Slide for logistics and news

Agenda for this deck

- Discuss key takeaways from the Hestness data scaling paper
- "Where are they now?" - How do these results hold up in 2024?
    - Discuss the data scaling database
- Return to our question from last week: what are the actual orders of magnitude here?

Hestness data scaling paper

- Note that this paper discusses both data scaling and compute scaling (we haven't talked about compute much)

In short: this paper reports on a number of experiments that involve training models with different sized datasets with the ultimate goal of predicting "if has x times more data, how much better would our model be?"

Key Takeaway #1: Power Laws in Theory

Theory says: $\varepsilon(m) \propto \alpha m^{\beta_g}$

- ε is generalization error
- m is "number of samples"
- α (alpha) is constant (specific to task)
- β_g is scaling exponent (specific to task). β_g should be -0.5 or -1 in theory, but in practice is more like -0.07 to -0.35

Power Laws in Practice

When you run the big expensive experiments... it does seem like power law describe the scaling pretty well!

- holds across models, optimizers, regularizers, and loss function choices
- steepness is domain specific (picking a better model or optimizer just improves the intercept)

Regions of scaling

- small training set region (i.e. random guess)
- Power law scaling region
- Irreducibile error region aka Bayes region aka flat zone

Quick run-down on history

- lots of old efforts to explain data scaling in theory (e.g. provide theoretical bounds on how much data do I need to get a certain performance)
- but they need to be paired with empirical investigation
- spoiler: in the Hestness et al. paper, observed power laws don't quite match theoretical expectations

Aside: the coin flip model in the appendix

The appendix has a cool theorem in the appendix ("POWER-LAW LEARNING CURVE FOR COUNTING MODEL CLASSIFIER")

- We have a coin flip "probability estimator"
- We count how many times we get heads
- Discuss this more if time.

What about model parameters

We've mostly ingored for now. When we're talking about data-centric tasks, its convenient to just assume away -- "let's assume we've got a big enough model and a decent enough architecture". In practice, these things are intertwined.

A nice quote on the topic from Hestness et al. :)

"Rather than reason through these complexities, it is currently easier for researchers and practitioners to over-parameterize models to fit training data"

Key Takeaway: Using data scaling to estimating costs of progress

Hestness et al. paper frames the investigation in terms of economic decision-making. If we understand data scaling, we might be better informed about when to look for bugs, when to buy more data, etc.

Hestness et al. experiments

Lots of interesting details on experiments! For 419 -- we won't test on this, but worth looking at if you're interested.

- Subdivide datasets into shards using powers of two (0.1%, 0.2%, 0.4%, 0.8%...)
- single held out test set
- find the best model for each dataset size (expensive!)

Key Takeaway: Understand how you might set up a data scaling experiment

Review question: What are the key components to include in a script that runs a data scaling experiments?

On loss function choice

"We report validation losses that are sums or unweighted averages over distance metrics measuring
the error per model-predicted output"

- loss function matters for the exact characteristics of the learning curve
- intuitively, if we have accuracy on the y-axis that will be different than cross-entropy (they're different measurements)

Very quick summary of results

- Machine translation: -0.128
- Language modeling: -0.09 to -0.06
- Image classification: -0.309 to -0.488 (depends on eval procedure, top 1 vs top 5)
- Speech recognition: -0.299
- None are -0.5!

Takeaway from the actual coefficient values

Do not need to memorize these!

Key ideas:

- "probably dependent on aspects of the problem domain or data distribution" (Sec 5.1)
- Not quite -0.5 or -1

Are we *sure* it's a power law?

Discussion question: How could we be really sure?

So, what does this all mean? Just get more data?

Seems like "just get more data" is a pretty good strategy.

Furthermore, we can start to do some very basic economic reasoning.

What parameters would we need to make a decision like this?

Are we in the flat zone?

Authors provide three reasons for irreducible error:

- mislabeled samples (training data influence might help with that... 🤔)
- information theoretic lower bound (i.e. how "model-able" is our fundamental data generating functions)

None of their experiments got to the flat zone. One open question for gen AI companies and other users of "web-scale" data is figuring this out!

Key Takeaway: Relationship between data scaling and influence?

So, how do these 2017 data scaling experiments related to our 2023 data influence survey?

Are data scaling and training data influence the same thing?

Key Takeaway: Relationship between data scaling and influence

Training data influence methods all tell us about counterfactual worlds -- what if we had more data, what if we had less data.

So does data scaling.

Training data influence focuses on specific data points -- what if we had more of a certain group. Data scaling focuses on random draws across a distribution / data-generating function.

Data scaling data base today

See review by Villalobos

https://epochai.org/blog/scaling-laws-literature-review

Follow up work to understand the regions

Revisiting Neural Scaling Laws in Language and Vision - Alabdulmohsin et al. (Google) - https://arxiv.org/abs/2209.06640v2

- $\frac{\varepsilon_x - \varepsilon_\infty}{(\varepsilon_0 - \varepsilon_x)^\alpha} = \beta x^c $

Broken Neural Scaling Laws- Caballero et al. - https://arxiv.org/abs/2210.14891 - https://github.com/ethancaballero/broken_neural_scaling_laws

What are the orders of magnitude

Villalobos also made a public Google sheet. Here, we can see data units and data range!

e.g. "Scaling Vision Transformers" from Zhai et al. uses 1e8 to 1e10 images (roughly: one hundred million to ten billion)

"Data Scaling Laws in NMT: The Effect of Noise and Architecture" from Bansal et al. uses 5e5 to 5e8 sentence (roughly: five hundred thousand to five hundred million)

What are the orders of magnitude

Other experiments focus more on the 1000 to 10000 range of images, or 1 million to 1 billion range of characters

Other domains

Take some guesses at the size (in # images, # characters, total GB) etc. of a dataset of interest?