NeuKron: Constant-Size Lossy Compression of Sparse Reorderable Matrices and Tensors (Supplementary Document)

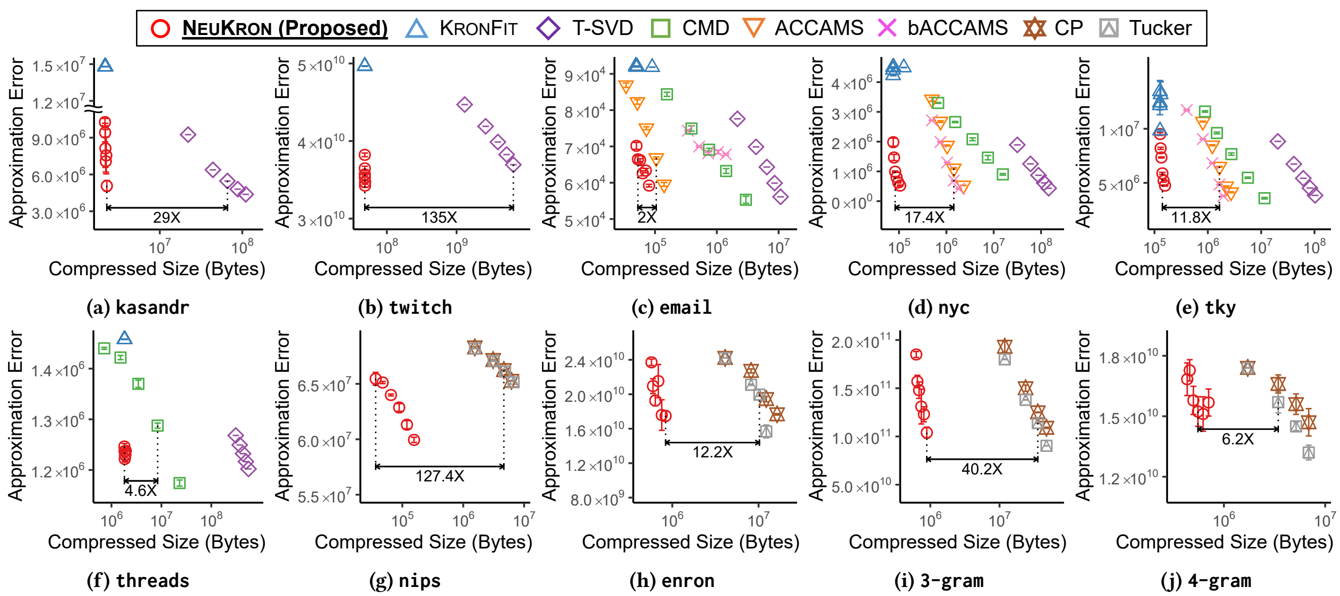

Figure 1: <u>NeuKron significantly outperforms the competitors even in *non-reorderable* matrices and tensors.</u> Even when the permutations of indices for all modes are included, the outputs of NeuKron require up to two orders of magnitude smaller space than those of the competitors with similar approximation error.

Table 1: The effect of γ on approximation error. We report the means and standard errors of approximation errors on the email, nyc, tky, and kasandr datasets.

Dataset	γ	Approximation error
email	1	90561.25 ± 467.996
	10	**58691.88** ± 335.143
	∞	59113.75 ± 891.544
nyc	1	421451.2 ± 4842.068
	10	402673.6 ± 17291.959
	$\boldsymbol{\infty}$	**397947.5** ± 2393.016
tky	1	4166292.3 ± 143013.605
	10	**3981669.6** ± 91907.201
	∞	4034389.1 ± 48117.964
kasandr	1	6315784.36 ± 140974.6535
	10	**4300280.71** ± 488804.599
	∞	4385800.32 ± 496004.629

Table 2: The training time per epoch on all datasets. We report the means and standard errors of training time on each dataset.

Dataset (Hidden Dimension)	Training time
email (30)	0.19 ± 0.010
nyc (30)	0.21 ± 0.004
tky (30)	0.32 ± 0.005
kasandr (60)	1.93 ± 0.005
threads (60)	5.49 ± 0.012
twitch (90)	566.82 ± 3.308
nips (50)	6.31 ± 0.081
enron (90)	80.69 ± 0.266
3-gram (90)	27.19 ± 0.089
4-gram (90)	41.09 ± 0.785

1 EFFECTIVENESS OF NEUKRON ON A NON-REORDERABLE SETTING (RELATED TO SECTION 1)

NeuKron can also be applied to non-reorderable matrices and tensors if the mapping between the original and new orders of mode indices are stored additionally. Even with this additional space requirement, NeuKron still yielded the best trade-off between the approximation error and the compressed size, as seen in Figure 1. Especially, the compression size of NeuKron was two orders of magnitude smaller than those of the competitors with similar approximation error on the twitch and nips datasets.

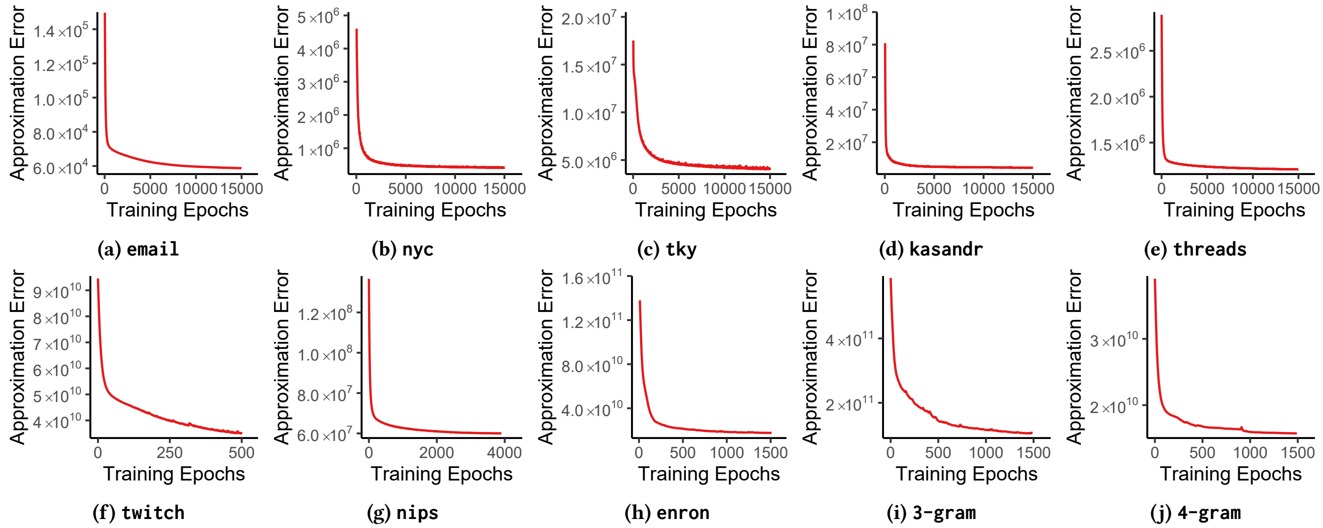

Figure 2: <u>Approximation error of NeuKron after each epoch.</u> In most cases, the approximation error drops rapidly in early iterations, especially within one third of the total epochs that are determined by the termination condition in Section 6.1.

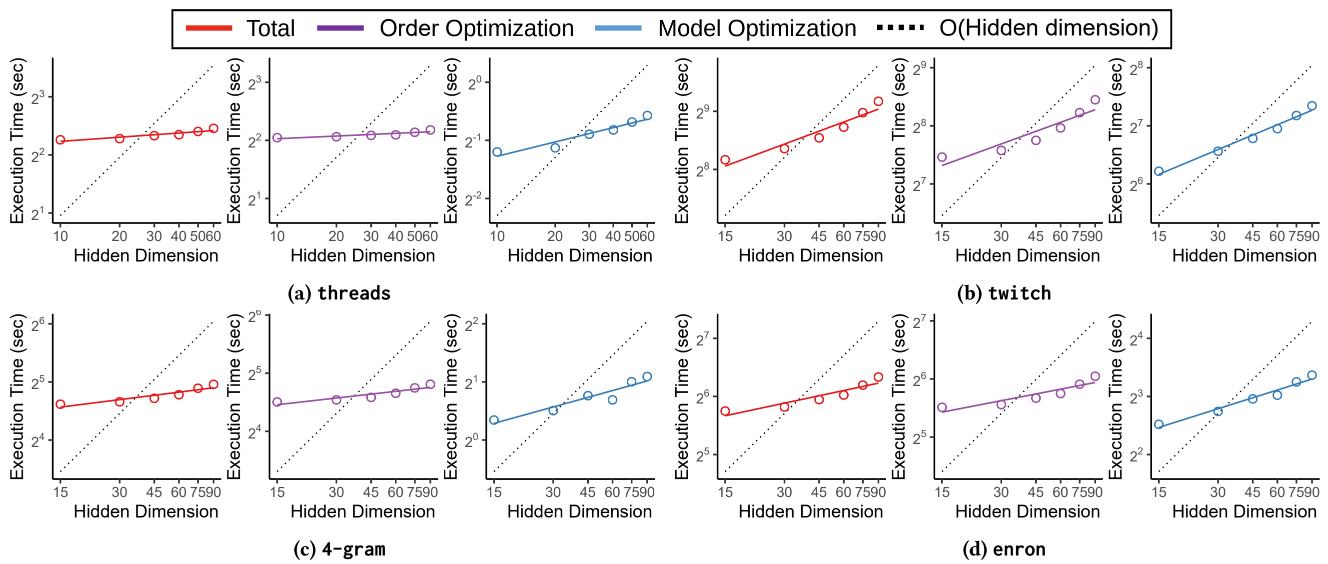

Figure 3: <u>The training time of NeuKron is empirically sub-linear in the hidden dimension h of NeuKron.</u> As in Figure 5 of the main paper, we measure the total elapsed time, the elapsed time for order optimization, and the elapsed time for model optimization.

2 HYPERPARAMETER ANALYSIS (RELATED TO SECTION 6.1)

We investigate how the approximation error of NeuKron varies depending on γ values. We considered three γ values and four datasets (email, nyc, tky, and kasandr) and reported the approximation error in Table 1. Note that setting γ to ∞ results in a hill climbing algorithm that switches rows/column in pairs only if the approximation error decreases. The results show that, empirically, the approximation error was smallest when γ was set to 10 on all datasets except for the nyc dataset.

3 SPEED AND SCALABILITY ON HIDDEN DIMENSION (RELATED TO SECTION 6.4)

We report the average the training time per epoch of NeuKron. The training time per epoch varied from less than 1 second to more than 9 minutes depending on the dataset. As seen in Figure 2, the training plots of all datasets dropped dramatically within one third of total epochs that were determined by the termination condition in Section 6.1. Thus, a model that worked well enough could be obtained before convergence.

We also analyzed the effect of the hidden dimension h on the training time per epoch of NeuKron. As seen in Figure 3, both the elapsed time for order optimization and the elapsed for model optimization were empirically sublinear in the hidden dimension.