Spiking Transformer Networks: A Rate Coded Approach for Processing Sequential Data

Etienne Mueller, Viktor Studenyak, Daniel Auge, Alois Knoll
Technical University of Munich
Munich, Germany
etienne.mueller@tum.de

2021 7th International Conference on Systems and Informatics (ICSAI) | 978-1-6654-2624-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICSAI53574.2021.9664146

Abstract—**Machine learning applications are steadily increasing in performance, while also being deployed on a growing number of devices with limited energy resources. To minimize this trade-off, researchers are continually looking for more energy efficient solutions. A promising field involves the use of spiking neural networks in combination with neuromorphic hardware, significantly reducing energy consumption since energy is only consumed as information is being processed. However, as their learning algorithms lag behind conventional neural networks trained with backpropagation, not many applications can be found today. The highest levels of accuracy can be achieved by converting networks that are trained with backpropagation to spiking networks. Spiking neural networks can show nearly the same performance in fully connected and convolutional networks. The conversion of recurrent networks has been shown to be challenging. However, recent progress with transformer networks could change this. This type of network not only consists of modules that can easily be converted, but also shows the best accuracy levels for different machine learning tasks. In this work, we present a method to convert the transformer architecture to networks of spiking neurons. With only minimal conversion loss, our approach can be used for processing sequential data with very high accuracy while offering the possibility of reductions in energy consumption.**

I. Introduction

An increasing number of applications are benefiting from today's advances in neural networks. Tasks such as machine translation, speech recognition, and object detection are often deployed in power restricted environments and must process large amounts of data while minimizing energy use. By reducing the power consumption of these tasks, the battery life of edge devices and the range of electric vehicles can directly be influenced and extended.

Common approaches for reducing energy consumption include the development of new architectures or improved hardware optimization. Another promising research direction is the use of biologically inspired spiking neural networks (SNNs) which communicate with short pulses rather than continuous-valued activation functions. Compared to today's prevailing analog neural networks (ANNs), SNNs not only promise ultra-low powered neuromorphic hardware [1], but also provide superior computational power [2]. Despite these advantages, SNNs are not yet widely used in applications, as training remains challenging and results in inferior performance generally lacks behind that of ANNs trained with backpropagation. The three main approaches for training SNNs are (1) unsupervised learning approaches such as spike-timing-dependent plasticity [3], (2) supervised learning approaches that try to adapt gradient descent based backpropagation [4]–[6], and (3) conversion approaches, which is the focus of this work.

Training ANNs with backpropagation and mapping the weights to SNNs with the same architecture lead to optimal performance for spiking networks [7]. While earlier work has demonstrated the difficulties of translating the sigmoid activation function to spiking neurons [8], the dominant use of rectified linear units (ReLUs) [9] offers new possibilities. As both the activation of ReLUs and the firing rate of spiking neurons increase linearly with their input, this approach resulted in a near-lossless conversion [10]. Since then, this method has been well theorized and expanded for use with convolutional neural networks (CNNs) [11]. It enables the conversion of deep networks for object detection, including YOLO [12], ResNet [13], and RetinaNet [14].

However, for recurrent neural networks (RNNs) an approach with little to no conversion loss has been absent thus far. The conversion of an Elman network has resulted in an accuracy loss of 8.4% [15]. As this type of network is vulnerable to vanishing or exploding gradients, it is not commonly implemented in many of todays applications. More common recurrent architectures based on long short-term memory (LSTM) [16] or gated recurrent units (GRU) [17] are based on sigmoid and tanh activation functions and are also susceptible to the previously mentioned issues.

As RNNs based on LSTMs or GRUs are generally time consuming and computationally expensive to train, they are slowly being replaced by superior architectures in various applications. Most notably, transformer networks have revolutionized the field of natural language processing (NLP), as they are suited to model temporal sequences and achieve remarkable performance in machine translation [18]. Moreover, multi-head self-attention, which is a mechanism for performing queries on a sequence of information, has led to better performance compared to previous methods. Today, the best performing networks for solving NLP tasks, including BERT [19] and GPT-3 [20], are already variants of transformer networks. Furthermore, architectures using multi-head self-attention for image processing have shown excellent performance while also being less computationally expensive [21]. In addition, recent research has introduced DEtection TRansformer (DETR), which is an architecture for object detection which performs on par with the highly-optimized Faster RCNN algorithm [22].

978-1-6654-2624-4/21/$31.00 2021 IEEE

To leverage the use of transformers in SNNs and for energy efficient computing on neuromorphic hardware, we present a conversion method for transformer architectures to networks of spiking neurons. This approach is presented in section II and its performance subsequently evaluated in section III by running experiments for NLP and image classification.

II. Methodology

A. Architecture of Spiking Transformer Networks

The networks in our approach are similar to those in the original work for NLP [18] (see fig. 1) and image classification [21] (see fig. 2), respectively. The general architecture consists of a preprocessing step of the input depending on the data type, a subsequent transformer encoder, and various layers with classification of the output layer.

The transformer encoder is based on multi-head self-attention, which consists of multiple scaled-dot attention modules. The scaled-dot attention is computed by stacked layers of matrix multiplication, scaling, softmax, and another matrix multiplication layer. Following multi-head self-attention is a subsequent dense layer, ReLU activation, and another dense layer.

As the max pooling operation is difficult to realize in spiking networks, we instead implement an average pooling layers. For the classifying the output, a softmax layer is generally used. Different spiking operators have been presented to represent the same outcome [11]. For simplicity, we include an additional layer which accumulates all spikes of the output layer. A common softmax layer can then be used for evaluating accuracy.

B. Conversion Approach

Due to the similarity between ReLU and rate coded spiking neurons, conversion of a network can be achieved by replacing the ReLU activation function with simple integrate and fire (IF) neurons. This type of neuron offers a low computational complexity by integrating their input until it reaches a threshold. Next, it emits a spike that is transmitted to the connecting neurons and then resets before starting over again. The spike rate $r(t)$ is computed using the ReLU activation a of the initial ANN [11]:

$$r(t) = ar_{max} - \frac{V(t) - V(0)}{tV_{thr}} \quad (1)$$

where $V(t)$ is the membrane potential and V_{thr} is the firing threshold. Instead of resetting to 0, we use the *reset-by-subtraction* method [23], as it has demonstrated better efficiency for conversion.

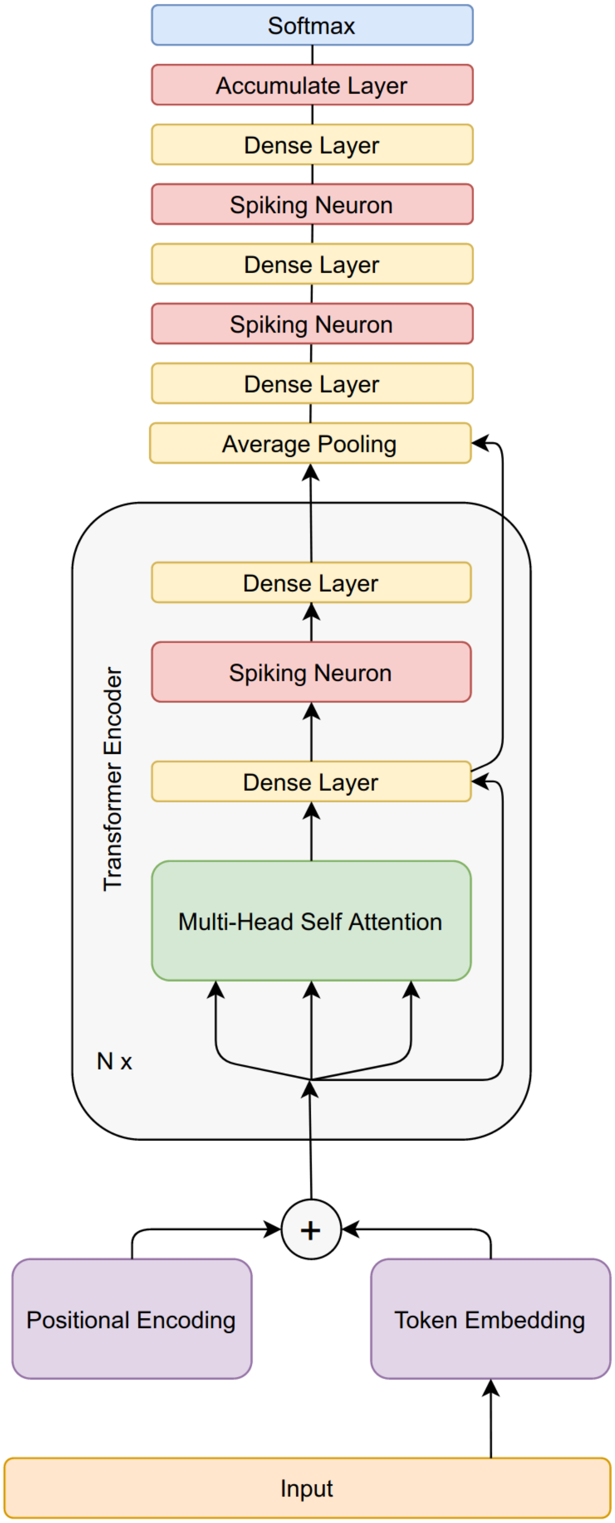

Fig. 1. Architecture of the converted spiking NLP transformer network. The ReLU activation of the trained ANN are replaced by spiking neuron Models. In contrast to the original work for NLP [18] we use average pooling instead of max pooling after the transformer encoder module.

In contrast to the ReLU activation function, spiking neurons have an upper limit as determined by their maximum firing rate r_{max} [10]. Thus, the weights of the original analog network have to be normalized to prevent extended inference time or even decreased overall accuracy. For this purpose, we take a subset of the training data and compute the activations for the layers that have to be converted. To discard extreme outliers, we implement the *robust normalization* algorithm [11], which only normalizes the p^{th} percentile of the activation. The authors suggest values for p in the range of [99.0, 99.999]; in our case we found $p = 99.0\%$ performed the best.

III. Experiments

For the evaluation of the presented conversion method, we train two types of transformer networks: an architecture similar to (1) the original paper for NLP sentiment classification [18] and (2) the proposed vision transformer architecture for image classification [21]. The networks are then converted to spiking networks using the methods mentioned earlier and subsequently evaluated in a simulation over 50 time steps.

A. Experiment 1: NLP Transformer

For the first experiment, we train an NLP transformer network on the IMDB movie review sentiment classification dataset [24]. The dataset consists of 25,000 movie reviews encoded as a list of word indices and labeled as positive/negative sentiment, and is split in half into training and test set. A vocabulary list is used for the 20,000 most frequently used words and limit the review length to 200 words. Shorter reviews are filled with zero padding. The labels are transformed to categorical vectors, which can have a positive or negative state.

The input data of the network consists of the reviews represented as sequences of encoded words. Embedding layers are used for the word sequences as well as for encoding positions of the words. Next, the token embeddings are added to the positional embeddings, and the result is fed into the transformer encoder. The data then passes an average pooling layer, two densely connected layers with ReLU activation and a succeeding softmax layer for classification.

B. Experiment 2: Vision Transformer

For the second experiment, we train a vision transformer network for digit classification of the MNIST dataset [25]. The dataset comprises of images of handwritten digits from zero to nine with according labels. The training set consists of 60,000 and the test set of 10,000 images, all with a resolution of 28×28 pixels.

For use in transformer networks, the images are divided into patches of 4×4 pixels, resulting in 7 patches per row and column. These 49 patches are then flattened and made into a linear projection of all patches. Positional encodings together with class embeddings are added to the linear projections of the patches. The subsequent transformer encoder uses the same architecture as described previously and is followed by a dense layer activated with ReLU and a softmax activated dense layer for classification.

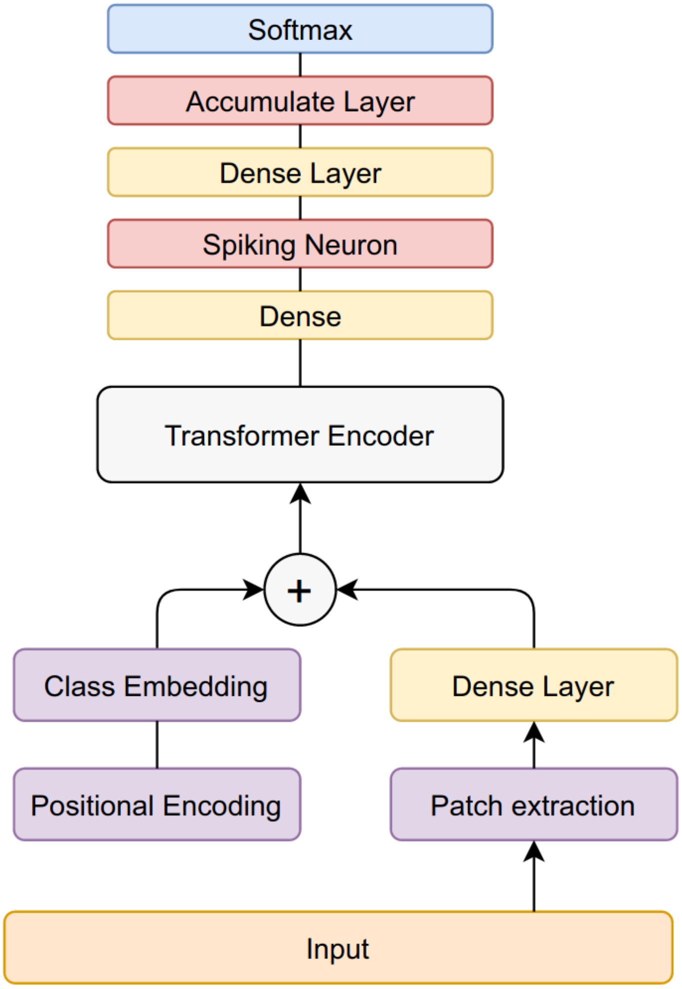

Fig. 2. Architecture of the converted spiking vision transformer network. The conversion approach and the transformer encoder are equal to those used in the spiking NLP transformer (see Fig. 1)

C. Training and Conversion

The training process of both the NLP transformer and vision transformer are conducted equally: the networks are trained for 2 epochs optimizd with Adam and a batch size of 64. To fit the maximum spike rate of the IF neurons, the resulting weights are subsequently scaled using the robust normalization strategy with the percentile set to $p = 99.0$. Lastly, the ReLU activation functions are replaced with the spiking neuron model.

The performance of the spiking transformer networks is then evaluated by calculating the accuracy of the test set in a simulation over 50 time steps. As the choice of the random seed can influence the accuracy of the resulting networks, each experiment is run 50 times and the results are averaged.

D. Results

The transformer for NLP achieved an averaged accuracy of 86.36% on the test set of the IMDB movie review senti-

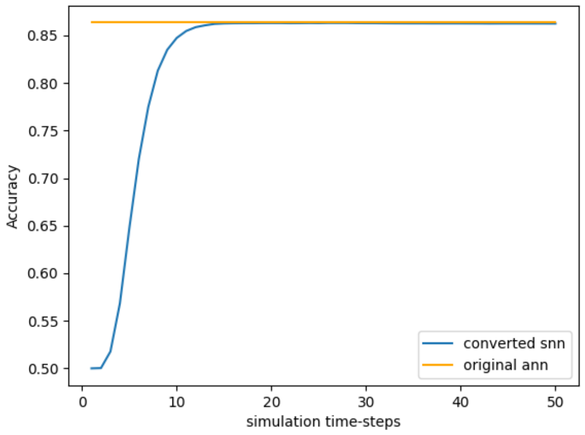

Fig. 3. Results of experiment 1: averaged accuracy of the converted spiking NLP transformer network over 50 time steps on the IMDB sentiment classification dataset

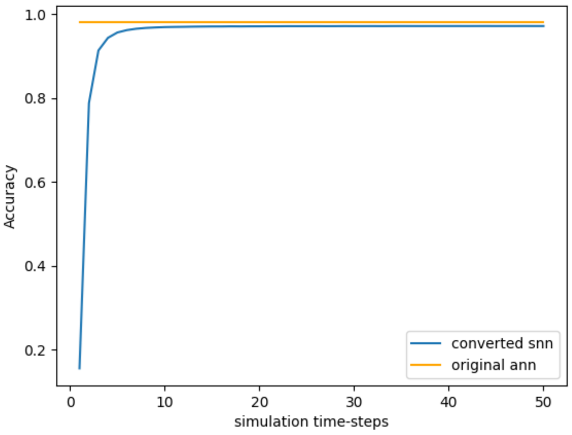

Fig. 4. Results of experiment 2: averaged accuracy of the converted spiking vision transformer network over 50 time steps on the MNIST image classification dataset

ment classification dataset. The converted spiking transformer achieved a similar accuracy of 86.25% after the simulation over 50 time steps. After only 13 time steps, the accuracy evened out at 86.1%, as can be seen in fig. 3. This yielded a conversion loss of only 0.11% from the original ANN to the resulting SNN.

The vision transformer in the second experiment achieved an averaged accuracy of 97.99% on the MNIST testing set. After conversion, the spiking network averaged an accuracy of 97.16% at the end of the simulation and also evened out after 13 time steps at 97.02%, as shown in Fig. 4. The loss of the vision transformer produces a higher conversion loss than in the first experiment, resulting in 0.83% loss.

The conversion of the vision transformer shows a larger loss in accuracy compared to the NLP transformer. As both converted networks share similar architectures and contain a comparable amount of spiking neurons, the source of error presumably originates in the more complex preprocessing of the input data in the vision transformer.

IV. Discussion

We introduced a method for converting conventional transformer architectures to networks of spiking neurons. The obtained architectures have only marginal conversion loss: 0.83% for the vision transformer and an even lower 0.11% for the NLP transformer. This approach enables training of conventional transformer networks with highly optimized gradient descent based learning methods and generating SNNs with nearly the same accuracy.

As previous research on conversion methods has mainly focused on computer vision, the application of sequential data has been sparse. However, with 86.1% accuracy on the IMDB dataset, our approach for NLP transformers performs far superior compared to 80.8% with a fully connected 5 layer spiking network in previous work [26]. The 97.16% accuracy on the MNIST dataset of our spiking vision transformer is close to better optimized conversion approaches of spiking CNNs, with a 2 layer architecture achieving 98.32% [27] and a ResNet8 reaching 99.59% [13].

Despite these promising results, there remains considerable potential for further improvement. As the transformer architectures had to be adapted prior to training, investigating approaches for spiking max pooling or softmax could lead to increased accuracy. As the first layers of the networks are not communicating with spikes, the preprocessing of the input data with spiking embedding layers could result in an end-to-end spiking network. Consequently, the application of neuromorphic hardware would enable energy efficient computation.

Overall, our approach shows significant potential for processing sequential data with SNNs. With further optimization, this method will be able to perform with the same accuracy while potentially leading to appreciable reductions in energy consumption.

Acknowledgment

The authors thank Infineon Technologies AG for supporting this research.

References

[1] C.-S. Poon and K. Zhou, “Neuromorphic Silicon Neurons and Large-Scale Neural Networks: Challenges and Opportunities,” *Frontiers in Neuroscience*, vol. 5, 2011.

[2] W. Maass, “Networks of spiking neurons: The third generation of neural network models,” *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[3] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann, “Regulation of Synaptic Efficacy by Coincidence of Postsynaptic APs and EPSPs,” *Science*, vol. 275, no. 5297, pp. 213–215, 1997.

[4] J.-P. Pfister, T. Toyoizumi, D. Barber, and W. Gerstner, “Optimal Spike-Timing-Dependent Plasticity for Precise Action Potential Firing in Supervised Learning,” *Neural Computation*, vol. 18, no. 6, pp. 1318–1348, 2006.

[5] E. Neftci, S. Das, B. Pedroni, K. Kreutz-Delgado, and G. Cauwenberghs, "Event-driven contrastive divergence for spiking neuromorphic systems," *Frontiers in Neuroscience*, vol. 7, 2014.

[6] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate Gradient Learning in Spiking Neural Networks," *arXiv:1901.09948 [cs, q-bio]*, 2019.

[7] A. Sengupta, Y. Ye, R. Wang, C. Liu, and K. Roy, "Going Deeper in Spiking Neural Networks: VGG and Residual Architectures," *Frontiers in Neuroscience*, vol. 13, 2019.

[8] Y. Cao, Y. Chen, and D. Khosla, "Spiking Deep Convolutional Neural Networks for Energy-Efficient Object Recognition," *International Journal of Computer Vision*, vol. 113, no. 1, pp. 54–66, 2014.

[9] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on Machine Learning*, 2010, p. 8.

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

[11] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of Continuous-Valued Deep Networks to Efficient Event-Driven Networks for Image Classification," *Frontiers in Neuroscience*, vol. 11, 2017.

[12] Y. Kim, "Convolutional Neural Networks for Sentence Classification," *arXiv:1408.5882 [cs]*, 2014.

[13] Y. Hu, H. Tang, Y. Wang, and G. Pan, "Spiking Deep Residual Network," *arXiv:1805.01352 [cs]*, 2018.

[14] E. Mueller, J. Hansjakob, D. Auge, and A. Knoll, "Minimizing Inference Time: Optimization Methods for Converted Deep Spiking Neural Networks," in *International Joint Conference on Neural Network*, 2021, p. 8.

[15] P. U. Diehl, G. Zarrella, A. Cassidy, B. U. Pedroni, and E. Neftci, "Conversion of artificial recurrent neural networks to spiking neural networks for low-power neuromorphic hardware," in *2016 IEEE International Conference on Rebooting Computing (ICRC)*, 2016, pp. 1–8.

[16] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv:1406.1078 [cs, stat]*, 2014.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *arXiv:1706.03762 [cs]*, 2017.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv:1810.04805 [cs]*, 2019.

[20] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," *arXiv:2005.14165 [cs]*, 2020.

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv:2010.11929 [cs]*, 2021.

[22] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," in *Computer Vision – ECCV 2020*, ser. Lecture Notes in Computer Science, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, pp. 213–229.

[23] B. Rueckauer, I.-A. Lungu, Y. Hu, and M. Pfeiffer, "Theory and Tools for the Conversion of Analog to Spiking Convolutional Neural Networks," *arXiv:1612.04052 [cs, stat]*, 2016.

[24] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning Word Vectors for Sentiment Analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 2011, p. 9.

[25] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[26] M. M. O. Alawad, H.-J. O. Yoon, and G. O. Tourassi, "Energy Efficient Stochastic-Based Deep Spiking Neural Networks for Sparse Datasets," Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States). Oak Ridge Leadership Computing Facility (OLCF), Tech. Rep., 2017.

[27] E. Stromatias, M. Soto, T. Serrano-Gotarredona, and B. Linares-Barranco, "An Event-Driven Classifier for Spiking Neural Networks Fed with Synthetic or Dynamic Vision Sensor Data," *Frontiers in Neuroscience*, vol. 11, 2017.