Natural Language Processing (CSE 517): Language Model-Based NLP

Noah Smith

University of Washington
nasmith@cs.washington.edu

Winter 2023

Acknowledgments: Yanai Elazar, Hila Gonen, Suchin Gururangan, Yushi Hu, Luke Zettlemoyer.

Reading: "ChatGPT is a blurry JPEG of the Web." Ted Chiang, *New Yorker*.
https://www.newyorker.com/tech/annals-of-technology/chatgpt-is-a-blurry-jpeg-of-the-web

How do we use language models?

(This lecture is a biased and incomplete summary of what's happening today in NLP.)

Three Variants of LM-Based NLP

1. LM as **encoder** of text, usually for analysis tasks (not generation); can be used to learn representations where information flows from all input tokens to all others. Typically finetuned with supervised learning. Widely deployed already. Example: BERT.
2. LM as **encoder and decoder** of text. Suitable for sequence-to-sequence tasks. Examples: modern machine translation models, T5.
3. LM as **decoder** of text, suitable for generation tasks (but also analysis); getting a lot of attention now. Example: GPT-3.

For all of these, there are many different training procedures, inference procedures, and evaluation frameworks!

Scale

The general trend over the past few years has been larger models (in terms of layers, number of parameters, etc.) pretrained on larger datasets; hence the term "large language model." While there are new models coming out all the time, there aren't enough systematic experiments, or consistent evaluations, to fully understand all the factors in LM success.

Open question: Is it possible to achieve results on par with the best seen, but with less data, less parameters, less computationally-intensive training?

I think the data matters most, and quality may be more important than quantity, for some definition of quality. See Chinchilla (Hoffmann et al., 2022) and Llama (Touvron et al., 2023).

The Cost of Scale

Costs (thanks to Luke Zettlemoyer), not including data preparation, comparison runs at smaller scale, architecture, experiments, etc.:

	params.	hardware	days	est. AWS cost
GPT-3 (OpenAI)	175B	1500 GPUs	60	$3M
PaLM (Google)	540B	6144 TPUs	57	$25M

Latest: Llama (Meta), which is somewhat more open.

Finetuning

Recall that, since ELMo, the best performance for LM-based models has usually been obtained by applying supervised learning, initialized by the pretrained LM, on task-specific data.

That's still mostly true today, but fully finetuning the largest models is beyond most researchers' budgets. (Research topic: efficient finetuning, e.g., with adaptors; Houlsby et al., 2019.)

The extremely long tail of possible NLP applications has become impossible to ignore, as we observe people interacting with ChatGPT (more about this later). Perhaps we're moving away from supervised learning on large samples to ... something else.

Prompting

The simplest kind of generation from an LM is to continue from a prompt:

$$\underset{\boldsymbol{x} \in \mathcal{V}^{\dagger}}{\operatorname{argmax}}\, p_{\mathrm{LM}}(\boldsymbol{x} \mid \boldsymbol{x}_{prompt})$$

(Various decoding algorithms, from greedy to beam search.)

Over the past five years, the LMs have become increasingly *fluent*, even under this simple use case.

As a result, considerable research on using LMs essentially on their own to do NLP. Early use case: testing LMs for factual “knowledge” by prompting them (Petroni et al., 2019). How far can we go with prompting?

Prompting

Consider a task in which we want to map inputs to outputs. Some evaluation settings:

- "Zero shot": encode the input as a sequence, feed to the LM as a prompt, treat the continuation as output. E.g., "Translate from English to French: I'm hungry."
- "Few shot": like zero shot, but include a few input-output pairs to the prompt.

This trend was set off by GPT-3 (Brown et al., 2020). People sometimes call this setup "in-context learning," but this is rather different from conventional machine learning (where parameters get updated, for example).

It's very appealing because there's no specialized training required! Once you build your LM, it can do anything.*

*Anything you can encode in a prompt. And not usually as well as a finetuned model where training data is available.

Instruction Tuning

Key idea: finetune a LM to be better at interpreting prompts that include explicit instructions about what you want.

To do this, we need data that includes instructions (in natural language) alongside input-output pairs. (Supervision returns!)

- Example models: Flan-T5 (Chung et al., 2022)
- Example dataset: Supernatural-Instructions (Wang et al., 2022)

Interesting result: can get strong performance with much smaller models (Schick and Schütze, 2021)!

A Parallel Development: Pretraining on Code

Applying language models to code has led to exciting developments in tools for programmers.

Example model: Codex (Chen et al., 2021)

It's now standard to include some code in LMs trained primarily on natural language text. Some attribute the models' apparent reasoning capabilities to this (Madaan et al., 2022).

Data, Again

In general, more data leads to better models.

There's not a lot of discussion about how data is selected, except:

- Including source code, as discussed.
- Multilingual datasets seem to lead to some crosslingual transfer capabilities.
- "Quality filters" are sometimes applied, but these can have negative consequences, too (Gururangan et al., 2022).

In general, the datasets are too big for anyone to fully understand what's in them.

ChatGPT

As best we can tell, ChatGPT starts with GPT-3 (or a similarly strong base), pretrained LM (on text and code). It's then finetuned to follow instructions (proprietary data, not instructions for classic NLP tasks), then finetuned on human feedback (e.g., learn a reward function from annotator judgments, then apply to the LM through reinforcement learning; Ouyang et al., 2022).

What Kinds of Things Does ChatGPT Do?

- Generates fresh text (reports, letters, stories, poems, etc.) that you request, on any (?) topic.
- Summarizes long text in shorter form.
- Revises text to have a different style.
- Generates code to your specifications.
- Maintains some coherence across iterations ("turns" of a "conversation").
- If you point out problems in what it gives you, it might apologize and try again. Or antagonize you.

Concerns

Public misunderstanding

“Hallucinations” (bullshit)

Toxic language

Privacy

Data rights

The LM as a Component, Again

Despite the many surprising things models like ChatGPT can say, it's important to remember that they are **fluency machines**; they do not model truth, correctness, or respect.

Big trends right now:

- Let LM interact with external tools (e.g., a search engine, databases, solvers, ...), including attributation/citation to sources of information.
- Relatedly: "nonparametric" aspects to LM design (the model uses a stored corpus it can access at inference time; Khandelwal et al., 2020)
- Multilingual language models trained on text from many languages; these show some "transfer" capabilities.
- LMs to synthesize data for NLP tasks, with some human input as well (e.g., Liu et al., 2022)
- Efficient finetuning (e.g., adaptors; Houlsby et al., 2019)
- Ensembles of smaller and/or cheaper LMs (Tim's lecture)
- Multimodality, e.g., text-to-image, image-to-text, ...

References I

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askel, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, S. Arun Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. arXiv:2107.03374.

References II

Hyung Won Chung, Le Hou, S. Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Wei Yu, Vincent Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed Huai hsin Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

Suchin Gururangan, Dallas Card, Sarah K. Drier, Emily Kalah Gade, Leroy Z. Wang, Zeyu Wang, Luke Zettlemoyer, and Noah A. Smith. Whose language counts as high quality? measuring language ideologies in text data selection. In *Proc. of EMNLP*, 2022.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and L. Sifre. Training compute-optimal large language models, 2022. arXiv:2203.15556.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *Proc. of ICML*, 2019.

References III

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *Proc. of ICLR*, 2020.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. WANLI: Worker and ai collaboration for natural language inference dataset creation. In *Proc. of EMNLP*, 2022.

Aman Madaan, Shuyan Zhou, Uri Alon, Yiming Yang, and Graham Neubig. Language models of code are few-shot commonsense learners. In *Proc. of EMNLP*, 2022. URL `https://aclanthology.org/2022.emnlp-main.90`.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. Training language models to follow instructions with human feedback, 2022. arXiv:2203.02155.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. Language models as knowledge bases?, 2019. arXiv:1909.01066.

Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners. In *Proc. of NAACL*, 2021. URL `https://aclanthology.org/2021.naacl-main.185`.

References IV

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models, 2023.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, M. Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddharth Deepak Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hanna Hajishirzi, and Daniel Khashabi. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proc. of EMNLP*, 2022.