

Benchmark Highlight

Selected results from nine models running on NukeTorch. The full reports include the other results, configurations, and trade-offs.

One Apple M4 Max, 40-core GPU, 128 GB unified memory, macOS 26.7. Author-run; no independent reproduction yet.

Model	Reference	Speed single	Speed batch	RAM	Energy per item
Decider 2B	PyTorch MPS	5.29× / 8.24×	7.32×	48% less	87% less
Laya	PyTorch MPS	1.68× / 1.42×	2.23×	65% less	56% less
Von	PyTorch MPS	4.15× / 1.74×	4.28×	70% less	70% less
Parakeet TDT 0.6B v3	MLX	1.69×	2.97×	8% less	53% less
Whisper large-v3-turbo	MLX	1.18×	1.77×	33% less	41% less
Whisper tiny	MLX	2.02×	2.54×	49% less	49% less
F5-TTS	MLX	1.50×	1.80×	73% less	48% less
Fish Speech S2 Pro	MLX	1.21×	4.41×	18% less	73% less
Kokoro 82M	MLX	2.53×	3.37×	74% less	67% less

Speed. Reference median time divided by NukeTorch median time. Paired single-request figures are short / long text.

RAM and energy. RAM is single-request peak process footprint. Energy per item is estimated whole-machine energy from battery-controller readings, with one energy run per engine. Percentages compare NukeTorch with the named reference.

Selection and conditions. These are the primary reference comparisons in the full summary. Laya uses PyTorch's fastest batch setting; Parakeet uses its default; F5-TTS uses its original five-pass batch series. Speed and energy use separate runs. F5-TTS and Fish energy cover 170 items; their speed batches cover 1,000. Precision and execution settings can differ. Decider's default batch has lower answer agreement than its single-request configuration.

Model reports and measurement notes: nuketorch.ai.