29.2 A 28nm 27.5TOPS/W Approximate-Computing-Based Transformer Processor with Asymptotic Sparsity Speculating and Out-of-Order Computing

Yang Wang[1], Yubin Qin[1], Dazheng Deng[1], Jingchuan Wei[1], Yang Zhou[1], Yuanqi Fan[1], Tianbao Chen[2], Hao Sun[1], Leibo Liu[1], Shaojun Wei[1], Shouyi Yin[1]

[1]Tsinghua University, Beijing, China
[2]Tsing Micro, Beijing, China

2022 IEEE International Solid- State Circuits Conference (ISSCC) | 978-1-6654-2800-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISSCC42614.2022.9731686

Recently, Transformer-based models have achieved tremendous success in many AI fields, from NLP to CV, using the attention mechanism [1-3]. This mechanism captures the global correlations of input by indicating every two tokens' relevance with attention scores and uses normalized scores, defined as attention probabilities, to weight all input tokens to obtain output tokens with a global receptive field. A Transformer model consists of multiple blocks, named multi-head, working with the attention mechanism. Figure 29.2.1 details the computation of an attention block with query (Q), key (K), and value-matrix (V), computed by tokens and weight matrices. First, Q is multiplied by K^T to generate the attention score matrix. The scores in each row, represented as X_i, indicate a token's relevance with all others. Second, the row-wise softmax with inputs of X_i-X_{max} normalizes attention scores to probabilities (P), expanding the large scores and reducing the small scores exponentially. Finally, probabilities are quantized and then multiplied by V to produce the output. Each output token is a weighted sum of all input tokens, where the strongly related tokens have large weight values. Global attention-based models achieve 20.4% higher accuracy than LSTM for NLP and 15.1% higher accuracy than ResNet-152 for classification.

Global attention contains many weakly related tokens (WR-Tokens) with small scores, introducing 3 challenges for energy-efficient computation on edge devices. (1) For the entire attention block, WR-Tokens take an average of 93.1% energy consumption but have limited influence since softmax reduces small scores to near-zero probabilities, weakening their contribution. (2) In Q×K^T, 34.3% of the computations are redundant as many near-zeros from softmax become zero after n-bit quantization, indicating output sparsity determined by $X_i<X_{max}+\ln(1/2^n)$. This sparsity is hard to predict as X_{max} is a variable and different for each row, leading to a dynamic speculation threshold. (3) In P×V, 65.3% of hardware resources are wasted since near-zero probabilities have many 0-valued MSBs, incurring 0-value partial-product (PPs) computing.

This paper presents a Transformer processor solving these challenges with three key features: (1) A big-exact-small-approximate (BESA) processing element (PE) saves 1.62× MAC power for WR-Tokens with a self-gating approximate multiplier. It computes small values with large errors for energy-saving, while computing large values exactly, matching the error tolerance of attention. (2) A bidirectional asymptotic speculation unit (BASU) skips 46.7% of the redundant computations by capturing sparsity. BASU exploits attention's local properties to find the varying X_{max} rapidly, improving the ability to exploit the presence of sparsity. (3) An out-of-order PE-line computing scheduler (OPCS) improves hardware utilization by 1.81× by reordering operands to enable dovetailing 2 operations into 1 multiplication, which omits "0" PPs computing.

Figure 29.2.2 shows the processor's overall architecture, consisting of 4 attention cores (AC) with BESA PEs, 4 BASUs, 32 OPCSs, a quantizer, a reorder unit and 336KB SRAM. AC has 8 PE lines, each of which has 16 PEs for 8 outputs in 1 row. ACs generate Q, K, V from input tokens accurately. Then, the attention block calculates the strongly related tokens (SR-Tokens) exactly but others approximately. The PE supports dovetailed computing by using PPs computing logic in 1 multiplier for 2 multiplications. BASU contains 8 sign-based splitters with two 128-deep FIFOs, an X_{max} updater and a speculator. During Q×K^T, it controls ACs to prioritize diagonal scores to detect X_{max} rapidly for efficient sparsity speculation. OPCS comprises a softmax unit, a folding unit, and a 4-to-8 asymmetrical BENES network. For P×V, OPCS receives results from the 12b quantizer and reorders operands before sending them to the PE line for computing.

Figure 29.2.3 details the BESA PEs, computing each head's SR-Tokens filtered by the top-k selector from the prior head exactly, but others approximately. For small values, PE detects their magnitude with MSBs to gate the computation of LSBs for energy reduction. It produces 23.5% error, but softmax reduces the error to 0.25%. The previous approximate methods focus on reducing average error, but cannot adapt to the error tolerance of the attention block. The BESA PE has a 12b booth multiplier, consisting of 6 PPs rows with 24 columns. It uses approximate PP generators in row-2 and row-5, outputting exact results if booth values (BVs) are ±1/0; otherwise, inverting the correct results. Columns 9-to-16 use approximate 4-2 compressors, retaining exactness if the first input (A_1) is 0. The multiplier connects approximate PPs to A_1 of approximate compressors, leading to exact compression results if approximate PPs are "0". In approximate mode, a self-gating code generator performs cascaded OR (AND) for positive (negative) data via 6b MSBs to generate a 6b signal with more 0-bits for small values to disable compressors. For exact mode, it modifies 2 BVs to "0" to make approximate PPs be exactly "0", ensuring exact results for approximate compressors. The result is compensated with addition if a modified bit changes from "1" to "0" or subtraction for "0" to "1". The BESA PE reduces 40.8% of the MAC energy for the attention block of GPT-2 with 89.6% accuracy for sentiment analysis, only 0.4% less accuracy than entirely exact computing.

Figure 29.2.4 shows the workflow of BASU, exploiting diagonal-prior computing to capture row-varied X_{max} rapidly for a high speculation ratio. BASU controls AC to perform positive MACs first and terminates negative computation once observing sparsity. Since a specific token always has strong relevance with its near tokens, the operand reorder unit schedules 8 scores on the matrix diagonal to be computed first for finding an approximate X_{max} (X^*_{max}). It produces a 1.54× higher speculation ratio than sequential computing that misses 49.6% of the sparsity. An X_{max} updater compares X^*_{max} with the newly generated maximum score for update, asymptotically approaching X_{max}. For computing X_i, the splitter separately sends 128×2 Q/K operands to positive and negative FIFOs according to the XOR of sign bits. BASU prioritizes the positive FIFO, and the speculator terminates computation of the negative FIFO when observing X_n, satisfying $X_n<X^*_{max}+\ln(1/2^n)$. It is lossless as $X_i<X_n<X^*_{max}+\ln(1/2^n)$. BASU reduces 43.2% of the redundant computations for GPT-2.

Figure 29.2.5 depicts the OPCS, reducing operands' effective bit-width (EBW) with adaptive folding and changing the computation order with an asymmetrical BENES network. It allows merging of operands to omit "0" PPs for higher hardware utilization. OPCS receives attention scores from AC and normalizes scores to probabilities with the LUT-based softmax units. Then, the folding unit fetches 32 probabilities data and locates the leading one (LO). It inverts the bits after the LO if there are more than 3 successive "1"s following the LO. The compensator shifts the operand based on the LO location for results recovery. Then, the four 4-to-8 asymmetrical BENES routers reorder the 32 data items. Each router receives 8 data and fixes 4 of them, determined by the sorting-based matcher. The other 4 data are reordered with a 4-to-8 switch array, which has a static front network, saving 56.7% of power and 46.2% of area. The 32 Vs are also reordered by reusing the control logic. Finally, an output aggregator sends the reordered operands to the PE line for computing. OPCS improves 1.97× energy efficiency for GPT-2 in the summary stage.

Figure 29.2.6 shows the measurement results. Under a voltage of 0.56-to-1.1V with a 50-to-510MHz frequency, the power of the processor is 12.06-to-272.8mW. The peak energy efficiency is 27.56TOPS/W for 0.56V at 50MHz (14.28TOS/W for 1.1V at 510MHz). It is 8.83× better than a GPU [6] and 13.85× higher than A[3] [2] since the BESA PE reduces the energy of WR-Tokens, and BASU omits ineffectual operations. In addition, it achieves 3.73× speedup vs. ELSA [3] with the high speculation efficiency achieved by BASU and significantly reduced PP computing time with OPCS. Moreover, the feed-forward network (FFN) can also use the BESA PE to reduce computation energy. BASU supports global-level redundant token speculation in Q/K/V generation, while OPCS can effectively handle the pruned weights or ReLU-based sparsity in a FFN. They lead to 3.54×, 2.79×, 2.41× higher energy efficiency for GPT-2, ViT, and Swin-Transformer models than without optimization. Figure 29.2.7 shows a summary and die photo with a 6.82mm^2 area.

Acknowledgment:
This work was supported in part by NSFC Grant U19B2041 and NSFC Grant 61774094, National Key Research and Development Program 2018YFB2202600, Beijing Science and Technology Project Z191100007519016, Beijing Innovation Center for Future Chip, and BAAI. Corresponding author: Shouyi Yin (yinsy@tsinghua.edu.cn).

References:

[1] A. Dosovitskiy et al., "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale", *ICLR*, 2021.
[2] T. J. Ham et al., "A[3]: Accelerating Attention Mechanisms in Neural Networks with Approximation," *IEEE HPCA*, pp. 328-341, 2020.
[3] T. J. Ham et al., "ELSA: Hardware-Software Co-design for Efficient, Lightweight Self-Attention Mechanism in Neural Networks," *ACM/IEEE ISCA*, pp. 692-705, 2021.
[4] D. Kadetotad et al., "An 8.93 TOPS/W LSTM Recurrent Neural Network Accelerator Featuring Hierarchical Coarse-Grain Sparsity for On-Device Speech Recognition," *IEEE JSSC*, vol. 55, no. 7, pp. 1877-1887, 2020.
[5] T. Tambe et al., "A 25mm^2 SoC for IoT Devices with 18ms Noise-Robust Speech-to-Text Latency via Bayesian Speech Denoising and Attention-Based Sequence-to-Sequence DNN Speech Recognition in 16nm FinFET," *ISSCC*, pp. 158-160, 2021.
[6] "NVIDIA A100 Tensor Core GPU Architecture," NVIDIA Whitepaper.

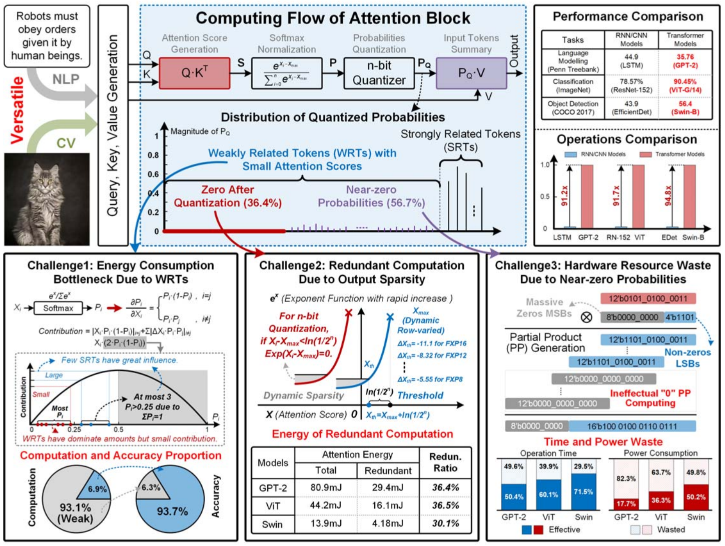

Robots must obey orders given it by human beings.
Versatile
NLP
CV
Query, Key, Value Generation
Computing Flow of Attention Block
Attention Score Generation
Softmax Normalization
Probabilities Quantization
Input Tokens Summary
Q·K^T
n-bit Quantizer
P_Q·V
Output
Distribution of Quantized Probabilities
Weakly Related Tokens (WRTs) with Small Attention Scores
Strongly Related Tokens (SRTs)
Zero After Quantization (36.4%)
Near-zero Probabilities (56.7%)
Performance Comparison
Operations Comparison
Challenge1: Energy Consumption Bottleneck Due to WRTs
Few SRTs have great influence.
WRTs have dominate amounts but small contribution.
Computation and Accuracy Proportion
93.1% (Weak)
6.9%
6.3%
93.7%
Challenge2: Redundant Computation Due to Output Sparsity
Energy of Redundant Computation
Challenge3: Hardware Resource Waste Due to Near-zero Probabilities
Time and Power Waste

Figure 29.2.1: Challenges for energy-efficient attention computation due to weakly related tokens.

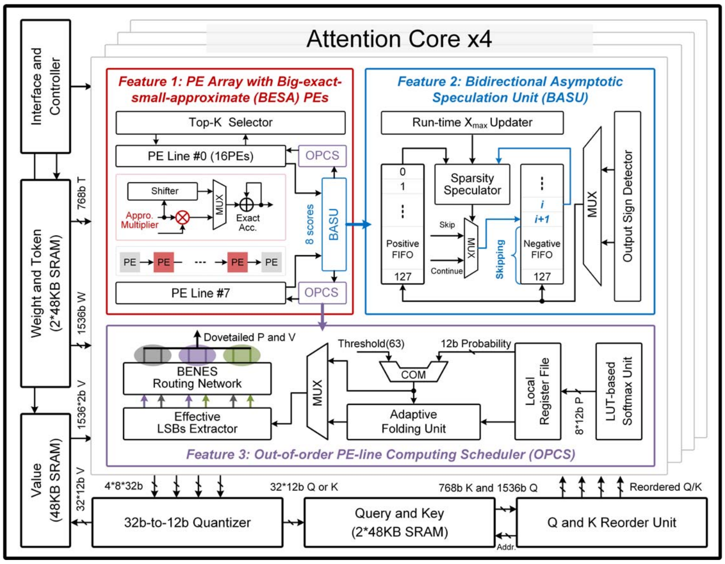


Figure 29.2.2: The overall architecture of the proposed Transformer processor.

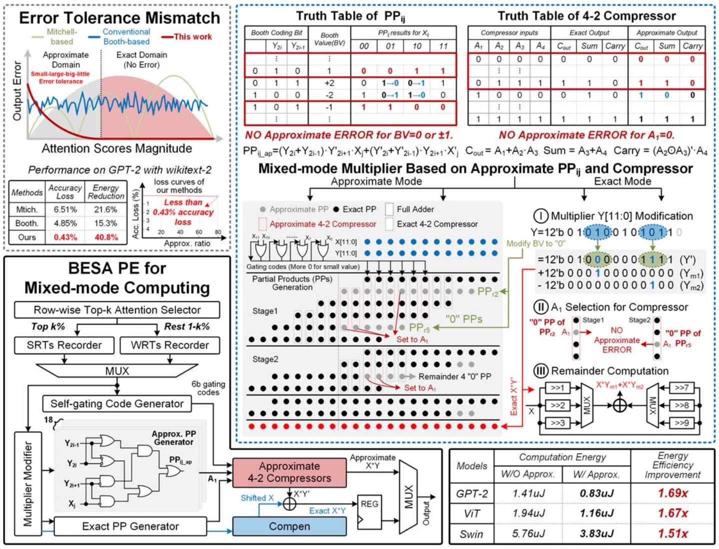

Error Tolerance Mismatch
Attention Scores Magnitude
Performance on GPT-2 with wikitext-2
Truth Table of PP_ij
Truth Table of 4-2 Compressor
NO Approximate ERROR for BV=0 or ±1.
NO Approximate ERROR for A_1=0.
Mixed-mode Multiplier Based on Approximate PP_ij and Compressor
BESA PE for Mixed-mode Computing
Multi-version Top-k Attention Selector
SRTs Recorder
WRTs Recorder
Self-gating Code Generator
Multiplier Modifier
Approximate 4-2 Compressors
Exact PP Generator
Compen
Computation Energy
Energy Efficiency Improvement
GPT-2 1.41uJ 0.83uJ 1.69x
ViT 1.94uJ 1.16uJ 1.67x
Swin 5.76uJ 3.83uJ 1.51x

Figure 29.2.3: Big-exact-small-approximate (BESA) PE with self-gating for mixed-mode computing.

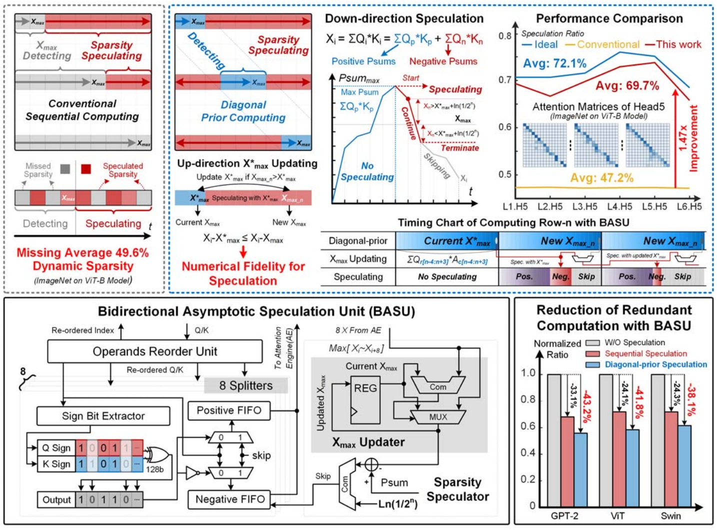

Conventional Sequential Computing
Diagonal Prior Computing
Down-direction Speculation
Up-direction X*max Updating
Missing Average 49.6% Dynamic Sparsity
Numerical Fidelity for Speculation
Performance Comparison
Avg: 72.1%
Avg: 69.7%
Avg: 47.2%
1.47x Improvement
Timing Chart of Computing Row-n with BASU
Bidirectional Asymptotic Speculation Unit (BASU)
Operands Reorder Unit
8 Splitters
Sign Bit Extractor
Positive FIFO
Negative FIFO
X_max Updater
Sparsity Speculator
Reduction of Redundant Computation with BASU
GPT-2
ViT
Swin

Figure 29.2.4: Bidirectional asymptotic speculation unit (BASU) with early diagonal computing and positive followed by negative speculating.

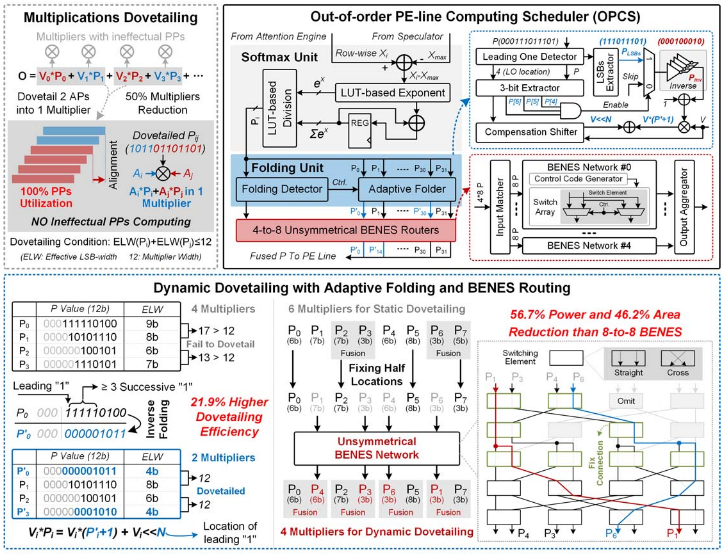
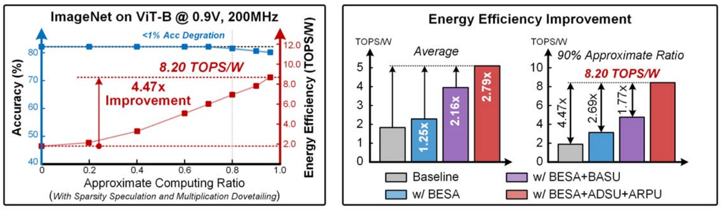

Multiplications Dovetailing
Out-of-order PE-line Computing Scheduler (OPCS)
Softmax Unit
LUT-based Exponent
Folding Unit
Folding Detector
Adaptive Folder
4-to-8 Unsymmetrical BENES Routers
BENES Network #0
BENES Network #4
Dynamic Dovetailing with Adaptive Folding and BENES Routing
56.7% Power and 46.2% Area Reduction than 8-to-8 BENES
21.9% Higher Dovetailing Efficiency
Unsymmetrical BENES Network
4 Multipliers for Dynamic Dovetailing

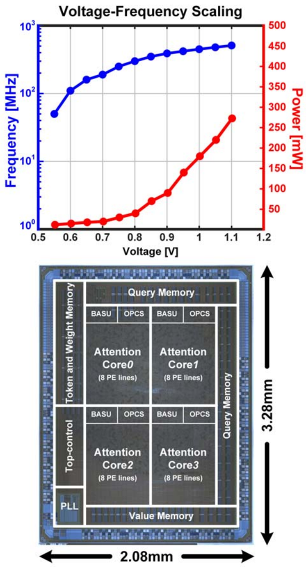
Figure 29.2.5: Out-of-order PE-line computing scheduler with adaptive folding and 4-to-8 unsymmetrical BENES network.



	GPU A100 [6]	JSSC'20 [4]	ISSCC'21 [5]	HPCA'20 [2]	ISCA'21 [3]	This Work
Function[1]	CNN, RNN, SpMM	LSTM, RNN	RNN, CNN, LSTM	Transformer[2]	Transformer	**Transformer[3]**
Sparsity Support	Input	Input	Input	NO	Output	**Input/Output**
Technique (nm)	7	65	7	40	40	**28**
Die Area (mm^2)	826	7.74	19.6	2.08	1.26	**6.82**
Supply Voltage (V)	NA	0.68 – 1.1	0.55 – 0.75	1.1	1.1	**0.56 – 1.1**
Frequency (MHz)	1410	8 - 80	1000 - 1600	1000	1000	**50 – 510**
Precision	FP64/32/16, INT8/4	6 (Weight), 13 (Activation)	HFP8, FP32/16, INT4/2	INT9	INT8, FP16	**INT12**
Power (mW)	400000	1.85 – 67.3	NA	110.42	969.36	**12.06 – 272.8**
Performance (TOPS or TFLOPS)	9.7@FP64 2496@INT4[4]	0.025 – 0.16	8@FP16 102.4@INT4	0.22	1.09	**0.52 – 4.07[5]**
Energy Efficiency (TOPS/W or TFLOPS/W)	0.024@FP64 6.24@INT4[4]	2.45 – 8.93	0.98@FP16 16.5@INT4	1.99	1.12	**1.91 – 27.56[5]**

[1] The main computations of RNN, LSTM and Transformer are all matrix multiplication. It is fair to compare them together.
[2] Only support single token NLP tasks, such as Q/A application.
[3] Support all attention-based tasks.
[4] 50% structured sparsity.
[5] P and V multiplication with 90% weakly related tokens ratio.

Figure 29.2.6: Measurement results and performance comparison table.



Specifications		
Technology		28nm CMOS
Die Area (mm^2)		6.82
SRAM (KB)		336
Voltage (V)		0.56 - 1.1
Frequency (MHz)		50 - 510
Data Precision		INT12
Power (mW)		12.06 – 272.8
Peak Performance (TOPS)	MM [1]	0.522 @ 1.1V, 510MHz
	QK^T	0.522[2] – 1.26[3] @ 1.1V, 510MHz
	PV	0.522[2] – 4.07[4] @1.1V, 510MHz
Energy Efficiency (TOPS/W)	MM [1]	1.91 @ 1.1V, 510MHz
		4.25 @ 0.56V, 50MHz
	QK^T	2.60[2] – 6.27[2][3] @ 1.1V, 510MHz
		5.45[2] – 12.47[2][3] @ 0.56V, 50MHz
	PV	2.60[2] – 14.28[2][4] @ 1.1V, 510MHz
		5.45[2] – 27.56[2][4] @ 0.56V, 50MHz

[1] Including Q, K, V generation, and FFN computation.
[2] 90% weakly related tokens for approximate computing.
[3] 90% output sparsity ratio in Q and K^T multiplication.
[4] 90% near-zero and zero probabilities in P and V multiplication.

Figure 29.2.7: Chip micrograph and performance summary.