LAS: Loss-less ANN-SNN Conversion for Fully Spike-Driven Large Language Models

Long Chen, Xiaotian Song, Yanan Sun *

College of Computer Science, Sichuan University
lchen@stu.scu.edu.cn, songxt@stu.scu.edu.cn, ysun@scu.edu.cn

Abstract

Spiking Large Language Models (LLMs) have emerged as an energy-efficient alternative to conventional LLMs through their event-driven computation. To effectively obtain spiking LLMs, researchers develop different ANN-to-SNN conversion methods by leveraging pre-trained ANN parameters while inheriting the energy efficiency of SNN. However, existing conversion methods struggle with extreme activation outliers and incompatible nonlinear operations of ANN-based LLMs. To address this, we propose a loss-less ANN-SNN conversion for fully spike-driven LLMs, termed LAS. Specifically, LAS introduces two novel neurons to convert the activation outlier and nonlinear operation of ANN-based LLMs. Moreover, LAS tailors the spike-equivalent Transformer components for spiking LLMs, which can ensure full spiking conversion without any loss of performance. Experimental results on six language models and two vision-language models demonstrate that LAS achieves loss-less conversion. Notably, on OPT-66B, LAS even improves the accuracy of 2% on the WSC task. In addition, the parameter and ablation studies further verify the effectiveness of LAS.

Code — https://github.com/lc783/LAS

Introduction

Large Language Models (LLMs), in recent years, have revolutionized artificial intelligence by achieving state-of-the-art performance in language processing (Liu et al. 2024a) and multimodal tasks (Wang et al. 2024). However, there exist significant challenges in the training and inference process of LLMs, particularly the computational complexity and unsustainable energy consumption. This gap has driven an urgent search for more efficient computing paradigms that can support the ever-growing scale of LLMs. Inspired by low-power biological neural systems, Spiking Neural Networks (SNNs) offer a promising alternative (Bohte, Kok, and La Poutré 2000; Gerstner et al. 2014). More specifically, SNNs use discrete, sparse spikes to encode and process information, which can significantly reduce energy consumption compared than traditional Artificial Neural Networks (ANNs) (Davies et al. 2018; Yao et al. 2024).

*Corresponding author.

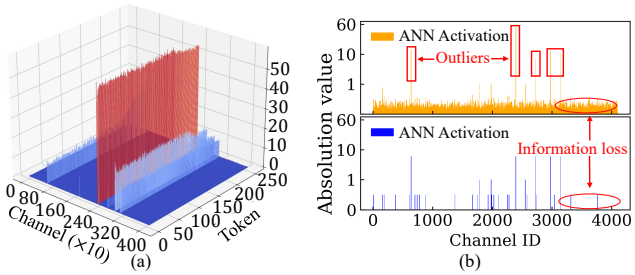


Figure 1: Visualizations of outliers on OPT-7B. (a) Extensive outliers from attention mechanism. (b) The information loss of the converted activations.

To effectively build the SNN models, existing methods can be divided into direct training and ANN-to-SNN conversion. The former one uses surrogate gradients to overcome the challenges posed by the non-differentiable nature of spike events (Neftci, Mostafa, and Zenke 2019; Zenke and Vogels 2021; Song et al. 2024). However, the direct training method naturally suffers from huge computational costs, which are unaffordable for most researchers using this method to build large SNN models in practice. The later one, i.e., ANN-to-SNN conversion, involves converting pre-trained ANNs into SNNs by transferring their learned parameters into a spiking framework, thus preserving accuracy while benefiting from the energy efficiency of spike-based computation (Cao, Chen, and Khosla 2015; Rueckauer et al. 2016, 2017). Through ANN-to-SNN conversion, we can easily obtain the high-performance SNN models.

Although many conversion methods have been successfully applied to Convolutional Neural Networks (CNNs) (Deng and Gu 2021; Li and Zeng 2022), extending them to Transformer-based LLMs remains two main challenges. First, as shown in Figure 1, LLMs often exhibit activation outliers that significantly affect model performance. When these values are represented by spiking neurons, many activations are compressed into a narrow range, leading to severe information loss. Second, transformer-based LLMs have more complex architecture than CNNs. Specifically, LLMs depend on nonlinear operations, e.g., Self-Attention, LayerNorm, GELU, and Softmax. Unfortunately, accurately representing these components using the linear behavior of

spiking neurons still remains a significant challenge.

To address these issues, we propose a loss-less ANN-SNN Conversion for fully spike-driven LLMs, termed LAS. More specifically, to address activation outliers, we propose the Outlier-Aware Threshold neuron, which employs dual Multi-Threshold sub-neurons to process normal and outlier activations separately. Next, to approximate nonlinear operations, we introduce the Hierarchically Gated neuron, leveraging a hierarchical decomposition approximation through grouped spiking sub-neurons. Finally, we design the Spike-Equivalent LLM architecture, converting all key modules into spike-equivalent counterparts without converse error. Our contributions are summarized as follows:

- **Two Novel Neurons.** We propose the Outlier-Aware Threshold Neuron to handle extreme activations via dual sub-neurons, and the Hierarchically Gated Neuron to approximate nonlinear functions through hierarchical decomposition approximation.
- **Spike-equivalent LLM Component.** We present a fully spike-based LLMs by converting all key components into spike-equivalent modules, including self-attention, feed-forward networks, layer normalization, and softmax.
- **SOTA Results on Eight LLMs.** We validate the proposed LAS method on both language and vision-language tasks. Notably, on the large OPT-66B model, LAS surpasses the performance of the vanilla ANN model by 2% in WSC task.

Related Works

Spiking Neurons for ANN-to-SNN conversion

The Integrate-and-Fire (IF) neuron (Cao, Chen, and Khosla 2015) has dominated implementations of ANN-SNN conversion method due to its theoretically established equivalence with ReLU activations under rate coding schemes (Rueckauer et al. 2017; Bu et al. 2023). This characteristic makes IF neurons particularly computationally efficient for implementing ReLU-based model. Additionally, the Leaky Integrate-and-Fire (LIF) (Teeter et al. 2018) neurons improve robustness by adding a leakage mechanism to prevent infinite potential accumulation. Subsequent studies have successfully applied these two neurons to CNNs (Diehl et al. 2015; Rueckauer et al. 2016; Deng and Gu 2021; Li and Zeng 2022; Hao et al. 2024). However, these neurons inherently based on linear dynamics that fundamentally limit their capacity to process nonlinear and non-monotonic functions, e.g., GELU. This intrinsic limitation severely restricts their compatibility with Transformer-based LLMs where such nonlinearities are ubiquitously employed. In contrast, the Few Spikes (FS) neuron (Stöckl and Maass 2021) employs temporal coding with parameterized spike dynamics, which can effectively emulate non-monotonic activation functions over few time steps. Nevertheless, when applying FS to LLMs, a primary issue is the presence of activation outliers, which enlarge the representation step sizes and subsequently will cause significant accuracy loss.

ANN-SNN conversion for Transformer and LLM

Transformer and LLM architecture primarily relies on attention mechanisms and nonlinear operations like Softmax, LayerNorm, and activation functions, which are challenging to directly convert into spiking forms. For example, SpikeZIP-TF (You et al. 2024) aligns activation-quantized Transformer ANNs with SNNs. ECMT (Huang et al. 2024) retains nonlinear functions through an expectation compensation module. SpikeLLM (Xing et al. 2024a) integrates quantization by combining the proposed Generalized IF neurons with saliency-aware spiking, enabling the model to scale up to 70B parameters. However, all these methods fail to convert nonlinear operations into spike, and they are not fully spike-driven architectures. To address this, SpikedAttention (Hwang et al. 2024) introduces trace-driven matrix multiplication and a winner-oriented spike shift to implement spike-based Softmax but struggles with LayerNorm and GELU activations. STA (Jiang et al. 2024) approximates nonlinear operations via universal group operators but still involves partial floating-point computation in Softmax and LayerNorm. In contrast, LAS successfully converts nonlinear operations of LLMs into the spiking forms, thus achieving the fully spike-driven LLMs. Moreover, existing methods designed for converting nonlinear operations only focus on small vision Transformers, yet LAS can get SOTA performance on LLMs, e.g., OPT-66B and Qwen2-VL-7b.

Preliminary

FS neuron is a variation of the standard spiking neuron model. Unlike conventional spiking models, it employs fixed temporal parameters $\theta(t)$ (threshold), $h(t)$ (reset strength), and $d(t)$ (output weight) across T time steps to approximate the activation function $f(x)$ of its ANN counterpart. This approximation is realized by aggregating weighted spikes $\hat{f}(x) = \sum_{t=1}^{T} d(t)s(t)$, where $s(t) \in \{0, 1\}$ denotes the binary spike at timestep t.

The dynamics of neuron begin with an initial membrane potential $v(1) = x$, where x is the gate input. At each timestep t, the membrane potential updates according to :

$$v(t+1) = v(t) - h(t)s(t), \tag{1}$$

which exist a reset mechanism modulated by $h(t)$ after spike emission. A spike $s(t) = 1$ is fire when the membrane potential exceeds the threshold $\theta(t)$:

$$s(t) = \Theta\big(v(t) - \theta(t)\big) = \Theta\Big(x - \textstyle\sum_{j=1}^{t-1} h(j)\, s(j) - \theta(t)\Big), \tag{2}$$

where $\Theta(\cdot)$ represents the Heaviside step function. By optimizing the parameters $\{\theta(t), h(t), d(t)\}$, the FS neuron emulates the target activation $f(x)$ with few time steps.

Methodology

The framework of the proposed LAS method is illustrated in Figure 2. The Transformer-based LLM can be converted to fully spike-driven LLMs by using the proposed Outlier-Aware Threshold (OAT) and Hierarchically Gated (HG) neurons. More specifically, we insert the OAT neuron before every linear layer and matrix operation to deal with the

outliers of LLMs. Moreover, the HG neuron is developed to simulate the nonlinear functions of LLM components.

Spike Neurons Tailored for Spiking LLMs

OAT neuron. To reduce LLM energy consumption, we insert spiking neuron before each linear and matrix operation, replacing floating-point computation with low-power spike events. However, activation outliers in LLMs expand the value range, causing single spiking neuron compress most values into the same bin, resulting information loss. Moreover, the single threshold scheme is incapable of handling the bipolar nature of activations, which include both positive and negative values. To address this, we propose the OAT neuron, which comprises two Multi-Threshold (MT) sub-neurons that separately handle normal and outlier activations. Each MT neuron employs multiple thresholds to handle positive and negative activation efficiently, reducing energy and latency while while preserving fidelity.

Concretely, let $\mathbf{v}(1) \in \mathbb{R}^n$ be the vector of input membrane potentials at time step 1, which serves as gate input. The OAT neuron dynamics follow:

$$\mathcal{M}_{\text{out}} = \Theta\big(|\mathbf{v}(1)| - \theta_{\text{nor}}\big), \quad \mathcal{M}_{\text{nor}} = \mathbf{1} - \mathcal{M}_{\text{out}}, \tag{3}$$

$$s_i = \begin{cases} \text{MT-N}_{\text{nor}}(v_i(1)), & \mathcal{M}_{\text{out}} = 1 \\ \text{MT-N}_{\text{out}}(v_i(1)), & \mathcal{M}_{\text{nor}} = 1 \end{cases}, \tag{4}$$

where $\Theta\big(\cdot\big)$ is the Heaviside function. The normal threshold θ_{nor} determines the binary masks $\mathcal{M}_{\text{out}}$ and $\mathcal{M}_{\text{nor}}$. $\text{MT-N}(\cdot)$ is function of MT neuron. $\text{MT-N}_{\text{nor}}(\cdot)$ processes normal activations using threshold θ_{nor}, while $\text{MT-N}_{\text{out}}(\cdot)$ handles outlier activations with a distinct threshold θ_{out} (where $\theta_{out} > \theta_{nor} > 0$). Finally, s_i is the output spike.

Each MT neuron extends the FS neuron by integrating multiple thresholds, following (Huang et al. 2024; Hao et al. 2024). At time step t, we set $\theta(t) = h(t) = d(t) = \alpha \cdot 2^{-t}$, where α is normal or outlier threshold, enabling a coarse-to-fine approximation of continuous activations. The neuron adopts symmetric base thresholds $\pm\theta(t)$ and discretizes them into H levels. The k-th threshold is defined as $\lambda_k = \theta(t) + \frac{k-1}{H}\theta(t)$. At each time step, the membrane potential $v(t)$ selects the nearest threshold λ_k for spike generation and membrane reset, with dynamics defined as:

$$v(t) = v(t-1) - h(t)z(t), \tag{5}$$

$$s(t) = \begin{cases} 1, & |v(t)| \geq \theta \\ 0, & otherwise \end{cases}, \tag{6}$$

$$d(t) = \begin{cases} \lambda_H, & v(t) \geq 2\theta(t) \\ \lambda_k, & \theta(t) + \frac{k-1}{H}\theta(t) \leq v(t) < \theta + \frac{k}{H}\theta(t) \\ 0, & otherwise \\ -\lambda_k, & -\theta(t) - \frac{k-1}{H}\theta(t) < v(t) \leq -\theta - \frac{k}{H}\theta(t) \\ -\lambda_H, & v(t) \leq -2\theta(t) \end{cases}, \tag{7}$$

here, $k = 1, ..., H$. The MT neuron achieves significant efficiency improvements over conventional spiking approaches. Where rate-coded neurons require N time steps to represent N distinct values, our binary coding scheme reduces this to $\frac{1}{H}\log_2(N)$ time steps.

HG neuron. FS neurons can approximate nonlinear functions with sparse spikes. However, their error is significantly amplified in the presence of activation outliers in LLMs. To address this, we propose the HG neuron, composed of N FS sub-neurons for hierarchical approximation. Specially, the activation space is partitioned into N intervals $\{(\gamma_{i-1}, \gamma_i]\}_{i=1}^{N}$. For each input v_j, a gating function assigns it to a unique sub-neuron FS_i as follows:

$$G(v_j) \in \{i \in \{1, \ldots, N\} \mid \gamma_{i-1} < v_j \leq \gamma_i\} \tag{8}$$

Only the selected sub-neuron FS_i is activated to process v_j:

$$\hat{f}(v_j) = FS_{G(v_j)}(v_j) \tag{9}$$

Each sub-neuron FS_i operates exclusively on the subset of inputs within its assigned interval:

$$V_i := \{v_j \in v \mid G(v_j) = i\}, \hat{V}_i := \{FS_i(v_j) \mid v_j \in V_i\} \tag{10}$$

The final output is obtained by merging all $\hat{V}_i$ while preserving the original input order: $\hat{f}(v) = \{\hat{V}_1, \hat{V}_2, \ldots, \hat{V}_N\}$. Since each sub-neuron handles only a disjoint subset of inputs and they can operate in parallel, the HG neuron incurs no additional energy or latency cost. Thresholds γ_i are dynamically set based on the activation distribution of pre-trained LLMs, ensuring efficient coverage of both typical and outlier values. To train each sub-neuron FS_i without real data, we define a uniform distribution D over the interval $(\gamma_{i-1}, \gamma_i]$ and draw M samples $\{x_j\}$ from D so as to cover all possible inputs in that range. The resulting synthetic dataset $\{(x_j, f(x_j))\}$ serves as our training data.

Spike-Equivalent LLM Components

Spike-Equivalent Self-Attention Self-attention is the key component of Transformer architectures. We introduce Spike-Equivalent Self-Attention, which reformulates conventional self-attention using three spiking-friendly primitives: Spike Activation–Weight (SAW) Multiplication, Spike Activation–Activation (SAA) Multiplication, and Spike-Equivalent Softmax (described in following section).

SAW Multiplication. The input spike trains are projected via fixed weight matrices to produce spiking queries, keys, and values. Concretely, let $W \in \mathbb{R}^{n \times d}$ be a fixed weight matrix and variable features X, we can conclude that:

$$Q = W \cdot X = W \cdot \sum_{t=1}^{T} \theta(t) X_s(t) = \sum_{t=1}^{T} W \cdot \theta(t) X_s(t), \tag{11}$$

where $X_s(t) \in \{0, 1\}^d$ is the binary spike input and $\theta(t)$ is the scalar threshold at time step t. The term $W \cdot v(t) X_s(t)$ represents the weighted spike output for each time step, which is accumulated over time to produce the final result.

SAA Multiplication. This operation is performed between dynamically generated spike-based matrices. Taking the dot-product attention between queries and keys as an example, the spike-based attention score can be expressed as:

$$A_T = Q_s \cdot K_s = \sum_{t=1}^{T} \theta_q(t) Q_s(t) \sum_{t=1}^{T} \theta_k(t) K_s(t), \tag{12}$$

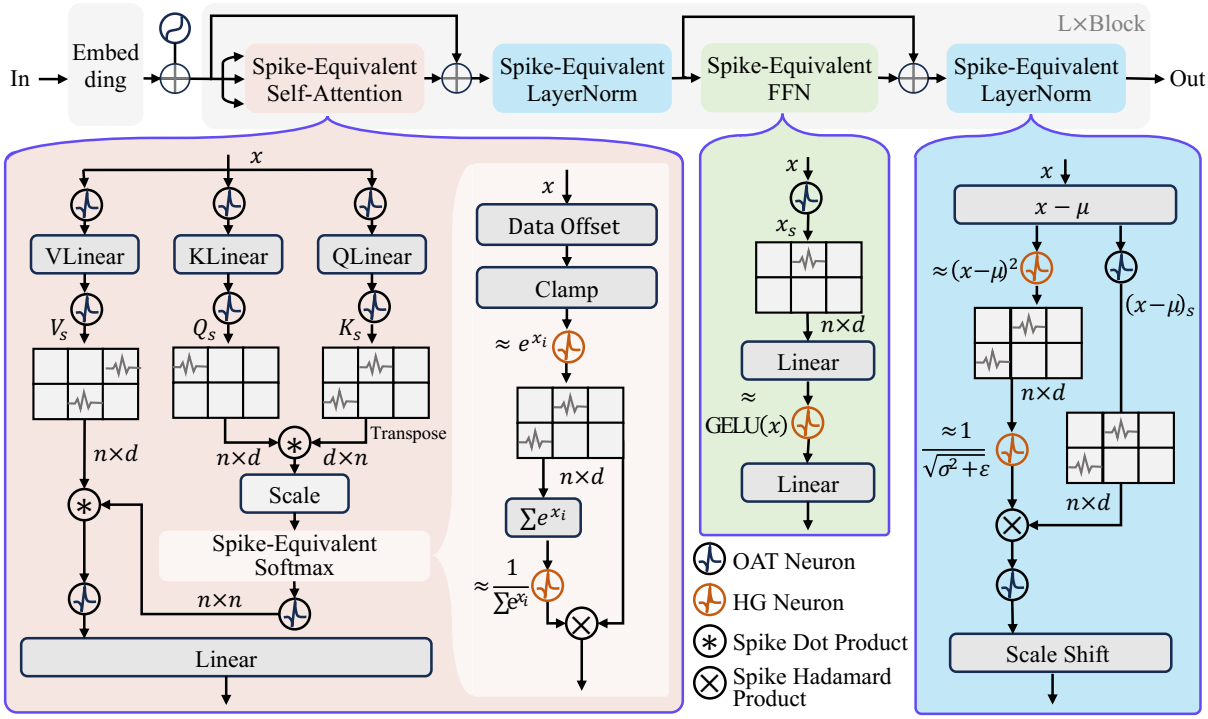


Figure 2: Spike-Equivalent Transformer Block in LAS. The self-attention, feed-forward, softmax and layer normalization modules are fully implemented in spiking form. OAT and HG neurons are specifically designed to handle activation outliers and nonlinear operations. n and d denote the number of tokens and channel dimensions, respectively.

where A_T denotes the attention score matrix accumulated over T time steps, which is equivalent to ANNs. To compute the expected matrix product output incrementally in SNNs, we decompose the calculation at each time step t as follows:

$$\begin{aligned} A_s(t) = \theta_q(t)\theta_k(t)Q_s(t)K_s(t) \\ + \ \theta_q(t)Q_s(t)S_k(t) + S_q(t)\theta_k(t)K_s(t) \end{aligned} \tag{13}$$

where $S_q(t) = \sum_{i=1}^{t-1} \theta_q(i)\, Q_s(i)$ and $S_k(t) = \sum_{i=1}^{t-1} \theta_k(i)\, K_s(i)$ represent the accumulated spikes of query and key. $\theta_q(t)\theta_k(t)$ serves as the spike weight, and the computation only used the binary operations. This design eliminates costly multiplications and enables efficient, incremental spike-based attention computation over time. The detailed proof is provided in the **Appendix A1**.

Spike-Equivalent Feed-Forward Network Conventional Feed-Forward Networks (FFNs) consist of two linear layers with a nonlinear activation. To reduce energy consumption, we replace all floating-point operations with discrete spike events. This is achieved by first converting the input to each linear layer into binary spike trains use the OAT neuron, followed by approximating the activation function via the HG neuron. Formally, the spike-equivalent FFN is defined as:

$$\mathrm{FFN}(x) = \hat{f}\big(\phi(x)\, W_1 + b_1\big)\, W_2 + b_2, \tag{14}$$

where $\phi(\cdot)$ is the OAT neuron, and $\hat{f}(\cdot)$ is the HG neuron that approximates the activation function. Both components take floating-point inputs and emit binary spike outputs, ensuring that the entire FFN operates purely via spike events without any floating-point computation. An advanced variant, the gated FFN, which has demonstrated improved performance, is detailed in **Appendix A2**.

Approximation for Non-Linearity

To address the mismatch between the high-dimensional input of operations like LayerNorm and Softmax and the unary processing nature of HG neuron, we decompose these operations into the simpler, spike-compatible primitives, and apply HG neurons to approximate nonlinear functions.

Spike-Equivalent LayerNorm. We propose a spike-compatible variant of LayerNorm by separating the standard mean–variance normalization and inverse square root scaling into two stages, both implemented with spike event. The transformation for an input x_i is defined as:

$$\begin{aligned} \hat{\mathrm{LN}}(x_i) &= \gamma \cdot \phi\Big(\phi(x_i - \mu) \circ \hat{f}_{\mathrm{InvSqrt}}(\sigma^2)\Big) + \beta \\ &\approx \gamma \cdot \left(\frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}}\right) + \beta \end{aligned} \tag{15}$$

where $\circ$ represents the spike Hadamard product, following the same implementation as in Eq.(12), and σ^2 is computed by spike-based squaring and summing x, with the squaring operation itself approximated by HG neurons. The function $\hat{f}_{\mathrm{InvSqrt}}(\cdot)$ employs HG neuron to approximate $1/\sqrt{\sigma^2 + \epsilon}$.

Model	S	T	PIQA	ARC	OpenbookQA	Winogrande	COPA	WSC	RTE
Flipped-11B (Ye et al. 2022)	✗	N/A	60.34	30.81	17.60	57.85	67.00	65.57	52.71
T0-11B (Sanh et al. 2022)	✗	N/A	73.67	68.39	29.00	62.98	81.00	75.09	84.48
Pythia-12b (Biderman et al. 2023)	✗	N/A	76.00	70.08	26.60	63.54	84.00	81.68	55.60
UL2-20B (Tay et al. 2023)	✗	N/A	74.59	55.18	22.00	64.01	84.00	78.02	52.71
BLOOM-176B (Le Scao et al. 2023)	✗	N/A	77.00	75.93	47.20	67.00	84.00	-	57.40
OPT-2.7B (Zhang et al. 2022)	✗	N/A	73.78	60.73	25.00	61.33	77.00	78.02	55.25
LAS (OPT-2.7B)	✓	16	73.61	61.24	24.40	60.62	78.00	78.02	54.97
OPT-7B (Zhang et al. 2022)	✗	N/A	76.26	65.57	27.60	65.43	81.00	82.05	55.25
FAS (OPT-7B) (Chen et al. 2025)	✓	16	73.23	64.73	27.00	60.38	83.00	77.66	55.60
LAS (OPT-7B)	✓	16	76.22	65.95	27.40	65.75	80.00	81.69	55.96
OPT-13B (Zhang et al. 2022)	✗	N/A	75.95	67.13	27.20	65.27	81.00	82.78	58.12
LAS(OPT-13B)	✓	16	76.28	67.38	26.20	65.51	80.00	82.05	57.40
OPT-30B (Zhang et al. 2022)	✗	N/A	77.58	70.03	30.20	68.35	82.00	82.42	57.76
LAS (OPT-30B)	✓	16	77.80	70.24	30.80	68.35	82.00	81.32	57.76
OPT-66B (Zhang et al. 2022)	✗	N/A	78.73	71.72	30.40	69.98	85.00	82.78	60.55
LAS (OPT-66B)	✓	16	**78.67**	**71.25**	**32.00**	**68.27**	**85.00**	**85.71**	**59.93**

Table 1: Comparison of zero-shot task performance between LAS and other models across different parameter scales.

Spike-Equivalent Softmax. The Softmax function for an input vector $z \in \mathbb{R}^n$ is given by :

$$\sigma_i(z) = \frac{\exp(z_i)}{\sum_{j=0}^{n-1}\exp(z_j)} = \frac{\exp\big(z_i - \max(z)\big)}{\sum_{j=0}^{n-1}\exp\big(z_j - \max(z)\big)}, \tag{16}$$

where $\max(z)$ is used to stabilize the exponential. This formulation consists of exponentiation, max-subtraction, and reciprocal normalization. Although the HG neuron can approximate the exponential and reciprocal functions, it cannot directly capture the dynamic subtraction of $\max(z)$. To address this, we reconstruct $z_i - \max(z)$ in the spike form. Let $z_i(t)$ be the input of neuron i at time step t. We define a corrected spike output:

$$\hat{z}_i(t) = z_i(t) + \max\Big(\sum_{j=1}^{t-1} z(j)\Big) - \max\Big(\sum_{j=1}^{t} z(j)\Big), \tag{17}$$

where $\hat{z}_i(t)$ is $z_i - \max(z)$ output at time step t, the detailed derivation is provided in **Appendix A3**. Finally, we use HG neurons to approximate the remaining nonlinearities. Denoting $\hat{f}_{\exp}(\cdot)$ and $\hat{f}_{\text{inv}}(\cdot)$ as the HG neuron approximation of the exponential function and the reciprocal, respectively. The spike-equivalent output is computed as :

$$\hat{\sigma}_i(z) = \hat{f}_{\exp}\big(\hat{z}_i\big) \circ \hat{f}_{\text{inv}}\Big(\sum_{j=0}^{n-1}\hat{f}_{\exp}\big(\hat{z}_j\big)\Big). \tag{18}$$

This design implements Softmax normalization in a fully event-driven manner, making it compatible with neuromorphic accelerators.

Experiments

Experimental Setup

To evaluate our method, we converted pre-trained BERT-base, the OPT family (2.7B–66B), GPT-2, LLaVA1.5-7B, and Qwen2-VL-7B into spiking LLMs with 16 time steps. We then assessed language understanding on GLUE and zero-shot reasoning on PIQA, ARC, OpenBookQA, Winogrande, COPA, WSC, and RTE. We assessed language generation on Enwik8 and WikiText-103. Finally, we measured multimodal performance on ScienceQA, RealWorldQA, BLINK, POPE, HallusionBench, MMStar, and MME. Additional details are provided in **Appendix A4**.

Overall Results

Experiments on NLG Tasks. We evaluate LAS using various scales of OPT and GPT-2 models, as shown in Tables 1 and 4. On zero-shot tasks, LAS consistently maintains or improves accuracy across OPT models from 2.7B to 66B. For example, it achieves 32.00 on OpenbookQA and 85.71 on WSC, exceeding the original OPT-66B scores of 30.40 and 82.78, respectively, using only 16 time steps. Notably, even though BLOOM-176B was evaluated in a one-shot setting, our 66B model outperforms it on four tasks, highlighting our LAS's superiority. Furthermore, LAS consistently reflects the expected trend of increased accuracy with larger model scales, indicating faithful preservation of capabilities across sizes. In GPT model, LAS matches GPT-2 on Enwik8 with a score of 0.97 and shows only a slight degradation on WikiText-103, while substantially outperforming existing direct training and ANN-SNN conversion methods.

Experiments on NLU Tasks. FAS achieves near-lossless conversion for language understanding tasks. As presented in Table 2, with 16 time steps, LAS reaches 92.55% accuracy on SST-2, which closely matches the original BERT's 92.66%, and even surpasses the ANN by 0.02% on QQP. It also significantly outperforms existing SNN models; for example, SpikingBERT achieves only 88.19% accuracy on SST-2 despite using 60 time steps. Notably, our method narrows the accuracy gap to under 0.1% across all NLU tasks, demonstrating the effectiveness of our lossless conversion.

Model	S	T	QQP	MNLI-m	SST-2	QNLI	RTE	MRPC	STS-B
BERT (Devlin et al. 2019)	✗	N/A	90.71	84.11	92.66	90.99	64.98	84.56/89.19	88.70/88.48
CBoW (Wang et al. 2018)	✗	N/A	75.00	57.10	79.50	62.50	71.90	75.00/83.70	70.60/71.10
BiLSTM (Wang et al. 2018)	✗	N/A	85.30	66.70	87.50	77.00	58.50	77.90/85.10	71.60/72.00
BiLSTM + Attn, CoVe (Wang et al. 2018)	✗	N/A	83.50	67.90	89.20	72.50	58.10	72.80/82.40	59.40/58.00
GenSen (Subramanian et al. 2018)	✗	N/A	82.60	71.40	87.20	62.50	78.40	80.40/86.20	81.30/81.80
SNN-TextCNN (Lv, Xu, and Zheng 2023)	✓	50	0.00*	64.91	80.91	64.91	47.29	-/80.62	0.00*/-
spikeBERT (Lv et al. 2024)	✓	4	68.17	71.42	85.39	66.37	57.47	-/81.98	-/18.73*
SpikeLM (Xing et al. 2024b)	✓	4	-	77.10	87.00	85.30	69.00	-/85.70	84.90/-
SpikingBERT (Bal and Sengupta 2024)	✓	60	86.82	78.10	88.19	85.20	66.06	79.17/85.15	82.20/81.90
SPR (BERT) (Hao et al. 2023a)	✓	8 (16†)	87.48	77.56	90.48	87.75	64.98	78.68/85.76	86.71/86.50
QCFS (BERT) (Bu et al. 2023)	✓	8	88.42	79.57	89.91	86.80	56.68	78.92/85.37	86.18/85.82
COS (BERT) (Hao et al. 2023b)	✓	8 (8†)	88.85	79.91	89.79	87.37	63.18	79.66/86.33	86.49/86.23
FAS (BERT) (Chen et al. 2025)	✓	4	90.38	82.77	91.17	90.13	**66.06**	**86.02/90.22**	87.46/87.26
LAS (BERT)	✓	16	**90.73**	**84.19**	**92.66**	**90.92**	65.34	84.80/89.42	**88.76/88.53**

Table 2: Comparing LAS with SOTA models of BERT on the GLUE evaluation set. S denotes whether an SNN or not. T is the time steps. * denotes non-convergence. † indicates additional time steps required to gather the necessary prior information. The three blocks group models of non-SNN, direct trained and ANN-SNN converted.

Model	ScienceQA	RealWorldQA	BLINK	POPE	HallusionBench	MMStar	MME
LLaVA1.5-7b (Liu et al. 2024b)	67.22	52.41	41.22	82.81	81.36	33.40	1732.41
Qwen2-VL-7b (Bai et al. 2023)	83.09	67.18	51.55	85.32	130.66	53.73	2240.55
LAS (LLaVA1.5-7b)	66.38	49.15	41.01	80.79	67.93	33.86	1613.04
LAS (Qwen2-VL-7b)	**81.40**	**66.79**	**52.18**	**84.77**	**125.81**	**53.86**	**2222.59**

Table 3: Compare the performance of LAS and SOTA multimodal LLMs on vision-language tasks.

Model	S	T	En8	WT
GPT-2 (Radford et al. 2019)	✗	N/A	0.96	16.53
Transformer-SSA (Hussain 2023)	✗	N/A	1.02	16.91
AstroSNN (Shen et al. 2023)	✓	–	1.14	32.97
spikeGPT (Zhu et al. 2023)	✓	1024	1.26	18.01
SPR (GPT-2) (Hao et al. 2023a)	✓	32 (16†)	1.01	19.24
QCFS (GPT-2) (Bu et al. 2023)	✓	32	1.02	19.36
COS (GPT-2) (Hao et al. 2023b)	✓	16 (16†)	1.01	19.15
FAS (GPT-2) (Chen et al. 2025)	✓	16	0.97	16.84
Our (GPT-2)	✓	**16**	**0.97**	**16.79**

Table 4: Comparing LAS with SOTA GPT models on the NLG dataset. 'En8' stands for Enwik8, with BPB as the metric. 'WT' is WikiText-103 using perplexity. The lower the better for both metrics.

Experiments on Vision-Language Tasks. As shown in Table 3, LAS achieves strong performance with minimal degradation. On Qwen2-VL-7B, LAS achieves scores of 66.79 on RealWorldQA and 84.77 on POPE, closely matching the ANN baseline, and even outperforming it on BLINK and MMStar. Notably, although the LLaVA1.5 model has the same parameter size as Qwen2-VL-7B, it is inferior compared to Qwen2-VL-7B. LAS preserves this performance gap, highlighting that the quality of the pre-trained ANN significantly influences the converted SNN. This highlights the importance of choosing high-quality pre-trained LLMs for conversion. More case studies on the vision-language tasks are provided in **Appendix A5**.

Energy Analysis

We first compare the energy consumption of nonlinear operations. A native GELU implementation requires approximately 70 FLOPs per activation due to the exponents in tanh. STA (Jiang et al. 2024) introduce a Universal Group Operator (UGO) to approximate GELU, reducing computational cost by 59%. In contrast, our HG neuron encodes the GELU nonlinearity using at most 16 spikes, reducing the energy cost to near zero while maintaining high fidelity.

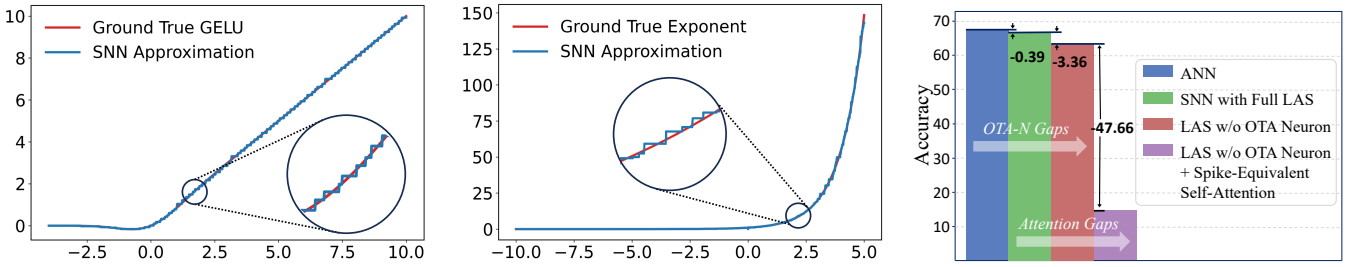
We evaluate the energy efficiency of our SNN on the STS-B task by comparing its energy consumption to the original BERT model across varying threshold levels in the MT neuron, as shown in Table 5. At $H = 1$, the energy ratio is 1.03, which is comparable to that of the ANN model. As H increases, efficiency improves rapidly. The ratio drops to 0.63 at $H = 3$, 0.48 at $H = 5$, and 0.41 at $H = 10$. For $H > 5$, the ratio remains below 0.50, such as 0.39 at $H = 15$, indicating consistent energy savings. The detailed energy estimation methods are provided in **Appendix A6**.

Model	Metric	Original (ANN)	Ours (SNN)						
			H=1	H=3	H=5	H=7	H=10	H=12	H=15
BERT (Devlin et al. 2019)	acc	88.70	88.73	88.80	88.79	88.77	88.76	88.79	88.78
	energy (%)	1	1.03	0.63	0.48	0.50	0.41	0.43	0.39

Table 5: Accuracy and energy consumption of BERT under different H values

Model	S	T	QQP	MNLI-m	SST-2	QNLI	RTE	MRPC	STS-B
BERT (Devlin et al. 2019)	✗	N/A	90.71	84.11	92.66	90.99	64.98	84.56/89.19	88.70/88.48
LAS (BERT)	✓	10	68.64	32.96	49.08	49.73	47.29	68.63/81.29	20.58/26.53
	✓	11	86.29	71.41	79.36	86.14	47.65	76.96/82.06	79.56/82.51
	✓	13	90.69	84.11	92.55	90.91	66.06	84.31/89.12	88.78/88.60
	✓	16	**90.73**	**84.19**	**92.55**	**90.92**	**65.34**	**84.80/89.42**	**88.79/88.58**

Table 6: Comparing BERT with different Time step.



(a) An approximation for GELU

(b) An approximation for exponent.

(c) Ablations on components of Qwen-VL.

Figure 3: Illustrations of nonlinear function approximations and ablation study results.

Parameter and Efficacy Studies

Parameter Study on Time Steps. We evaluate the performance of the spiking BERT model under varying time steps, as shown in Table 6. The results show a nonlinear relationship between performance and time steps. At 16 time steps, the spiking BERT closely matches the ANN model, achieving 90.92% on QNLI compared to 90.99% indicating near-lossless conversion. Performance remains stable down to 13 time steps, still reaching 90.91% on QNLI. However, at 11 time steps, performance degrades significantly, dropping to 86.14% on QNLI and 47.65% on RTE. These results suggest that 13 to 16 time steps are sufficient for LAS to handle activation outliers and nonlinear dynamics, while fewer steps lead to substantial information loss.

Efficacy study of HG neuron. We evaluate the HG neuron's ability to approximate nonlinear functions using GELU and exponential curves with 16 time steps. As shown in Figure 3a and Figure 3b, it closely matches both the shape and transition points of the original functions. The outputs nearly overlap with the true curves, demonstrating high-fidelity approximation of complex nonlinearities.

Ablation Study

We conduct ablation experiments on the RealWorldQA benchmark using the Qwen2-VL-7B model to evaluate the contribution of each LAS component. As shown in Figure 3c, the full LAS model achieves 66.79% accuracy, only 0.39% lower than the original ANN. Replacing the OAT neuron with a single MTN leads to a 3.36% drop, confirming the importance of dual sub-neuron processing for preserving information fidelity. The most substantial decline occurs when spike-equivalent self-attention is removed, reducing accuracy to 15.77%, underscoring its critical role in maintaining model performance. These results highlight that both the OAT neuron and spike-equivalent attention are essential for effective LLM conversion.

Conclusion

This paper proposes a Loss-less ANN-SNN conversion method for fully spike-driven LLMs, termed LAS. Specifically, by introducing two specialized neurons that address activation outliers and nonlinear operations, LAS can transform all floating-point computations of ANN-based LLMs into energy-efficient spike computations. Moreover, the proposed spike-equivalent modules for self-attention, feedforward layers, Softmax function, and layer normalization further eliminate performance degradation. Experiments demonstrate SOTA performance of LAS across language understanding, generation, and multimodal reasoning tasks with only 16 time steps, achieving near-lossless conversion for models up to 66B parameters. To the best of our knowledge, it is the first time obtaining high-performance and fully spike-driven LLMs with such a model size.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant 62276175, Innovative Research Group Program of Natural Science Foundation of Sichuan Province under Grant 2024NSFTD0035, and Major Science and Technology Special Project of Sichuan Province under Grant 2025ZDZX0013.

References

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 1(2): 3.

Bal, M.; and Sengupta, A. 2024. Spikingbert: Distilling bert to train spiking language models using implicit differentiation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 10998–11006.

Biderman, S.; Schoelkopf, H.; Anthony, Q. G.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M. A.; Purohit, S.; Prashanth, U. S.; Raff, E.; et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, 2397–2430. PMLR.

Bohte, S. M.; Kok, J. N.; and La Poutré, J. A. 2000. SpikeProp: backpropagation for networks of spiking neurons. In *ESANN*, volume 48, 419–424. Bruges.

Bu, T.; Fang, W.; Ding, J.; Dai, P.; Yu, Z.; and Huang, T. 2023. Optimal ANN-SNN conversion for high-accuracy and ultra-low-latency spiking neural networks. *arXiv preprint arXiv:2303.04347*.

Cao, Y.; Chen, Y.; and Khosla, D. 2015. Spiking deep convolutional neural networks for energy-efficient object recognition. *International Journal of Computer Vision*, 113: 54–66.

Chen, L.; Song, X.; Song, A.; Chen, B.; Lv, J.; and Sun, Y. 2025. FAS: Fast ANN-SNN Conversion for Spiking Large Language Models. *arXiv preprint arXiv:2502.04405*.

Davies, M.; Srinivasa, N.; Lin, T.-H.; Chinya, G.; Cao, Y.; Choday, S. H.; Dimou, G.; Joshi, P.; Imam, N.; Jain, S.; et al. 2018. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1): 82–99.

Deng, S.-W.; and Gu, S. 2021. Optimal Conversion of Conventional Artificial Neural Networks to Spiking Neural Networks. *ArXiv*, abs/2103.00476.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics*.

Diehl, P. U.; Neil, D.; Binas, J.; Cook, M.; Liu, S.-C.; and Pfeiffer, M. 2015. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. *2015 International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Gerstner, W.; Kistler, W. M.; Naud, R.; and Paninski, L. 2014. *Neuronal dynamics: From single neurons to networks and models of cognition*. Cambridge University Press.

Hao, Z.; Bu, T.; Ding, J.; Huang, T.; and Yu, Z. 2023a. Reducing ann-snn conversion error through residual membrane potential. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11–21.

Hao, Z.; Ding, J.; Bu, T.; Huang, T.; and Yu, Z. 2023b. Bridging the Gap between ANNs and SNNs by Calibrating Offset Spikes. *ArXiv*, abs/2302.10685.

Hao, Z.; Shi, X.; Liu, Y.; Yu, Z.; and Huang, T. 2024. LM-HT SNN: Enhancing the performance of SNN to ANN counterpart through learnable multi-hierarchical threshold model. *arXiv preprint arXiv:2402.00411*.

Huang, Z.; Shi, X.; Hao, Z.; Bu, T.; Ding, J.; Yu, Z.; and Huang, T. 2024. Towards High-performance Spiking Transformers from ANN to SNN Conversion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 10688–10697.

Hussain, M. S. 2023. The information pathways hypothesis: Transformers are dynamic self-ensembles. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 810–821.

Hwang, S.; Lee, S.; Park, D.; Lee, D.; and Kung, J. 2024. Spikedattention: Training-free and fully spike-driven transformer-to-snn conversion with winner-oriented spike shift for softmax operation. *Advances in Neural Information Processing Systems*, 37: 67422–67445.

Jiang, Y.; Hu, K.; Zhang, T.; Gao, H.; Liu, Y.; Fang, Y.; and Chen, F. 2024. Spatio-temporal approximation: A training-free snn conversion for transformers. In *The Twelfth International Conference on Learning Representations*.

Le Scao, T.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A. S.; Yvon, F.; Gallé, M.; et al. 2023. Bloom: A 176b-parameter open-access multilingual language model. *hal-03850124*.

Li, Y.; and Zeng, Y. 2022. Efficient and Accurate Conversion of Spiking Neural Network with Burst Spikes. In *International Joint Conference on Artificial Intelligence*.

Liu, A.; Feng, B.; Xue, B.; Wang, B.; Wu, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; et al. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Lv, C.; Li, T.; Xu, J.; Gu, C.; Ling, Z.; Zhang, C.; Zheng, X.; and Huang, X. 2024. SpikeBERT: A Language Spikformer Learned from BERT with Knowledge Distillation. arXiv:2308.15122.

Lv, C.; Xu, J.; and Zheng, X. 2023. Spiking Convolutional Neural Networks for Text Classification. In *International Conference on Learning Representations*.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-Based Optimization to Spiking Neural Networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Rueckauer, B.; Lungu, I.-A.; Hu, Y.; and Pfeiffer, M. 2016. Theory and tools for the conversion of analog to spiking convolutional neural networks. *arXiv preprint arXiv:1612.04052*.

Rueckauer, B.; Lungu, I.-A.; Hu, Y.; Pfeiffer, M.; and Liu, S.-C. 2017. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in neuroscience*, 11: 294078.

Sanh, V.; Webson, A.; Raffel, C.; Bach, S. H.; Sutawika, L.; Alyafeai, Z.; Chaffin, A.; Stiegler, A.; Le Scao, T.; Raja, A.; et al. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *ICLR 2022-Tenth International Conference on Learning Representations*.

Shen, G.; Zhao, D.; Dong, Y.; Li, Y.; Li, J.; Sun, K.; and Zeng, Y. 2023. Astrocyte-Enabled Advancements in Spiking Neural Networks for Large Language Modeling. *ArXiv*, abs/2312.07625.

Song, X.; Song, A.; Xiao, R.; and Sun, Y. 2024. One-step spiking transformer with a linear complexity. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 3142–3150.

Stöckl, C.; and Maass, W. 2021. Optimized spiking neurons can classify images with high accuracy through temporal coding with two spikes. *Nature Machine Intelligence*, 3(3): 230–238.

Subramanian, S.; Trischler, A.; Bengio, Y.; and Pal, C. J. 2018. Learning General Purpose Distributed Sentence Representations via Large Scale Multi-task Learning. In *International Conference on Learning Representations*.

Tay, Y.; Dehghani, M.; Tran, V. Q.; Garcia, X.; Wei, J.; Wang, X.; Chung, H. W.; Bahri, D.; Schuster, T.; Zheng, S.; Zhou, D.; Houlsby, N.; and Metzler, D. 2023. UL2: Unifying Language Learning Paradigms. In *The Eleventh International Conference on Learning Representations*.

Teeter, C.; Iyer, R.; Menon, V.; Gouwens, N.; Feng, D.; Berg, J.; Szafer, A.; Cain, N.; Zeng, H.; Hawrylycz, M.; et al. 2018. Generalized leaky integrate-and-fire models classify multiple neuron types. *Nature communications*, 9(1): 709.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *BlackboxNLP@EMNLP*.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Xing, X.; Gao, B.; Zhang, Z.; Clifton, D. A.; Xiao, S.; Du, L.; Li, G.; and Zhang, J. 2024a. Spikellm: Scaling up spiking neural network to large language models via saliency-based spiking. *arXiv preprint arXiv:2407.04752*.

Xing, X.; Zhang, Z.; Ni, Z.; Xiao, S.; Ju, Y.; Fan, S.; Wang, Y.; Zhang, J.; and Li, G. 2024b. SpikeLM: Towards General Spike-Driven Language Modeling via Elastic Bi-Spiking Mechanisms. *arXiv preprint arXiv:2406.03287*.

Yao, M.; Richter, O.; Zhao, G.; Qiao, N.; Xing, Y.; Wang, D.; Hu, T.; Fang, W.; Demirci, T.; De Marchi, M.; et al. 2024. Spike-based dynamic computing with asynchronous sensing-computing neuromorphic chip. *Nature Communications*, 15(1): 4464.

Ye, S.; Kim, D.; Jang, J.; Shin, J.; and Seo, M. 2022. Guess the instruction! flipped learning makes language models stronger zero-shot learners. *arXiv preprint arXiv:2210.02969*.

You, K.; Xu, Z.; Nie, C.; Deng, Z.; Guo, Q.; Wang, X.; and He, Z. 2024. SpikeZIP-TF: Conversion is all you need for transformer-based SNN. *arXiv preprint arXiv:2406.03470*.

Zenke, F.; and Vogels, T. P. 2021. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. *Neural computation*, 33(4): 899–925.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhu, R.-J.; Zhao, Q.; Li, G.; and Eshraghian, J. K. 2023. Spikegpt: Generative pre-trained language model with spiking neural networks. *arXiv preprint arXiv:2302.13939*.