SAFARI
SAFARI Research Group
safari.ethz.ch

RUBICON: A Framework for Designing Efficient Deep Learning-Based Genomic Basecallers

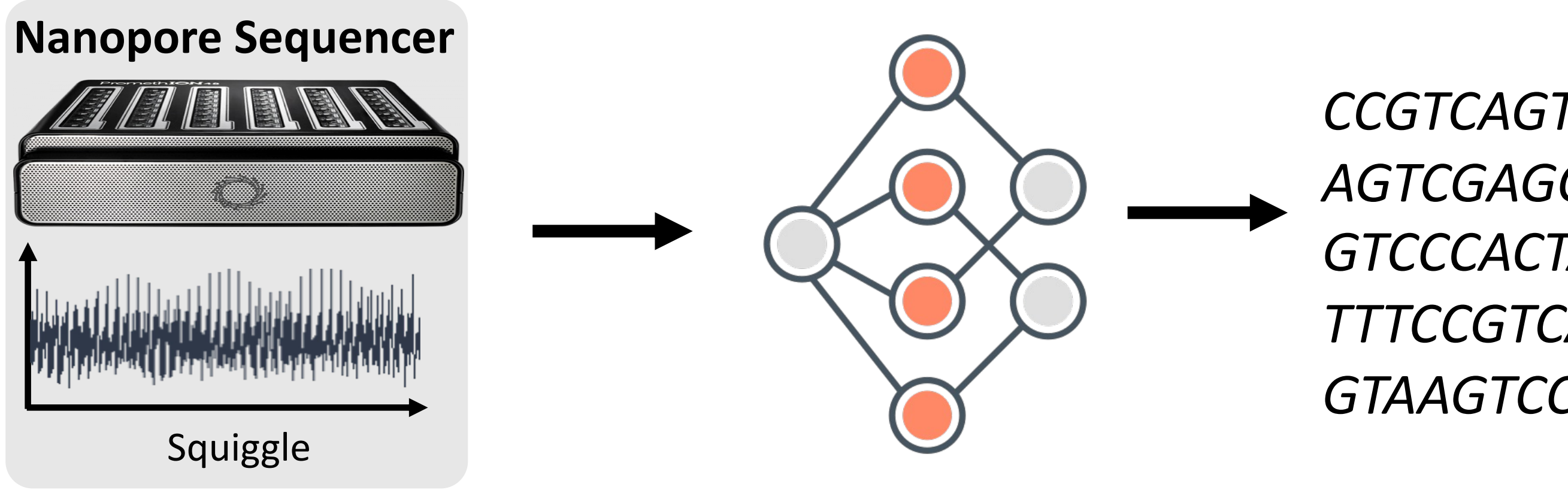
Gagandeep Singh[1,2] Mohammed Alser[1] Alireza Khodamoradi[2] Kristof Denolf[2]
Can Firtina[1] Meryem Banu Cavlak[1] Henk Corporaal[3] Onur Mutlu[1]
[1] ETH zürich [2] AMD [3] TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Full Paper
https://arxiv.org/abs/2211.03079

1: Background: Genomic Basecalling

Basecalling is the first step in the genomics pipeline that converts noisy electrical signals to nucleotide bases (i.e., A, C, G, T)

Modern basecallers use complex deep learning-based models

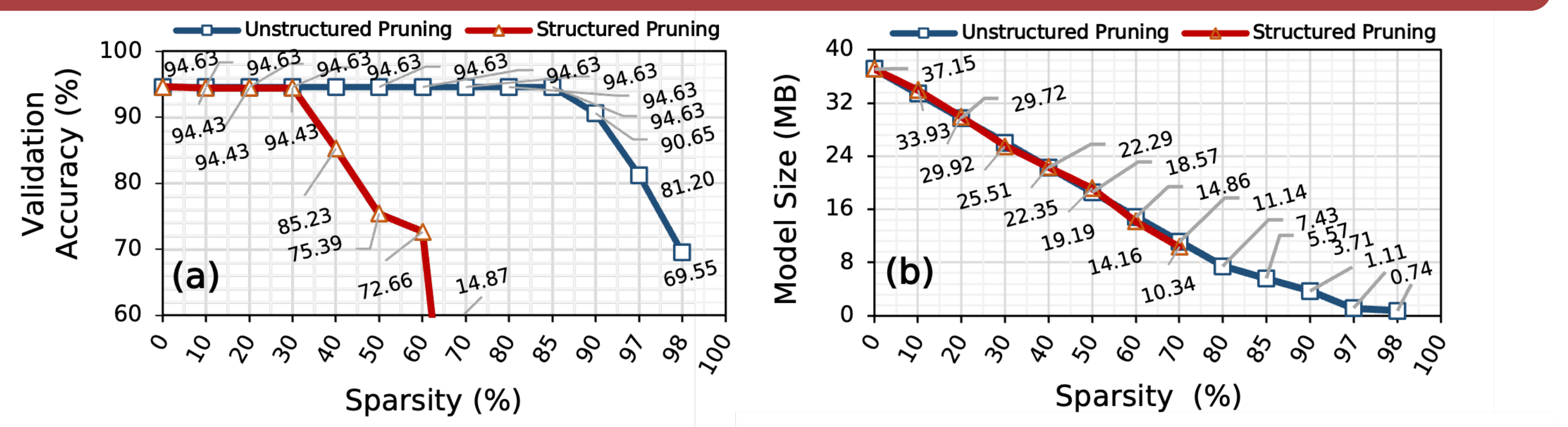


The accuracy and speed of basecalling have critical implications for all the steps in genome analysis

2: Motivation: Analyzing a State-of-the-Art Basecaller

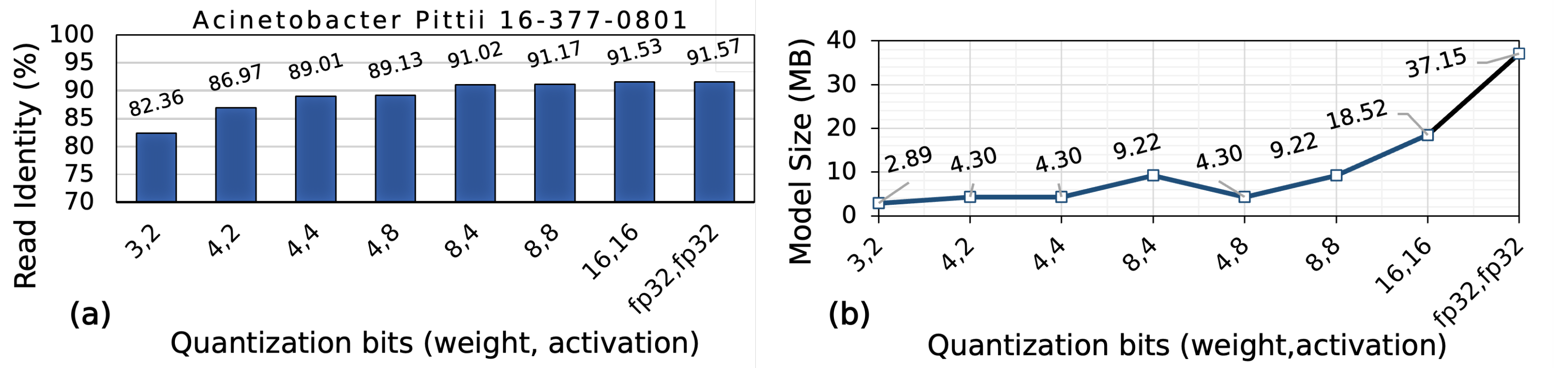
KEY OBSERVATION 1: Effect of Pruning

Basecallers are often adapted from the speech recognition domain leading to over-parametrized models.
85% of weights can be pruned using unstructured pruning leading to 6.67x lower model size without any loss in accuracy.



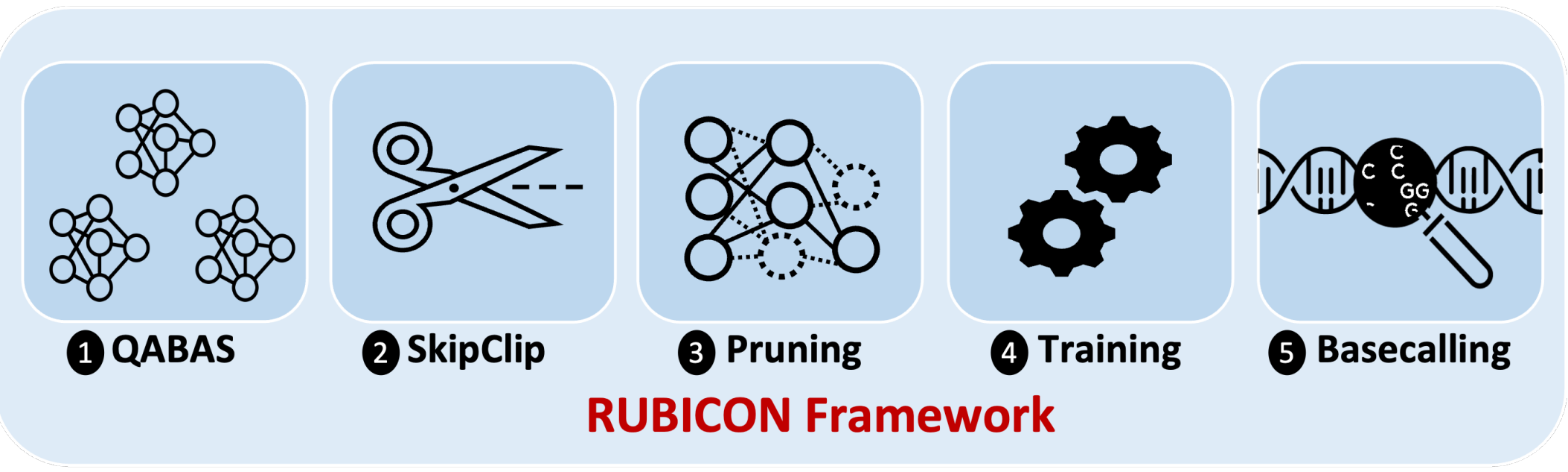
KEY OBSERVATION 2: Effect of Quantization

Current basecallers use floating-point precision to represent each neural network layer present in a basecaller.
Basecallers can provide full accuracy with 4x lower bits for weights and activations.



Our goal is to develop a comprehensive framework for specializing and optimizing a deep learning-based basecaller that provides high efficiency and performance

3: RUBICON: A Framework for Designing Efficient Deep Learning-Based Genomic Basecallers

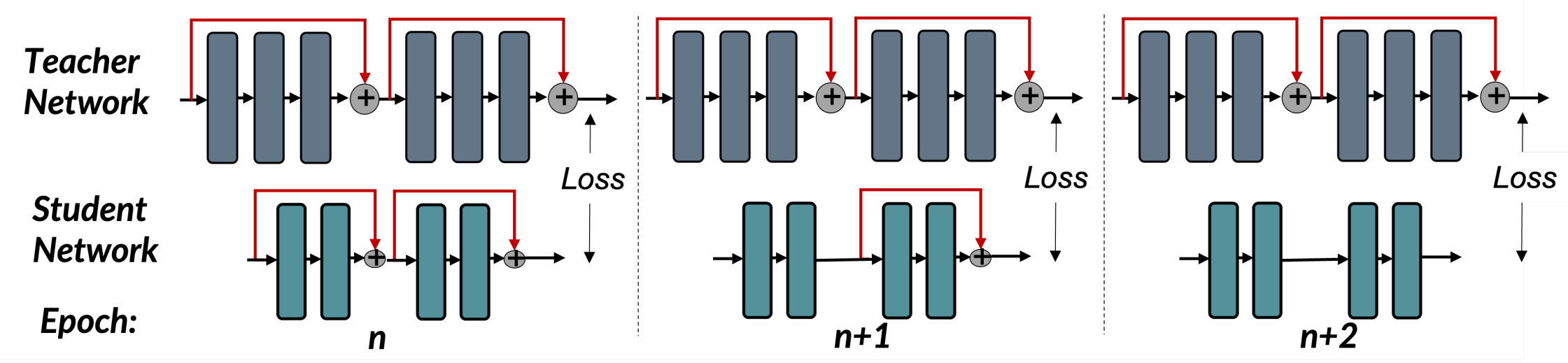


RUBICON provides **five key modules:**

(1) **QABAS:** Quantization-aware basecalling architecture search
(2) **SkipClip:** Skip connection removal by teaching
(3) **Pruning:** Structured and unstructed pruning with knowledge distillation
(4) **Training:** Model training with knowledge distillation
(5) **Basecalling:** Integrated official ONT basecalling modules

SkipClip: Skip Connection Removal by Teaching

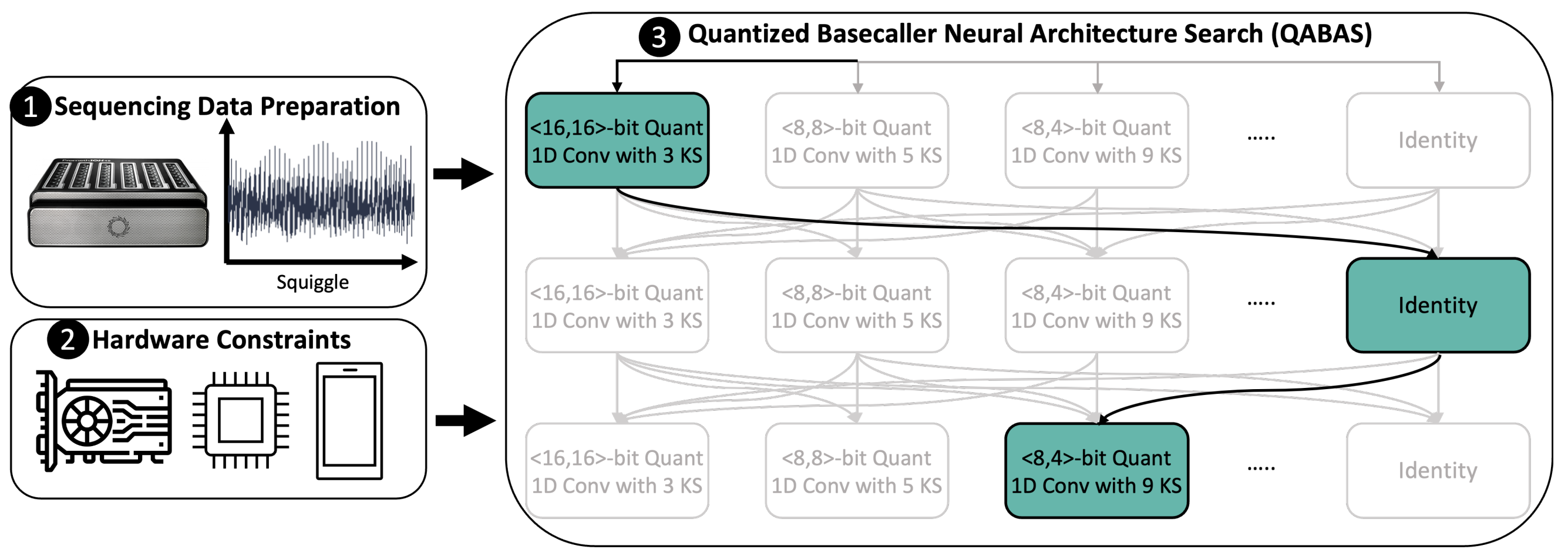
- SkipClip removes all the skip connections present in modern basecallers to reduce resource and storage requirements without any loss in basecalling accuracy
- SkipClip uses knowledge distillation, where we train a smaller network (student) without skip connections to mimic a pre-trained bigger network (teacher) with skip connections



SkipClip removes a skip connection from the student network after every *n* epochs

QABAS: Quantization-Aware Basecalling Architecture Search

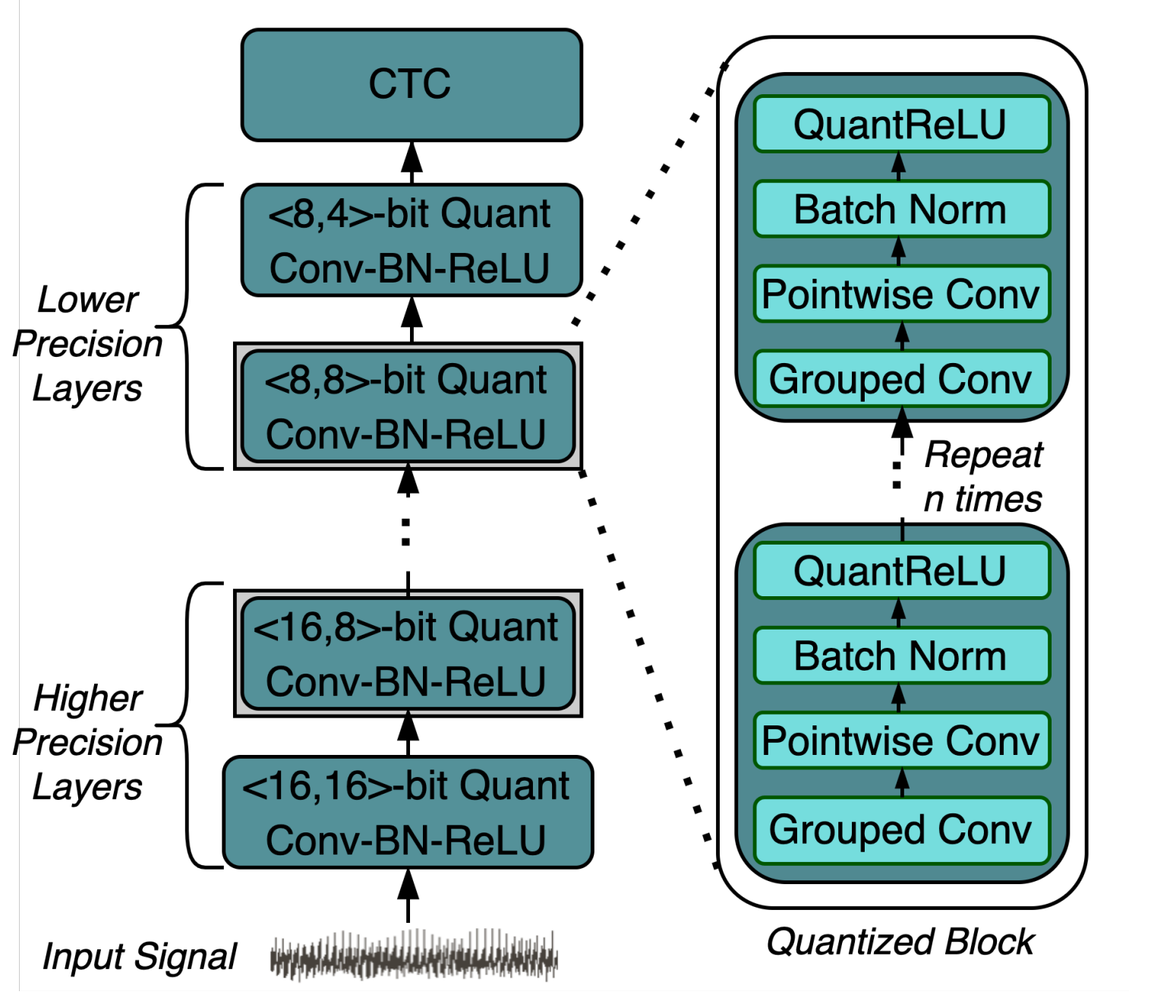
- QABAS automates the process of finding efficient and high-performance hardware-aware genomics basecallers
- QABAS uses neural architecture search (NAS) to evaluate millions of different basecaller architectures


1 Sequencing Data Preparation
Squiggle
2 Hardware Constraints
3 Quantized Basecaller Neural Architecture Search (QABAS)
<16,16>-bit Quant 1D Conv with 3 KS
<8,8>-bit Quant 1D Conv with 5 KS
<8,4>-bit Quant 1D Conv with 9 KS
.....
Identity

During the search phases, QABAS automatically quantizes a neural network model by exploring and finding the best bit-width precision (e.g., 4-b, 8-b, and 16-b) for each neural network layer while jointly searching for the best kernel size (KS) and the number of layers (by using identity operator)

RUBICALL: A Hardware-Optimized Basecaller

- RUBICALL is the first hardware-optimized basecaller that uses mixed-precision computation
- RUBICALL **is** developed using QABAS and SkipClip from RUBICON



4: Evaluation & Key Results

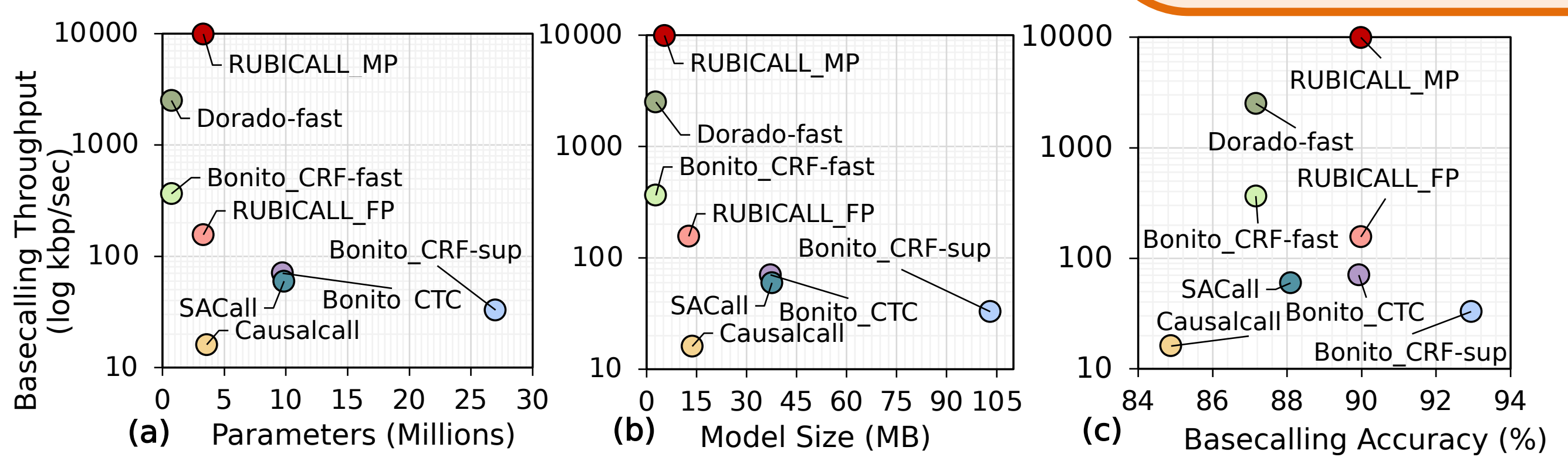
Comparison of RUBICALL with State-of-the-art Basecallers

We evaluate RUBICALL using:

(1) Versal ACAP VC2802, a cutting-edge spatial vector computing system from AMD-Xilinx (RUBICALL-MP) using mixed-precision computation
(2) AMD Mi50 GPU (RUBICALL-FP) using 32-bit floating-point precision computation

Comparison to six state-of-the-art basecallers:

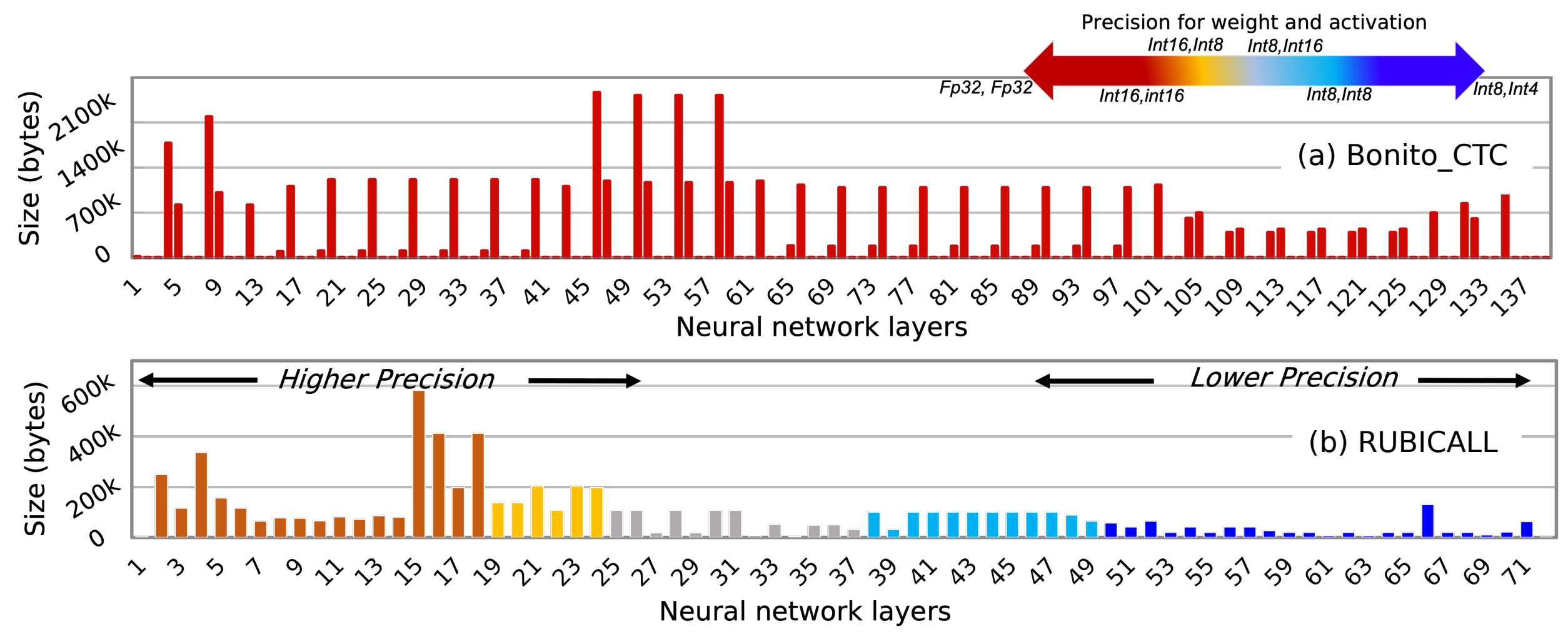
(1) Bonito-CTC, an expert-designed convolutional neural network-based basecaller from ONT
(2) Bonito-CRF-fast, a throughput-optimized recurrent neural network-based basecaller from ONT
(3) Bonito-CRF-sup, an accuracy-optimized recurrent neural network-based basecaller from ONT
(4) Dorado-fast, a LibTorch version of Bontio-CRF_fast that is optimized for low precision
(5) SACall, a transformer-based basecaller with attention mechanism
(6) Causalcall, a state-of-the-art hand-tuned basecaller



KEY OBSERVATION

RUBICALL-MP provides the ability to basecall accurately, quickly, and efficiently scale basecalling by providing reductions in both model size and neural network model parameters.

Explainability Into QABAS Results



KEY OBSERVATION

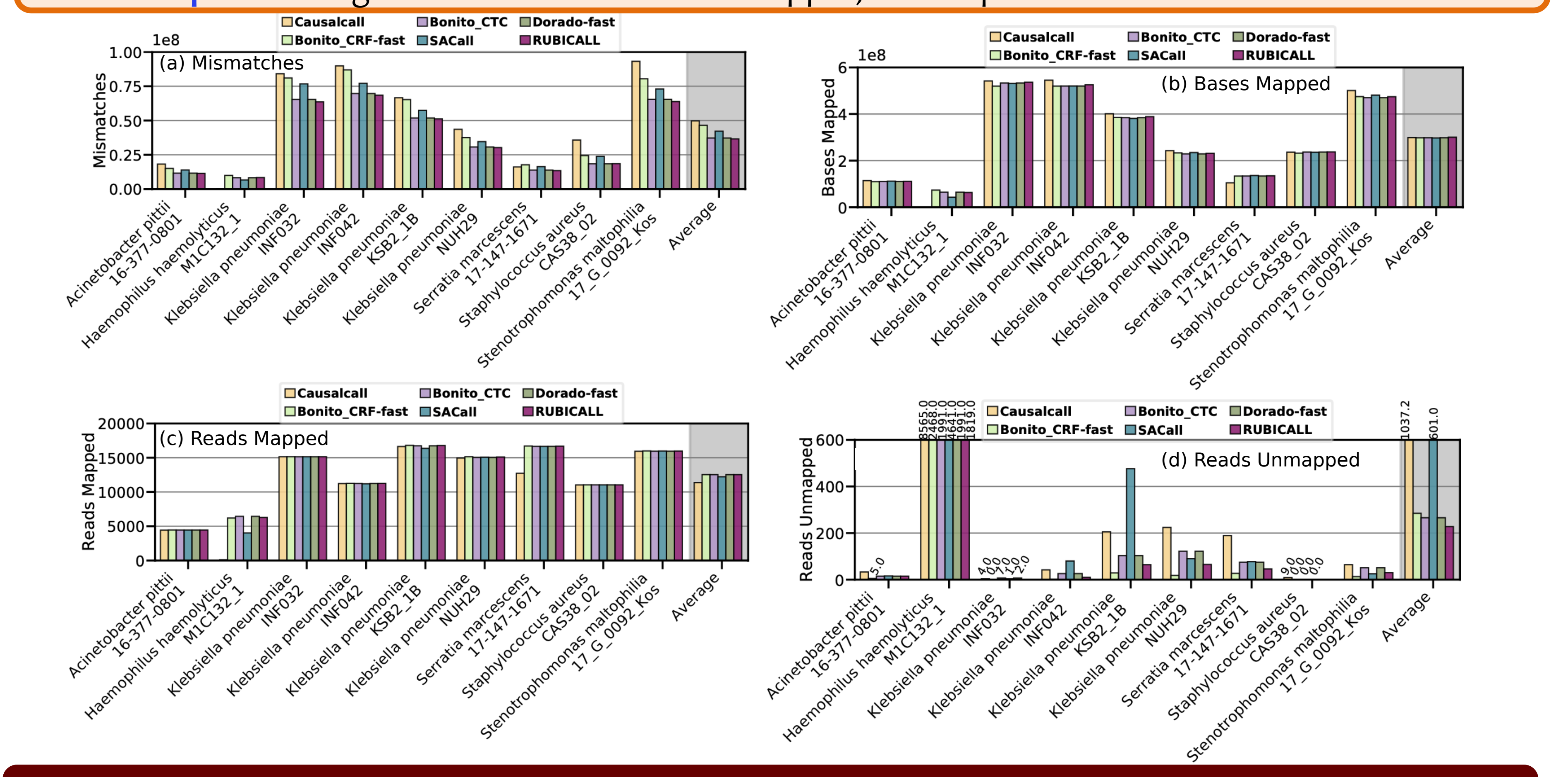
QABAS uses more bits in the initial layers than the final layers in RUBICALL.
QABAS learns that the input to RUBICALL uses an analog squiggle that requires higher precision, while the output is only the nucleotide bases (A, C, G, T), which can be represented using lower precision.

KEY OBSERVATION

RUBICALL has every layer quantized to a different quantization domain. The state-of-the-art basecallers use the same floating-point precision for all the neural network layers, which leads to high processing and memory demands.

Downstream Analysis: Read Mapping

We map the resulting basecalled reads from each evaluated basecaller to the reference genome of the same species using the state-of-the-art read mapper, minimap2



KEY RESULTS

- **RUBICALL provides, on average, 2.97% higher basecalling accuracy with 3.96x higher basecalling throughput** compared to the fastest basecaller (Dorado-fast)
- **RUBICALL uses 6.88x and 2.94x lower model size and parameters** than an expert-designed basecaller (Bonito_CTC), respectively
- **RUBICALL provides 141.15x higher basecalling throughput without any loss in basecalling accuracy** compared to Bonito_CTC by leveraging mixed precision computation
- **RUBICALL provides 301.92x higher basecalling throughput** compared to the most accurate basecaller (Bonito_CRF-sup)
- **RUBICALL provides the highest-quality read mapping with largest number of mapped bases and mapped reads** than our evaluated basecallers