This appears to be a relatively standard *sparse coding* problem, where we want to take the activations of neural network layers and find out which directions correspond to features.[20]

The advantage of this is that we don't need to worry about whether we're damaging model performance. On the other hand, many other things are harder:

- **It's no longer easy to know how many features you have to enumerate.** A monosemantic model represents a feature per neuron, but when finding an overcomplete basis there's an additional challenge of identifying how many features to use for it.

- **Solutions are no longer integrated into the surface computational structure.** Neural networks can be understood in terms of their surface structure – neurons, attention heads, etc – and virtual structure that implicitly emerge (*e.g.* virtual attention heads [40]). A model described by an overcomplete basis has "virtual neurons": there's a further gap between the surface and virtual structure.

- **It's a different, major engineering challenge.** Seriously attempting to solve superposition by applying sparse coding to real neural nets suggests a *massive* sparse coding problem. For truly large language models, one would be starting with something like a millions (neurons) by billions (tokens) matrix and then trying to do an extremely overcomplete factorization, perhaps trying to factor it to be a thousand or more times *larger*. This is a major engineering challenge which is different from the standard distributed training challenges ML labs are set up for.

- **Interference is no longer pushing in your favor.** If you try to train models without superposition, interference between features is pushing the training process to have less superposition. If you instead try to decode superposition after the fact, whatever amount of superposition is "baked in" by the training process and you don't have part of the objective pushing in your favor.

APPROACH 3: HYBRID APPROACHES

In addition to approaches which address superposition purely at training time, or purely after the fact, it may be possible to take "hybrid approaches" which do a mixture. For example, even if one can't change models without superposition, it may be possible to produce models with *less* superposition, which are then easier to decode.[21] Alternatively, it may be possible for architecture changes to make finding an overcomplete basis easier or more computationally tractable in large models, separately from trying to reduce superposition.

Additional Considerations

Phase Changes as Cause For Hope. Is totally getting rid of superposition a realistic hope? One could easily imagine a world where it can only be asymptotically reduced, and never fully eliminated. While the results in this paper seem to suggest that superposition is hard to get rid of because it's actually very useful, the upshot of it corresponding to a phase change is that there's a regime *where it totally doesn't exist*. If we can find a way to push models in the non-superposition regime, it seems likely it can be totally eliminated.

Any superposition-free model would be a powerful tool for research. We believe that most of the research risk is in whether one can make *performant* superposition free models, rather than whether it's possible to make superposition free models at all. Of course, ultimately, we need to make performant models. But a non-performant superposition free model could still be a very useful research tool for studying superposition in normal models. At present, it's challenging to study superposition in models because we have no ground truth for what the features are. (This is also the reason why the toy models described in this paper can be studied – we do know what the features are!) If we had a superposition-free model, we may be able to use it as a ground truth to study superposition in regular models.

Local bases are not enough. Earlier, when we considered the geometry of non-uniform superposition, we observed that models often form *local orthogonal bases*, where co-occurring features are orthogonal. This suggests a strategy for locally understanding models on sufficiently narrow sub-distributions. However, if our goal is to eventually make useful statements about the safety of models, we need mechanistic accounts that hold for the full distribution (and off distribution). Local bases seem unlikely to give this to us.

Discussion

To What Extent Does Superposition Exist in Real Models?

Why are we interested in toy models? We believe they are useful proxies for studying the superposition we suspect might exist in real neural networks. But how can we know if they're actually a useful toy model? Our best validation is whether their predictions are consistent with empirical observations regarding polysemanticity. To the best of our knowledge they are. In particular:

- **Polysemantic neurons exist.** Polysemantic neurons form in our third model, just as they are observed in a wide range of neural networks.
- **Neurons are sometimes "cleanly interpretable" and sometimes "polysemantic", often in the same layer.** Our third model exhibits both polyemantic and non-polysemantic neurons, often at the same time. This is analogous to how real neural networks often have a mixture of polysemantic and non-polysemantic neurons in the same layer.
- **InceptionV1 has more polysemantic neurons in later layers.** Empirically, the fraction of neurons which are polysemantic in InceptionV1 increases with depth. One natural explanation is that as features become higher-level the stimuli they detect become rarer and thus sparser (for example, in vision, a high-level floppy ear feature is less common than a low-level Gabor filter's edge). A major prediction of our model is that superposition and polysemanticity increase as sparsity increases.
- **Early Transformer MLP neurons are extremely polysemantic.** Our experience is that neurons in the first MLP layer in Transformer language models are often extremely polysemantic. If the goal of the first MLP layer is to distinguish between different interpretations of the same token (eg. "die" in English vs German vs Dutch vs Afrikans), such features would be very sparse and our toy model would predict lots of polysemanticity.

This doesn't mean that everything about our toy model reflects real neural networks. Our intuition is that some of the phenomena we observe (superposition, monosemantic vs polysemantic neurons, perhaps the relationship to adversarial examples) are likely to generalize, while other phenomena (especially the geometry and learning dynamics results) are much more uncertain.

Open Questions

This paper has shown that the superposition hypothesis is true in certain toy models. But if anything, we're left with many more questions about it than we had at the start. In this final section, we review some of the questions which strike us as most important: what do we know, and would we like for future work to clarify?