The main innovation that has moved methods in natural language processing from largely *not working* to largely *working* is pretraining. At a high level, this idea returns exactly to the start of the course. Pretraining means trying to learn from a huge amount of text (and usually now also images, video, audio, etc. But we'll focus on text.) All of the topics we've covered so far—expressivity of neural networks, optimization, tokenization, and parallelizable architectures—are to some extent in service of better pretraining.

The method we introduced in lecture 1 is similar to word2vec [Mikolov et al., 2013], and we saw that our simple word prediction algorithm led to very interesting learned structure. By scaling up the expressivity of the architecture and the scale, interesting things keep happening!

Recall language modeling

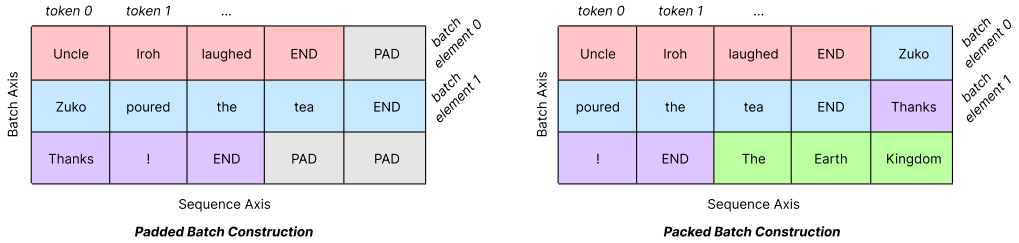

Figure 1: Comparison of padding and packing for batch construction.

What predictive problem are we going to be training our network on? You guessed it, languge modeling. Recall that we have some learnable parameters θ in our distribution p_θ, we have a data distribution $\mathcal{D}$, and we're going to optimize:

$$\min_\theta \mathbb{E}_{x\sim\mathcal{D}} \left[-\log p_\theta(x)\right] \tag{1}$$

But at this point let's get a bit more specific. This math suggests that we're optimizing specifically over entire documents. In practice, something slightly messier happens.

Let's say I have a batch size B and a maximum sequence length n. I've probably set B and n such that I have as long sequences as I can, and as large a batch as I can, such that it'll fit on my GPU cluster. So, that's Bn tokens I can potentially learn from. However, if I try to optimize the math above, suggesting that I optimize for the likelihood of whole documents, this means I need to filter out documents longer than n tokens. Furthermore, any documents shorter than n tokens I need to pad with useless blanks that won't be trained on, in order to fill out the batch. That's wasted compute!

Instead, batches are *packed* with tokens. That is, we string together a bunch of documents and just pick the first Bn tokens, even if they cross document boundaries. If they do cross document boundaries, we include a document separator token.

We then optimize a similar-looking objective over this new distribution over tokens, call it $\tilde{\mathcal{D}}$.

$$\min_\theta \mathbb{E}_{x_{1:t}\sim\tilde{\mathcal{D}}} \left[-\log p_\theta(x_t \mid x_{<t})\right] \tag{2}$$

This is a bit of a technical detail—much of the time we're still optimizing document likelihoods, but not always due to the packing (Figure 1), but I think it helps build intuition—we're looking for useful token sequences to learn from, and a lot, as fast as possible.

Data distributions

First, take a look at Table 1. Here are two "real" pretraining documents from FineWeb [Penedo et al., 2024].

What token sequences should we learn from? Sometimes we approximate the training distribution of language models as *the whole internet*, but this is wrong for various reasons. Still, it's a useful intuition, and a large part of pretraining datasets is often Common Crawl dumps, which are large public crawls of the internet. But cleaning, filtering, formatting the text in raw crawls is critical.

Here are some types of filtering or processing found in the FineWeb dataset [Penedo et al., 2024]:

- Text extraction from HTML! Actually not easy to get right.
- Language filtering for mostly-English data
- URL filtering to avoid adult content
- Deduplication
- Personally identifiable information heuristic removal

It's **very expensive** to test hypotheses about what kind of web data is best for pretraining. Vaguely, some notion of quality is usually used—Wikipedia is high quality, random web pages might be low-quality. One nice concrete intuition is as follows, however.

In DataCompLM, they trained models to score each web document with how alike it is to a combination of Reddit Explain-Like-I'm-Five data and a synthetically generated question-and-long-response chatbot dataset called OpenHermes2.5 [Teknium, 2023]. That is, when filtering down from a raw web dump, documents are kept when the model predicts that they're *more* like these sources. And this worked very well!

Scaling laws (estimates)

Part of the promise of pretraining is that it's been shown to work (and thus is to some extent predicted to work) across many orders of magnitude. This is especially important because of the immense costs of scaling to each additional order of magnitude. See Figure 2, in which scaling the amount of computation, the amount of data, and the number of parameters leads to a decay in loss that scales linearly with the logarithm of the increase in cost. While sometimes stated as natural laws, I think of them as useful estimates.

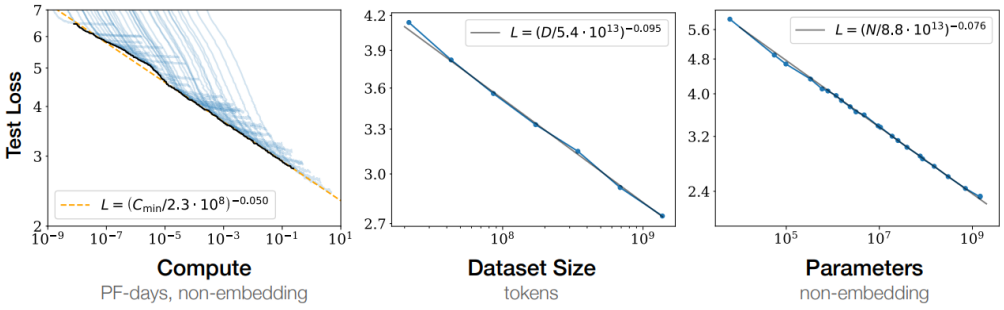

Figure 2: The scaling law plots from Figure 1 of [Kaplan et al., 2020].

FineWeb Examples
Previous abstract Next abstract Session 40 - The Interstellar Medium. Display session, Tuesday, June 09 Gamma Ray Burst (GRB) explosions can make kpc-size shells and holes in the interstellar media (ISM) of spiral galaxies if much of the energy heats the local gas to above 10^7 K. Disk blowout is probably the major cause for energy loss in this case, but the momentum acquired during the pressurized expansion phase can be large enough that the bubble still snowplows to a kpc diameter. This differs from the standard model for the origin of such shells by multiple supernovae, which may have problems with radiative cooling, evaporative losses, and disk blow-out. Evidence for giant shells with energies of $\sim 10^53$ ergs are summarized. Some contain no obvious central star clusters and may be GRB remnants, although sufficiently old clusters would be hard to detect. The expected frequency of GRBs in normal galaxies can account for the number of such shells. Program listing for Tuesday
Wikipedia sobre física de partículas Rapidinho. Me falaram que a definição de física de partículas da Wikipedia era muito ruim. E de fato, era assim: Particle physics is a branch of physics that studies the elementary particle|elementary subatomic constituents of matter and radiation, and their interactions. The field is also called high energy physics, because many elementary particles do not occur under ambient conditions on Earth. They can only be created artificially during high energy collisions with other particles in particle accelerators. Particle physics has evolved out of its parent field of nuclear physics and is typically still taught in close association with it. Scientific research in this area has produced a long list of particles. Mas hein? Partículas que só podem ser criadas em aceleradores? Física de partículas é ensinada junto com física nuclear? A pesquisa produz partículas (essa é ótima!)? Em que mundo essa pessoa vive? Reescrevi: Particle Physics is a branch of physics that studies the existence and interactions of particles, which are the constituents of what is usually referred as matter or radiation. In our current understanding, particles are excitations of quantum fields and interact following their dynamics. Most of the interest in this area is in fundamental fields, those that cannot be described as a bound state of other fields. The set of fundamental fields and their dynamics are summarized in a model called the Standard Model and, therefore, Particle Physics is largely the study of the Standard Model particle content and its possible extensions. Eu acho que ficou bem melhor. Vamos ver em quanto tempo algum editor esquentado da Wikipedia vai demorar para reverter. Atualmente está um saco participar da Wikipedia por causa dessas pessoas.

Table 1: Some example documents from FineWeb. Note that they're still pretty messy.

Loss spikes, scaling troubles

Other learning objectives

References

[Kaplan et al., 2020] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. (2020). Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

[Li et al., 2024] Li, J., Fang, A., Smyrnis, G., Ivgi, M., Jordan, M., Gadre, S. Y., Bansal, H., Guha, E., Keh, S. S., Arora, K., et al. (2024). Datacomp-lm: In search of the next generation of training sets for language models. *Advances in Neural Information Processing Systems*, 37:14200–14282.

[Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

[Penedo et al., 2024] Penedo, G., Kydlíček, H., Lozhkov, A., Mitchell, M., Raffel, C. A., Von Werra, L., Wolf, T., et al. (2024). The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849.

[Teknium, 2023] Teknium (2023). Openhermes 2.5: An open dataset of synthetic data for generalist llm assistants. https://huggingface.co/datasets/teknium/OpenHermes-2.5.