M, 3B)
  - "Very" large LMs: models of 100+ billion parameters

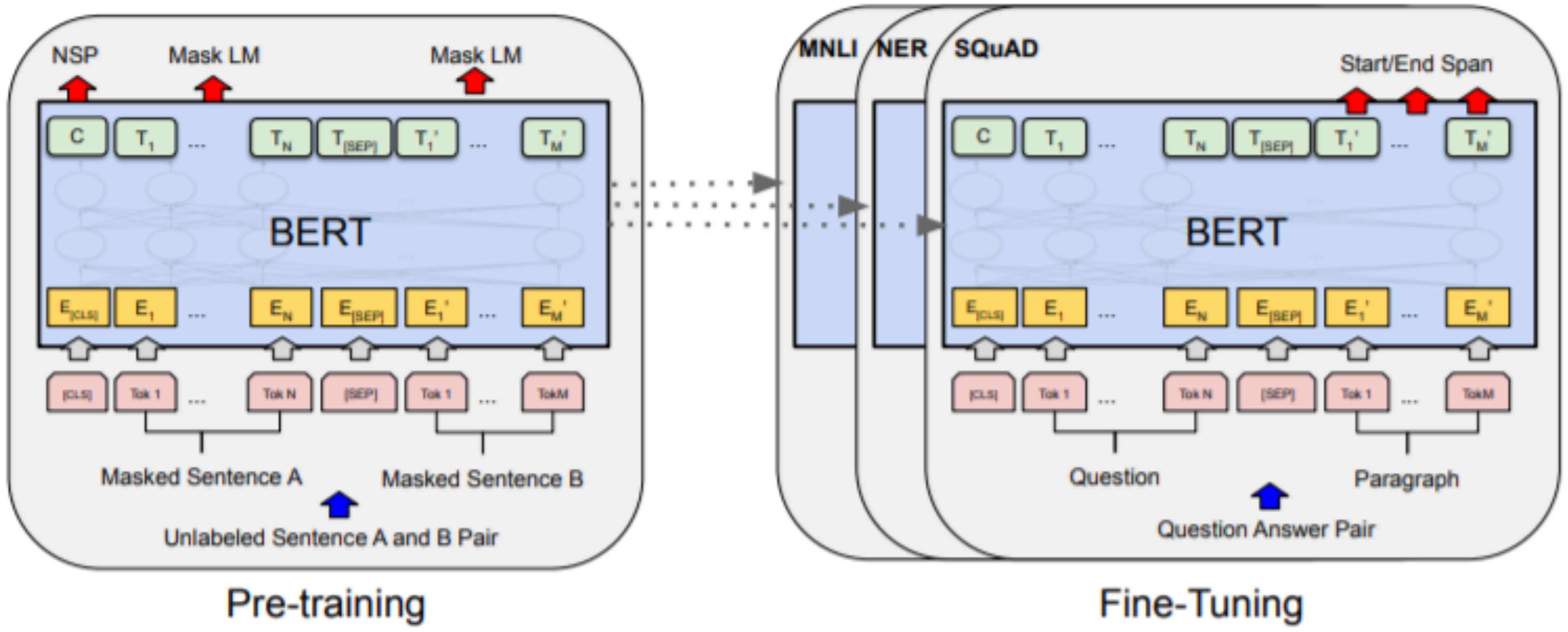
- Larger model sizes ⇒ larger compute, more expensive during inference
- Different sizes of LMs have different ways to adapt and use them
  - Fine-tuning, zero-shot/few-shot prompting, in-context learning…
- Emergent properties arise from model scale
- Trade-off between model size and corpus size

Q: Do largest models always give the best performance today?

Pre-training and adaptation

- **Pre-training**: trained on huge amounts of unlabeled text using “self-supervised” training objectives

- **Adaptation**: how to use a pre-trained model for your downstream task?
    - What types of NLP tasks (input and output formats)?
    - How many annotated examples do you have?



Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // ________

LM

Circulation revenue has increased by 5% in Finland. // Finance

They defeated … in the NFC Championship Game. // Sports

Apple … development of in-house chips. // Tech

The company anticipated its operating profit to improve. // ________

LM

http://ai.stanford.edu/blog/understanding-incontext/

Why LLMs?

- The promise: one single model to solve many NLP tasks

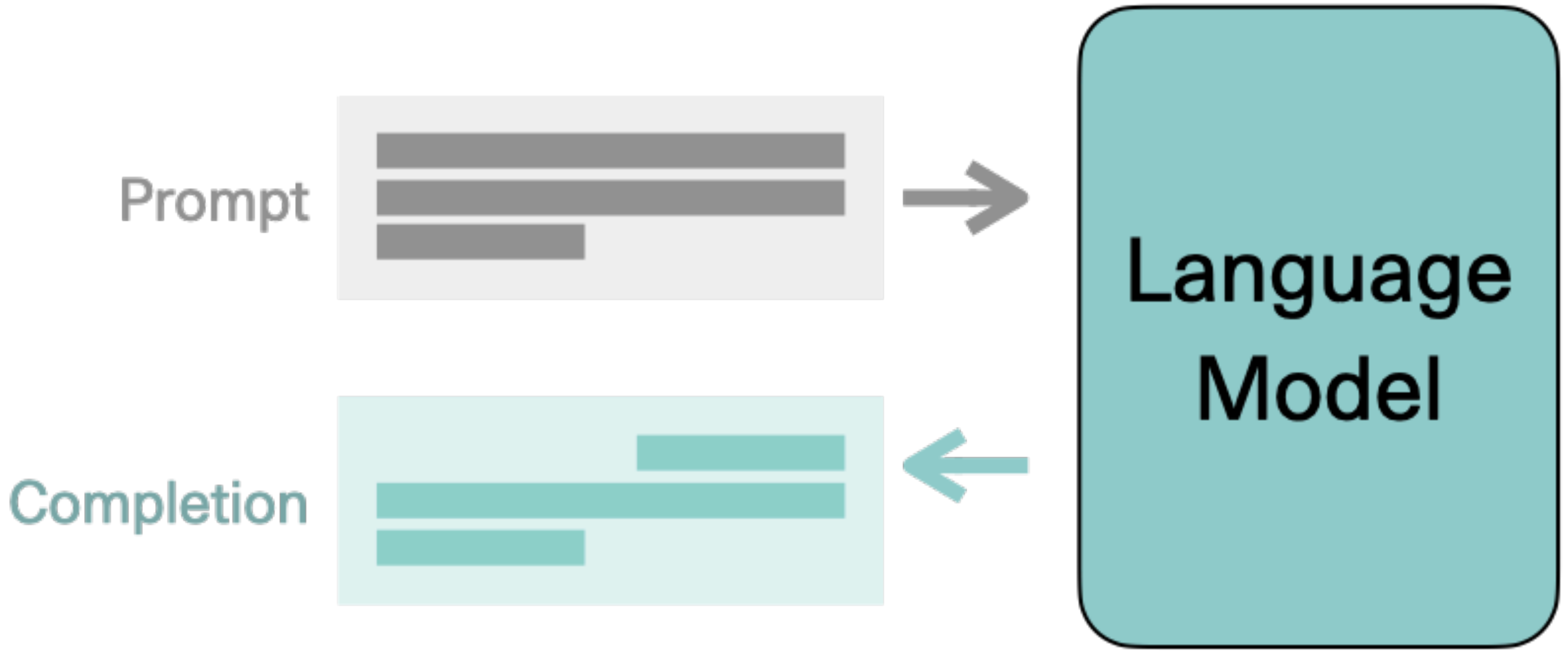
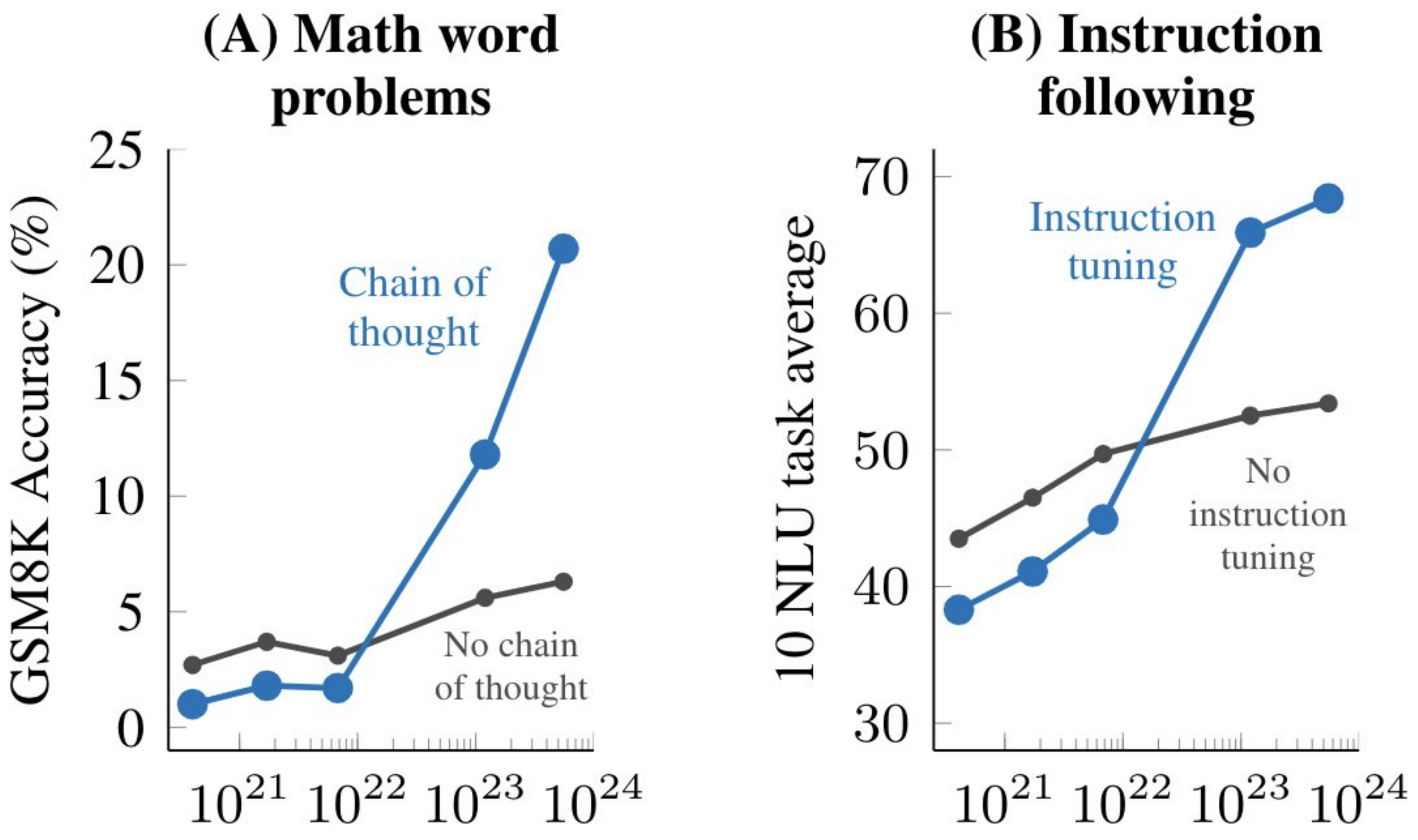


Image credit: Jay Alammar

- Emergent properties in LLMs



(Wei et al., 2022)

What are we going to cover in the class?

Part I. What are LLMs (3 lectures)

What are LLMs?					
Sep 12 (Mon)	**BERT/RoBERTa (encoder-only models)** 1. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	1. RoBERTa: A Robustly Optimized BERT Pretraining Approach 2. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators		Danqi Chen	
Sep 14 (Wed)	**T5/BART (encoder-decoder models)** 1. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5)	1. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension			
Sep 19 (Mon)	**GPT-3 (decoder-only models)** 1. Language Models are Few-Shot Learners (GPT-3)	1. Language Models are Unsupervised Multitask Learners (GPT-2) 2. PaLM: Scaling Language Modeling with Pathways 3. OPT: Open Pre-trained Transformer Language Models			

Part II. How to use and adapt LLMs (6 lectures)

Sep 21 (Wed)	**Prompting for few-shot learning** 1. Making Pre-trained Language Models Better Few-shot Learners (blog post) 2. How Many Data Points is a Prompt Worth?	1. True Few-Shot Learning with Language Models 2. Cutting Down on Prompts and Parameters: Simple Few-Shot Learning with Language Models			
Sep 26 (Mon)	**Prompting as parameter-efficient fine-tuning** 1. Prefix-Tuning: Optimizing Continuous Prompts for Generation 2. The Power of Scale for Parameter-Efficient Prompt Tuning	1. LoRA: Low-Rank Adaptation of Large Language Models 2. Towards a Unified View of Parameter-Efficient Transfer Learning			
Sep 28 (Wed)	**In-context learning** 1. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? 2. An Explanation of In-context Learning as Implicit Bayesian Inference (we don't expect you to read this paper in depth, you can check out this blog post instead)	1. What Makes Good In-Context Examples for GPT-3? 2. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity			

Prompting, in-context learning

Part II. How to use and adapt LLMs (6 lectures)

Oct 3 (Mon)	**Calibration of prompting LLMs** 1. Calibrate Before Use: Improving Few-Shot Performance of Language Models 2. Surface Form Competition: Why the Highest Probability Answer Isn't Always Right	1. Noisy Channel Language Model Prompting for Few-Shot Text Classification			
Oct 5 (Wed)	**Reasoning** 1. Chain of Thought Prompting Elicits Reasoning in Large Language Models 2. Large Language Models are Zero-Shot Reasoners	1. Explaining Answers with Entailment Trees 2. Generating Natural Language Proofs with Verifier-Guided Search 3. Faithful Reasoning Using Large Language Models			
Oct 10 (Mon)	**Knowledge** 1. Language Models as Knowledge Bases? 2. How Much Knowledge Can You Pack Into the Parameters of a Language Model?	1. Fast Model Editing at Scale			

Calibration, reasoning, knowledge

Part III. Dissecting LLMs: data, model scaling and risks (5 lectures)

Oct 12 (Wed)	**Data** 1. Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus	1. The Pile: An 800GB Dataset of Diverse Text for Language Modeling 2. Deduplicating Training Data Makes Language Models Better
Oct 24 (Mon)	**Scaling** 1. Training Compute-Optimal Large Language Models	1. Scaling Laws for Neural Language Models 2. Emergent Abilities of Large Language Models
Oct 26 (Wed)	**Privacy** 1. Extracting Training Data from Large Language Models	1. Quantifying Memorization Across Neural Language Models 2. Deduplicating Training Data Mitigates Privacy Risks in Language Models
Oct 31 (Mon)	**Bias & Toxity** 1. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models 2. OPT paper, Section 4	1. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 2. Whose Language Counts as High Quality? Measuring Language Ideologies in Text Data Selection
Nov 2 (Wed)	**Bias & Toxity II** 1. Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP	1. Challenges in Detoxifying Language Models 2. Detoxifying Language Models Risks Marginalizing Minority Voices

Data, scaling, privacy, bias, toxity

Part IV. Beyond Current LLMs: Models and Applications (5 lectures)

Nov 7 (Mon)	**Sparse models** 1. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity	1. Efficient Large Scale Language Modeling with Mixtures of Experts 2. Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models
Nov 9 (Wed)	**Retrieval-based LMs** 1. Improving language models by retrieving from trillions of tokens	1. Generalization through Memorization: Nearest Neighbor Language Models 2. Training Language Models with Memory Augmentation 3. Few-shot Learning with Retrieval Augmented Language Models
Nov 14 (Mon)	**Training LMs with human feedback** 1. Training language models to follow instructions with human feedback	1. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback 2. LaMDA: Language Models for Dialog Application
Nov 16 (Wed)	**Code Models** 1. Evaluating Large Language Models Trained on Code	1. A Conversational Paradigm for Program Synthesis 2. InCoder: A Generative Model for Code Infilling and Synthesis
Nov 21 (Mon)	**Multimodal LMs** 1. Flamingo: a Visual Language Model for Few-Shot Learning	1. Learning Transferable Visual Models From Natural Language Supervision (CLIP)

Sparse/retrieval-based models

Training LMs with human feedback

Code models/visual LMs

Presentation & feedback scheduling

- We will send out a sign-up form tonight (your priority of topics and blackout dates)
- **Complete it by Friday evening 11:59pm** and we will finalize the schedule by the end of the week

- If you are still not sure whether you will stay in this course, please let us know ASAP
  - We still have a number of people on the waitlist
  - We can't let you take the course if you don't get a presentation slot, sorry ☹

- We need two volunteers for next Wednesday's lecture (topic: T5/BART)!

Sep 14 (Wed)	**T5/BART (encoder-decoder models)** 1. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5)	1. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension

Discussion: What are you most excited about LLMs and want to learn from the class?