Effective Architectures for Long Contexts and Reasoning

Matvey Kairov
kairov.mv@phystech.edu

MIPT Center for Cognitive Modeling

2024 – Scientific Research Seminar

Table of Contents

- Introduction
- RMT Architecture Overview
- Task Dataset
- Motivation
- Problem Formulation
- Proposed Methodology
  - Architectural Modifications
  - Heuristic Modifications
- Initial Experiments & Results
- Conclusion & Future Work

Introduction

- Transformers [1] are the most widely adopted type of architecture for solving tasks on sequences.
- The attention mechanism, integral to Transformers, has quadratic computation complexity and memory consumption.
- On very long sequences (e.g. $> 10^4$ tokens long) traditional Transformers rapidly degrade in performance.
- Architectures such as the Recurrent Memory Transformer (RMT) [2] were proposed with the goal of reducing complexity of processing long input sequences.

[1] Vaswani et al. "Attention is all you need". 2017

[2] Bulatov et al. "Recurrent Memory Transformer". 2022.

RMT Architecture Overview

- RMT extends the Transformer model by incorporating recurrent memory to process long sequences efficiently.
- For segment τ, given an initial hidden state H_τ^0, memory tokens H_τ^{mem} are appended at the beginning and the end of the hidden state:

$$\tilde{H} := [H_\tau^{mem} \circ H_\tau^0 \circ H_\tau^{mem}], \tag{1}$$

$$\bar{H}_\tau^N := \text{Transformer}(\tilde{H}_\tau^0), \tag{2}$$

$$\bar{H}_\tau^N =: [H_\tau^{read} \circ H_\tau^0 \circ H_\tau^{write}]. \tag{3}$$

H_0^{write} is used as the memory state on the next segment.

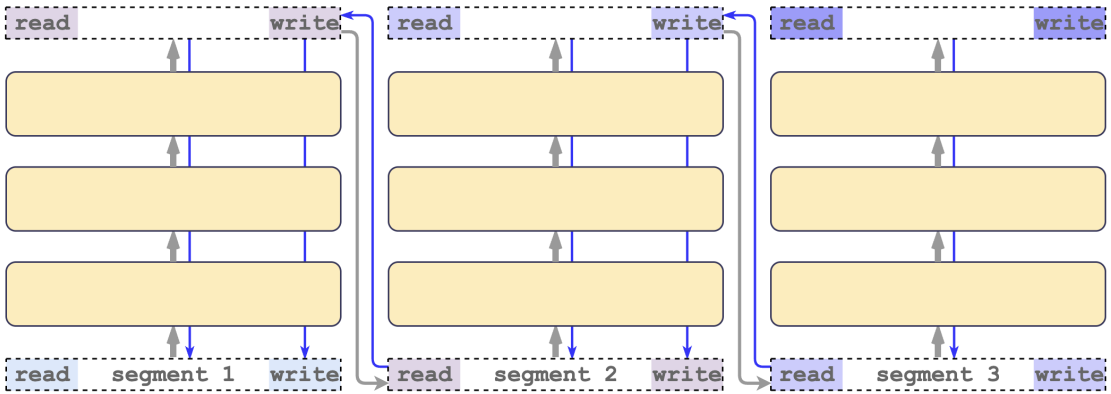


Figure: RMT architecture.

Task Dataset

- Standard text datasets do not account for the needs of long-context memory retrieval.
- We chose **BABILong** [3] as our long-context benchmark.
- It requires the model to retrieve relevant facts from a vast pool of information.
- A BABILong task consists of a large noise text with inserted facts.
- Based on those facts, the model is expected to answer a question, which is appended at the end of the sequence.

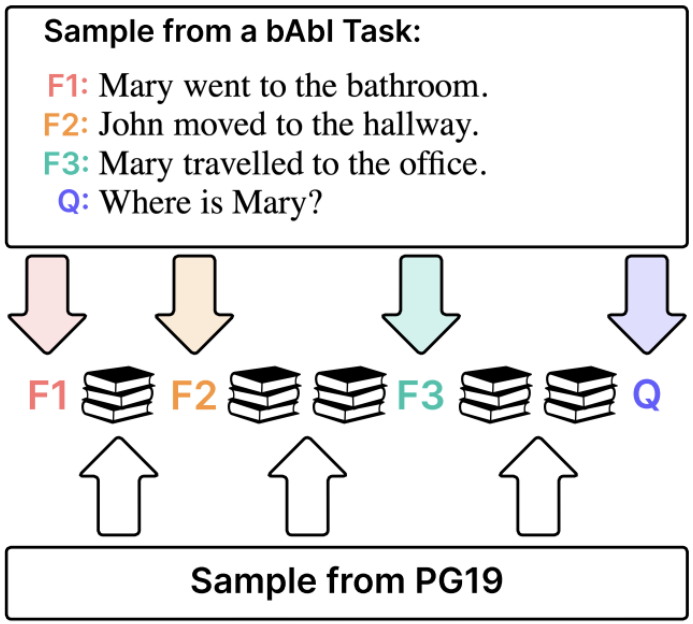


Figure: BABILong task example.

[3] Kuratov et al. "BABILong: Testing the Limits of LLMs with Long Context Reasoning-in-a-Haystack". 2024.

Motivation

- RMT and its variants such as ARMT [4] and RMT-R [5] achieve outstanding performance on long contexts.
- Authors employ a convoluted training curriculum which involves training on sequences of varying length in a strict order.
- This leads to a relatively long training time and difficulties when integrating these models with popular deep learning APIs (e.g. HF Transformers Trainer [6]).

The goal of the current research is to eliminate the need for a complicated training curriculum while maintaining high performance on language modeling tasks.

---

[4] Rodkin et al. "Associative Recurrent Memory Transformer". 2024.

[5] Kuratov et al. "In Search of Needles in a 11M Haystack: Recurrent Memory Finds What LLMs Miss". 2024.

[6] `https://huggingface.co/docs/transformers/main_classes/trainer`

Problem Formulation

The optimization process considers two main objectives:

1. Model Performance — maximizing the likelihood of observed data.
2. Training Efficiency — reducing computational overhead during optimization.

Given a set of parameter configurations Θ and a set of training heuristics H, the general objective function can be formulated as:

$$\theta^*, h^* = \arg \min_{\theta \in \Theta, h \in H} [\mathcal{L} + \lambda \mathcal{C}] \tag{4}$$

Problem Formulation

Tasks in BABILong involve classification, so the first term accounts for minimizing binary cross-entropy over predicted class probabilities. BABILong tasks consist of a noise text x_{noise}, facts x_{facts}, questions x_{quest} and answers y. Given the number of sequences in the dataset M, the number of classes C and predictions y_i', the objective takes the form:

$$\mathcal{L} = -\sum_{i=1}^{M}\sum_{c=1}^{C} y_i^{(c)} \log p_{\theta,h}\left(y_i' = c|x_i\right). \tag{5}$$

The second term regulates computational complexity of training. We focus on avoiding inefficient training heuristics (given a set of heuristics costs c_k), penalizing high computational costs, reducing the total number of parameters, and encouraging faster convergence.

$$\mathcal{C} = \sum_k c_k \cdot \mathbf{1}_{h_k} + \beta\|\theta\|_0 + \text{FLOPs}\left(\theta\right) + \gamma T_{\text{epochs}}. \tag{6}$$

Proposed Methodology

The proposed methodology involves modifying both architecture and training heuristics to enable efficient single-pass training.

- **Architectural Modifications:**
  - Introduce structural adjustments to improve memory efficiency.
  - Optimize parameterization to maintain inference-time constraints.
- **Training Heuristics:**
  - Explore alternative scheduling strategies to replace progressive segment training.
  - Introduce auxiliary training mechanisms that accelerate convergence without affecting inference cost.

Proposed Methodology

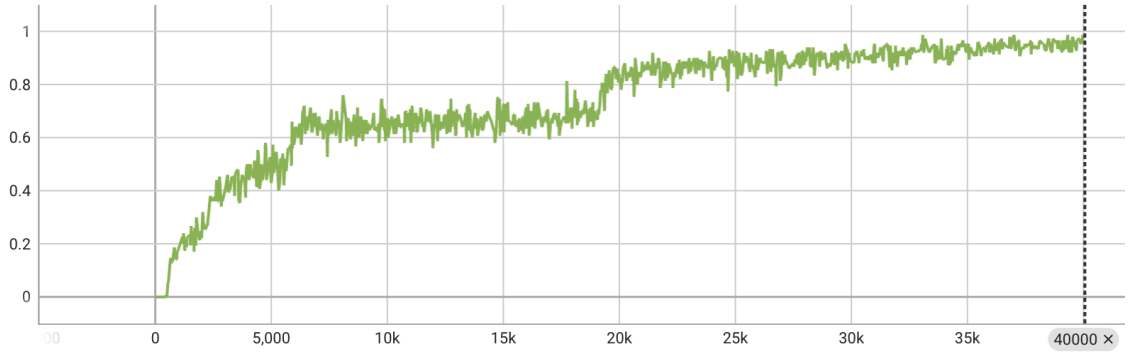
Center for Cognitive Modeling

We proposed the following architectural and heuristic modifications to the original RMT model:

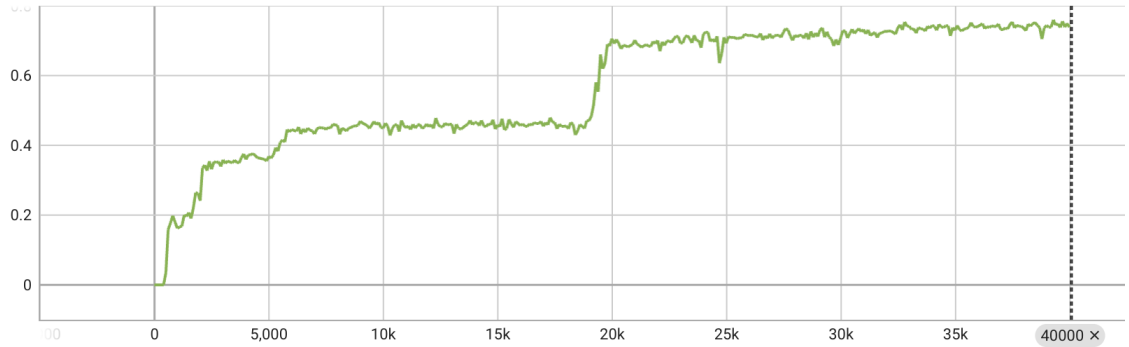
- **Single-Pass RMT Training**: we train RMT in a single pass, without curriculum learning.
- **RMT-R**: we introduce memory retrieval mechanisms for improved handling of long-term dependencies.
- **RMT-B**: we implement memory updates on each transformer layer.
- **RMT-BR**: we combine RMT-B with the memory retrieval mechanism.
- **Sep-Q**: we pass the <|gen|> token in a separate segment, further encouraging the optimization of the memory mechanism.

Experiments: Single-Pass RMT

- We train RMT on the QA1 task from BABILong from scratch.
- Results:
  - model overfits;
  - very long training time: >2 days for 6 segments $\times$ 512 tokens.



(a) Training accuracy.



(b) Validation accuracy.

Experiments: RMT-R

- RMT-R preserves tokens from previous segments and attends to them in each segment.
- After each segment, memory tokens are combined with past tokens using a transformer-like block.
- We use Rotary Positional Embeddings (RoPE) [7] to retain memory order during the Attention operation.

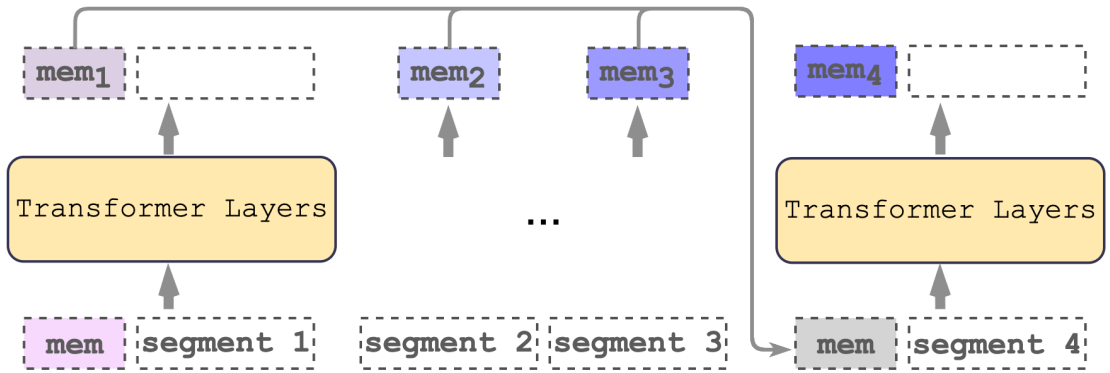
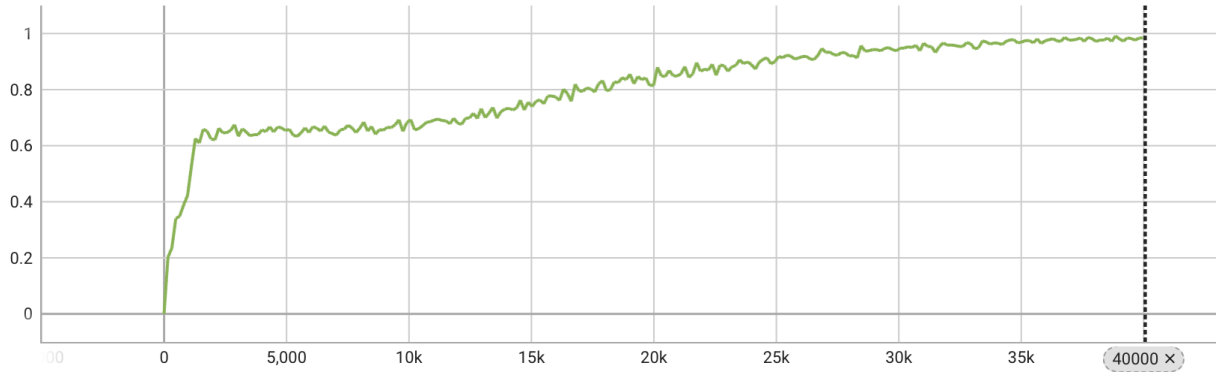


Figure: RMT-R architecture.

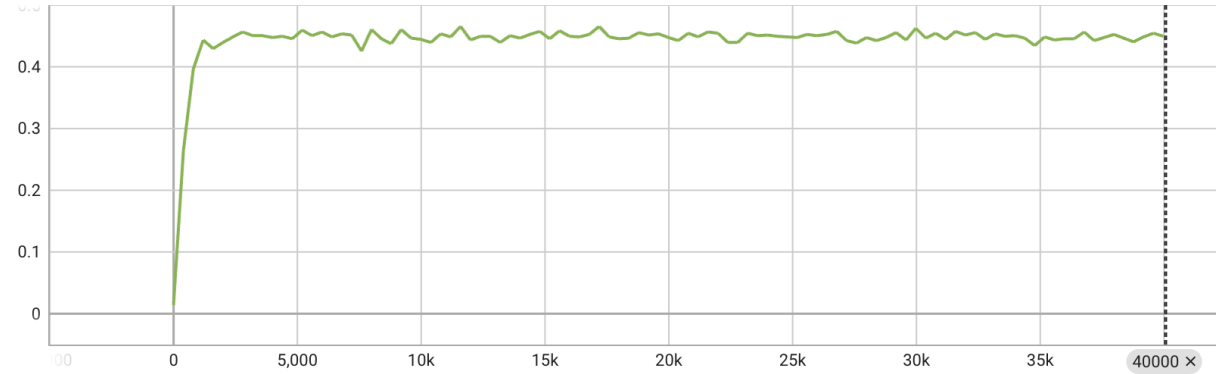
[7] Su et al. "RoFormer: Enhanced Transformer with Rotary Position Embedding". 2021.

Experiments: RMT-R

Results: model overfits during training.



(a) Training accuracy.



(b) Validation accuracy.

Figure: 16 memory tokens, 6 segments $\times$ 512 tokens.

Experiments: RMT-B & RMT-BR

- RMT-B applies the idea behind RMT on each individual Transformer layer.
- Each layer has its own set of memory tokens and performs memory updates.
- RMT-BR inherits the memory aggregation module from RMT-R with the same working principle, but on a lower level.

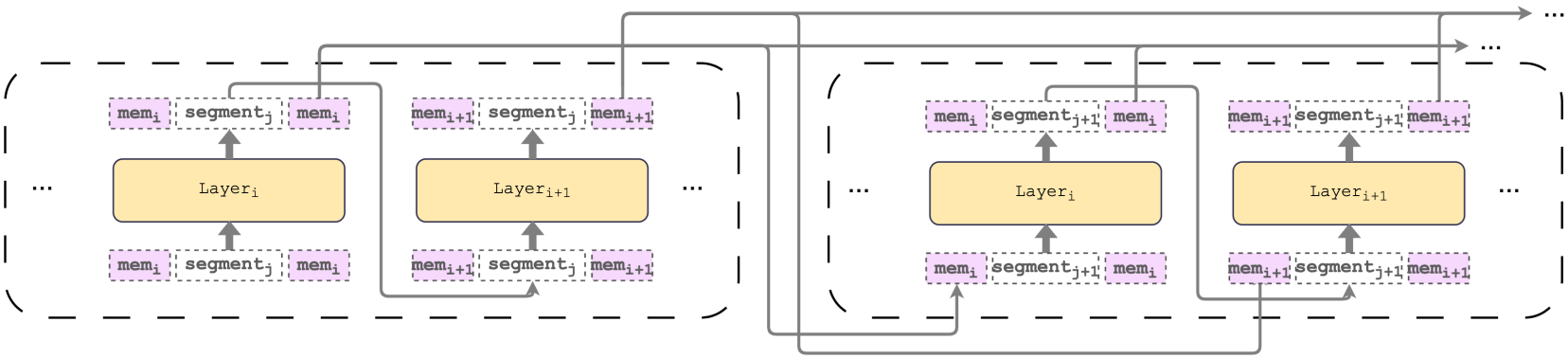
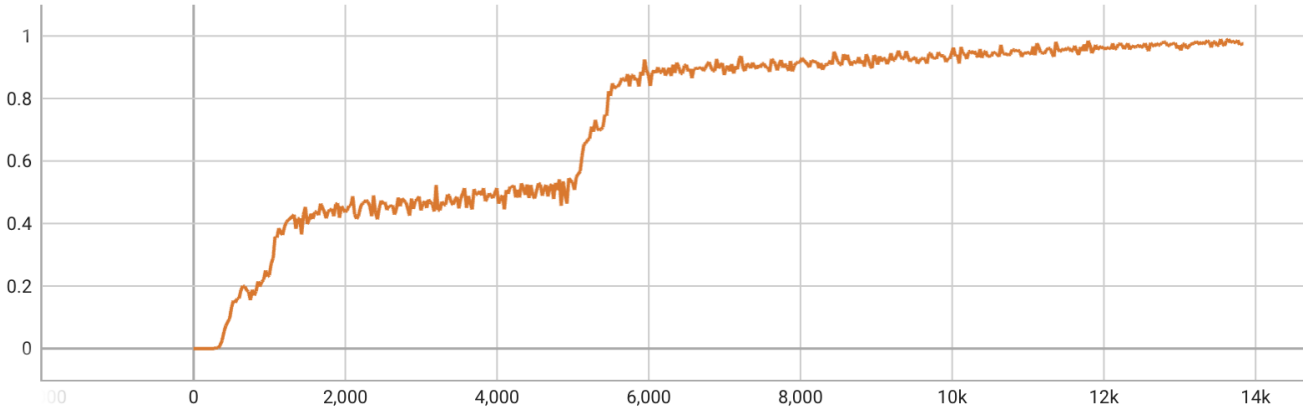


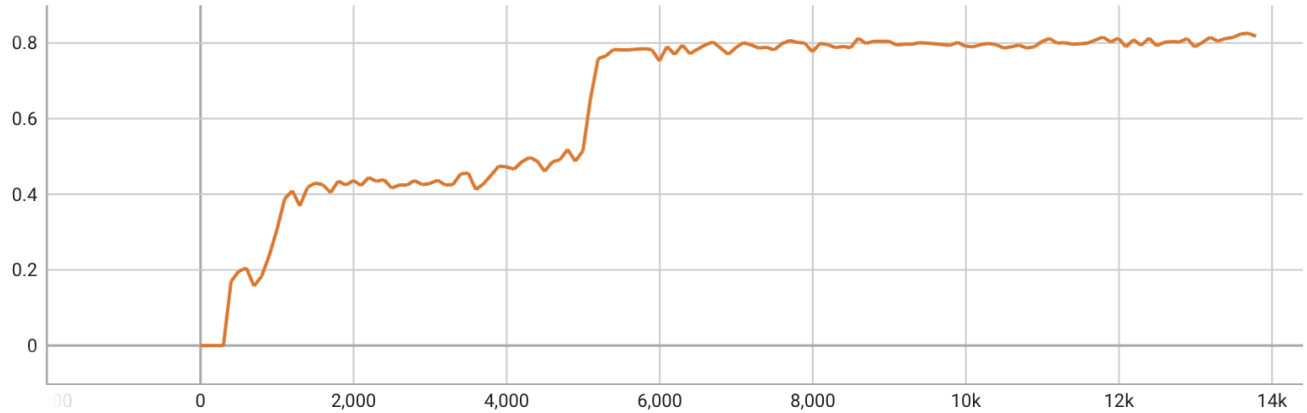
Figure: RMT-BR architecture.

Experiments: RMT-BR

Result: model overfits.



(a) Training accuracy.



(b) Validation accuracy.

Figure: 16 memory tokens, 3 segments $\times$ 512 tokens per segment.

Sep-Q

- Heuristic approach: add the question, <|gen|> and answer tokens in a separate segment after the main text body.
- The main idea behind the approach is to motivate memory retention by eliminating the chance of the question and the answer being in the same segment.

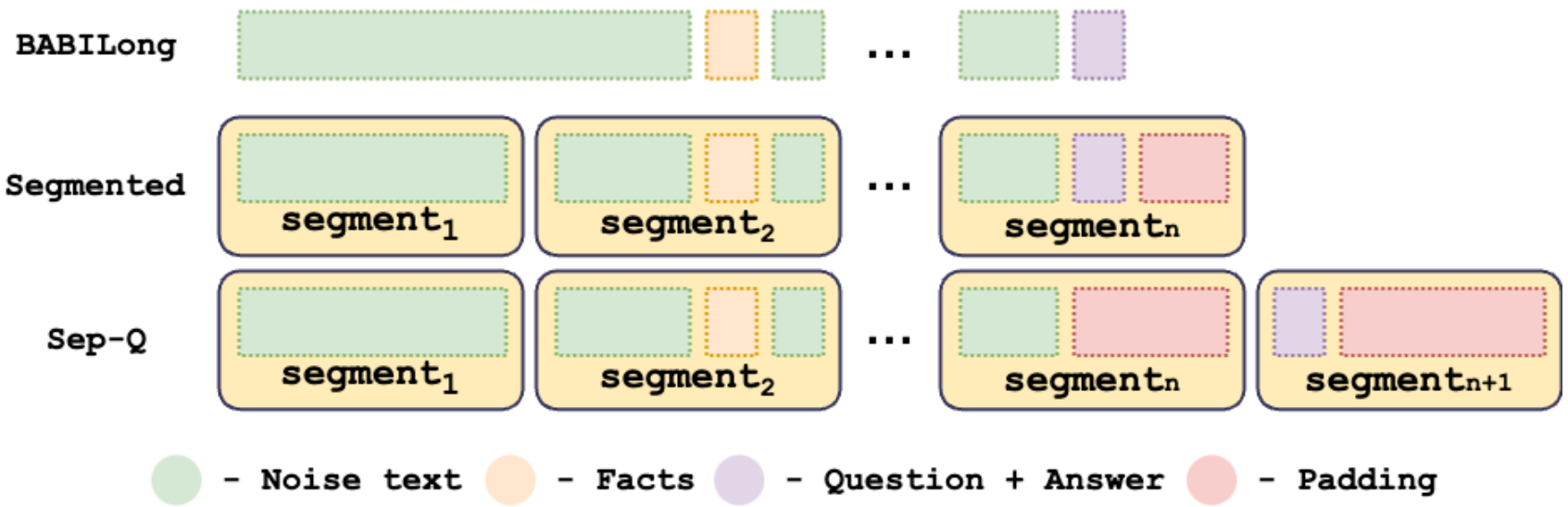
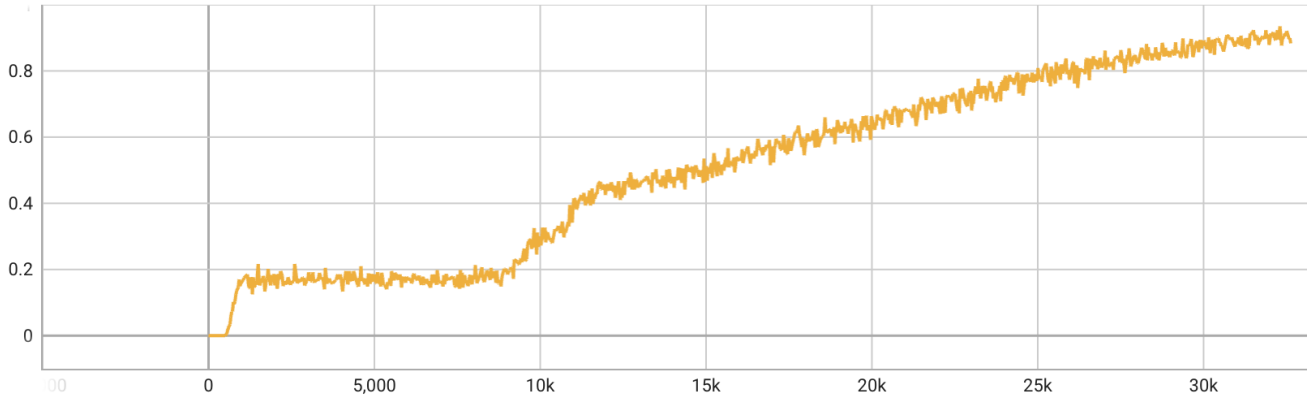


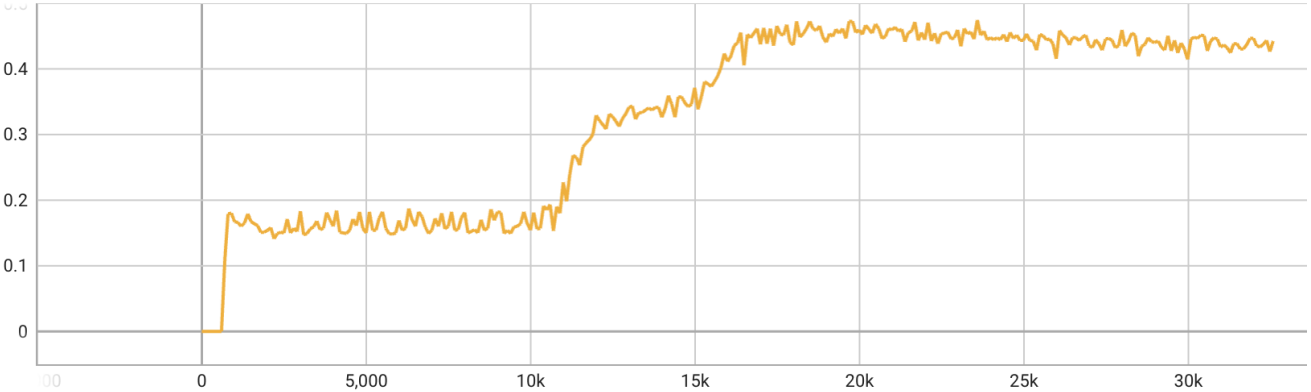
Figure: BABILong Sep-Q variant.

Sep-Q: experiment results

Result: model overfits during training.



(a) Training accuracy.



(b) Validation accuracy.

Figure: RMT-BR, 4 memory tokens, 3 segments $\times$ 128 tokens, no retrieval.

Conclusion

Center for Cognitive Modeling

- We attempted training **RMT** without a curriculum.
- We modified **RMT-R** and proposed **RMT-B** and **RMT-BR**, with the main goal of shortening gradient paths for memory tokens.
- We proposed **Sep-Q**, a training heuristic for the **BABILong** dataset which is designed to motivate memory retention.
- Experiments show that the models tend to overfit before converging.
- Future plans:
  - Further tuning of proposed architectures.
  - Using pre-trained GPT layers for aggregation.
  - Experiments with architectures such as ARMT.
  - New heuristic approaches, such as a loss for memory retention efficiency.

Thank you!

kairov.mv@phystech.edu