

PyTorch TorchInductor for SGLang Inference on RISC-V CPUs

CONFERENCE

—NORTH AMERICA 2026—

Jui-Ting Chen¹, Jenq-Kuen Lee¹, Chi-Bang Kuan²

¹Department of Computer Science, National Tsing Hua University, Taiwan

²Mediatek



Introduction

- ATen is a PyTorch's C++ library, providing the core tensor abstractions and mathematical primitives.
- Pytorch experimental ATen RVV (RISC-V Vector Extension) PR (#175746) supplies vector primitives.
- However, SGLang inference remained slow in our experiments using ATen RVV with the TorchNative attention backend.

Llama-3.2-1B · Batch=1/input=64/outout=8 Three-run medians

Llama-3.2-1B	Prefill(token/s)	Decode(token/s)
Before ATen RVV	0.325	0.165
ATen RVV #175746	0.546	0.332

- We extend TorchInductor to generate BF16 RVV GEMV/GEMM kernels for selected Linear operations, targeting the model's compute-intensive matrix computations.

Capability gap in PyTorch CPU compilation

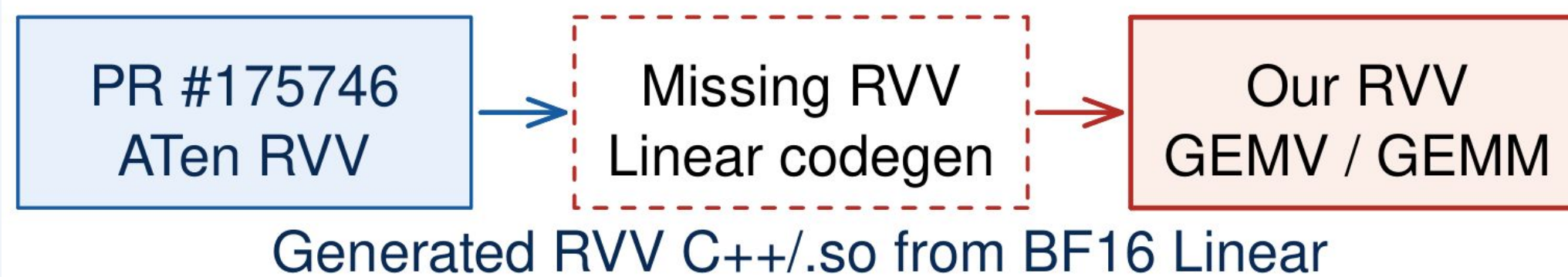


Fig. 1. Our addition: Inductor-generated BF16 Linear.

Contribution

1	VecRVV target	RISC-V ISA selection + JIT flags
2	BF16 kernels	M=1 GEMV; M≥2 GEMM RVV template
3	Regional routing	Supported Linear shapes
4	Packed weight	Explicit weight ownership
5	Serving buckets	M=2–15 bounded routing
6	Hybrid integration	Inductor Linear + SGLang RVV attention

Compile Flow

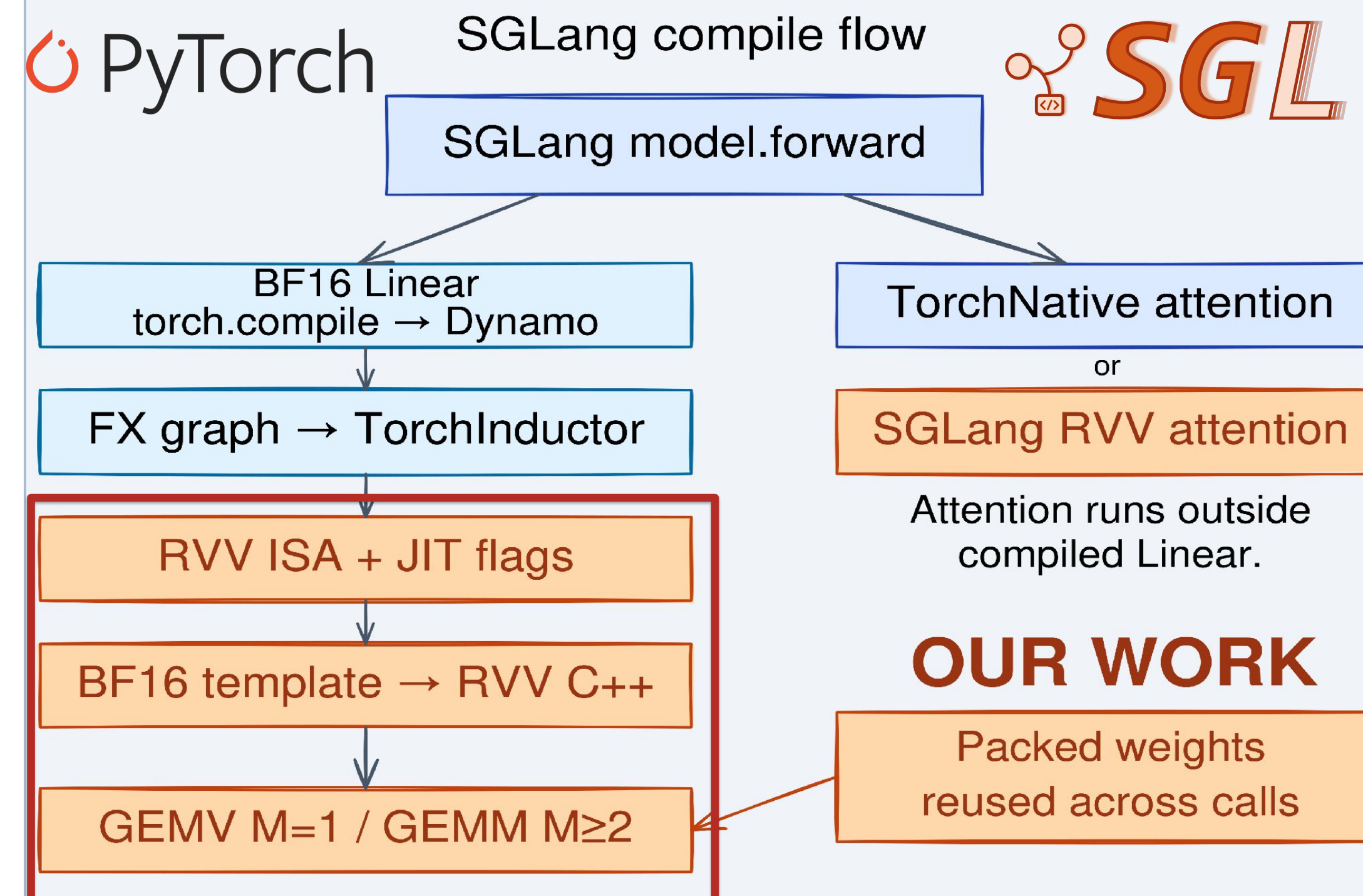
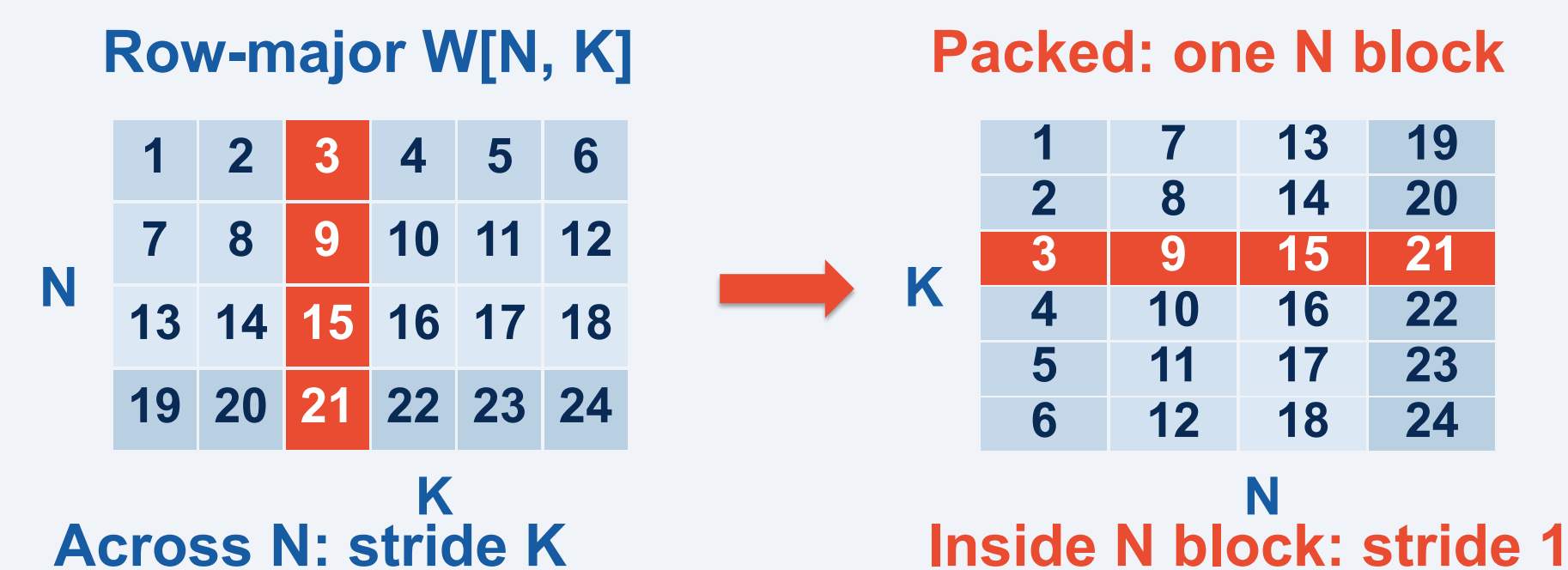


Fig. 2. SGLang compile flow and attention choice.

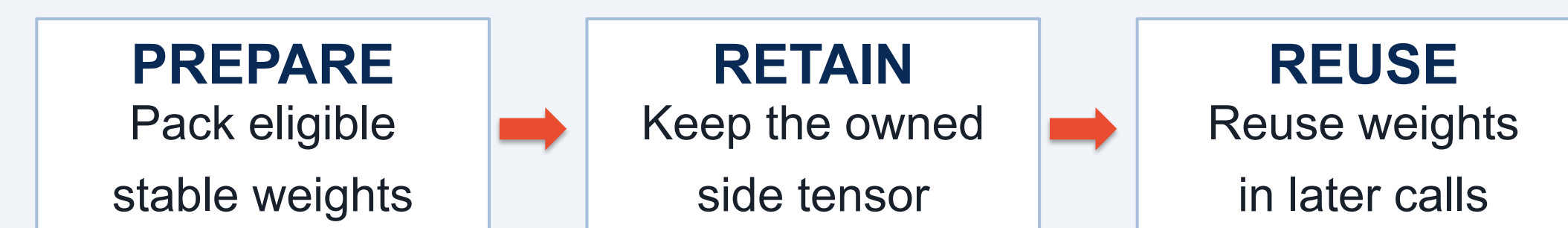
Optimization

Packed weight

Example: M=1 decode. For one input channel k, load weights for several output channels together.



Retain the layout across repeated Linear calls



Contiguous vector reads; layout reused across calls.

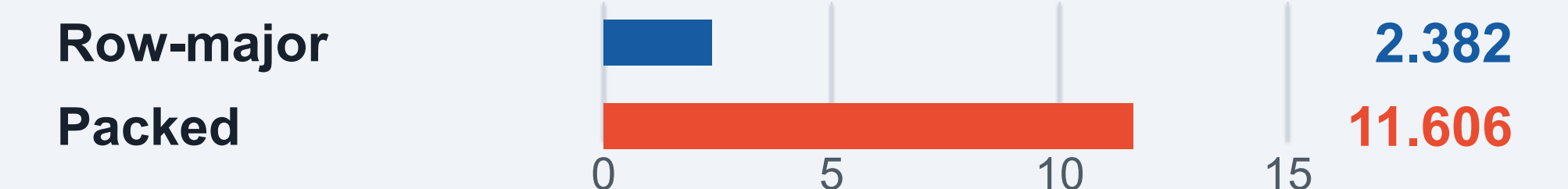
Experiment

Experiment setup

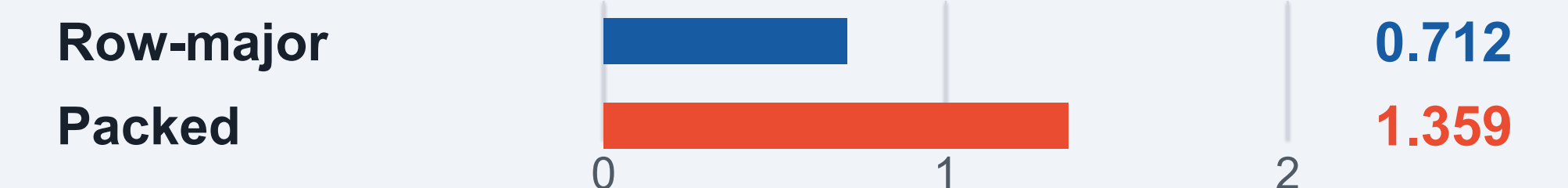
CPU	Banana Pi BPI-F3 · RVV 1.0
Model	Llama-3.2-1B · BF16
Workload	batch 1 · input 64 · output 8
Execution	8 threads · TorchNative attention

Same Inductor RVV Linear + TorchNative attention

Prefill throughput (token/s) **4.848×**



Decode throughput (token/s) **1.913×**

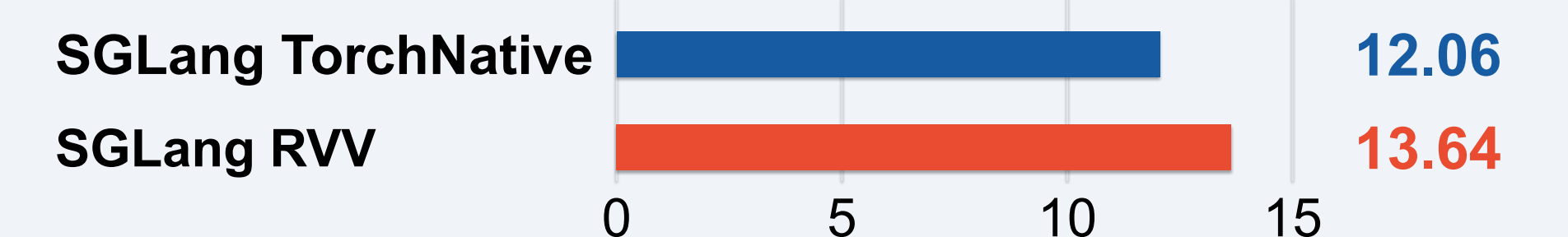


Ten-run medians. Peak RSS +0.926 GiB.

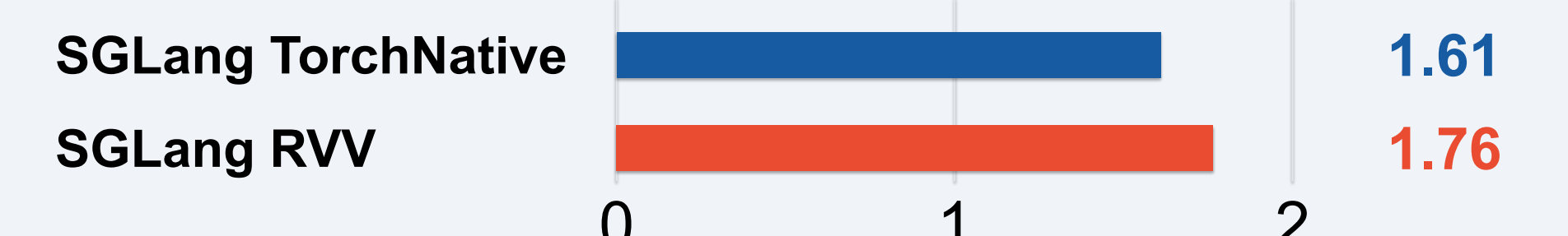
Exploratory Hybrid attention

Both modes use packed Inductor RVV Linear.
Only the SGLang attention backend changes.

Prefill throughput (tok/s)



Decode throughput (tok/s)



Separate pinned stack · Three-run medians

Acknowledgement

This work is supported in part by
Taiwan NSTC, ITRI, Andes and Mediatek.

