

Spiking Neural Unit

KAC KNUM

Wszystkie obrazki/wzory z arxiv.org/pdf/1812.07040.pdf

Spiking Neural Unit

KAC KNUM

DISCLAIMER

Plan na dziś

1. Kontekst
2. Pomysł
3. Wyniki

Deep Networks Incorporating Spiking Neural Dynamics

Stanisław Woźniak, Angeliki Pantazi,
and Evangelos Eleftheriou

IBM Research Labs

~ grudzień 2018

arxiv.org/pdf/1812.07040.pdf

Trochę tła: Spiking Neural Networks

- zupełnie inny framework sieci neuronowych
- wierniejsze odwzorowanie działania mózgu

Trochę tła: Spiking Neural Networks

- zupełnie inny framework sieci neuronowych
- wierniejsze odwzorowanie działania mózgu
- podstawowy model neuronu: Leaky Integrate-and-Fire (LIF)

$$\tau \frac{dV_m(t)}{dt} = -V_m(t) + RI(t)$$

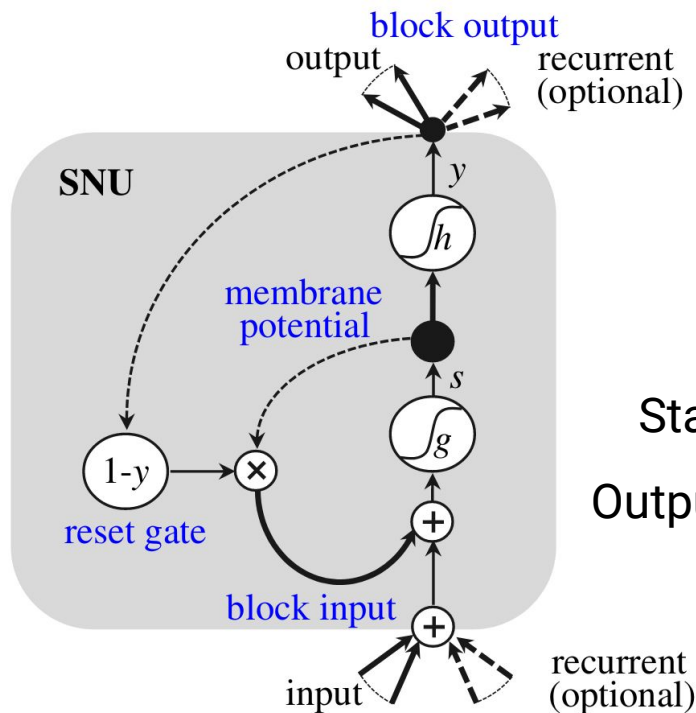
Trochę tła: Spiking Neural Networks

- zupełnie inny framework sieci neuronowych
- wierniejsze odwzorowanie działania mózgu
- podstawowy model neuronu: Leaky Integrate-and-Fire (LIF)

$$\tau \frac{dV_m(t)}{dt} = -V_m(t) + RI(t)$$

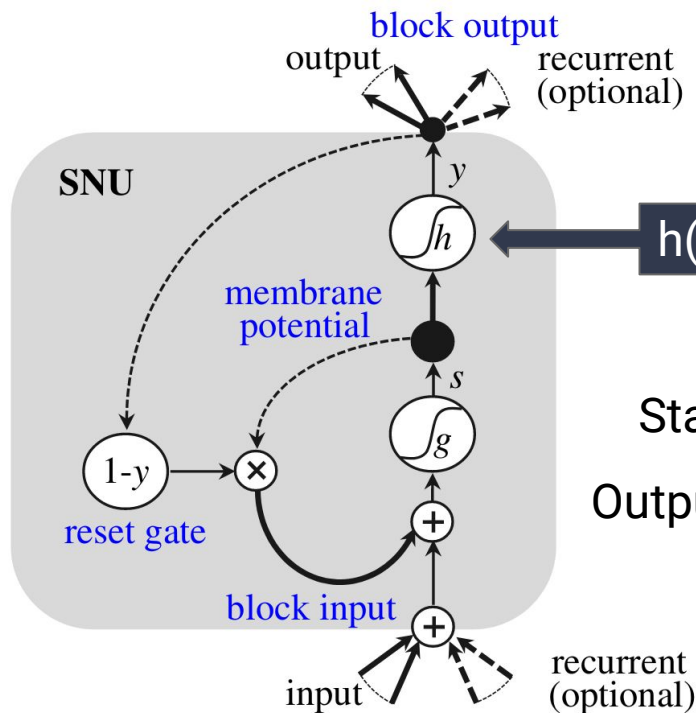
- brak skutecznych algorytmów uczenia

Propozycja: Spiking Neural Unit



$$\text{Stan: } s_t = g(Wx_t + l(\tau) \odot s_{t-1} \odot (1 - y_{t-1}))$$
$$\text{Output: } y_t = h(s_t + b)$$

Propozycja: Spiking Neural Unit

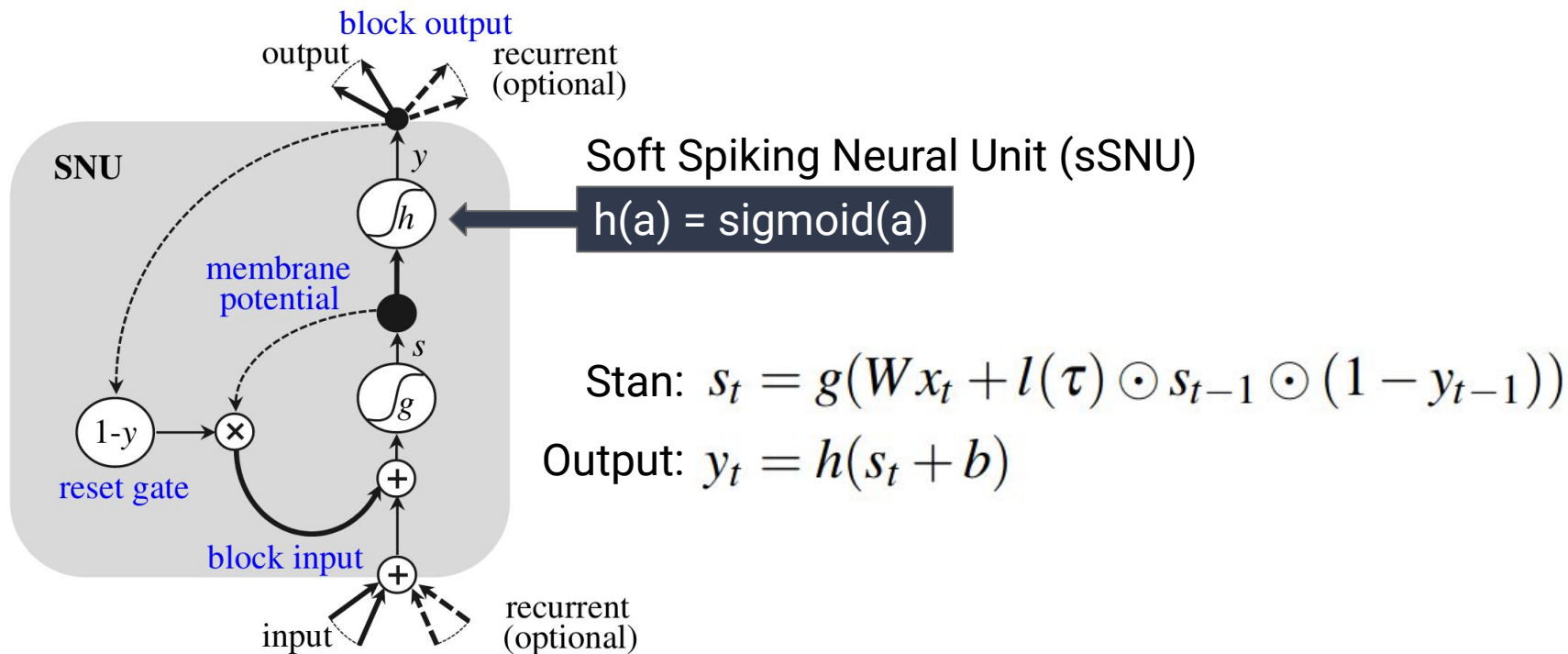


$$h(a) = 1 \text{ if } a > 0 \text{ else } 0$$

$$\text{Stan: } s_t = g(Wx_t + l(\tau) \odot s_{t-1} \odot (1 - y_{t-1}))$$

$$\text{Output: } y_t = h(s_t + b)$$

Propozycja: soft Spiking Neural Unit



Funkcja straty

$$L = \text{MSE}(\bar{y}_{(a,b)}, \hat{y})$$

gdzie (a, b) to przedział czasu, podczas którego oceniamy wynik

czy to
działa?

Benchmark I: Jittered MNIST

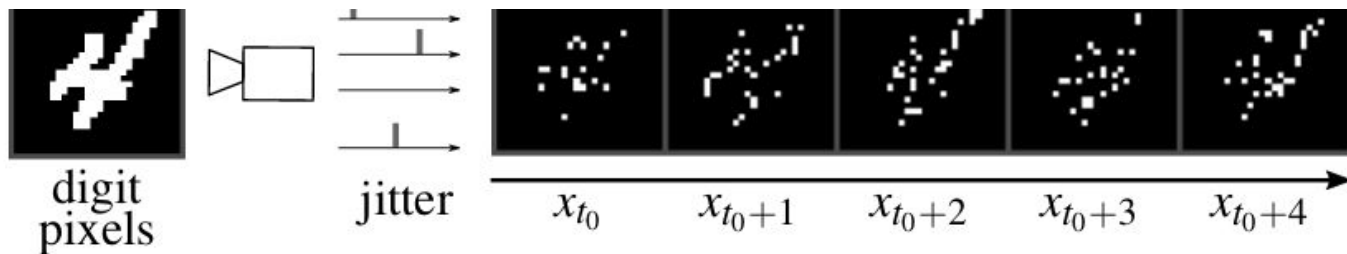
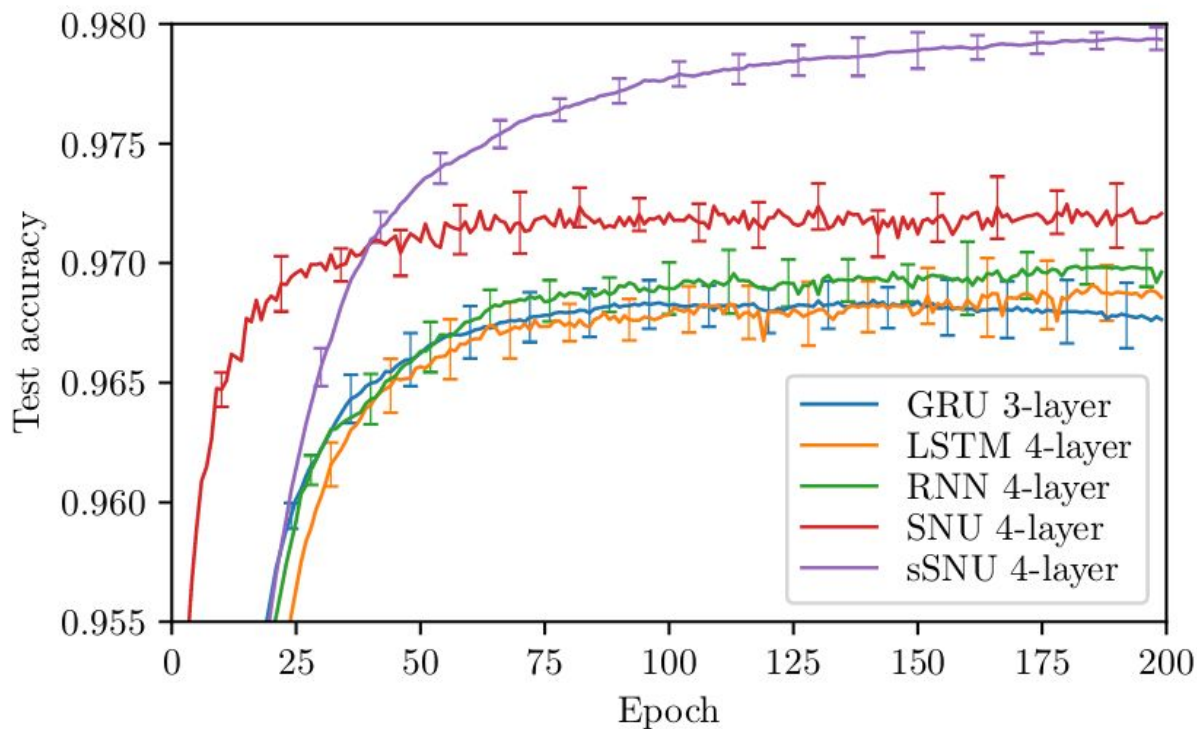


FIG. 5. **Jittered MNIST:** The input spikes arrive from an asynchronous camera, conveying jittered images of the digits.

Benchmark I: Jittered MNIST



Benchmark I: Jittered MNIST

TABLE III. **Detailed comparison for the jittered MNIST:** the number of model parameters and the test accuracy obtained from 10 trials.

Network	Total # of parameters	Mean accuracy	Maximum accuracy
GRU 3-layer	658 880	0.9694	0.9708
LSTM 4-layer	1 208 460	0.9699	0.9719
RNN 4-layer	400 494	0.9708	0.9718
SNU 4-layer	269 322	0.9741	0.9754
sSNU 4-layer	269 322	0.9796	0.9802

Benchmark II: polyphonic music prediction

Network	# of hidden units	# of hidden layer param.	Total # of parameters
RNN tanh ⁵⁸	100	$\approx 20.1 \times 10^3$	$\approx 30.2 \times 10^3$
GRU ⁵⁸	46	$\approx 20.2 \times 10^3$	$\approx 25.0 \times 10^3$
LSTM ⁵⁸	36	$\approx 19.8 \times 10^3$	$\approx 23.4 \times 10^3$
LSTM ⁶¹	≈ 87	$\approx 61.2 \times 10^3$	$\approx 69.0 \times 10^3$
NIAF-LSTM ⁶¹	≈ 110	$\approx 87.6 \times 10^3$	$\approx 97.3 \times 10^3$
SNU	150	13.350×10^3	26.638×10^3
sSNU	150	13.350×10^3	26.638×10^3

Podsumowując

- Spiking Neural Unit wnosi dynamikę Spiking Neural Networks do znanych nam Artificial Neural Networks
- SNU ma mniej parametrów i wygląda na to, że szybciej się uczy (w porównaniu z LSTMami i GRU)
- istnieją benchmarki, na których SNU osiągają wyniki lepsze niż LSTMy i GRU

dzięki.
pytania?