

Peer Review

Review of: "Superposition in Transformers: A Novel Way of Building Mixture of Experts"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The authors propose a way to hold two language models inside one set of layers. A base GPT-2 (English) and a copy fine-tuned on French are both frozen; their hidden states are mixed layer by layer with a coefficient $\alpha(l)$ that comes from a B-spline over the layer index plus a per-layer bias, clamped to $[0, 1]$ (§3.2.2). At selected layers, a small autoencoder then takes the blended state and is trained to reconstruct either the base or the fine-tuned hidden state, depending on the input domain (§3.4.1, §3.5.4). The training signal is a language-modelling loss plus a reconstruction loss, with an optional regularizer on the spline control points and biases (§3.5.2, Appendix A.1, Eq. 3–7). After training, the paper says one can either hard-merge the weights with the learned $\alpha(l)$ (§3.3.1) or keep the spline and autoencoders as a gate. The experimental claim is that this "superposition" keeps most of both models' abilities. Table 1 shows the merged model at perplexity 43.14 on English and 33.20 on French, against 28.88 / 132.02 for the base model and 54.42 / 29.57 for the fine-tuned one. §4.2.2 reports that it beats linear interpolation and task arithmetic on perplexity and next-token accuracy, and §4.3 plus Appendix B use t-SNE, PCA, a neuron-level threshold (Table 2), and a clustering-based diversity measure (Figures 4–5) to argue that the autoencoders produce language-appropriate states and more shared ("polysemantic") neurons.

2. Whether the conclusion holds on the data

I like the core idea, and I want to say so first. Learning the mixing coefficient as a smooth function of depth, instead of one global scalar, is a sensible prior, and the choice of English versus French gives a clean, measurable domain split. The authors also compare against two real merging baselines rather than

only against the parents, and Table 1 is laid out so that the trade-off can be read directly. That is a good starting point. What follows is mostly about how far the current evidence reaches.

How much forgetting is actually prevented. The abstract says the method "effectively mitigates catastrophic forgetting" and "preserves original model capabilities." Table 1 supports a more modest reading. On English, the merged model goes from 28.88 (Base Model, English) to 43.14 (Merged Model, English): $43.14 / 28.88 = 1.494$, so a 49.4% rise in perplexity. On French, the cost is smaller: $33.20 / 29.57 = 1.123$, a 12.3% rise over the fine-tuned model. Put another way, of the English gap between the fine-tuned model (54.42) and the base model (28.88), the merged model recovers $(54.42 - 43.14) / (54.42 - 28.88) = 11.28 / 25.54 = 44.2\%$. Of the French gap, it recovers $(132.02 - 33.20) / (132.02 - 29.57) = 98.82 / 102.45 = 96.5\%$. So the method keeps French nearly intact and gives back less than half of what fine-tuning cost on English. That is a real, reportable result. It is not yet "preserves original capabilities," and I would phrase the abstract and §7 around these two recovery figures.

The missing reference point. The comparison that tells a reader whether the extra machinery pays off is the trivial router: send English to the base model and French to the fine-tuned model. Because §3.5.4 says domain labels guide the autoencoder target, and the experiments appear to use them, this router uses the same information and would score $28.88 / 29.57$ by construction. Its cost is two full models. The paper's argument is therefore really about the parameter-versus-quality trade-off, and I would make that explicit: the number of parameters needed at inference for each option, next to its per-language perplexity. At the moment, I could not tell from the text whether the evaluated "Merged Model" in Table 1 is the hard merge of §3.3.1 (one GPT-2 plus autoencoders) or a system that still runs both frozen parents to produce h_{base} and h_{fine} . If it is the second, the parameter-efficiency claim in §1 (contribution 3) needs to be revisited.

The baselines in §4.2.2. Here I found several things I could not reconcile. The merged model's perplexity is given as $M\text{-PPL} = 47.01$, which matches neither of its Table 1 values (43.14, 33.20). I assume 47.01 is on the combined validation set, but the text does not say. Linear interpolation and task arithmetic are both labelled "I-PPL" (60.29 and 61.30) and "I-Acc" (0.3039 and 0.2957); a separate label for task arithmetic would avoid confusion. More important, both baselines are described as measured "during the last epoch of training," yet neither is a trained method in its usual form. If the interpolation weight or task-arithmetic scale was learned, or if the baselines were given the same epochs and data, it would help to say so. Finally, the baselines are reported only as pooled numbers. A per-language row for each, in Table 1, is

what would show whether the merged model wins on both languages or wins on average by favoring French.

The JSD paragraph. The Jensen–Shannon divergence is bounded (by $\ln 2 \approx 0.693$ in nats, or by 1 in bits). The reported values M-JSD = 82.9, 45.5, 36.5, and 44.2 are outside that range, so a scaling factor (perhaps $\times 100$) or a summed rather than averaged quantity must be involved; please state which. The interpretation also runs the wrong way, as far as I can read it. The text says the drop from 82.9 to 45.5 "indicates that the merged model's output distribution quickly diverges from the initial average." A falling divergence means the distributions are getting closer, not further apart. If the reference distribution is the average of the two parents, the numbers say the merged model moves toward that average, which is the opposite of the conclusion drawn in the next sentence ("distinct from the simple average").

Neuron-level evidence. Table 2 lists polysemantic neurons at 1.8229% (Base), 2.0833% (Fine-Tuned), and 3.3854% (Merged). These percentages are exactly $14/768$, $16/768$, and $26/768$ ($14 / 768 = 0.018229$; $16 / 768 = 0.020833$; $26 / 768 = 0.033854$), which suggests the analysis is over 768 units of layer 4. If so, the headline difference is 26 versus 14 neurons, from one run, with a threshold of 0.05 that the authors themselves call possibly unreliable (§4.3.2). I would report the counts, show how they change across a range of thresholds, and add a second seed. Two conceptual points too. First, a neuron whose average activation is similar for English and French is language-agnostic; that is not the same as polysemantic in the sense of Elhage et al. [14], where one unit represents several unrelated features. Second, the sparsity column (0.0166, 0.0154, 0.0179, in %) differs in the third decimal; the text rightly calls this "minimal and potentially arbitrary," so I would not describe it as "enhanced neuron utilization."

Data split. §4.1 says the English set is 6,000 samples "repeated with range(3)" (so 18,000 items, matching 18,000 French articles) in the 1D experiment, and repeated twice in the 2D experiment. Then 10% of the combined samples go to validation. If the split is done after repetition, the same English text can sit in both training and validation, which would lower English validation perplexity for any model that trains on it. The English data is also described as "a subset of the GPT-2 training data," which means the base model is being evaluated on text it has seen. Both points favor particular rows of Table 1 in different directions. I would split before duplicating and use held-out English text (for example, a WikiText test split) so the two languages are comparable.

Visual evidence. The t-SNE plots (Figure 3, Figures 6–7) and PCA trajectories (Figure 8) are useful illustrations. They are not measurements. Because the autoencoder is trained to reconstruct exactly the

state it is later plotted against, clustering near that target is expected. A number would say more here: the reconstruction MSE per language and per layer, next to the MSE of the plain blended state before the autoencoder.

Scale. All results come from one GPT-2 model, one language pair, and, as far as I can see, one training run. That is fine for a first report. The title and §7 ("a significant advancement", "a step toward more flexible, powerful, and general-purpose AI systems") read as though the result had already been shown at a larger scale or on other domains, and I would tone those down until it has.

3. Reproducibility

The methods section gives the general form of every loss, and the parameter count per autoencoder is stated with its formula. I checked it: assuming GPT-2 small (hidden size 768) and the stated bottleneck of 576, $4 \times 768 \times 576 + 768 + 576 = 1,769,472 + 1,344 = 1,770,816$, which matches the "approximately 1.7 million" in §4.1. The factor 4 is not explained, though; a single-layer encoder and decoder would give $2 \times d \times b$. A sentence on the autoencoder's layer structure would settle it.

Beyond that, I could not find the details needed to repeat the experiment:

- No code or checkpoint link anywhere in the text.
- The GPT-2 size is not stated; I inferred 768 from the arithmetic above.
- The French fine-tuning itself (corpus, number of steps, learning rate, sequence length) is not described. The French model is one of the two things being merged, so this matters.
- The spline is under-specified: degree k , number of control points N , and knot placement are not given, nor which layers carry autoencoders in the 1D run (for 2D, the note in §4.3.3 says layers 4 to 10).
- Loss weights λ_{LM} , λ_{Recon} , λ_{Alpha} , λ_{MSE} , and λ_{L2} (Eq. 1), the four regularizer weights in Eq. 7, and σ_{target} in Eq. 6 have no values.
- Optimizer, learning rate, batch size, number of epochs, seeds, and hardware are missing. The 2D bottleneck size is not given either.
- Whether domain labels were used at inference in Table 1 is not stated.

As rendered on Qeios, I could not read the training algorithms: A.2 shows only the caption of Algorithm 1, and A.3 shows only its section heading. It would help to check how these display on the Qeios page or to provide them as text.

4. What specifically to fix

1. **Abstract and §7.** Replace "effectively mitigates catastrophic forgetting" and "preserves original model capabilities" with the measured trade-off from Table 1: about 96% of the French gap recovered, about 44% of the English gap, English perplexity up 49% over the base model.
2. Add a routing baseline to Table 1 (base for English, fine-tuned for French) with its parameter count, and a parameter column for every row.
3. **§3.3 and Table 1.** State which inference mode produced the merged-model numbers: the hard merge of §3.3.1, or spline plus autoencoders with both parents running. If both were tried, report both.
4. §4.2.2: say which dataset gives M-PPL = 47.01, rename the task-arithmetic metrics so they do not share the "I-" prefix with interpolation, explain what "last epoch of training" means for these two baselines, and split all three methods by language.
5. Fix the JSD paragraph in §4.2.2. Give the units or scaling behind 82.9 / 45.5 / 36.5 / 44.2, name the reference distribution, and correct "diverges" (a falling JSD means convergence).
6. **Table 2.** Report neuron counts (26 vs 14 vs 16 out of 768, if my reading is right) next to the percentages, add a threshold sweep, and use "language-shared" rather than "polysemantic," or justify the stronger term.
7. §4.1: split train and validation before repeating English samples, and evaluate English on text outside GPT-2's training data.
8. Add per-layer reconstruction MSE per language (with and without the autoencoder) to support Figures 3 and 6–7 quantitatively.
9. At least three seeds for the Table 1 numbers, with standard deviations. The gaps between methods in §4.2.2 (for example, 0.3270 vs 0.3039 accuracy) need a spread to be read.
10. A hyperparameter table and a code link (see §3 of this review for the list), plus a text version of the training algorithms in A.2 and A.3.
11. A short ablation: spline $\alpha(l)$ versus a single learned scalar α , and autoencoders on versus off. Without it, I cannot tell which component carries the result.
12. **Title.** There is no learned gate or router in the experiments, so "Mixture of Experts" may mislead readers who expect one. "Merging" or "superposing two fine-tunes" describes what is shown.
13. Small language fixes: "might does not" (§5), "Additionnaly" (§4.2.2), "aferomentionned" (A.1), "learing" (§2). The duplicated paragraph after Table 1 can go.

5. Recommendation

Major revisions. The idea is original, and the first experiment is honest: Table 1 already shows a meaningful gain over the fine-tuned model on English and near parity on French. What the paper needs is mostly additional reporting (inference mode, routing baseline, per-language baselines, seeds, hyperparameters) and a more careful reading of what the numbers show, including the JSD interpretation and the English perplexity cost. None of this requires a new method. I would be glad to read a revised version.

Declarations

Potential competing interests: No potential competing interests to declare.