A Training details and additional results

A.1 Algorithmic tasks

Datasets were randomly generated by uniformly sampling tokens from dictionary into task sequences and generating targets accordingly to the tasks. After generation, datasets are fixed for all experiments.

Copy and reverse use sequences of sizes 24, 40, 120, 240, and 360, making total copy/reverse input length 48/72, 80/120, 240/360, 480/720, 720/1080. The associative retrieval task consists of 4 key-value pairs and one randomly selected key; the answer consists of one value. Train, validation and test sizes of copy 24, reverse 24 and associative retrieval datasets are 100000, 5000 and 10000.

Transformer-XL had the same cache size on training and validation to match RMT.

For training all models on copy and reverse, we used constant learning rate 1e-4 with reduction on plateau with decay factor of 0.5. Copy and reverse were solved by models with 4 layers and 4 heads, associative retrieval models had 6 layers and 4 heads. Models with the same context size and memory size were trained for the same number of steps and the same training parameters.

Experiments with sequence length 24 were conducted on a single Nvidia GTX 1080 Ti GPU from 1 hour to 2-3 days. Copy and reverse on longer sequence lengths were done on more powerful Tesla V100 using 1-3 devices with training time varying from 1 hour to 3-4 days.

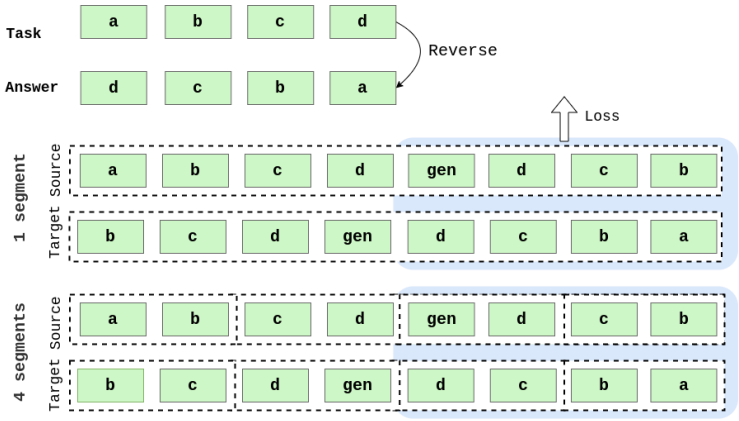

Figure 1: **Reverse task in one and four segments setting for decoder-only models.** Dotted lines show segment borders.

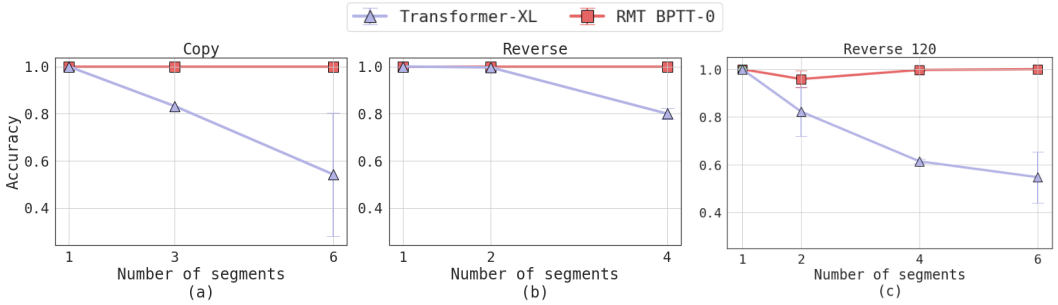

Figure 2: **RMT scales better with limited memory size.** Test set per-character accuracy on copy (a), reverse with a sequence length 24 (b) and 120 (c). Memory size is set to a half of the segment size. RMT solves the tasks almost perfectly with limited memory unlike Tr-XL.

A.2 Associative retrieval

We used code for the task dataset generation from (Ba et al., 2016)[1].

A.3 Quadratic equations

This dataset consists of equations with integer coefficients and step-by-step solutions using the discriminant. Process of equation generation is started from uniformly sampling real roots x_1, x_2 from -100 to 100. The answer of an equation is represented as x_1, x_2. Next, we find the equation as multiplication of two parentheses $(x - x_1)(x - x_2) = 0$, which is expanded to $x^2 - (x_1 + x_2)x + x_1x_2 = 0$. Next, we multiply all coefficients by a random natural number α from 1 to 10. The final equation form is $\alpha x^2 - \alpha(x_1 + x_2)x + \alpha x_1x_2 = 0$. A dataset sample is made of these stages in reversed order. We also provide a string with the discriminant calculation to help find the equation roots. 20 percent of equations in the dataset do not have real roots.

Example equation string:

*-4*x^2+392*x-2208=0*,

solution string:

*x^2-98*x+552=0;D=98^2-4*1*552=7396=86^2;x=(98-86)/2=6;x=(98+86)/2=92* ,

and answer:

6,92

Each solution step is tokenized on char level and padded to the length of 30 tokens. The total length of each training sample is 180, the dataset has 100000 training, 10000 validation and 20000 test samples.

For this task we used models with 6 layers, 6 heads and segment sizes 180 and 30. The training was performed with the same schedule as copy and reverse on a single GTX 1080 ti for 1-2 days. Memory size for RMT and Transformer-XL was chosen equal to the segment length.

A.4 Enwik8

We verified our experimental setup by reproducing Transformer-XL results on enwik8 dataset (Table 1). We used 12-layer Baseline (Transformer), Transformer-XL, RMT in all enwik8 experiments. All results on enwik8 dataset are in Table 1. We used 2 NVIDIA A100 80Gb GPUs, training time varied from 10 to 30 hours depending on sequence length, memory size, and number of BPTT unrolls.

A.5 WikiText-103

We used 16-layer models in all experiments on WikiText-103 dataset. Training hyperparameters were used from (Dai et al., 2019) and authors PyTorch scripts[2]. All results on WikiText-103 dataset are in Table 2. In most of the WikiText-103 experiments, we used 2 NVIDIA A100 80Gb GPUs, training time varied from 10 to 30 hours depending on sequence length, memory size, and number of BPTT unrolls. All models except the ones noted with 2x steps were trained for 200k batches. Transformer-XL did not benefit from longer training unlike the Tr-XL + RMT model. For training the combined model we used an auxiliary loss for memory tokens, it was added to the main loss with a multiplier of 0.01. We set a new fixed special token to be predicted from memory as target in the auxiliary loss.

[1] https://github.com/GokuMohandas/fast-weights/blob/539fb10e3c384d5f782af2560bf28631cd0eaa61/fw/data_utils.py

[2] https://github.com/kimiyoung/transformer-xl

Table 1: Test set bits-per-character on enwik8. Our experimental setup shows similar scores to the original paper (Dai et al., 2019) with segment length 512.

MODEL	MEMORY	SEGMENT LEN	$BPC_{\pm STD}$
TR-XL (DAI ET AL., 2019)	512	512	1.06
TR-XL (OURS)	512	512	1.071
TR-XL	200	128	1.140
TR-XL	100	128	1.178
TR-XL	75	128	1.196
TR-XL	40	128	1.230 ± 0.001
TR-XL	20	128	1.261
TR-XL	10	128	1.283 ± 0.001
RMT BPTT-1	5	128	1.241 ± 0.002
RMT BPTT-2	5	128	1.231 ± 0.002
RMT BPTT-1	10	128	1.240 ± 0.006
RMT BPTT-2	10	128	1.228 ± 0.003
RMT BPTT-0	20	128	1.301
RMT BPTT-1	20	128	1.229
RMT BPTT-2	20	128	1.222

Table 2: Test set perplexity on WikiText-103. All experiments with RMT and Tr-XL models.

MODEL	MEMORY	SEGMENT LEN	$PPL_{\pm STD}$
BASELINE	0	150	29.95 $\pm$ 0.15
MT	10	150	29.63 $\pm$ 0.06
MT	25	150	29.67 $\pm$ 0.03
MT	75	150	29.69 $\pm$ 0.02
MT	150	150	29.82 $\pm$ 0.35
TR-XL (PAPER)	150	150	24.0
TR-XL (OURS)	150	150	24.12 $\pm$ 0.05
TR-XL (OURS) 2X STEPS	150	150	24.67
TR-XL	75	150	24.68 $\pm$ 0.01
TR-XL 2X STEPS	75	150	24.49
TR-XL	25	150	25.57 $\pm$ 0.02
RMT BPTT-0	10	150	26.85 $\pm$ 0.02
RMT BPTT-1	10	150	25.92 $\pm$ 1.07
RMT BPTT-2	10	150	25.32 $\pm$ 0.61
RMT BPTT-3	10	150	25.04 $\pm$ 0.07
RMT BPTT-0	25	150	29.73
RMT BPTT-1	25	150	24.91
RMT BPTT-2	25	150	24.85 $\pm$ 0.31
TR-XL + RMT BPTT-3	70 + 5	150	24,53
TR-XL + RMT BPTT-3	75 + 5	150	24,47 $\pm$ 0.05
TR-XL + RMT BPTT-0	140 + 10	150	24,25
TR-XL + RMT BPTT-1	150 + 10	150	24,30 $\pm$ 0.09
TR-XL + RMT BPTT-3 2X STEPS	150 + 10	150	23,99 $\pm$ 0.09
BASELINE	0	50	39.05 $\pm$ 0.01
TR-XL	200	50	25.14
TR-XL	100	50	25.66 $\pm$ 0.01
TR-XL	50	50	26.54 $\pm$ 0.01
TR-XL	25	50	27.57 $\pm$ 0.09
TR-XL	10	50	28.98 $\pm$ 0.11
TR-XL	5	50	30.06 $\pm$ 0.07
TR-XL	1	50	32.35 $\pm$ 0.03
RMT BPTT-0	1	50	31.33 $\pm$ 1.26
RMT BPTT-1	1	50	28.71 $\pm$ 0.03
RMT BPTT-2	1	50	28.44
RMT BPTT-3	1	50	28.40 $\pm$ 0.03
RMT BPTT-0	5	50	30.32 $\pm$ 0.18
RMT BPTT-1	5	50	27.05 $\pm$ 0.20
RMT BPTT-2	5	50	26.83 $\pm$ 0.18
RMT BPTT-3	5	50	26.75 $\pm$ 0.26
RMT BPTT-4	5	50	26.67 $\pm$ 0.03
RMT BPTT-0	10	50	30.69 $\pm$ 0.01
RMT BPTT-1	10	50	27.95 $\pm$ 1.32
RMT BPTT-2	10	50	26.62 $\pm$ 0.34
RMT BPTT-3	10	50	26.37 $\pm$ 0.01
RMT BPTT-4	10	50	26.25 $\pm$ 0.19
RMT BPTT-0	25	50	29.75
RMT BPTT-1	25	50	26.32
RMT BPTT-2	25	50	27.31
RMT BPTT-0	50	50	29.75
RMT BPTT-1	50	50	26.03

B Operations with Memory

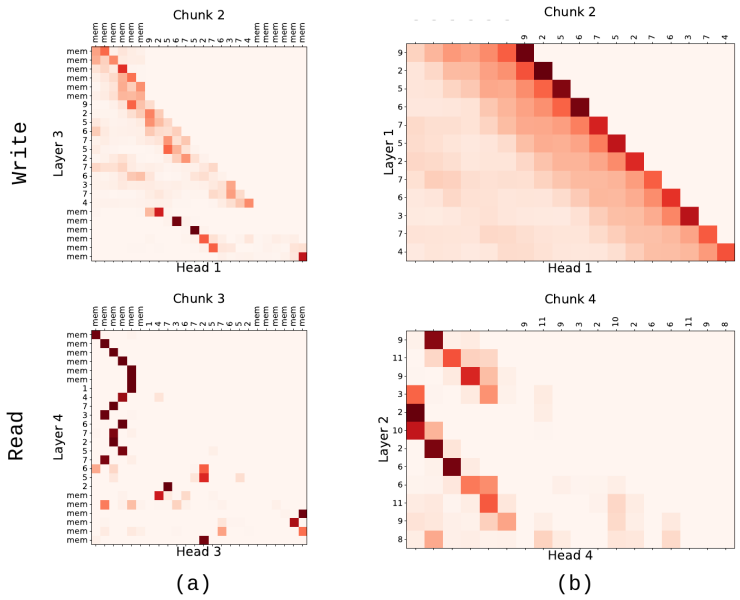

Figure 3: Approaches to compression and decompression of sequence with length 12 and memory with size 6. (a) - RMT, (b) - Transformer-XL. Tr-XL mixes representations of tokens and reads multiple symbols from one cached state. RMT manages to compress the whole segment into memory tokens.

References

Jimmy Ba, Geoffrey E Hinton, Volodymyr Mnih, Joel Z Leibo, and Catalin Ionescu. Using fast weights to attend to the recent past. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL `https://proceedings.neurips.cc/paper/2016/file/9f44e956e3a2b7b5598c625fcc802c36-Paper.pdf`.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context, 2019.