Understanding and Improving Length Generalization in HSA Models

Jiaqi Leng[1], Xiang Hu[3], Junxiong Wang, Jianguo Li[3], Wei Wu[3], Yucheng Lu[2]

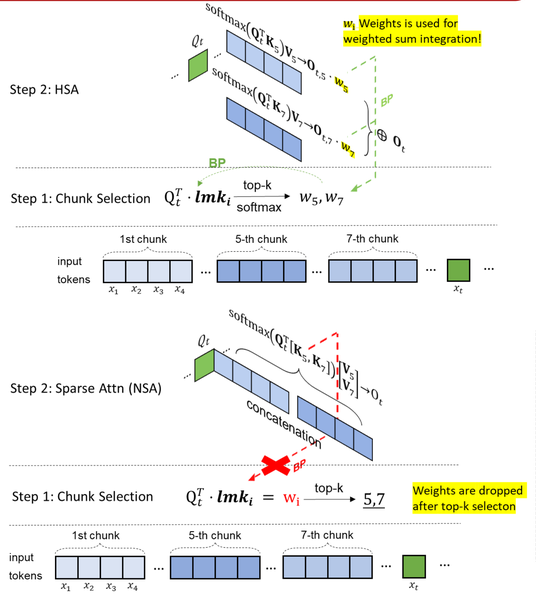
[1] Fudan University [2] NYU Shanghai [3] Ant Group

Contact Author: *Jiaqi Leng*
Email: lengjiaqi18@gmail.com

TL;DR

- **Problem:** Long-context models need **length extrapolation, efficiency** and **random access to relevant information**.
- **HSA:** HSA is a sparse attention mechanism that **selects top relevant chunks** and **performs weighted-sum integration** of the selected chunks for long-range context access.
- **Focus of this paper:** *What architectural choices make HSA generalize to extreme lengths*?
- **Answer:** Strong extrapolation requires **better retrieval representations, stable integration** and **train-test alignment.**
- **Headline result:** Train on **4K**, generalize to **8M** on BabiLong and **32M** on RULER.

Preliminary: HSA



Methodology

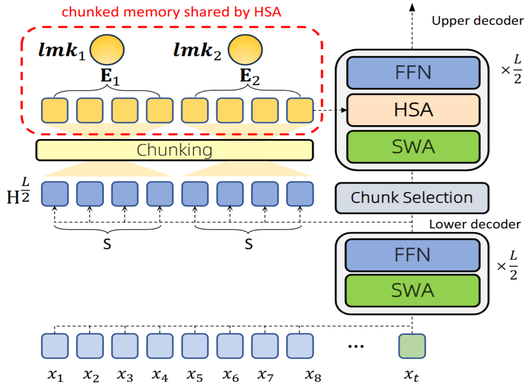


Fig.1: Overall model architecture

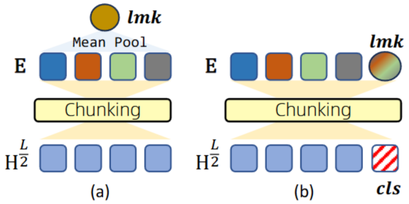


Fig.2: Encoder + CLS improves retrieval

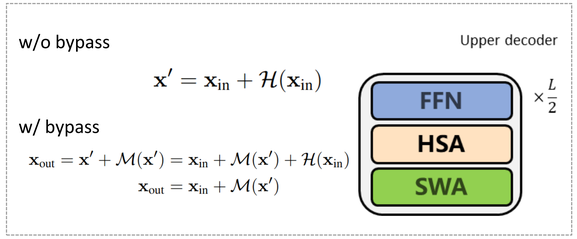


Fig.3: Bypass improves integration.

Methodology Roadmap

Overall model architecture (Fig.1)

- Chunk tokens into coarse memory units for retrieval.
- Use HSA for sparse global retrieval and SWA for local modeling.

Retrieval representations (Fig.2, Tbl.1)

- Chunk Encoder + CLS improves chunk selection.

Better information integration (Fig.3)

- Bypass design stabilizes the use of retrieved global information.

Better train–test alignment (Fig.4)

- Sparse Top-K training matches sparse retrieval at inference time.

Table 1: Unified view of chunk processing.

Configuration	$f(\mathbf{H}_{[i]})$	$g(\mathbf{H}_{[i]})$
NSA	$\text{MeanPool}(\mathbf{H}_{[i]})$	$\mathbf{H}_{[i]}$
HSA w/o Encoder	$\text{MeanPool}(\text{Norm}(\mathbf{H}_{[i]}))$	$\text{RMSNorm}(\mathbf{H}_{[i]})$
HSA w/ Encoder w/o CLS	$\text{MeanPool}(\text{Encoder}(\mathbf{H}_{[i]}))$	$\text{Encoder}(\mathbf{H}_{[i]})$
HSA w/ Encoder w/ CLS	$\text{Encoder}([\mathbf{x}_{\text{CLS}}; \mathbf{H}_{[i]}])[0]$	$\text{Encoder}([\mathbf{x}_{\text{CLS}}; \mathbf{H}_{[i]}])[1:]$

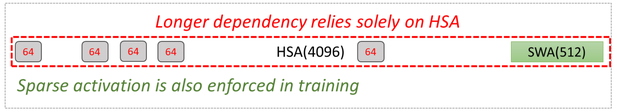


Fig.4: Sparse training aligns train and test.

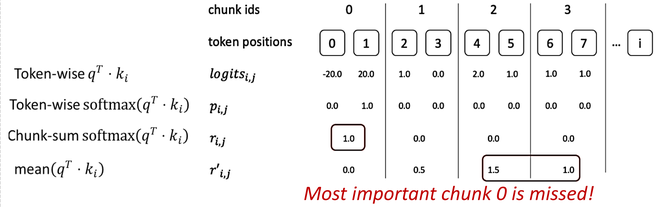


Fig.5: Why chunk selection needs nonlinearity.

Results

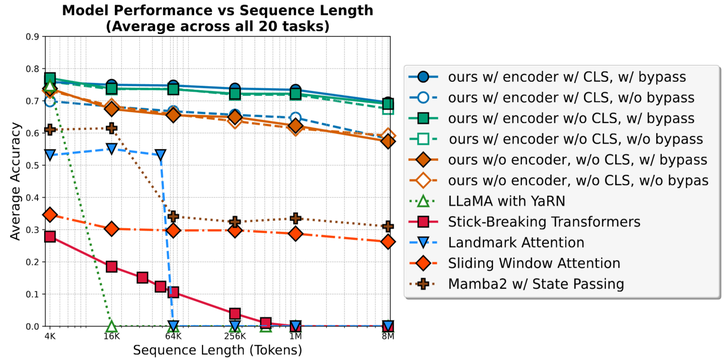


Fig.6: BabiLong evaluation results. Our SWA+HSA model remains robust up to 8M tokens, while common baselines degrade rapidly beyond training length.

Model	4K	32K	128K	32M
Mamba2	65.43	1.12	0.00	—
Llama-YaRN	85.00	0.00	0.00	—
Stick-Break	45.36	43.28	27.27	—
Landmark Attention	53.33	39.33	0.00	—
Ours (SWA+HSA)	**87.92**	**84.86**	**85.65**	**79.77**

Table 2: RULER average accuracy (S-N, MQ-N, and VT). Our SWA+HSA model remains strong from 4K to 32M, while common baselines collapse shortly beyond training length.

Why does HSA generalize?

1. **Better retrieval representations:** Chunk Encoder + CLS improves chunk selection quality.
2. **More stable information integration:** Bypassing Residual Path helps retrieved global information be used effectively.
3. **Better train–test alignment:** Sparse Top-K training prepares the model for retrieval under extreme context lengths.