UNDERSTANDING CLIPPING IN ZEROTH-ORDER OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zeroth-order optimization has emerged as a resource-efficient alternative to Stochastic Gradient Descent (SGD) when fine-tuning LLMs, thanks to the inherent low dimensionality of the loss landscape. It relies on a two-point estimate of the gradient that only accesses the function value and not the gradient, bypassing the resource-heavy backpropagation. Recently, it was accidentally discovered that per-sample clipping can improve the performance of the trained model, while running a differentially private version of zeroth-order optimization (Zhang et al., 2024a) with a very large privacy parameter ϵ (corresponding to very little privacy). In this paper, we systematically investigate this phenomenon and demonstrate that (i) the optimal choice of learning rate η is inversely proportional to the clipping threshold c, satisfying $\eta c = \text{constant}$; (ii) clipping drives the zeroth-order optimization to a different solution with higher test accuracy; and (iii) this phenomenon is only observed with per-sample clipped zeroth-order optimization and not other zeroth-order alternatives that also provide robust estimates of the gradient. Further analyses with artificially injected label noise reveals that clipping leads to more robust predictions, which stems from the implicit down weighting of hard-to-fit training examples.

1 INTRODUCTION

Zeroth-Order (ZO) optimization methods allow one to minimize an objective function without access to its gradients. This is necessary (i) when the learner has access to the function evaluation only, as in finding adversarial examples for a black-box model with API access (Chen et al., 2017; Tu et al., 2019; Liu et al., 2020), or (ii) when computing the gradient is prohibitively expensive, as in back-propagation on a model with tens of billions of parameters (Malladi et al., 2023; Zhang et al., 2024a). Compared to First-Order (FO) methods that use gradient information to iteratively update the model, a major bottleneck in adopting ZO optimization in practice is the dimension-dependent convergence rate. ZO methods typically search over all dimensions, wasting a lot of iterations on unnecessary directions with small improvements, resulting in a factor of d slower convergence compared to FO counterparts for d-dimensional optimization problems (Duchi et al., 2015).

The first practical breakthrough, therefore, of ZO methods came in a relatively lower dimensional problem of adversarial attacks, where the optimization is over the input pixels of, say, the Inception-v3 network (Szegedy et al., 2016): $d = 299 \times 299 \times 3 = 268,203$ (Chen et al., 2017). Empirically, ZO methods proved to be successful in finding adversarial examples for a black-box model where we only have API access to the function output (Ye et al., 2018). The next breakthrough in applying ZO methods came in LLM fine-tuning (Malladi et al., 2023; Zhang et al., 2024a). Malladi et al. (2023) theoretically showed that the convergence of ZO methods only scales as the effective rank of the Hessian matrix instead of the number of parameters to be optimized. Empirical evidence suggests that in LLM fine-tuning, the effective rank is significantly smaller than the number of parameters. Since ZO methods do not require back-propagation, which is memory heavy, this makes ZO fine-tuning a memory-efficient alternative for LLM fine-tuning.

Problem formulation. Consider fine-tuning model weights $w \in \mathbb{R}^d$ given n samples, $\min_{w\in\mathbb{R}^d}(1/n)\sum_{i=1}^{n} f_i(w)$, where f_i is the loss on the i-th fine-tuning example. Given access to a FO oracle, Stochastic Gradient Descent (SGD) updates the model as per mini-batch gradient $g(w_t) = (1/|B_t|)\sum_{i\in B_t}\nabla f_i(w_t)$ with learning rate η: $w_{t+1} \leftarrow w_t - \eta\, g(w_t)$. *Zeroth-Order SGD*

[0]Code Base will be released during camera ready version.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(ZO-SGD) only accesses the ZO oracle that can only evaluate the function value, and commonly uses a two-point estimate of the mini-batch gradient:

$$g_\lambda(w_t) \;=\; \frac{1}{|B_t|}\sum_{i\in B_t}\frac{f_i(w_t+\lambda u_t)-f_i(w_t-\lambda u_t)}{2\lambda}u_t\;, \tag{1}$$

where $\lambda>0$ is the *smoothing parameter* and $u_t\in\mathbb{R}^d$ is a random vector with i.i.d. standard Gaussian entries. This approach can also be applied to SVRG and momentum methods (Liu et al., 2018; Chen et al., 2019). While empirically measuring the performance of ZO-SGD under differential privacy, (Zhang et al., 2024a) first reported that adding *per-sample* clipping to ZO-SGD, which we call *clipped ZO-SGD*, can improve performance of LLM fine-tuning. The clipped estimate is:

$$g_{\lambda,c}(w_t) \;=\; \frac{1}{|B_t|}\sum_{i\in B_t}\mathrm{Clip}_c\Big(\frac{f_i(w_t+\lambda u_t)-f_i(w_t-\lambda u_t)}{2\lambda}\Big)u_t\;, \tag{2}$$

where $\mathrm{Clip}_c:\mathbb{R}\to\mathbb{R}$ is defined as $\mathrm{Clip}_c(a)=a$ if $|a|\le c$ and $\mathrm{Clip}_c(a)=\mathrm{sign}(a)\cdot c$ otherwise.

Contributions. We systematically investigate the gain of clipping in ZO-SGD in four fine-tuning tasks of SNLI (Bowman et al., 2015), MNLI (Williams et al., 2018), SST-2 (Socher et al., 2013), and Trec (Li & Roth, 2002) on two pretrained base models of RoBERTa (Liu et al., 2019) and OPT (Zhang et al., 2022a). In all these settings, we identify a *clipping trade-off*: optimal choice of the learning rate inversely scales with the clipping threshold c (Section 3.1) for a wide range of c that we refer to as *fully clipped regime*. More importantly, we demonstrate that the models trained in this regime of clipping converge to a different solution compared to those trained with no clipping (Section 3.2). This new type of solution achieves higher accuracy but suffers from higher loss in both train and test data, indicating that clipping introduces implicit bias for higher accuracy predictions (Section 3.3). We show that this is unique to per-sample clipping, and other robust estimates of the gradient estimate, including batch clipping and Winsorized mean, do not exhibit this phenomenon (Section 3.4). To investigate the origin of higher accuracy, we analyze the calibration of resulting predictions and demonstrate that clipping leads to overconfident and miscalibrated predictions (Section 4.1). Further analysis with artificially injected label noise reveals that clipping effectively down weights samples that are harder to fit. This in turn makes the model robust to training examples that are mislabeled or ambiguous, leading to higher accuracy (Section 4.2). This hypothesis is further supported by our results showing that clipped ZO-SGD performs similarly with data pruned ZO-SGD, where we select a subset of data with smallest loss to retrain a model from scratch with unclipped ZO-SGD (Section 4.3).

2 Preliminary

The two-point estimate in Eq. (1) converges to an unbiased estimate of the gradient as $\lambda\to 0$, i.e., $\lim_{\lambda\to0}\mathbb{E}_{u_t}[g_\lambda(w_t)]=(1/|B_t|)\sum_{i\in B_t}\nabla f_i(w_t)$ (Duchi et al., 2015). This acts as a surrogate for the mini-batch gradient and justifies the use of ZO methods. With per-sample clipping, the gradient estimate is no longer unbiased, and the clipped ZO-SGD is optimizing for a different objective function. It is challenging to interpret what that objective function is, but one can reason about the gradient estimate as a *weighted average* of the individual gradients, with the weight depending only on the gradient norm, $\|\nabla f_i(w)\|$, of the individual gradient. A proof is in Appendix C.

Proposition 2.1. *For the per-sample clipped gradient estimate defined in Eq. (2),*

$$\lim_{\lambda\to0}\;\mathbb{E}_u\big[\,g_{\lambda,c}(w)\,\big] \;=\; \frac{1}{|B_t|}\sum_{i\in B_t}h_c(\|\nabla f_i(w)\|)\,\nabla f_i(w)\;, \tag{3}$$

where $h_c(\|\nabla f_i(w)\|)=2\Phi(c/\|\nabla f_i(w)\|)-1$, and $\Phi(\cdot)$ is the standard normal CDF.

When the clipping threshold c is large enough such that no clipping happens, then this recovers the (asymptotically) unbiased gradient estimate of Eq. (1). We refer to this as an *unclipped* regime. When c is small enough that clipping always happens, the weight function converges to $\sqrt{2/\pi}(c/\|\nabla f_i(w)\|)$ and the gradient estimate simplifies to a voting algorithm where each sample contributes equally regardless of its gradient magnitude. The resulting model update in this case is

$$\mathbb{E}_{u_t}\big[\,w_{t+1}|w_t\,\big] \;\leftarrow\; w_t \;+\; \sqrt{\frac{2}{\pi}}\,c\,\eta\,\frac{1}{|B_t|}\sum_{i\in B_t}\frac{\nabla f_i(w_t)}{\|\nabla f_i(w_t)\|}\;. \tag{4}$$

We refer to this as a *fully clipped* regime. The clipped ZO-SGD can be interpreted as a smooth interpolation between these two extremes. We investigate which regime dominates for real data, depending on the choice of the clipping threshold c, and what the consequences are for the fine-tuned LLMs.

3 THE TRADE-OFF BETWEEN CLIPPING THRESHOLD AND LEARNING RATE

In the fully clipped regime, Eq. (4) suggests that the clipping threshold c and the learning rate η can be seamlessly exchanged as long as we keep $c\eta$ fixed. To test this idea, we perform a geometric grid search over (c, η) and track the resulting model performance. We demonstrate that (i) clipped ZO-SGD typically operates as if it is either in the fully clipped regime or the unclipped regime; (ii) the model converges to different solutions depending on the regime; (iii) in the fully clipped regime, the model suffers higher loss but achieves better accuracy; (iv) this phenomenon is unique to per-sample clipping. A survey of related work is provided in App. A and a detailed experimental set-up in App. D.

3.1 THE *clipping trade-off* AT $\eta\, c =$ CONSTANT

To understand how the optimal learning rate η scales with the clipping threshold c, we sweep through a range of parameters and measure the test accuracy averaged over four fine-tuned models trained on four separate tasks (Figure 1, left). We make two important observations: (i) the optimal learning rate $\eta^*(c)$ for each clipping threshold c lies on the line satisfying $\eta c = 0.0005$ highlighted by orange outline in the diagonal except for $c = \infty$; and (ii) the best average test accuracy of 0.8640 is achieved with clipping, specifically with $c = 5$. This phenomenon, which we call the *clipping trade-off*, is quite robust. The same behavior is demonstrated on each individual fine-tuning task (Figures 13-20 in the appendix) and across two model families (OPT family in Figure 21 in the appendix) in LM fine-tuning.

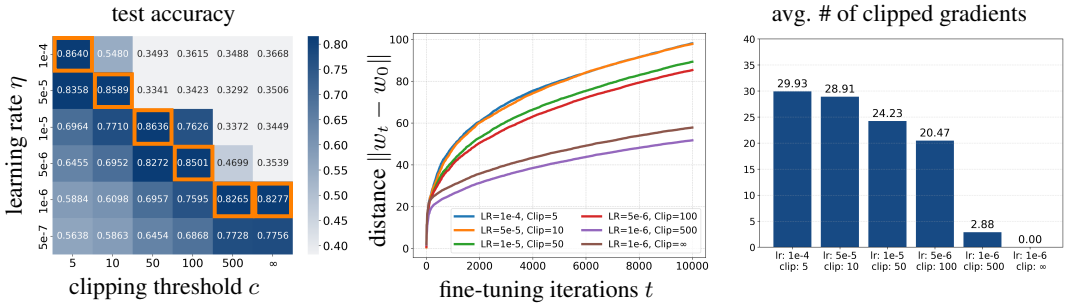

Figure 1: (left) Sweeping through the learning rate η and the clipping threshold c, we observe that the optimal learning rate for each clipping threshold lies on the curve $\eta \times c = 5 \times 10^{-4}$. Each cell shows the test accuracy averaged over four models fine-tuned from RoBERTa (Liu et al., 2019) on SNLI (Bowman et al., 2015), MNLI (Williams et al., 2018), SST-2 (Socher et al., 2013) and Trec (Li & Roth, 2002), respectively. $c = \infty$ is the no clipping baseline. (middle) For those orange cells, tracking how fast SNLI fine-tuned weights w_t move away from the pre-trained checkpoint of RoBERTa w_0, we observe two distinct behaviors depending on the clipping threshold: $c \leq 100$ and $c \geq 500$. (right) For the same orange cells, the average number of clipped examples per iteration (batch size = 64) exhibits similar two distinct regimes.

One explanation of this inverse relation might be that the clipping in Eq. (2) is effectively scaling down the gradient estimate g_λ in Eq. (1) and the learning rates need to compensate for that in order to reach the similar optimal point. This would mean that models trained with the same effective speed of $\eta\, c = 5 \times 10^{-4}$ should be similar. In Figure 1 (middle), we track the Euclidean distance from the initial pre-trained checkpoint: $\|w_t - w_0\|$. This reveals two distinct paths, possibly towards two distinct basins, indicating that clipping is not just rescaling the gradient estimates but rather driving the model towards potentially a better minima, one that can only be reached with clipping.

This dichotomy suggests that when $c \leq 100$, the dynamics of clipped ZO-SGD is driven by the *fully clipped regime*, and when $c > 100$, it is driven by the *unclipped regime*. In the fully clipped regime,

significantly more samples are clipped as shown in Figure 1 (right), and hence the optimal learning rate is inversely proportional to the clipping threshold c as suggested by Eq. (4). This phenomenon of convergence to two distinct solutions is also consistent with the fact that the dynamics of Eq. (4) in the fully clipped regime is distinct from the gradient descent (ZO or FO), due to each sample being weighted differently. We provide detailed analyses of the two types of solutions in Section 3.2 and the statistics of clipped gradients in Appendix G.2.

3.2 Two types of solutions along the clipping trade-off line

Previous experiments suggest that hyperparameter choices along the *clipping trade-off line* (i.e., $\eta c = 5 \times 10^{-4}$) converge to two different types of solutions depending on whether clipping is significant enough ($c \leq 100$). To visualize the geometry of these solutions, similar to (Li et al., 2018; Garipov et al., 2018), we project the high-dimensional train loss and train accuracy landscapes onto the 2D plane spanned by three model checkpoints along this clipping trade-off line: w_1 ($\eta = 5e{-}5$, $c = 10$), w_2 ($\eta = 1e{-}5$, $c = 50$), and w_3 ($\eta = 1e{-}6$, $c = \infty$).

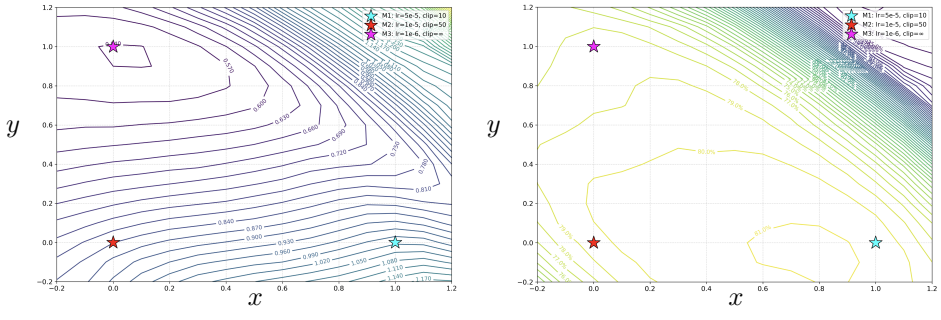

Figure 2: Pretrained RoBERTa fine-tuned on SNLI. (left) Train loss landscape projected down to (x, y)-plane evaluated at $w_2 + (w_1 - w_2)x + (w_3 - w_2)y$ for three model checkpoints w_1, w_2, w_3. (right) Train Accuracy landscape projected down to (x, y)-plane evaluated at $w_2 + (w_1 - w_2)x + (w_3 - w_2)y$ for three model checkpoints w_1, w_2, w_3. This suggests that per-sample clipping introduces an implicit bias towards higher accuracy solutions.

The train loss landscape (Figure 2, left) clearly shows that unclipped ZO-SGD, w_3, converged close to a local minima of the train loss, whereas clipped ZO-SGD, w_1 and w_2, converged to solutions with higher train loss. On the other hand, the train accuracy landscape (Figure 2, right) illustrates that w_1 and w_2 converged to a solution with higher accuracy compared to w_3. This suggests that per-sample clipping introduces an implicit bias in the training that favors higher accuracy at the cost of higher loss. These checkpoints generalize well and the same trend holds for test loss and accuracy, illustrated in Figure 22 in the appendix.

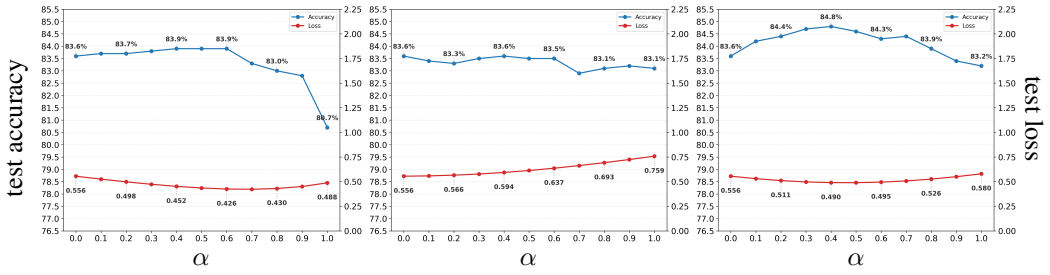

Figure 3: For models w_1, w_2, and w_3 from Figure 2, ensemble model of $\mathrm{SoftMax}((1-\alpha)\, h(x; w_2) + \alpha\, h(x; w_3))$, where h is the logit output, achieves better SNLI accuracy than the two original models (left) but two clipped models, w_2 and w_1, lack diversity and do not show ensemble gain (middle). Two clipped models can still get ensemble gain, if they use two different random seeds (right).

Another way to measure how two model weights are related is to visualize the ensemble model performances (Ainsworth et al., 2023; Nasery et al., 2025; Wortsman et al., 2022). Averaging the logit output of two models and using the softmax to predict, known as an ensemble model, has been observed to give better performance than the two original models when the two model weights are distinct, typically due to different random seeds or different pre-trained base models. Figure 3

(middle) demonstrates how the ensemble of two clipped models w_1 and w_2 provides no gain as expected, since we use the same random seeds and initialization of RoBERTa; they only differ in the hyper parameter (c, η). On the other hand, the ensemble of a clipped model w_1 and an unclipped model w_3 enjoys the ensemble performance gain in both test loss and accuracy (Figure 3, left). This suggests that ensemble model is exploiting the diversity between the two models found in the fully clipped and unclipped regimes. Figure 3 (right) shows as a sanity check that such a diversity can also be exploited by training two clipped models with different random seeds.

3.3 Implicit bias towards higher accuracy

We first ask if clipped ZO-SGD is implicitly regularized and the resulting prediction generalizes better than unclipped ZO-SGD. Comparing the accuracy and loss between train data and test data (Figure 1 left and Figure 4), all the models in the orange trade-off line generalize well; there is no difference in generalization between the two regimes. Next, we ask if clipped ZO-SGD has implicit bias towards certain types of solutions. The prominent distinction for the models trained in fully clipped regime ($c \leq 100$) is that their accuracy and loss are both higher than the unclipped counterparts ($c \geq 500$). Comparing the train loss on the orange cells in the $c \leq 100$ regime, more clipping leads to more bias, resulting in significantly larger train loss for $c = 5$ compared to $c = 100$. However, despite achieving significantly different train losses, all those models trained with clipping achieve similar accuracy that is significantly better than the models trained without clipping. The gain of clipping is quite robust to the choice of hyperparameters, as long as we are on the clipping trade-off line, i.e. $\eta c =$ constant. We provide the training loss and accuracy over the training iterations in Figures 13-20 in the appendix for each task: SNLI, MNLI, SST-2, and Trec.

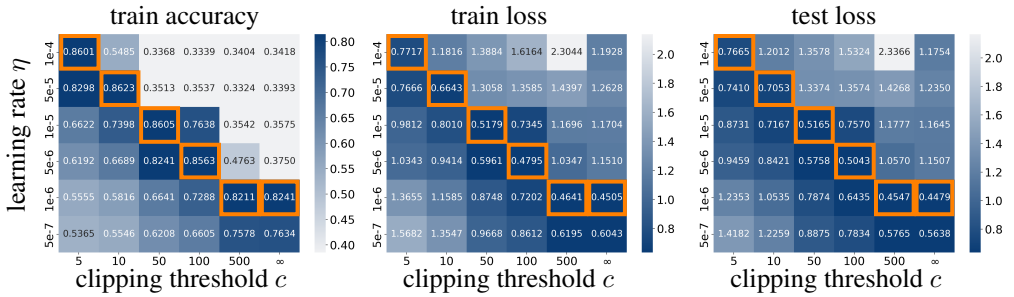

Figure 4: The same setting as Figure 1 but showing the average train accuracy, train loss, and test loss. Both clipped and unclipped ZO-SGD generalizes well, but the clipped ZO-SGD has an implicit bias towards higher accuracy (and higher loss) models.

3.4 Connections to robust gradient estimation

One natural conjecture on why the clipped method converges to a solution with higher accuracy is that it provides a *robust estimate* of the gradient, in which case other robust gradient estimators might achieve a similar gain. We test this hypothesis on the following two robust estimators.

ZO optimization with *batch-clipping* applies clipping to the average gradient:

$$g_{\lambda,\bar{c}}(w_t) \;=\; \mathrm{Clip}_{\bar{c}}\Big(\frac{1}{|B_t|}\sum_{i\in B_t}\frac{f_i(w_t+\lambda u_t)-f_i(w_t-\lambda u_t)}{2\lambda}\Big)u_t\;. \tag{5}$$

When applied to the standard first-order SGD, batch-clipping has been empirically shown to achieve better generalization by converging to a wider and flatter basin for FashionMNIST, SVHN, and CIFAR10, when the noise in the gradient is heavy-tailed (Wang et al., 2022; Wang & Rhee, 2025). Their theoretical analysis shows that batch-clipping makes SGD with heavy-tailed noise exponentially slower to escape wider basins. Further, (Zhang et al., 2020) showed that this batch-clipped SGD can achieve faster convergence under (L_0, L_1)- smoothness, where smoothness is proportional to the gradient norm: $\|\nabla^2\mathcal{L}(w)\| \leq L_0 + L_1\|\nabla\mathcal{L}(w)\|$. It was further empirically demonstrated that the training loss of LSTM on Penn Treebank and ResNet20 on CIFAR 10 exhibit (L_0, L_1)-smoothness. In our case of ZO optimization, however, Figure 5 (middle) shows that batch-clipping

cannot drive the optimization to the better solution, and hence cannot achieve SNLI test accuracy above 82%.

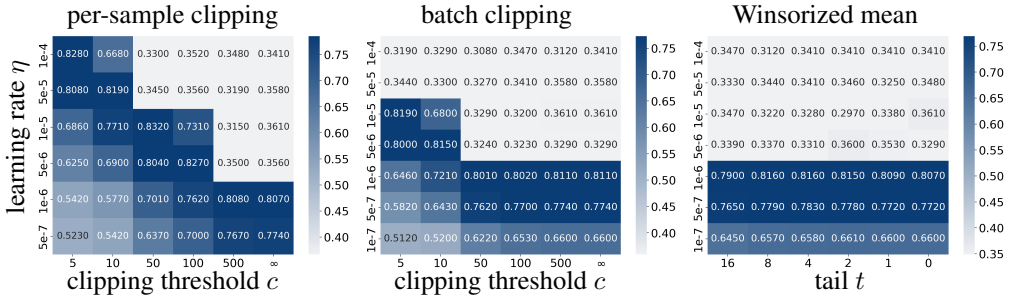

Figure 5: Test accuracy achieved by per-sample clipped ZO-SGD for the same setting as Figure 1 but only for the SNLI fine-tuning task (left). Replacing per-sample clipping with two other robust gradient estimators, $g_{\lambda,\bar{c}}$ in Eq. (5) (middle) and $g_{\lambda,k}$ in Eq. (6) (right), we observe that clipping trade-off phenomenon only happens with the clipped ZO-SGD; no inverse relation between η and strength of clipping c and t, and accuracy capped by 82.9%.

Winsorized mean is a 1D robust mean estimator which adaptively clips the data according to k-th order statistics (see Appendix B for details). We apply this to zeroth-order optimization:

$$g_{\lambda,k}(w_t) = \frac{1}{|B_t|}\sum_{i\in B_t} \mathrm{Clip}_{[L_t,U_t]}\Big(\frac{f_i(w_t+\lambda u_t) - f_i(w_t-\lambda u_t)}{2\lambda}\Big)u_t \ , \tag{6}$$

where L_t and U_t are the $(k+1)$-th and $(|B_t|-k)$-th order statistics of the mini-batch scalar finite differences, and $\mathrm{Clip}_{[L_t,U_t]}(a) = \max(L_t, \min(a, U_t))$. Similar to clipping outside the mean, Figure 5 (right) shows that Winsorized mean cannot achieve the test accuracy of clipped ZO-SGD.

4 Empirical analysis of clipped ZO-SGD

4.1 Clipping leads to miscalibration

To better understand the structural properties of these solutions, we analyze the models' reliability diagrams on SNLI and SST-2 in Figure 6. Following standard evaluation practices (Minderer et al., 2021), we define model confidence for an input x as the maximum predicted probability among all classes, $p^*(x) = \max_y p_w(y|x)$ where $p_w(y|x)$ is the probability assigned to label y on input x for model w. A model achieves perfect top-label calibration when this confidence score accurately reflects the true likelihood of the prediction being correct, meaning $\mathbb{P}_{(x,y)\sim\mathrm{TestSet}}(y \in \arg\max p_w(\cdot|x) \mid |p^*(x) - p| \le 1/(2m)) = p$. To construct these reliability diagrams and calculate the Expected Calibration Error (ECE), the dataset of n samples is partitioned into $m = 15$ bins based on their predicted confidence scores. Within each bin B_i, we compute both the average confidence and the empirical accuracy (the fraction of correct predictions). The overall ECE is then quantified as the expected absolute difference between these two measures across all bins, weighted by the proportion of samples in each (Minderer et al., 2021):

$$\mathrm{ECE} \ = \ \sum_{i=1}^{m} \frac{|B_i|}{n}\big|\mathrm{accuracy}(B_i) - \mathrm{confidence}(B_i)\big|$$

The results in Figure 6 suggest that clipping drives the model to a minima that is miscalibrated with higher loss but better accuracy. This is similar to other highly accurate or strongly regularized modern networks that tend toward systematic overconfidence (Minderer et al., 2021). When comparing configurations along the ηc boundary, the predictive behaviors diverge sharply. The unclipped baseline (w_3) remains well-calibrated (SNLI ECE: 0.0211), distributing its confidence mass more evenly across intermediate bins, accurately reflecting the model's true performance. Conversely, models trained with strict clipping (w_1, w_2) exhibit much higher ECE scores (0.1374 and 0.1067). Tight clipping concentrates the vast majority of predictions into the top confidence bin (> 0.9). While standard ZO-SGD finds minima with calibrated confidence, clipping drives the optimization into a fundamentally different solution.

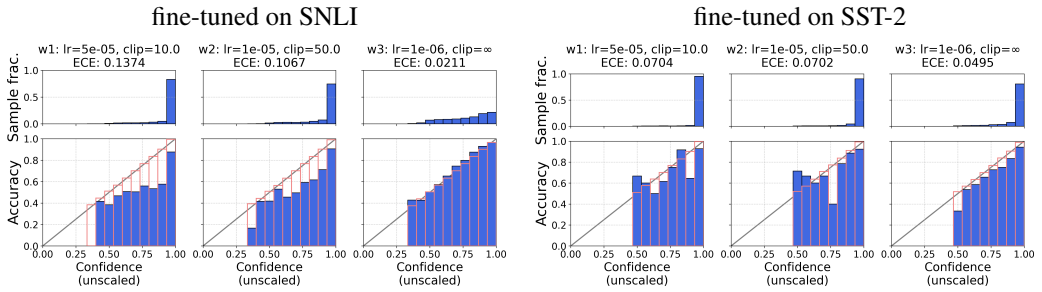

Figure 6: Model calibration after fine-tuning on SNLI (left) and SST-2 (right). Reliability diagrams (DeGroot & Fienberg, 1983) comparing clipped models (w_1, w_2) to an unclipped baseline (w_3). The clipped models consistently display miscalibration (blue bars away from the red box) and higher Expected Calibration Error (ECE).

Figure 6 shows that clipping results in many more confident predictions. We hypothesize that without clipping, there is a significant penalty in being confident on wrong predictions, optimization weighs those examples heavily, and the model learns to be more conservative. With clipping, overconfident wrong predictions do not get fixed as fast and the model can be more aggressive in increasing the confidence of easy examples. This results in the overconfidence of both correct predictions on easy examples and wrong predictions on hard examples. To test this idea, we intentionally plant hard-to-fit examples that have wrong labels in the next section.

4.2 Injected Label Noise Experiment

To further investigate which samples get clipped in clipped ZO-SGD, we introduce artificial label noise by randomly mislabeling subsets of the training data at varying corruption levels, ranging from 0% to 90% as shown in Figure 7. When label noise is injected, the clipped model ($\eta = 1 \times 10^{-5}, c = 50.0$) is more robust, maintaining higher test accuracy than the unclipped baseline. For example, at a 30% corruption rate, the clipped model achieves a 73.7% test accuracy compared to the unclipped model's 68.1%. Generally, this robustness gap between clipped and unclipped ZO-SGD increases as we inject more noisy labels until it reaches 50%. Beyond 50%, clipped ZO-SGD loses its robustness due to the uncorrupted examples being clipped (Figure 8, right). On the other hand, training loss is consistently higher for clipped training, due to the systematic bias that pushes away from minimizing the training loss.

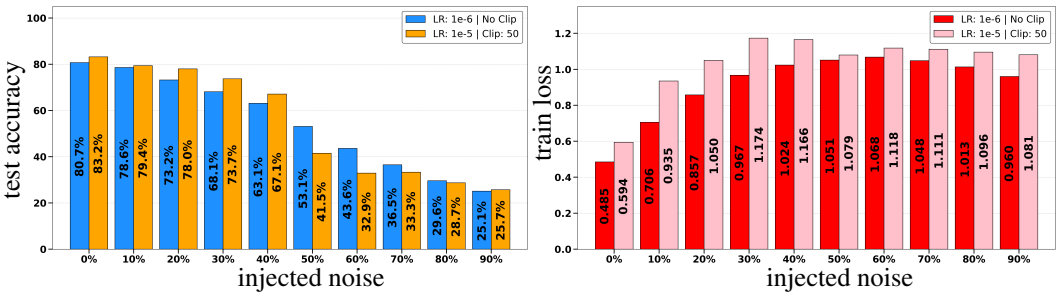

Figure 7: Test accuracy and training loss achieved by models trained with clipped ZO-SGD and unclipped ZO-SGD for various fractions of training data with injected noise. Clipping makes the training more robust to label noise until the injected noise corrupts 50% of the training data.

Figure 7 shows that, in the regime where injected noise fraction is between 0% and 40%, clipped ZO-SGD achieves higher test accuracy. We hypothesize that this is due to robustness of clipping; it down weights hard-to-fit samples, which includes examples with artificially injected label noise, examples with noisy human labels, and examples that are inherently ambiguous what the correct label is (see Figure 9 for examples). Since we can track the samples with injected noise, we can quantify how many times those examples are clipped as shown in Figure 8. This confirms that noise injected examples are clipped significantly more frequently, due to the gradient estimate being larger. This confirms our hypothesis on the origin of robustness when clipping and the resulting accuracy gain. When the injected noise reaches 50% of the training data, the ground-truth labels

become ambiguous and both noise-injected and clean examples behave similarly. The accuracy gain from robustness is also eliminated at this point as we see in Figure 7.

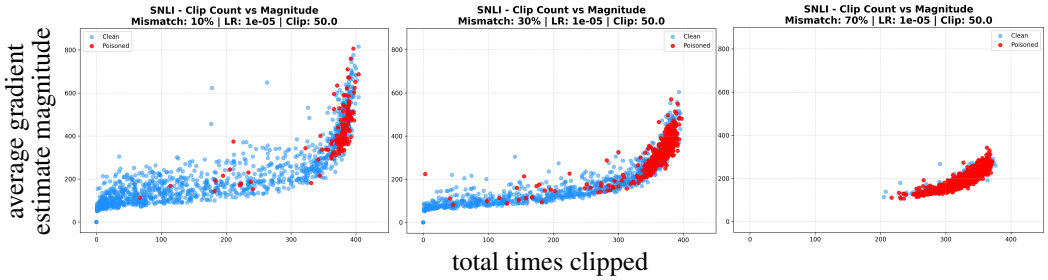

Figure 8: For each of SNLI training example, the average magnitude of the gradient estimates is positively correlated with the number of times the gradient estimate is clipped under clipped ZO-SGD with learning rate $\eta = 1e-5$ and clipping threshold $c = 50$. Examples with injected label noise are clearly clipped more frequently due to larger gradient estimates, when noise injection rate is 10% and 30% (left and middle). When 70% of training data is noisy, it becomes ambiguous what the groundtruth labels are and clean and noisy examples behave similarly (right).

Mislabeled example, ID: 1093

Premise: A blue excavator digging a large hold in cement.
Hypothesis: a child was there
Original SNLI label: Entailment, **Model predicted:** Contradiction
Clipping stats: 389 clips, **Avg magnitude of gradient estimate:** 671.54

Ambiguous example, ID: 1312

Premise: Asian city scene of people in street with bright lights and glass buildings behind.
Hypothesis: There are people in front of buildings that are brightly lit.
Original SNLI label: Entailment, **Model predicted:** Contradiction
Clipping stats: 386 clips, **Avg magnitude of gradient estimate:** 622.05

Figure 9: SNLI training examples that are frequently clipped are mislabeled or ambiguous. Label is for the relation between the premise and the hypothesis. The clipping and magnitude stats are from the 10% corrupted scenario of Figure 8, left panel. More examples in Appendix J.

4.3 Clipping as Implicit Data Pruning

The injected label noise experiments in Section 4.2 lead to a hypothesis that the accuracy gain with clipping in Figure 1 (left) originates from down-weighting the noisy examples in the *original* training datasets. To test this idea, we design the following data-pruning experiment on the SNLI and SST-2 datasets. In the first phase, we train a model with unclipped ZO-SGD with the entire training dataset, and rank the training examples according to their loss. In the second phase, we again train another model from random initialization using unclipped ZO-SGD, but with a subset of $a\%$ of the training data containing the lowest-loss examples for $10 \leq a \leq 100$. As shown in Figure 10, as we decrease a from 100%, perhaps surprisingly, the accuracy increases initially. It reaches its peak when trained with 90% of data, and then keeps degrading afterwards. We believe that there is a certain fraction of ambiguous or mislabeled examples in the original data, which achieve relatively higher loss in the first phase of training. Discarding such examples produces a better performing model. We can interpret clipped ZO-SGD as adaptively detecting and down-weighting such examples, without the need for tuning the hyperparameter a or the need for training twice. This is supported by the fact that in Figure 10, the clipped accuracy in dashed red improves upon the peak performance with pruning.

4.4 Connections to Edge-of-Stability (EoS)

When full batch gradient descent (GD) is applied to training deep neural networks, (Cohen et al., 2021) discovered that the largest eigenvalue, $\lambda_1(H)$, of the Hessian hovers around the threshold of

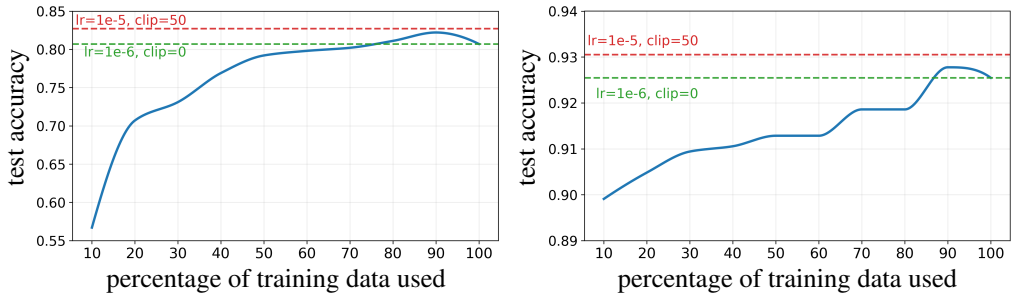

Figure 10: (left) SNLI and (right) SST-2 test accuracy across subsets of the $a\%$ lowest-loss training examples. The test accuracy of unclipped ZO-SGD peaks at 90% in both cases, corresponding to removing the 10% highest-loss examples. This improves over training on 100% of data with ZO-SGD (green dashed line). Clipped ZO-SGD on 100% of the training data (red dashed line) performs slightly better, but more robustly over the hyperparameter choices and with only one training.

linear stability, i.e., $2/\eta$, where η is the step size. This is known as Edge-of-Stability (EoS). In full batch zeroth-order gradient descent, EoS happens around $\mathrm{Trace}(H) \leq 2/\eta \leq \mathrm{Trace}(H) + 2\lambda_1(H)$, as empirically shown by Song et al. (2026). We first reproduce this EoS result by training a CNN on CIFAR-10 from random initialization (Figure 11, left), where the $2/\eta$ line is sandwiched between the predicted upper and lower bounds. With per-sample clipping, the curvature bounds become noisier and exhibit larger oscillations before eventually converging to the EoS threshold (Figure 11, middle). However, the emergence of EoS appears to depend on the initialization regime. When the same model is initialized with ImageNet-pretrained weights, it does not reach the EoS threshold (Figure 11, right). Hence, we should not expect EoS in our setting of fine-tuning LLMs, which is confirmed in our experiments (appx. I)

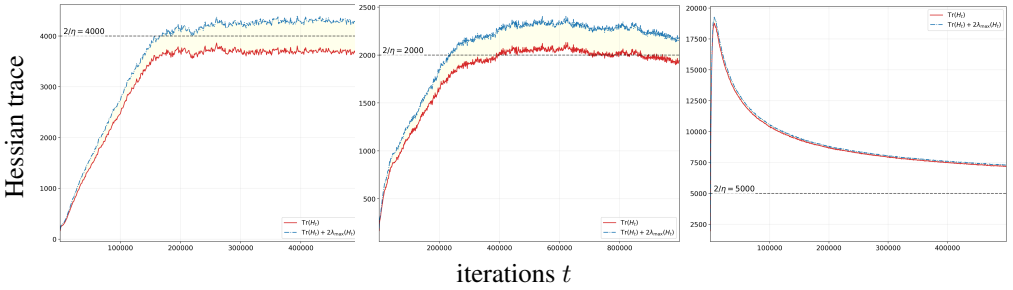

Figure 11: (left) We reproduce the training from random initialization on a subset of CIFAR-10 to demonstrate the EoS phenomenon reported in (Song et al., 2026). (middle) We repeat the experiment shown in the left panel with per-sample clipping. (right) We reproduce the EoS phenomenon reported in the leftmost panel using ImageNet-pretrained ResNet weights.

5 Conclusion

Zeroth-order optimization is emerging as an important resource-efficient tool for fine-tuning LLMs, due to the fact that they do not require backpropagation and hence require less memory. Given its increasing relevance, we systematically investigate a recently observed phenomenon where clipping each gradient estimate in ZO-SGD provides a better accuracy in both train and test data. We first show that this phenomenon is robustly observed in four fine-tuning tasks and two families of pre-trained LLMs. We demonstrate that, as predicted by the asymptotic model update in the *fully clipped regime*, the optimal choice of step size η is inversely proportional to the clipping threshold c, and as long as we keep ηc fixed, the clipping gain persists for a wide range of these hyperparameters. In this regime, the model converges to a solution different from the unclipped ZO-SGD, and achieves higher accuracy. We further show that this phenomenon is unique to per-sample clipped ZO-SGD, by comparing it to other robust gradient aggregation methods applied to ZO-SGD. Our experiments with artificially injected label noise and retraining with pruned training data suggests that the gain is coming from the inherent down weighting of hard-to-fit examples in clipped ZO-SGD.

AI USAGE

In this work, we used generative AI tools to correct grammar and spelling errors, improve readability, and assist in developing the experimental codebase (specifically for debugging and adapting to resource limitations), and identify relevant literature. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate or prove mathematical claims, propose hypotheses, design experiments, clean datasets, or interpret results, and the remaining required disclosure tasks (such as translation and qualitative data analysis) are not applicable to this work.

We have carefully reviewed all AI-assisted work. Any code written or modified by LLMs was manually verified, tested, and executed by the authors to guarantee correctness. We also independently read and verified the accuracy and relevance of all AI-suggested literature before citing it in this paper. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To ensure the full reproducibility of our findings, we have detailed our theoretical and experimental frameworks throughout the paper and its appendices. Additionally we will make the complete codebase publicly available at the release of the camera ready version. The complete mathematical proof for the asymptotic behavior of the clipped gradient estimator (Proposition 2.1) is provided in Appendix C . Full details of our experimental setup, including dataset processing for SNLI, MNLI, SST-2, and TREC, model specifications for RoBERTa and OPT, and exact hyperparameter sweep ranges for the learning rate and clipping threshold, are explicitly documented in Appendix D. The step-by-step formulation for the Winsorized mean baseline is detailed in Appendix B. Exhaustive learning curves, loss landscapes, and hyperparameter configurations for all specific tasks are available in Appendices E, F, and G. Finally, the source code required to execute all Zeroth-Order fine-tuning and label noise experiments will be provided in the supplementary materials. Our implementation builds directly upon the original DPZero and ZO codebases, from (Zhang et al., 2024a) and (Song et al., 2026), and includes all custom add-ons to that existing framework necessary to reproduce the specific experiments presented in this paper.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In *International Conference on Machine Learning*, 2022.

Samuel Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *The Eleventh International Conference on Learning Representations*, 2023.

S Arora, Z Li, and A Panigrahi. Understanding gradient descent on the edge of stability in deep learning. In *International Conference on Machine Learning*, 2022.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, 2015.

Aochuan Chen, Yimeng Zhang, Jinghan Jia, James Diffenderfer, Konstantinos Parasyris, Jiancheng Liu, Yihua Zhang, Zheng Zhang, Bhavya Kailkhura, and Sijia Liu. Deepzero: Scaling up zeroth-order optimization for deep model training. In *International Conference on Learning Representations*, volume 2024, pp. 50185–50206, 2024.

Jiahe Chen and Ziye Ma. Accelerating zeroth-order spectral optimization with partial orthogonalization from power iteration. *arXiv preprint arXiv:2605.09034*, 2026.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 15–26, 2017.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in neural information processing systems*, 32, 2019.

Xiangyi Chen, Steven Z Wu, and Mingyi Hong. Understanding gradient clipping in private SGD: A geometric perspective. In *Advances in Neural Information Processing Systems*, volume 33, pp. 13773–13782, 2020.

Jeremy M Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021.

A Damian, E Nichani, and JD Lee. Self-stabilization: The implicit bias of gradient descent at the edge of stability. In *International Conference on Learning Representations*, 2023.

Morris H DeGroot and Stephen E Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Huang Fang, Xiaoyun Li, Chenglin Fan, and Ping Li. Improved convergence of differential private SGD with gradient clipping. In *International Conference on Learning Representations*, 2023.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Advances in neural information processing systems*, volume 31, 2018.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pp. 2137–2146. PMLR, 2018.

Anastasia Koloskova, Hadrien Hendrikx, and Sebastian U Stich. Revisiting gradient clipping: Stochastic bias and tight convergence guarantees. In *International Conference on Machine Learning*, 2023.

Yicheng Lang, Changsheng Wang, Yihua Zhang, Mingyi Hong, Zheng Zhang, Wotao Yin, and Sijia Liu. Powering up zeroth-order training via subspace gradient orthogonalization. *arXiv preprint arXiv:2602.17155*, 2026.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in neural information processing systems*, volume 31, 2018.

Xin Li and Dan Roth. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.

Tianyi Lin, Zeyu Zheng, and Michael Jordan. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems*, 35: 26160–26175, 2022.

Sijia Liu, Bhavya Kailkhura, Pin-Yu Chen, Paishun Ting, Shiyu Chang, and Lisa Amini. Zeroth-order stochastic variance reduction for nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

Sijia Liu, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred O Hero III, and Pramod K Varshney. A primer on zeroth-order optimization in signal processing and machine learning. *IEEE Signal Processing Magazine*, 37(5):43–54, 2020.

Sijia Liu, Yicheng Lang, Soumyadeep Pal, Changsheng Wang, Yancheng Huang, Chongyu Fan, James Diffenderfer, Bhavya Kailkhura, and Yihua Zhang. Position: Zeroth-order optimization in deep learning is underexplored, not underpowered. *arXiv preprint arXiv:2605.15622*, 2026.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53038–53075, 2023.

Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Ann Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lučić. Revisiting the calibration of modern neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 15682–15694. Curran Associates, Inc., 2021.

Anshul Nasery, Jonathan Hayase, Pang Wei Koh, and Sewoong Oh. Pleas—merging models with permutations and least squares. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30493–30502. IEEE, 2025.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning?, 2021. URL https://arxiv.org/abs/2008.11687.

Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(1):1703–1713, 2017.

Richard Socher, Alex Perelygina, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Minhak Song, Liang Zhang, Bingcong Li, Niao He, Michael Muehlebach, and Sewoong Oh. Zeroth-order optimization at the edge of stability. *arXiv preprint arXiv:2604.14669*, 2026.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. Black-box tuning for language-model-as-a-service. In *International Conference on Machine Learning*, pp. 20841–20855. PMLR, 2022.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Liyan Tan, Yequan Zhao, Yifan Yang, Ruijie Zhang, Xinling Yu, and Zheng Zhang. Grzo: Group-relative zeroth-order optimization for large language model fine-tuning. *arXiv preprint arXiv:2606.02857*, 2026.

Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 742–749, 2019.

Xingyu Wang and Chang-Han Rhee. Global dynamics of heavy-tailed sgds in nonconvex loss landscape: Characterization and control. *arXiv preprint arXiv:2510.20905*, 2025.

Xingyu Wang, Sewoong Oh, and Chang-Han Rhee. Eliminating sharp minima from sgd with truncated heavy-tailed noise. In *International Conference on Learning Representations*, 2022.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, 2018.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pp. 23965–23998. PMLR, 2022.

Haishan Ye, Zhichao Huang, Cong Fang, Chris Junchi Li, and Tong Zhang. Hessian-aware zeroth-order optimization for black-box adversarial attack. *arXiv preprint arXiv:1812.11377*, 2018.

Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations*, 2020.

Liang Zhang, Bingcong Li, Kiran Koshy Thekumparampil, Sewoong Oh, and Niao He. DPZero: Private fine-tuning of language models without backpropagation. In *International Conference on Machine Learning (ICML)*, 2024a.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022a.

Y Zhang, P Li, J Hong, J Li, Y Zhang, W Zheng, P-Y Chen, JD Lee, W Yin, M Hong, et al. Revisiting zeroth-order optimization for memory-efficient LLM fine-tuning: A benchmark. In *International Conference on Machine Learning*, 2024b.

Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jinfeng Yi, Mingyi Hong, Shiyu Chang, and Sijia Liu. How to robustify black-box ml models? a zeroth-order optimization perspective. *arXiv preprint arXiv:2203.14195*, 2022b.

A Related Work

A.1 Zeroth-Order Optimization

Zeroth-order (ZO) optimization has become increasingly relevant for scenarios where computing exact gradients is either computationally prohibitive or strictly impossible (see (Liu et al., 2026) for a survey of latest advances). Classical applications heavily focused on black-box optimization (Nesterov & Spokoiny, 2017; Ghadimi & Lan, 2013; Shamir, 2017; Lin et al., 2022), where access to the objective function is limited to forward evaluations. Significant algorithmic advances have resulted from this productive line of research, including (Zhang et al., 2022b; Chen et al., 2024; Ilyas et al., 2018; Sun et al., 2022; Tan et al., 2026; Chen & Ma, 2026; Lang et al., 2026). Recently, however, there has been a significant push to adapt ZO techniques for memory-efficient training of large neural networks. Because backpropagation requires caching intermediate activations, scaling standard first-order methods to billions of parameters often exceeds available hardware memory. Recent approaches like MeZO (Malladi et al., 2023; Zhang et al., 2024b;a) address this by utilizing a standard two-point ZO estimator, demonstrating that models can be effectively fine-tuned using only forward passes. This dramatically reduces the memory footprint, making ZO an attractive alternative for aligning and adapting modern language models under strict hardware constraints.

A.2 Gradient Clipping and Private Fine-Tuning

Beyond standard memory constraints, ZO methods are actively being adapted to solve challenges in differentially private (DP) machine learning. A fundamental mechanism in deep DP learning is gradient clipping, which tightly bounds the sensitivity of model updates to ensure no single training example disproportionately influences the weights (Abadi et al., 2016; Chen et al., 2020; Fang et al., 2023; Koloskova et al., 2023).

In traditional DP-SGD, clipping requires the explicit computation and materialization of per-example gradients, which exacerbates the memory bottleneck of backpropagation. To circumvent this, recent work integrates clipping directly into the zeroth-order estimation process. By clipping the scalar loss differences or the forward-pass directional derivatives, algorithms like DPZero (Zhang et al., 2024a) can bound the sensitivity of the update step without ever performing backpropagation. This use of clipping seamlessly bridges the memory benefits of ZO optimization with the rigorous privacy guarantees of DP, allowing for the private fine-tuning of massive autoregressive architectures that would otherwise be impossible to train with standard first-order clipping techniques.

A.3 The Edge of Stability

The optimization dynamics of deep neural networks frequently exhibit a non-equilibrium behavior known as the Edge of Stability (EoS). Originally documented in first-order full-batch gradient descent (Cohen et al., 2021; Ahn et al., 2022; Arora et al., 2022; Damian et al., 2023), the EoS phenomenon occurs when a network's training trajectory naturally gravitates toward and oscillates around regions where the maximum eigenvalue of the Hessian (often referred to as the "sharpness") hovers just above $2/\eta$, where η is the learning rate.

While first-order stability is dictated almost entirely by this maximum eigenvalue, the dynamics shift significantly under zeroth-order methods. Because ZO methods rely on stochastic gradient estimates, their mean-square linear stability is governed not just by the sharpness, but by the entire Hessian spectrum (Song et al., 2026). Specifically, theoretical bounds show that ZO methods operate at a distinct edge of stability characterized by a dependence on both the maximum eigenvalue and the Hessian trace. When trained with large step sizes, ZO optimization induces an implicit regularization effect that suppresses the growth of the Hessian trace, forcing the model to converge along a modified stability boundary uniquely dictated by the variance of the zeroth-order estimator.

B Detailed Formulation of the Winsorized Mean Calculations

To isolate the specific dynamics induced by static gradient clipping, we benchmark our approach against alternative statistical robustification techniques, most notably the Winsorized mean. Unlike

standard clipping, which applies a fixed, data-independent threshold c, Winsorization dynamically bounds the finite differences based on the empirical distribution of the current batch.

For a given batch B_t at iteration t, we first compute the scalar finite difference for each sample $i \in B_t$ along the random perturbation direction u_t:

$$v_i = \frac{f_i(x_t + \lambda u_t) - f_i(x_t - \lambda u_t)}{2\lambda} \, . \tag{7}$$

To apply Winsorization, we sort these scalars to obtain the order statistics:

$$v_{(1)} \leq v_{(2)} \leq \cdots \leq v_{(|B_t|)} \, . \tag{8}$$

For a chosen trimming parameter k (where $0 < k < |B_t|/2$), we define the dynamic lower and upper bounds as the $(k+1)$-th and the $(|B_t| - k)$-th order statistics, respectively:

$$L_t = v_{(k+1)} \quad \text{and} \quad U_t = v_{(|B_t|-k)} \, . \tag{9}$$

The k-Winsorized method replaces the k lowest values with L_t and the k highest values with U_t. The zeroth-order gradient estimator is then computed as the mean of these bounded values:

$$g_\lambda(x_t) = \left(\frac{1}{|B_t|} \sum_{i \in B_t} \mathrm{Clip}_{[L_t, U_t]}(v_i) \right) u_t \, , \tag{10}$$

where the bounding function is defined as:

$$\mathrm{Clip}_{[L_t, U_t]}(a) = \begin{cases} L_t & \text{if } a < L_t \\ a & \text{if } L_t \leq a \leq U_t \\ U_t & \text{if } a > U_t \end{cases} \, . \tag{11}$$

This formulation ensures that the gradient estimate remains robust to extreme outlier perturbations without completely discarding the mass of those samples, offering a data-dependent alternative to standard threshold clipping.

C Proof of Proposition 2.1

We first prove that for any $g \in \mathbb{R}^d$ and $c > 0$,

$$\begin{aligned} \mathbb{E}_{u \sim \mathcal{N}(0, \mathbf{I}_{d\times d})} \big[\, \mathrm{Clip}_c(\langle u, g \rangle) u \, \big] \; &= \; \mathbb{E}_{u \sim \mathcal{N}(0, \mathbf{I}_{d\times d})} \big[\, \nabla_u \mathrm{Clip}_c(\langle u, g \rangle) \, \big] \\ &= \; \mathbb{E}_{u \sim \mathcal{N}(0, \mathbf{I}_{d\times d})} \big[\, g \, \mathbf{1}_{|\langle u, g \rangle| < c} \, \big] \\ &= \; \big(2\Phi(c/\|g\|) - 1 \big) g \, , \end{aligned}$$

where $\Phi(\cdot)$ is the standard normal CDF. The first line follows from Stein's identity for a 1-Lipschitz function $\mathrm{Clip}_c(\cdot)$. Together with the fact (Duchi et al., 2015) that

$$\lim_{\lambda \to 0} g_{\lambda, c}(w) \; = \; \langle u, (1/|B_t|) \sum_{i \in B_t} \nabla f_i(w) \rangle u \, ,$$

this proves the desired claim by replacing g with $\nabla f_i(w)$ for each term.

D Experimental Setup

The primary goal of our evaluation is to understand how per-sample gradient clipping affects the convergence behavior of Zeroth-Order (ZO) optimization. Our study is motivated by an observation from the original DPZero framework by Zhang et al. (Zhang et al., 2024a). Although their work focused on differential privacy, they noted that clipping also improved optimization in the non-private setting:

> "...the non-private baseline MeZO also appears to benefit from clipping. For instance, without clipping, the original MeZO encounters non-convergence issues at a stepsize of 5×10^{-6}. Conversely, incorporating clipping permits the use of larger stepsizes and yields better results. A thorough investigation of this phenomenon is reserved for future research."

This paper investigates that phenomenon directly. We adapt the DPZero framework for large language model fine-tuning while removing all privacy-related components by disabling Gaussian noise injection ($\sigma = 0$). This isolates the clipping threshold (c) and learning rate (η) as the only variables under study.

The original DPZero work evaluated both RoBERTa and OPT models using a few-shot learning setup with at most 512 training samples per class. We follow the same setup and evaluate performance across tasks of varying complexity: sentiment analysis on SST-2 (Socher et al., 2013), natural language inference on SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018), and question classification on TREC (Li & Roth, 2002).

For each model–dataset pair, we perform a grid search over learning rates ($\eta \in [5\times10^{-7}, 1\times10^{-4}]$) and clipping thresholds ($c \in [1.0, 500.0]$), including an unclipped baseline ($c = \infty$).

We then evaluate several aspects of clipped ZO optimization. First, we study the trade-off between clipping and learning rate by measuring final validation accuracy and convergence speed, allowing us to characterize the observed "diagonal trend" between η and c. Second, we identify failure modes by tracking training loss to determine when models either diverge due to high variance or fail to learn because clipping is too restrictive. Finally, we analyze how clipping shapes the loss landscape by measuring distance from initialization ($\|w_t - w_0\|$) and evaluating linear mode connectivity between checkpoints. Together, these experiments map how different (η, c) settings influence convergence behavior and solution geometry.

E Additional experimental results for clipped zeroth-order optimization: RoBERTa

We provide the train accuracy and loss when sweeping through learning rate η and clipping threshold c in the figure below.

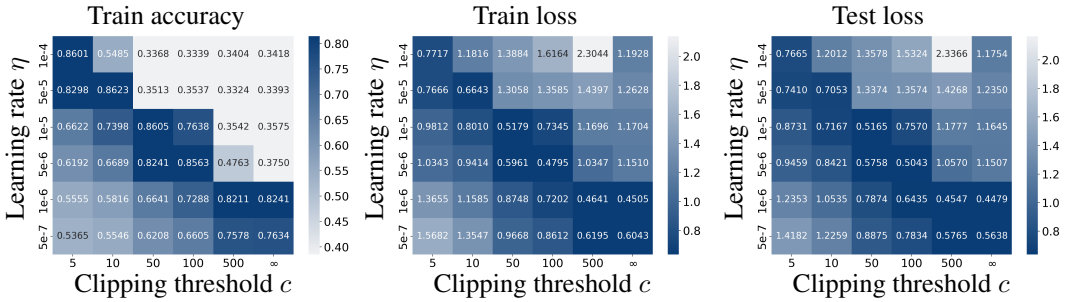

Figure 12: Each cell shows the train accuracy (left), train loss (middle), and test loss (right) averaged over four models fine-tuned from RoBERTa (Liu et al., 2019) on SNLI (Bowman et al., 2015), MNLI (Williams et al., 2018), SST-2 (Socher et al., 2013) and Trec (Li & Roth, 2002), respectively. $c = \infty$ is the no clipping baseline.

We provide extended empirical results for the fine-tuning of RoBERTa (355M) using the clipped zeroth-order (ZO) estimator. We evaluate the performance across four standard downstream classification tasks: SST-2, MNLI, SNLI, and TREC. Across all datasets, we observe a consistent, structural interplay between the learning rate (η) and the clipping threshold (c).

E.1 Observed Trends: SNLI

The Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) is a widely adopted benchmark formatted as a 3-class problem to predict entailment, contradiction, or neutral relation-

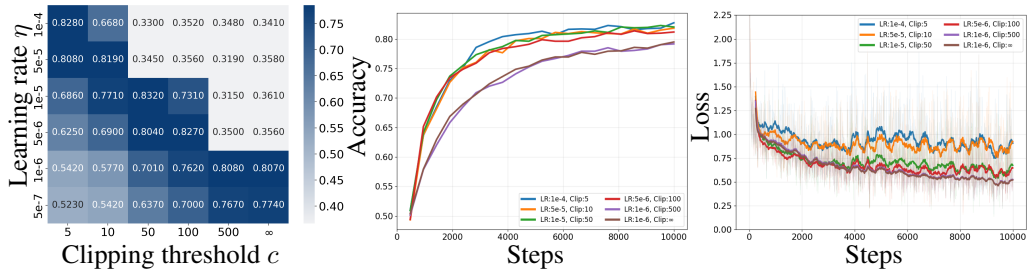

Figure 13: **Left:** Test accuracy grid for SNLI on RoBERTA after sweeping through the learning rate η and the clipping threshold c, we observe that the optimal learning rate for each clipping threshold lies on the curve $\eta \times c = 5 \times 10^{-4}$. **Middle:** Learning curves for η and c pairs along the ηc line depicting the various rates of convergence. **Right:** SNLI train loss of the configurations along the line $\eta\, c$ = constant line. The higher loss and higher accuracy achieved along the diagonal ($\eta c = 5 \times 10^{-4}$) with clipping ($c \leq 100$) compared to the no-clipping best loss implies that the gain of clipping is from better generalization and not better optimization.

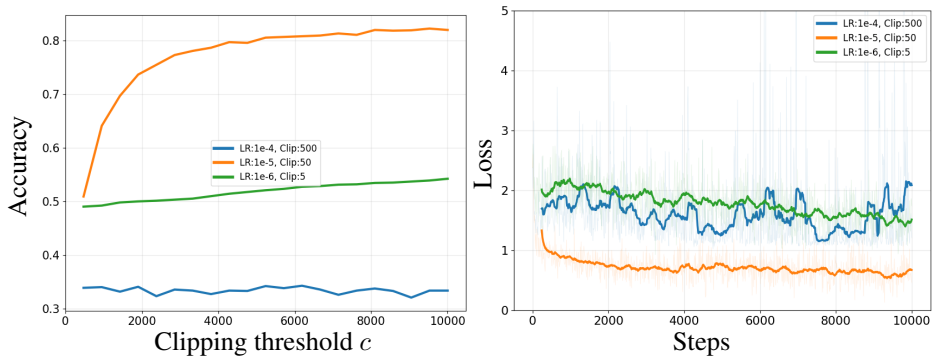

Figure 14: SNLI Learning curves (**left**) and loss curves (**right**) comparing configurations that deviate from the ηc diagonal ($\eta = 10^{-4}, c = 500$ and $\eta = 10^{-6}, c = 5$) against the configuration $\eta = 10^{-5}, c = 50$. Off-diagonal configurations show two failure modes: high η with large c leads to unstable training and random-guess accuracy (about 33%), while low η with small c yields stable but overly slow convergence, plateauing at 49–57% accuracy within 10,000 steps.

ships between sentence pairs. Training with the clipped zeroth-order estimator reveals a strict, inversely proportional trend between the learning rate (η) and the clipping threshold (c). The test accuracy heatmaps demonstrate that as the learning rate increases, the clipping threshold must be correspondingly decreased to maintain stability. The peak performance zone, yielding the highest test accuracies ($\sim$83%), lie along this optimal diagonal band (e.g., $\eta = 5 \times 10^{-5}$ paired with $c = 10$, and $\eta = 1 \times 10^{-5}$ paired with $c = 50$) satisfying the *clipping trade-off* at $\eta\, c =$ constant.

Configurations deviating from this diagonal exhibit two distinct failure modes. In the upper-right region of the hyperparameter grid (high η, large c), the models fail to learn entirely. The test loss trajectories exhibit extreme, volatile spikes, and accuracy flatlines at the random guessing baseline of approximately 33%. Conversely, configurations in the lower-left region (low η, small c) produce stable and smooth loss curves. However, their convergence is excessively slow; these models fail to reach peak accuracy within the allocated 10,000-step limit, stalling between 49% and 57% accuracy (Figure 14).

Further examination of the continuous learning trajectories reveals that hyperparameter configurations split into distinct behavioral groups. Aggressive pairings—characterized by larger learning rates and smaller clipping thresholds (e.g., $\eta = 10^{-4}, c = 5$ and $\eta = 5 \times 10^{-5}, c = 10$)—exhibit rapid initial convergence, quickly reaching near-peak accuracy within the first 2,000 to 4,000 iterations. However, this speed is accompanied by a highly variant and elevated training loss profile. Instead, conservative configurations with lower learning rates and larger clipping (e.g.,

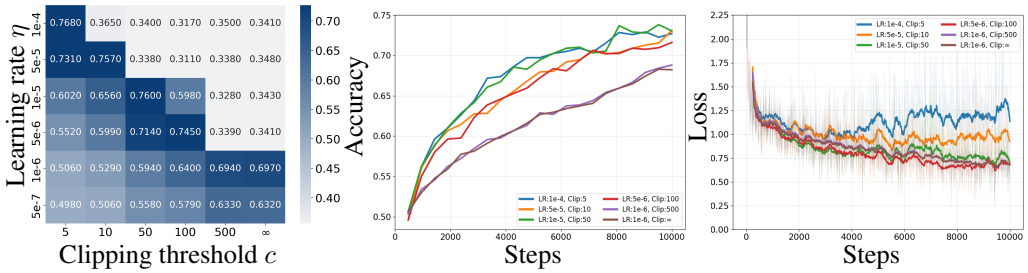

Figure 15: **Left:** MNLI test accuracy for RoBERTa across learning rates η and clipping thresholds c. The best learning rate at each threshold follows $\eta c = 5 \times 10^{-4}$. **Middle:** Learning curves for (η, c) pairs along this line, showing different convergence rates. **Right:** MNLI training loss for configurations satisfying $\eta c =$ constant. As with SNLI (Appendix E.1), clipped runs ($c \leq 100$) achieve higher loss but better accuracy than the best unclipped run, suggesting clipping improves generalization rather than optimization.

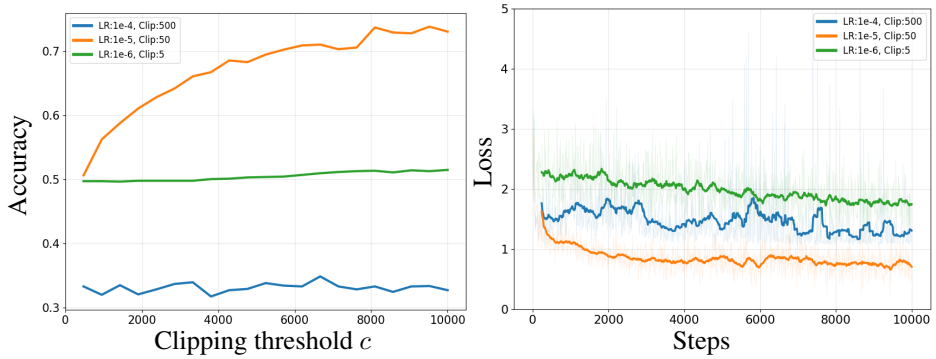

Figure 16: MNLI learning curves (**left**) and loss curves (**right**) comparing configurations that deviate from the ηc diagonal ($\eta = 10^{-4}, c = 500$ and $\eta = 10^{-6}, c = 5$) against the configuration $\eta = 10^{-5}, c = 50$. Similar to SNLI, the off-diagonal settings perform worse: high η with loose c causes unstable training and random-guess accuracy, while low η with strict c converges slowly and plateaus below the baseline.

$\eta = 10^{-6}, c = 500$ or $c = \infty$) display a much shallower ascent in accuracy, but descend cleanly to lower, smoother training loss plateaus. The magnitude of the clipping threshold dictates the optimization trajectory, steering the models to converge into distinctly different solutions.

E.2 Observed Trends: MNLI

Like SNLI, the Multi-Genre Natural Language Inference (MNLI) corpus (Williams et al., 2018) is a widespread 3-class sentence understanding benchmark requiring models to predict entailment, contradiction, or neutral relationships across diverse text genres.

Empirical results on the MNLI dataset perfectly mirror the structural dynamics observed in SNLI, confirming that the clipping trade-off is a robust and consistent phenomenon across different datasets. The test accuracy heatmap reveals that the highest performance (approximately 76%) also lies strictly along the established diagonal cells, maintaining an inversely proportional relationship between the learning rate (η) and the clipping threshold (c).

Deviating from this optimal band yields two distinct failure modes. In the high η and large c regime, failing to sufficiently clip the large learning rates results in the accuracy remaining trapped at the 33% random-guessing baseline. On the opposite end of the spectrum, employing a low η with a small c produces smooth, linear accuracy climbs. However, the step sizes are too small, preventing the model from traversing the loss landscape quickly enough to reach a similar performance of the configurations along the diagonal band within the 10,000-step computational limit.

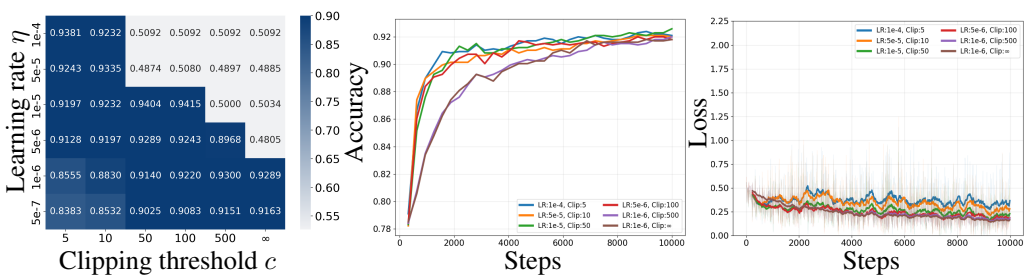

Figure 17: The clipping trade-off trend is less prominent for Roberta fine-tuned on SST-2.

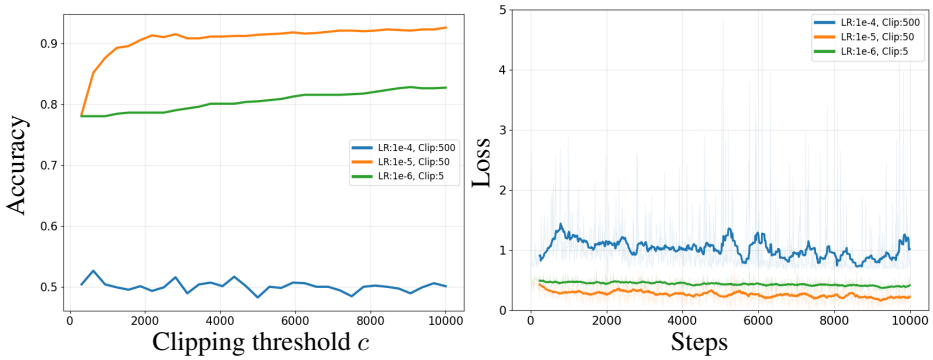

Figure 18: SST-2 learning (**left**) and loss (**right**) curves comparing off-diagonal configurations ($\eta = 10^{-4}, c = 500$ and $\eta = 10^{-6}, c = 5$) with $\eta = 10^{-5}, c = 50$. Similar to SNLI and MNLI, the off-diagonal settings perform worse: high η with loose c leads to unstable training and chance-level accuracy, while low η with strict c converges slowly and plateaus below the baseline.

E.3 Observed Trends: SST-2

The Stanford Sentiment Treebank (SST-2) (Socher et al., 2013) is a standard binary classification task designed to predict the sentiment of movie reviews. Training with the clipped zeroth-order estimator also shows the trade-off. The optimal performance is concentrated along the diagonal band, maintaining the inversely proportional relationship between the learning rate (η) and the clipping threshold (c). Configurations parameterized along this optimal band (e.g., $\eta = 1 \times 10^{-5}$ paired with $c = 50$) achieve the highest convergence, consistently exceeding 93% test accuracy. Outside this optimal diagonal exposes severe optimization bottlenecks similar to SNLI and MNLI.

E.4 Observed Trends: TREC

The TREC dataset (Li & Roth, 2002) is a question classification benchmark requiring models to categorize open-domain, fact-based questions into one of six distinct semantic categories. Training

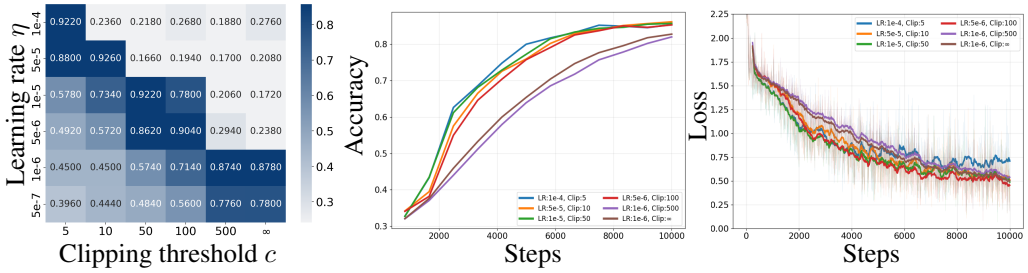

Figure 19: Clipping trade-off trends for RoBERTa fine-tuned on TREC, showing the same *clipping trade-off* behavior observed in E.1, E.2, and E.3.

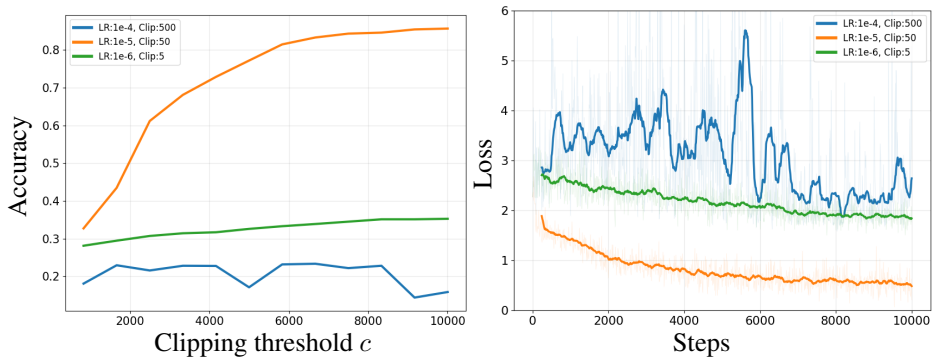

Figure 20: TREC learning curves (**left**) and loss curves (**right**) comparing configurations that deviate from the ηc diagonal ($\eta = 10^{-4}, c = 500$ and $\eta = 10^{-6}, c = 5$) against the configuration $\eta = 10^{-5}, c = 50$. Similar to SNLI, SST-2, and MNLI, the off-diagonal settings perform worse: high η with loose c causes unstable training and random-guess accuracy, while low η with strict c converges slowly and plateaus below the baseline.

with the clipped zeroth-order estimator on this 6-class problem confirms the clipping trade-off. Configurations parameterized away from this stability boundary exhibit severe performance degradation 20.

F Additional Experiments: OPT

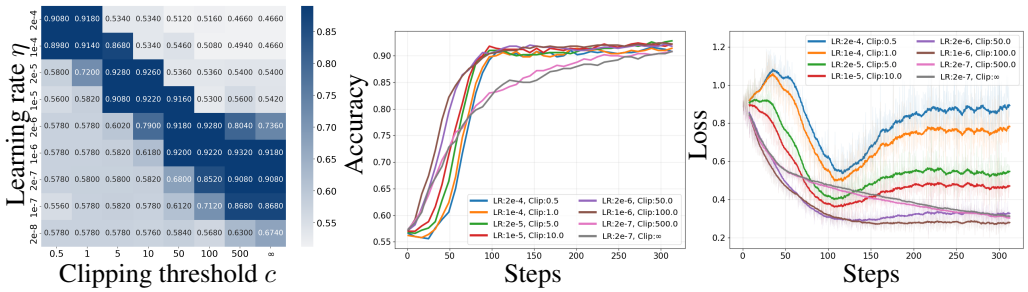

Figure 21: Experiments on private fine-tuning OPT (1.3B) for SST-2 with 2. **Left:** Test accuracy % when varying the stepsize and clipping threshold together. **Middle:** Accuracy plots depicting the convergence of the diagonal band configurations. **Right:** (Smoothed) training loss curves of the diagonal band configurations. Consistent with previous experiments, we observe a similar behavior with clipping and zeroth order methods converging with different behaviors when changing models from Roberta to OPT.

F.1 Observed Trends: SST-2

To verify that the clipping trade-off generalizes to beyond Roberta for more advanced architectures, we extend our evaluation to the OPT (Zhang et al., 2022a) model fine-tuned on the SST-2 binary classification benchmark (Socher et al., 2013). The test accuracy heatmap demonstrates that the OPT architecture strongly preserves the diagonal scaling trend previously observed in encoder-only models. Peak performance forms a distinct structural band where higher learning rates (η) strictly require tighter clipping thresholds (c). Accuracies exceeding 90%, and reaching up to 92.8%, are clustered entirely along this optimal diagonal (e.g., $\eta = 2 \times 10^{-5}$ paired with $c = 5.0$, and $\eta = 1 \times 10^{-6}$ paired with $c = 100.0$). Deviating from this boundary yields familiar failure modes. In the upper-right regime (e.g., $\eta = 1 \times 10^{-4}$ with $c \geq 10.0$), the unconstrained variance causes complete model failure, with accuracy catastrophically collapsing to the binary random-guessing baseline of approximately 49% to 54%. In the overly constrained lower-left region (e.g., $\eta \leq 1 \times 10^{-6}$ with

$c \leq 1.0$), the model avoids this collapse but stagnates, plateauing at a sub-optimal 57% to 58% accuracy within the given training budget.

An analysis of the continuous learning trajectories reveals unified convergence for configurations situated on the optimal diagonal. Despite utilizing drastically different hyperparameter pairings, all sampled configurations successfully navigate to a similar peak accuracy of 91% to 93%. However, the speed of this traversal varies significantly. Configurations parameterized with a higher base learning rate (e.g., $\eta = 2 \times 10^{-5}, c = 5.0$) rapidly optimize the objective, achieving 90% accuracy in fewer than 50 epochs. Conversely, conservative setups utilizing lower learning rates and looser clipping thresholds (e.g., $\eta = 1 \times 10^{-6}, c = 100.0$) exhibit a much shallower ascent, requiring 150 to 200 epochs to cross the same performance threshold.

Examining the corresponding training loss profiles exposes a direct correlation between the learning rate and the zeroth-order estimator's variance. High-η configurations inherently display highly volatile, thick loss bands throughout the training run, reflecting the larger stochastic perturbations. In contrast, configurations with low learning rates (e.g., $\eta = 2 \times 10^{-7}, c = 500.0$) produce tightly clustered, smooth loss curves.

G Additional experiments from Section 3

G.1 Loss Landscape

As discussed in Section 3.2, previous experiments suggest that hyperparameter choices along the *clipping trade-off line* (i.e., $\eta c = 5 \times 10^{-4}$) converge to two different types of solutions depending on whether clipping is significant enough ($c \leq 100$). To visualize the geometry of these solutions, similar to (Li et al., 2018; Garipov et al., 2018), we project the high-dimensional test loss and test accuracy landscapes onto the 2D plane spanned by three model checkpoints along this clipping trade-off line: w_1 ($\eta = 5e{-}5$, $c = 10$), w_2 ($\eta = 1e{-}5$, $c = 50$), and w_3 ($\eta = 1e{-}6$, $c = \infty$). Projecting these test metrics onto the exact same 2D plane defined by checkpoints w_1, w_2, and w_3, we observe that the test landscapes perfectly mirror the training dynamics in 2. The test loss landscape (Figure 22, left) shows that the unclipped baseline (w_3) resides in a local minima characterized by lower overall test loss. However, the test accuracy landscape (Figure 22, right) demonstrates that the clipped models (w_1 and w_2) navigate to a fundamentally distinct basin that achieves superior generalization accuracy, despite residing in a region of higher test loss. This confirms that the implicit bias introduced by per-sample clipping, trading higher loss for improved accuracy, is not an artifact of training set memorization, but a robust structural property that translates directly to test performance.

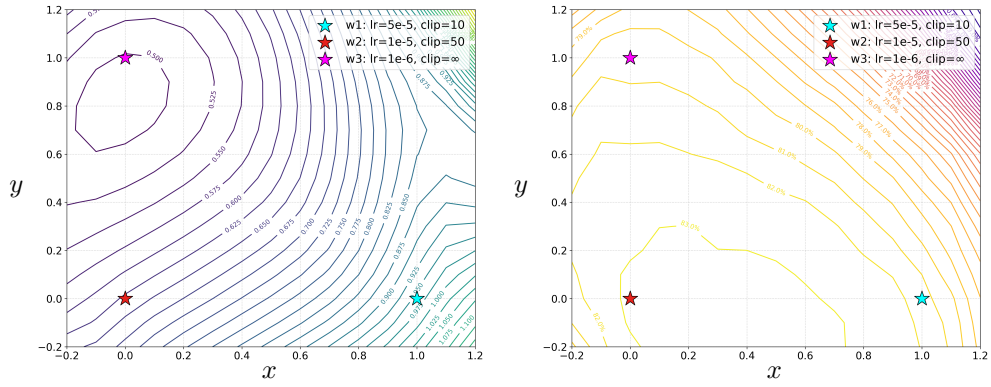

Figure 22: Pretrained RoBERTa fine-tuned on SNLI. (left) Test loss landscape projected down to (x, y)-plane evaluated at $w_2 + (w_1 - w_2)x + (w_3 - w_2)y$ for three model checkpoints w_1, w_2, w_3. (right) Test Accuracy landscape projected down to (x, y)-plane evaluated at $w_2 + (w_1 - w_2)x + (w_3 - w_2)y$ for three model checkpoints w_1, w_2, w_3.

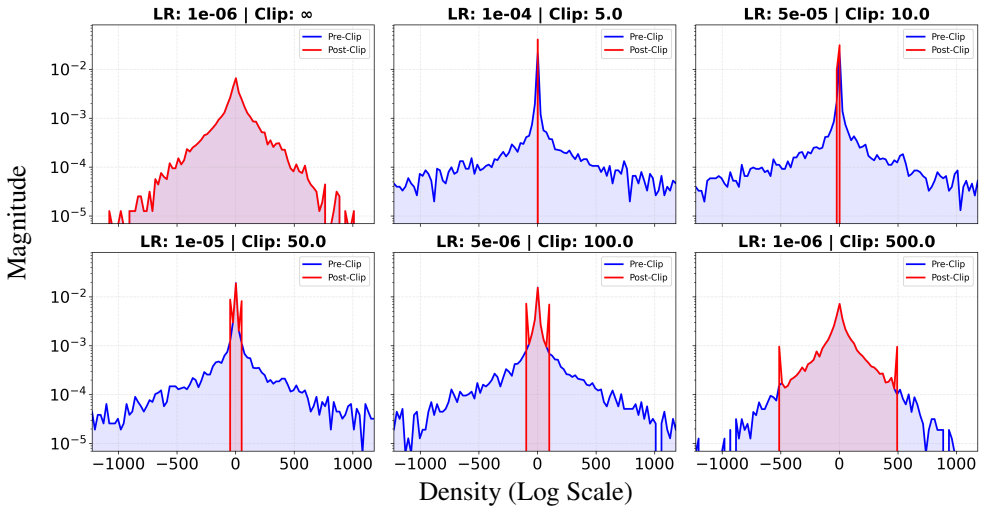

Figure 23: Histogram depicting the spread of the gradient estimator before and after clipping

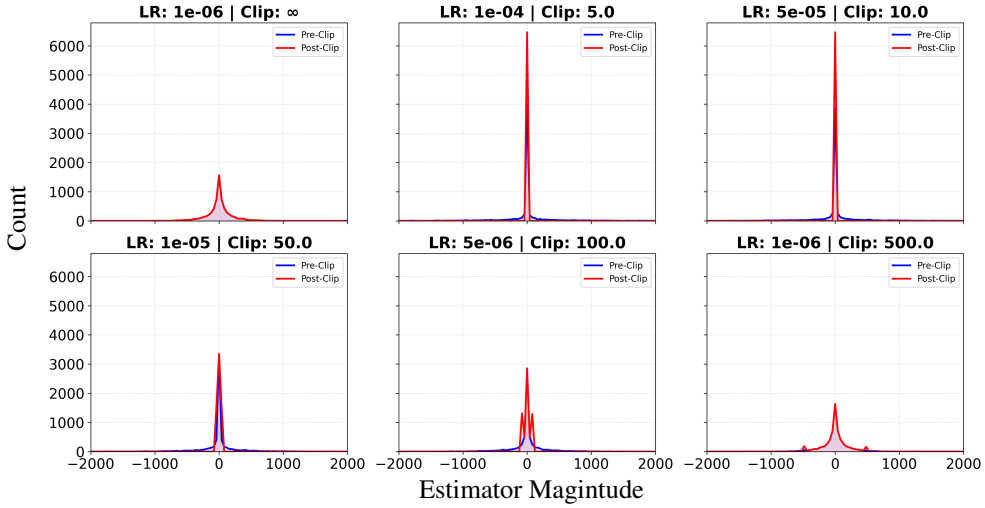

Figure 24: Histogram depicting raw counts of gradient estimator before and after clipping

G.2 Gradient Estimator Spread

To physically observe how the clipping threshold controls the optimization step, we visualize the empirical distribution of the per-sample zeroth-order gradient estimators before and after clipping. Figure 23 and Figure 24 display these distributions across six parameterizations sampled from the optimal diagonal, utilizing both raw counts to show the central mass and log-scaled density to highlight the tail behavior.

Examining the linear-scale count distributions highlights a distinct shift in the central mass of the estimator. While unclipped ZO estimation ($c = \infty$) naturally produces a smoothly decaying, wide-shouldered distribution around zero, the introduction of clipping severely distorts this geometry. Specifically, strict clipping constraints visually stretch the limits of the effective distribution while simultaneously inducing significantly thinner, sharper central peaks. This geometric transformation occurs because the clipping function forcibly redistributes the probability mass from the continuous heavy tails, pushing the values inward to aggregate exactly at the structural limits ($\pm c$). Conse-

quently, the gradient estimator transitions from a broad, continuous spread into a highly concentrated, truncated structure where the update signal is strictly confined to the immediate center and the absolute clipping boundaries.

H ADDITIONAL EXPERIMENTS FROM SECTION 4

H.1 INJECTED LABEL NOISE

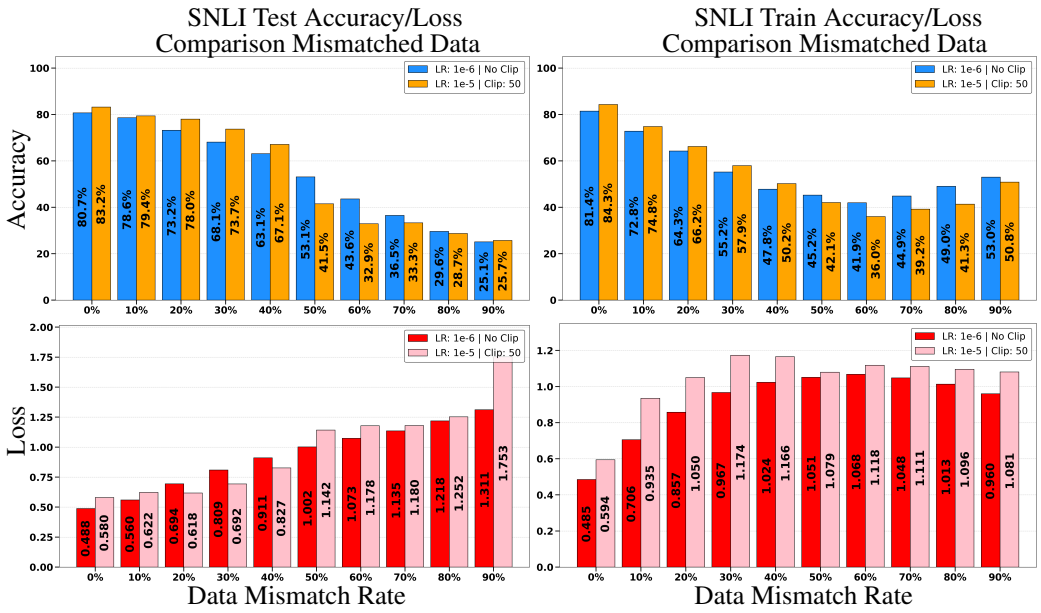

Figure 25: Comparison of test and training accuracy, as well as cross-entropy loss, between clipped ($\eta = 1 \times 10^{-5}, c = 50.0$) and unclipped ($\eta = 1 \times 10^{-6}$) ZO-SGD models across varying rates of injected label mismatch of SNLI training data set. The clipped model demonstrates increased robustness to label noise up to a 50% corruption threshold, maintaining higher test accuracy despite systematically higher training losses[cite: 8].

To further investigate the robustness of clipping, we introduce artificial label noise by randomly mislabeling subsets of the training data at varying corruption levels, ranging from 0% to 90% of the SNLI training set. Figure 25 demonstrates that in the regime where the injected noise fraction is between 0% and 40%, the clipped model ($\eta = 1 \times 10^{-5}, c = 50.0$) is considerably more robust and maintains a higher test accuracy than the unclipped baseline. For example, at a 30% corruption rate, the clipped model achieves a 73.7% test accuracy compared to the unclipped model's 68.1%. Generally, this robustness gap between clipped and unclipped ZO-SGD progressively widens as we inject more noisy labels, up until the corruption rate reaches 50%. Once the injected noise reaches 50% of the training data, the ground-truth labels become fundamentally ambiguous, causing both noise-injected and clean examples to behave similarly during optimization. Conversely, the training loss is consistently higher for clipped training across all corruption levels. This inflated loss may be occurring because the clipping mechanism essentially waters down the effect of poorly fitted samples. By explicitly capping the influence of mislabeled or inherently ambiguous examples, the mechanism seems introduces a systematic bias that prevents the model from minimizing the training loss through the memorization of bad data.

H.2 CLIPPING LEADS TO MISCALIBRATION

To provide a deeper empirical understanding of the predictive uncertainty discussed in Section 4.2, Figures 26, 27, and 28 detail the segmented calibration behaviors of the clipped and unclipped models under injected label noise. By partitioning the reliability diagrams into clean and poisoned subsets, we can isolate exactly how the clipping mechanism affects confidence and accuracy.

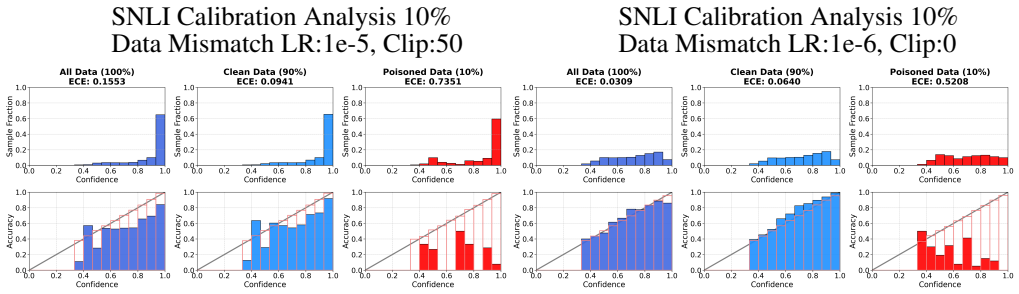

Figure 26: Model Calibration for SNLI fine-tuning under 10%, injected label noise. The plots compare the Expected Calibration Error (ECE) and confidence distributions for the clipped model ($\eta = 1 \times 10^{-5}, c = 50.0$) **(left)** versus the unclipped baseline ($\eta = 1 \times 10^{-6}, c = \infty$) **(right)** across all data, clean data, and poisoned data subsets.

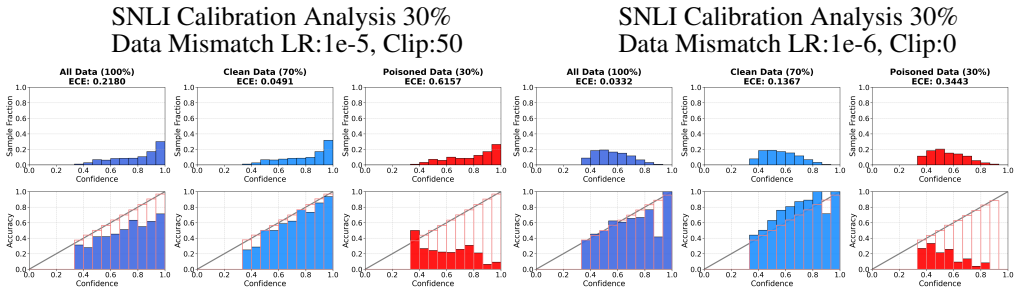

Figure 27: Model Calibration for SNLI fine-tuning under 30%, injected label noise. The plots compare the Expected Calibration Error (ECE) and confidence distributions for the clipped model ($\eta = 1 \times 10^{-5}, c = 50.0$) **(left)** versus the unclipped baseline ($\eta = 1 \times 10^{-6}, c = \infty$) **(right)** across all data, clean data, and poisoned data subsets.

At a 10% data mismatch rate, the clipped model maintains a low Expected Calibration Error (ECE) of 0.0941 on the clean data, but exhibits a massive ECE spike of 0.7351 specifically on the poisoned data. This divergence is even more pronounced at the 30% mismatch rate, where the clipped model achieves a highly calibrated ECE of 0.0491 on clean data, while the poisoned ECE remains heavily inflated at 0.6157. Furthermore, the histograms reveal the clipped model concentrates the vast majority of its prediction mass in the highest confidence bin (> 0.9) across both clean and poisoned samples, creating a severe right-skewed distribution. Because the model correctly resists memorizing the poisoned labels, its empirical accuracy relative to those incorrect labels is exceptionally low. This combination of maximum confidence and minimal empirical accuracy pulls the reliability bars far below the perfect calibration diagonal, forming a stark horizontal flatline near zero accuracy in the reliability diagram. This mathematically explains the massive ECE spikes on the rejected poisoned subset.

Conversely, the unclipped baseline exhibits seemingly better calibration on the poisoned data at the 30% mismatch rate, achieving an ECE of 0.3443 compared to the clipped model's 0.6157. Visually, the unclipped model's sample fraction histogram is much broader and flatter, distributing prediction mass across intermediate confidence bins rather than strictly at the extremes. However, this lower ECE is a symptom of optimization failure rather than generalization success, as the unclipped model achieves better calibration here because it actively memorizes the noisy labels. Its reliability bars form a gradual, upward-sloping staircase that tracks closer to the diagonal, bringing its predictions into closer alignment with the incorrect poisoned targets. Once the data corruption rate reaches the 50% threshold, the ground-truth signals become fundamentally ambiguous, and the robust decision boundary of the clipped model breaks down. At this stage, the clipped model loses its high-confidence predictive structure across all subsets, with its sample fraction mass shifting abruptly away from the right edge and dispersing toward the center, heavily resembling the flatter shape of the unclipped baseline. As a result, the poisoned ECE of the clipped model drops to 0.1482, converging toward the unclipped model's ECE of 0.1204. At this point, the reliability diagrams for both models exhibit similarly flattened and dispersed prediction distributions.

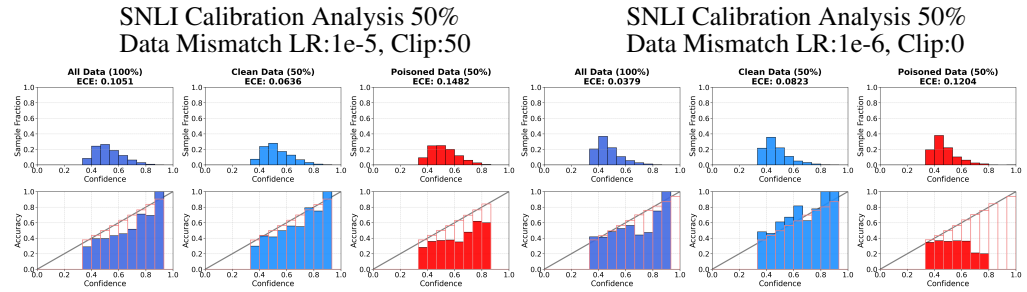

Figure 28: Model Calibration for SNLI fine-tuning under 50%, injected label noise. The plots compare the Expected Calibration Error (ECE) and confidence distributions for the clipped model ($\eta = 1 \times 10^{-5}, c = 50.0$) **(left)** versus the unclipped baseline ($\eta = 1 \times 10^{-6}, c = \infty$) **(right)** across all data, clean data, and poisoned data subsets.

I Additional Edge-of-Stability Experiments

I.1 The Impact of Clipping

We measure the Hessian trace $\mathrm{Tr}(H_t)$ and its upper bound $\mathrm{Tr}(H_t) + 2\lambda_{\max}(H_t)$ to compare random and pretrained initializations, both with and without gradient clipping.

Without Clipping. Without clipping, the randomly initialized CIFAR-10 model exhibits classical EoS: the trace rises and plateaus exactly at the $2/\eta = 4000$ threshold (29, **middle**). Conversely, the pretrained ResNet does not converge to its $2/\eta = 5000$ threshold; its trace starts near 17,500 and smoothly decays to roughly 7,500 (29, **left**). The DPZero-RoBERTa model also shows no EoS; the trace briefly increases to 30,000 before flattening to 0 (29, **right**).

With Clipping. With clipping, the divergence between random and pretrained initializations remains. For the randomly initialized CIFAR-10 model, clipping preserves the EoS phenomenon (30, **middle**). The trace rises to the $2/\eta = 2000$ threshold and oscillates along it, widening the gap between $\mathrm{Tr}(H_t)$ and its upper bound before eventually converging at the EOS threshold. However, this convergence takes $1{,}000{,}000$ steps in our experimental setup, compared with $250{,}000$ steps in the setup without clipping. For the pretrained ResNet, clipping does not induce EoS (30, **right**); the trace starts above 16,000 and decays toward 6,000. For DPZero-RoBERTa, clipping increases training volatility (30, **left**). The trace misses the $2/\eta = 40000$ threshold, exhibiting erratic spikes up to 1.5×10^6 and negative values before converging to 0.

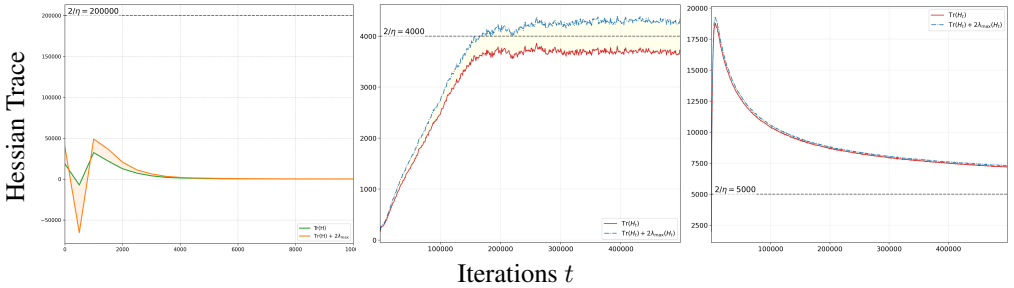

Figure 29: (left) We measure standard spectral features of the Hessian matrix over training iterations in fine-tuning RoBERTa model on SST-2 dataset. Notice that there is no EoS phenomenon. (middle) We reproduce the training from random initialization on a subset of CIFAR-10 to demonstrate EoS phenomenon as reported in (Song et al., 2026). (right) We replicate EoS phenomenon as reported in (Song et al., 2026) but with ImageNet pretrained ResNet weights.

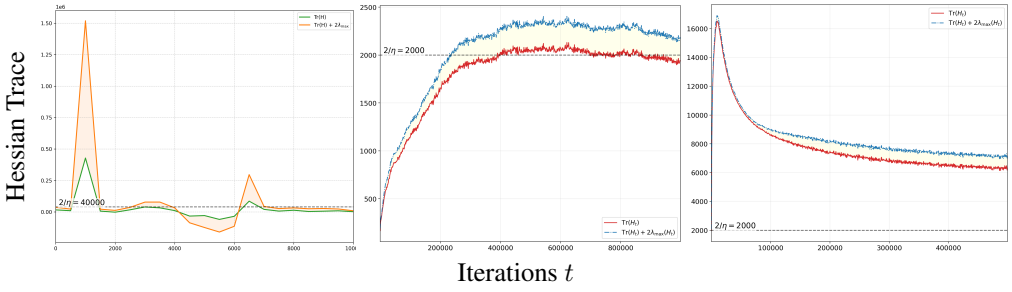

Figure 30: We repeat the experiments in 29 however we add per sample clipping. (left) We measure standard spectral features of the Hessian matrix over training iterations in fine-tuning RoBERTa model on SST-2 dataset. Notice that there is no EoS phenomenon. (middle) We reproduce the training from random initialization on a subset of CIFAR-10 to demonstrate EoS phenomenon as reported in (Song et al., 2026). (right) We replicate EoS phenomenon as reported in (Song et al., 2026) but with pretrained ResNet Weights.

I.2 Lack of EoS in Pretrained Models

Standard EoS features curvature that rises, fluctuates near the stability threshold, and converges along $2/\eta$. In contrast, neither pretrained model converges to this threshold during fine-tuning (29 and 30). We suspect this occurs because pretrained models are already situated in basins that are flat relative to the fine-tuning objective, allowing optimization to proceed without escaping the initial basin (Neyshabur et al., 2021).

I.3 Experimental Limitations of EOS in LLM Fine-Tuning

Finally, we note that the RoBERTa EOS evaluations exhibit greater variability than the other settings. Due to compute limits and other practical constraints, the fine-tuning trajectory was relatively volatile, making it difficult to establish a clear EoS trend. The resulting spiking behavior and absence of a clear EoS are nevertheless informative, suggesting that LLM fine-tuning with ZO-Roberta captures substantially different dynamics in this setting and motivating further investigation under more stable fine-tuning conditions.

J Additional Mismatched/Ambiguous Examples from SNLI training data

Additional Mismatched/Ambiguous Examples from SNLI training data that are overcome due to the robustness of clipping.

Mislabeled example, ID: 174

Premise: Bicyclists waiting at an intersection.
Hypothesis: Bikers stop and wait for traffic at the intersection.
Original SNLI label: Entailment
Model predicted: Contradiction
Clipping stats: 388 clips, **Avg magnitude of gradient estimate:** 563.87

Mislabeled example, ID: 433

Premise: A bearded man in a black t-shirt sits in front of a desk holding a computer.
Hypothesis: A man waiting with his computer.
Original SNLI label: Entailment
Model predicted: Contradiction
Clipping stats: 394 clips, **Avg magnitude of gradient estimate:** 482.56

Mislabeled example, ID: 1147

Premise: A boy in a blue, yellow, and orange shirt holding his arms out from his sides.
Hypothesis: The shirt is blue.
Original SNLI label: Contradiction
Model predicted: Neutral
Clipping stats: 384 clips, **Avg magnitude of gradient estimate:** 432.83

Mislabeled example, ID: 17

Premise: A few people in a restaurant setting, one of them is drinking orange juice.
Hypothesis: The diners are at a restaurant.
Original SNLI Label: Entailment
Model Predicted: Neutral
Clipping Stats: 404 clips, **Avg magnitude of gradient estimate:** 816.07

Ambiguous example, ID: 1427

Premise: Two women, each embracing a little girl, catch up at a small family gathering.
Hypothesis: The women left their children behind.
Original SNLI Label: Contradiction
Model Predicted: Entailment
Clipping Stats: 399 clips, **Avg magnitude of gradient estimate:** 684.28

Ambiguous example, ID: 189

Premise: People waiting at a light on bikes.
Hypothesis: The people are riding their bikes to the gym.
Original SNLI Label: Neutral
Model Predicted: Entailment
Clipping Stats: 392 clips, **Avg magnitude of gradient estimate:** 759.27

Ambiguous example, ID: 1385

Premise: Two people standing in front of a large statue of a woman, other statues and busts visible in the background.
Hypothesis: Two people standing standing near a large statue, with other states nearby.
Original SNLI label: Entailment
Model predicted: Contradiction
Clipping stats: 389 clips, **Avg magnitude of gradient estimate:** 544.60

Ambiguous example, ID: 1266

Premise: Girl is blowing to a butterfly.
Hypothesis: The girl blows a butterfly.
Original SNLI label: Entailment
Model predicted: Contradiction
Clipping stats: 366 clips, **Avg magnitude of gradient estimate:** 524.82