Compress then Serve: Serving Thousands of LoRA Adapters with Little Overhead

Rickard Brüel Gabrielsson ¹ Jiacheng Zhu ¹ Onkar Bhardwaj ² Leshem Choshen ¹² Kristjan Greenewald ² Mikhail Yurochkin ² Justin Solomon ¹

Abstract

Fine-tuning large language models (LLMs) with low-rank adaptations (LoRAs) has become common practice, often yielding numerous copies of the same LLM differing only in their LoRA updates. This paradigm presents challenges for systems that serve real-time responses to queries that each involve a different LoRA. Prior works optimize the design of such systems but still require continuous loading and offloading of Lo-RAs, as it is infeasible to store thousands of Lo-RAs in GPU memory. To mitigate this issue, we investigate the efficacy of compression when serving LoRAs. We propose a method for the joint compression of LoRAs into a shared basis paired with LoRA-specific scaling matrices. We extend our algorithm to learn clusters of LoRAs that are amenable to joint compression, allowing it to scale gracefully to large LoRA collections. Our experiments with up to 1000 LoRAs demonstrate that compressed LoRAs preserve performance while offering major throughput gains in realistic serving scenarios with over a thousand LoRAs, maintaining 80% of the throughput of serving a single LoRA.

1. Introduction

The myriad uses for foundation models (FMs) have led to a proliferation of specialized models, each fine-tuned to perform a downstream task. To avoid fine-tuning foundation models with billions of parameters, parameter-efficient fine-tuning (PEFT) algorithms were proposed. An especially successful PEFT method is low-rank adaptation (LoRA) (Hu et al., 2021), which learns low-rank additive changes to neural network matrices. Because of the

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

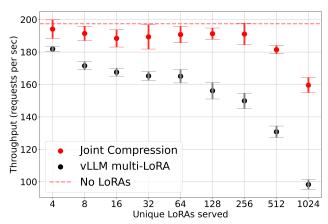


Figure 1: Throughput gains when serving 1000s of compressed LoRAs with vLLM.

low-rank parameterization, these matrices (called adapter weights) contain orders of magnitude fewer parameters than the base model. Still, LoRA can achieve performance on par with full fine-tuning (Hu et al., 2021).

LoRA's popularity has triggered a growing need to serve large collections of LoRA adapters at scale. Proprietary and open-source LLM providers offer fine-tuning services (OpenAI, 2024; TogetherAI, 2024; Predibase, 2024) with user bases likely in the thousands or even hundreds of thousands. As each user wants to use their own finetuned version of the LLM, serving a dedicated fine-tuned LLM per user becomes infeasible. To this end, S-LoRA (Sheng et al., 2023) proposes a system where only the base LLM is placed on an inference server and individual LoRA adapters are switched as needed at inference time. S-LoRA optimizes the system's inner workings via custom CUDA kernels and memory management to increase throughput when serving multiple LoRAs. Multi-LoRA system design has also been adopted in vLLM (Kwon et al., 2023), a stateof-the-art LLM serving engine. Despite optimized system designs, serving LoRAs still has a fundamental limitation: when the number of adapters is large, they need to be constantly loaded and offloaded from GPU memory to accommodate incoming requests, degrading throughput.

The problem of accommodating multiple LoRA adapters is also apparent when placing LLMs on edge devices,

where smaller LLMs are fine-tuned for various tasks, and the adapters are swapped depending on the task at hand (Gunter et al., 2024). In this setting, the number of adapters is smaller, e.g., a few dozen (Gunter et al., 2024), but the memory constraints are also more stringent.

In this work, we consider the problem of compressing a collection of LoRAs. We have two key objectives: (1) preserving the performance of the original LoRAs and (2) improving the throughput of serving many LoRAs. We formulate LoRA compression as a reconstruction problem, where the goal is to approximate the original adapters via collections of matrices of a smaller size. We compress Lo-RAs jointly by finding a shared basis and LoRA-specific scaling matrices and propose a joint diagonalization-based algorithm (JD). To improve reconstruction error for large numbers of LoRAs while keeping the number of parameters in check, we propose a clustering approach where each cluster is compressed independently using the joint diagonalization algorithm. Our clustering algorithm is based on alternating between optimizing the cluster assignments and the per-cluster reconstruction error.

Figure 1 showcases the benefits of joint compression. When serving up to 64 unique LoRAs, we use JD without clustering and for 128 or more, we pick the number of clusters to match the performance of compressed and original LoRAs. In each case, the GPU memory footprint of the compressed and original LoRAs is matched for a fair comparison to vLLM's multi-LoRA inference engine. When serving over 1000 LoRAs, compression increases throughput 1.6× and maintains 80% of the throughput of serving the base LLM (or a single LoRA merged into the LLM). §6 presents detailed results.

We summarize our main contributions below:

- We formulate the problem of compressing a collection of LoRAs and propose a joint compression scheme based on joint diagonalization.
- For large numbers of LoRAs, we scale joint compression by proposing a clustering algorithm where each cluster is jointly compressed to minimize reconstruction error.
- We establish theoretical guarantees for the reconstruction error of our compression formulation and relate reconstruction loss to performance empirically.
- We train a collection of more than 1000 high-quality LoRAs for Mistral-7B-Instruct-v0.2 (Jiang et al., 2023a) on 1000 natural instruction tasks (Wang et al., 2022) and demonstrate that our compression techniques preserve the performance of the original LoRAs. We will release over a 1000 LoRAs to facilitate future work as well as the code for our method.
- We incorporate LoRA compression into a state-of-theart LLM serving system and demonstrate that it is possible to serve over 1000 LoRAs across thousands of asyn-

chronous requests with throughput comparable to serving a *single LoRA*.

2. Related Work

Parameter-efficient fine-tuning (PEFT) has become prevalent for updating foundation models thanks to the need for efficiency in training and communication (Lialin et al., 2023). Many PEFT methods have been proposed, e.g. (Houlsby et al., 2019; Liu et al., 2022b) and LoRA (Hu et al., 2021) became the standard, partially due to the ease of switching between LoRAs in inference time.

Several works improve LoRA (Liu et al., 2024; Wang et al., 2024), sometimes with algebraic methods like SVD (Meng et al., 2024; Zhang et al., 2023; Jiang et al., 2023b) or by leveraging its statistical properties (Zhu et al., 2024; Zeng & Lee, 2024). Relatively few, however, accelerate inference times. S-LoRA (Sheng et al., 2023) provides an efficient means of switching between LoRAs. Wen & Chaudhuri (2024) adapt training to reduce batch multiplications, accelerating inference. Our method achieves a similar outcome (see Appendix D) without changing the LoRA formulation or requiring that LoRAs be trained in a dedicated way; future improvements to LoRA will also benefit from this aspect of our work (e.g., Meng et al. (2024)).

Punica (Chen et al., 2023) introduces Segmented Gather Matrix-Vector Multiplication (SGMV) to optimize multi-LoRA serving by parallelizing feature-weight multiplications in batches and grouping requests that use the same LoRA. Our approach, by contrast, reduces parameters as a means to serve multiple LoRAs efficiently, providing an orthogonal strategy that can be seamlessly integrated with Punica's methods to enhance performance. In our vLLM experiments, we leveraged the Punica kernel for multi-LoRA implementation, demonstrating the application of our method in conjunction with Punica's optimizations.

Other research proposes alternative PEFT methods that can be more parameter-efficient than LoRA. For example, VeRA (Kopiczko et al., 2024) fine-tunes LLMs by sharing global static parameters while learning local scaling variables; (IA)³ (Liu et al., 2022a) also reduces adapter parameter counts. However, none of these approaches has been as extensively tested or widely-adopted as LoRA. As a result, work that builds on LoRA enjoys a practical advantage due to its broad acceptance in practice.

There are many efforts to compress models (Cheng et al., 2017; Gholami et al., 2022; Sharma et al., 2024; Li et al., 2018). Predominantly, pruning and sparsification methods delete weights (Yadav et al., 2023a), and quantization methods reduce the weights' precision (Dettmers et al., 2024). Some works compress weights to reduce model size but typically require decompression and hence do not save

GPU memory (Hershcovitch et al., 2024). Similarly to our work, a few note increased performance and generalization after compression (Yadav et al., 2023a; Nadjahi et al., 2023; Hershcovitch et al., 2024; Sharma et al., 2024).

Our work also relates to model merging (Choshen et al., 2022; Wortsman et al., 2022; Matena & Raffel, 2021) and mixtures of experts (Muqeeth et al., 2024; Yadav et al., 2024). These methods reuse models trained by others (Choshen et al., 2023; Raffel, 2023), serving them together as one compressed model. Despite this similarity, these methods create a single general model that acts on any input, while ours yields more performant per-task solutions.

3. Rank-Based LoRA Compression

LoRA updates are parameterized by pairs of matrices A, B, whose product BA updates the fixed weight matrices $W_0 \in \mathbb{R}^{d_B \times d_A}$ of a neural network foundation model. Given an input x to a layer, the output of the LoRA-updated model at this layer is $(W_0 + BA)x$.

In formulating our compression algorithms, we consider a collection of given LoRA adapters $\{(A_i, B_i)\}_{i=1}^n$ that we would like to serve. We let r_i refer to the rank of the LoRA adapter-pair (A_i, B_i) , i.e., $B_i \in \mathbb{R}^{d_B \times r_i}$, $A_i \in \mathbb{R}^{r_i \times d_A}$.

While our compression technique has access only to a collection of $\{(A_i,B_i)\}_{i=1}^n$ pairs, in our experiments we will assess the efficacy of compression by comparing how the compressed matrices perform relative to uncompressed LoRAs on typical data. For this reason, although in this section we optimize a Frobenius norm reconstruction error relative to the product B_iA_i , this is a proxy for the nonlinear and complex way that compression errors in the adapters impact transformer performance. Our experiments will thus focus on the performance of the compressed LoRAs against the uncompressed versions on real data in §6.

Our compression methods significantly reduce the overall number of parameters. Reducing parameters theoretically accelerates storage and serving of a collection of LoRAs. This reduction, however, alters the computational dynamics during inference, so parameter reduction alone does not immediately imply faster throughput. In light of the complexities of GPU optimization, we experimentally assess throughput under realistic conditions in §6.4.

3.1. Joint Diagonalization

To scale to many LoRAs, the compressed number of parameters should not scale linearly with n. Hence compressing each LoRA individually (e.g., via SVD as in our experimental baselines) is inherently limited. To address this, we suggest a Joint Diagonalization (JD) method, which optimizes a shared basis onto which we can project the set of

n LoRAs. This allows structure to be shared, implicitly grouping and/or merging the collection of LoRAs.

In this model, each LoRA product B_iA_i is factorized into the form $U\Sigma_iV$, where U and V are shared across all LoRAs and Σ_i is specific to each LoRA. In this formulation, every Σ_i shares the same rank r. This allows U and V to be pre-loaded onto the GPU, with Σ_i loaded when necessary for each batch. The matrices Σ_i can be either diagonal or small square matrices, thus significantly reducing the number of LoRA-specific parameters and accelerating multi-LoRA serving.

Objective function. Motivated by the relationship of singular value decomposition to minimizing the Frobenius norm of the reconstruction error, we also propose to minimize the Frobenius norm of the adapter matrix approximation error. Specifically, we use the following objective:

$$\min_{\{\Sigma_i\}_{i=1}^n, U, V} \sum_{i=1}^n \|B_i A_i - U \Sigma_i V^\top\|_{\text{Fro}}^2.$$
 (1)

Note this problem is *not* solved by a single matrix SVD, since U and V are shared among all terms but the Σ_i 's are not. Using the Frobenius norm has the added benefit of making the objective convex in each argument separately, suggesting the possibility of efficient optimization. This objective function is underdetermined, however, so we consider two constrained regimes below.

Full Σ_i approximation. The first method we call JD-Full. Without loss of generality, U and V can be constrained to be orthogonal, so long as Σ_i remains an unconstrained full matrix. JD-Full adopts this restriction to make the optimization better posed, but note it does not restrict the expressiveness of the objective equation 1. This setting yields the following optimization problem:

$$JD-Full_r(\{B_i A_i\}_{i=1}^n) =$$

$$\underset{\{\Sigma_i\}_{i=1}^n}{\operatorname{argmin}} \sum_{i=1}^n \|B_i A_i - U \Sigma_i V^\top\|_{\operatorname{Fro}}^2$$

$$U^\top U = V^\top V = I_r$$

$$(JD-Full) \qquad (2)$$

An efficient alternating algorithm to optimize this objective function can be found in Appendix A.

Diagonal Σ_i **approximation.** As an alternative, we can leave U, V unconstrained (other than to have r columns) and instead constrain the matrices Σ_i to be diagonal (but not necessarily positive). This formulation yields the following optimization problem:

$$\operatorname{JD-Diag}_r(\{B_i A_i\}_{i=1}^n) = \underset{\{\Sigma_i\}_{i=1}^n, U, V}{\operatorname{argmin}} \sum_{i=1}^n \|B_i A_i - U \operatorname{diag}(\Sigma_i) V^\top\|_{\operatorname{Fro}}^2$$

$$(\operatorname{JD-Diag}) \quad (3)$$

Appendix A provides an efficient alternating least squares algorithm for this objective. This diagonal version has per-LoRA parameter savings when compared to JD-Full, since the diagonal Σ_i only needs r parameters instead of r^2 .

3.2. Clustering

As the number of LoRAs n grows and becomes more diverse, the rank r needed for Joint Diagonalization to achieve good performance will tend to increase. This increases the size of each Σ_i that needs to be stored, especially for JD-Full which will require $O(nr^2)$ storage for these matrices. If the necessary r grows proportionally to n, then this storage will eventually become the bottleneck.

To resolve this limitation with very large n, we propose to group the n LoRAs into k clusters C_j . Each cluster is given its own rank r for JD compression, and the clusters are chosen such that the overall reconstruction error is minimized. Specifically, the overall objective is

$$\min_{\{\{C_j\},U_j,V_j\},\{\Sigma_i\}} \sum_{i} \sum_{i \in C_i} ||B_i A_i - U_j \Sigma_i V_j||_F^2,$$

optimized by alternating between cluster assignments and the JD of each cluster; Appendix A.3 provides details. Typically, the goal with large n is to have k grow with n as r becomes fixed. Comparing k rank-r JD-Full clusters to a rank-kr JD-Full single cluster compression, the clustered approach requires $O(dkr + nr^2)$ parameters, while the single-cluster approach requires $O(dkr + nk^2r^2)$ parameters due to the increased sizes of the Σ_i s. While these two approaches have the same rank, they may have different reconstruction abilities. Empirically, we find that multiple clusters significantly aid performance for $n \ge 100$.

4. Theoretical Analysis

In this section, we seek to better understand the role of the joint diagonalization method in §3.1 and how it motivates the clustering approach. We focus on the full- Σ_i case with orthogonal U, V matrices. Note that, for the same r, the r-JD-Diag has at least as large reconstruction error as r-JD-Full since it imposes an additional constraint on the Σ_i .

Firstly, note that perfect reconstruction can be achieved if and only if r is large enough, since there exist U, V such that all the B_i , A_i are in the spans of U, V resp. if and only if $r \geq \tilde{r}$:

Proposition 1. Suppose $rank(B_iA_i) = r_i$ for all i, and let

$$\tilde{r} = \max \left\{ \operatorname{rank}([A_1, \dots, A_n]), \operatorname{rank}([B_1^\top \dots, B_n^\top]) \right\}.$$

Note $\max_i r_i \leq \tilde{r} \leq \sum_{i=1}^n r_i$. Then JD-Full (equation 2) with $r = \tilde{r}$ compresses losslessly (perfect reconstruction), while $r < \tilde{r}$ will give nonzero reconstruction error.

Due to training noise, \tilde{r} will equal $\sum_{i=1}^{n} r_i$ almost always. This implies that in most realistic settings, the joint diagonalization approach is a lossy reconstruction.

This reconstruction loss can be significant, as the following theorem shows (proved in Appendix B):

Theorem 1. Consider n LoRAs $\{A_i, B_i\}_{i=1}^n$ with $r, n \leq d^2$, and form the matrix $L = [\operatorname{vec}(B_1A_1) \cdots \operatorname{vec}(B_nA_n)]$. Let σ_j be the singular values of L, sorted from largest to smallest, and let $\bar{\sigma}_j$ be the singular values of $\sum_{i=1}^n B_i A_i$. Then, using JD-Full (equation 2),

$$\sum_{j=1}^{r} \bar{\sigma}_{j}^{2} \leq \sum_{i=1}^{n} \|\Sigma_{i}\|_{\operatorname{Fro}}^{2} = \sum_{i=1}^{n} \|U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} \leq \sum_{j=1}^{\min(r^{2}, n)} \sigma_{j}^{2},$$

implying the sum of squared Frobenius norms of the reconstructed LoRAs satisfies

$$\begin{split} & \frac{\sum_{i=1}^{n} \|U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2}}{\sum_{i=1}^{n} \|B_{i}A_{i}\|_{\operatorname{Fro}}^{2}} \leq \frac{\sum_{j=1}^{\min(r^{2},n)} \sigma_{j}^{2}}{\sum_{j=1}^{n} \sigma_{j}^{2}} \leq 1, \text{ and} \\ & \frac{\sum_{i=1}^{n} \|U\Sigma_{i}V^{\top} - B_{i}A_{i}\|_{\operatorname{Fro}}^{2}}{\sum_{i=1}^{n} \|B_{i}A_{i}\|_{\operatorname{Fro}}^{2}} \geq 1 - \frac{\sum_{j=1}^{\min(r^{2},n)} \sigma_{j}^{2}}{\sum_{j=1}^{n} \sigma_{j}^{2}}. \end{split}$$

In other words, reconstruction error is unavoidable if L's singular values are not concentrated in the top r^2 entries.

Remark 1 (Lower bound and merging). The lower bound $\sum_{j=1}^{r} \bar{\sigma}_{j}^{2}$ could be achieved by setting all the Σ_{i} equal, i.e., using a fully merged model instead of only merging the subspaces U, V and allowing Σ_{i} to vary with i.

Remark 2 (Upper bound and grouping). The upper bound is smallest when the LoRAs are relatively clustered, i.e., when groups of vectors $vec(B_iA_i)$ are similar. This situation raises the magnitude of the largest singular values of L, raising the upper bound in the proposition. As the LoRAs are $d \times d$ matrices that can be thought of as points in \mathbb{R}^{d^2} , for typical values of d well into the hundreds, it is likely that unrelated LoRAs will be unclustered, i.e., they will have relatively low inner products with each other.

For orthogonal LoRAs, the singular values of L are the norms of the LoRAs, suggesting the following corollary:¹

¹A result for isotropic Gaussian LoRAs could be obtained via the quantiles of the Marchenko-Pastur Law.

Corollary 1. Suppose (e.g., due to normalization) that the inputs to the joint diagonalization algorithm all have unit Frobenius norm, i.e., $||B_iA_i||_{Fro} = 1$. Moreover, assume that the LoRAs are all orthogonal in the sense $\operatorname{tr}((B_iA_i)(B_jA_j)^\top) = 0$ for $i \neq j$. Then, using the JD-Full method equation 2, we have $1 \leq \sum_{i=1}^n ||\Sigma_i||_{Fro}^2 \leq \min(r^2, n)$, implying that the sum of squared Frobenius norms of the reconstructed LoRAs satisfies

$$1 - \frac{1}{n} \ge \frac{\sum_{i=1}^{n} \|U\Sigma_{i}V^{\top} - B_{i}A_{i}\|_{Fro}^{2}}{\sum_{i=1}^{n} \|B_{i}A_{i}\|_{Fro}^{2}} \le 1 - \min\left(\frac{r^{2}}{n}, 1\right).$$

This implies that for the common setting where $r^2 \ll n$, the reconstructed LoRAs will be significantly smaller than the original LoRAs, with significant reconstruction error.

Our analysis illustrates the tradeoffs of joint diagonalization. If the LoRAs are similar or well-clustered, reconstruction error will be low. On the other hand, if the LoRAs are random and orthogonal, reconstruction error will be high.

Since the loss space of transformers is highly complex, increasing reconstruction error does not necessarily degrade LLM performance. Interestingly, Figure 3 below shows that while large reconstruction error rapidly decreases performance, moderate (but still relatively large, at around 60%) reconstruction error does not damage performance and may even slightly outperform the zero-error setting. At the same reconstruction error, clustering outperforms non-clustering. This motivates our focus on minimizing reconstruction error, while also suggesting that our approach achieves something deeper than compression. Specifically, joint diagonalization finds subspaces that are shared among many LoRAs when r is large and *merges* subspaces when r is small. When r is particularly small, this tendency towards averaging all or some of the LoRAs connects to merging LoRAs, whose empirical success (Shah et al., 2023; Huang et al., 2024) could explain the procedure's success despite the nonlinearity of transformers.

Appendix H.11 explores this idea further, comparing reconstruction of real-world LoRAs to reconstruction of randomly sampled LoRAs. The reconstruction error is generally large, but significantly lower than the reconstruction error for random noise, indicating that a major shared component between the LoRAs is successfully retained.

That said, as the number of LoRAs grows, the shared component may not be significant enough to maintain sufficiently low reconstruction error with low rank r. This motivates the introduction of *clustering* in §3.2, since clustering seeks to find groups of LoRAs that are similar and better compressible by joint diagonalization. In particular, if the number of clusters k grows with n, the reconstruction error may no longer degrade with n even when r is fixed.

In the extreme case where k = n, each LoRA is com-

pressed independently. By the Eckart-Young Theorem, JD applied to a single LoRA reduces to an SVD, replacing each rank- r_i LoRA adapter B_iA_i with a reduced rank- r_i approximation, where typically $r < \frac{1}{n} \sum_{i=1}^n r_i$:

$$SVD_r(B_iA_i) = U_i\Sigma_iV_i^{\top}, \quad \forall i = 1, \dots, n.$$
 (4)

As $\Sigma_i V_i^{\top}$ can be saved as a single matrix, this approach has $rn(d_A+d_B)$ parameters. We refer to this k=n method as r-SVD and find that it underperforms our other methods while slightly outperforming the baseline uncompressed LoRAs. This result parallels Jiang et al. (2023b)'s observation that lowering LoRA ranks is beneficial for multi-task learning and model merging.

5. Training & Performance Evaluation

5.1. Training

We trained LoRA adapters on 1000 natural instruction tasks (Wang et al., 2022) using Mistral-7B-Instruct-v0.2 (Jiang et al., 2023a) as the base. We set all LoRA adapter ranks to 16 (i.e., $\forall i, r_i = 16$), except for those in our ablation study (Appendix H.1), where we vary the LoRA rank.

We selected 10 diverse tasks (Table 2 in Appendix C) manually for consistent evaluation across experiments and randomly sampled an additional 990 tasks, resulting in a total of 1000 tasks (Table 3). The tasks went through a robust reviewing protocol to ensure high quality and diversity. Each task data was divided into training, validation, and test sets.

Hyperparameters, such as early stopping, were tuned using the validation sets. Table 1, Appendix C shows that on the test sets, LoRA consistently outperformed the base model in terms of Rouge scores and loss metrics.

5.2. Evaluation

We evaluated multiple metrics for the natural instruction tasks, including cross-entropy loss, Rouge-1, Rouge-L (Lin, 2004), exact match, and *agreement* between uncompressed and compressed LoRA. Here, *agreement* measures the exact match in task-generations between the uncompressed LoRA model and the compressed LoRA model, rather than comparing to ground truth data. While detailed results and discussions for all metrics are provided in Appendix H, our primary focus in the main text is on Rouge-L. We find that all metrics correlate, but Rouge-L correlates most strongly with downstream utility. This finding aligns with prior work (Wang et al., 2022), which demonstrates that Rouge-L correlates well with classification accuracy.

While cross-entropy is used for optimization during training, identical generation outputs across models can yield different cross-entropy losses. Exact match is too rigid and

does not account for the variability in task responses. Similarly, agreement does not capture the inexactness associated with most of our tasks, nor does it account for the performance gains or losses of the compressed LoRAs. Arguably, practitioners are primarily concerned with task performance in the settings for which the LoRA was designed, rather than exact generational agreement between models.

Joint diagonalization optimizes reconstruction error measured by the Frobenius norm, bounded by our theoretical analysis in §4. We empirically study the relation between the reconstruction error and downstream Rouge-L performance in Section 6.2.

Instead of listing absolute performance, we compute the performance difference between the base model and the LoRA model for each task via the ratio

 $Performance \ relative \ to \ LoRA := \frac{method\text{-performance}}{LoRA\text{-performance}}$

for the method in question, highlighting relative improvement wrt the uncompressed LoRAs.

6. Experiments

6.1. Task Performance

For each method, we vary the number n of compressed LoRAs and the compression rank r. We run each experiment three times with different random seeds and report the mean and standard deviation. See Table 7 for results evaluated on the same ten manually-selected tasks (Table 2) across settings. Every compressed collection of LoRAs contains these 10 tasks (i.e., in-distribution tasks), and each collection contains the smaller collections as subsets.

We normalize each LoRA adapter to have a Frobenius norm of one prior to running joint diagonalization. This normalization enhances performance and reduces the variance in reconstruction error. We restore the original norms of the LoRA adapters before reconstruction and testing.

Figure 2 illustrates the Rouge-L scores of the compressed LoRAs divided by the Rouge-L scores of the uncompressed LoRAs. JD variants often increase generalization and outperform the original LoRA. Notably, our JD methods approach the compression efficacy of a single LoRA, and with clustering, this aggressive reduction in size also maintains performance in larger collections. Appendix H includes tables of additional relative and absolute metrics.

For efficiency, we limited the JD methods to ten iterations instead of full convergence. While the alternating algorithm quickly reaches an approximate minimizer, squeezing out the last few digits of precision takes many more iterations with limited to no performance gain. Appendix H.12 also evaluates an alternative iterative algorithm that con-

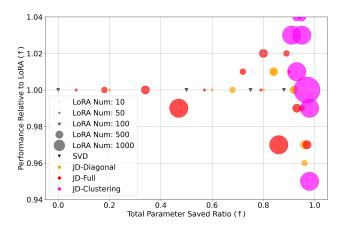


Figure 2: Performance after compression. We compare the performance of compressed LoRAs relative to uncompressed ones, with higher values on both axes reflecting better performance. The Total Parameter Saved Ratio depicts the number of parameters saved for a system with a large number n of different LoRAs. It is computed as: $r_{total} := 1 - \frac{\text{num. parameters after compression}}{\text{num. parameters before compression}}.$

verges more rapidly once U, V are close to a minimizer, with minimal performance differences.

6.2. Performance and Reconstruction Error

Figure 3 relates reconstruction error and performance. The y-axis measures the mean performance improvement of Rouge-L relative to uncompressed LoRA, and the x-axis quantifies the mean relative reconstruction error between the compressed reconstruction of the product BA and the original product BA. Although performance and reconstruction error relate non-linearly, we see a decreasing, somewhat exponential trend. Notably, minimizing reconstruction error does not yield optimal performance, indicating that mild lossy reconstruction may enhance generalization. Interestingly, under the clustering approach, compared to non-clustering, even more aggressive lossy reconstruction can outperform less lossy reconstruction, suggesting that reconstruction error is even less critical for performance in the clustering scenario.

To select hyperparameters (compression rank and number of clusters) for the clustering experiments, we first assessed reconstruction error on a single LoRA module over a range of settings (see Appendix G). These preliminary experiments enabled efficient selection of cluster counts and rank values for compressing all LoRA modules.

6.3. Benefits of Compression

Compressing LoRAs reduces their parameter counts, thus lowering their overall memory footprint. While this offers

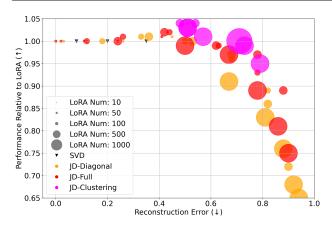


Figure 3: Reconstruction error vs. performance.

many benefits, in both training and inference scenarios parameters are often transferred between different memory hierarchies (e.g., from CPU to GPU), and these transfers usually scale linearly with the amount of data moved. At the same time, compressed LoRAs alter the forward-pass formulation (unless merged with the base model weights). As shown in Figure 5 in the Appendix E, although compression greatly reduces memory usage and transfer time, it does not affect the forward-pass latency. Of the various ways to leverage these improvements, this work focuses on optimizing inference for multiple LoRAs using vLLM.

6.4. Throughput of Serving Compressed LoRAs

The previous sections demonstrate how to select an appropriate joint compression setting guided by the reconstruction error, such that the performance of the original LoRAs is preserved. Naturally, the rank and/or the number of clusters for the compression needs to increase as we compress larger LoRA collections to match LoRA performance.

Figure 4 studies how throughput with various compression settings compares to the vLLM multi-LoRA throughput with the matched GPU memory footprint. Specifically, for each number of unique LoRAs served and each compression setting, we compute the corresponding number of Lo-RAs to be placed on the GPU during serving and report the ratio of the two throughputs. For example, when serving 64 unique LoRAs and using rank 64 JD-Full compression, we report the ratio of throughputs of rank 64 JD-Full and vLLM multi-LoRA with 6 LoRAs allowed on the GPU at a time (see Appendix F for details). As the number of unique LoRAs increases, vLLM multi-LoRA throughput degrades as it needs to schedule the requests and load and offload the adapters. We note that vLLM multi-LoRA already employs advanced optimizations, such as efficient scheduling and non-blocking CPU-GPU communication when swapping LoRAs as well as techniques introduced in S-LoRA (Sheng et al., 2023; Kwon et al., 2023), but system opti-

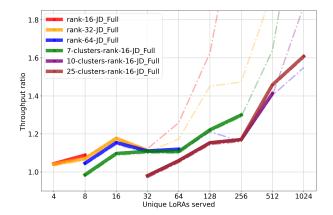


Figure 4: Throughput ratio when serving varying numbers of LoRAs with vLLM. Highlighted settings preserve at least 99% of the uncompressed LoRA performance.

mization alone is insufficient to mitigate throughput degradation when serving many LoRAs.

Figure 4 shows that across LoRA collection sizes our compression techniques improve the throughput of vLLM multi-LoRA. Additionally, we highlight regions for each compression setting where compression is sufficiently moderate to achieve 99%+ of LoRA performance, according to the results in §6.2. Compression with a larger rank or too many clusters does not improve baseline throughput when serving a smaller number of LoRAs and should not be used in such cases. For example, rank 16 JD-Full improves baseline throughput with 4 and 8 LoRAs, but will underperform with more LoRAs, while 25 clusters rank 15 JD-Full does not improve throughput with 32 or fewer LoRAs, but when serving 1000+ LoRAs it improves the throughput significantly while maintaining the performance. Overall, an appropriate joint compression setting improves vLLM multi-LoRA throughput and preserves performance for LoRA collections of any size between 4 and 1024, as in Figure 1. Appendix F provides compression settings for each collection size.

vLLM extensively uses custom CUDA kernels. To accommodate our compression techniques, we minimally adjusted the vLLM code to generate additional kernels needed by the compressed LoRAs and used the Punica (Chen et al., 2023) kernel to further accelerate matrix multiplication. Pseudocode is given in §F.4 to show how we use the batch multiplication kernel. There likely is room for improvement to optimize the newly added kernels.

Additional details. In this experiment, we considered a varying number of rank-16 LoRAs, using a dataset of Shakespeare sonnets as inputs² arriving *asynchronously*.

²https://www.kaggle.com/
datasets/shivamshinde123/
william-shakespeares-sonnet/data

We measured throughput, i.e., the number of requests served per second when generating ten tokens per request. The base was Mistral 7B Instruct; we simulated random LoRAs and assigned inputs to LoRAs at random. Experiments were conducted on H100 80GB GPU capped at 40% memory consumption to reflect situations where a service provider might want to serve many LoRAs from cheaper hardware with lower memory than higher-end GPUs. This setting applies to the scenario where the LLM is large compared to the size of GPU and yet a provider may want to serve many LoRAs efficiently using one device.

6.5. Recommendations

JD-Full is generally preferred over JD-Diag, although for smaller numbers of LoRAs (less than 100), the performance difference is negligible. While JD-Full alone is effective up to 100 LoRAs, incorporating clustering at scales of 500-1000 LoRAs significantly enhances performance.

We recommend the following procedure for hyperparameter selection. For ≤ 100 LoRAs, JD-Full can be used without substantial degradation, using a rank \approx (number of LoRAs/2) + 7. Beyond 100 LoRAs, clustering becomes increasingly critical. A robust method for any number of LoRAs up to 1000 uses JD-Full with clustering. Specifically, select a LoRA module from the middle of the network, apply a compression rank of 16, and experiment with an exponentially increasing number of clusters. Compute the reconstruction error for each setting on this module across all LoRAs—a computationally efficient process. Choose the minimal number of clusters that achieves a reconstruction loss below 0.6, and then use these settings across LoRA modules. Figure 6 in the Appendix illustrates this procedure applied to 500 LoRAs.

Tuning hyperparameters as discussed above using reconstruction loss as a validation metric is convenient since it can be done efficiently on CPU without expensive LLM evaluation. As our experiments demonstrate, compression settings that achieve below 0.6 reconstruction loss reliably preserve 99% or more of the LoRA performance, sometimes even outperforming the original LoRAs.

For inference, this procedure is executed as a preprocessing step before deploying our inference server. As new Lo-RAs are submitted, they are initially served uncompressed. A background CPU job can periodically re-run the compression algorithm and update the served LoRA parameters with the compressed versions.

7. Discussion

This study introduces approaches to LoRA compression, addressing significant challenges emerging as customization of foundation models such as LLMs and diffusion

models becomes increasingly popular. Our contributions include theoretical formulations, empirical validation, and practical implementations that enhance the understanding and application of LLMs in scalable environments.

Our findings have several implications. Our theoretical bounds on reconstruction error not only increase confidence in the use of compressed models but also lay a groundwork for future explorations. Demonstrating that our compression techniques can preserve up to 100% of the original LoRAs' performance highlights the effectiveness of our methods. Furthermore, integrating LoRA compression into state-of-the-art LLM serving systems demonstrates potential for resource optimization, with throughput for thousands of LoRAs nearing that of a single LoRA.

Our promising results suggest several future research directions. First, further compression may be possible via quantization, since joint-diagonalization and quantization are independent compression strategies. Second, when scaling to hundreds of thousands of LoRAs, joint compression, while effective, will be insufficient to fit all LoRAs onto the GPU, thus requiring a procedure to schedule the requests. Clustering offers opportunities for efficient scheduling that incorporates the cluster assignments of LoRAs corresponding to the incoming requests.

Privacy presents another research direction, particularly regarding the possibility of information leakage during joint compression. As a preliminary study, Appendix H.2 investigates whether a base model with an adapter A for task T_A , after being jointly compressed alongside an adapter B for task T_B , inadvertently improves on T_B . Such an outcome would indicate that adapter A acquired information from adapter B. Our ablation study shows no performance gains on T_B , suggesting that the compressed adapter A remains independent and does not leak—or gain—information from adapter B. A more detailed investigation of the privacy properties of joint compression is an interesting next step.

In conclusion, our research advances LLM deployment by providing robust, scalable, and efficient compression. The ability of compressed LoRAs to maintain high performance while saving resources opens avenues for the broad application and adoption of LLMs across various industries. We encourage the community to build upon our findings and shared LoRAs to further enhance these technologies.

Impact Statement

This paper presents work whose goal is to advance machine learning. There are no societal consequences of our work that we feel must be specifically highlighted here.

Acknowledgements

The MIT Geometric Data Processing Group acknowledges the generous support of Army Research Office grants W911NF2010168 and W911NF2110293, of National Science Foundation grant IIS2335492, from the CSAIL Future of Data program, from the MIT–IBM Watson AI Laboratory, from the Wistron Corporation, and from the Toyota–CSAIL Joint Research Center.

References

- Lequn Chen, Zihao Ye, Yongji Wu, Danyang Zhuo, Luis Ceze, and Arvind Krishnamurthy. Punica: Multi-tenant lora serving, 2023. URL https://arxiv.org/abs/2310.18547.
- Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017.
- Leshem Choshen, Elad Venezian, Noam Slonim, and Yoav Katz. Fusing finetuned models for better pretraining. *ArXiv*, abs/2204.03044, 2022.
- Leshem Choshen, Elad Venezian, Shachar Don-Yehiya, Noam Slonim, and Yoav Katz. Where to start? analyzing the potential value of intermediate models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 1446–1470, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main. 90. URL https://aclanthology.org/2023.emnlp-main.90.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.
- Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. In *Low-Power Computer Vision*, pp. 291–326. Chapman and Hall/CRC, 2022.
- Tom Gunter, Zirui Wang, Chong Wang, Ruoming Pang, Andy Narayanan, Aonan Zhang, Bowen Zhang, Chen Chen, Chung-Cheng Chiu, David Qiu, et al. Apple intelligence foundation language models. *arXiv preprint arXiv:2407.21075*, 2024.
- Moshik Hershcovitch, Leshem Choshen, Andrew Wood, Ilias Enmouri, Peter Chin, Swaminathan Sundararaman, and Danny Harnik. Lossless and near-lossless compression for foundation models. *arXiv* preprint arXiv:2404.15198, 2024.

- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameterefficient transfer learning for nlp. In *International con*ference on machine learning, pp. 2790–2799. PMLR, 2019.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.
- Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient crosstask generalization via dynamic lora composition, 2024.
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023a.
- Weisen Jiang, Baijiong Lin, Han Shi, Yu Zhang, and James T Kwok. Byom: Building your own multi-task model for free. *arXiv preprint arXiv:2310.01886*, 2023b.
- Dawid Jan Kopiczko, Tijmen Blankevoort, and Yuki M Asano. VeRA: Vector-based random matrix adaptation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NjNfLdxr3A.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.
- Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes, 2018.
- Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.15647*, 2023.
- Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.
- Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, 2022a. URL https://arxiv.org/abs/2205.05638.

- Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel.
 Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. Advances in Neural Information Processing Systems, 35:1950–1965, 2022b.
- Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation, 2024.
- Michael Matena and Colin Raffel. Merging models with fisher-weighted averaging. *arXiv preprint* arXiv:2111.09832, 2021.
- Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *arXiv preprint arXiv:2404.02948*, 2024.
- Mohammed Muqeeth, Haokun Liu, Yufan Liu, and Colin Raffel. Learning to route among specialized experts for zero-shot generalization. *arXiv preprint arXiv:2402.05859*, 2024.
- Kimia Nadjahi, Kristjan Greenewald, Rickard Brüel Gabrielsson, and Justin Solomon. Slicing mutual information generalization bounds for neural networks. In *ICML 2023 Workshop Neural Compression: From Information Theory to Applications*, 2023. URL https://openreview.net/forum?id=cbLcwK3SZi.
- OpenAI. Openai fine-tuning api. https://platform.openai.com/docs/guides/fine-tuning, 2024.
- Predibase. Multi-lora inference server that scales to 1000s of fine-tuned llms. https://loraexchange.ai, 2024.
- Colin Raffel. Building machine learning models like open source software. *Communications of the ACM*, 66(2): 38–40, 2023.
- Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras. arXiv preprint arXiv:2311.13600, 2023.
- Pratyusha Sharma, Jordan T. Ash, and Dipendra Misra. The truth is in there: Improving reasoning in language models with layer-selective rank reduction. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ozX92bu8VA.
- Ying Sheng, Shiyi Cao, Dacheng Li, Coleman Hooper, Nicholas Lee, Shuo Yang, Christopher Chou, Banghua

- Zhu, Lianmin Zheng, Kurt Keutzer, Joseph E. Gonzalez, and Ion Stoica. S-lora: Serving thousands of concurrent lora adapters, 2023.
- Together AI. Together fine-tuning. https://www.together.ai/products#fine-tuning, 2024.
- Sheng Wang, Boyang Xue, Jiacheng Ye, Jiyue Jiang, Liheng Chen, Lingpeng Kong, and Chuan Wu. Prolora: Partial rotation empowers more parameter-efficient lora. *ArXiv*, abs/2402.16902, 2024. URL https://api.semanticscholar.org/CorpusID: 268032580.
- Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. arXiv preprint arXiv:2204.07705, 2022.
- Yeming Wen and Swarat Chaudhuri. Batched low-rank adaptation of foundation models, 2024.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-theart natural language processing, 2020.
- Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak 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*, 2022.
- Prateek Yadav, Leshem Choshen, Colin Raffel, and Mohit Bansal. Compeft: Compression for communicating parameter efficient updates via sparsification and quantization. *arXiv* preprint arXiv:2311.13171, 2023a.
- Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b. URL https://openreview.net/forum?id=xtaX3WyCjl.
- Prateek Yadav, Colin Raffel, Mohammed Muqeeth, Lucas Caccia, Haokun Liu, Tianlong Chen, Mohit Bansal, Leshem Choshen, and Alessandro Sordoni. A survey on

- model moerging: Recycling and routing among specialized experts for collaborative learning. *arXiv* preprint *arXiv*:2408.07057, 2024.
- Yuchen Zeng and Kangwook Lee. The expressive power of low-rank adaptation, 2024.
- Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-
- tuning. In The Eleventh International Conference on Learning Representations, 2023.
- Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. Asymmetry in lowrank adapters of foundation models. *arXiv preprint arXiv:2402.16842*, 2024.

A. Joint Diagonalization Algorithms

A.1. Alternating Methods

Our goal is to derive algorithms that optimize equation 1. Common to both methods, we expand the objective functional:

$$\sum_{i} \|B_{i}A_{i} - U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} = \sum_{i} \operatorname{tr}((B_{i}A_{i} - U\Sigma_{i}V^{\top})(B_{i}A_{i} - U\Sigma_{i}V^{\top})^{\top}) \text{ by definition}$$

$$= \sum_{i} \left[\operatorname{tr}(B_{i}A_{i}A_{i}^{\top}B_{i}^{\top}) - 2\operatorname{tr}(B_{i}A_{i}V\Sigma_{i}^{\top}U^{\top}) + \operatorname{tr}(U\Sigma_{i}V^{\top}V\Sigma_{i}^{\top}U^{\top}) \right]$$

$$= \operatorname{const.} - 2\sum_{i} \operatorname{tr}(B_{i}A_{i}V\Sigma_{i}^{\top}U^{\top}) + \sum_{i} \|U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2}.$$

$$(5)$$

Using this expansion, we now consider the two settings discussed in §3.1.

Case 1: Non-diagonal Σ_i , orthogonal U, V. Setting the derivative of equation 5 with respect to Σ_i to zero, we find

$$\Sigma_i = \Sigma_i^*(U, V) = U^{\top} B_i A_i V. \tag{6}$$

We simplify our objective function after plugging in this expression:

$$\sum_{i} \|B_{i}A_{i} - U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} + \operatorname{const.} = \sum_{i} \left[\|\Sigma_{i}\|_{\operatorname{Fro}}^{2} - 2\operatorname{tr}(B_{i}A_{i}V\Sigma_{i}^{\top}U^{\top}) \right] \text{ from equation 5}$$

$$= \sum_{i} \left[\operatorname{tr}(U^{\top}B_{i}A_{i}VV^{\top}A_{i}^{\top}B_{i}^{\top}U) - 2\operatorname{tr}(B_{i}A_{i}VV^{\top}A_{i}^{\top}B_{i}^{\top}UU^{\top}) \right] \text{ from equation 6}$$

$$= -\sum_{i} \operatorname{tr}(B_{i}A_{i}VV^{\top}A_{i}^{\top}B_{i}^{\top}UU^{\top}).$$

Substituting equation 6, we find

$$U_{opt}, V_{opt} = \arg\max_{\substack{U^{\top}U = I\\VV^{\top} = I}} \sum_{i=1}^{n} \|U^{\top}B_{i}A_{i}V\|_{Fro}^{2} = \arg\max_{\substack{U^{\top}U = I\\VV^{\top} = I}} \sum_{i=1}^{n} \|\Sigma_{i}^{*}(U, V)\|_{Fro}^{2}.$$
(7)

Note that

$$\sum_{i=1}^{n} \|U^{\top} B_i A_i V\|_{\text{Fro}}^2 = \operatorname{tr}\left(\left(\sum_{i=1}^{n} B_i A_i V V^{\top} A_i^{\top} B_i^{\top}\right) U U^{\top}\right)$$
$$= \operatorname{tr}\left(\left(\sum_{i=1}^{n} B_i^{\top} A_i^{\top} U U^{\top} A_i B_i\right) V V^{\top}\right),$$

by the identity $||A||_{Fro}^2 = \operatorname{tr}(A^{\top}A)$. Hence, we optimize equation 7 by alternating between U and V:

- U iteration: Define $M := \sum_i B_i A_i V V^\top A_i^\top B_i^\top$. Parenthesizing this expression properly requires only O((m+n)r) storage/computation time. With this definition, we maximize $\operatorname{tr}(MUU^\top)$ over U satisfying $U^\top U = I$. Since M is positive semidefinite, the optimum is to take U to be the r eigenvectors of M with largest eigenvalue, equivalent to an SVD problem.
- V iteration: Define $N := \sum_i A_i^\top B_i^\top U U^\top B_i A_i$. Similarly to the previous step, we take V to contain the r eigenvectors of N with largest eigenvalue, again solvable using an SVD.

This method decreases the objective in each step.

Case 2: Diagonal Σ_i . If constrain Σ_i to be diagonal, we interpret our objective function equation 1 as a "triple least

squares" problem. We compute gradients:

$$\nabla_{U} \sum_{i} \|B_{i}A_{i} - U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} = 2\sum_{i} (U\Sigma_{i}V^{\top} - B_{i}A_{i})V\Sigma_{i}^{\top}$$

$$\nabla_{V} \sum_{i} \|B_{i}A_{i} - U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} = 2\sum_{i} (V\Sigma_{i}^{\top}U^{\top} - A_{i}^{\top}B_{i}^{\top})U\Sigma_{i}$$

$$\nabla_{\Sigma_{i}} \sum_{i} \|B_{i}A_{i} - U\Sigma_{i}V^{\top}\|_{\operatorname{Fro}}^{2} = 2U^{\top}(U\Sigma_{i}V^{\top} - B_{i}A_{i})V$$

These expressions suggest efficient $r \times r$ linear systems to solve for U, V:

$$U = \left(\sum_{i} B_{i} A_{i} V \Sigma_{i}^{\top}\right) \left(\sum_{i} \Sigma_{i} V^{\top} V \Sigma_{i}^{\top}\right)^{-1}$$
$$V = \left(\sum_{i} A_{i}^{\top} B_{i}^{\top} U \Sigma_{i}\right) \left(\sum_{i} \Sigma_{i}^{\top} U^{\top} U \Sigma_{i}\right)^{-1}.$$

For Σ_i , we extract the diagonal from our gradient above:

$$\operatorname{diag}(U^{\top}U\Sigma_{i}V^{\top}V)_{j} = (U^{\top}U\Sigma_{i}V^{\top}V)_{jj}$$

$$= \sum_{m} (U^{\top}U)_{jm}\Sigma_{imm}(V^{\top}V)_{mj}$$

$$= (U^{\top}U \circ V^{\top}V)\operatorname{diag}(\Sigma_{i})$$

$$\operatorname{diag}(U^{\top}B_{i}A_{i}V)_{j} = \sum_{m} (U^{\top}B_{i})_{jm}(A_{i}V)_{mj}$$

$$= \sum_{m} (U^{\top}B_{i})_{jm}(V^{\top}A_{i}^{\top})_{jm}$$

$$= (U^{\top}B_{i} \circ V^{\top}A_{i}^{\top})\mathbf{1}$$

$$\Longrightarrow \operatorname{diag}(\Sigma_{i}) = (U^{\top}U \circ V^{\top}V)^{-1}(U^{\top}B_{i} \circ V^{\top}A_{i}^{\top})\mathbf{1}$$

Here o denotes the Hadamard product.

Combining these expressions, we use a simple coordinate descent algorithm cycling between the following three steps:

- 1. Solve for U
- 2. Solve for *V*
- 3. Solve for the Σ_i 's
- 4. Optionally, normalize so $\sum_{i} \|\Sigma_{i}\|_{\text{Fro}}^{2} = 1$

A.2. Additional Eigenvalue Iteration Algorithm

For the first case in §A.1, we introduce an alternative algorithm that eschews the use of SVD. This alternative is optimized for GPU execution, enabling tractable runs to convergence.

To derive this algorithm, we employ Lagrange multipliers to formulate the derived objective from equation 7:

$$U_{opt}, V_{opt} = \arg\max_{\substack{U^{\top}U = I\\VV^{\top} = I}} \sum_{i=1}^{n} \|U^{\top}B_{i}A_{i}V\|_{Fro}^{2},$$
(8)

yielding the expression

$$\Lambda = -\frac{1}{2} \|U^{\top} B_i A_i V\|_{\text{Fro}}^2 - \frac{1}{2} \text{tr}(X^{\top} (I - U^{\top} U)) - \frac{1}{2} \text{tr}(Y^{\top} (I - V^{\top} V)). \tag{9}$$

Taking the derivatives gives

$$\nabla_U \Lambda = -\sum_i B_i (A_i V) (V^\top A_i^\top) (B_i^\top U) + UX$$
(10)

$$\nabla_V \Lambda = -\sum_i A_i^\top (B_i^\top U) (U^\top B_i) (A_i V) + V Y \tag{11}$$

Setting these derivatives to zero shows

$$\sum_{i} B_i(A_i V)(V^{\top} A_i^{\top})(B_i^{\top} U) = UX$$
(12)

$$\sum_{i} A_i^{\top}(B_i^{\top}U)(U^{\top}B_i)(A_iV) = VY. \tag{13}$$

Here, one can show that the Lagrange multiplier matrices X and Y are diagonal and nonnegative, since the problem reduces to an eigenvalue problem when either U or V is fixed; this is essentially the argument behind the alternating algorithm in Appendix A. Hence, taking inspiration from classical eigenvalue iteration, we use the following updates to improve our estimates of U and V:

$$U_0^{(k+1)} \leftarrow \sum_{i} B_i(A_i V^{(k)}) ((V^{(k)})^\top A_i^\top) (B_i^\top U^{(k)})$$
(14)

$$V_0^{(k+1)} \leftarrow \sum_i A_i^{\top} (B_i^{\top} U^{(k)}) ((U^{(k)})^{\top} B_i) (A_i V^{(k)})$$
(15)

$$U^{(k+1)} \leftarrow \text{orthogonalize}(U_0^{(k+1)}) \tag{16}$$

$$V^{(k+1)} \leftarrow \text{orthogonalize}(V_0^{(k+1)}) \tag{17}$$

Here, the function orthogonalize orthogonalizes the columns of a matrix, e.g. by using the Q part of the reduced-size QR factorization. Although we lack a formal convergence proof, in practice we find that this method reliably reaches a local optimum of our problem.

By executing matrix operations in the specified sequence, these computations can be rapidly performed on GPUs. Note the expressions above are parenthesized to avoid constructing a large matrix product as an intermediate computation.

A.3. Clustering algorithm

Initialization: We run joint diagonalization with a single U, V then perform k-means with k clusters on the space of Σ_i 's. This gives us our first clusters and we can use random initialization U_j, V_j for each cluster but the Σ_i can be maintained as initialization.

Step 1: Using the alternating JD algorithms from earlier in this section, we optimize the problem $\min_{U_i, V_i, \Sigma_i} \sum_{i \in C_i} ||B_i A_i - U_j \Sigma_i V_j^\top||_F^2$ for each j independently.

Step 2: New cluster assignment for $i : \min_j \min_{\Sigma_i} ||B_i A_i - U_j \Sigma_i V_j^{\top}||_F^2$. If any assignment changes we go to Step 1, else we have converged.

B. Proof of Theorem 1

Proof. For the lower bound, note that by Jensen's inequality,

$$\sum_{i=1}^{n} \|U^{\top} B_i A_i V\|_{\text{Fro}}^2 \ge \left\| U^{\top} \sum_{i=1}^{n} B_i A_i V \right\|_{\text{Fro}}^2,$$

for any U, V. Hence,

$$\sup_{U,V \in St(k,d)} \sum_{i=1}^{n} \|U^{\top} B_i A_i V\|_{Fro}^2 \ge \sup_{U,V \in St(k,d)} \left\| U^{\top} \sum_{i=1}^{n} B_i A_i V \right\|_{Fro}^2.$$
 (18)

By the definition of singular value decomposition, the right hand side of equation 18 is maximized with U, V being the top r singular vectors of $\sum_{i=1}^{n} B_i A_i$, yielding $\|U^{\top} \sum_{i=1}^{n} B_i A_i V\|_{\text{Fro}}^2 = \sum_{i=1}^{r} \bar{\sigma}_i^2$. Recalling that $\Sigma_i = U^{\top} B_i A_i V$ yields the lower bound.

For the upper bound, recall that $\Sigma_i = U^{\top} B_i A_i V$. Rearranging,

$$\operatorname{vec}(\Sigma_i) = (V^{\top} \otimes U^{\top})\operatorname{vec}(B_i A_i).$$

Define

$$\bar{\Sigma} := [\operatorname{vec}(\Sigma_1), \dots, \operatorname{vec}(\Sigma_n)].$$

By our previous simplification,

$$\bar{\Sigma} = (V^{\top} \otimes U^{\top})L.$$

Now

$$\sum_{i=1}^{n} \|\Sigma_i\|_{\operatorname{Fro}}^2 = \|\bar{\Sigma}\|_{\operatorname{Fro}}^2 = \operatorname{tr}\left(((V \otimes U)(V \otimes U)^{\top})(LL^{\top})\right)$$

Since U, V are orthogonal and size $d \times r$, the top r^2 eigenvalues of the symmetric matrix $(V \otimes U)(V \otimes U)^{\top}$ will be equal to 1, and the rest will equal 0. The eigenvalues of the symmetric matrix LL^{\top} will be equal to the squared singular values of L. We can then apply the Von Neumann trace inequality to obtain the upper bound.

The last statement follows from the Pythagorean theorem and the fact that the Σ_i is a projection of B_iA_i to the U,V subspace.

Note that we have only used the fact that the matrix $(V \otimes U)$ has singular values equal to 1; we have not used the fact that it has Kronecker product structure. On the other hand, each vector $\text{vec}(B_iA_i)$ is a sum of r_i Kronecker products and cannot be expressed as a Kronecker product. As a result, while the upper bound in the Von Neumann trace inequality is achieved if the eigenvectors of the two matrices align, the Kronecker product structure is a severe constraint and the upper bound we have provided is generous.

C. Training LoRAs

We trained LoRA adapters on 500 natural instruction tasks (Wang et al., 2022) using Mistral-7B-Instruct-v0.2 (Jiang et al., 2023a) as the base model. All LoRA adapters were configured with a rank of 16, i.e., $\forall i, r_i = 16$. We selected 10 diverse tasks manually for consistent evaluation across experiments and randomly sampled an additional 490 tasks, resulting in a total of 500 tasks. These tasks were exclusively in English (both input and output), ensuring higher quality and thorough review (Wang et al., 2022). Each task dataset was divided into training, validation, and test sets (80-10-10). Hyperparameters, such as early stopping, were tuned using the validation sets; that is, we train for five epochs and take the best-performing epoch-checkpoint per validation loss. Evaluation on the test sets demonstrated that LoRA consistently outperformed the base model in terms of both Rouge scores and loss metrics (see Table 1).

In Table 1, we compare metrics between base model and LoRA finetuning.

Table 1: Comparison of metrics before and after LoRA training across 1000 tasks.

Metric	Base Model	LoRA
Loss	4.14 ± 3.07	0.56 ± 0.58
Exact Match	1.81 ± 6.56	51.38 ± 40.90
Rouge-1	21.70 ± 19.22	68.88 ± 29.73
Rouge-L	20.62 ± 18.21	67.80 ± 30.15

In Table 3 we include all 1000 tasks that were used.

We use Huggingface (Wolf et al., 2020) in our implementation. For the base model, we use quantization with configuration:

Table 2: Main Evaluation Tasks

Task Number	Name	Type	Domain
task280	stereoset_classification_stereotype_type	classification	stereoset
task190	snli_classification	snli	image captions
task391	causal_relationship	commonsense	cause and effect
task290	tellmewhy_question_answerability	answerability	story
task1391	winogrande_easy_answer_generation	commonsense	social and physical
task1342	amazon_us_reviews_title	title generation	amazon reviews
task442	com_qa_paraphrase_question_generation	question generation	wikipedia
task620	ohsumed_medical_subject_headings_answer_generation	keyword tagging	scientific
task1598	nyc_long_text_generation	data to text	restaurants
task039	qasc_find_overlapping_words	overlap extraction	natural science

```
BitsAndBytesConfig(
    load_in_4bit=True,
    bnb_4bit_use_double_quant=True,
    bnb_4bit_quant_type="nf4",
    bnb_4bit_compute_dtype=torch.bfloat16,
)
```

and LoRA configuration:

```
LoraConfig(
    r=16,
    lora_alpha=32,
    target_modules=["q_proj", "k_proj", "v_proj"],
    lora_dropout=0.05,
    bias="none",
    task_type="CAUSAL_LM",
    init_lora_weights=init_lora_weights,
)
```

D. Avoiding Batched Matrix Multiplication (BMM)

Fast LoRA (Wen & Chaudhuri, 2024) aims to alleviate the batched matrix multiplication (BMM) bottleneck when serving many LoRAs. They propose an adapter parameterization that replaces addition with elementwise multiplication, avoiding BMM and improving LoRA throughput at lower ranks. Our JD LoRA formulation also circumvents or heavily reduces the impact of BMM as discussed below, and both individual and joint compression methods can be applied to Fast LoRAs.

In the envisioned deployment scenario, a service provider hosts a large collection of LoRAs. Upon receiving a request, each user specifies both the input data and the desired LoRA identifier. The provider then processes the base model augmented with the specified LoRA for each user's data. As a provider is batching a collection of requests for GPU parallelization, they can expect to frequently have more than one unique LoRA identifier per batch.

Traditionally, a specific LoRA is integrated into the base model by transforming $W_0 \to W_0 + B_i A_i$. Serving multiple LoRAs conventionally would necessitate maintaining and executing a separate copy of the base model for each LoRA, bringing substantial computational overhead. Alternatively, the computation for $W_0 x$ and $B_i A_i x$ can be performed independently and subsequently merged. This strategy necessitates only a single instance of $W_0 x$ computation and storage of LoRA-specific parameters rather than the entire base model.

Consider the batch processing of BAx, where boldface indicates that B_i , A_i are stacked into tensors of dimensions ($b \times b$)

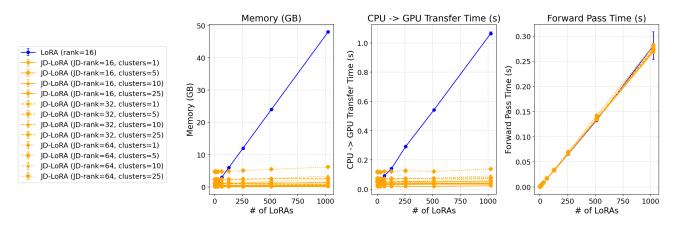


Figure 5: Memory load, transfer time, and forward-pass performance of LoRA and JD-LoRA.

 $m \times r$) and $(b \times r \times n)$ respectively, with batched data **x** shaped $(b \times l \times n)$:

$$\begin{aligned} \mathbf{A}\mathbf{x} &\leftrightarrow (b \times r \times n) \times (b \times l \times n) \rightarrow (b \times l \times r) \text{ bmm} \\ \mathbf{B}(\mathbf{A}\mathbf{x}) &\leftrightarrow (b \times m \times r) \times (b \times l \times r) \rightarrow (b \times l \times m) \text{ bmm}. \end{aligned}$$

Here, "bmm" denotes batched matrix multiplication, a known bottleneck in both throughput and latency. Consider the corresponding operations for our joint compression scheme, $U\Sigma V^{\top}x$:

$$V^{\top}\mathbf{x} \leftrightarrow (\tilde{r} \times n) \times (b \times l \times n) \rightarrow (b \times l \times \tilde{r}) \text{ broadcasted}$$

$$\mathbf{\Sigma}(V^{\top}\mathbf{x}) \leftrightarrow (b \times \tilde{r}) \times (b \times l \times \tilde{r}) \rightarrow (b \times l \times \tilde{r}) \text{ broadcasted}$$

$$U(\mathbf{\Sigma}V^{\top}\mathbf{x}) \leftrightarrow (m \times \tilde{r}) \times (b \times l \times \tilde{r}) \rightarrow (b \times l \times m) \text{ broadcasted}$$

In our optimized setup, batched matrix multiplications can be completely circumvented if the Σ_i matrices are diagonal. If not, given that $\tilde{r} \ll m, n$, any required batched matrix multiplication remains computationally inexpensive.

E. Simple Timing Experiments

In Figure 5, we present a set of simple experiments comparing the memory load, transfer time, and forward-pass performance of LoRA and JD-LoRA. These experiments were conducted across multiple clusters and various rank configurations. For memory usage, we measured all 96 LoRA modules in the Mistral model; however, for transfer time and forward-pass performance, we tested only a single LoRA module. We also implemented F-LoRA (Wen & Chaudhuri, 2024), but contrary to their reported results, we were unable to achieve faster forward-pass performance than standard LoRA.

F. GPU Memory Usage Computation for JD Compression.

The GPU memory consumption is primarily influenced by the number of parameters that need to be stored and processed during inference. In this section, we introduce the detail of how we compute the GPU consumption of our method, and how we find the number of vLLM multi-LoRA that share the same GPU utilization.

- D: Hidden dimension size (e.g., D = 4098).
- r: Rank of the shared basis matrices for compression (e.g., r = 16, 32, 64).
- N: Maximum number of LoRA modules being served simultaneously (max_lora_num).
- c: Number of clusters in our clustering method (e.g., c = 7, 10, 25).

In Figure 1, we use different JD-compression settings for serving different number of unique LoRAs. Specifically:

• Serving 4 unique LoRAs:

Ours: rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 2.

• Serving 8 unique LoRAs:

Ours: rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 2.

• Serving 16 unique LoRAs:

Ours: rank 32 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 3.

• Serving 32 unique LoRAs:

Ours: rank 64 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 5.

• Serving 64 unique LoRAs:

Ours: rank 64 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 6.

• Serving 128 unique LoRAs:

Ours: 7 clusters, rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 8.

• Serving 256 unique LoRAs:

Ours: 10 clusters, rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 10.

• Serving 512 unique LoRAs:

Ours: 25 clusters, rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 26.

• Serving 1024 unique LoRAs:

Ours: 25 clusters, rank 16 JD-Full.

vLLM multiLoRA baseline: max-gpu-lora = 28

F.1. Baseline GPU Memory Usage

The baseline for our comparison is the standard LoRA method with a rank of 16. The total parameter count for the baseline is given by:

Params_{baseline} =
$$D \times 2 \times 16$$
.

This accounts for the parameters in the LoRA-adapted layers, where the factor of 2 represents the weights and biases.

F.2. GPU Memory Usage for JD Full Method

For the Joint Decomposition (JD) Full method without clustering, the total parameter count is:

Params_{ID Full} =
$$D \times 2 \times r + N \times r^2$$
.

- $D \times 2 \times r$: Parameters for the base model adapted with rank-r LoRA.
- $N \times r^2$: Additional parameters introduced by each of the N LoRA modules, each of size $r \times r$.

The GPU memory usage ratio relative to the baseline is:

$$\mbox{GPU Usage Ratio}_{\mbox{\scriptsize JD_Full}} = \frac{\mbox{\scriptsize Params}_{\mbox{\scriptsize JD_Full}}}{\mbox{\scriptsize Params}_{\mbox{\scriptsize baseline}}} = \frac{D \times 2 \times r + N \times r^2}{D \times 2 \times 16}.$$

F.3. GPU Memory Usage for Clustering Method

When employing clustering, the parameter count changes due to the addition of cluster-specific parameters:

Params_{Clustering} =
$$D \times 2 \times r \times c + N \times (r^2 + 1)$$
.

• $D \times 2 \times r \times c$: Parameters for the base model adapted with rank-r LoRA across c clusters.

• $N \times (r^2 + 1)$: Additional parameters for each LoRA module and cluster assignments.

The GPU memory usage ratio is:

$$\text{GPU Usage Ratio}_{\text{Clustering}} = \frac{\text{Params}_{\text{Clustering}}}{\text{Params}_{\text{baseline}}} = \frac{D \times 2 \times r \times c + N \times (r^2 + 1)}{D \times 2 \times 16}.$$

F.4. Punica

In our vLLM experiments, we specifically used the Punica kernel for implementing multi-LoRA, applying our approach in conjunction with Punica's capabilities. Our custom function, add_lora_slice_with_sigma, implements the following key steps:

- 1. Initialize Buffers: Creates temporary storage for intermediate calculations if not already provided.
- 2. Apply Matrix A: Transforms x using matrix A, storing the result in buffer.
- 3. Apply Matrix Sigma: Further transforms buffer using Sigma, storing the result in buffer_sigma.
- 4. Apply Matrix B and Update y: Finally, transforms buffer_sigma using B, applies scaling, and updates a slice of y in place.

Below is the pseudocode for add_lora_slice_with_sigma, illustrating the integration:

Listing 1: Pseudocode for 'add_lora_slice_with_sigma'

```
Function add lora slice with sigma(y, x, wa t all, wb t all, wsigma t all, indices,
      layer_idx, scale, y_offset, y_slice_size, buffer=None):
        Initialize buffers if not provided
      if buffer is None:
          buffer = create_tensor(shape=(x.size(0), R), dtype=float32)
4
          buffer_sigma = create_tensor(shape=(buffer.size(0), R), dtype=float32)
       # Step 1: Apply matrix A
      dispatch_bgmv_low_level(buffer, x, wa_t_all, indices, layer_idx, scale=1.0)
       # Step 2: Apply matrix Sigma
      dispatch_bgmv_low_level(buffer_sigma, buffer, wsigma_t_all, indices, layer_idx, scale
10
       # Step 3: Apply matrix B and update y slice
      dispatch_bgmv_low_level(y, buffer_sigma, wb_t_all, indices, layer_idx, scale, y_offset
11
          , y_slice_size)
  End Function
```

G. Selecting Number of Clusters

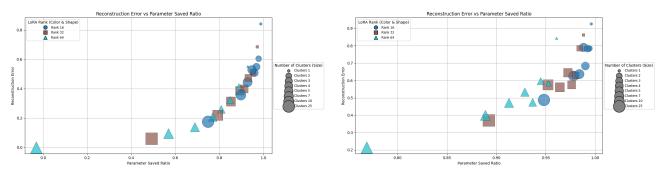
To identify optimal hyperparameters for the clusters compression method, we analyzed the relationship between reconstruction error and the parameter saved ratio for a single LoRA module, as shown in Figure 6. By comparing the results across different numbers of Low-Rank Adaptation (LoRA) configurations (100 and 500, depicted in subfigures 6a and 6b), we were able to observe the trade-off between model size reduction and reconstruction accuracy. Based on these findings, we selected the rank and number-of-clusters hyperparameters that effectively balance these two objectives. The chosen settings were then used to conduct full-scale experiments.

H. Additional Results

This section elaborates on the results that underpin the figures presented in the main text and showcases a consistent correlation across various evaluation metrics. Additionally, we assess the significance of achieving convergence and the performance of compression on new unseen LoRA models.

H.1. LoRAs of different ranks

In Table 4, we report the performance of our compression method on LoRAs with ranks uniformly sampled between 16 and 64 (mean rank of 43). In Table 5, we present compression results for LoRAs of rank 43, matching the average rank from Table 4. Because these same-rank LoRAs have an identical parameter count, they also exhibit identical parameter-saving



- (a) Recon. Error vs Parameter Saved Ratio for 100 LoRAs
- (b) Recon. Error vs Parameter Saved Ratio for 500 LoRAs

Figure 6: Comparison of reconstruction error against the parameter saved ratio for different numbers of LoRA configurations for a single LoRA module. The left subplot shows results for 100 LoRAs, while the right subplot displays results for 500 LoRAs. These plots illustrate the trade-off between reconstruction accuracy and compression efficiency, providing insights into optimal parameter settings for compression.

ratios. While performance declines slightly for LoRAs with varying ranks, our compression method still preserves over 99% of the original performance.

H.2. Privacy Ablation

We investigate whether jointly compressing certain tasks results in improved performance on tasks within that same compressed group, compared to tasks outside of the group. In other words, we examine whether information about which tasks were compressed together could be inferred from subsequent performance differences. Table 6 presents the results.

In the Compressed Together setting, tasks (task1391, task190, task280, task290, and task391) are compressed jointly with five other tasks. We then evaluate their cross-task performance within this group. In the Compressed Separately setting, the same set of tasks is each compressed alongside nine other tasks that are not part of the original group. We again evaluate the cross-task performance using the same sets of tasks, allowing us to compare and assess any differences in performance attributable to joint versus separate compression.

Table 6: Privacy Ablation: Performance on tasks compressed together vs. performance on tasks compressed separately

Source Task	Task Combination	Test Loss	Exact Match	Rouge1	RougeL
	Com	pressed Toge	ether		
	task1391 on task190	2.320	29	29.00	29.00
	task1391 on task280	1.445	10	12.00	12.00
task1391	task1391 on task290	2.662	0	0.00	0.00
	task1391 on task391	1.560	4	7.00	7.00
	Average	1.997	10.8	12.00	12.00
	task190 on task1391	1.392	0	0.00	0.00
	task190 on task280	1.647	2	2.00	2.00
task190	task190 on task290	1.838	0	0.00	0.00
task190	task190 on task391	2.622	2	2.00	2.00
	Average	1.875	1.0	1.00	1.00
	task280 on task1391	2.259	14	16.47	16.47
	task280 on task190	2.922	20	20.50	20.50
task280	task280 on task290	0.729	33	43.07	43.07
	task280 on task391	2.318	44	62.52	62.52
	Average	2.057	27.8	35.64	35.64
	task290 on task1391	0.867	36	46.58	46.58
	task290 on task190	1.553	36	36.00	36.00
task290	task290 on task280	1.036	43	43.40	43.40
	task290 on task391	0.461	59	86.33	86.33
	Average	0.979	43.5	53.08	53.08
	task391 on task1391	0.502	65	65.75	65.75

task391

Table 6: Privacy Ablation: Performance on tasks compressed together vs. performance on tasks compressed separately (continued)

Source Task	Task Combination	Test Loss	Exact Match	Rouge1	RougeL
	task391 on task190	1.421	31	31.00	31.00
	task391 on task280	0.417	69	69.00	69.00
	task391 on task290	0.265	71	90.33	90.33
	Average	0.651	59.0	64.02	64.02
	Comp	ressed Sepa	rately		
	task1391 on task190	2.347	30	30.00	30.00
	task1391 on task280	1.543	11	13.58	13.58
task1391	task1391 on task290	2.477	0	0.00	0.00
	task1391 on task391	1.253	10	18.00	18.00
	Average	1.905	12.8	15.40	15.40
	task190 on task1391	1.388	0	0.00	0.00
	task190 on task280	1.603	2	2.00	2.00
task190	task190 on task290	1.771	0	0.00	0.00
	task190 on task391	2.326	4	4.00	4.00
	Average	1.772	1.5	1.50	1.50
	task280 on task1391	3.111	2	5.13	5.13
	task280 on task190	2.711	19	19.00	19.00
task280	task280 on task290	0.948	16	22.09	22.09
	task280 on task391	2.449	39	53.11	53.11
	Average	2.305	19.0	24.83	24.83
	task290 on task1391	0.848	41	47.58	47.58
	task290 on task190	1.355	38	38.00	38.00
task290	task290 on task280	1.050	41	41.00	41.00
	task290 on task391	0.463	59	86.33	86.33
	Average	0.929	44.8	53.23	53.23
	task391 on task1391	0.428	68	68.74	68.74
	task391 on task190	1.507	31	31.00	31.00
task391	task391 on task280	0.368	70	70.00	70.00
	task391 on task290	0.269	73	91.00	91.00
	Average	0.643	60.5	65.18	65.18
	J				

H.3. Relative Rouge-L Performance and Compression Rate

Table 7 presents comprehensive results from the experiments underlying Figure 2 for each evaluation task. Additionally, we incorporate results using the Ties-merging benchmark (Yadav et al., 2023b), which consolidates all LoRA-adapters into a single adapter of identical configuration and parameter count; this integration significantly compromises performance.

H.4. Absolute Rouge-L Performance and Compression Rate

Table 8 provides the full results behind Table 7, but with Rouge-L scores instead of relative performance compared to LoRA.

H.5. Relative Rouge-1 Performance and Compression Rate

Table 9 provides full results for relative performance of Rouge-1, which shows the same trends as the results for relative performance of Rouge-L (Table 7).

H.6. Absolute Rouge-1 Performance and Compression Rate

Table 10 provides full results for absolute performance of Rouge-1, which shows the same trends as the results for absolute performance of Rouge-L (Table 8).

H.7. Relative Exact-Match Performance and Compression Rate

Table 11 provides full results for relative performance of exact-match, which shows the same trends as the results for relative performance of Rouge-L (Table 7).

H.8. Loss and Compression Rate

Table 12 provides full results for test loss (cross-entropy), which shows the same trends as the results for relative performance of Rouge-L (Table 7).

H.9. Agreement and Compression Rate

Table 13 provides full results for *agreement*, which shows the same trends as the results for relative performance of Rouge-L (Table 7). Note that *agreement* measures the exact match in task generations between the uncompressed LoRA model and the compressed LoRA model, rather than comparing to the task's ground truth data. The comparison is very strict and requires an exact match between the generations of the two models (LoRA and the compressed LoRA), comparing each sample one at a time.

H.10. Reconstruction Error and Compression Rate

Table 14 provides the full results of the experiments behind Figure 3 for every evaluation task.

H.11. Reconstruction Error: Trained vs. Random

Table 15 provides the reconstruction error on random (untrained) LoRA matrices. Comparing with Table 14, we find that reconstruction error is consistently higher on random (untrained LoRA) matrices than on trained LoRA matrices. This demonstrates that after training, LoRAs have a shared structure that JD exploits.

H.12. Convergence

Table 16 presents outcomes where the JD-Full algorithm is executed until convergence. Our convergence criterion is defined as follows:

$$\max\left(\|U_{t+1} - U_t U_t^\top U_{t+1}\|_{\operatorname{Fro}}/\|U_{t+1}\|_{\operatorname{Fro}}, \|V_{t+1} - V_t V_t^\top V_{t+1}\|_{\operatorname{Fro}}/\|V_{t+1}\|_{\operatorname{Fro}}\right) < \tau \tag{19}$$

where the tolerance threshold τ is set to 0.001. Due to the slow per-iteration computation times of the primary JD-Full algorithm, which quickly reaches an approximate optimum but then has a long tail of convergence for final digits of precision, we devised an alternative eigenvalue iteration algorithm (Appendix A.2) optimized for GPU acceleration. Our analysis indicates that adherence to this convergence criterion does not significantly alter the results.

H.13. Out-of-distribution Performance (LoRA-hub)

For completeness, we incorporate results using the protocol of LoRA-hub (Huang et al., 2024). That is, 100 LoRA-adapters are sampled, independent of the evaluation task, representing a measure of out-of-distribution performance. This also means that each result on a task is averaged across all 100 LoRA-adapters (as there is no *a priori* LoRA-to-task mapping). These results were obtained without normalizing the LoRA-adapters before applying the JD algorithms, a step we later identified as beneficial. We present performance comparison in Table 18. Table 17 presents the average agreement between uncompressed and compressed LoRA across 10 evaluation tasks. Results per task for JD-diagonal and JD-full are shown in Table 19 and Table 20, respectively.

From these tables, we find that the JD algorithms successfully maintain performance in this out-of-distribution context.

Table 3: List of all 1000 Tasks

Task ID	Description	Task ID	Description	Task ID	Description		
task100	concentrated all cleanurs from index its y concentrated all cleanurs from index its y pile treatment of the pile pile treatment of pile treatment pile treatm	tak1009		task101	review and conceiming all elements from index i to) review and conceiming all elements from index i to) grid conditions bed size, ple conditions bed size, the conditions of ple conditions to consider the contractions to consider the ple contractions to be relations by p directly a ple contraction to be relations by p directly a ple contraction to be relations by p directly p dir	task102 task1047	commengen sortener generation pile transistion englisht skripe pile transistion englisht selven pile transistion englisht engl
task 1048 task 1062 task 1082	pib translation telugu english pib translation marathi bengali pib translation marathi hindi	task105 task107 task1083	ph translution beaught kinds place to the property of the prop	task1053 task1079	pib translation hindi urdu pib translation english gujarati pib translation english marathi	mak102 mak1047 mak1057 mak1058 mak1058 mak1050 mak1090 mak11010 mak1115 mak1115 mak1125 mak1125 mak1125 mak1125 mak1126 mak1135 mak1126 mak1135 mak1146 mak1146 mak115 mak	pro translation region tempo più translation cerginiti mella contestunibi serdificazioni dissificazioni tvo number a
task1094 task110 task1106	array or prosuces ted translation on pt logic floxt sentence generation ted translation ar it	task1095 task1101 task1109	cracks mensioned array ted translation ja gl ted translation en st ted translation en st	task1097 task1102 task111	introperational per la professional de la constitución de la constituc	task1098 task1105 task1110	test translation en pr test translation ja lis test translation ar gl test translation be gl
task1111 task1116 task1120	ted translation he it alt id ja translation alt ja fil answer generation count from the country of letter	task1113 task1117 task1122 task1130	ted translation he fa alt is ind answer generation alt klem ja translation alt klem ja translation	task1114 task1118 task1127 task1132	ted translation he pt alt is fill translation alt is th translation	task1115 task112 task1128 task1135	alt ja id translation and simple perfected identification alt this translation the state of the
trak (1088) trak (1094) trak (110) trak (112) trak (113) trak (113) trak (1147) trak (1147) trak (1147) trak (1150)	near recommenses are classification near ar commenses are classification country currency	mak1089 mak1095 mak1100 mak1100 mak11109 mak1113 mak11117 mak1120 mak1144 mak1144 mak1144 mak1148 mak1148 mak1148 mak1148 mak1149 mak1190 mak1200 mak1200 mak1200 mak1200 mak1200 mak1200 mak1200 mak1200 mak1200 mak1200	and transduction by far and the process of the proc	tank109 tank1097 tank1097 tank1102 tank111 tank1114 tank1114 tank11132 tank1140 tank1145 tank1145 tank1145 tank1149 tank1152 tank1158 tank1158 tank117 tank1189 tank1197 tank1197 tank1197	nor pl communeme me classification nor jap communeme me classification item check edible	task1141 task1146 task115	two matter can. The control of the
task1156 task1156 task1167 task1186	delete max min bard analogical reasoning tools penn treebank course pox tagging me hmno classification	task1151 task1157 task1168 task1188	owap max run burd analogical reasoning rocens for containers berown course pos tagging count max fros char	task1152 task1158 task117 task1189	bard analogical reasoning cancalon bard analogical reasoning munipalating items up translation on de check char in strine	task1154 task116 task118 task119	hard analogical reasoning travel consilerate constructive reasoning senses/2 (2019 task 10 open vocabulary mathematical answer generation senses/2 (2019 task 10 open vocabulary mathematical answer reneration senses/2 (2019 task 10 open vocabulary mathematical answer reneration
task1190 task1194 task1199	add integer to list kth largest element atomic classification sattr	task1191 task1196 task1200	food vog noaveg atomic classification oeffect atomic classification seffect	task1192 task1197 task1201	food flavor profile* atomic classification oreact atomic classification xintent	task1193 task1198 task1202	food course classification atomic classification owner atomic classification xneed
task1203 task1207 task1210 task1214	atomic classification madesmof	task1208 task1211 task1215	atomic classification sreason	task1205 task1209 task1212 task1216	atomic classification happeoperty	task1206 task121 task1213 task1217	zest text monitoration storage classification desires
task1214 task1219 task1224 task1228 task1233 task1238	atoric classification twant ted translation en cs ted translation in ar- ted translation in ar- ted translation ar- ted translation ar- ted translation gl en	task122 task1225 task1229 task1235	comala ino index addition ted translation ja he ted translation es he ted translation es he	task1216 task1221 task1226 task123	atomic classification causes ted translation on the ted translation or on contain sort dictionary ted translation for or	task1217 task1222 task1227 task1232 task1237 task1240	atonic zonove generation ted translation in on ted translation or ja ted translation or ja ted translation at es
task1238 task1241 task1246 task1252	ted translation gl en ted translation gl ar ted translation gl pt ted translation it gl	task1215 task1225 task1225 task1225 task1235 task1239 task1242 task1247 task1244 task1246 task1264	domes (Casudacino Basadoveut amonic Casudacino apubbod amonic Casudacino apubbod to a compation ja bac to a compation ja bac to a compation be to a compation be to a compation be to a compation of the to a compation of the to a compation of the to a compation of the and translation of the	task123 task1236 task124 task1244 task1248 task1255	our internation of our constant of dictionary constant are dictionary constant pair averages constant pair averages to distantains of plated translations if a cold translation in pt.	task1240 task1245 task125 task1257	tool transitionise by it tool transitionise by are tool transitionise by are tool transitionise give tool transitionise to come
task1258 task1263 task127	ted translation pl es	task126 task1264 task1270	ned translation it fa scan refrestred test generations command action all soft translation of pt scan refrestred test generation command action short strengs informativeness chamistation test chamistation test chamistation soft translation soft translati	tusk1266	ted translation fa pa	task1262	
task1273 task1279 task1283 task1288	on transcence per ra warm lengt sext generation action constraind all ted translation in pt ted translation pt gl hrugo quality classification rlus retro-marabrasium	task1274 task128 task1284 task1289	ted translation pt en scan structured test generation command action short hrugo informativemens classification tree classification	task1276 task1280 task1285 task129	ted translation pt ex ted translation pt it kpa keypoint matching scan lone text remeation action command short	task 1272 task 1278 task 1281 task 1286 task 1292	ted translation for the ted translation for the properties assume that the properties assume that properties as properties
task 1288 task 1293 task 1309 task 1312 task 1316 task 132	The control of the co	task1294 task131 task1313	while parameter varietisms with get a grane and successful long war he get a grane and successful long underly valling cold. I want to the cold of th	task130 task1310 task1314	kpa keypoint mulching scale long text generation action command action long arrantereview rating chaotification country abstraction country text contract country text contract country text contract country contract contr	mak1296 mak1292 mak1308 mak1311 mak1315 mak1319 mak1322 mak1336 mak1338 mak1338 mak1342 mak1350 mak1370 mak1370 mak1370 mak1370 mak1370 mak1370	amazonteview rating classification find canne array
task132 task1323 task1327	dais text medification open subsitles he en translation que zee answer generation from question	task1320 task1324 task1328	constay domain til open subbilis is en translation qa zer relation generation from question	task1321 task1325 task1330	country continent que ze question generation on subject relation open substitles on te translation	task1322 task1326 task1331	country by basined profits go are question generated from moreover go are question generated from moreover go are question generated from moreover go are question generated from the good good good good good good good goo
mich 323 mich 327 mich 327 mich 332 mich 339 mich 336 mich 353 mich 360 mich 371 mich 378 mich 381 mich 383	check leap year peixian equity evaluation corpus text completion glue cola grammatical correctness classification had encore translation on hi	task1333 task1340 task1347 task1354	check valstity date determyryy mer teat compression compression glae sts-b-similarity classification sent como classification	task1336 task1341 task1351 task1355	country continent per requesting guarantine on subject relation per requesting guarantine on subject relation prices are guilty evaluation corpus guarder classifier points are guilty evaluation corpus and corpus summeratation and corpus summeratation and corpus summeratation and corpus summeratation and consideration subsidies and consideration subsidies definitely supplementations and relations and relations and relations	task1338 task1342 task1352 task1359	persuas equity evaluation corpus sentament classather attancos un reviewes title hind encorp intrashitien to en nature viewes annount encoration
task1360 task1371 task1378 task1381	numer sense multiple choice qui generation newscomm translation quared connect answer generation	task1364 task1373 task1379	hans answer generation netwoems translation quarel incorrect answer generation	task137 task1375 task138 task1383	detoxifying-Iran classification toxicity newscorner translation detoxifying-Iran classification fluency	task1370 task1377 task1380 task1384	newcome classification newcome translation quezel correct option generation quezel correct option generation deal or modeling and topic resident
task1385 task139 task1394	anli rl entailment deteorifying-linx classification topicality meta wox task classification	task1386 task1390 task1397	spaints with control another and it? embalment wechised coefference curepa code to fr on translation	task1387 task1391 task1398	spired with mechanisms and it's citalianced winogrande casy answer generation obeq question generation	task1389 task1393 task1399	heal to no unacopy saturnamenta hellawang completion superglux cops text completion obea survey generation
task140 task1404 task141 task142	detectifying-lins classification style date conversion odd-man-out classification category odd-man-out classification no category	task1400 task1405 task1412 task1420	obsa incorrect answer generation find median web questions question answering mathea weneral	task1401 task1406 task1418 task1421	obea sentence generation ich sraullest element bless semantic relation classification mulhas other	task1403 task1409 task1419 task1422	check validity date muddyyyy dart text generation date text generation multipa gain multipa prioric
task1423 task1427 task1431	mathqu geometry country region in world head qu answer generation	task1424 task1428 task1432	mathiga probability country surface area head qui language translation en to es	task1425 task1429 task1433	country iso numeric evalution semantic nelation classification head qu language translation es to en	task1426 task143 task1434	constry independenc year one of man-out classification generate category head on classification
task1445 task1449 task1453	closest integers disease entity extraction before dataset person entity extraction bic corpus	task1446 task145 task146	Analysis of the general state of the state o	task1447 task1451 task147	drang to maintena drang estraction ade drang dose extraction afs argument similarity gay marriage	task1448 task1452 task1479	disease entity extraction nebi dataset location entity extraction the corpus organization entity extraction the corpus
trafe 1 39 trafe 1 40 trafe 1 42 trafe 1 43	ars argument quality gay marriage chemical extraction chemipot dataset organism substance extraction anem dataset bengali personal bate speech binary classification	task1480 task1484 task1488 task1491	gene extraction julpha dataset gene extraction lineaeux dataset sarcasmeletection headline classification benguli political bate speech binary classification	task1481 task1485 task1489 task1492	and it of conductors; and it is conductors;	maki 1993 maki 1999 maki 1403 maki 1409 maki 1419 maki 1422 maki 1426 maki 143 maki 1434 maki 1444 maki 1445 maki 1452 maki 1453 maki 1453 maki 1453 maki 1453	gene extraction contempted dataset cell extraction meern dataset afs argument quality elegath perulty bengin has the speech classification bengin has the speech classification
task1502 task1506	adverse drug event classification hatexplain classification celebrity minimal dob span tomos find leading easy class	task1496 task1503 task1507	agend with centred uniford work and of the control	mak1271 mak1273 mak1273 mak1285 mak1285 mak1380 mak1380 mak1381		task1505 task1509	afs irgument quality gan control could) serundac relation classification collation autoropyan
task1519 task153 task1538	qu'ut question generation tomps find location bad clean malayalam offenseval dravidian classification	task152 task1533 task1541	evalution relation extraction tomap find location casy noise duby dubby format classification with planty (auditation with planty (auditation play answer generation are less transferation	task1520 task1534 task1542	quart answer generation daily dialog question classification every ith element from starting	task1529 task1537 task1544	witall I. Estimistation tartial offeneroal drawdistant classification conditions canned critaly recognitions network generation
task1545 task1551 task1560	special processors of the processor of t	mai 279 1 mai 27	wiga binary classification jileg answer generation nest text modification promara question remention	task1517 task1530 task1534 task1542 task1549 task1549 task1554 task1558 task1574 task158 task158	larent calendarian yot el laware gamenta yot el laware gamenta wenyin helimat inne natuta helimat laware helimat laware helimat laware helimat laware perpara katindarian	task1519 task1529 task1537 task1544 task155 task156 task1565 task1575 task1575 task1580	summer out amount prosection where the control of t
task1566 task1572 task1576 task1581 task1585	samsum summury amazon reviews multi english language classification equic-perturbed answer generation resulf basererum	task1573 task1577 task1582	surnoum classification amazon reviews multi japanese language classification bless hypernym generation clock fromency of surefu	task1574 task158 task1583	armanon reviews multi language identification count frequency of words bless memory classification dislamars fort merenties	task1575 task1580 task1584	amanna reviews multi sentiment classification coase-perturbed quotions generation evalution ascerproy classification
task1581 task1585 task1593 task1598 task1601 task1601	yahoo answers topics classification nyc long text generation websquestions answer generation	task1594 task1599 task1602	and text modification propun question permittin propun question permittin auxuro reviews multi japanes laspuage clausification bless hypersup programies bless hypersup programies bless hypersup programs yahro autworn topics question promittion smcalibor chandication webspecies question promittion webspecies question promittion to the text clausification.	task1583 task1590 task1595 task160 task1603 task1607	—questions you governed event/amind text generation 1 replace letter in a sentence smealflow sentence generation	task 1584 task 1592 task 1596 task 1600 task 1604 task 1608	contributed test generation 2 secolifies sentence generation c then test classification
task1609 task1617	ethos text classification xquad en question generation ce alligned translate tel eng count words ending with letter	task1606 task161 task1618	ethos text classification count words containing letter ce alligned classify let eng	task1610 task162 task1645	ethos text classification yound or answer generation count words starting with letter count words starting with letter motival amountum any obtained text classification	tusk1622 tusk1622	squad en answer generation ce alligent translate eng tel died qui text modication died qui text modication died qui text modication died qui text modication
task1648 task1655 task1660		task1649 task1656 task1661 task167 task1678	opus books en-no translation geous answer generation super glue classification	task1657 task1662	opus books en-ex translation geous quastion generation cede ru classification	task166 task166	opus books cu-en translation claris suttence generation trainglecomp question generation
task1669 task1677 task1686 task1691 task1706	nikh translation super gleu question super gleu question pemeralion super gleu question pemeralion super gleu question superale-cu translation superale-cu translation superale-cu translation que aimana translation que aimana translation que aimana translation que aimana superale que que que que que que que que que qu	task1678 task1678 task1689 task1703 task171	strategycpa quevison generalisen markpa natwer selection qed amara translation lyspeech textmedification	task1670 task168 task169 task1704	ethen text classification with the control of the c	task1676 task1685 task1690 task1705 task1712	squad-ca translation meny-00th translation qod amaza translation jupeech e kusolisation
task1713 task1721	l'appecch classification convail senience generation civil comments obsecuity classification civil comments accordoncity classification	task1714 task1722	methy is avever edection gold attent translation gold attent translation gold translation con- gold translation con- gold translation con- gold translation con- constant increase generation coli comments threat classification coli comments threat classification coli comments threat classification coli control translation gold t	task169 task1704 task1711 task172 task1723 task1727	poki text generation pol translation en fa civil comments sexually explicit classification with what is the effect	task1720 task1724	pola classification civil comments toucity classification civil comments insult classification civil comments insult classification with all white to test
task1729 task176	break decompose questions intervention extraction	task1731 task177	quartz question amovering pora-runt paraphrasing onicerne extraction	task174 task178	spl translation on ja quartz question answering thytne generation	task175 task179	spli translation on pl participant extraction break generate question
task190 task201 task207 task211	untervention consection sulficialismination must leavification max element lists looic Zirat classification	task192 task202 task208 task219		task195 task205 task209 task022	sentament 49 classification remove even elements stancedefection classification controlana maxima imperiorinal binary	task196 task206 task210 task223	sentimental 440 annover generation collists conjecture less the conjecture less than the conjecture conjecture conjecture conjecture conjecture less than the conjecture le
tride211 tride227 tride024 tride360 tride360 tride257 tride257 tride253 tride260 tride274 tride274 tride260 tride260 tri	and southern and a second a second and a second a second and a second	mic219 mic248 mic248 mic248 mic248 mic248 mic249 mic251 mic250 mic255 mic255 mic279 mic229 mic285 mic291 mic286 mic291 mic286 mic291 mic286 mic291 mic286 mic291 mic286 mic310 mi	and contraction for the institution and contraction for the institution of the institutio	mik022 mik229 mik224 mik229 mik244 mik229 mik244 mik259 mik254 mik256 mik256 mik266 mi	names over determine control process of the c	mai/223 mi/245 mi/245 mi/245 mi/245 mi/255 mi/255 mi/256 mi/256 mi/257 mi/257 mi/258 mi/273 mi/274 m	strangener programmer god inner translation
task257 task263 task269	spl translation ar on spl translation pl on one counterfactual story generation	task260 task265 task270	not translation the or spl translation the paper reviews language identification cay counterfactual context generation	task261 task266 task272	nyl translation or to nyl translation or to paper reviews neviewer perspective classification europaid translation	tusk262 tusk267 tusk273	spl translation is en spl translation is een concatenate and reverse all elements from index i to j curepart classification
task274 task278 task284 task290	overraling legal classification stereoset sentence generation antistereotype imdb classification tellmostry question answerability	task275 task279 task285 task291	enhanced wee paraphrase generation stereoset classification stereotype irrab answer generation serroval 2020 back commenceme validation	task276 task280 task286 task292	enhanced was classification storoosed classification storootype type old offense judgment storycommonscence character text someration	tusk277 tusk283 tusk288 tusk293	statement sentence penerations statemetry e deram incorrect aniswer generationed grigament grid of the statement of grigament summanisation statement grid of the statement of the statement of the statement of the statement of the statement
task294 task298 task305	storycommonsense motiv text generation storycloze correct end classification jeopandy answer generation normal	task295 task299 task306	sernoval 2020 bask4 commonstens reasoning stocyclosus serfactors generation journey answer generation double	task296 task300 task307	storycloze correct end classification storycloze coder generation jeepandy answer generation final	task297 task304 task308	sterýclose incorrect end classificiálem numeric (mod band reschiálem jeopardy answer generation all
task318 task322 task326	stereoset classification gender jigsaw classification theat jigsaw classification obscene	task319 task323 task327	stereoset classification profession jigsaw classification secually explicit jigsaw classification secually explicit jigsaw classification toxic	task320 task324 task328	steronoset classification race jipurw classification disagree jipurw classification insuk jipurw classification insuk	tusk321 tusk325 tusk329	tions speak to the form and the street in th
task033 task335 task340 task344 task035	wanogrande antwer generation hatereal classification aggresive en winornt classification gender pro hybridea answer reneration	task336 task341 task345	gap answer generation haleeval classification aggressive es winsom classification gender anti hybridus answer meneration	task333 task337 task342 task346	hateeval classification bate on hateeval classification individual on winomit classification profession pro hybridga classification profession	task334 task034 task343 task347	haloreni ekasaheziota hale es winogrande questan modificazion object winogrande questan modificazion object winosti classificazion profession auti h brichada incorrect answer enerazion for the control of the control
task035 task354 task358 task364	winogrande question modification person casino classification negotiation no need casino classification negotiation uv part	task350 task355 task359	wimont classification gender identifiability pro casino classification negotiation other need casino classification negotiation vosch fair	task351 task356 task362 task366	winous classification gender identifiability anti- casino classification negotiation self need spelin yenaral prompt response sub classification	task353 task357 task363 task367	cisino classification negotifion elicit pref cusino classification negotifion estull talk sst2 polarity classification synthetic remove flush synthetic remove flush
task368 task372 task376	synthetic even or odd calculation synthetic palindrome numbers reverse order of words	task369 task373 task377	worster consideration geniary acertimatesty peo cation classification negotiation order need symbolic remove content of the content of the symbolic remove colds symbolic remove colds symbolic remove colds symbolic remove colds symbolic remove colds symbolic remove colds symbolic remove colds	task370 task374 tssk378	synthetic remove divisible by 3 synthetic per or ng calculation reverse words of given length	task371 task375	symbolic product of first symbolic product of first classify type of sentence in debate agrees to type, classification
task380 task384 task388 task391 task399	reverse orace or weens bood by use no question socializar question classification socializar question classification causal relationship semental 2018 task! tweet sadness detection mickey it sentence perturbation generation	task381 task385 task389 task393	henines works of poers aregun boulq quantum generation boulq quantum generation korque generate temporal quantion plausable round generation purso paraphrase classification miskey at sentence perturbation generation	task382 task386 task039 task397 task403	que find overlapping words	task383 task387 task390 task398 task406	syndhesi tentrer laine. Leading for effective and the control of
task399 task408 task413 task417	semeral 2018 task1 tweet sudness detection mickey it sentence perturbation generation mickey on sentence perturbation generation mickey or sentence perturbation generation	task239 task490 task490 task414 task424 task429 task434 task438	panos paraphrase e lausification miskey at sentence perturbation generation miskey at sentence perturbation generation history and sentence perturbation generation	task403 task410 task415 task425 task043	words, commonwearts inferenced words on commonwearts inferenced words on the commonweart inferenced was sufficient to the commonwearts of the comm	task411 task416 task426	mickey fr sentence perturbation generation mickey vi sentence perturbation generation mickey de sentence perturbation generation
task428 task431	senteral inversion senteral object count alt en ja answer generation	task429 task434 task438	senteval tense alt on hi answer generation ong guj parallel corpus on gu translation	task435	exsential terms answering incomplete questions alt on ju translation ong gpi parallel corpus gu on translation	task430 task436	mikely de sentence penturbuluni guneration hindierigishe (verse lès en classification santesi adaptet const excessi adaptet const excessità l'armis distripti generatia words
task440 task448 task452 task456	ong guj parallel corpus gu-en classification opus paracrawl en il translation opus paracrawl en ig translation matres intention classification	task441 task449 task453 task457	ong gup parallel corpus gu-on langsunge identification opus paracrawl ig en translation wag answer generation matrex conditional classification	task442 task045 task454 task458	cen qa paraphrase question generation miscellaneous serience pumphrasing swag incorrect answer generation matres negation classification	task446 task450 task455 task455	open paracrami en so translation open paracrami lo en translation swag coefect generation marken statie ekossification
task456 task046 task047 task477 task490 task494	scitorio interzioni dali caj asserver possezioni dali caj asserver possezioni cen gal prasileli corpus que ci lessificazioni cen gal prasileli corpus que ci lessificazioni cen gal prasileli corpus que consistente materia interinati chicolicazioni materia interinati chicolicazioni materia interinati chicolicazioni materia consistenti generali consistenti materia consistenti generali consistenti materia copioni protessioni retrivero polissi protessioni retrivero polissi protessioni positioni coi dil alphabetical demensa ini to promete i sinno chestificationi materiali consistenti consistenti consistenti con promete i sinno chestificationi materiali consistenti consistenti con materiali consistenti consistenti con materiali con mat	tmk457 tmk460 tmk471 tmk483 tmk491 tmk495	makey is electrone production generation makey as universe production generation to the control of the contro	tmk458 tmk461 tmk472 tmk488 tmk492 tmk496	quoper question generation haspart classification extract all alphabetical elements from list in order move incorrect answer generation	task459 task462 task462 task475 task489 task493 task497	contain them to destribe queented works only per personnel on the traditions makes the traditions makes the traditions makes to the destributions makes to the destributions play play played post contained play play play to the traditions makes queen generalies
task494 task499 task506 task513	review polarity answer generation extract and add all numbers from list position of all alphabetical elements in list	task495 task050 task507	position of all numerical elements in list	task496 task504 task509	semeval answer generation count all alphabetical elements in list collate of all alphabetical and numerical elements in list separately	task497 task505 task512	
task517	position of all alphabetical elements in list argument stance classification ento classify emetien of dialogue curopart en es translation alt translation is en alt translation is en alt translation is en alt translation en th	task/95 task/97 task/97 task/98 task/98 task/98 task/97 task/98 task/98	argument consequence classification emo different dialogue emotions exerceparl ex en translation alt translations the em alt translations the em alt translations en emit.	task521 task532 task539	spirity ejection journame centra; all allybridged elements from list in order activa; all allybridged elements from list in order activa; all allybridged elements in list separately activa; all ally activates activates activates activates and activates	task523 task533 task541	name a conjugate an official and a more in last current and a current an
task542 task551 task564 task568	an translation ja en alt translation en th discofuse classification circa question generation	task543 task560 task565 task573	au translation bh en alt translation en enk circa answer generation air dialogue classification	task547 task561 task566 task574	ast translation entit en alt translation en by circa classification air dialogue sentence generation	task550 task563 task567 task575	uscomes entenze generalnos discofuse amore generalnos discofuse amore generalnos circa text generalnos air dalospor elessóriation
trade 55 tra	discotine classification curva question general presenta que transparente permentina succiding questione percentina succiding questione percentina proposition classification manuscript question proposition classification manuscript question percentina procedu question percentina percen	tmk585 tmk573 tmk577 tmk585 tmk586 tmk059 tmk596 tmk600 tmk600 tmk610 tmk617 tmk620 tmk627	all translations much consistent and translations are consistent as a construction of the construction of	traik504 traik509 traik519 traik515 traik521 traik521 traik539 traik539 traik547 traik561 traik561 traik574 traik574 traik579 traik583 traik590 traik594 traik690 traik594 traik691 traik601 traik601 traik601	all manifolium eith en international either international either art delinging sentence generation art delinging sentence generation art delinging sentence generation authority organization authority organization authority organization authority organization authority organization also generation al	mai-596 mai-512 mai-512 mai-512 mai-512 mai-512 mai-513 mai-513 mai-523 mai-523 mai-5241 mai-528 mai-5	Twice creations described in the control of the con
task592 task596 task605	sciq incorrect answer generation mecha question generation find the longost common subsequence in two lists	task593 task600 task606	sciq explanation generation find the longest common substring in two strings sum of all numbers in list between positions i and j	task594 task601 task607	sciq question generation flores translation selocu shic intentional offense birary classification	task595 task604 task608	mocha anowe generation flores translation entons slice sexual offense binary classification
task616 task062	powersamy washed through custofication cola classification bigbench repeat copy logic thric sentence based on given word sentence generation	task617 task620 task627	amazonerview category lext generation obsurned medical subject headings answer generation xlwic word with same meaning sentence generation	task618 task622 task628	replace alphabets in a list by their position in english alphabet	task619 task625 task629	chosered a blaste title generation whose true or false answer generation whose true or false answer generation depends 14 clossification depends 14 clossification
task683 task633 task639	dbpedia 14 answer generation	task630 task636	depends 14 classification extract and seet unique alphabets in a list all elements except first i refresd classification	task631 task637 task640	dispediu 14 incorrect answer generation extract and sort unique digits in a list coeli classification	task638 task641	dispedia 14 cinositucation multi vor classification cutil classification cutil classification autores rementation
task642 task665 task664 task6660	munt wor user unerance generation earli classification timetravel consistent sentence classification bible fa en translation mizan fa en translation	tak643 tak651 tak657 tak662	refresd classification opus 100 en ar translation quant fix or translation global voices fix en translation	task645 task652 task659 task664	surremerization parsiritu en fa translation tep fa en translation rummiu assewer generation abstract algebra	task648 task653 task066 task665	
task660 task666 task671 task675 task681	ambigna text generation astronomy ambigna text generation geogle welformed query sentence generation lope edi malayalam text classification	task662 task667 task672 task679 task682	numerous answer generation business ethics numerousness hope eth english text classification online privacy policy text classification	task664 task667 task673 task668 task683	assactiventi answer generation georgie wellformed query classification abductivental incorrect answer generation online privacy policy text purpose answer generation	task665 task670 task674 task680 task684	marropos speciales (Binterialistis) geogle selfilemente query sententece generation hope edi tamil lest classification oline provey polse jest information type generation
task685 task689 task693 task697	bible f. on translation mixtum for translation mixtum for translation and translation substitution of translation substitution of translation substitution of translation substitution of translation substitution substitution substitution mixtum for any exercision conceptual physics mixtum for any exercision conceptual physics mixtum for any exercision of substitution substitution of translation mixtum for any exercision conceptual physics mixtum for any exercision for mixtum mixtum for any exercision for mixtum for any exercision for any ex	task686 task069	open 100 are translation plant by the continuation of the continua	task687 task691	personne on the transferent control of the control	task688 task692 task696 task070 task704 task708	immerior langer (maniferer) clauselation analyting quasing memoriant perpendiant personal personal person
task700 task705 task709	mmenta answer generation torna togic mmenta answer generation high school chemistry mmenta answer generation high school macroeconomics mmenta answer generation high school psychology	tak698 tak701 tak706 tak071	manulu answer generation high school computer science manulu answer generation high school mathematics abductiventi answer generation	task699 task703 task707 task710 task716 task072	manula answer generation high school geography manula answer generation high school microeconomics manula answer generation high school statistics	task704 task708 task713 task717 task720	shabot only informed catalogists must be a series of the catalogists must be a series processed by the class of prices must be a series prices of the catalogists must be a series of the catalogist
task714 task718 task721 task725 task073	mmentu answer generation human sexuality mmentu answer generation machine learning mmentu answer generation medical genetics mmentu answer generation nutrition	task701 task715 task719 task722 task726 task731	mmenus answer generation international law mmenu naswer generation management mmenus naswer generation random topic mmenus naswer generation phikosophy	task716 task072 task723 task727 task732	mmmu answer generation jurisprudence abductiventi answer generation mmmlu answer generation moral disputes mmmlu answer generation prehistory	task717 task720 task724 task728 task733	mmuta answer generation logical fallicies mutalia answer generation materialenting mutalia answer generation metal securion mutalia answer generation metal securion mutalia answer generation profesional accounting
task734 task739	communicated answer generation minute answer generation sociology thorses question generation thorses answer remember from	task735	mmmlu answer generation professional psychology mmmlu answer generation us foreign policy squad 1.1 question generation at 2 arithmetic custoines arithmetic	task736 task740	manufacture protection policies of stations manufacture protection virology books) answer generation virology books) answer generation quantity yelp restaurant review classification	task737 task741 task075	muntha assver gionentino incurity statales muntha assvery generation world religious boots) arower generation world religious seazad 1.1 amore encerations
task/50 task/50 task/54 task/61 task/077 task/774 task/778	aqua multiple cheice answering syamp common-division quastion answering app review classification	task751 task755 task762	avamp subtraction question answering find longest substring and replace its screed lowercase version in both lists ernea if sk translation	task740 task756 task764 task771 task776 task078	ssamp makiplication question answering find longert substring and return all unique alphabets in it emen bg el classification	task753	warm addition question answering sphale correcting safe instalke qod sammarization
task077 task774 task778 task785	spiash explanation to sel passox german text modification passox english french translation passox korean spanish translation	task770 task775 task779 task787	purex english text modification purex chance text modification purex english spanish translation purex korean chance translation	task771 task776 task078 task788	pawax kecean text modification pawax japanese text modification all elements except last i pawax kecean japanese translation	task772 task777 task784 task789	parox resear local modifications parox regular locas attendation parox locas french translation parox locas french translation parox french anglish translation
task079 task795 task080	contain contrast strings passex spanish english translation piga answer generation	task790 task796 task802	punwx feench kerean translation punwx spanish kerean translation punwx german kerean translation punwx feenan kerean translation	task792 task797 task803	pawax french german translation pawax spanish french translation pawax german french translation pawax delimen french translation	task794 task799 task804	pawax ferech jujunese translation pawax spatish chience translation pawax gentaus quantido translation
trade 7 85 trade 799 trade 799 trade 799 trade 799 trade 809	amende novem generation (round lage) and continued and con	mm1/25 mm	mentals was one generation enable to the committee of the	mat/788 mat/792 mat/997 mat/993 mat/997 mat/993 mat/997 mat/993 mat/813 mat/819 mat/813 mat/813 mat/823 mat/824 mat/825 mat/826 mat/826 mat/827 mat/828 mat/897 mat/828	Boood, some flementen quantify and the properties of the properti	maid:7:00 maid:7	winger addition quident automatique of the control
task083 task833 task839	babi tl single supporting fact answer generation poem sentiment classification oft classification financial phraschapt classification	tmk830 tmk834 tmk084 pul-84*	poleval2019 mt translation mathdataset classification bobi t1 single supporting fact identify relevant fact francial obrasebank classification	task831 task835 task840 task840	giga fron classification muthalaset answer generation para plt on ex translation nubratedos classification	task832 task838 task841 task841	polesi 2019 pet classification cld classification para pil de cu translation para pil de cu translation para pil de cu translation
task850 task857 task860	synthetic longest palindrome inquisitive question generation prost med generation	task851 task858 task861	synthetic multiply evens manusitive span detection prost meq answers generation	task852 task859 task862	synthetic multiply odds prost question generation audiv multidiv question answering	task856 task866 task863	conv ai 2 classification translated symbol aribmetic and/or mathic spacetion answering
task864 task868 task874 task879	aware sangleop question answering mawps sangleop question answering opus shosanasy se schema guided dole8 classification	task865 task087 task875 task088	mawps addusb question answering new operator addusb arithmetic emotion classification admits/ typo verification	task866 task872 task877 task888	mawps multidiv question answering opus xhouarasy translation eng xhoua kdo4 translation reviews classification	task873 task878 task889	manys mantep question artewering open shouseney standaion shous eng kded translation georentionse classification
task089 task893 task899 task902 task908	swap words verification gap fill the blank coreference resolution freebase qu topic generation describes correspondent to the control of the core	task890 task896 task990	gewd classification man language classification equation learner algebra depositive regimen segme classification	task891 task897 task900 task904 task091	gap coreference resolution freebase qu topic question generation freebase qu category classification but smooth offermate classification	task892 task898 task901 task905 task910	pap reverse conference resolutions freshessa qua asserge generations forebress qua subgrey quantities generations forebress quantities of conference of the conference of the based chandidation of chandidations
task908 task911 task922 task926	dialogre identify familial relationships bianet translation event Zmind word generation	task909 task913 task923	dialogre prevalent speakers bianet translation event2mind classifier	task991 task914 task924 task928	all elements from index i to j biaset translation event2mind word generation	task910 task992 task925	
task093 task094 task096	statistics technically manusar terturoscurps construction of generation conclud come perf word generation consults occuration in consultation of the consultation of the consultation of the consultation of the librispeche are test auto completion consultation of the consultation of the plus testations or with material	task933 task995 task961	you magazine to promote alytic transfer with auto style transfer contain max absolute value ancora-ca-ner text auto completion	task955 task962	all elements from index is 10 busine translation in 10 busine translati	task992 task925 task929 task936 task956 task963 task967 task969	institute colorisational makes colorisation consideration per disconsideration personal control and colorisation personal control and colorisation personal control and colorisation personal colorisation provided personal licençation in ment to seed production licençation in ment to seed production personal colorisation personal colorisation personal colorisation leaders (and a licenside personal colo
task964 task997 task990	Isbrispeech asr text auto completion conala remove duplicates pib translation urda marathi	task965 task998 task991	hbrespeech are missing word prediction constal loi intersection pib translation english tural	task966 task982 task995	nuletaker fact checking based on given context pib translation tarnil bengali pib translation bengali english	task967 task999 task996	ruletaker incorrect fact generation based on given paragraph reverse elements between index i and j pils translation english bengan

Table 4: Results for Different Numbers of Clusters (different LoRA-ranks)

Metric	Uncompressed	1 Cluster	2 Clusters	4 Clusters	8 Clusters
Loss	0.5009	1.6023	0.9986	0.5603	0.5000
Exact match	69.6103	37.3392	60.1071	66.6474	69.6289
ROUGE-1	79.5755	50.3598	71.5644	76.2930	79.0576
ROUGE-L	78.9355	49.7491	71.0193	75.6152	78.3958
Recon. error	0	0.8311	0.6990	0.4846	0.3246
Agreement	1.0	36.6706	62.6699	72.2231	80.5545
Exact match ratio	1.0	0.5724	0.8079	0.9219	0.9541
Relative ROUGE-1	1.0	0.6220	0.8533	0.9406	0.9917
Relative ROUGE-L	1.0	0.6221	0.8550	0.9404	0.9915
Param. saved ratio	1.0	0.99	0.98	0.97	0.94

Table 5: Results for Different Numbers of Clusters (same LoRA-ranks of 43)

Metric	Uncompressed	1 Cluster	2 Clusters	4 Clusters	8 Clusters
Loss	0.5764	0.7767	0.6277	0.5841	0.5659
Exact match	61.7746	53.2000	60.2000	61.5000	61.3000
ROUGE-1	79.9322	74.7695	79.4621	80.6950	80.4298
ROUGE-L	77.7961	72.4257	77.2141	78.3369	78.1883
Recon. error	0	0.7333	0.5594	0.3999	0.2502
Agreement	1.0	0.0000	6.6667	6.6667	6.6667
Exact match ratio	1.0	0.8175	0.9261	0.9618	1.0206
Relative ROUGE-1	1.0	0.9560	1.0153	1.0310	1.0241
Relative ROUGE-L	1.0	0.9501	1.0118	1.0268	1.0231
Param. saved ratio	1.0	0.99	0.98	0.97	0.94

Model Type	Method Type					Ta	sks					. Average	Para. Saved
Woder Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	raia. Saveu
	base lora	$\begin{array}{c} 0.26 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.19 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c }\hline 0.42 \pm 0.00 \\ 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 0.11 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c c} 0.11 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.19 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.28 \pm 0.21 \\ 1.00 \pm 0.00 \end{array}$	1.00/1.00 0.00/0.00
TIES	10 50 100 500	$\begin{array}{c} 0.81 \pm 0.00 \\ 0.59 \pm 0.00 \\ 0.55 \pm 0.00 \\ 0.37 \pm 0.00 \end{array}$	$\begin{array}{c} 0.57 \pm 0.02 \\ 0.41 \pm 0.00 \\ 0.40 \pm 0.00 \\ 0.26 \pm 0.00 \end{array}$	$\begin{array}{c} 0.45 \pm 0.04 \\ 0.18 \pm 0.05 \\ 0.20 \pm 0.05 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.10 \pm 0.01 \\ 0.03 \pm 0.01 \\ 0.01 \pm 0.02 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.91 \pm 0.01 \\ 0.88 \pm 0.00 \\ 0.83 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.47 \pm 0.00 \\ 0.31 \pm 0.00 \\ 0.33 \pm 0.00 \\ 0.29 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.69 \pm 0.01 \\ 0.65 \pm 0.00 \\ 0.64 \pm 0.00 \\ 0.57 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.57 \pm 0.00 \\ 0.62 \pm 0.00 \\ 0.57 \pm 0.02 \\ 0.37 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.82 \pm 0.01 \\ 0.32 \pm 0.04 \\ 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.85 \pm 0.00 \\ 0.84 \pm 0.00 \\ 0.82 \pm 0.00 \\ 0.43 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.62 \pm 0.23 \\ 0.48 \pm 0.28 \\ 0.44 \pm 0.30 \\ 0.31 \pm 0.26 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 1.00 1.00 / 1.00
SVD	SVD 2 SVD 4 SVD 8 SVD 16	$\begin{array}{c} 0.98 \pm 0.03 \\ 0.99 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.07 \pm 0.02 \\ 1.04 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.98 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 0.99 \pm 0.02 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.00 \pm 0.10 \\ 0.99 \pm 0.08 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.04 \\ 1.00 \pm 0.03 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	0.88 / 0.88 0.75 / 0.75 0.50 / 0.50 0.00 / 0.00
10 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 1.02 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.00 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.99 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.96 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.02 \pm 0.02 \\ 0.97 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.13 \pm 0.03 \\ 1.05 \pm 0.03 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.99 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.98 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.01 \pm 0.05 \\ 1.00 \pm 0.03 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	1.00 / 0.90 1.00 / 0.80 1.00 / 0.60 1.00 / 0.20 1.00 / -0.60
10 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.06 \pm 0.01 \\ 1.04 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 0.99 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.98 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.02 \\ 0.96 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.07 \pm 0.00 \\ 1.00 \pm 0.02 \\ 0.98 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.03 \\ 1.00 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	1.00/0.90 0.99/0.79 0.97/0.57 0.88/0.07 0.50/-1.10
50 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.98 \pm 0.04 \\ 1.00 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.98 \pm 0.01 \\ 1.02 \pm 0.02 \\ 1.05 \pm 0.02 \\ 1.08 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.92 \pm 0.06 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.84 \pm 0.07 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.92 \pm 0.02 \\ 0.95 \pm 0.02 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.68 \pm 0.05 \\ 0.84 \pm 0.02 \\ 0.99 \pm 0.01 \\ 0.98 \pm 0.01 \\ 0.97 \pm 0.03 \end{array}$	$ \begin{vmatrix} 0.87 \pm 0.10 \\ 1.00 \pm 0.13 \\ 1.09 \pm 0.03 \\ 1.11 \pm 0.03 \\ 1.01 \pm 0.03 \end{vmatrix} $	$\begin{array}{c} 0.88 \pm 0.07 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.83 \pm 0.02 \\ 0.88 \pm 0.01 \\ 0.90 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.89 \pm 0.10 \\ 0.96 \pm 0.07 \\ 1.00 \pm 0.05 \\ 1.01 \pm 0.04 \\ 1.00 \pm 0.02 \end{array}$	1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84 1.00 / 0.68
50 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.99 \pm 0.04 \\ 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.02 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.06 \pm 0.02 \\ 1.06 \pm 0.01 \\ 1.02 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.95 \pm 0.02 \\ 0.98 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.94 \pm 0.01 \\ 0.96 \pm 0.00 \\ 0.98 \pm 0.01 \\ 0.98 \pm 0.00 \\ 0.99 \pm 0.00 \end{array}$	$\begin{array}{c} 0.64 \pm 0.10 \\ 0.95 \pm 0.01 \\ 1.03 \pm 0.01 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$ \begin{vmatrix} 1.01 \pm 0.15 \\ 1.09 \pm 0.02 \\ 1.11 \pm 0.00 \\ 1.03 \pm 0.04 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.97 \pm 0.02 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.87 \pm 0.00 \\ 0.89 \pm 0.01 \\ 0.98 \pm 0.02 \\ 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.12 \\ 0.99 \pm 0.05 \\ 1.02 \pm 0.04 \\ 1.01 \pm 0.03 \\ 1.00 \pm 0.01 \end{array}$	1.00/0.98 0.99/0.95 0.97/0.89 0.88/0.72 0.50/0.18
100 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.80 \pm 0.07 \\ 0.95 \pm 0.06 \\ 1.01 \pm 0.03 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.06 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.06 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{array}{c} 0.96 \pm 0.01 \\ 0.91 \pm 0.06 \\ 0.98 \pm 0.02 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array} $	$\begin{array}{c} 0.50 \pm 0.09 \\ 0.80 \pm 0.14 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.00 \end{array}$	$ \begin{array}{c} 0.78 \pm 0.01 \\ 0.89 \pm 0.06 \\ 0.94 \pm 0.01 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \end{array} $	$\begin{array}{c} 0.28 \pm 0.07 \\ 0.60 \pm 0.10 \\ 0.88 \pm 0.05 \\ 1.00 \pm 0.03 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.52 \pm 0.10 \\ 0.77 \pm 0.26 \\ 1.11 \pm 0.08 \\ 1.11 \pm 0.02 \\ 1.11 \pm 0.03 \end{array}$	$\begin{array}{c} 0.78 \pm 0.03 \\ 0.91 \pm 0.02 \\ 0.96 \pm 0.02 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{array}{c} 0.81 \pm 0.02 \\ 0.83 \pm 0.02 \\ 0.87 \pm 0.03 \\ 0.89 \pm 0.02 \\ 0.98 \pm 0.01 \end{array} $	$\begin{array}{c} 0.72 \pm 0.22 \\ 0.86 \pm 0.14 \\ 0.97 \pm 0.07 \\ 1.00 \pm 0.05 \\ 1.01 \pm 0.04 \end{array}$	1.00 / 0.99 1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84
100 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.95 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.01 \\ 0.97 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.92 \pm 0.01 \\ 0.95 \pm 0.00 \\ 0.96 \pm 0.00 \\ 0.98 \pm 0.00 \\ 0.99 \pm 0.00 \end{array}$	$\begin{array}{c} 0.64 \pm 0.03 \\ 0.86 \pm 0.03 \\ 0.99 \pm 0.01 \\ 1.03 \pm 0.01 \\ 0.98 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.16 \\ 1.12 \pm 0.03 \\ 1.09 \pm 0.01 \\ 1.10 \pm 0.01 \\ 1.00 \pm 0.03 \end{array}$	$\begin{array}{c} 0.87 \pm 0.02 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.02 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.87 \pm 0.00 \\ 0.89 \pm 0.00 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.11 \\ 0.97 \pm 0.07 \\ 0.99 \pm 0.05 \\ 1.02 \pm 0.04 \\ 1.00 \pm 0.01 \end{array}$	1.00/0.99 0.99/0.97 0.97/0.93 0.88/0.80 0.50/0.34
100 w/clusters (C)	16 C 5 16 C 7	1.13 ± 0.01 1.12 ± 0.01	1.03 ± 0.01 1.01 ± 0.01	1.00 ± 0.00 1.00 ± 0.00	1.00 ± 0.00 1.00 ± 0.00	0.99 ± 0.01 0.99 ± 0.01	0.96 ± 0.00 0.96 ± 0.01	1.01 ± 0.02 1.02 ± 0.02	1.23 ± 0.02 1.24 ± 0.05	1.05 ± 0.01 1.03 ± 0.01	0.99 ± 0.06 0.99 ± 0.05	1.04 ± 0.08 1.04 ± 0.08	1.00/0.95 1.00/0.93
500 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.57 \pm 0.07 \\ 0.61 \pm 0.12 \\ 0.73 \pm 0.02 \\ 0.84 \pm 0.00 \\ 0.99 \pm 0.03 \end{array}$	$ \begin{vmatrix} 0.55 \pm 0.03 \\ 0.55 \pm 0.08 \\ 0.63 \pm 0.11 \\ 0.92 \pm 0.02 \\ 0.99 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.04 \\ 0.83 \pm 0.02 \\ 0.89 \pm 0.04 \\ 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.78 \pm 0.16 \\ 0.84 \pm 0.12 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.85 \pm 0.04 \\ 0.91 \pm 0.02 \\ 0.94 \pm 0.00 \\ 0.94 \pm 0.00 \\ 0.96 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.68 \pm 0.07 \\ 0.71 \pm 0.05 \\ 0.83 \pm 0.05 \\ 0.88 \pm 0.02 \\ 0.92 \pm 0.03 \end{vmatrix} $	$\begin{array}{c} 0.24 \pm 0.01 \\ 0.29 \pm 0.05 \\ 0.45 \pm 0.09 \\ 0.60 \pm 0.15 \\ 0.66 \pm 0.06 \end{array}$	$ \begin{vmatrix} 0.43 \pm 0.01 \\ 0.47 \pm 0.08 \\ 0.50 \pm 0.07 \\ 0.53 \pm 0.01 \\ 0.84 \pm 0.14 \end{vmatrix} $	$\begin{array}{c} 0.76 \pm 0.06 \\ 0.79 \pm 0.04 \\ 0.82 \pm 0.02 \\ 0.85 \pm 0.05 \\ 0.92 \pm 0.02 \end{array}$	$ \begin{vmatrix} 0.79 \pm 0.01 \\ 0.79 \pm 0.01 \\ 0.80 \pm 0.02 \\ 0.80 \pm 0.02 \\ 0.84 \pm 0.01 \end{vmatrix} $	$ \begin{array}{c} 0.65 \pm 0.20 \\ 0.68 \pm 0.20 \\ 0.76 \pm 0.18 \\ 0.83 \pm 0.15 \\ 0.91 \pm 0.11 \end{array} $	1.00 / 1.00 1.00 / 1.00 1.00 / 0.99 1.00 / 0.98 1.00 / 0.97
500 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.57 \pm 0.01 \\ 0.79 \pm 0.05 \\ 1.02 \pm 0.00 \\ 1.03 \pm 0.01 \\ 1.03 \pm 0.00 \end{array}$	$\begin{array}{c} 0.43 \pm 0.07 \\ 0.54 \pm 0.04 \\ 0.96 \pm 0.01 \\ 0.97 \pm 0.02 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.78 \pm 0.01 \\ 0.93 \pm 0.02 \\ 0.94 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.00 \\ 0.97 \pm 0.00 \\ 0.96 \pm 0.00 \\ 0.98 \pm 0.00 \\ 0.99 \pm 0.01 \end{array}$	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.90 \pm 0.01 \\ 0.97 \pm 0.01 \\ 0.96 \pm 0.00 \\ 0.97 \pm 0.01 \end{array}$	$\begin{array}{c} 0.64 \pm 0.00 \\ 0.69 \pm 0.01 \\ 0.73 \pm 0.01 \\ 0.87 \pm 0.01 \\ 0.99 \pm 0.02 \end{array}$	$\begin{array}{c} 0.53 \pm 0.03 \\ 0.50 \pm 0.00 \\ 0.54 \pm 0.01 \\ 1.07 \pm 0.02 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.86 \pm 0.02 \\ 0.91 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.83 \pm 0.00 \\ 0.83 \pm 0.01 \\ 0.86 \pm 0.00 \\ 0.87 \pm 0.00 \\ 0.87 \pm 0.00 \end{array}$	$\begin{array}{c} 0.75 \pm 0.17 \\ 0.81 \pm 0.16 \\ 0.89 \pm 0.14 \\ 0.97 \pm 0.06 \\ 0.99 \pm 0.05 \end{array}$	1.00/1.00 0.99/0.99 0.97/0.96 0.88/0.86 0.50/0.47
500 w/clusters (C)	16 C 7 16 C 10 16 C 25 64 C 5 64 C 7	1.09 1.10 1.10 1.09 1.12	1.00 1.01 1.00 0.98 1.02	0.99 1.00 1.00 1.00 1.00	1.00 0.99 0.99 1.00 1.00	0.98 0.97 0.99 0.99 1.00	0.95 0.93 0.96 0.96 0.96	0.72 0.70 0.98 0.99 0.99	0.87 1.30 1.31 1.18 1.22	0.98 1.02 1.03 1.04 1.04	0.90 0.88 0.91 0.87 0.93	0.95 0.99 1.03 1.01 1.03	1.00/0.98 1.00/0.98 1.00/0.95 0.97/0.93 0.97/0.91
1000 w/clusters (C)	16 C 25	1.09	0.98	1.00	1.00	0.97	0.96	0.72	1.30	1.05	0.91	1.00	1.00/0.97

Table 7: Relative In-Distribution ROUGE-L scores for various tasks and methods

Model Type	Method Type					Tas	sks					Average	Para. Saved
Model Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	raia. Saveu
	base lora	$\begin{array}{c} 24.44 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c }\hline 1.60 \pm 0.00 \\ 86.00 \pm 0.00 \\ \end{array}$	19.13 ± 0.00 99.00 ± 0.00	39.22 ± 0.00 93.67 ± 0.00	$\begin{array}{c} 10.27 \pm 0.00 \\ 94.33 \pm 0.00 \end{array}$	$\begin{array}{c} 35.46 \pm 0.00 \\ 74.88 \pm 0.00 \end{array}$	$ 7.85 \pm 0.00 \\ 74.40 \pm 0.00 $	$\begin{array}{c} 6.22 \pm 0.00 \\ 26.68 \pm 0.00 \end{array}$	17.82 ± 0.00 95.00 ± 0.00	$\begin{array}{c} 38.87 \pm 0.00 \\ 50.32 \pm 0.00 \end{array}$	20.24 ± 13.27 78.87 ± 22.56	1.00/1.00 0.00/0.00
	10 50	76.50 ± 0.00 55.80 ± 0.00	49.00 ± 1.73 35.00 ± 0.00	44.33 ± 4.04 18.00 ± 5.20	9.80 ± 0.58 2.42 ± 0.50	78.56 ± 0.96 85.78 ± 0.96	35.24 ± 0.00 23.03 ± 0.00	51.37 ± 0.67 48.03 ± 0.00	15.26 ± 0.12 16.50 ± 0.00	77.67 ± 1.15 30.00 ± 3.46	42.72 ± 0.01 42.47 ± 0.02	48.05 ± 23.61 35.70 ± 23.01	1.00 / 1.00
TIES	100	52.43 ± 0.00	34.00 ± 0.00	19.67 ± 4.62	1.09 ± 1.66	83.33 ± 0.00	24.89 ± 0.00	47.52 ± 0.00	15.18 ± 0.42	1.00 ± 0.00	41.19 ± 0.03	32.03 ± 24.50	1.00 / 1.00
	500	35.18 ± 0.00	22.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	78.00 ± 0.00	21.46 ± 0.00	42.22 ± 0.04	9.93 ± 0.13	1.00 ± 0.00	21.50 ± 0.03	23.27 ± 23.64	1.00 / 1.00
	SVD 2 SVD 4	93.15 ± 2.77 94.01 ± 3.60	92.24 ± 1.85 89.21 ± 0.71	99.09 ± 0.18 99.05 ± 0.09	93.44 ± 0.14 93.65 ± 0.03	93.89 ± 0.35 94.66 ± 0.63	73.74 ± 0.51 74.89 ± 0.33	74.55 ± 0.98 73.61 ± 1.15	26.80 ± 2.79 26.34 ± 2.13	95.06 ± 1.35 93.98 ± 0.77	50.21 ± 0.44 50.47 ± 0.54	79.11 ± 22.72 78.90 ± 22.68	0.88 / 0.88
SVD	SVD 8	95.00 ± 0.00	87.40 ± 0.59	99.05 ± 0.09	93.65 ± 0.03	94.36 ± 0.38	74.58 ± 0.12	75.07 ± 0.00	26.71 ± 0.27	95.51 ± 1.09	50.89 ± 0.07	81.01 ± 21.74	0.50 / 0.50
	SVD 16	95.00 ± 0.00	86.00 ± 0.00	99.00 ± 0.00	93.67 ± 0.00	94.33 ± 0.00	74.90 ± 0.03	74.23 ± 0.18	26.68 ± 0.00	95.00 ± 0.00	50.30 ± 0.02	78.36 ± 22.97	0.00 / 0.00
	16 D 32 D	96.67 ± 0.58 95.67 ± 0.58	87.00 ± 1.00 90.00 ± 1.00	99.00 ± 0.00 99.00 ± 0.00	94.00 ± 0.67 93.00 ± 0.33	93.11 ± 0.38 94.89 ± 0.51	72.08 ± 0.06 73.86 ± 0.31	76.26 ± 1.19 71.92 ± 0.84	30.11 ± 0.79 27.89 ± 0.70	94.00 ± 1.73 94.67 ± 0.58	49.30 ± 0.46 50.36 ± 0.26	79.15 ± 22.18 79.13 ± 22.75	1.00 / 0.90 1.00 / 0.80
10 diagonal (D)	64 D	95.00 ± 0.00	88.33 ± 0.58	99.00 ± 0.00 99.00 ± 0.00	93.67 ± 0.00	94.78 ± 0.38	73.80 ± 0.31 74.61 ± 0.13	74.97 ± 0.84 74.97 ± 0.58	26.35 ± 0.70 26.35 ± 0.25	96.00 ± 0.00	50.99 ± 0.06	79.13 ± 22.73 79.37 ± 22.94	1.00 / 0.60
-	128 D 256 D	95.00 ± 0.00 95.00 ± 0.00	86.67 ± 0.58 86.00 ± 0.00	99.00 ± 0.00 99.00 ± 0.00	93.67 ± 0.00 93.67 ± 0.00	94.33 ± 0.00 94.33 ± 0.00	74.92 ± 0.13 74.88 ± 0.00	74.96 ± 0.51 74.40 ± 0.00	26.45 ± 0.23 26.68 ± 0.00	95.00 ± 0.00 95.00 ± 0.00	50.21 ± 0.12 50.27 ± 0.02	79.02 ± 22.84 78.92 ± 22.77	1.00 / 0.20 1.00 / -0.60
	16 F	97.00 ± 0.00	91.00 ± 1.00	99.00 ± 0.00	93.56 ± 0.19	93.56 ± 0.69	73.60 ± 0.36	74.40 ± 0.00	28.66 ± 0.03	96.00 ± 1.00	50.15 ± 0.20	79.75 ± 22.72	1.00/0.90
	32 F	96.67 ± 0.58	89.33 ± 0.58	99.00 ± 0.00	93.22 ± 0.19	94.44 ± 0.19	74.11 ± 0.19	71.74 ± 0.59	26.74 ± 0.50	94.67 ± 0.58	50.63 ± 0.24	79.06 ± 23.01	0.99/0.79
10 full (F)	64 F 128 F	95.00 ± 0.00 95.00 ± 0.00	88.67 ± 0.58 86.67 ± 0.58	99.00 ± 0.00 99.00 ± 0.00	93.67 ± 0.00 93.67 ± 0.00	94.56 ± 0.38 94.33 ± 0.00	74.56 ± 0.13 75.04 ± 0.03	75.47 ± 0.58 74.40 ± 0.00	26.26 ± 0.34 26.53 ± 0.13	96.00 ± 0.00 95.00 ± 0.00	50.89 ± 0.17 50.36 ± 0.03	79.41 ± 22.97 79.00 ± 22.81	0.97/0.57 0.88/0.07
	256 F	95.00 ± 0.00 95.00 ± 0.00	86.00 ± 0.00	99.00 ± 0.00	93.67 ± 0.00	94.33 ± 0.00 94.33 ± 0.00	74.90 ± 0.03	74.29 ± 0.19	26.68 ± 0.00	95.00 ± 0.00	50.30 ± 0.03	78.92 ± 22.77	0.50/-1.10
	16 D	92.76 ± 3.53	84.67 ± 1.15	99.00 ± 0.00	86.17 ± 5.81	79.68 ± 6.21	69.07 ± 1.54	50.65 ± 3.97	23.27 ± 2.60	83.90 ± 6.43	41.86 ± 0.96	71.10 ± 23.99	1.00 / 0.98
50 diagonal (D)	32 D 64 D	95.33 ± 2.08 97.00 ± 0.00	87.33 ± 2.08 90.33 ± 1.53	99.00 ± 0.00 99.00 ± 0.00	92.60 ± 0.29 93.78 ± 0.19	90.32 ± 1.04 93.00 ± 0.58	71.16 ± 1.47 72.37 ± 0.35	62.51 ± 1.64 73.39 ± 0.93	26.60 ± 3.54 29.06 ± 0.80	93.33 ± 1.15 95.67 ± 0.58	44.35 ± 0.41 45.43 ± 0.34	76.25 ± 23.81 78.90 ± 23.29	1.00 / 0.96 1.00 / 0.92
30 diagonai (D)	128 D	97.00 ± 0.00 96.33 ± 0.58	90.55 ± 1.53 92.67 ± 0.58	99.00 ± 0.00 99.00 ± 0.00	93.78 ± 0.19 93.56 ± 0.19	93.00 ± 0.58 93.00 ± 0.58	72.37 ± 0.35 73.32 ± 0.24	73.03 ± 0.93	29.06 ± 0.80 29.51 ± 0.93	95.07 ± 0.58 95.00 ± 0.00	43.43 ± 0.34 50.16 ± 0.74	78.90 ± 23.29 79.56 ± 22.51	1.00 / 0.92
	256 D	95.67 ± 0.58	88.33 ± 0.58	99.00 ± 0.00	93.56 ± 0.19	94.67 ± 0.67	74.82 ± 0.24	72.36 ± 2.07	26.90 ± 0.75	95.33 ± 0.58	50.73 ± 0.46	79.14 ± 22.90	1.00 / 0.68
	16 F 32 F	94.06 ± 3.54 97.00 ± 0.00	85.67 ± 1.15 85.67 ± 1.53	98.67 ± 0.58 99.00 ± 0.00	90.35 ± 1.37 93.67 ± 0.00	89.90 ± 1.91 92.22 ± 0.69	70.32 ± 0.66 71.88 ± 0.30	47.62 ± 7.28 71.01 ± 1.02	26.88 ± 3.96 29.07 ± 0.65	92.33 ± 1.53 95.67 ± 1.53	43.68 ± 0.24 44.97 ± 0.41	73.95 ± 24.73 78.02 ± 23.18	1.00/0.98 0.99/0.95
50 full (F)	64 F	96.67 ± 0.00	91.00 ± 2.00	99.00 ± 0.00 99.00 ± 0.00	93.56 ± 0.19	92.22 ± 0.69 93.22 ± 0.51	73.16 ± 0.30 73.16 ± 0.41	76.28 ± 0.51	29.67 ± 0.65 29.67 ± 0.12	95.07 ± 1.53 95.33 ± 0.58	49.31 ± 1.00	79.72 ± 22.50	0.99/0.93
` ′	128 F	97.00 ± 0.00	91.00 ± 1.00	99.00 ± 0.00	93.33 ± 0.00	94.11 ± 0.51	73.51 ± 0.23	73.17 ± 0.58	27.53 ± 1.12	95.00 ± 1.00	50.56 ± 0.06	79.42 ± 22.93	0.88/0.72
	256 F	95.00 ± 0.00	88.00 ± 0.00	99.00 ± 0.00	93.67 ± 0.00	94.44 ± 0.19	74.25 ± 0.21	74.97 ± 0.58	26.79 ± 0.09	96.00 ± 0.00	50.86 ± 0.19	79.30 ± 22.82	0.50/0.18
	16 D 32 D	76.43 ± 7.07 90.10 ± 5.85	76.67 ± 4.93 84.00 ± 1.00	91.61 ± 2.75 99.00 ± 0.00	89.99 ± 1.07 85.52 ± 5.34	47.55 ± 8.56 75.69 ± 12.75	58.08 ± 0.72 66.62 ± 4.18	20.77 ± 5.50 44.66 ± 7.26	13.90 ± 2.79 20.49 ± 7.07	73.93 ± 3.13 86.67 ± 1.86	40.74 ± 0.85 42.01 ± 0.94	58.97 ± 26.83 69.48 ± 25.14	1.00 / 0.99 1.00 / 0.98
100 diagonal (D)	64 D	95.56 ± 2.49	86.67 ± 0.58	99.00 ± 0.00	92.24 ± 1.68	90.89 ± 1.17	70.35 ± 0.45	65.62 ± 4.03	29.58 ± 2.02	91.67 ± 2.31	43.64 ± 1.36	76.52 ± 23.02	1.00 / 0.96
	128 D 256 D	96.00 ± 0.00 95.00 ± 0.00	87.33 ± 1.15 91.00 ± 0.00	99.00 ± 0.00 99.00 ± 0.00	93.89 ± 0.19 93.56 ± 0.19	93.00 ± 0.58 93.11 ± 0.19	72.70 ± 0.30 73.05 ± 0.20	74.34 ± 2.07 74.52 ± 0.95	29.66 ± 0.54 29.67 ± 0.67	93.67 ± 0.58 95.33 ± 0.58	44.82 ± 0.89 49.42 ± 0.65	78.44 ± 22.87 79.37 ± 22.38	1.00 / 0.92 1.00 / 0.84
	16 F	90.70 ± 1.07	83.00 ± 2.65	96.00 ± 3.00	91.22 ± 2.94	87.94 ± 0.54	68.72 ± 1.05	47.57 ± 2.54	23.75 ± 4.33	82.33 ± 2.08	41.51 ± 0.67	71.27 ± 24.23	1.00/0.99
	32 F	95.33 ± 1.53	85.00 ± 1.00	99.00 ± 0.00	93.50 ± 0.22	91.44 ± 0.84	70.94 ± 0.02	63.64 ± 1.98	29.82 ± 0.81	91.67 ± 0.58	43.94 ± 0.18	76.43 ± 23.01	0.99/0.97
100 full (F)	64 F 128 F	97.00 ± 0.00 96.33 ± 0.58	85.67 ± 1.53 90.33 ± 0.58	99.00 ± 0.00 99.00 ± 0.00	93.78 ± 0.19 93.00 ± 0.00	92.56 ± 0.19 93.89 ± 0.19	72.11 ± 0.08 73.11 ± 0.36	73.29 ± 0.64 76.50 ± 1.01	29.15 ± 0.24 29.45 ± 0.35	94.33 ± 1.53 96.00 ± 0.00	44.97 ± 0.05 49.81 ± 0.34	78.18 ± 23.03 79.74 ± 22.47	0.97/0.93 0.88/0.80
	256 F	96.33 ± 0.58	88.67 ± 0.58	99.00 ± 0.00	93.67 ± 0.00	94.89 ± 0.19	74.40 ± 0.16	72.90 ± 0.12	26.77 ± 0.68	96.00 ± 0.00	50.83 ± 0.09	79.35 ± 23.04	0.50/0.34
100 w/clusters (C)	16 C 5 16 C 7	$\begin{array}{c} 98.33 \pm 0.47 \\ 97.67 \pm 0.47 \end{array}$	89.00 ± 0.82 87.00 ± 0.82	$\begin{array}{c} 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \end{array}$	93.25 ± 0.40 93.46 ± 0.29	$\begin{array}{c} 92.89 \pm 0.87 \\ 93.11 \pm 0.68 \end{array}$	$ 72.32 \pm 0.36 \\ 72.52 \pm 0.43 $	77.08 ± 1.67 77.66 ± 1.30	$\begin{array}{c} 28.26 \pm 0.38 \\ 28.51 \pm 1.26 \end{array}$	96.67 ± 0.47 95.33 ± 0.47	68.30 ± 15.72 68.46 ± 14.94	81.51 ± 20.63 81.27 ± 20.35	1.00/0.95 1.00/0.93
	16 D	54.44 ± 6.87	47.00 ± 2.83	82.21 ± 3.59	73.38 ± 14.97	80.08 ± 3.71	51.02 ± 5.31	17.49 ± 1.10	11.58 ± 0.21	72.67 ± 6.03	39.65 ± 0.28	53.16 ± 24.97	1.00 / 1.00
500 diagonal (D)	32 D 64 D	58.08 ± 11.52 69.21 ± 2.03	47.00 ± 7.07 54.50 ± 9.19	82.06 ± 1.69 88.33 ± 4.04	78.62 ± 11.23 91.11 ± 0.38	85.57 ± 1.48 88.78 ± 0.38	52.98 ± 3.81 62.36 ± 3.52	21.73 ± 3.95 33.36 ± 6.69	12.53 ± 2.26 13.34 ± 1.86	75.33 ± 4.04 77.67 ± 2.31	39.78 ± 0.42 40.42 ± 0.98	55.66 ± 25.48 62.16 ± 26.05	1.00 / 1.00 1.00 / 0.99
500 diagonai (D)	128 D	79.77 ± 0.37	79.50 ± 9.19	95.89 ± 2.83	91.11 ± 0.38 91.89 ± 1.39	88.67 ± 0.38	65.92 ± 1.79	44.98 ± 10.98	13.34 ± 1.86 14.14 ± 0.19	81.00 ± 5.00	40.42 ± 0.98 40.34 ± 0.80	67.82 ± 26.05	1.00 / 0.99
	256 D	93.83 ± 2.52	85.00 ± 0.00	99.00 ± 0.00	93.78 ± 0.19	90.56 ± 0.38	68.95 ± 1.92	49.39 ± 4.36	22.33 ± 3.78	87.33 ± 2.31	42.15 ± 0.73	72.83 ± 25.93	1.00 / 0.97
	16 F 32 F	54.30 ± 1.13 75.10 ± 4.92	37.00 ± 5.66 46.50 ± 3.54	77.67 ± 0.58 91.67 ± 1.53	91.00 ± 0.00 91.56 ± 0.19	90.56 ± 0.19 91.56 ± 0.38	62.47 ± 0.79 67.37 ± 0.83	47.56 ± 0.29 51.17 ± 0.81	14.18 ± 0.67 13.44 ± 0.02	79.00 ± 1.00 81.67 ± 1.53	41.58 ± 0.23 41.92 ± 0.42	60.31 ± 24.42 65.84 ± 25.64	1.00/1.00 0.99/0.99
500 full (F)	52 F 64 F	75.10 ± 4.92 96.94 ± 0.42	46.50 ± 3.54 82.50 ± 0.71	91.67 ± 1.53 93.33 ± 0.58	91.56 ± 0.19 93.89 ± 0.69	91.56 ± 0.38 90.67 ± 0.00	67.37 ± 0.83 72.30 ± 0.71	51.17 ± 0.81 54.63 ± 0.79	13.44 ± 0.02 14.49 ± 0.27	81.67 ± 1.53 86.33 ± 0.58	41.92 ± 0.42 43.16 ± 0.08	65.84 ± 25.64 72.49 ± 26.64	0.99/0.99
` ′	128 F 256 F	97.67 ± 0.58 98.00 ± 0.00	83.50 ± 2.12 88.50 ± 0.71	98.00 ± 0.00 99.00 ± 0.00	93.56 ± 0.19 93.78 ± 0.19	92.00 ± 0.00 93.00 ± 0.88	71.92 ± 0.19 72.45 ± 0.38	65.02 ± 0.81 73.77 ± 1.21	28.49 ± 0.55 27.59 ± 0.39	93.00 ± 0.00 95.33 ± 0.58	43.85 ± 0.12 43.81 ± 0.17	76.47 ± 23.77 78.18 ± 24.16	0.88/0.86 0.50/0.47
	16 C 7	95.00 ± 0.00	86.00	98.00	93.78 ± 0.19	93.00 ± 0.88	71.19	54.69	20.03	90.00	45.81 ± 0.17	74.66	1.00/0.98
	16 C 10	96.00	87.00	99.00	93.00	91.33	69.93	53.48	30.09	94.00	44.89	75.87	1.00/0.98
500 w/clusters (C)	16 C 25	96.00 95.00	86.00 84.00	99.00 99.00	92.71	93.00 92.67	72.13	74.59	30.21	95.00	46.66	78.53 77.99	1.00/0.95
	64 C 5 64 C 7	95.00 98.00	84.00 88.00	99.00 99.00	93.67 94.00	92.67	72.32 72.18	75.60 75.83	27.17 28.14	96.00 96.00	44.43 47.68	77.99	0.97/0.93 0.97/0.91

Table 8: Absolute In-Distribution ROUGE-L scores for various tasks and methods

Model Type	Method Type					Ta	sks					. Average	Para. Saved
Woder Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	raia. Saveu
	base lora	$\begin{array}{c} 0.26 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.19 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c }\hline 0.42 \pm 0.00 \\ 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 0.11 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c c} 0.11 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.19 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.29 \pm 0.22 \\ 1.00 \pm 0.00 \end{array}$	1.00/1.00 0.00/0.00
TIES	10 50 100 500	$\begin{array}{c} 0.81 \pm 0.00 \\ 0.59 \pm 0.00 \\ 0.55 \pm 0.00 \\ 0.37 \pm 0.00 \end{array}$	$\begin{array}{c} 0.57 \pm 0.02 \\ 0.41 \pm 0.00 \\ 0.40 \pm 0.00 \\ 0.26 \pm 0.00 \end{array}$	$\begin{array}{c} 0.45 \pm 0.04 \\ 0.18 \pm 0.05 \\ 0.20 \pm 0.05 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.10 \pm 0.01 \\ 0.03 \pm 0.01 \\ 0.01 \pm 0.02 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.91 \pm 0.01 \\ 0.88 \pm 0.00 \\ 0.83 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.52 \pm 0.00 \\ 0.34 \pm 0.00 \\ 0.36 \pm 0.00 \\ 0.31 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.71 \pm 0.01 \\ 0.67 \pm 0.00 \\ 0.65 \pm 0.00 \\ 0.58 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.58 \pm 0.00 \\ 0.62 \pm 0.00 \\ 0.57 \pm 0.02 \\ 0.37 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.82 \pm 0.01 \\ 0.32 \pm 0.04 \\ 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.80 \pm 0.00 \\ 0.78 \pm 0.00 \\ 0.78 \pm 0.00 \\ 0.41 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.62 \pm 0.22 \\ 0.48 \pm 0.27 \\ 0.44 \pm 0.29 \\ 0.32 \pm 0.26 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 1.00 1.00 / 1.00
SVD	SVD 2 SVD 4 SVD 8 SVD 16	$\begin{array}{c} 0.98 \pm 0.03 \\ 0.99 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.07 \pm 0.02 \\ 1.04 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.10 \\ 0.99 \pm 0.08 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.99 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.04 \\ 1.00 \pm 0.03 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	0.88 / 0.88 0.75 / 0.75 0.50 / 0.50 0.00 / 0.00
10 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 1.02 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.00 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.99 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.97 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.03 \pm 0.02 \\ 0.97 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.12 \pm 0.03 \\ 1.04 \pm 0.03 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.99 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.99 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$ \begin{array}{c} 1.01 \pm 0.04 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array} $	1.00 / 0.90 1.00 / 0.80 1.00 / 0.60 1.00 / 0.20 1.00 / -0.60
10 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{array}{c} 1.06 \pm 0.01 \\ 1.04 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array} $	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.99 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.99 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.02 \\ 0.96 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.07 \pm 0.00 \\ 1.00 \pm 0.02 \\ 0.98 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.02 \pm 0.03 \\ 1.00 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	1.00/0.90 0.99/0.79 0.97/0.57 0.88/0.07 0.50/-1.10
50 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.98 \pm 0.04 \\ 1.00 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.98 \pm 0.01 \\ 1.02 \pm 0.02 \\ 1.05 \pm 0.02 \\ 1.08 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.92 \pm 0.06 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.85 \pm 0.06 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.94 \pm 0.02 \\ 0.96 \pm 0.02 \\ 0.97 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.69 \pm 0.05 \\ 0.85 \pm 0.02 \\ 0.99 \pm 0.01 \\ 0.98 \pm 0.02 \\ 0.97 \pm 0.03 \end{array}$	$ \begin{vmatrix} 0.88 \pm 0.10 \\ 1.00 \pm 0.12 \\ 1.09 \pm 0.03 \\ 1.10 \pm 0.03 \\ 1.00 \pm 0.03 \end{vmatrix} $	$\begin{array}{c} 0.88 \pm 0.07 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.86 \pm 0.01 \\ 0.90 \pm 0.00 \\ 0.94 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.90 \pm 0.10 \\ 0.97 \pm 0.06 \\ 1.01 \pm 0.04 \\ 1.02 \pm 0.04 \\ 1.00 \pm 0.02 \end{array}$	1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84 1.00 / 0.68
50 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.99 \pm 0.04 \\ 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.02 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.06 \pm 0.02 \\ 1.06 \pm 0.01 \\ 1.02 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 0.95 \pm 0.02 \\ 0.98 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.95 \pm 0.01 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \\ 0.98 \pm 0.00 \\ 0.99 \pm 0.00 \end{array}$	$\begin{array}{c} 0.65 \pm 0.09 \\ 0.96 \pm 0.01 \\ 1.03 \pm 0.01 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$ \begin{vmatrix} 1.01 \pm 0.15 \\ 1.09 \pm 0.03 \\ 1.11 \pm 0.00 \\ 1.03 \pm 0.04 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.97 \pm 0.02 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.88 \pm 0.01 \\ 0.93 \pm 0.00 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.94 \pm 0.11 \\ 0.99 \pm 0.04 \\ 1.02 \pm 0.04 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \end{array}$	1.00/0.98 0.99/0.95 0.97/0.89 0.88/0.72 0.50/0.18
100 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.80 \pm 0.07 \\ 0.95 \pm 0.06 \\ 1.01 \pm 0.03 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.06 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.06 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.01 \\ 0.91 \pm 0.06 \\ 0.98 \pm 0.02 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.51 \pm 0.09 \\ 0.80 \pm 0.13 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.01 \\ 0.99 \pm 0.00 \end{array}$	$ \begin{array}{c} 0.81 \pm 0.02 \\ 0.91 \pm 0.05 \\ 0.95 \pm 0.01 \\ 0.98 \pm 0.00 \\ 0.98 \pm 0.00 \end{array} $	$\begin{array}{c} 0.30 \pm 0.07 \\ 0.62 \pm 0.10 \\ 0.90 \pm 0.05 \\ 1.00 \pm 0.03 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{array}{c} 0.54 \pm 0.11 \\ 0.78 \pm 0.25 \\ 1.11 \pm 0.07 \\ 1.11 \pm 0.02 \\ 1.11 \pm 0.03 \end{array} $	$\begin{array}{c} 0.78 \pm 0.03 \\ 0.91 \pm 0.02 \\ 0.96 \pm 0.02 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.83 \pm 0.02 \\ 0.85 \pm 0.01 \\ 0.88 \pm 0.02 \\ 0.92 \pm 0.00 \\ 0.99 \pm 0.02 \end{array}$	$\begin{array}{c} 0.73 \pm 0.21 \\ 0.87 \pm 0.14 \\ 0.98 \pm 0.07 \\ 1.00 \pm 0.05 \\ 1.01 \pm 0.04 \end{array}$	1.00 / 0.99 1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84
100 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.95 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.01 \\ 0.97 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.93 \pm 0.01 \\ 0.96 \pm 0.00 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.66 \pm 0.03 \\ 0.87 \pm 0.03 \\ 0.99 \pm 0.01 \\ 1.03 \pm 0.01 \\ 0.98 \pm 0.00 \end{array}$	$\begin{array}{c} 0.90 \pm 0.16 \\ 1.12 \pm 0.03 \\ 1.10 \pm 0.01 \\ 1.10 \pm 0.01 \\ 1.00 \pm 0.03 \end{array}$	$\begin{array}{c} 0.87 \pm 0.02 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.02 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.85 \pm 0.01 \\ 0.89 \pm 0.00 \\ 0.93 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.90 \pm 0.10 \\ 0.98 \pm 0.07 \\ 1.00 \pm 0.04 \\ 1.02 \pm 0.03 \\ 1.00 \pm 0.01 \end{array}$	1.00/0.99 0.99/0.97 0.97/0.93 0.88/0.80 0.50/0.34
100 w/clusters (C)	16 C 5 16 C 7	1.13 ± 0.01 1.12 ± 0.01	1.03 ± 0.01 1.01 ± 0.01	1.00 ± 0.00 1.00 ± 0.00	1.00 ± 0.00 1.00 ± 0.00	0.99 ± 0.01 0.99 ± 0.01	0.97 ± 0.00 0.97 ± 0.00	1.01 ± 0.02 1.02 ± 0.02	1.22 ± 0.02 1.22 ± 0.05	1.05 ± 0.01 1.03 ± 0.01	1.00 ± 0.05 1.01 ± 0.03	1.04 ± 0.07 1.04 ± 0.07	1.00/0.95 1.00/0.93
500 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.57 \pm 0.07 \\ 0.61 \pm 0.12 \\ 0.73 \pm 0.02 \\ 0.84 \pm 0.00 \\ 0.99 \pm 0.03 \end{array}$	$ \begin{vmatrix} 0.55 \pm 0.03 \\ 0.55 \pm 0.08 \\ 0.63 \pm 0.11 \\ 0.92 \pm 0.02 \\ 0.99 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.04 \\ 0.83 \pm 0.02 \\ 0.89 \pm 0.04 \\ 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.78 \pm 0.16 \\ 0.84 \pm 0.12 \\ 0.97 \pm 0.00 \\ 0.98 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.85 \pm 0.04 \\ 0.91 \pm 0.02 \\ 0.94 \pm 0.00 \\ 0.94 \pm 0.00 \\ 0.96 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.73 \pm 0.07 \\ 0.75 \pm 0.05 \\ 0.86 \pm 0.03 \\ 0.90 \pm 0.02 \\ 0.93 \pm 0.02 \end{vmatrix} $	$\begin{array}{c} 0.24 \pm 0.02 \\ 0.30 \pm 0.05 \\ 0.46 \pm 0.09 \\ 0.62 \pm 0.14 \\ 0.68 \pm 0.05 \end{array}$	$ \begin{vmatrix} 0.45 \pm 0.01 \\ 0.49 \pm 0.07 \\ 0.51 \pm 0.07 \\ 0.54 \pm 0.01 \\ 0.85 \pm 0.14 \end{vmatrix} $	$\begin{array}{c} 0.76 \pm 0.06 \\ 0.79 \pm 0.04 \\ 0.82 \pm 0.02 \\ 0.85 \pm 0.05 \\ 0.92 \pm 0.02 \end{array}$	$ \begin{vmatrix} 0.81 \pm 0.00 \\ 0.82 \pm 0.01 \\ 0.83 \pm 0.01 \\ 0.83 \pm 0.01 \\ 0.85 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.66 \pm 0.20 \\ 0.69 \pm 0.20 \\ 0.77 \pm 0.18 \\ 0.84 \pm 0.15 \\ 0.92 \pm 0.11 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 0.99 1.00 / 0.98 1.00 / 0.97
500 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.57 \pm 0.01 \\ 0.79 \pm 0.05 \\ 1.02 \pm 0.00 \\ 1.03 \pm 0.01 \\ 1.03 \pm 0.00 \end{array}$	$\begin{array}{c} 0.43 \pm 0.07 \\ 0.54 \pm 0.04 \\ 0.96 \pm 0.01 \\ 0.97 \pm 0.02 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.78 \pm 0.01 \\ 0.93 \pm 0.02 \\ 0.94 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.00 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.00 \\ 0.97 \pm 0.00 \\ 0.96 \pm 0.00 \\ 0.98 \pm 0.00 \\ 0.99 \pm 0.01 \end{array}$	$\begin{array}{c} 0.86 \pm 0.01 \\ 0.92 \pm 0.00 \\ 0.97 \pm 0.01 \\ 0.97 \pm 0.00 \\ 0.97 \pm 0.00 \end{array}$	$\begin{array}{c} 0.65 \pm 0.00 \\ 0.70 \pm 0.01 \\ 0.74 \pm 0.01 \\ 0.88 \pm 0.01 \\ 1.00 \pm 0.02 \end{array}$	$ \begin{vmatrix} 0.55 \pm 0.02 \\ 0.52 \pm 0.00 \\ 0.55 \pm 0.01 \\ 1.07 \pm 0.02 \\ 1.04 \pm 0.02 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.86 \pm 0.02 \\ 0.91 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.84 \pm 0.00 \\ 0.85 \pm 0.00 \\ 0.87 \pm 0.00 \\ 0.90 \pm 0.00 \\ 0.93 \pm 0.00 \end{array}$	$\begin{array}{c} 0.76 \pm 0.17 \\ 0.81 \pm 0.16 \\ 0.89 \pm 0.14 \\ 0.98 \pm 0.05 \\ 1.00 \pm 0.03 \end{array}$	1.00/1.00 0.99/0.99 0.97/0.96 0.88/0.86 0.50/0.47
500 w/clusters (C)	16 C 7 16 C 10 16 C 25 64 C 5 64 C 7	1.09 1.10 1.10 1.09 1.12	1.00 1.01 1.00 0.98 1.02	0.99 1.00 1.00 1.00 1.00	1.00 0.99 0.99 1.00 1.00	0.98 0.97 0.99 0.99 1.00	0.96 0.94 0.97 0.97 0.97	0.72 0.72 0.98 0.99 1.00	0.88 1.29 1.30 1.17 1.22	0.98 1.02 1.03 1.04 1.04	0.93 0.92 0.96 0.93 0.99	0.95 1.00 1.03 1.02 1.04	1.00/0.98 1.00/0.98 1.00/0.95 0.97/0.93 0.97/0.91
1000 w/clusters (C)	16 C 25	1.09	0.98	1.00	1.00	0.97	0.97	0.74	1.29	1.05	0.94	1.00	1.00/0.97

Table 9: Relative In-Distribution ROUGE-1 scores for various tasks and methods

M-d-l T	Mathad Tona					Tas	ks					A	D C1
Model Type	Method Type	task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	Para. Saved
	base lora	$\begin{array}{c} 24.44 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.60 \pm 0.00 \\ 86.00 \pm 0.00 \end{array}$	19.13 ± 0.00 99.00 ± 0.00	39.22 ± 0.00 93.67 ± 0.00	10.42 ± 0.00 94.33 ± 0.00	39.88 ± 0.00 78.43 ± 0.00	8.05 ± 0.00 74.90 ± 0.00	6.96 ± 0.00 26.87 ± 0.00	17.82 ± 0.00 95.00 ± 0.00	55.03 ± 0.00 68.66 ± 0.00	22.43 ± 16.49 81.14 ± 20.67	1.00 / 1.00 0.00/0.00
TIES	10 50 100 500	$76.50 \pm 0.00 \\ 55.80 \pm 0.00 \\ 52.43 \pm 0.00 \\ 35.18 \pm 0.00$	$\begin{array}{c} 49.00 \pm 1.73 \\ 35.00 \pm 0.00 \\ 34.00 \pm 0.00 \\ 22.00 \pm 0.00 \end{array}$	$\begin{array}{c} 44.33 \pm 4.04 \\ 18.00 \pm 5.20 \\ 19.67 \pm 4.62 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 9.80 \pm 0.58 \\ 2.42 \pm 0.50 \\ 1.09 \pm 1.66 \\ 0.00 \pm 0.00 \end{array}$	$\begin{array}{c} 78.56 \pm 0.96 \\ 85.78 \pm 0.96 \\ 83.33 \pm 0.00 \\ 78.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 40.44 \pm 0.00 \\ 26.75 \pm 0.00 \\ 28.57 \pm 0.00 \\ 24.32 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 53.10 \pm 0.67 \\ 49.96 \pm 0.00 \\ 48.89 \pm 0.00 \\ 43.80 \pm 0.04 \end{array}$	$ \begin{vmatrix} 15.48 \pm 0.12 \\ 16.73 \pm 0.00 \\ 15.18 \pm 0.42 \\ 9.96 \pm 0.13 \end{vmatrix} $	$\begin{array}{c} 77.67 \pm 1.15 \\ 30.00 \pm 3.46 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 54.89 \pm 0.06 \\ 53.87 \pm 0.02 \\ 53.44 \pm 0.02 \\ 27.90 \pm 0.03 \end{array}$	$\begin{array}{c} 49.98 \pm 23.33 \\ 37.43 \pm 23.49 \\ 33.76 \pm 25.22 \\ 24.40 \pm 23.79 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 1.00 1.00 / 1.00
SVD	SVD 2 SVD 4 SVD 8 SVD 16	$\begin{array}{c} 93.15 \pm 2.77 \\ 94.01 \pm 3.60 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 92.24 \pm 1.85 \\ 89.21 \pm 0.71 \\ 87.40 \pm 0.59 \\ 86.00 \pm 0.00 \end{array}$	$\begin{array}{c} 99.09 \pm 0.18 \\ 99.05 \pm 0.09 \\ 99.05 \pm 0.09 \\ 99.00 \pm 0.00 \end{array}$	$\begin{array}{c} 93.44 \pm 0.14 \\ 93.65 \pm 0.03 \\ 93.65 \pm 0.03 \\ 93.67 \pm 0.00 \end{array}$	$\begin{array}{c} 93.89 \pm 0.35 \\ 94.66 \pm 0.63 \\ 94.36 \pm 0.38 \\ 94.33 \pm 0.00 \end{array}$	$ \begin{vmatrix} 77.33 \pm 0.29 \\ 78.42 \pm 0.23 \\ 78.21 \pm 0.03 \\ 78.44 \pm 0.03 \end{vmatrix} $	$\begin{array}{c} 75.40 \pm 1.01 \\ 74.09 \pm 1.12 \\ 75.57 \pm 0.00 \\ 74.73 \pm 0.18 \end{array}$	$\begin{array}{c} 26.90 \pm 2.68 \\ 26.47 \pm 2.06 \\ 26.88 \pm 0.27 \\ 26.87 \pm 0.00 \end{array}$	$\begin{array}{c} 95.06 \pm 1.35 \\ 93.98 \pm 0.77 \\ 95.51 \pm 1.09 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 67.71 \pm 0.49 \\ 69.37 \pm 0.21 \\ 69.33 \pm 0.08 \\ 68.62 \pm 0.04 \end{array}$	$\begin{array}{c} 81.33 \pm 20.85 \\ 81.22 \pm 20.80 \\ 83.02 \pm 19.87 \\ 80.76 \pm 21.05 \end{array}$	0.88 / 0.88 0.75 / 0.75 0.50 / 0.50 0.00 / 0.00
10 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 96.67 \pm 0.58 \\ 95.67 \pm 0.58 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 87.00 \pm 1.00 \\ 90.00 \pm 1.00 \\ 88.33 \pm 0.58 \\ 86.67 \pm 0.58 \\ 86.00 \pm 0.00 \end{array}$	$\begin{array}{c} 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \end{array}$	$\begin{array}{c} 94.00 \pm 0.67 \\ 93.00 \pm 0.33 \\ 93.67 \pm 0.00 \\ 93.67 \pm 0.00 \\ 93.67 \pm 0.00 \end{array}$	$\begin{array}{c} 93.11 \pm 0.38 \\ 94.89 \pm 0.51 \\ 94.78 \pm 0.38 \\ 94.33 \pm 0.00 \\ 94.33 \pm 0.00 \end{array}$	$ \begin{array}{c} 76.08 \pm 0.17 \\ 77.46 \pm 0.24 \\ 78.28 \pm 0.07 \\ 78.45 \pm 0.16 \\ 78.43 \pm 0.00 \end{array} $	$\begin{array}{c} 77.26 \pm 1.47 \\ 72.53 \pm 1.00 \\ 75.47 \pm 0.58 \\ 75.46 \pm 0.51 \\ 74.90 \pm 0.00 \end{array}$	$\begin{array}{c} 30.15 \pm 0.72 \\ 27.98 \pm 0.71 \\ 26.53 \pm 0.25 \\ 26.64 \pm 0.23 \\ 26.87 \pm 0.00 \end{array}$	$\begin{array}{c} 94.00 \pm 1.73 \\ 94.67 \pm 0.58 \\ 96.00 \pm 0.00 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 68.25 \pm 0.18 \\ 69.16 \pm 0.41 \\ 69.36 \pm 0.05 \\ 68.70 \pm 0.14 \\ 68.59 \pm 0.03 \end{array}$	$\begin{array}{c} 81.55 \pm 20.03 \\ 81.44 \pm 20.80 \\ 81.64 \pm 21.06 \\ 81.29 \pm 20.92 \\ 81.18 \pm 20.86 \end{array}$	1.00 / 0.90 1.00 / 0.80 1.00 / 0.60 1.00 / 0.20 1.00 / -0.60
10 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 97.00 \pm 0.00 \\ 96.67 \pm 0.58 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 91.00 \pm 1.00 \\ 89.33 \pm 0.58 \\ 88.67 \pm 0.58 \\ 86.67 \pm 0.58 \\ 86.00 \pm 0.00 \end{array}$	$\begin{array}{c} 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \end{array}$	$\begin{array}{c} 93.56 \pm 0.19 \\ 93.22 \pm 0.19 \\ 93.67 \pm 0.00 \\ 93.67 \pm 0.00 \\ 93.67 \pm 0.00 \end{array}$	$\begin{array}{c} 93.56 \pm 0.69 \\ 94.44 \pm 0.19 \\ 94.56 \pm 0.38 \\ 94.33 \pm 0.00 \\ 94.33 \pm 0.00 \end{array}$	$77.64 \pm 0.25 \\ 77.84 \pm 0.21 \\ 78.19 \pm 0.08 \\ 78.46 \pm 0.03 \\ 78.44 \pm 0.03$	$\begin{array}{c} 75.78 \pm 1.25 \\ 72.24 \pm 0.59 \\ 75.97 \pm 0.58 \\ 74.90 \pm 0.00 \\ 74.79 \pm 0.19 \end{array}$	$\begin{array}{c} 28.71 \pm 0.09 \\ 26.84 \pm 0.50 \\ 26.43 \pm 0.34 \\ 26.72 \pm 0.13 \\ 26.87 \pm 0.00 \end{array}$	$\begin{array}{c} 96.00 \pm 1.00 \\ 94.67 \pm 0.58 \\ 96.00 \pm 0.00 \\ 95.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 68.69 \pm 0.08 \\ 69.55 \pm 0.08 \\ 69.38 \pm 0.11 \\ 68.65 \pm 0.03 \\ 68.64 \pm 0.03 \end{array}$	$\begin{array}{c} 82.09 \pm 20.68 \\ 81.38 \pm 21.11 \\ 81.69 \pm 21.07 \\ 81.24 \pm 20.91 \\ 81.17 \pm 20.86 \end{array}$	1.00/0.90 0.99/0.79 0.97/0.57 0.88/0.07 0.50/-1.10
50 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 92.76 \pm 3.53 \\ 95.33 \pm 2.08 \\ 97.00 \pm 0.00 \\ 96.33 \pm 0.58 \\ 95.67 \pm 0.58 \end{array}$	$\begin{array}{c} 84.67 \pm 1.15 \\ 87.33 \pm 2.08 \\ 90.33 \pm 1.53 \\ 92.67 \pm 0.58 \\ 88.33 \pm 0.58 \end{array}$	$\begin{array}{c} 99.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 86.17 \pm 5.81 \\ 92.60 \pm 0.29 \\ 93.78 \pm 0.19 \\ 93.56 \pm 0.19 \\ 93.56 \pm 0.19 \end{array}$	$\begin{array}{c} 79.83 \pm 6.08 \\ 90.35 \pm 1.00 \\ 93.00 \pm 0.58 \\ 93.00 \pm 0.58 \\ 94.67 \pm 0.67 \end{array}$	$\begin{array}{c} 73.55 \pm 1.39 \\ 75.43 \pm 1.33 \\ 76.27 \pm 0.49 \\ 77.24 \pm 0.19 \\ 78.45 \pm 0.14 \end{array}$	$ \begin{array}{c} 51.72 \pm 3.78 \\ 63.84 \pm 1.64 \\ 74.39 \pm 0.90 \\ 73.76 \pm 1.25 \\ 72.86 \pm 2.07 \end{array} $	$\begin{array}{c} 23.75 \pm 2.66 \\ 26.97 \pm 3.21 \\ 29.28 \pm 0.81 \\ 29.58 \pm 0.93 \\ 27.00 \pm 0.77 \end{array}$	$\begin{array}{c} 83.90 \pm 6.43 \\ 93.33 \pm 1.15 \\ 95.67 \pm 0.58 \\ 95.00 \pm 0.00 \\ 95.33 \pm 0.58 \end{array}$	$\begin{array}{c} 59.05 \pm 0.94 \\ 61.94 \pm 0.32 \\ 64.84 \pm 0.27 \\ 69.04 \pm 0.54 \\ 69.61 \pm 0.18 \end{array}$	$\begin{array}{c} 73.44 \pm 22.08 \\ 78.61 \pm 21.60 \\ 81.36 \pm 20.83 \\ 81.92 \pm 20.44 \\ 81.45 \pm 21.00 \end{array}$	1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84 1.00 / 0.68
50 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 94.06 \pm 3.54 \\ 97.00 \pm 0.00 \\ 96.67 \pm 0.58 \\ 97.00 \pm 0.00 \\ 95.00 \pm 0.00 \end{array}$	$\begin{array}{c} 85.67 \pm 1.15 \\ 85.67 \pm 1.53 \\ 91.00 \pm 2.00 \\ 91.00 \pm 1.00 \\ 88.00 \pm 0.00 \end{array}$	$\begin{array}{c} 98.67 \pm 0.58 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \end{array}$	$\begin{array}{c} 90.35 \pm 1.37 \\ 93.67 \pm 0.00 \\ 93.56 \pm 0.19 \\ 93.33 \pm 0.00 \\ 93.67 \pm 0.00 \end{array}$	$\begin{array}{c} 89.97 \pm 1.78 \\ 92.22 \pm 0.69 \\ 93.22 \pm 0.51 \\ 94.11 \pm 0.51 \\ 94.44 \pm 0.19 \end{array}$	$ \begin{array}{c} 74.46 \pm 0.58 \\ 75.86 \pm 0.22 \\ 77.17 \pm 0.38 \\ 77.23 \pm 0.17 \\ 77.97 \pm 0.24 \end{array} $	$\begin{array}{c} 49.03 \pm 7.07 \\ 71.68 \pm 0.65 \\ 77.11 \pm 0.51 \\ 73.67 \pm 0.58 \\ 75.47 \pm 0.58 \end{array}$	$\begin{array}{c} 27.14 \pm 3.94 \\ 29.26 \pm 0.70 \\ 29.75 \pm 0.03 \\ 27.62 \pm 1.12 \\ 26.96 \pm 0.09 \end{array}$	$\begin{array}{c} 92.33 \pm 1.53 \\ 95.67 \pm 1.53 \\ 95.33 \pm 0.58 \\ 95.00 \pm 1.00 \\ 96.00 \pm 0.00 \end{array}$		$\begin{array}{c} 76.19 \pm 22.80 \\ 80.39 \pm 20.81 \\ 82.09 \pm 20.33 \\ 81.74 \pm 20.97 \\ 81.58 \pm 20.92 \end{array}$	1.00/0.98 0.99/0.95 0.97/0.89 0.88/0.72 0.50/0.18
100 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$76.43 \pm 7.07 \\ 90.10 \pm 5.85 \\ 95.56 \pm 2.49 \\ 96.00 \pm 0.00 \\ 95.00 \pm 0.00$	$76.67 \pm 4.93 \\ 84.00 \pm 1.00 \\ 86.67 \pm 0.58 \\ 87.33 \pm 1.15 \\ 91.00 \pm 0.00$	$\begin{array}{c} 91.61 \pm 2.75 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 89.99 \pm 1.07 \\ 85.52 \pm 5.34 \\ 92.24 \pm 1.68 \\ 93.89 \pm 0.19 \\ 93.56 \pm 0.19 \end{array}$	$\begin{array}{c} 47.89 \pm 8.62 \\ 75.88 \pm 12.57 \\ 90.89 \pm 1.17 \\ 93.00 \pm 0.58 \\ 93.11 \pm 0.19 \end{array}$	$ \begin{vmatrix} 63.17 \pm 1.31 \\ 71.15 \pm 3.61 \\ 74.57 \pm 0.50 \\ 76.68 \pm 0.18 \\ 76.93 \pm 0.23 \end{vmatrix} $	22.23 ± 5.27 46.10 ± 7.39 67.07 ± 3.81 74.84 ± 2.23 75.13 ± 0.84	$ \begin{vmatrix} 14.46 \pm 2.89 \\ 21.04 \pm 6.76 \\ 29.78 \pm 1.92 \\ 29.79 \pm 0.50 \\ 29.75 \pm 0.73 \end{vmatrix} $	$73.93 \pm 3.13 \\ 86.67 \pm 1.86 \\ 91.67 \pm 2.31 \\ 93.67 \pm 0.58 \\ 95.33 \pm 0.58$	$\begin{array}{c} 57.17 \pm 1.05 \\ 58.64 \pm 1.02 \\ 60.28 \pm 1.51 \\ 63.49 \pm 0.34 \\ 67.89 \pm 1.34 \end{array}$		1.00 / 0.99 1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84
100 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 90.70 \pm 1.07 \\ 95.33 \pm 1.53 \\ 97.00 \pm 0.00 \\ 96.33 \pm 0.58 \\ 96.33 \pm 0.58 \end{array}$	83.00 ± 2.65 85.00 ± 1.00 85.67 ± 1.53 90.33 ± 0.58 88.67 ± 0.58	$\begin{array}{c} 96.00 \pm 3.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ 99.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 91.22 \pm 2.94 \\ 93.50 \pm 0.22 \\ 93.78 \pm 0.19 \\ 93.00 \pm 0.00 \\ 93.67 \pm 0.00 \end{array}$	$\begin{array}{c} 87.94 \pm 0.54 \\ 91.44 \pm 0.84 \\ 92.56 \pm 0.19 \\ 93.89 \pm 0.19 \\ 94.89 \pm 0.19 \end{array}$	$\begin{array}{c c} 73.07 \pm 0.93 \\ 75.00 \pm 0.19 \\ 76.01 \pm 0.13 \\ 77.04 \pm 0.30 \\ 78.16 \pm 0.18 \end{array}$	49.41 ± 2.04 65.09 ± 2.23 73.96 ± 0.89 77.33 ± 1.01 73.40 ± 0.12	$ \begin{array}{c c} 24.17 \pm 4.22 \\ 30.20 \pm 0.81 \\ 29.46 \pm 0.21 \\ 29.49 \pm 0.35 \\ 26.86 \pm 0.68 \end{array} $	$\begin{array}{c} 82.33 \pm 2.08 \\ 91.67 \pm 0.58 \\ 94.33 \pm 1.53 \\ 96.00 \pm 0.00 \\ 96.00 \pm 0.00 \end{array}$	$\begin{array}{c} 58.18 \pm 0.44 \\ 60.92 \pm 0.26 \\ 64.07 \pm 0.37 \\ 68.76 \pm 0.25 \\ 69.47 \pm 0.23 \end{array}$	$73.60 \pm 22.23 \\ 78.72 \pm 20.72 \\ 80.58 \pm 20.59 \\ 82.12 \pm 20.35 \\ 81.64 \pm 21.15$	1.00/0.99 0.99/0.97 0.97/0.93 0.88/0.80 0.50/0.34
100 w/clusters (C)	16 C 5 16 C 7	$98.33 \pm 0.47 \\ 97.67 \pm 0.47$	89.00 ± 0.82 87.00 ± 0.82	99.00 ± 0.00 99.00 ± 0.00	93.25 ± 0.40 93.46 ± 0.29	92.89 ± 0.87 93.11 ± 0.68	76.33 ± 0.28 76.55 ± 0.29	78.24 ± 2.26 79.03 ± 2.05	28.45 ± 0.40 28.62 ± 1.29	96.67 ± 0.47 95.33 ± 0.47	75.93 ± 8.31 76.55 ± 6.91	82.81 ± 20.02 82.63 ± 19.76	1.00/0.95 1.00/0.93
500 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 54.44 \pm 6.87 \\ 58.08 \pm 11.52 \\ 69.21 \pm 2.03 \\ 79.77 \pm 0.37 \\ 93.83 \pm 2.52 \end{array}$	$\begin{array}{c} 47.00 \pm 2.83 \\ 47.00 \pm 7.07 \\ 54.50 \pm 9.19 \\ 79.50 \pm 2.12 \\ 85.00 \pm 0.00 \end{array}$	82.21 ± 3.59 82.06 ± 1.69 88.33 ± 4.04 95.89 ± 2.83 99.00 ± 0.00	$\begin{array}{c} 73.38 \pm 14.97 \\ 78.62 \pm 11.23 \\ 91.11 \pm 0.38 \\ 91.89 \pm 1.39 \\ 93.78 \pm 0.19 \end{array}$	$\begin{array}{c} 80.13 \pm 3.68 \\ 85.57 \pm 1.48 \\ 88.78 \pm 0.38 \\ 88.67 \pm 0.00 \\ 90.56 \pm 0.38 \end{array}$	57.42 ± 5.29 59.19 ± 3.70 67.71 ± 2.59 70.27 ± 1.73 73.25 ± 1.86		$ \begin{vmatrix} 12.19 \pm 0.30 \\ 13.15 \pm 1.94 \\ 13.80 \pm 1.95 \\ 14.63 \pm 0.25 \\ 22.93 \pm 3.86 \end{vmatrix} $	72.67 ± 6.03 75.33 ± 4.04 77.67 ± 2.31 81.00 ± 5.00 87.33 ± 2.31	$\begin{array}{c} 55.79 \pm 0.20 \\ 56.07 \pm 0.52 \\ 56.78 \pm 0.73 \\ 56.88 \pm 0.55 \\ 58.48 \pm 0.20 \end{array}$	$\begin{array}{c} 55.64 \pm 24.25 \\ 58.16 \pm 24.56 \\ 64.61 \pm 24.79 \\ 70.20 \pm 24.63 \\ 75.20 \pm 23.90 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 0.99 1.00 / 0.98 1.00 / 0.97
500 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 54.30 \pm 1.13 \\ 75.10 \pm 4.92 \\ 96.94 \pm 0.42 \\ 97.67 \pm 0.58 \\ 98.00 \pm 0.00 \end{array}$	$\begin{array}{c} 37.00 \pm 5.66 \\ 46.50 \pm 3.54 \\ 82.50 \pm 0.71 \\ 83.50 \pm 2.12 \\ 88.50 \pm 0.71 \end{array}$	$\begin{array}{c} 77.67 \pm 0.58 \\ 91.67 \pm 1.53 \\ 93.33 \pm 0.58 \\ 98.00 \pm 0.00 \\ 99.00 \pm 0.00 \end{array}$	$\begin{array}{c} 91.00 \pm 0.00 \\ 91.56 \pm 0.19 \\ 93.89 \pm 0.69 \\ 93.56 \pm 0.19 \\ 93.78 \pm 0.19 \end{array}$	$\begin{array}{c} 90.56 \pm 0.19 \\ 91.56 \pm 0.38 \\ 90.67 \pm 0.00 \\ 92.00 \pm 0.00 \\ 93.00 \pm 0.88 \end{array}$	$ \begin{vmatrix} 67.63 \pm 0.45 \\ 72.03 \pm 0.15 \\ 75.99 \pm 0.64 \\ 75.80 \pm 0.16 \\ 76.33 \pm 0.29 \end{vmatrix} $	$ \begin{array}{c} 48.81 \pm 0.35 \\ 52.63 \pm 0.86 \\ 55.63 \pm 1.07 \\ 66.19 \pm 0.81 \\ 74.60 \pm 1.21 \end{array} $	$ \begin{vmatrix} 14.70 \pm 0.65 \\ 13.93 \pm 0.02 \\ 14.74 \pm 0.27 \\ 28.67 \pm 0.49 \\ 27.82 \pm 0.42 \end{vmatrix} $	$79.00 \pm 1.00 \\ 81.67 \pm 1.53 \\ 86.33 \pm 0.58 \\ 93.00 \pm 0.00 \\ 95.33 \pm 0.58$	$\begin{array}{c} 57.66 \pm 0.19 \\ 58.50 \pm 0.20 \\ 59.43 \pm 0.05 \\ 61.53 \pm 0.13 \\ 63.70 \pm 0.14 \end{array}$	$62.69 \pm 23.46 \\ 68.24 \pm 24.29 \\ 74.69 \pm 25.01 \\ 78.84 \pm 21.50 \\ 80.75 \pm 21.60$	1.00/1.00 0.99/0.99 0.97/0.96 0.88/0.86 0.50/0.47
500 w/clusters (C)	16 C 7 16 C 10 16 C 25 64 C 5 64 C 7	95.00 96.00 96.00 95.00 98.00	86.00 87.00 86.00 84.00 88.00	98.00 99.00 99.00 99.00 99.00	93.67 93.00 92.71 93.67 94.00	91.67 91.33 93.00 92.67 93.33	75.10 74.17 76.42 76.45 76.42	55.52 55.14 75.42 76.43 76.67	20.50 30.29 30.40 27.49 28.48	90.00 94.00 95.00 96.00 96.00	63.57 63.09 66.07 64.10 68.00	76.90 78.30 81.00 80.48 81.79	1.00/0.98 1.00/0.98 1.00/0.95 0.97/0.93 0.97/0.91
1000 w/clusters (C)	16 C 25	95.00	84.00	99.00	93.67	90.67	76.43	56.71	30.20	97.00	64.61	78.73	1.00/0.97

Table 10: Absolute In-Distribution ROUGE-1 scores for various tasks and methods

Model Type	Method Type					Ta	sks					. Average	Para. Saved
Woder Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	raia. Saveu
	base lora	$\begin{array}{c} 0.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c }\hline 0.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{c} 0.02 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{c} 0.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{ c c c c c c }\hline 0.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	1.00 / 1.00 0.00 / 0.00
TIES	10 50 100 500	$\begin{array}{c} 0.69 \pm 0.00 \\ 0.45 \pm 0.00 \\ 0.41 \pm 0.00 \\ 0.22 \pm 0.00 \end{array}$	$\begin{array}{c} 0.57 \pm 0.02 \\ 0.41 \pm 0.00 \\ 0.40 \pm 0.00 \\ 0.26 \pm 0.00 \end{array}$	$\begin{array}{c} 0.45 \pm 0.04 \\ 0.18 \pm 0.05 \\ 0.20 \pm 0.05 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.10 \pm 0.01 \\ 0.03 \pm 0.01 \\ 0.01 \pm 0.02 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.57 \pm 0.03 \\ 0.70 \pm 0.02 \\ 0.65 \pm 0.00 \\ 0.60 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.39 \pm 0.01 \\ 0.36 \pm 0.00 \\ 0.36 \pm 0.00 \\ 0.32 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.21 \pm 0.00 \\ 0.21 \pm 0.00 \\ 0.21 \pm 0.00 \\ 0.07 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.82 \pm 0.01 \\ 0.32 \pm 0.04 \\ 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.38 \pm 0.28 \\ 0.27 \pm 0.22 \\ 0.23 \pm 0.22 \\ 0.15 \pm 0.20 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 1.00 1.00 / 1.00
SVD	SVD 2 SVD 4 SVD 8 SVD 16	$\begin{array}{c} 0.98 \pm 0.03 \\ 0.99 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.07 \pm 0.02 \\ 1.04 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.99 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.98 \pm 0.01 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.98 \pm 0.03 \\ 1.11 \pm 0.00 \\ 1.02 \pm 0.05 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.94 \pm 0.01 \\ 0.97 \pm 0.02 \\ 1.00 \pm 0.00 \\ 0.99 \pm 0.01 \end{array}$	$ \begin{vmatrix} 1.03 \pm 0.17 \\ 0.99 \pm 0.13 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.00 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{array}{c} 0.15 \pm 0.29 \\ 0.90 \pm 0.17 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array} $	$\begin{array}{c} 0.91 \pm 0.28 \\ 1.00 \pm 0.08 \\ 1.00 \pm 0.02 \\ 1.00 \pm 0.00 \end{array}$	0.88 / 0.88 0.75 / 0.75 0.50 / 0.50 0.00 / 0.00
10 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 1.02 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.01 \pm 0.02 \\ 0.98 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.96 \pm 0.01 \\ 1.02 \pm 0.02 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.11 \pm 0.11 \\ 1.11 \pm 0.00 \\ 1.11 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.89 \pm 0.03 \\ 0.93 \pm 0.01 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 1.19 \pm 0.04 \\ 1.10 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.99 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.33 \pm 0.58 \\ 0.67 \pm 0.58 \\ 0.67 \pm 0.58 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.95 \pm 0.27 \\ 0.98 \pm 0.19 \\ 0.98 \pm 0.19 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	1.00 / 0.90 1.00 / 0.80 1.00 / 0.60 1.00 / 0.20 1.00 / -0.60
10 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.06 \pm 0.01 \\ 1.04 \pm 0.01 \\ 1.03 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 0.98 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.15 \pm 0.06 \\ 1.11 \pm 0.00 \\ 1.07 \pm 0.06 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.92 \pm 0.02 \\ 0.92 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 1.17 \pm 0.04 \\ 1.02 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 1.01 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.67 \pm 0.58 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.20 \\ 1.01 \pm 0.05 \\ 1.01 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	1.00/0.90 0.99/0.79 0.97/0.57 0.88/0.07 0.50/-1.10
50 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.91 \pm 0.06 \\ 1.00 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.98 \pm 0.01 \\ 1.02 \pm 0.02 \\ 1.05 \pm 0.02 \\ 1.08 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.91 \pm 0.09 \\ 1.00 \pm 0.01 \end{vmatrix} $	$\begin{array}{c} 0.78 \pm 0.05 \\ 0.90 \pm 0.03 \\ 0.95 \pm 0.02 \\ 0.95 \pm 0.02 \\ 1.01 \pm 0.02 \end{array}$	$\begin{array}{c} 0.89 \pm 0.29 \\ 0.85 \pm 0.42 \\ 1.15 \pm 0.17 \\ 1.04 \pm 0.06 \\ 1.11 \pm 0.00 \end{array}$	$\begin{array}{c} 0.34 \pm 0.06 \\ 0.56 \pm 0.04 \\ 0.81 \pm 0.03 \\ 0.92 \pm 0.03 \\ 0.95 \pm 0.04 \end{array}$	$\begin{array}{c} 0.50 \pm 0.45 \\ 0.98 \pm 0.23 \\ 1.14 \pm 0.00 \\ 1.21 \pm 0.07 \\ 1.02 \pm 0.04 \end{array}$	$\begin{array}{c} 0.86 \pm 0.07 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.67 \pm 0.58 \\ 1.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.72 \pm 0.35 \\ 0.83 \pm 0.34 \\ 0.91 \pm 0.33 \\ 0.99 \pm 0.20 \\ 1.01 \pm 0.04 \end{array}$	1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84 1.00 / 0.68
50 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.96 \pm 0.05 \\ 1.02 \pm 0.00 \\ 1.02 \pm 0.01 \\ 1.02 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.06 \pm 0.02 \\ 1.06 \pm 0.01 \\ 1.02 \pm 0.00 \end{array}$	$\begin{array}{c} 1.00 \pm 0.01 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.95 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.87 \pm 0.01 \\ 0.92 \pm 0.03 \\ 0.96 \pm 0.02 \\ 0.99 \pm 0.02 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 1.04 \pm 0.06 \\ 1.15 \pm 0.06 \\ 1.22 \pm 0.00 \\ 1.15 \pm 0.06 \\ 1.04 \pm 0.06 \end{vmatrix} $	$\begin{array}{c} 0.31 \pm 0.08 \\ 0.73 \pm 0.04 \\ 0.94 \pm 0.01 \\ 0.92 \pm 0.01 \\ 0.99 \pm 0.00 \end{array}$	$\begin{array}{c} 0.98 \pm 0.23 \\ 1.17 \pm 0.04 \\ 1.17 \pm 0.04 \\ 1.10 \pm 0.08 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.97 \pm 0.02 \\ 1.01 \pm 0.02 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.81 \pm 0.35 \\ 0.90 \pm 0.33 \\ 0.94 \pm 0.33 \\ 1.02 \pm 0.07 \\ 1.01 \pm 0.02 \end{array}$	1.00/0.98 0.99/0.95 0.97/0.89 0.88/0.72 0.50/0.18
100 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.54 \pm 0.16 \\ 0.85 \pm 0.15 \\ 1.00 \pm 0.04 \\ 1.01 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.06 \\ 0.98 \pm 0.01 \\ 1.01 \pm 0.01 \\ 1.02 \pm 0.01 \\ 1.06 \pm 0.00 \end{array}$	$\begin{array}{c} 0.90 \pm 0.04 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.89 \pm 0.05 \\ 0.86 \pm 0.13 \\ 0.98 \pm 0.02 \\ 1.01 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.42 \pm 0.08 \\ 0.70 \pm 0.14 \\ 0.88 \pm 0.04 \\ 0.95 \pm 0.02 \\ 0.96 \pm 0.01 \end{array}$	$\begin{array}{c} 0.44 \pm 0.00 \\ 0.74 \pm 0.28 \\ 1.07 \pm 0.06 \\ 1.11 \pm 0.00 \\ 1.11 \pm 0.11 \end{array}$	$\begin{array}{c} 0.08 \pm 0.02 \\ 0.28 \pm 0.07 \\ 0.58 \pm 0.09 \\ 0.81 \pm 0.06 \\ 0.92 \pm 0.02 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 0.48 \pm 0.55 \\ 1.10 \pm 0.04 \\ 1.21 \pm 0.00 \\ 1.21 \pm 0.07 \end{array}$	$\begin{array}{c} 0.76 \pm 0.05 \\ 0.91 \pm 0.02 \\ 0.96 \pm 0.02 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.49 \pm 0.36 \\ 0.68 \pm 0.36 \\ 0.86 \pm 0.32 \\ 0.91 \pm 0.33 \\ 0.93 \pm 0.33 \end{array}$	1.00 / 0.99 1.00 / 0.98 1.00 / 0.96 1.00 / 0.92 1.00 / 0.84
100 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.85 \pm 0.03 \\ 0.99 \pm 0.02 \\ 1.02 \pm 0.00 \\ 1.01 \pm 0.01 \\ 1.01 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 0.99 \pm 0.01 \\ 1.00 \pm 0.02 \\ 1.05 \pm 0.01 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.95 \pm 0.06 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.80 \pm 0.02 \\ 0.90 \pm 0.03 \\ 0.94 \pm 0.01 \\ 0.98 \pm 0.01 \\ 1.02 \pm 0.01 \end{array}$	$ \begin{array}{c} 0.81 \pm 0.17 \\ 1.04 \pm 0.06 \\ 1.04 \pm 0.06 \\ 1.15 \pm 0.06 \\ 1.19 \pm 0.06 \end{array} $	$\begin{array}{c} 0.29 \pm 0.04 \\ 0.55 \pm 0.04 \\ 0.78 \pm 0.01 \\ 0.94 \pm 0.01 \\ 0.93 \pm 0.01 \end{array}$	$ \begin{array}{c} 0.60 \pm 0.34 \\ 1.07 \pm 0.07 \\ 1.14 \pm 0.00 \\ 1.21 \pm 0.00 \\ 1.02 \pm 0.04 \end{array} $	$\begin{array}{c} 0.87 \pm 0.02 \\ 0.96 \pm 0.01 \\ 0.99 \pm 0.02 \\ 1.01 \pm 0.00 \\ 1.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.33 \pm 0.58 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.71 \pm 0.33 \\ 0.85 \pm 0.32 \\ 0.89 \pm 0.31 \\ 0.97 \pm 0.28 \\ 1.02 \pm 0.06 \end{array}$	1.00/0.99 0.99/0.97 0.97/0.93 0.88/0.80 0.50/0.34
100 w/clusters (C)	16 C 5 16 C 7	1.13 ± 0.01 1.12 ± 0.01	1.03 ± 0.01 1.01 ± 0.01	1.00 ± 0.00 1.00 ± 0.00	1.00 ± 0.01 1.00 ± 0.00	0.97 ± 0.03 0.98 ± 0.03	1.24 ± 0.09 1.16 ± 0.11	0.88 ± 0.06 0.92 ± 0.05	1.42 ± 0.07 1.45 ± 0.04	1.05 ± 0.01 1.03 ± 0.01	0.65 ± 0.46 0.69 ± 0.49	1.04 ± 0.19 1.04 ± 0.18	1.00/0.95 1.00/0.93
500 diagonal (D)	16 D 32 D 64 D 128 D 256 D	$\begin{array}{c} 0.22 \pm 0.10 \\ 0.27 \pm 0.18 \\ 0.40 \pm 0.04 \\ 0.61 \pm 0.04 \\ 0.95 \pm 0.02 \end{array}$	$ \begin{vmatrix} 0.55 \pm 0.03 \\ 0.55 \pm 0.08 \\ 0.63 \pm 0.11 \\ 0.92 \pm 0.02 \\ 0.99 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.81 \pm 0.05 \\ 0.82 \pm 0.02 \\ 0.89 \pm 0.04 \\ 0.97 \pm 0.03 \\ 1.00 \pm 0.00 \end{array}$	$ \begin{vmatrix} 0.33 \pm 0.49 \\ 0.49 \pm 0.37 \\ 0.91 \pm 0.01 \\ 0.93 \pm 0.05 \\ 1.00 \pm 0.01 \end{vmatrix} $	$\begin{array}{c} 0.70 \pm 0.03 \\ 0.75 \pm 0.01 \\ 0.80 \pm 0.01 \\ 0.80 \pm 0.00 \\ 0.86 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.15 \pm 0.17 \\ 0.22 \pm 0.11 \\ 0.48 \pm 0.06 \\ 0.74 \pm 0.17 \\ 0.85 \pm 0.28 \end{vmatrix} $	$\begin{array}{c} 0.03 \pm 0.01 \\ 0.05 \pm 0.05 \\ 0.13 \pm 0.04 \\ 0.22 \pm 0.11 \\ 0.28 \pm 0.06 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ 0.02 \pm 0.04 \\ 0.05 \pm 0.08 \\ 0.12 \pm 0.08 \\ 0.55 \pm 0.39 \end{vmatrix} $	$\begin{array}{c} 0.76 \pm 0.06 \\ 0.79 \pm 0.04 \\ 0.82 \pm 0.02 \\ 0.85 \pm 0.05 \\ 0.92 \pm 0.02 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.35 \pm 0.35 \\ 0.39 \pm 0.34 \\ 0.51 \pm 0.35 \\ 0.61 \pm 0.36 \\ 0.73 \pm 0.36 \end{array}$	1.00 / 1.00 1.00 / 1.00 1.00 / 0.99 1.00 / 0.98 1.00 / 0.97
500 full (F)	16 F 32 F 64 F 128 F 256 F	$\begin{array}{c} 0.21 \pm 0.02 \\ 0.54 \pm 0.08 \\ 0.99 \pm 0.03 \\ 1.02 \pm 0.01 \\ 1.03 \pm 0.00 \end{array}$	$\begin{array}{c} 0.43 \pm 0.07 \\ 0.54 \pm 0.04 \\ 0.96 \pm 0.01 \\ 0.97 \pm 0.02 \\ 1.03 \pm 0.01 \end{array}$	$\begin{array}{c} 0.78 \pm 0.01 \\ 0.93 \pm 0.02 \\ 0.94 \pm 0.01 \\ 0.99 \pm 0.00 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.90 \pm 0.00 \\ 0.92 \pm 0.01 \\ 1.01 \pm 0.03 \\ 1.00 \pm 0.01 \\ 1.00 \pm 0.01 \end{array}$	$\begin{array}{c} 0.86 \pm 0.01 \\ 0.90 \pm 0.01 \\ 0.87 \pm 0.00 \\ 0.92 \pm 0.00 \\ 0.95 \pm 0.03 \end{array}$	$\begin{array}{c} 0.59 \pm 0.06 \\ 0.63 \pm 0.13 \\ 1.04 \pm 0.17 \\ 1.15 \pm 0.06 \\ 1.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.21 \pm 0.01 \\ 0.26 \pm 0.02 \\ 0.36 \pm 0.00 \\ 0.61 \pm 0.01 \\ 0.78 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.12 \pm 0.04 \\ 0.14 \pm 0.00 \\ 0.14 \pm 0.00 \\ 1.07 \pm 0.00 \\ 1.07 \pm 0.00 \end{vmatrix} $	$\begin{array}{c} 0.83 \pm 0.01 \\ 0.86 \pm 0.02 \\ 0.91 \pm 0.01 \\ 0.98 \pm 0.00 \\ 1.00 \pm 0.01 \end{array}$	$ \begin{vmatrix} 0.00 \pm 0.00 \\ \end{vmatrix} $	$\begin{array}{c} 0.50 \pm 0.34 \\ 0.57 \pm 0.34 \\ 0.71 \pm 0.39 \\ 0.87 \pm 0.33 \\ 0.88 \pm 0.31 \end{array}$	1.00/1.00 0.99/0.99 0.97/0.96 0.88/0.86 0.50/0.47
500 w/clusters (C)	16 C 7 16 C 10 16 C 25 64 C 5 64 C 7	1.08 1.10 1.10 1.09 1.13	1.00 1.01 1.00 0.98 1.02	0.99 1.00 1.00 1.00 1.00	1.00 0.98 0.99 1.00 1.01	0.92 0.91 0.97 0.96 0.98	1.01 1.01 1.12 1.12 1.12	0.39 0.37 0.81 0.83 0.90	0.62 1.51 1.42 1.33 1.42	0.98 1.02 1.03 1.04 1.04	0.00 0.00 0.00 0.00 0.00	0.80 0.89 0.95 0.94 0.96	1.00/0.98 1.00/0.98 1.00/0.95 0.97/0.93 0.97/0.91
1000 w/clusters (C)	16 C 25	1.09	0.98	1.00	1.00	0.89	1.01	0.39	1.42	1.05	0.00	0.88	1.00/0.97

Table 11: Relative In-Distribution exact match scores for various tasks and methods

Model Type	Method Type					Ta	sks					. Average	Para. Saved
wioder Type	wichiod Type	task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	l ara. Saved
	base lora	8.59 ± 0.08 0.36 ± 0.01	$\begin{array}{c} 9.15 \pm 0.00 \\ 0.17 \pm 0.00 \end{array}$	$\begin{array}{c} 2.55 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	2.88 ± 0.00 0.12 ± 0.00	2.34 ± 0.00 0.11 ± 0.00	3.46 ± 0.04 0.76 ± 0.02	6.40 ± 0.18 1.17 ± 0.07	5.55 ± 0.00 1.94 ± 0.00	8.60 ± 0.00 0.16 ± 0.00	$ \begin{vmatrix} 2.67 \pm 0.00 \\ 0.85 \pm 0.00 \end{vmatrix} $	5.19 ± 2.65 0.57 ± 0.59	1.00 / 1.00 0.00 / 0.00
SVD	SVD 2 SVD 4 SVD 8	0.32 ± 0.01 0.33 ± 0.01 0.35 ± 0.01	$\begin{array}{c} 0.15 \pm 0.00 \\ 0.16 \pm 0.00 \\ 0.17 \pm 0.00 \end{array}$	$\begin{array}{c} 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	0.12 ± 0.00 0.12 ± 0.00 0.12 ± 0.00	0.10 ± 0.00 0.11 ± 0.00 0.11 ± 0.00	$ \begin{vmatrix} 0.76 \pm 0.02 \\ 0.76 \pm 0.02 \\ 0.77 \pm 0.02 \end{vmatrix} $	1.13 ± 0.08 1.14 ± 0.08 1.16 ± 0.07	1.94 ± 0.00 1.94 ± 0.00 1.94 ± 0.00	$\begin{array}{c} 0.13 \pm 0.00 \\ 0.14 \pm 0.00 \\ 0.15 \pm 0.00 \end{array}$	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	0.57 ± 0.60 0.56 ± 0.59 0.51 ± 0.58	0.88 / 0.88 0.75 / 0.75 0.50 / 0.50
	SVD 16	0.36 ± 0.01	0.17 ± 0.00	0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.11 ± 0.00	0.76 ± 0.02	1.14 ± 0.06	1.94 ± 0.00	0.16 ± 0.00	0.85 ± 0.00	0.56 ± 0.59	0.00 / 0.00
	16 D	0.33 ± 0.01	0.15 ± 0.01	0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00	0.76 ± 0.03	1.13 ± 0.08	1.95 ± 0.01	0.14 ± 0.00	1.00 ± 0.02	0.57 ± 0.61	1.00 / 0.90
10 diagonal (D)	32 D 64 D	0.33 ± 0.01 0.35 ± 0.01	0.16 ± 0.00 0.17 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.10 ± 0.00 0.11 ± 0.00	0.75 ± 0.02 0.75 ± 0.02	1.11 ± 0.07 1.11 ± 0.07	1.93 ± 0.00 1.94 ± 0.00	0.14 ± 0.01 0.15 ± 0.00	$0.88 \pm 0.00 \\ 0.84 \pm 0.00$	0.55 ± 0.60 0.55 ± 0.59	1.00 / 0.80 1.00 / 0.60
	128 D 256 D	$\begin{array}{c} 0.35 \pm 0.01 \\ 0.36 \pm 0.01 \end{array}$	$\begin{array}{c} 0.17 \pm 0.00 \\ 0.17 \pm 0.00 \end{array}$	$\begin{array}{c} 0.01 \pm 0.00 \\ 0.01 \pm 0.00 \end{array}$	$\begin{array}{c} 0.12 \pm 0.00 \\ 0.12 \pm 0.00 \end{array}$	0.11 ± 0.00 0.11 ± 0.00	0.75 ± 0.02 0.75 ± 0.02	1.11 ± 0.07 1.12 ± 0.07	1.94 ± 0.00 1.94 ± 0.00	$\begin{array}{c} 0.16 \pm 0.00 \\ 0.16 \pm 0.00 \end{array}$	$\begin{array}{c} 0.84 \pm 0.00 \\ 0.85 \pm 0.00 \end{array}$	0.56 ± 0.59 0.56 ± 0.59	1.00 / 0.20 1.00 / -0.60
	16 F 32 F	0.33 ± 0.00 0.33 ± 0.01	0.15 ± 0.00 0.16 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.10 ± 0.00 0.10 ± 0.00	0.76 ± 0.02 0.75 ± 0.02	1.20 ± 0.02 1.11 ± 0.07	1.95 ± 0.00 1.94 ± 0.00	0.13 ± 0.00 0.14 ± 0.00	0.97 ± 0.00 0.86 ± 0.00	0.57 ± 0.61 0.55 ± 0.60	1.00/0.90 0.99/0.79
10 full (F)	64 F	0.34 ± 0.01	0.16 ± 0.00 0.16 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00 0.11 ± 0.00	0.75 ± 0.02	1.11 ± 0.07	1.94 ± 0.00 1.94 ± 0.00	0.14 ± 0.00 0.15 ± 0.00	0.84 ± 0.00	0.55 ± 0.60 0.55 ± 0.59	0.97/0.57
	128 F 256 F	0.35 ± 0.01 0.36 ± 0.01	$0.17 \pm 0.00 \\ 0.17 \pm 0.00$	0.01 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.11 ± 0.00 0.11 ± 0.00	0.75 ± 0.02 0.75 ± 0.02	1.12 ± 0.07 1.12 ± 0.07	1.94 ± 0.00 1.94 ± 0.00	0.16 ± 0.00 0.16 ± 0.00	0.84 ± 0.00 0.85 ± 0.00	0.56 ± 0.59 0.56 ± 0.59	0.88/0.07 0.50/-1.10
	16 D	0.61 ± 0.06	0.19 ± 0.02	0.03 ± 0.01	0.29 ± 0.04	0.36 ± 0.04	0.95 ± 0.05	1.73 ± 0.21	2.66 ± 0.22	0.32 ± 0.11	1.98 ± 0.01	0.91 ± 0.88	1.00 / 0.98
50 diagonal (D)	32 D 64 D	0.37 ± 0.02 0.33 ± 0.02	0.16 ± 0.00 0.15 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.19 ± 0.03 0.12 ± 0.00	0.18 ± 0.01 0.10 ± 0.00	0.85 ± 0.05 0.79 ± 0.02	1.37 ± 0.14 1.12 ± 0.08	2.12 ± 0.05 1.97 ± 0.01	0.16 ± 0.00 0.13 ± 0.01	1.65 ± 0.03 1.13 ± 0.03	0.71 ± 0.73 0.59 ± 0.63	1.00 / 0.96 1.00 / 0.92
	128 D	0.33 ± 0.01	0.15 ± 0.00	0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00	0.76 ± 0.03	1.10 ± 0.05	1.93 ± 0.01	0.14 ± 0.00	0.93 ± 0.01	0.56 ± 0.60	1.00 / 0.84
	256 D	0.34 ± 0.01	0.16 ± 0.00	0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00	0.76 ± 0.03	1.11 ± 0.05	1.93 ± 0.00	0.15 ± 0.00	0.85 ± 0.00	0.55 ± 0.59	1.00 / 0.68
	16 F 32 F	0.47 ± 0.06 0.36 ± 0.02	0.17 ± 0.00 0.16 ± 0.00	0.02 ± 0.00 0.01 ± 0.00	0.20 ± 0.02 0.14 ± 0.00	0.19 ± 0.04 0.11 ± 0.00	0.86 ± 0.03 0.80 ± 0.03	1.71 ± 0.10 1.14 ± 0.08	2.20 ± 0.04 2.00 ± 0.01	0.17 ± 0.01 0.14 ± 0.00	1.84 ± 0.07 1.32 ± 0.02	0.78 ± 0.80 0.62 ± 0.65	1.00/0.98 0.99/0.95
50 full (F)	64 F	0.33 ± 0.01	0.15 ± 0.00	0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00	0.77 ± 0.03	1.10 ± 0.06	1.94 ± 0.00	0.13 ± 0.00	1.02 ± 0.00	0.57 ± 0.61	0.97/0.89
	128 F 256 F	0.33 ± 0.01 0.35 ± 0.01	$\begin{array}{c} 0.16 \pm 0.00 \\ 0.16 \pm 0.00 \end{array}$	0.00 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.10 ± 0.00 0.11 ± 0.00	0.76 ± 0.03 0.76 ± 0.03	1.11 ± 0.05 1.11 ± 0.05	1.93 ± 0.00 1.94 ± 0.00	0.14 ± 0.00 0.15 ± 0.00	0.87 ± 0.00 0.84 ± 0.00	0.55 ± 0.60 0.55 ± 0.59	0.88/0.72 0.50/0.18
	16 D	1.69 ± 0.49	0.26 ± 0.04	0.18 ± 0.07	0.34 ± 0.02	1.01 ± 0.20	1.45 ± 0.10	3.59 ± 0.25	3.72 ± 0.72	0.44 ± 0.20	2.37 ± 0.09	1.51 ± 1.32	1.00 / 0.99
100 diagonal (D)	32 D 64 D	0.67 ± 0.24 0.39 ± 0.06	0.18 ± 0.01 0.16 ± 0.00	0.06 ± 0.05 0.01 ± 0.00	0.31 ± 0.06 0.18 ± 0.02	0.35 ± 0.08 0.14 ± 0.01	1.04 ± 0.15 0.86 ± 0.02	1.97 ± 0.13 1.39 ± 0.07	2.88 ± 0.70 2.18 ± 0.04	0.22 ± 0.01 0.17 ± 0.00	2.12 ± 0.07 1.79 ± 0.02	0.98 ± 0.98 0.73 ± 0.76	1.00 / 0.98 1.00 / 0.96
	128 D 256 D	0.32 ± 0.00 0.32 ± 0.00	0.15 ± 0.00 0.15 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.12 ± 0.00	0.10 ± 0.00 0.10 ± 0.00	0.79 ± 0.02 0.77 ± 0.02	1.19 ± 0.02 1.16 ± 0.00	2.00 ± 0.01 1.94 ± 0.00	0.14 ± 0.01 0.13 ± 0.00	1.24 ± 0.04 0.96 ± 0.01	0.61 ± 0.65 0.56 ± 0.61	1.00 / 0.92 1.00 / 0.84
	16 F	0.66 ± 0.07	0.19 ± 0.01	0.03 ± 0.01	0.25 ± 0.02 0.15 ± 0.01	0.29 ± 0.02	0.99 ± 0.07	2.50 ± 0.51	2.63 ± 0.03	0.24 ± 0.02	2.21 ± 0.08	1.00 ± 1.01	1.00/0.99
100 full (F)	32 F 64 F	0.40 ± 0.00 0.34 ± 0.01	0.17 ± 0.00 0.15 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.13 ± 0.01 0.12 ± 0.00	0.13 ± 0.01 0.11 ± 0.00	0.85 ± 0.02 0.79 ± 0.01	1.53 ± 0.12 1.23 ± 0.07	2.17 ± 0.06 1.98 ± 0.01	0.15 ± 0.01 0.15 ± 0.00	1.93 ± 0.04 1.26 ± 0.01	0.75 ± 0.80 0.61 ± 0.65	0.99/0.97 0.97/0.93
	128 F	0.32 ± 0.00	0.15 ± 0.00	0.01 ± 0.00	0.12 ± 0.00	0.10 ± 0.00	0.77 ± 0.02	1.16 ± 0.01	1.94 ± 0.00	0.13 ± 0.00	0.99 ± 0.01	0.57 ± 0.61	0.88/0.80
	256 F 16 C 5	0.33 ± 0.00 0.34 ± 0.01	0.16 ± 0.00 0.15 ± 0.00	0.00 ± 0.00 0.01 ± 0.00	0.12 ± 0.00 0.14 ± 0.01	0.10 ± 0.00 0.11 ± 0.00	0.76 ± 0.02 0.79 ± 0.02	1.15 ± 0.01 0.97 ± 0.27	1.93 ± 0.00 1.97 ± 0.00	0.14 ± 0.00 0.13 ± 0.00	0.86 ± 0.00 0.77 ± 0.25	0.56 ± 0.60 0.54 ± 0.58	0.50/0.34
100 w/clusters (C)	16 C 7	0.34 ± 0.01	0.15 ± 0.00	0.01 ± 0.00	0.13 ± 0.00	0.10 ± 0.00	0.78 ± 0.02	0.96 ± 0.27	1.96 ± 0.00	0.14 ± 0.00	0.74 ± 0.22	0.53 ± 0.57	1.00/0.93
	16 D	2.95 ± 0.28	0.73 ± 0.29	0.27 ± 0.09	0.67 ± 0.28	0.52 ± 0.07	2.06 ± 0.30	4.85 ± 0.31	3.94 ± 0.42	0.50 ± 0.05	2.50 ± 0.03	1.94 ± 1.59	1.00 / 1.00
500 diagonal (D)	32 D 64 D	2.33 ± 0.30 1.67 ± 0.18	0.62 ± 0.17 0.43 ± 0.16	0.24 ± 0.05 0.13 ± 0.04	0.50 ± 0.16 0.29 ± 0.02	0.37 ± 0.07 0.23 ± 0.02	1.86 ± 0.25 1.32 ± 0.28	4.73 ± 0.35 3.99 ± 0.36	3.81 ± 0.59 3.41 ± 0.28	0.39 ± 0.04 0.32 ± 0.05	2.46 ± 0.05 2.35 ± 0.11	1.77 ± 1.57 1.45 ± 1.39	1.00 / 1.00 1.00 / 0.99
500 unagonai (D)	128 D	1.12 ± 0.02	0.23 ± 0.00	0.04 ± 0.03	0.21 ± 0.04	0.22 ± 0.03	1.08 ± 0.06	3.05 ± 0.87	3.09 ± 0.37	0.26 ± 0.03	2.31 ± 0.04	1.19 ± 1.21	1.00 / 0.98
	256 D	0.54 ± 0.03	0.18 ± 0.01	0.01 ± 0.00	0.16 ± 0.01	0.15 ± 0.01	0.92 ± 0.08	2.42 ± 0.14	2.51 ± 0.13	0.19 ± 0.01	2.09 ± 0.02	0.94 ± 0.99	1.00 / 0.97
	16 F 32 F	2.14 ± 0.06 1.17 ± 0.07	0.70 ± 0.04 0.48 ± 0.03	0.28 ± 0.00 0.08 ± 0.04	0.27 ± 0.01 0.21 ± 0.01	0.21 ± 0.00 0.17 ± 0.00	1.14 ± 0.04 0.99 ± 0.04	3.06 ± 0.27 2.69 ± 0.10	2.71 ± 0.01 2.47 ± 0.02	0.34 ± 0.01 0.25 ± 0.02	2.21 ± 0.01 2.11 ± 0.04	1.33 ± 1.09 1.08 ± 0.99	1.00/1.00 0.99/0.99
500 full (F)	64 F	0.51 ± 0.03	0.21 ± 0.00	0.02 ± 0.00	0.17 ± 0.01	0.14 ± 0.00	0.88 ± 0.04	2.19 ± 0.14	2.34 ± 0.03	0.20 ± 0.00	1.97 ± 0.02	0.89 ± 0.91	0.97/0.96
	128 F 256 F	0.39 ± 0.01 0.32 ± 0.01	0.16 ± 0.00 0.15 ± 0.00	0.01 ± 0.00 0.01 ± 0.00	0.13 ± 0.00 0.12 ± 0.00	0.11 ± 0.00 0.10 ± 0.00	0.81 ± 0.03 0.77 ± 0.01	1.42 ± 0.07 1.18 ± 0.04	2.03 ± 0.01 1.96 ± 0.00	0.16 ± 0.00 0.14 ± 0.01	1.71 ± 0.01 1.25 ± 0.00	0.71 ± 0.74 0.61 ± 0.65	0.88/0.86 0.50/0.47
	16 C 7	0.40	0.18	0.01	0.15	0.13	0.90	2.03	2.21	0.16	1.50	0.77	1.00/0.98
500 (1) (2)	16 C 10	0.36	0.16	0.01	0.14	0.13	0.87	2.19	2.04	0.15	1.38	0.74	1.00/0.98
500 w/clusters (C)	16 C 25 64 C 5	0.32 0.36	0.16 0.16	0.01 0.01	0.13 0.12	0.10 0.10	0.81 0.80	1.28 1.17	1.96 1.98	0.12 0.14	1.07 1.17	0.60 0.60	1.00/0.95 0.97/0.93
	64 C 7	0.34	0.15	0.01	0.12	0.10	0.79	1.14	1.96	0.13	1.08	0.58	0.97/0.91
1000 w/clusters (C)	16 C 25	0.37	0.16	0.01	0.13	0.13	0.86	2.12	2.04	0.14	1.35	0.73	1.00/0.97

Table 12: Absolute In-Distribution test loss for various tasks and methods

	l	<u> </u>				Tas	sks						<u> </u>
Model Type	Method Type	task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average	Para. Saved
	base lora	$\begin{array}{c} 0.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	0.00 ± 0.00 100.00 ± 0.00	$\begin{array}{c} 1.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	$\begin{array}{c} 0.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	0.00 ± 0.00 100.00 ± 0.00	0.00 ± 0.00 100.00 ± 0.00	$\begin{array}{c} 0.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	0.00 ± 0.00 100.00 ± 0.00	$\begin{array}{c} 0.00 \pm 0.00 \\ 100.00 \pm 0.00 \end{array}$	0.10 ± 0.30 100.00 ± 0.00	1.00 / 1.00 0.00 / 0.00
TIES	10 50 100	41.00 ± 0.00 24.00 ± 0.00 22.00 ± 0.00	53.67 ± 0.58 38.67 ± 0.58 38.00 ± 0.00	44.33 ± 4.04 17.67 ± 4.62 18.67 ± 4.62	10.33 ± 0.58 2.00 ± 0.00 1.00 ± 1.73	46.33 ± 4.04 56.33 ± 0.58 51.67 ± 4.62	1.00 ± 0.00 1.00 ± 0.00 1.00 ± 0.00	8.00 ± 0.00 8.00 ± 0.00 8.00 ± 0.00	8.00 ± 0.00 8.00 ± 0.00 7.33 ± 0.58	76.67 ± 1.15 29.67 ± 2.89 2.00 ± 0.00	1.00 ± 0.00 0.00 ± 0.00 0.00 ± 0.00	29.03 ± 25.69 18.53 ± 18.07 14.97 ± 17.20	1.00 / 1.00 1.00 / 1.00 1.00 / 1.00
TIES	500	8.00 ± 0.00	25.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	59.00 ± 0.00	0.00 ± 0.00	3.00 ± 0.00	6.00 ± 0.00	2.00 ± 0.00 2.00 ± 0.00	0.00 ± 0.00 0.00 ± 0.00	9.90 ± 18.12	1.00 / 1.00
SVD	SVD 2 SVD 4	88.33 ± 0.65 93.00 ± 0.00	91.91 ± 0.94 96.64 ± 0.50	$100.00 \pm 0.00 100.00 \pm 0.00$	97.25 ± 0.45 100.00 ± 0.00	92.83 ± 0.39 96.75 ± 0.87	76.50 ± 1.51 88.83 ± 1.53	66.00 ± 1.41 90.67 ± 1.23	58.08 ± 1.16 72.17 ± 0.58	98.67 ± 0.49 98.67 ± 0.49	5.83 ± 0.94 16.67 ± 1.78	77.42 ± 27.69 85.24 ± 24.39	0.88 / 0.88 0.75 / 0.75
	SVD 8 SVD 16	98.89 ± 0.60 100.00 ± 0.00	98.55 ± 0.52 100.00 ± 0.00	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	$100.00 \pm 0.00 100.00 \pm 0.00$	99.42 ± 0.51 100.00 ± 0.00	93.44 ± 0.73 99.67 ± 0.50	97.22 ± 0.44 99.50 ± 0.55	82.78 ± 1.64 99.67 ± 0.50	99.00 ± 0.00 100.00 ± 0.00	$60.00 \pm 0.87 \\ 98.11 \pm 0.78$	93.70 ± 11.59 99.69 ± 0.68	0.50 / 0.50 0.00 / 0.00
	16 D 32 D	83.33 ± 1.53 93.00 ± 1.00	88.33 ± 0.58 95.33 ± 0.58	100.00 ± 0.00 100.00 ± 0.00	97.00 ± 2.00 98.00 ± 1.00	88.33 ± 1.15 93.67 ± 1.53	57.00 ± 1.00 80.67 ± 2.31	48.67 ± 3.21 78.67 ± 1.15	50.67 ± 4.93 68.00 ± 1.73	97.67 ± 1.53 98.33 ± 0.58	5.33 ± 1.15 14.67 ± 2.52	71.63 ± 29.53 82.03 ± 24.99	1.00 / 0.90 1.00 / 0.80
10 diagonal (D)	64 D 128 D 256 D	99.00 ± 0.00 100.00 ± 0.00	97.00 ± 1.00 99.33 ± 0.58	100.00 ± 0.00 100.00 ± 0.00	100.00 ± 0.00 100.00 ± 0.00	98.00 ± 0.00 100.00 ± 0.00 100.00 ± 0.00	90.67 ± 1.53 96.67 ± 1.53 100.00 ± 0.00	95.33 ± 1.15 98.33 ± 1.15	79.67 ± 1.53 95.67 ± 2.31	99.00 ± 0.00 100.00 ± 0.00	55.00 ± 4.36 91.67 ± 3.51	91.37 ± 13.78 98.17 ± 2.94	1.00 / 0.60 1.00 / 0.20 1.00 / -0.60
	16 F	100.00 ± 0.00 83.00 ± 2.00	100.00 ± 0.00 93.00 ± 1.00	100.00 ± 0.00 100.00 ± 0.00	100.00 ± 0.00 98.33 ± 0.58	91.67 ± 0.58	64.33 ± 3.21	100.00 ± 0.00 59.33 ± 1.15	99.33 ± 1.15 52.67 ± 1.53	100.00 ± 0.00 98.33 ± 0.58	95.00 ± 1.00 6.33 ± 1.15	99.43 ± 1.57 74.70 ± 28.71	1.00/0.90
10 full (F)	32 F 64 F 128 F	91.33 ± 0.58 99.00 ± 0.00	96.00 ± 1.00 97.33 ± 0.58	100.00 ± 0.00 100.00 ± 0.00	98.33 ± 0.58 100.00 ± 0.00	94.33 ± 0.58 99.33 ± 1.15	84.00 ± 2.00 91.33 ± 1.53	83.00 ± 1.73 96.33 ± 0.58	70.33 ± 1.53 81.67 ± 2.31	99.00 ± 1.00 99.00 ± 0.00	22.00 ± 2.65 58.33 ± 1.53	83.83 ± 22.82 92.23 ± 12.76	0.99/0.79 0.97/0.57
	256 F	$\begin{array}{c} 99.67 \pm 0.58 \\ 100.00 \pm 0.00 \end{array}$	$\begin{array}{c} 99.33 \pm 0.58 \\ 100.00 \pm 0.00 \end{array}$	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	97.67 ± 1.15 99.67 ± 0.58	100.00 ± 0.00 99.67 ± 0.58	95.67 ± 1.15 99.67 ± 0.58	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	$\begin{array}{c} 91.00 \pm 1.00 \\ 98.00 \pm 1.00 \end{array}$	98.33 ± 2.89 99.70 ± 0.70	0.88/0.07 0.50/-1.10
	16 D 32 D	52.67 ± 4.51 69.67 ± 3.21	86.67 ± 3.06 88.67 ± 1.53	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	85.00 ± 3.46 95.00 ± 2.00	65.33 ± 3.79 80.00 ± 3.00	25.33 ± 5.03 36.67 ± 5.13	10.00 ± 1.00 17.00 ± 2.65	10.67 ± 10.26 26.33 ± 5.03	81.00 ± 6.56 95.00 ± 2.00	0.00 ± 0.00 0.00 ± 0.00	51.67 ± 36.18 60.83 ± 36.02	1.00 / 0.98 1.00 / 0.96
50 diagonal (D)	64 D 128 D 256 D	79.67 ± 2.52 90.00 ± 1.00 94.67 ± 0.58	91.00 ± 1.00 91.33 ± 0.58 96.33 ± 0.58	100.00 ± 0.00 100.00 ± 0.00 100.00 ± 0.00	97.67 ± 0.58 98.33 ± 0.58 99.67 ± 0.58	88.00 ± 1.00 90.67 ± 2.08 96.33 ± 1.15	52.00 ± 1.00 73.67 ± 2.08 87.33 ± 0.58	36.67 ± 5.69 63.67 ± 1.53 87.00 ± 2.65	41.33 ± 1.15 56.33 ± 0.58 71.67 ± 1.53	96.00 ± 1.00 98.00 ± 0.00 99.67 ± 0.58	0.33 ± 0.58 7.33 ± 1.15 31.67 ± 1.15	68.27 ± 32.66 76.93 ± 27.79 86.43 ± 20.41	1.00 / 0.92 1.00 / 0.84 1.00 / 0.68
	16 F 32 F	61.67 ± 3.06 71.00 ± 1.00	89.67 ± 1.15 89.00 ± 1.73	99.67 ± 0.58 100.00 ± 0.00	90.67 ± 2.52 98.00 ± 0.00	78.33 ± 3.51 85.00 ± 1.00	34.00 ± 1.00 47.00 ± 1.73	7.00 ± 3.46 29.00 ± 3.00	25.00 ± 6.24 35.00 ± 2.00	90.00 ± 1.00 98.00 ± 1.00	0.00 ± 0.00 0.00 ± 0.00	57.60 ± 36.59 65.20 ± 34.00	1.00/0.98
50 full (F)	64 F 128 F	81.67 ± 0.58 91.00 ± 1.00	93.67 ± 1.15 94.33 ± 0.58	100.00 ± 0.00 100.00 ± 0.00 100.00 ± 0.00	98.33 ± 0.58 99.00 ± 0.00	90.67 ± 2.08 93.33 ± 1.53	61.67 ± 1.53 81.67 ± 0.58	54.33 ± 1.53 75.00 ± 1.73	51.33 ± 1.15 67.67 ± 2.08	98.33 ± 0.58 98.67 ± 0.58	3.33 ± 0.58 16.67 ± 0.58	73.33 ± 29.89 81.73 ± 24.46	0.99/0.93 0.97/0.89 0.88/0.72
	256 F	97.00 ± 0.00	98.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	99.67 ± 0.58	92.00 ± 0.00	94.33 ± 1.15	79.67 ± 1.53	99.00 ± 0.00	57.33 ± 2.52	91.70 ± 13.11	0.50/0.18
400 11 1 (70)	16 D 32 D	33.00 ± 8.19 51.00 ± 7.81	79.33 ± 5.69 90.00 ± 1.00	89.33 ± 5.51 100.00 ± 0.00	80.00 ± 3.61 88.00 ± 7.00	35.33 ± 6.03 58.67 ± 11.68	4.00 ± 1.73 17.67 ± 10.26	3.00 ± 1.00 7.67 ± 2.89	0.00 ± 0.00 9.33 ± 12.86	71.33 ± 4.51 86.33 ± 2.52	0.00 ± 0.00 0.00 ± 0.00	39.53 ± 36.15 50.87 ± 38.43	1.00 / 0.99 1.00 / 0.98
100 diagonal (D)	64 D 128 D 256 D	68.00 ± 2.65 82.00 ± 2.00 90.00 ± 1.00	87.33 ± 1.53 90.00 ± 2.00 93.00 ± 2.00	100.00 ± 0.00 100.00 ± 0.00 100.00 ± 0.00	94.33 ± 4.04 97.33 ± 0.58 97.67 ± 0.58	80.33 ± 2.08 85.33 ± 0.58 91.67 ± 0.58	38.00 ± 3.00 55.33 ± 2.08 71.67 ± 5.13	19.67 ± 4.51 34.33 ± 3.79 59.67 ± 1.53	28.33 ± 1.53 36.67 ± 2.52 58.00 ± 0.00	92.67 ± 1.15 94.67 ± 0.58 97.67 ± 0.58	0.33 ± 0.58 0.00 ± 0.00 4.00 ± 1.00	60.90 ± 34.91 67.57 ± 32.95 76.33 ± 28.88	1.00 / 0.96 1.00 / 0.92 1.00 / 0.84
	16 F 32 F	49.00 ± 2.00 65.00 ± 3.46	89.67 ± 3.21 90.33 ± 1.53	97.00 ± 3.00 100.00 ± 0.00	84.33 ± 3.06 96.33 ± 1.53	65.33 ± 2.52 80.00 ± 2.65	20.67 ± 8.33 41.33 ± 3.21	6.33 ± 2.08 16.00 ± 0.00	8.33 ± 4.73 29.33 ± 2.08	81.33 ± 2.08 92.00 ± 2.65	0.00 ± 0.00 0.00 ± 0.00	50.20 ± 37.06 61.03 ± 35.43	1.00/0.99 0.99/0.97
100 full (F)	64 F 128 F	72.33 ± 0.58 84.33 ± 1.53	89.67 ± 1.53 92.33 ± 1.53	$100.00 \pm 0.00 \\ 100.00 \pm 0.00$	97.67 ± 0.58 98.00 ± 0.00	86.00 ± 1.00 91.33 ± 0.58	53.00 ± 1.00 68.67 ± 0.58	35.33 ± 1.53 56.00 ± 1.00	38.00 ± 1.73 57.67 ± 1.15	94.67 ± 0.58 99.00 ± 0.00	0.00 ± 0.00 5.33 ± 0.58	66.67 ± 32.54 75.27 ± 28.67	0.97/0.93 0.88/0.80
100 (1) (0)	256 F	91.67 ± 1.15 74.67 ± 0.94	96.67 ± 0.58 91.00 ± 0.82	100.00 ± 0.00 100.00 ± 0.00	100.00 ± 0.00 96.67 ± 1.25	94.33 ± 0.58 87.67 ± 1.70	84.67 ± 0.58 53.67 ± 2.05	78.00 ± 0.00 40.67 ± 2.87	69.67 ± 0.58 41.00 ± 4.55	99.00 ± 0.00 97.67 ± 1.25	22.00 ± 1.00 0.67 ± 0.94	83.60 ± 23.08 68.37 ± 31.46	1.00/0.95
100 w/clusters (C)	16 C 7	77.67 ± 0.47	90.33 ± 1.25 51.50 ± 3.54	100.00 ± 0.00 79.67 ± 4.93	97.33 ± 0.94 28.00 ± 42.44	90.33 ± 2.05 56.67 ± 2.89	58.67 ± 0.94	49.00 ± 2.16	49.00 ± 0.82	97.67 ± 0.47	3.67 ± 1.25	71.37 ± 29.48 28.90 ± 33.54	1.00/0.93
500 diagonal (D)	32 D 64 D	14.33 ± 11.02 25.67 ± 1.15	51.50 ± 3.54 52.50 ± 9.19 62.50 ± 12.02	79.67 ± 4.93 80.67 ± 2.08 87.33 ± 4.04	28.00 ± 42.44 43.00 ± 31.48 78.33 ± 1.15	60.67 ± 2.89 60.67 ± 0.58 65.33 ± 3.06	0.67 ± 1.15 0.67 ± 1.15 5.33 ± 3.51	1.67 ± 1.15 3.67 ± 1.15	0.00 ± 0.00 0.00 ± 0.00 1.00 ± 1.73	74.33 ± 4.04 76.67 ± 2.31	0.00 ± 0.00 0.00 ± 0.00 0.00 ± 0.00	32.10 ± 33.43 39.83 ± 35.80	1.00 / 1.00 1.00 / 1.00 1.00 / 0.99
300 diagonai (B)	128 D 256 D	38.33 ± 3.21 53.33 ± 0.58	85.50 ± 2.12 91.00 ± 2.83	96.00 ± 3.00 100.00 ± 0.00	81.33 ± 2.31 89.00 ± 2.65	65.67 ± 1.15 76.00 ± 2.00	11.67 ± 4.73 20.67 ± 6.81	5.33 ± 2.08 6.00 ± 1.73	2.00 ± 1.00 12.00 ± 8.00	80.00 ± 5.00 86.33 ± 2.31	0.00 ± 0.00 0.00 ± 0.00	45.24 ± 37.78 52.14 ± 38.64	1.00 / 0.98 1.00 / 0.97
	16 F 32 F	8.33 ± 2.08 33.67 ± 4.16	41.00 ± 5.66 51.00 ± 1.41	76.67 ± 0.58 92.67 ± 1.53	78.00 ± 0.00 77.00 ± 1.73	72.67 ± 0.58 75.00 ± 2.00	6.00 ± 0.00 14.33 ± 1.53	5.67 ± 0.58 8.00 ± 0.00	0.00 ± 0.00 0.00 ± 0.00	78.00 ± 1.00 80.67 ± 1.53	0.00 ± 0.00 0.00 ± 0.00	36.48 ± 35.46 42.97 ± 35.80	1.00/1.00 0.99/0.99
500 full (F)	64 F 128 F 256 F	56.00 ± 2.65 69.33 ± 0.58 79.67 ± 0.58	85.50 ± 0.71 88.50 ± 0.71 89.50 ± 0.71	94.33 ± 0.58 99.00 ± 0.00 100.00 ± 0.00	89.33 ± 2.89 96.33 ± 1.53 97.67 ± 0.58	74.33 ± 1.15 80.33 ± 1.15 87.33 ± 0.58	36.33 ± 1.15 45.00 ± 2.00 57.00 ± 1.00	9.00 ± 1.00 16.33 ± 0.58 35.00 ± 1.00	2.67 ± 1.15 31.00 ± 1.73 42.00 ± 1.00	84.00 ± 1.00 92.00 ± 0.00 95.00 ± 1.00	$\begin{array}{c} 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \\ 0.00 \pm 0.00 \end{array}$	52.03 ± 36.92 60.86 ± 35.07 67.59 ± 32.67	0.97/0.96 0.88/0.86 0.50/0.47
	16 C 7 16 C 10	63.00 69.00	90.00	99.00 100.00	96.00 98.00	78.00 81.00	31.00 34.00	9.00 8.00	15.00	89.00 95.00	1.00 1.00	57.10 61.20	1.00/0.98
500 w/clusters (C)	16 C 25 64 C 5	79.00 77.00	90.00 88.00	100.00 100.00	97.00 98.00	88.00 89.00	53.00 56.00	38.00 39.00	48.00 42.00	98.00 99.00	0.00	69.10 68.80	1.00/0.95 0.97/0.93
1000 w/clusters (C)	64 C 7	76.00	90.00	100.00	97.00	89.00 77.00	60.00	48.00	49.00	99.00	3.00	71.10	0.97/0.91
1000 W/CIUSICIS (C)	10025	75.00	70.00	100.00	70.00	77.00	39.00	1 0.00	J4.00	70.00	1.00	01.00	1.00/0.5/

Table 13: Absolute In-Distribution agreement for various tasks and methods

Model Type	Method Type					Ta	sks					Average
Woder Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	Average
	SVD 2	0.29 ± 0.00	0.43 ± 0.00	0.31 ± 0.00	0.40 ± 0.00	0.38 ± 0.00	0.31 ± 0.00	0.37 ± 0.00	0.31 ± 0.00	0.42 ± 0.00	0.30 ± 0.00	0.35 ± 0.05
SVD	SVD 4 SVD 8	0.16 ± 0.00 0.06 ± 0.00	0.24 ± 0.00 0.09 ± 0.00	0.16 ± 0.00 0.06 ± 0.00	0.25 ± 0.00 0.11 ± 0.00	0.23 ± 0.00 0.10 ± 0.00	0.17 ± 0.00 0.07 ± 0.00	0.22 ± 0.00 0.09 ± 0.00	0.16 ± 0.00 0.06 ± 0.00	0.25 ± 0.00 0.11 ± 0.00	0.16 ± 0.00 0.06 ± 0.00	0.20 ± 0.04 0.08 ± 0.02
	1				<u> </u>		I	I		l		
	16 D 32 D	0.37 ± 0.02 0.21 ± 0.01	0.51 ± 0.02 0.28 ± 0.00	0.36 ± 0.01 0.20 ± 0.01	0.57 ± 0.02 0.35 ± 0.00	0.55 ± 0.00 0.33 ± 0.01	0.39 ± 0.02 0.22 ± 0.01	0.49 ± 0.01 0.31 ± 0.01	0.36 ± 0.02 0.20 ± 0.01	0.53 ± 0.03 0.32 ± 0.01	0.39 ± 0.01 0.22 ± 0.00	0.45 ± 0.08 0.26 ± 0.06
10 diagonal (D)	64 D	0.21 ± 0.01 0.10 ± 0.00	0.28 ± 0.00 0.11 ± 0.01	0.20 ± 0.01 0.09 ± 0.00	0.33 ± 0.00 0.18 ± 0.01	0.33 ± 0.01 0.18 ± 0.00	0.22 ± 0.01 0.10 ± 0.00	0.31 ± 0.01 0.15 ± 0.01	0.20 ± 0.01 0.09 ± 0.00	0.32 ± 0.01 0.14 ± 0.00	0.22 ± 0.00 0.09 ± 0.00	0.20 ± 0.06 0.12 ± 0.04
g (=)	128 D	0.02 ± 0.00	0.01 ± 0.00	0.02 ± 0.00	0.03 ± 0.00	0.04 ± 0.00	0.02 ± 0.00	0.03 ± 0.00	0.02 ± 0.00	0.02 ± 0.00	0.02 ± 0.00	0.03 ± 0.01
	256 D	0.00 ± 0.00										
	16 F	0.35 ± 0.00	0.46 ± 0.00	0.34 ± 0.00	0.51 ± 0.00	0.47 ± 0.01	0.36 ± 0.01	0.45 ± 0.01	0.35 ± 0.01	0.49 ± 0.00	0.35 ± 0.01	0.41 ± 0.06
10 full (F)	32 F 64 F	0.20 ± 0.00 0.10 ± 0.00	0.24 ± 0.00 0.10 ± 0.00	0.20 ± 0.00 0.09 ± 0.00	0.30 ± 0.00 0.13 ± 0.00	0.29 ± 0.00 0.13 ± 0.00	0.22 ± 0.00 0.10 ± 0.00	0.27 ± 0.00 0.12 ± 0.00	0.20 ± 0.00 0.09 ± 0.00	0.27 ± 0.00 0.12 ± 0.00	0.21 ± 0.00 0.10 ± 0.00	0.24 ± 0.04 0.11 ± 0.02
10 Iuli (1)	128 F	0.10 ± 0.00 0.02 ± 0.00	0.10 ± 0.00 0.02 ± 0.00	0.02 ± 0.00	0.01 ± 0.00	0.13 ± 0.00 0.02 ± 0.00	0.10 ± 0.00 0.02 ± 0.00	0.12 ± 0.00 0.02 ± 0.00	0.02 ± 0.00	0.12 ± 0.00 0.01 ± 0.00	0.10 ± 0.00 0.02 ± 0.00	0.02 ± 0.02
	256 F	0.00 ± 0.00	0.00 ± 0.00									
	16 D	0.66 ± 0.01	0.69 ± 0.01	0.88 ± 0.01	0.76 ± 0.03	0.95 ± 0.02	0.91 ± 0.01	0.83 ± 0.02	0.88 ± 0.03	0.72 ± 0.02	0.88 ± 0.02	0.82 ± 0.10
	32 D	0.50 ± 0.01	0.52 ± 0.02	0.73 ± 0.01	0.58 ± 0.03	0.88 ± 0.03	0.79 ± 0.03	0.72 ± 0.01	0.75 ± 0.01	0.57 ± 0.02	0.75 ± 0.01	0.68 ± 0.12
50 diagonal (D)	64 D 128 D	0.34 ± 0.01	0.37 ± 0.01 0.22 ± 0.01	0.52 ± 0.00	0.38 ± 0.01	0.71 ± 0.02	0.58 ± 0.01	0.54 ± 0.00	0.56 ± 0.00 0.39 ± 0.00	0.44 ± 0.01	0.58 ± 0.01	0.50 ± 0.11
	256 D	0.21 ± 0.01 0.10 ± 0.00	0.22 ± 0.01 0.12 ± 0.00	0.31 ± 0.00 0.16 ± 0.00	0.22 ± 0.00 0.10 ± 0.00	0.51 ± 0.01 0.29 ± 0.01	0.42 ± 0.01 0.21 ± 0.00	0.38 ± 0.00 0.19 ± 0.00	0.39 ± 0.00 0.23 ± 0.01	0.27 ± 0.00 0.15 ± 0.00	0.40 ± 0.00 0.20 ± 0.00	0.33 ± 0.10 0.18 ± 0.06
	16 F	0.57 ± 0.01	0.60 ± 0.01	0.86 ± 0.01	0.71 ± 0.02	0.95 ± 0.01	0.88 ± 0.01	0.81 ± 0.00	0.83 ± 0.01	0.67 ± 0.01	0.86 ± 0.01	0.78 ± 0.12
	32 F	0.47 ± 0.01	0.48 ± 0.01	0.71 ± 0.00	0.71 ± 0.02 0.55 ± 0.01	0.78 ± 0.01	0.69 ± 0.01	0.69 ± 0.00	0.65 ± 0.01	0.53 ± 0.01	0.71 ± 0.00	0.63 ± 0.11
50 full (F)	64 F	0.33 ± 0.00	0.35 ± 0.00	0.45 ± 0.00	0.36 ± 0.00	0.56 ± 0.00	0.50 ± 0.01	0.47 ± 0.00	0.49 ± 0.00	0.39 ± 0.01	0.49 ± 0.00	0.44 ± 0.08
	128 F	0.19 ± 0.00	0.21 ± 0.00	0.25 ± 0.00	0.19 ± 0.00	0.35 ± 0.00	0.30 ± 0.00	0.28 ± 0.00	0.31 ± 0.00	0.24 ± 0.00	0.30 ± 0.00	0.26 ± 0.05
	256 F	0.09 ± 0.00	0.10 ± 0.00	0.10 ± 0.00	0.08 ± 0.00	0.16 ± 0.00	0.13 ± 0.00	0.12 ± 0.00	0.15 ± 0.00	0.11 ± 0.00	0.13 ± 0.00	0.12 ± 0.02
	16 D	0.90 ± 0.01	0.85 ± 0.01	0.87 ± 0.03	0.88 ± 0.02	0.68 ± 0.02	0.91 ± 0.01	0.97 ± 0.01	0.98 ± 0.01	0.96 ± 0.01	1.00 ± 0.00	0.90 ± 0.09
100 diagonal (D)	32 D 64 D	0.83 ± 0.02 0.67 ± 0.01	0.77 ± 0.00 0.63 ± 0.00	0.77 ± 0.01 0.59 ± 0.02	0.78 ± 0.00 0.63 ± 0.01	0.55 ± 0.02 0.40 ± 0.00	0.79 ± 0.01 0.62 ± 0.01	0.94 ± 0.02 0.86 ± 0.02	0.94 ± 0.03 0.82 ± 0.02	0.87 ± 0.00 0.71 ± 0.03	0.98 ± 0.01 0.93 ± 0.00	0.82 ± 0.12 0.68 ± 0.15
roo unagonar (D)	128 D	0.49 ± 0.01	0.47 ± 0.00	0.42 ± 0.01	0.45 ± 0.00	0.40 ± 0.00 0.27 ± 0.02	0.44 ± 0.01	0.73 ± 0.01	0.69 ± 0.02	0.59 ± 0.02	0.80 ± 0.02	0.53 ± 0.16
	256 D	0.32 ± 0.00	0.31 ± 0.00	0.26 ± 0.01	0.30 ± 0.00	0.15 ± 0.01	0.28 ± 0.00	0.51 ± 0.02	0.51 ± 0.02	0.40 ± 0.01	0.61 ± 0.01	0.36 ± 0.14
	16 F	0.88 ± 0.00	0.82 ± 0.00	0.84 ± 0.01	0.86 ± 0.00	0.67 ± 0.01	0.88 ± 0.01	0.99 ± 0.00	0.96 ± 0.01	0.91 ± 0.01	1.00 ± 0.00	0.88 ± 0.09
100 f.·II (T)	32 F 64 F	0.78 ± 0.00	0.72 ± 0.00	0.73 ± 0.00	0.74 ± 0.00	0.52 ± 0.00	0.74 ± 0.01	0.94 ± 0.01	0.89 ± 0.00	0.77 ± 0.02	0.99 ± 0.00	0.78 ± 0.13
100 full (F)	128 F	0.60 ± 0.00 0.40 ± 0.00	0.57 ± 0.00 0.38 ± 0.00	0.57 ± 0.00 0.35 ± 0.00	0.57 ± 0.00 0.37 ± 0.00	0.39 ± 0.00 0.25 ± 0.00	0.56 ± 0.00 0.37 ± 0.00	0.76 ± 0.00 0.52 ± 0.00	0.73 ± 0.00 0.54 ± 0.00	0.60 ± 0.00 0.45 ± 0.00	0.83 ± 0.01 0.60 ± 0.00	0.62 ± 0.12 0.42 ± 0.10
	256 F	0.21 ± 0.00	0.20 ± 0.00	0.18 ± 0.00	0.19 ± 0.00	0.13 ± 0.00	0.19 ± 0.00	0.30 ± 0.00	0.34 ± 0.00	0.26 ± 0.00	0.38 ± 0.00	0.24 ± 0.08
100 w/clusters (C)	16 C 5	0.46 ± 0.00	0.46 ± 0.00	0.45 ± 0.00	0.47 ± 0.01	0.61 ± 0.01	0.65 ± 0.01	0.61 ± 0.00	0.64 ± 0.02	0.45 ± 0.00	0.59 ± 0.01	0.54 ± 0.08
100 W/Clusters (C)	16 C 7	0.41 ± 0.01	0.42 ± 0.01	0.39 ± 0.01	0.43 ± 0.01	0.51 ± 0.01	0.56 ± 0.01	0.53 ± 0.01	0.55 ± 0.00	0.42 ± 0.01	0.54 ± 0.01	0.48 ± 0.06
	16 D	0.97 ± 0.00	0.73 ± 0.00	0.96 ± 0.00	1.00 ± 0.00	0.99 ± 0.01	0.96 ± 0.01	0.90 ± 0.00	0.92 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.94 ± 0.08
500 II 1 (D)	32 D	0.96 ± 0.00	0.70 ± 0.00	0.92 ± 0.01	0.98 ± 0.01	0.96 ± 0.01	0.93 ± 0.01	0.86 ± 0.00	0.89 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.92 ± 0.09
500 diagonal (D)	64 D 128 D	0.90 ± 0.01 0.82 ± 0.01	0.65 ± 0.00 0.60 ± 0.00	0.86 ± 0.01 0.76 ± 0.00	0.96 ± 0.02 0.90 ± 0.02	0.90 ± 0.01 0.83 ± 0.01	0.87 ± 0.01 0.78 ± 0.02	0.81 ± 0.00 0.74 ± 0.00	0.83 ± 0.01 0.74 ± 0.01	0.99 ± 0.01 0.97 ± 0.01	1.00 ± 0.00 1.00 ± 0.00	0.88 ± 0.10 0.81 ± 0.12
	256 D	0.52 ± 0.01 0.59 ± 0.02	0.50 ± 0.00 0.51 ± 0.00	0.76 ± 0.00 0.56 ± 0.01	0.81 ± 0.02	0.70 ± 0.02	0.76 ± 0.02 0.55 ± 0.01	0.57 ± 0.00	0.74 ± 0.01 0.54 ± 0.01	0.91 ± 0.01	1.00 ± 0.00 1.00 ± 0.01	0.67 ± 0.12 0.67 ± 0.17
	16 F	0.94 ± 0.00	0.67 ± 0.00	0.88 ± 0.00	1.00 ± 0.00	0.98 ± 0.00	0.90 ± 0.00	0.82 ± 0.00	0.83 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.90 ± 0.10
	32 F	0.88 ± 0.00	0.61 ± 0.00	0.81 ± 0.00	0.97 ± 0.01	0.94 ± 0.00	0.84 ± 0.00	0.75 ± 0.00	0.77 ± 0.00	0.99 ± 0.00	0.99 ± 0.00	0.86 ± 0.12
500 full (F)	64 F 128 F	0.80 ± 0.00	0.55 ± 0.00 0.46 ± 0.00	0.72 ± 0.00	0.86 ± 0.00	0.82 ± 0.01	0.76 ± 0.00	0.67 ± 0.00	0.70 ± 0.00	0.94 ± 0.00	0.99 ± 0.00	0.78 ± 0.13 0.67 ± 0.14
	256 F	0.64 ± 0.00 0.43 ± 0.00	0.46 ± 0.00 0.35 ± 0.00	0.60 ± 0.00 0.44 ± 0.00	0.74 ± 0.00 0.55 ± 0.00	0.65 ± 0.00 0.49 ± 0.00	0.63 ± 0.00 0.45 ± 0.00	0.56 ± 0.00 0.40 ± 0.00	0.58 ± 0.00 0.42 ± 0.00	0.85 ± 0.00 0.67 ± 0.00	0.96 ± 0.00 0.84 ± 0.00	0.67 ± 0.14 0.50 ± 0.14
-	16 C 7	0.68	0.70	0.64	0.72	0.85	0.90	0.93	0.92	0.71	0.83	0.79
	16 C 10	0.61	0.65	0.61	0.66	0.84	0.86	0.88	0.84	0.62	0.76	0.73
500 w/clusters (C)	16 C 25	0.42	0.41	0.42	0.44	0.57	0.64	0.63	0.62	0.40	0.58	0.51
	64 C 5 64 C 7	0.49 0.45	0.49 0.45	0.45 0.41	0.51 0.45	0.64 0.56	0.66 0.58	0.62 0.55	0.67 0.59	0.50 0.44	0.65 0.57	0.57 0.51
1000/-lu-tan (C)	16 C 25	0.58	0.64	0.54	0.64	0.81	0.87	0.90	0.84	0.57	0.74	0.71
1000 w/clusters (C)	16 C 23	0.38	0.04	0.34	0.04	0.81	0.87	0.90	0.84	0.37	0.74	0.71

Table 14: Reconstruction error In-Distribution for various tasks and methods

Model Type	Method Type		Tasks									
		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	. Average
	16 F	0.46 ± 0.01	0.63 ± 0.00	0.50 ± 0.00	0.55 ± 0.01	0.50 ± 0.00	0.49 ± 0.01	0.50 ± 0.01	0.50 ± 0.01	0.61 ± 0.01	0.47 ± 0.01	0.52 ± 0.06
	32 F	0.30 ± 0.01	0.37 ± 0.00	0.31 ± 0.00	0.35 ± 0.00	0.34 ± 0.00	0.31 ± 0.00	0.33 ± 0.00	0.31 ± 0.00	0.38 ± 0.00	0.30 ± 0.00	0.33 ± 0.03
10 full (F)	64 F	0.15 ± 0.00	0.15 ± 0.00	0.16 ± 0.00	0.17 ± 0.00	0.17 ± 0.00	0.16 ± 0.00	0.16 ± 0.00	0.16 ± 0.00	0.17 ± 0.00	0.15 ± 0.00	0.16 ± 0.01
	16 F	0.80 ± 0.02	0.82 ± 0.01	0.85 ± 0.01	0.90 ± 0.02	0.78 ± 0.01	0.95 ± 0.01	0.76 ± 0.01	0.75 ± 0.00	0.79 ± 0.01	0.82 ± 0.01	0.82 ± 0.06
	32 F	0.65 ± 0.01	0.67 ± 0.01	0.72 ± 0.01	0.76 ± 0.02	0.65 ± 0.01	0.82 ± 0.02	0.66 ± 0.01	0.65 ± 0.01	0.67 ± 0.02	0.69 ± 0.00	0.69 ± 0.06
50 full (F)	64 F	0.50 ± 0.01	0.52 ± 0.00	0.52 ± 0.00	0.55 ± 0.01	0.52 ± 0.01	0.62 ± 0.00	0.54 ± 0.01	0.51 ± 0.00	0.54 ± 0.01	0.57 ± 0.00	0.54 ± 0.03
	16 F	0.93 ± 0.02	0.90 ± 0.02	0.93 ± 0.01	0.91 ± 0.02	0.88 ± 0.03	0.98 ± 0.01	0.96 ± 0.01	0.78 ± 0.00	0.82 ± 0.00	0.93 ± 0.02	0.90 ± 0.06
	32 F	0.87 ± 0.01	0.81 ± 0.01	0.85 ± 0.02	0.80 ± 0.01	0.79 ± 0.02	0.91 ± 0.00	0.90 ± 0.01	0.74 ± 0.01	0.70 ± 0.02	0.85 ± 0.02	0.82 ± 0.07
100 full (F)	64 F	0.65 ± 0.04	0.69 ± 0.01	0.71 ± 0.01	0.67 ± 0.01	0.64 ± 0.01	0.76 ± 0.01	0.77 ± 0.01	0.67 ± 0.00	0.61 ± 0.00	0.75 ± 0.06	0.69 ± 0.06
-	16 F	0.98 ± 0.04	0.98 ± 0.01	0.99 ± 0.01	1.00 ± 0.00	0.99 ± 0.00	0.96 ± 0.05	0.93 ± 0.10	0.94 ± 0.09	1.00 ± 0.00	0.99 ± 0.00	0.98 ± 0.05
	32 F	0.92 ± 0.07	0.84 ± 0.20	0.92 ± 0.10	0.98 ± 0.02	0.97 ± 0.02	0.89 ± 0.08	0.82 ± 0.13	0.84 ± 0.11	0.99 ± 0.00	0.99 ± 0.02	0.92 ± 0.10
500 full (F)	64 F	0.80 ± 0.00	0.67 ± 0.21	0.78 ± 0.11	0.90 ± 0.07	0.86 ± 0.08	0.76 ± 0.00	0.67 ± 0.00	0.70 ± 0.00	0.96 ± 0.03	0.99 ± 0.00	0.81 ± 0.13

Table 15: **Reconstruction error on random LoRAs** The error is larger in comparison to reconstructing trained (i.e., non-random) LoRAs in Table 14 for the corresponding compression methods.

Model Type	Method Type					Т	Tasks					Average
model Type		task039	task190	task280	task290	task391	task442	task620	task1342	task1391	task1598	
	base	24.44	1.60	19.13	39.22	10.27	35.46	7.85	6.22	17.82	38.87	20.09
	lora	95.00	86.00	99.00	93.67	94.33	74.88	74.40	26.68	95.00	50.32	78.93
	32 F	97.00	90.00	99.00	93.33	94.67	74.09	72.13	27.83	94.00	50.71	79.28
10 full (F)	64 F	95.00	89.00	99.00	93.67	94.67	74.29	74.80	26.63	96.00	51.04	79.41
	32 F	96.00	88.00	99.00	93.67	92.33	72.30	75.97	29.89	94.00	45.68	78.68
50 full (F)	64 F	98.00	89.00	99.00	93.67	93.33	72.74	76.50	29.33	96.00	45.71	79.33
	32 F	92.10	83.00	99.00	93.67	92.00	71.09	63.29	27.87	88.00	42.36	75.24
100 full (F)	64 F	97.00	87.00	99.00	93.67	92.33	72.23	74.69	29.98	95.00	44.71	78.56
	32 F	68.92	43.00	87.00	91.67	90.67	70.08	51.16	14.40	83.00	41.97	64.19
500 full (F)	64 F	93.50	78.00	91.00	92.33	90.33	72.55	57.49	15.44	85.00	42.31	71.80

Table 16: Performance with convergence In-Distribution Rouge-L

Table 17: Agreement Comparison. 100 LoRAs

Configuration		Agreement (%)
Base Model		83.015
Uncompressed LoRAs		100.000
Joint Compression		
Diagonal	Rank 8	87.032
	Rank 16	88.908
	Rank 32	91.545
	Rank 64	94.659
Full	Rank 8	87.686
	Rank 16	90.163
	Rank 32	94.018
	Rank 64	96.918

Table 18: Performance Comparison. 100 LoRAs

Configuration		Average Performance
Base Model		32.28
Uncompressed LoRAs		48.32
Join Compression		
Diagonal	Rank 8	41.90
· ·	Rank 16	45.44
	Rank 32	46.89
	Rank 64	47.43
Full	Rank 8	43.88
	Rank 16	45.79
	Rank 32	46.83
	Rank 64	47.66

Table 19: Task-Based Performance Evaluation Across Different Models and Ranks

Task	Base Model	LoRA	Diagonal R8	Diagonal R16	Diagonal R32	Diagonal R64
Causal Judgement	57.47	64.37	55.17	58.62	58.62	58.62
Date Understanding	15.33	23.33	20.67	22.00	21.33	22.67
Formal Fallacies	51.33	56.00	52.67	52.67	53.33	54.67
Hyperbaton	6.67	68.00	57.33	63.33	67.33	68.00
Logical Deduction (5 Objects)	21.33	37.33	32.00	36.67	37.33	37.33
Logical Deduction (7 Objects)	12.67	44.00	31.33	42.67	44.67	45.33
Movie Recommendation	62.67	67.33	62.00	64.67	66.67	67.33
Object Counting	34.67	38.00	35.33	36.67	36.67	38.00
Snarks	50.00	61.54	53.85	56.41	58.97	57.69
Temporal Sequences	16.67	23.33	18.67	20.67	24.00	24.67
Average	32.88	48.32	41.90	45.44	46.89	47.43

Table 20: Task-Based Performance Evaluation Across Different Models and Ranks

Task	Base Model	LoRA	Full R8	Full R16	Full R32	Full R64
Causal Judgement	57.47	64.37	56.32	57.47	58.62	60.92
Date Understanding	15.33	23.33	19.33	22.00	22.67	22.67
Formal Fallacies	51.33	56.00	51.33	52.67	53.33	56.00
Hyperbaton	6.67	68.00	63.33	66.00	69.33	68.00
Logical Deduction (5 Objects)	21.33	37.33	35.33	36.00	35.33	37.33
Logical Deduction (7 Objects)	12.67	44.00	40.00	44.67	44.67	44.67
Movie Recommendation	62.67	67.33	63.33	65.33	67.33	67.33
Object Counting	34.67	38.00	35.33	36.67	37.33	37.33
Snarks	50.00	61.54	53.85	55.13	57.69	58.97
Temporal Sequences	16.67	23.33	20.67	22.00	22.00	23.33
Average	32.88	48.32	43.88	45.79	46.83	47.66