ISCAS 2020
Virtual, October 10-21
IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS
CAS

File Classification Based on Spiking Neural Networks

Ana Stanojevic, Giovanni Cherubini, Timoleon Moraitis and Abu Sebastian

IBM Research - Zurich, Rüschlikon, Switzerland

2020 IEEE International Symposium on Circuits and Systems
Virtual, October 10-21, 2020

"The global data sphere will grow from 33 zettabytes in 2018 to 175 zettabytes in 2025. Almost 30 percent of the world's data will have to be processed in real time." [8]

Table of contents

Introduction

Challenges posed by data:

- Data storage
- Data management
- Data analytics

Task: Efficient data classification system

Data

- Files of various types
- Described with metadata (set of key-value pairs)

Model

- Artificial neural networks (ANNs) require high-precision arithmetic and are in general energy inefficient
- Spiking neural networks (SNNs) are biologically inspired
- SNNs rely on sequences of spikes (ones and zeros) to propagate information [6][7]
- SNNs are well suited to process sparse and asynchronous data

Data

- Files of various types
- Described with metadata (set of key-value pairs)

Model

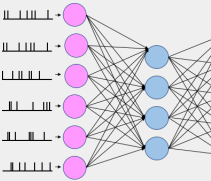
- Artificial neural networks (ANNs) require high-precision arithmetic and are in general energy inefficient
- Spiking neural networks (SNNs) are biologically inspired
- SNNs rely on sequences of spikes (ones and zeros) to propagate information [6][7]
- SNNs are well suited to process sparse and asynchronous data

Data

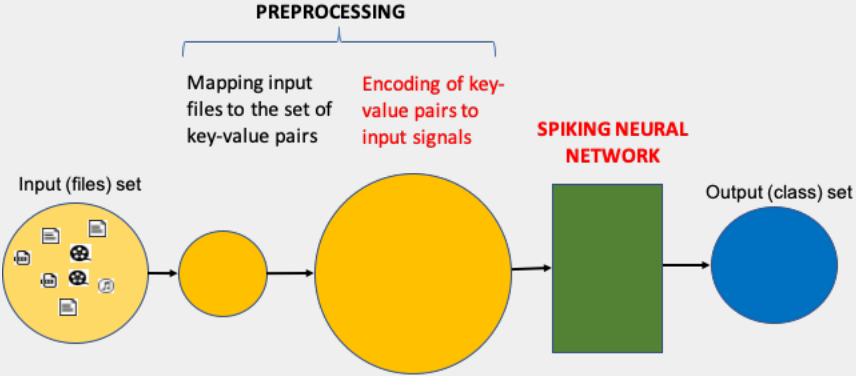
- Files of various types
- Described with metadata (set of key-value pairs)

Model

- Artificial neural networks (ANNs) require high-precision arithmetic and are in general energy inefficient
- Spiking neural networks (SNNs) are biologically inspired
- SNNs rely on sequences of spikes (ones and zeros) to propagate information [6][7]
- SNNs are well suited to process sparse and asynchronous data



System Architecture


PREPROCESSING
Mapping input files to the set of key-value pairs
Encoding of key-value pairs to input signals
SPIKING NEURAL NETWORK
Input (files) set
Output (class) set

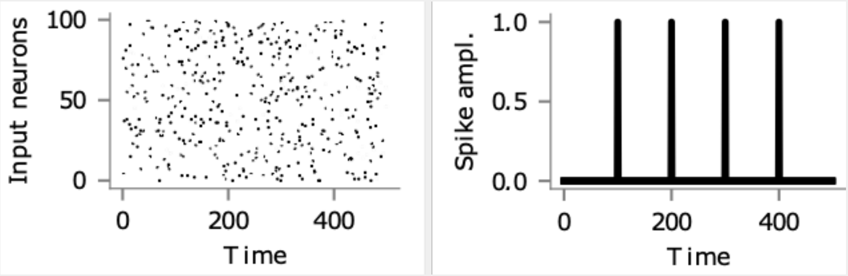


Figure: Input spike pattern (left). Zooming on spiking of one input neuron (right).

At the output of the SNN we generate output spike patterns, and we want to induce spikes at precise spike instants (target spike pattern)

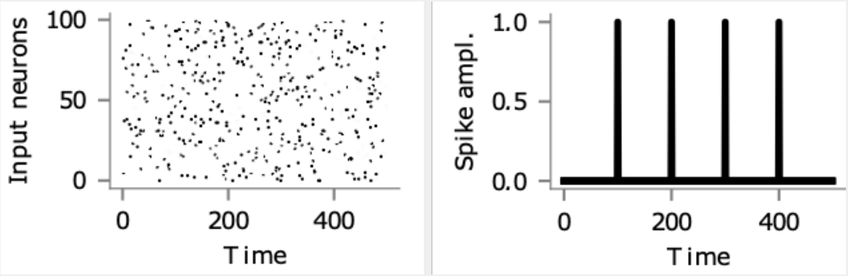


Figure: Input spike pattern (left). Zooming on spiking of one input neuron (right).

At the output of the SNN we generate output spike patterns, and we want to induce spikes at precise spike instants (target spike pattern)

Correlative Time Encoding (CTE)

CTE: Translate *similarities* among the input files into the *correlations* of input spike patterns

- each key *k* is associated with a set of input neurons of SNN
- neurons associated with key *k* generate spikes at pseudo-random time instants, depending on the value the key assumes for the given input file
- neurons are fixed for a given key *k* and pseudo-random spike patterns are fixed for a given key-value pair

CTE: Translate *similarities* among the input files into the *correlations* of input spike patterns

- each key k is associated with a set of input neurons of SNN
- neurons associated with key k generate spikes at pseudo-random time instants, depending on the value the key assumes for the given input file
- neurons are fixed for a given key k and pseudo-random spike patterns are fixed for a given key-value pair

Correlative Time Encoding (CTE)

CTE: Translate *similarities* among the input files into the *correlations* of input spike patterns

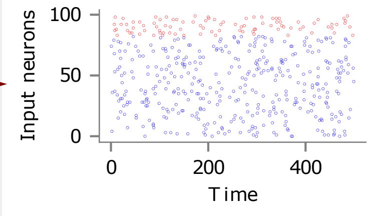
- each key *k* is associated with a set of input neurons of SNN
- neurons associated with key *k* generate spikes at pseudo-random time instants, depending on the value the key assumes for the given input file
- neurons are fixed for a given key *k* and pseudo-random spike patterns are fixed for a given key-value pair

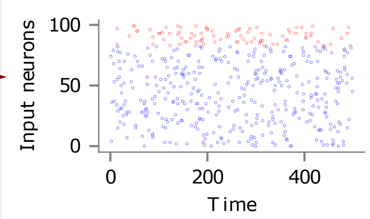
Correlative Time Encoding (CTE)

CTE: Translate *similarities* among the input files into the *correlations* of input spike patterns

- each key k is associated with a set of input neurons of SNN
- neurons associated with key k generate spikes at pseudo-random time instants, depending on the value the key assumes for the given input file
- neurons are fixed for a given key k and pseudo-random spike patterns are fixed for a given key-value pair

Correlative Time Encoding (CTE)





Spiking Neural Network - Learning

Devising an effective training algorithm is one of the biggest challenges that SNNs pose

Spiking Neural Network - Learning

STDP [10]

spike-timing-dependent plasticity

- Local learning rule
- Believed to underlie learning in the brain
- Weights reflect tight temporal correlations between the spikes of pre- and postsynaptic neuron

BPTT probabilistic [5]

backpropagation through time

- Two layers
- Probabilistic neuron model, maximize log-likelihood to induce spiking at target spike patterns
- Weight updates using derivatives
- Requires approximations

BPTT deterministic [13]

backpropagation through time

- Arbitrary number of layers
- Training using ANN frameworks
- Minimizes weighted cross entropy loss

Spiking Neural Network - Learning

STDP [10]

spike-timing-dependent plasticity

- Local learning rule
- Believed to underlie learning in the brain
- Weights reflect tight temporal correlations between the spikes of pre- and postsynaptic neuron

BPTT probabilistic [5]

backpropagation through time

- Two layers
- Probabilistic neuron model, maximize log-likelihood to induce spiking at target spike patterns
- Weight updates using derivatives
- Requires approximations

BPTT deterministic [13]

backpropagation through time

- Arbitrary number of layers
- Training using ANN frameworks
- Minimizes weighted cross entropy loss

Spiking Neural Network - Learning

STDP [10]

spike-timing-dependent plasticity

- Local learning rule
- Believed to underlie learning in the brain
- Weights reflect tight temporal correlations between the spikes of pre- and postsynaptic neuron

BPTT probabilistic [5]

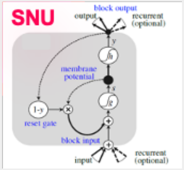
backpropagation through time

- Two layers
- Probabilistic neuron model, maximize log-likelihood to induce spiking at target spike patterns
- Weight updates using derivatives
- Requires approximations

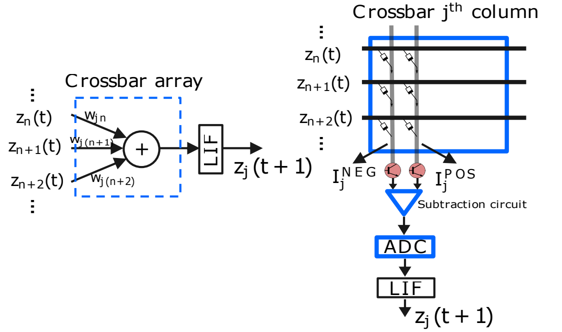
BPTT deterministic [13]

backpropagation through time

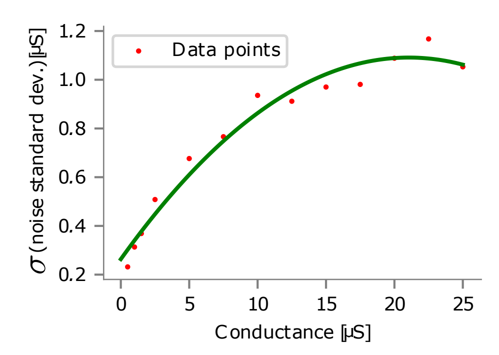
- Arbitrary number of layers



- Training using ANN frameworks
- Minimizes weighted cross entropy loss



Memristive devices [12][4] when organized in a crossbar array can be used to realize the neuronal connectivity in different layers of the SNNs [9][3]



Phase change memory (PCM) [2] devices are easy to program and have low-power consumption [1][11], however, they exibit non negligible conductance variations

Simulation Results

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Simulation Results for Different Data Sets.

Model	Adult Income	Nursery School	Car Evaluation	Connect Four
Log. Reg.	85.2%	93%	86.3%	75.6%
SVM (RBF)	85.3%	98.8%	92.2%	77.7%
STDP unsupervised rate encoded	77.5%	67.2%	50.5%	64.9%
STDP unsupervised	77.8%	70.3%	59.6%	65.8%
STDP supervised	75.6%	73.7%	70.2%	65.8%
Probab. BackProp. one layer	72.9%	73.9%	65.5%	59.8%
Probab. BackProp. two layers	79.2%	88.4%	87.7%	64.5%
Determ. BackProp. two layers	85.4%	99.6%	96.7%	74.4%
Determ. BackProp. five layers	85.7%	99.3%	97%	73.6%

Table: Inference Results with Hardware Simulator.

Model	Adult Income	Nursery School	Car Eval.	Connect Four
Software	85.4%	99.6%	96.7%	74.4%
HW (st. dev. = σ)	85.1%	99.4%	97%	73.5%
HW (st. dev. = 5 * σ)	81.6%	93.8%	93.2%	69.6%
HW (st. dev. = 10 * σ)	70.6%	73.1%	80.9%	44.4%

Table: Inference Results with Hardware Simulator.

Model	Adult Income	Nursery School	Car Eval.	Connect Four
Software	85.4%	99.6%	96.7%	74.4%
HW (st. dev. = σ)	85.1%	99.4%	97%	73.5%
HW (st. dev. = 5 * σ)	81.6%	93.8%	93.2%	69.6%
HW (st. dev. = 10 * σ)	70.6%	73.1%	80.9%	44.4%

Conclusion

SNN-based system for energy-efficient file classification

Novel CTE scheme to encode input files using pseudo-random spike patterns

The hardware implementation of the system using memristive arrays does not significantly affect performance

Real time classification of asynchronously injested files appears to be a promising application for SNNs, especially if implemented efficiently using memristive hardware

SNN-based system for energy-efficient file classification

Novel CTE scheme to encode input files using pseudo-random spike patterns

The hardware implementation of the system using memristive arrays does not significantly affect performance

Real time classification of asynchronously injested files appears to be a promising application for SNNs, especially if implemented efficiently using memristive hardware

SNN-based system for energy-efficient file classification

Novel CTE scheme to encode input files using pseudo-random spike patterns

The hardware implementation of the system using memristive arrays does not significantly affect performance

Real time classification of asynchronously injested files appears to be a promising application for SNNs, especially if implemented efficiently using memristive hardware

SNN-based system for energy-efficient file classification

Novel CTE scheme to encode input files using pseudo-random spike patterns

The hardware implementation of the system using memristive arrays does not significantly affect performance

Real time classification of asynchronously injested files appears to be a promising application for SNNs, especially if implemented efficiently using memristive hardware

References I

Irem Boybat, Manuel Le Gallo, SR Nandakumar, Timoleon Moraitis, Thomas Parnell, Tomas Tuma, Bipin Rajendran, Yusuf Leblebici, Abu Sebastian, and Evangelos Eleftheriou.
Neuromorphic computing with multi-memristive synapses.
Nature communications, 9(1):2514, 2018.

Geoffrey W Burr, Matthew J Brightsky, Abu Sebastian, Huai-Yu Cheng, Jau-Yi Wu, Sangbum Kim, Norma E Sosa, Nikolaos Papandreou, Hsiang-Lan Lung, Haralampos Pozidis, et al.
Recent progress in phase-change memory technology.
IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 6(2):146–162, 2016.

Geoffrey W. Burr, Robert M. Shelby, Abu Sebastian, Sangbum Kim, Seyoung Kim, Severin Sidler, Kumar Virwani, Masatoshi Ishii, Pritish Narayanan, Alessandro Fumarola, Lucas L. Sanches, Irem Boybat, Manuel Le Gallo, Kibong Moon, Jiyoo Woo, Hyunsang Hwang, and Yusuf Leblebici.
Neuromorphic computing using non-volatile memory.
Adv. Phys.-X, 2:89–124, 2017.

Leon Chua.
Resistance switching memories are memristors.
Applied Physics A, 102(4):765–783, 2011.

Brian Gardner, Ioana Sporea, and André Grüning.
Encoding spike patterns in multilayer spiking neural networks.
arXiv:1503.09129 [cs], 2015.

Samanwoy Ghosh-Dastidar and Hojjat Adeli.
Third generation neural networks: Spiking neural networks.
In Wen Yu and Edgar N. Sanchez, editors, *Advances in Computational Intelligence*, pages 167–178, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

References II

Filip Ponulak and Andrzej Kasiński.
Introduction to spiking neural networks: Information processing, learning and applications.
Acta Neurobiol. Exp., 71:409–33, January 2011.

David Reinsel, John Gantz, and John Rydning.
The digitization of the world from edge to core.
2018.

Sylvain Saïghi, Christian Mayr, Teresa Serrano-Gotarredona, Heidemarie Schmidt, Gwendal Lecerf, Jean Tomas, Julie Grollier, Sören Boyn, Adrien Vincent, Damien Querlioz, Selina La Barbera, Fabien Alibart, Dominique Vuillaume, Olivier Bichler, Christian Gamrat, and Bernabé Linares-Barranco.
Plasticity in memristive devices for spiking neural networks.
Front. Neurosci., 9, March 2015.

J. Sjöström and W. Gerstner.
Spike-timing dependent plasticity.
Scholarpedia, 5:1362, 2010.

Tomas Tuma, Manuel Le Gallo, Abu Sebastian, and Evangelos Eleftheriou.
Detecting correlations using phase-change neurons and synapses.
IEEE Electron Device Letters, 37(9):1238–1241, 2016.

H-S Philip Wong and Sayeef Salahuddin.
Memory leads the way to better computing.
Nature nanotechnology, 10(3):191, 2015.

Stanisław Woźniak, Angeliki Pantazi, Thomas Bohnstingl, and Evangelos Eleftheriou.
Deep learning incorporating biologically-inspired neural dynamics.
arXiv:1812.07040 [cs], 2018.