Algorithm-Hardware Implications of Softmax Approximations for In-Memory Computing based LLM Accelerators

Jan Finkbeiner[(1,2)], Sebastian Siegel[(1)], Chirag Sudarshan[(1)], Yuankang Zhao[(1,2)], John Paul Strachan[(1,2)], Emre Neftci[(1,2)]

[(1)] Forschungszentrum Jülich GmbH, [(2)] RWTH Aachen University

As large language models (LLMs) continue to scale in both size and particularly in sequence length, the softmax-attention mechanism of the underlying transformer architecture has emerged as a critical bottleneck. While in-memory computing (IMC) based architectures offer efficient acceleration for vector-matrix multiplications (VMMs), integrating softmax remains challenging due to its computational complexity, including long critical paths and high-precision arithmetics, resulting in significant hardware overhead. This work presents a comprehensive analysis of softmax approximations and softmax precision requirements in modern LLMs spanning model sizes from 125M to 405B parameters across several natural language processing (NLP) benchmarks. Based on these findings, we present an IMC-optimized reformulation of softmax for attention that results in negligible accuracy degradation. These optimizations include (a) transformation of complexity from sequence-length-dependency to the much smaller attention head dimension and (b) merging of softmax components within the adjacent IMC's VMM operations. In this work, we present both digital and analog implementations that surpass state-of-the-art in terms of area efficiency and latency. Our digital approach reduces area overhead to 81.2 µm^2 and energy to 0.69 pJ per input-output pair, resulting in more than 20× lower energy-delay-area product compared to state-of-the-art. Our analog implementation achieves 37.1 µm^2 and 0.33 pJ by leveraging a spike-based encoding. These results highlight the importance of rethinking softmax computation for IMC to enable scalable transformer accelerators.

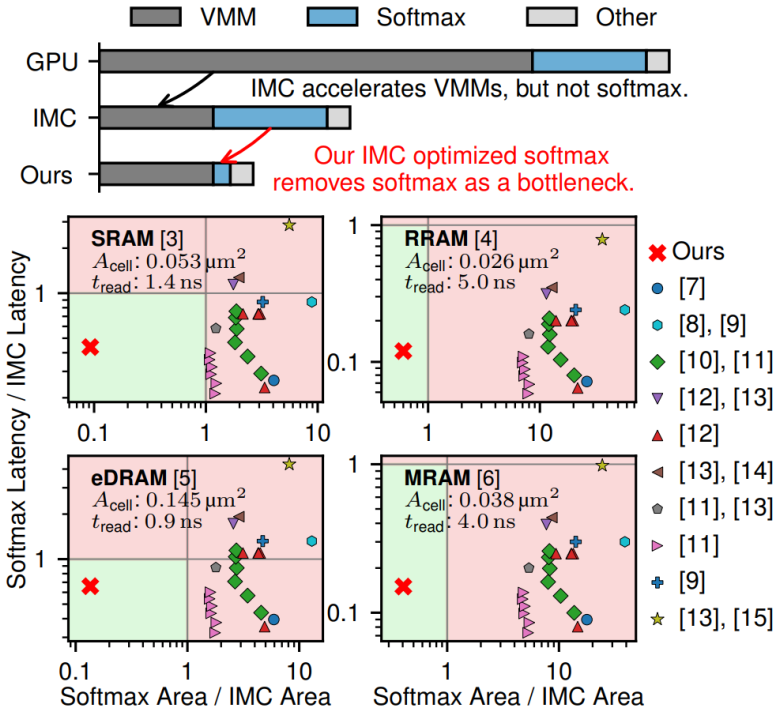

Figure 1: Top: Illustration of this work's motivation highlighting the need to remove softmax as bottleneck for IMC based LLM generation; Bottom: Comparison of normalized area and latency requirements for state-of-the-art softmax implementations versus ours, relative to IMC-VMM. The results are also normalized as per the memory technology. Two citations refer to the hardware reimplementation of previously published algorithm in the other. A_{cell} = cell area and t_{read} = read latency.