We thank the reviewers for their feedback. A common motive in all reviews is that decentralized training is a promising research direction that can solve major challenges of model scaling and accessibility. Up until now, this area has not been actively discussed in the academic community. Hence, we believe that the publication of Learning@home will impact diverse application fields and invite broad academic expertise that a single research team cannot have.

(R2) "The evaluation is weak, and the authors admit as much."
(R4) "If authors could provide further large scale experiments/evaluations, I will raise my score."
(R1) "not clear what performance trade-off exists with this new architecture and asynchronous updates"

To better support our claims, we conduct additional experiments on the language modeling task. Specifically, we train Transformer-XL [1] on the WikiText-2 dataset. Both baseline and DMoE models use official recommended parameters with regularization implemented in [2]. The `base` model contains 16 Transformer layers with hidden size of 400 and 900 units in the feedforward layer. We also train a `small` baseline model with 200 hidden and 450 feedforward units. Our DMoE Transformer uses 256 experts split evenly between 16 layers. Each expert is a Transformer layer with the same dimensions as layers of `small` baseline model. The DMoE layers route to top-4 experts, making our model equivalent to `small` in terms of FLOPs. As in Section 4.2, we train DMoE with 32 trainers (batch size 1 each), 1000ms average latency, and 10% failure rate. Figure 1 shows that DMoE outperforms the baseline with the same compute budget.

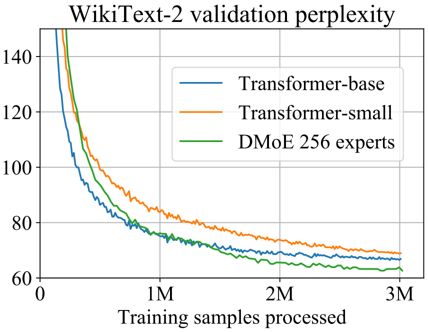

Figure 1: Results on WikiText-2.

(R1) "does . . . facilitate competitive natural language processing or computer vision performance?"
(R2) "The main motivation in the paper is a bit simplistic and inwards-facing on the ML community."

Fortunately, practical performance benefits of MoE-based models for natural language processing were demonstrated in a concurrent preprint [3]. This study reports training MoE-Transformer with 600B parameters for multilingual machine translation using 2048 TPUv3 accelerators with gains of up to +13.5 BLEU (page 16, Figure 6).

In turn, Learning@home infrastructure provides a way of training such models using volunteer hardware instead of a TPU cluster. We understand that this claim would be better supported by a training campaign with thousands of volunteers. Due to the restrictions of anonymity, the best evidence we can provide is that Learning@home runs reliably with $10,000$ CPU-only participants (L308) and models with a memory footprint of up to 192Gb (Section 4.2).

(R1)(R2)(R4) Feedback on paper clarity and presentation.

While all reviewers agree that the paper is well-written, they suggest similar improvements to the presentation. We will incorporate these suggestions in the final version of the paper:

- (R1, R4) Reduce the length of the "Related Work" section to free up space for additional experiments;
- (R2) In turn, create "Additional Related Work" section in supplementary materials for further details, including a more detailed discussion on the background and inner mechanisms of Distributed Hash Tables.
- (R1, R2, R4) Use the freed space to report WikiText-2 experiments (see above) and expand the conclusion.

(R1) "Volunteer computing over internet-connected servers and devices is a widespread technique"
(R4) "Making use of the poor individual devices for distributed deep learning model training is not a new idea"

Though this is technically correct, there is only one study (Kijsipongse et al, 2018) that applies volunteer computing to general deep learning. Their approach requires that the model fits in the GPU memory of the weakest participant. Other projects share this drawback and only operate on niche tasks such as playing board games (see L130-138).

(R1) "less conversational tone in scientific writing (L49: "all that power", L50: "way slower")"

We agree that most colloquial phrases can be replaced with more formal language without reducing text clarity.

(R1) "What is a block architecture?" The feedforward block used in Section 4.2 is the same as described in Section 4.1 (L290), but with half as many units in every dimension. We will clarify the description to avoid reader confusion.

(R1) "How do existing deep learning frameworks fail to support DMoE architecture?" Existing DL frameworks (e.g. TensorFlow/PyTorch) support mechanisms for model-parallel training, but these mechanisms can't recover from node failures. Other tools such as TorchElastic ensure fault tolerance but are incompatible with model-parallel training.

(R1) "Why does Learning@home have a higher throughput . . . at 0ms network delay?" Even without network delay, the batch processing time will still fluctuate due to device specifics, leading to delays in model-parallel training.

(R2) On 100Mb/s symmetric bandwidth assumption. We will add further justification of this assumption based on Speedtest global index [4]. To summarize, the symmetric bandwidth in top-20 countries is generally in the 90–200Mb/s range, and the global average is steadily increasing.

[1] Z. Dai et al. "Transformer-XL: Attentive Language Models beyond a Fixed-Length Context." ACL 2019.
[2] https://github.com/TimDettmers/transformer-xl
[3] D. Lepikhin et al. "Gshard: Scaling giant models with conditional computation and automatic sharding." arXiv:2006.16668 (2020).
[4] Speedtest Global Index for Fixed Broadband https://www.speedtest.net/global-index (11.08.2020)