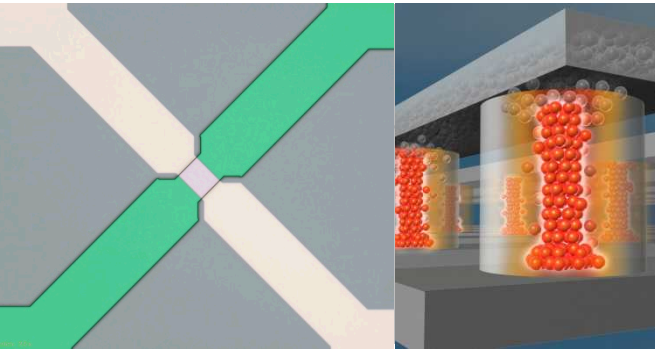
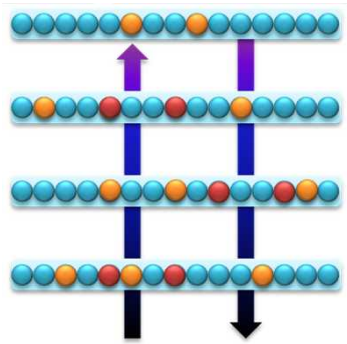
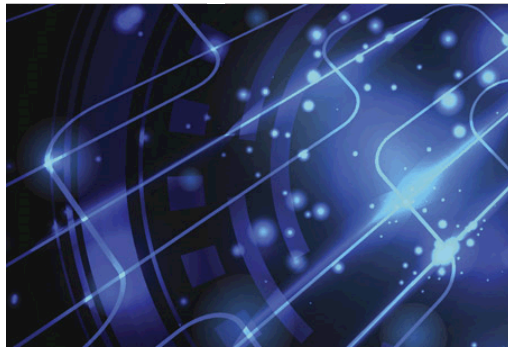
SAND2016-10966C

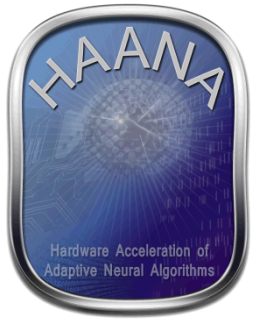
Exceptional service in the national interest

Sandia National Laboratories









Neurologically Inspired Architectures and Algorithms for Stream Processing

John Naegle
Sandia National Laboratories

Chesapeake Large Scale Analytics Conference, October 25-27, 2016

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration | LDRD LABORATORY DIRECTED RESEARCH & DEVELOPMENT

Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. XXXXXXXXXX

Acknowledgments

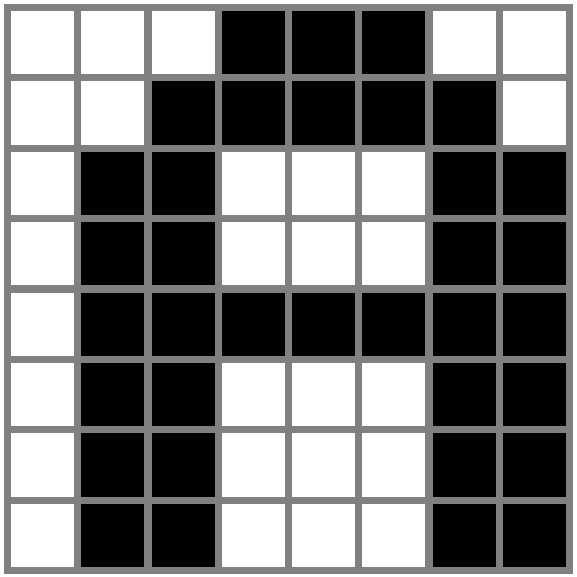
Algorithms: Brad Aimone, Ojas Parekh, Nadine Miner, Sandra Faust, Steve Verzi, Fred Rothganger, Frances Chance, Tu-Thach Quach, Chris Lamb, Sam Mulder, William Severa, Kris Carlson, Michael Smith, Cynthia Phillips
Architecture: John Naegle, Alex Hsia, Craig Vineyard, John Donaldson, Aaron Hill
Hardware: Matt Marinella, Tom Beechem, Ron Goeke, Alec Talin, Paul Kotula, Farid El Gabaly, Elliot Fuller, Jim Stevens, Sapan Agarwal, David Hughart, Andy Armstrong, David Henry, Gaadi Haase, Steve Wolfley, Derek Wilke, Michael Van Heukelom, Michael Thomas, Anthony McDonald, Carl Smith
Modeling: Steve Plimpton, Richard Schiek, Brian Tierney, Robert Bondi, Fred Rothganger
Application areas: Tim Draelos, Justin Doak, Jonathan Cox, Joe Ingram, Jason Wheeler

Partnerships:
David Follett, Duncan Townsend – Lewis Rhodes Labs
Isaac Richter - U. Rochester; Felix Wang – UIUC; Marek Osinski – UNM;

UNIVERSITY of ROCHESTER

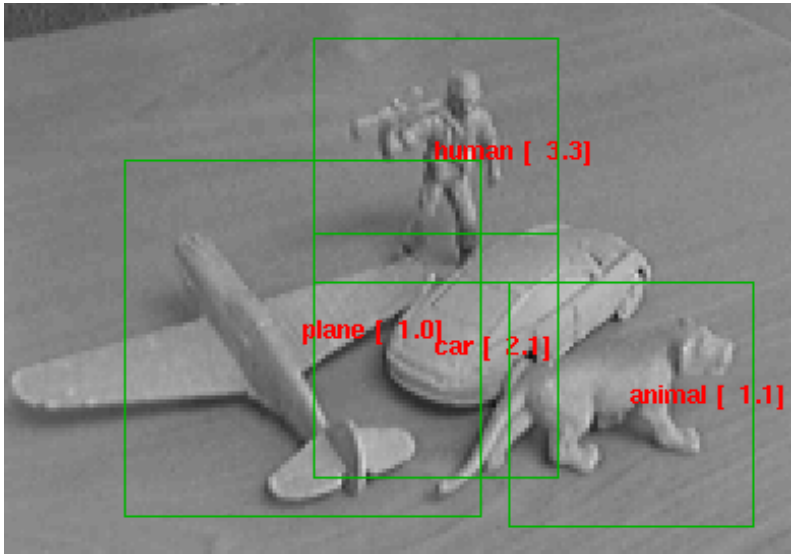
Neural-inspired (data-driven) computing is necessary for real-world problems

Conventional numerical computing



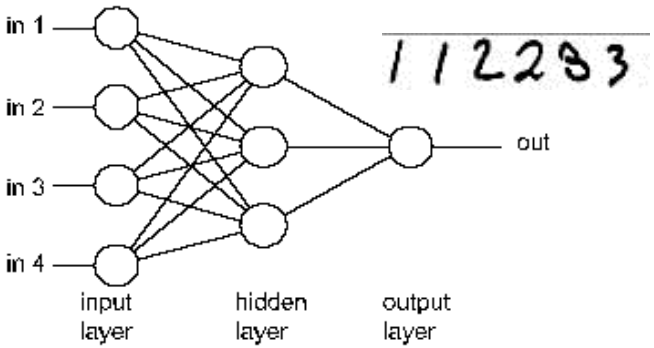
C. Lampert, VRML 2013

Data-driven computing

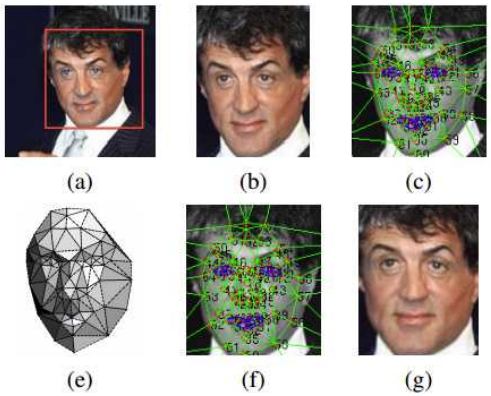


yann.lecun.com

Neural-inspired algorithms have matured & are achieving success on recognition problems

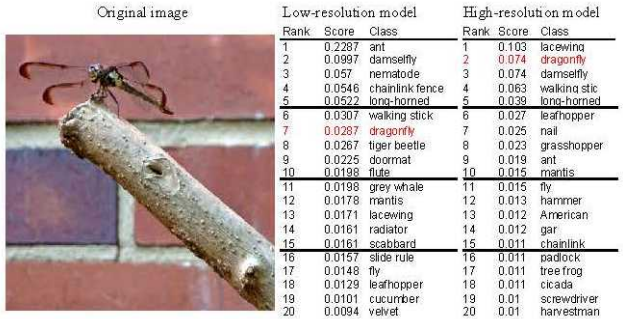


http://www.cheshireeng.com/Neuralyst/nnbg.htm

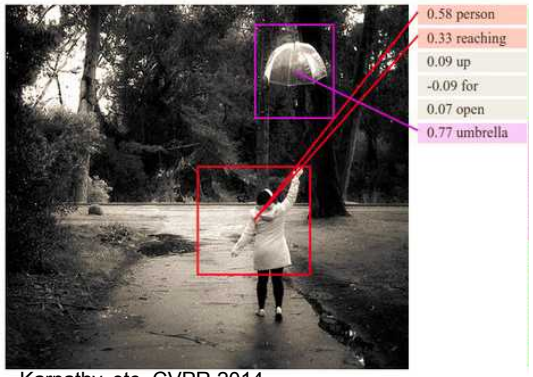


DeepFace (Facebook) - Taigman et al. CVPR 2014

FaceNet (Google) - Schroff et al. CVPR 2015



DeepImage (Baidu) - Wu et al. 2015

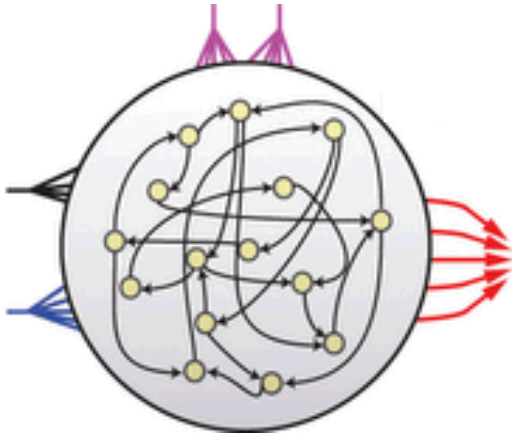


Karpathy etc. CVPR 2014

Modern machine learning only captures limited characteristics of neural computing

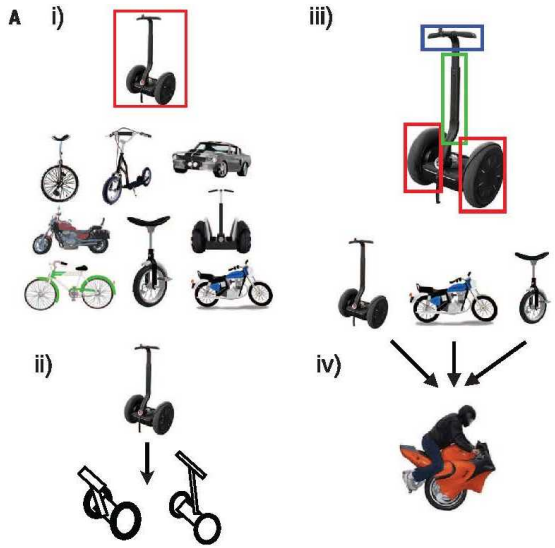
Temporal feature extraction via LSMs (Mante, Sussilo, Shenoy, Newsome, Nature 2013).

M. Smith talk: neural-inspired architecture for LSMs



Mante et al., Nature 2013, 503, 78

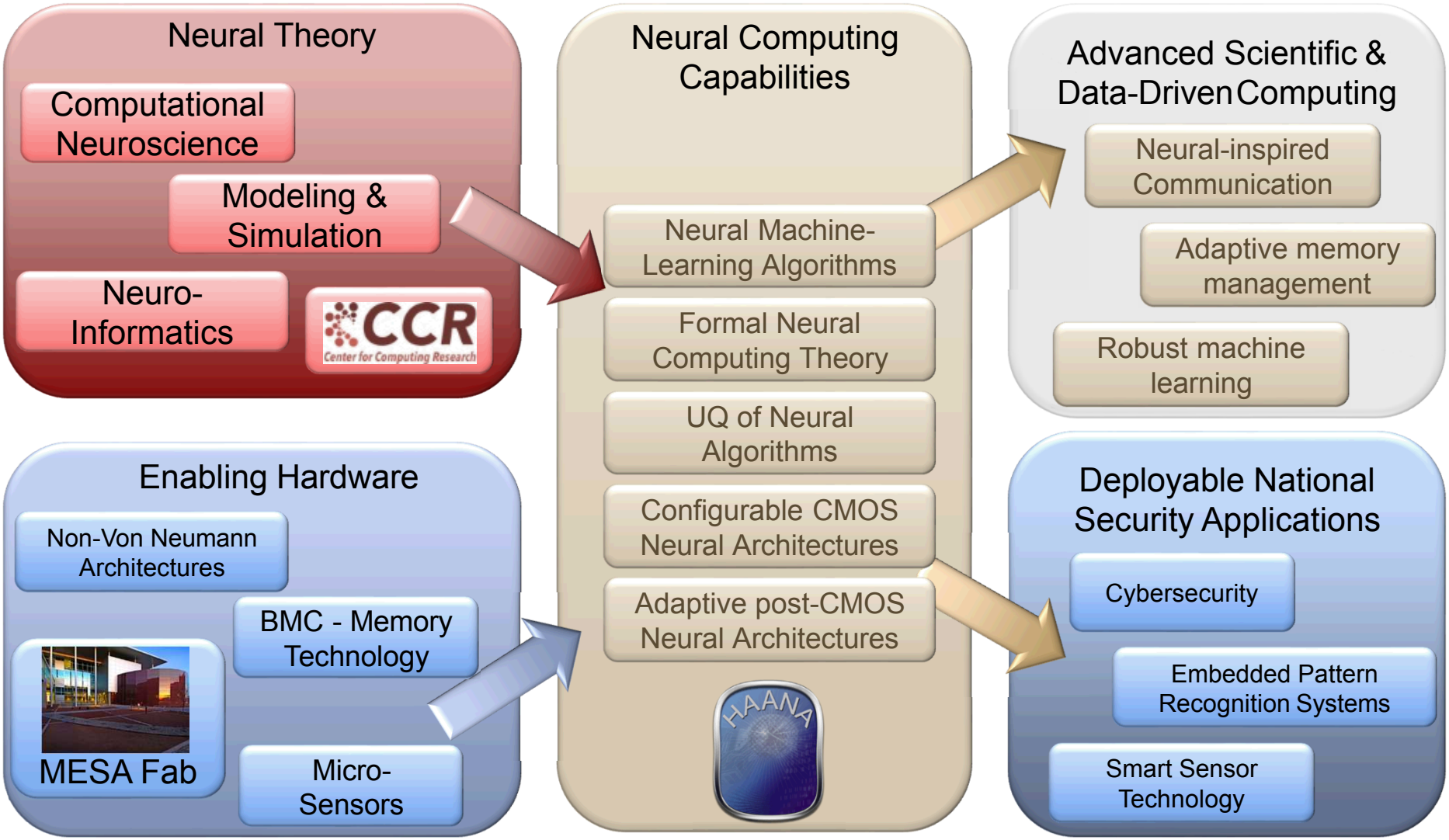
One-shot learning & generation of new concepts (Lake, Salakhutdinov, Tenenbaum, Science 2015).



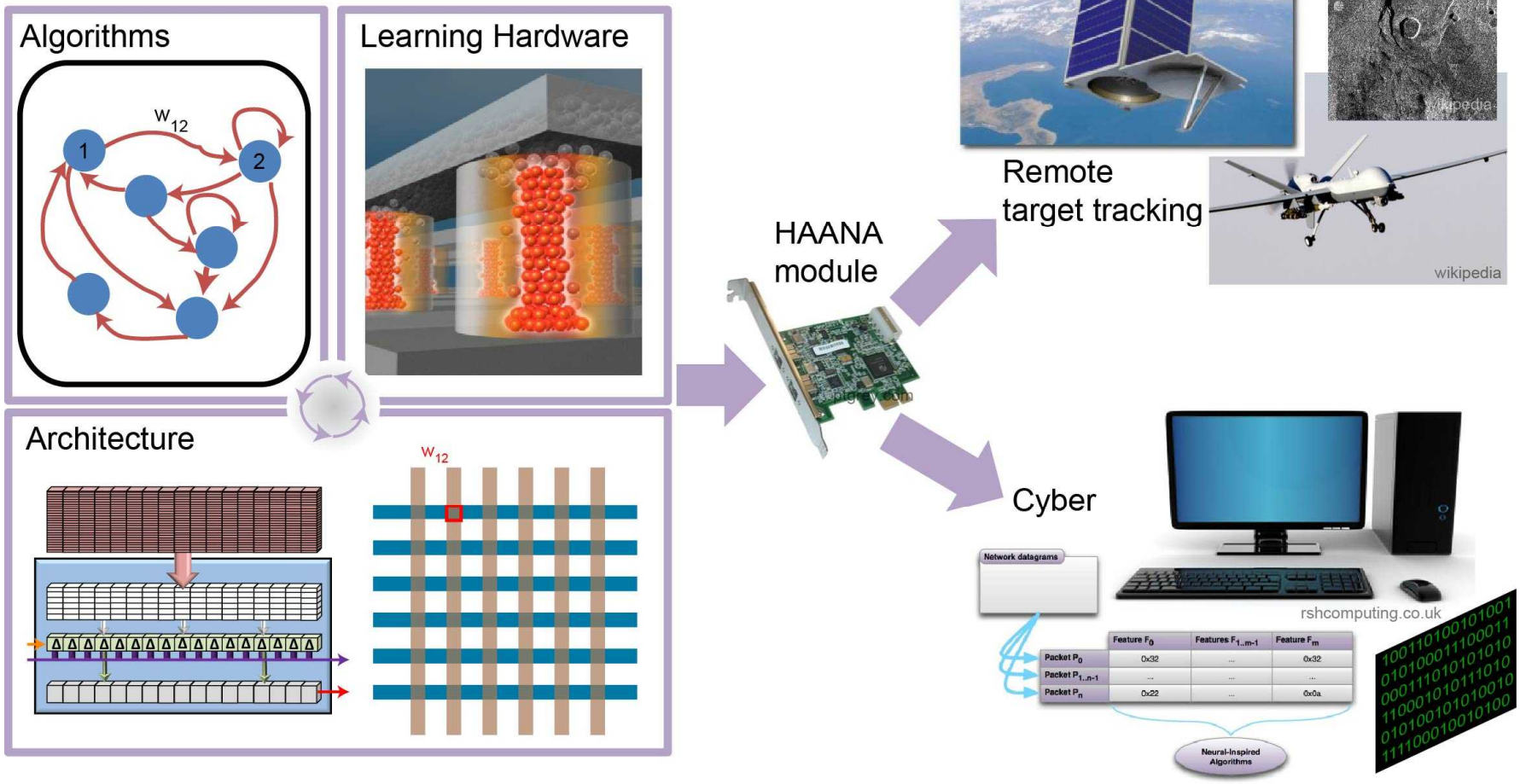
Lake et al., Science 2015, 350, 1332

Strengthen the connection between ML & neuroscience = neural machine learning

Neural computing at Sandia Labs leverages a large research foundation



Hardware Acceleration of Adaptive Neural Algorithms (HAANA)



End of Moore's Law- Motivation for Neuromorphic Architectures

- Post Moore's Law effects
  - Processing limited by power not complexity
  - We can build it but we can't cool it
- Operational cost > capital costs
  - Power, People & Space
- However, economics still driving change
  - Virtualization: Improved server utilization
  - Cloud: Reduced user complexity & cost
- Cloud/Virtualization approaching maturity

Changing Nature of Computing

- Cultural changes altering use model
  - Ubiquitous connectivity
  - Social media
  - Data explosion
  - Internet of things

- Time value of data increasing rapidly
  - Streaming Analytics fundamental to
    - Financial industry: Banking, Wallstreet, fraud detection
    - Social media: Google, Facebook
    - Intelligence community: NSA, CIA, FBI
    - etc.

von Neumann Architecture

- Strengths
  - Ubiquitous & bullet proof
  - Memory model offers maximum flexibility

- Weaknesses
  - Performance is power limited
  - Primary culprit: Memory model
  - Typically sub-optimal for high repetition tasks
    - Floating point, ex. Nvidia
    - Network Protocols

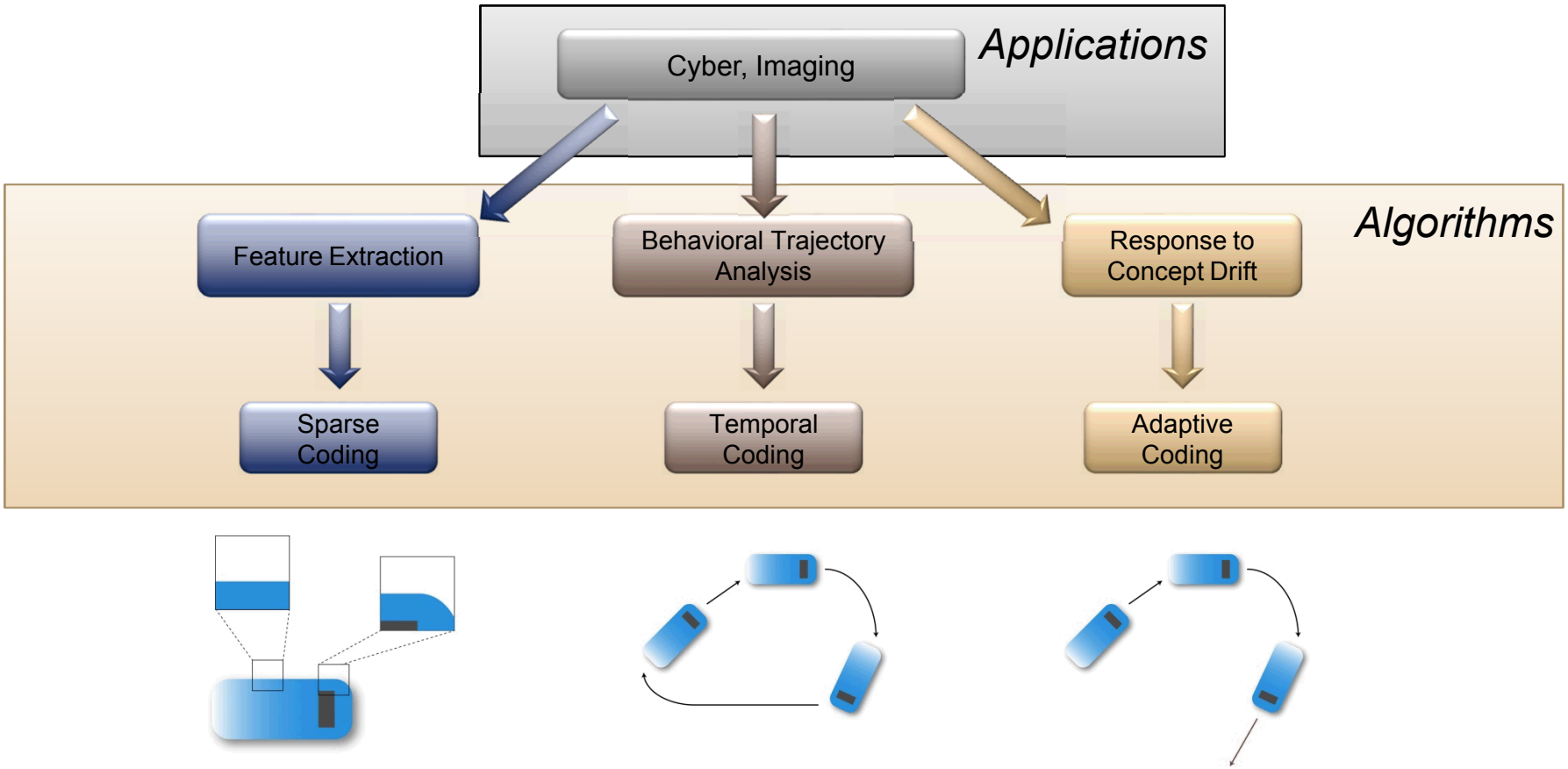
Neuromorphic Computing

- Strength, ideal for streaming analytics
  - Extraordinary at matching patterns
    - FPGA >1,000x ops/watt vs IA server demonstrated
    - ASIC >1,000,000x ops/watt projected
  - Not power or routing limited

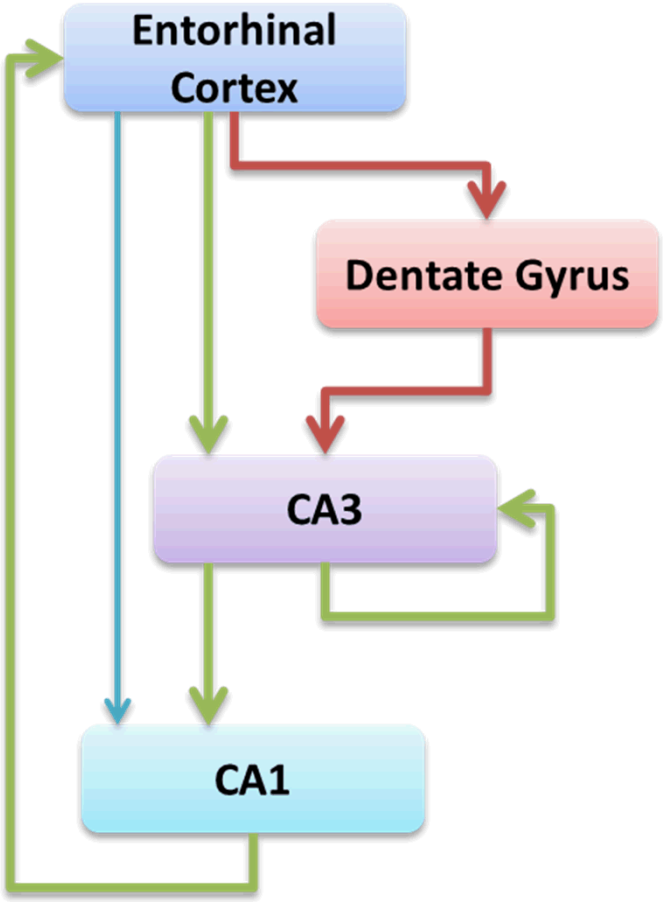
- Weaknesses
  - Not Turing complete in traditional sense
  - Not general purpose

- Architecturally processor or memory?
  - Neuromorphic computers blur the distinction

HAANA relies on applications to drive S&T for algorithms, architecture, and hardware

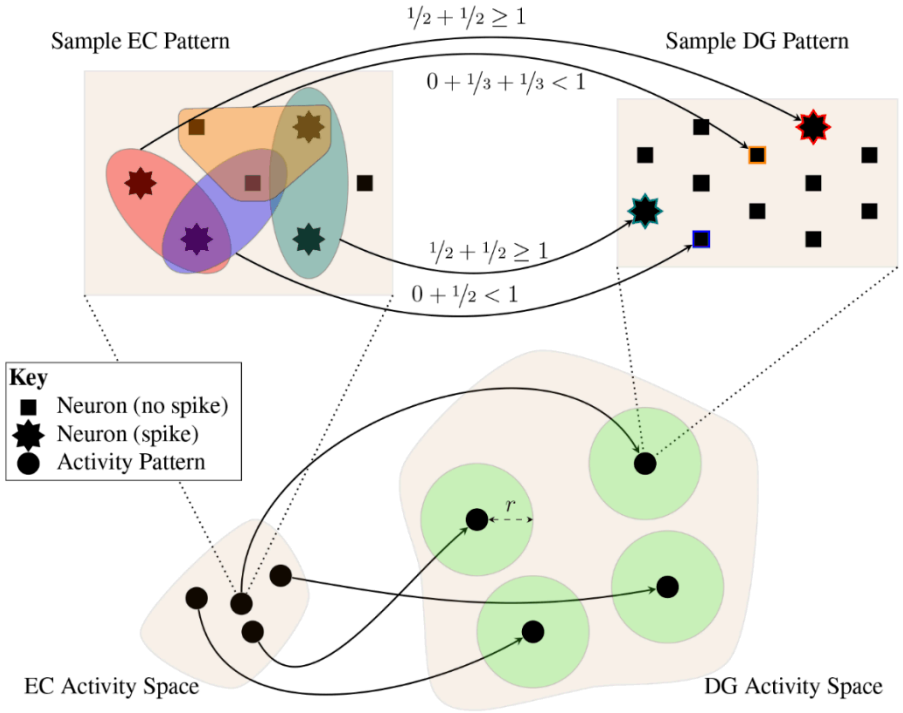


Neural-inspired algorithms: dentate gyrus pre-processing formats data for learning

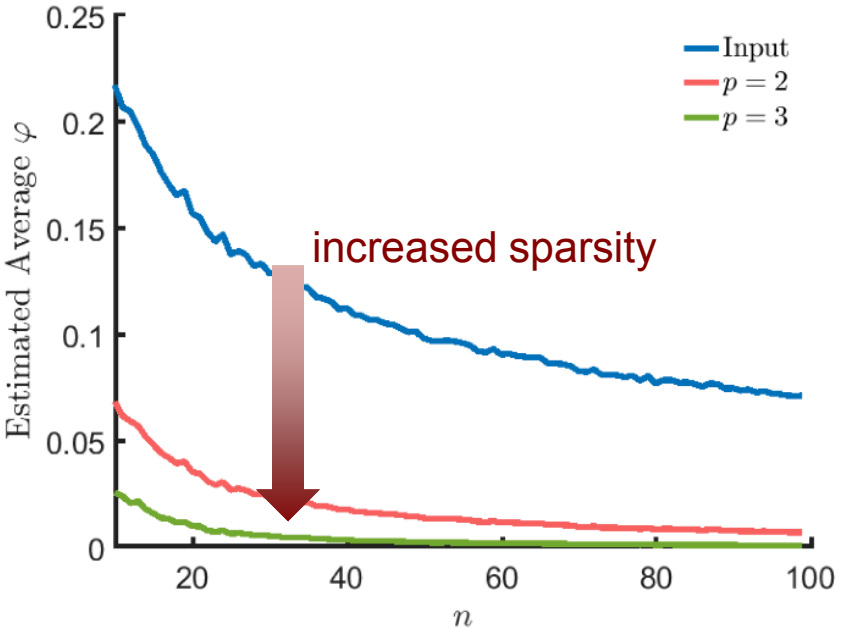


- Direct EC→CA3 is dense but weak (recall)
- Path via DG is sparse and strong (training)
- DG performs:
  1. Sparsity increases → Content-addressable Memory
  2. Pattern separation → Threat detection

Modeling the “sparsity transformation” in neurobiological systems



DG increases sparsity of EC input – improved separation of features

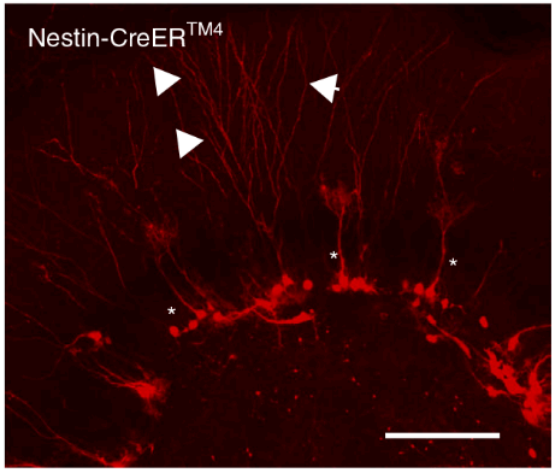


Sparse coding decreases the normalized correlation for any pair of vectors, and thus increases the estimated ‘sparsity’.

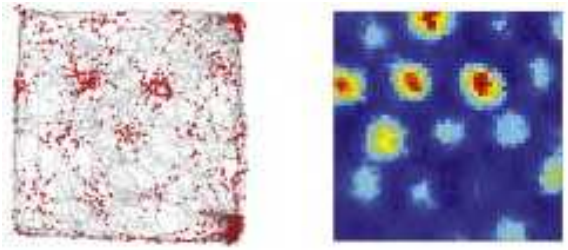
Severa et al, accepted Neural Computation 2016

Neurogenesis increases coding flexibility

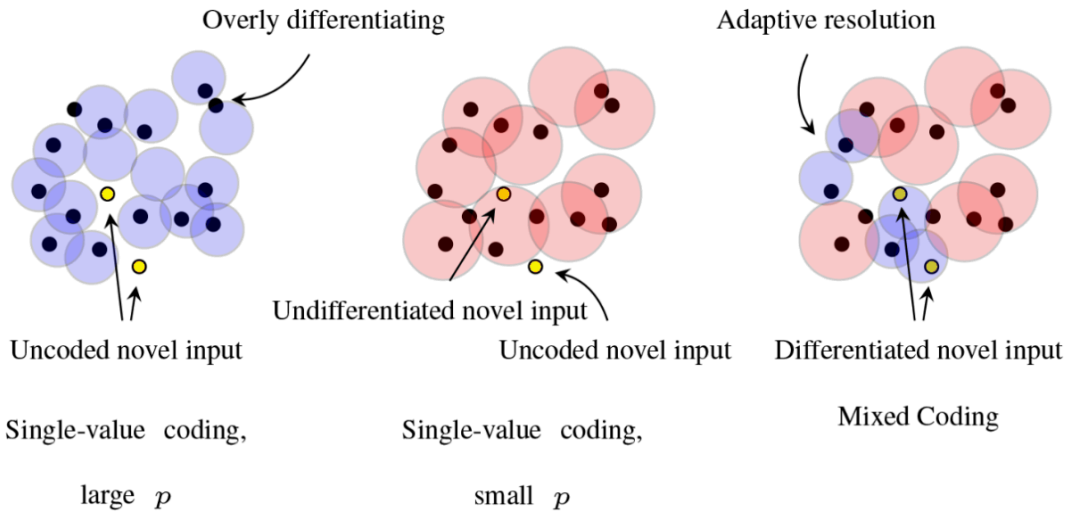
- Adult neurogenesis improves information capacity
- Grid cells (EC to DG) encode spatial dimensions



Dieni et al, Nature Comm 2016



Giocomo et al, Neuron 2011



- Dentate code is compatible with the introduction of new neurons and the refinement of old neurons
- Mixed heterogeneous code allows for adaptation to novel inputs, increased capacity

Severa et al, accepted Neural Computation 2016

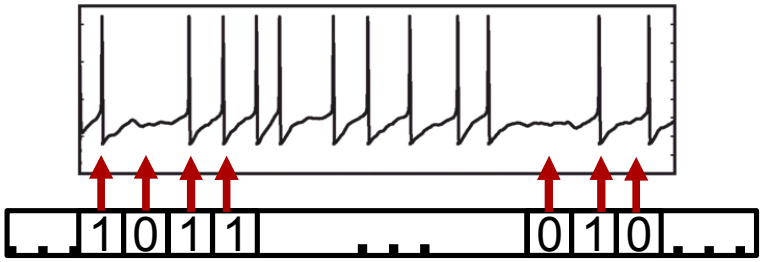
Quantifying information content in neural signals

- Information and probability are connected:

$$H(X) = \sum_{x \in X} p(x) \log\left(\frac{1}{p(x)}\right)$$

$p(x) = ?$

- Transform spike trains into compressed representations of information:



- Complexity and entropy are related and can be estimated:

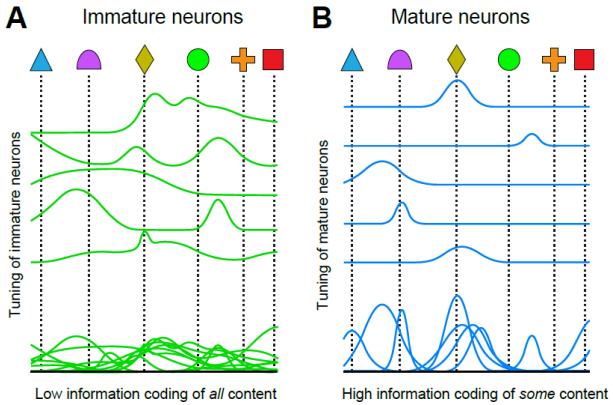
$$c_\alpha(x^n) = \frac{C_\alpha(x^n)}{n} * log_\alpha n \qquad \lim_{n \to \infty} \sup c_\propto(x^n) \leq H_\alpha(S)$$

Exploring the impact of neural information content in a real-world application

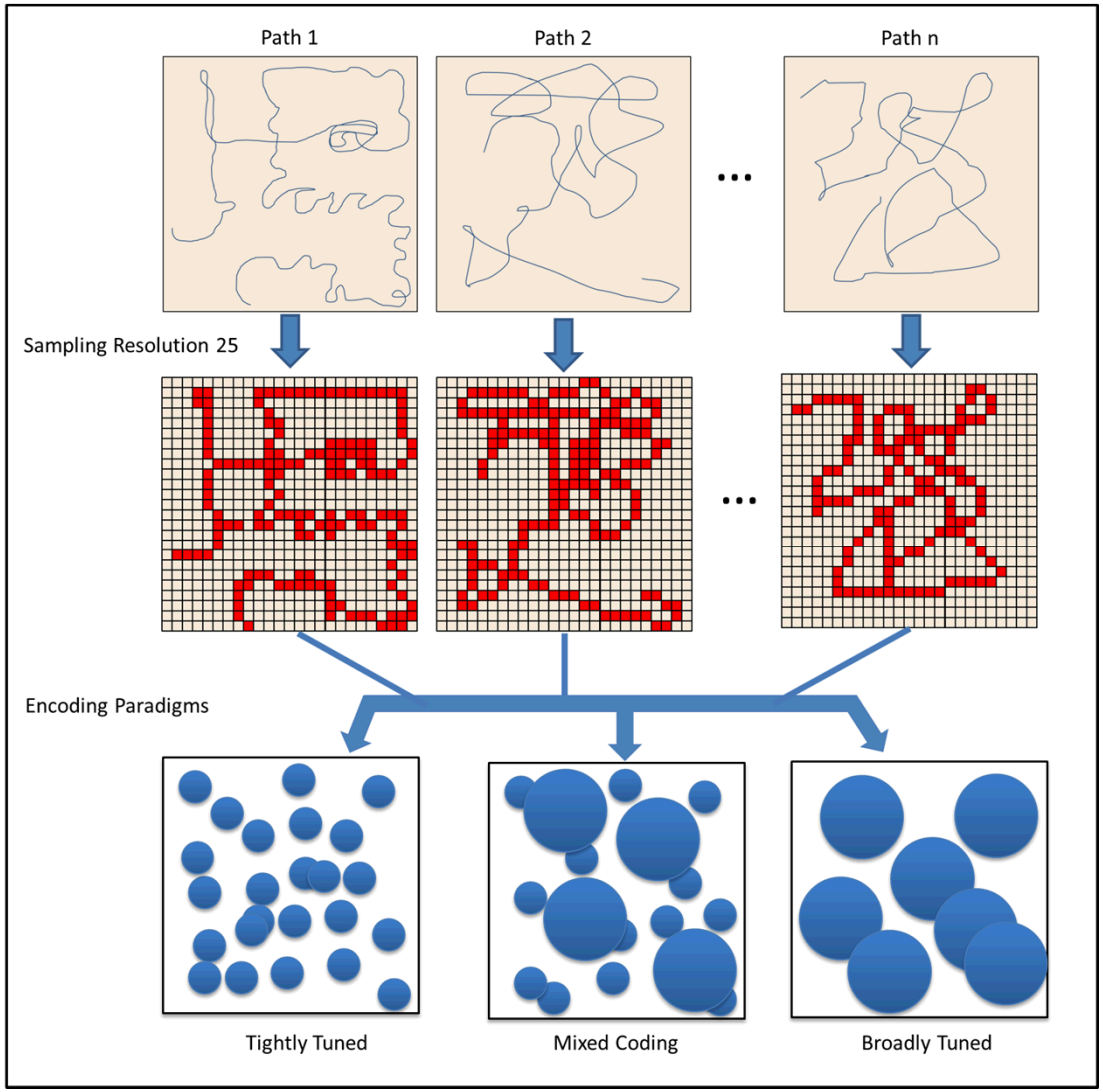
- Estimate the neural information content in place cell encoding of spatial locations

- Map information content to the neural space (resolution, efficiency)

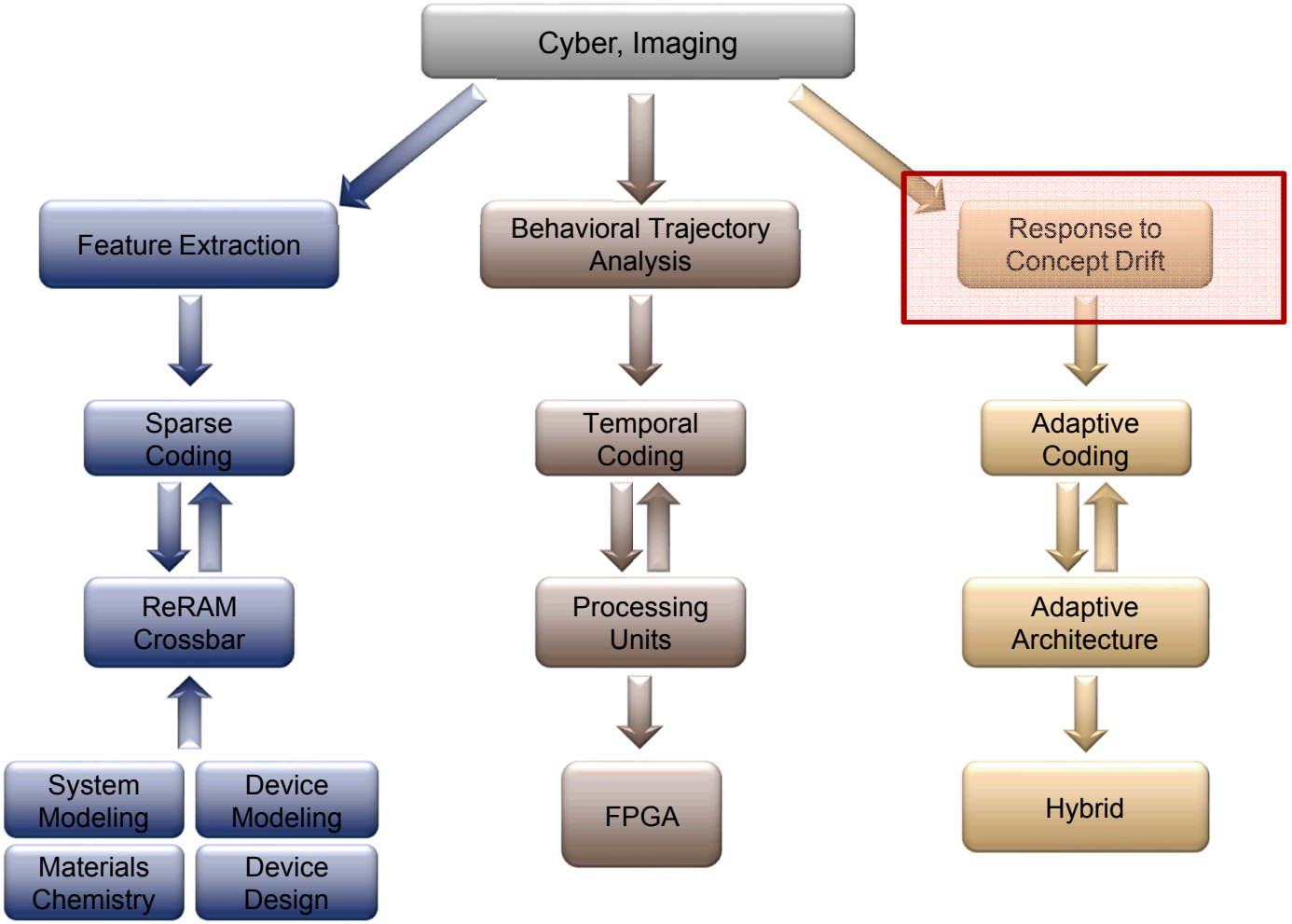
- Examine the impact of neurogenesis on encoding and information content



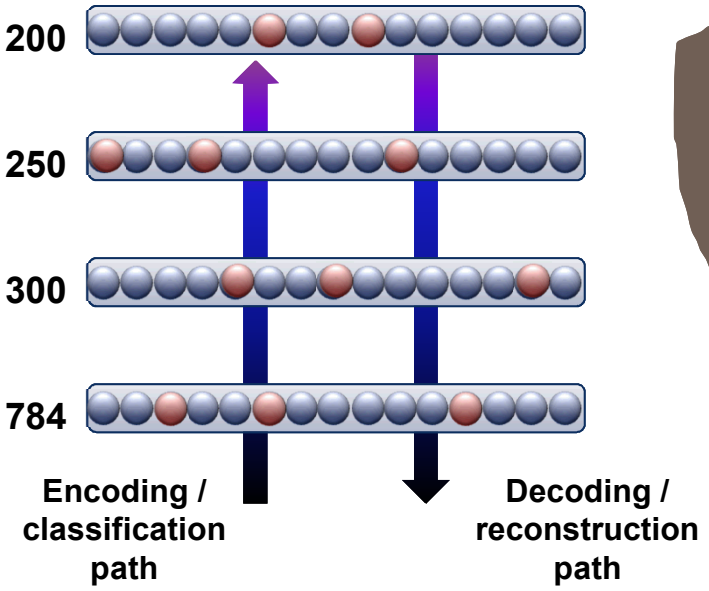
Aimone, Deng and Gage, Neuron 2011

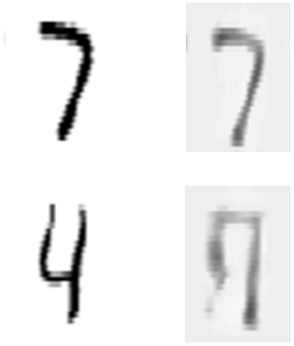


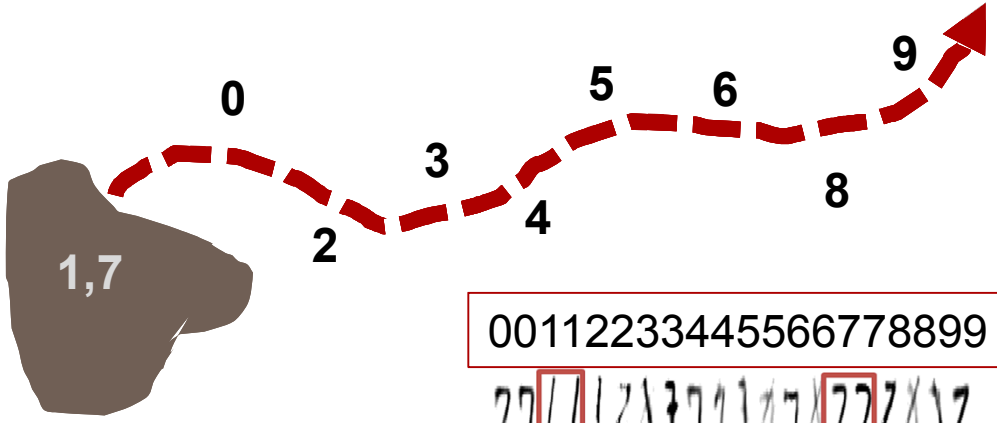
Algorithmic approaches to handle concept drift in data

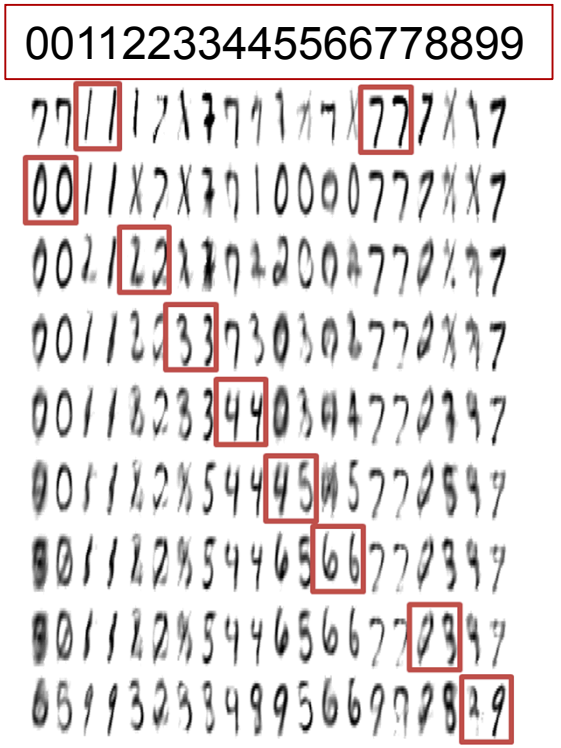


Data-driven computing methods are limited… by data; auto-encoder for data reconstruction



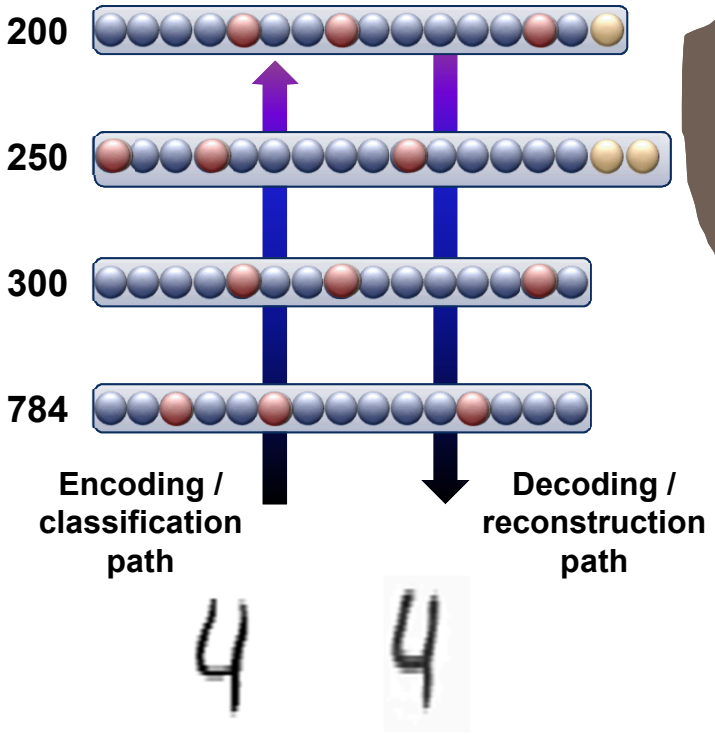


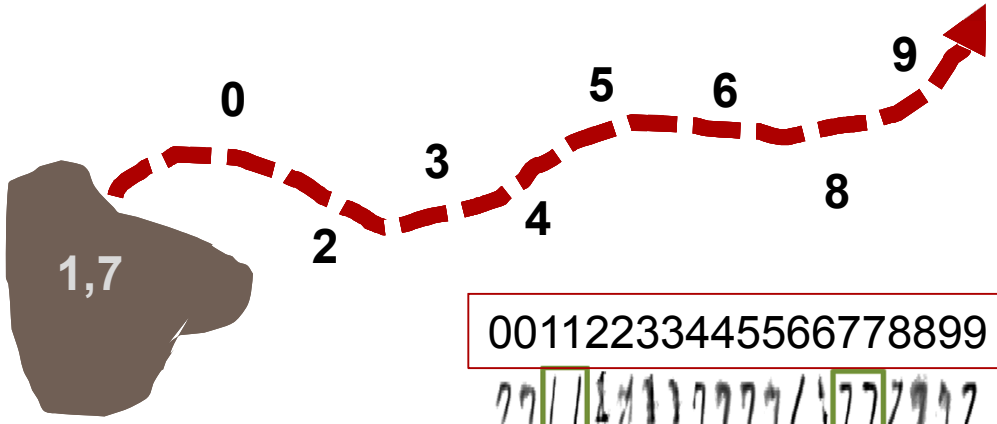


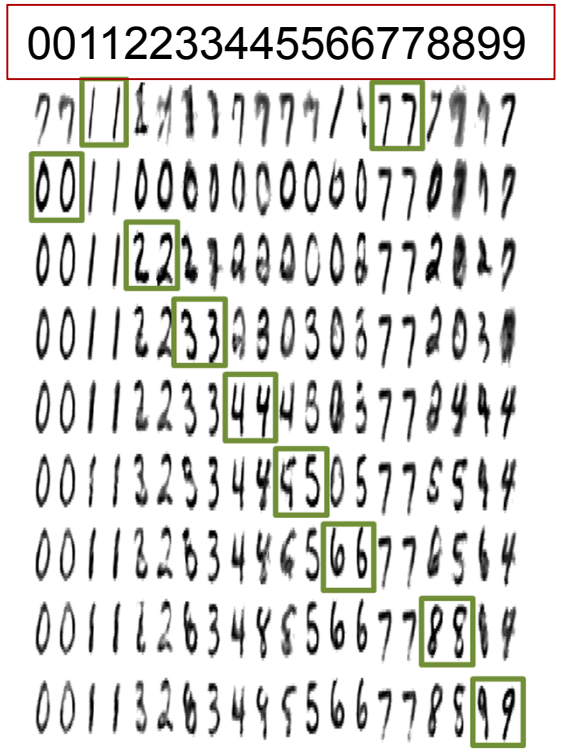


Draelos et al, ICLR 2016

“Neurogenic deep learning” enables adaptation to changing data

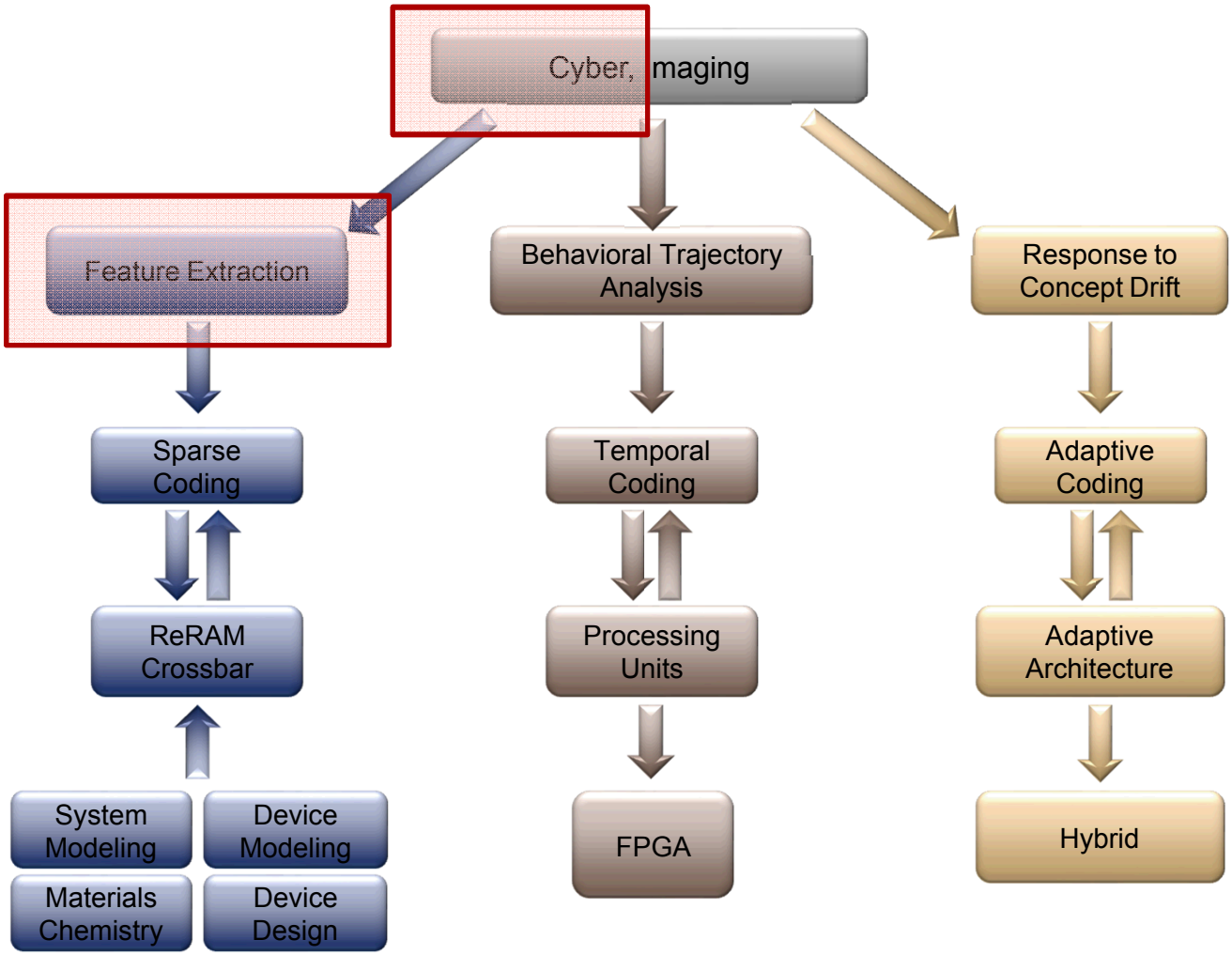




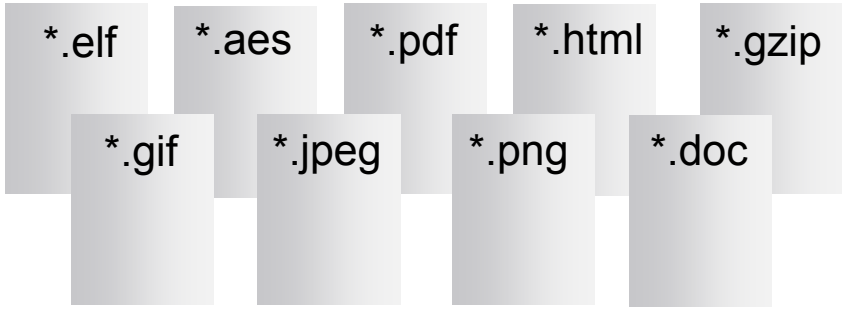


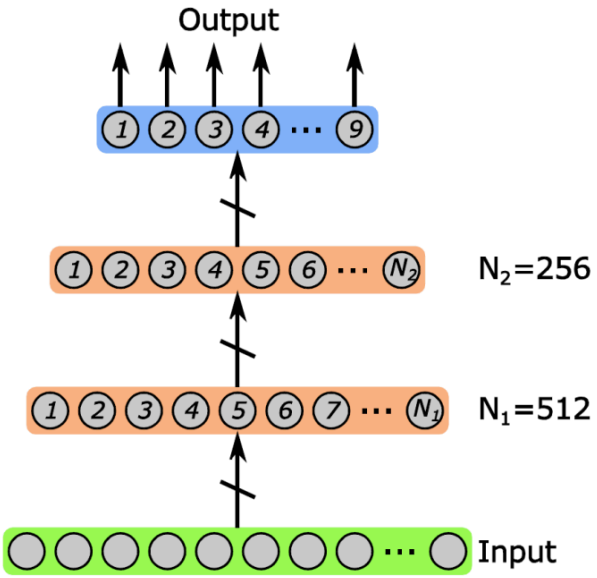
Draelos et al, ICLR 2016

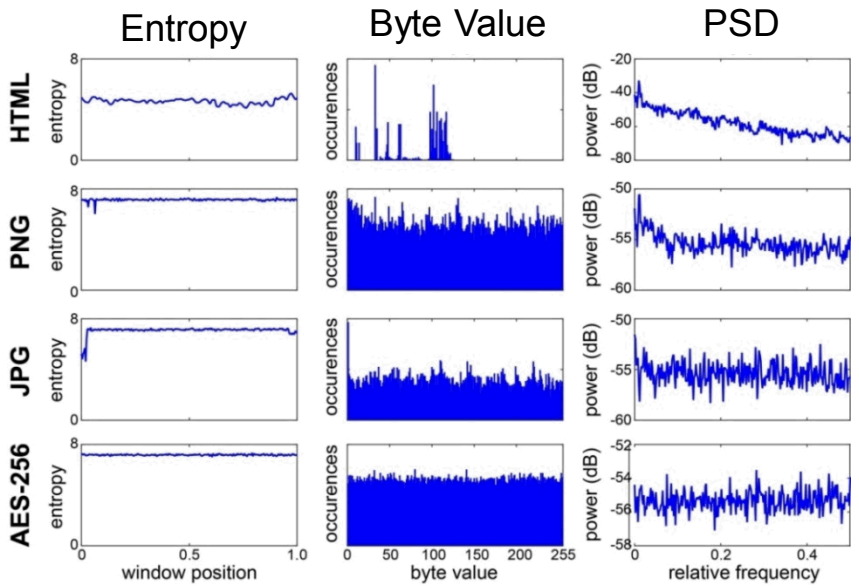
Extracting features from cyber data



Cyber data identification with a deep neural network classifier



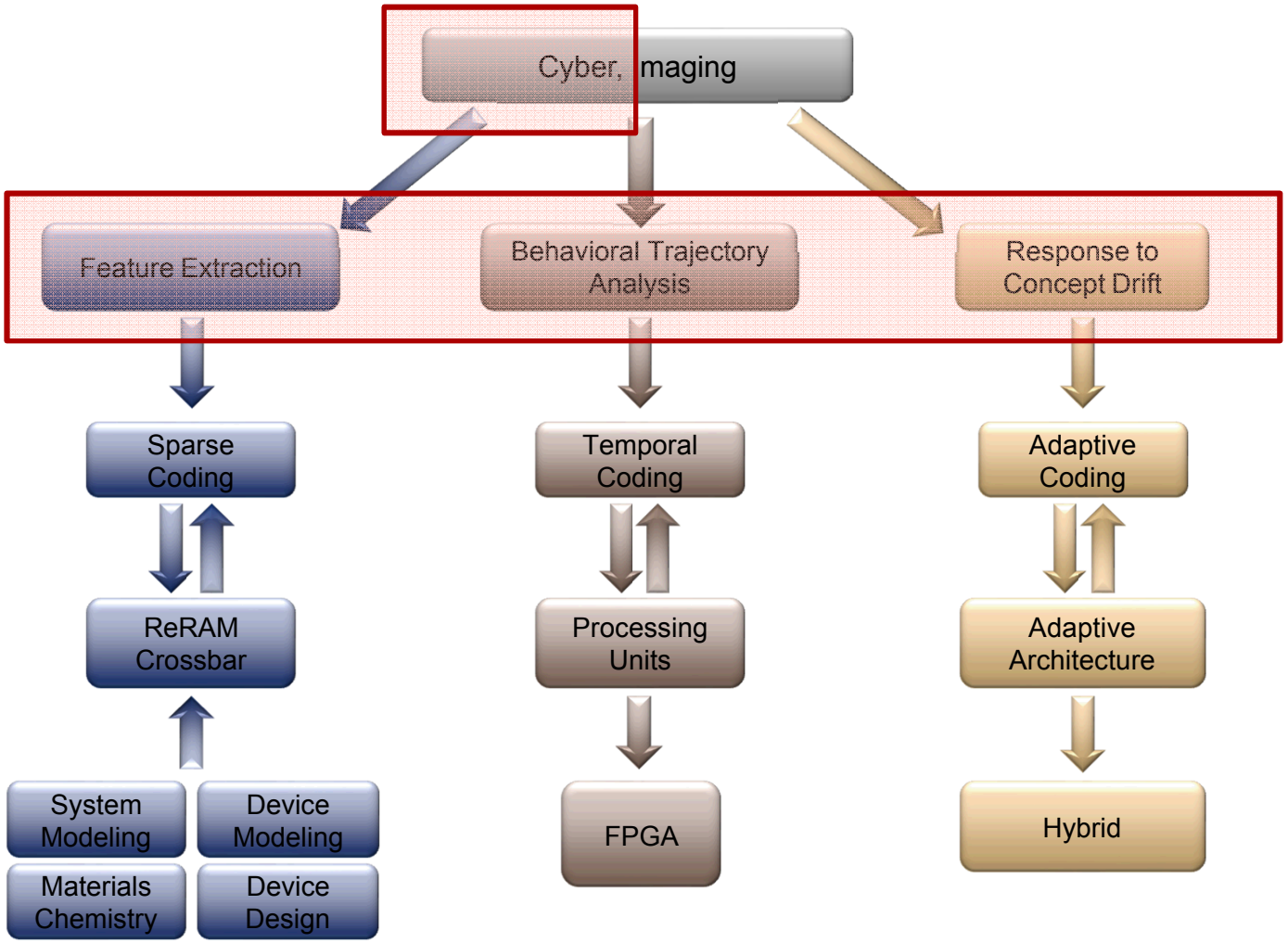




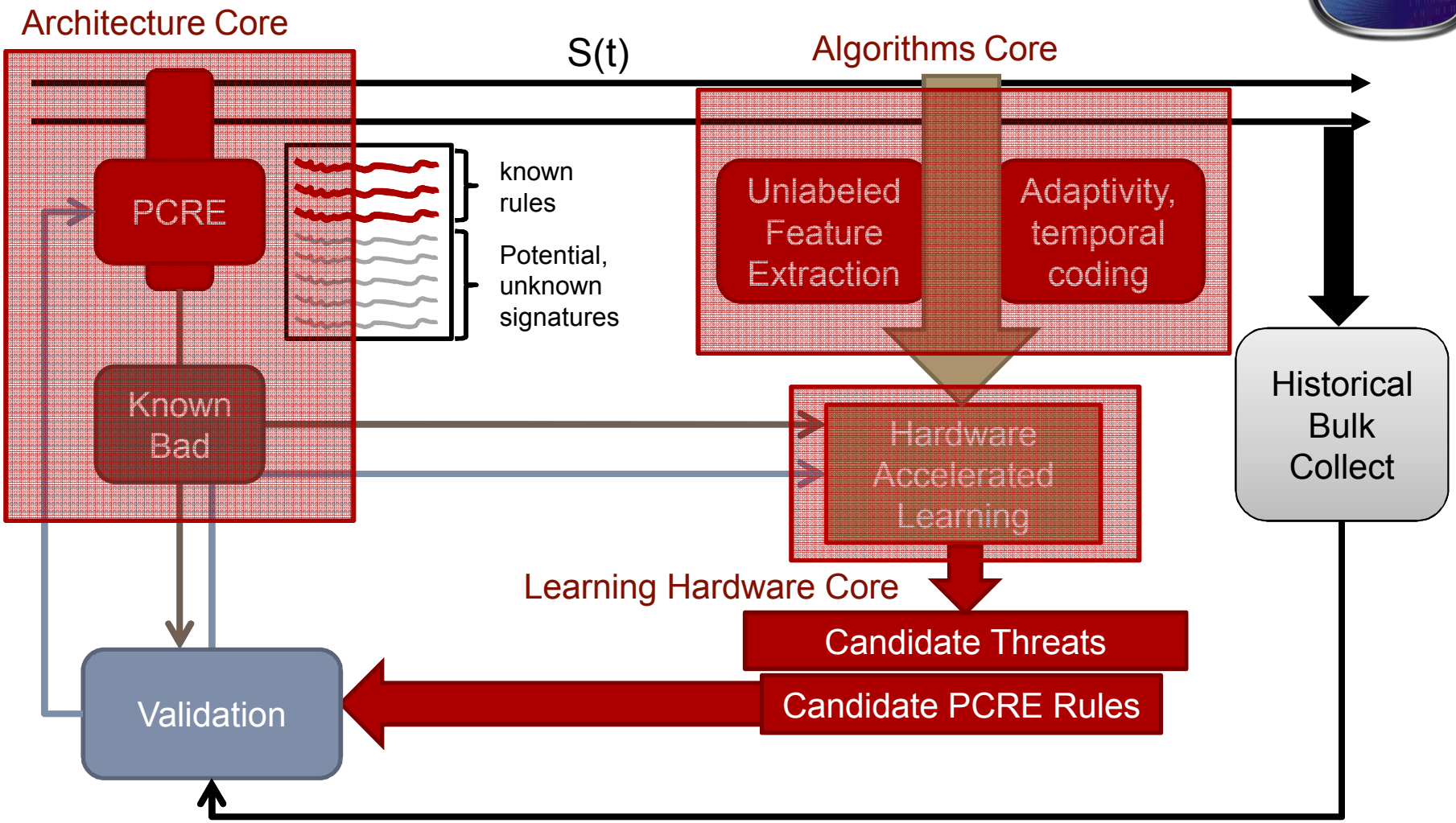
		Predicted Class								
		HTML	PNG	JPEG	GIF	PDF	DOC	ELF	GZIP	AES
Actual Class	HTML	100								
	PNG		91	1		1	1	1	1	4
	JPEG		1	99						
	GIF				100					
	PDF		1	3	1	95				
	DOC		1				99			
	ELF							100		
	GZIP								100	
	AES		5							95

Cox, Aimone, James; Complex Adaptive Systems, Nov. 2015; Procedia Comp Sci 61, 349

Dynamic learning applications in cyber data

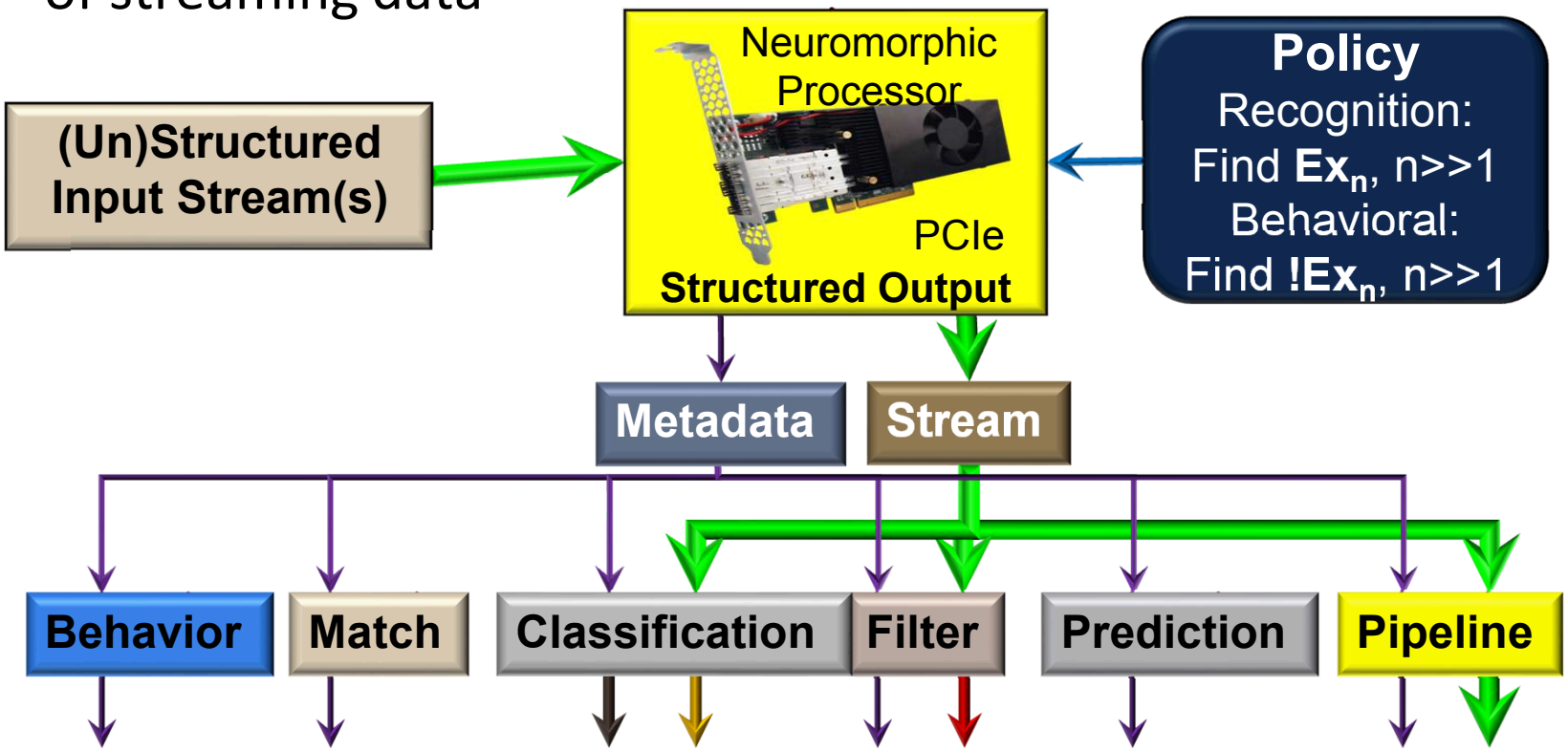


Tracking the Known: Bracketing Previously Identified Threats



Accelerate PCRE processing with Temporal Processing Unit (TPU)

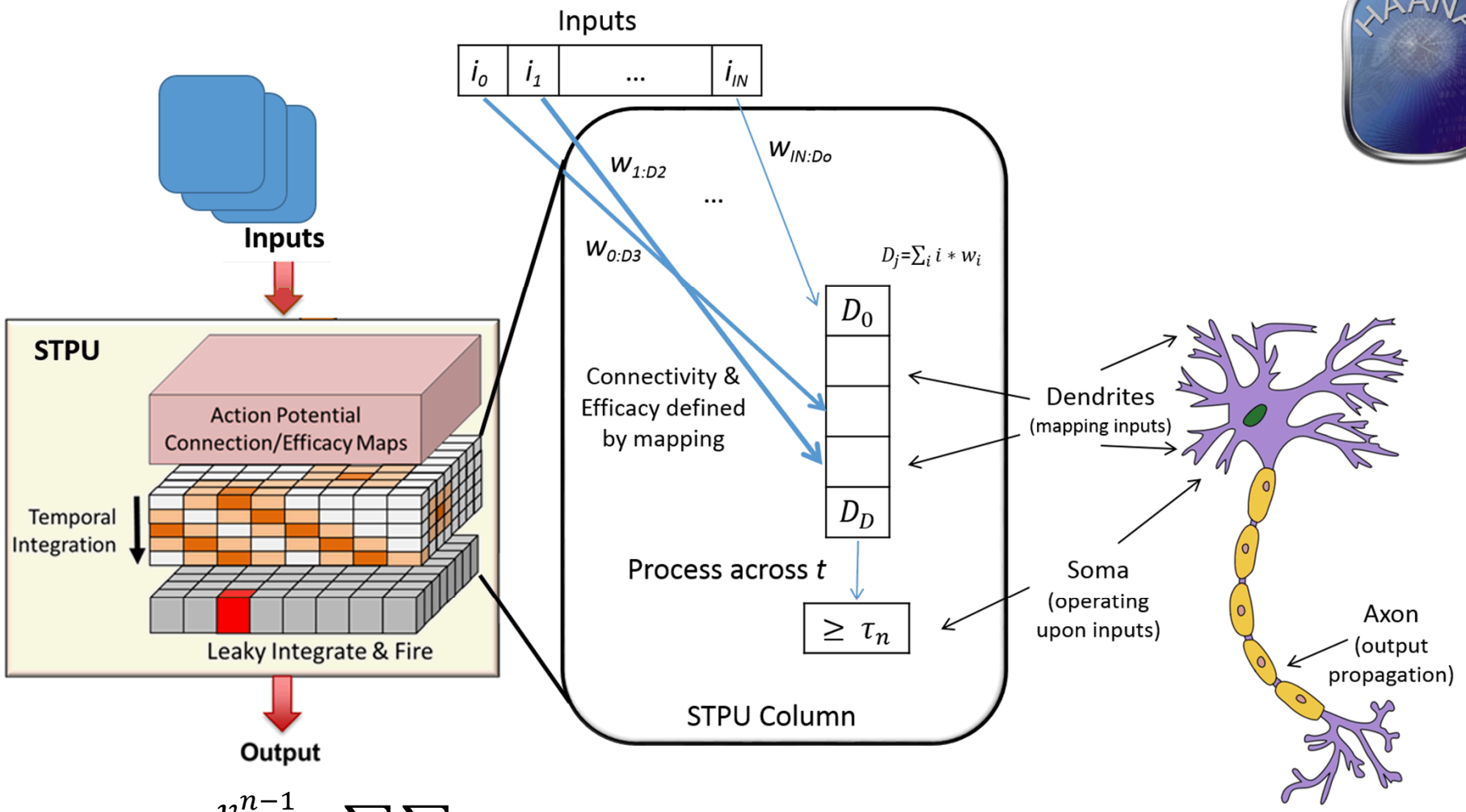
- Collaboration with Lewis Rhodes Labs (LRL)
- Patented neural inspired architecture for temporal processing of streaming data



TPU Integration for PCRE Processing

- API for usability
  - Hide complexity so no neural knowledge required by practitioners
  - Waterslide is an opensource highly optimized stream processing code
    - Hear all about this from Karl Anderson tomorrow
    - **Seamlessly integrated TPU API in a mater of weaks**
- Significant PCRE performance improvement
  - **My initial testing showed 15X improvement over Google re2 library**
- Significant demonstrated power reduction
  - David Follett will tell the rest of the story tomorrow
- Seamlessly integrated in the Tracking the Known testbed
- TPU product “Neuromorphic Cyber Microscope” is an R&D100 finalist in 3 categories

Spiking Temporal Processing Unit (STPU)



$$v_m^n = v_m^{n-1} - \frac{v_m^{n-1}}{\tau_m} + \sum_i \sum_j w_{mi} \cdot s(t - t_{ij} - d_i)$$

Liquid State Machine

- Developed by Wolfgang Maass
- Reservoir computing
  - Echo State Machines
  - Liquid State Machines
- Different items at different locations at different times
- Differences between the patterns are amplified by the liquid
- Mimics brain functionality
- Supervised learning

Maass, W., Markram, H., *On the Computational Power of Recurrent Circuits of Spiking Neurons*, Journal of Computer and System Sciences 69(4): 593-616, 2004.

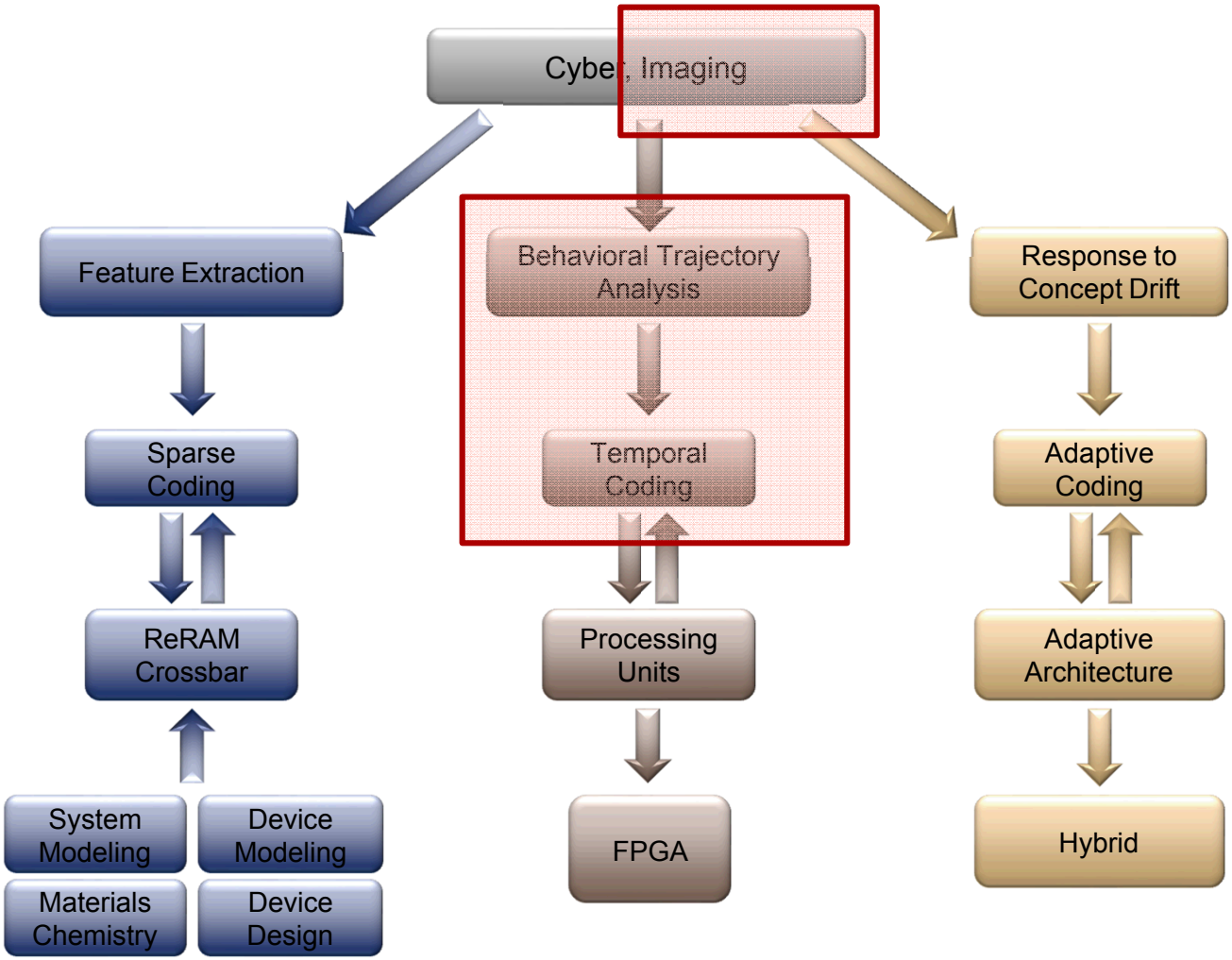
Demonstrated LSM Applications

- Speech and audio recognition
- Image Pattern Recognition
- Music Classification
- Robot Path Planning
- Fingerprint Scanners
- Facial emotion recognition

Mapping LSMs onto the STPU

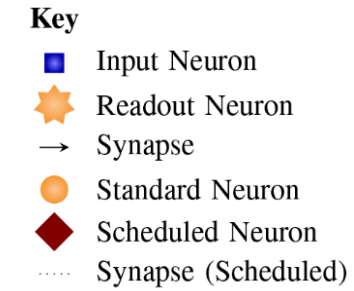
- Goals:
  - Implement the rich dynamics of a LSM efficiently
  - Recurrence
  - Exponential synaptic response functions
  - Drives improvements to STPU architecture
- **Currently implemented LSM on version 1 of STPU**
- Demonstrated speech recognition with minimal parameter tuning using ridge regression
- Comparison with Zhang et al. 2015:
  - Use state variables to keep track of synaptic responses. Time constants are binary (division becomes bit shifting)
  - STPU uses weights to put values into the temporal stack

Extracting features from cyber data

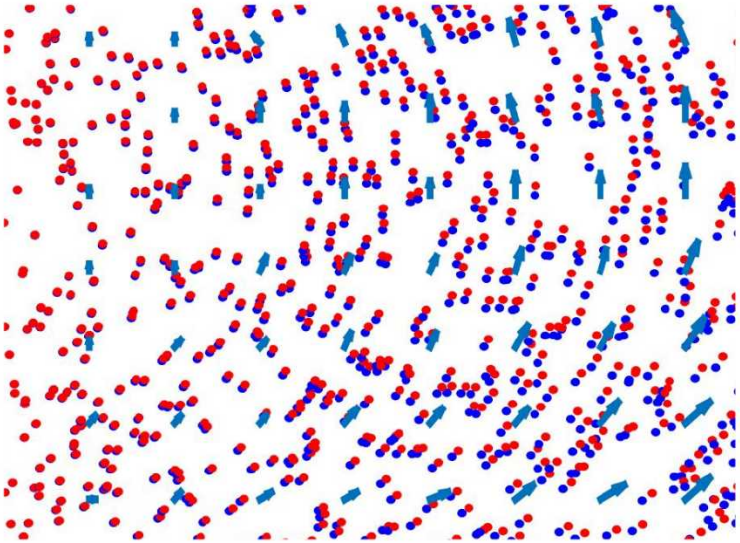


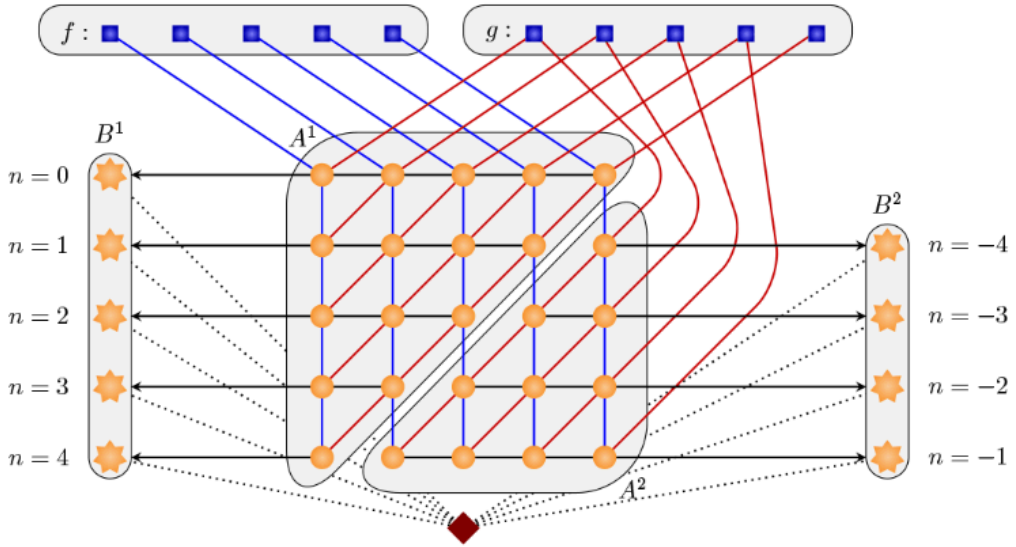
Spiking network algorithm for computing cross-correlations in particle image velocity (PIV)

- Data can be stored in the dynamics of a network
- Dynamics allow for trade-off: Time ↔ Neurons
- Neural algorithms can match or best traditional 'big *O*'
- Efficient implementation on the STPU



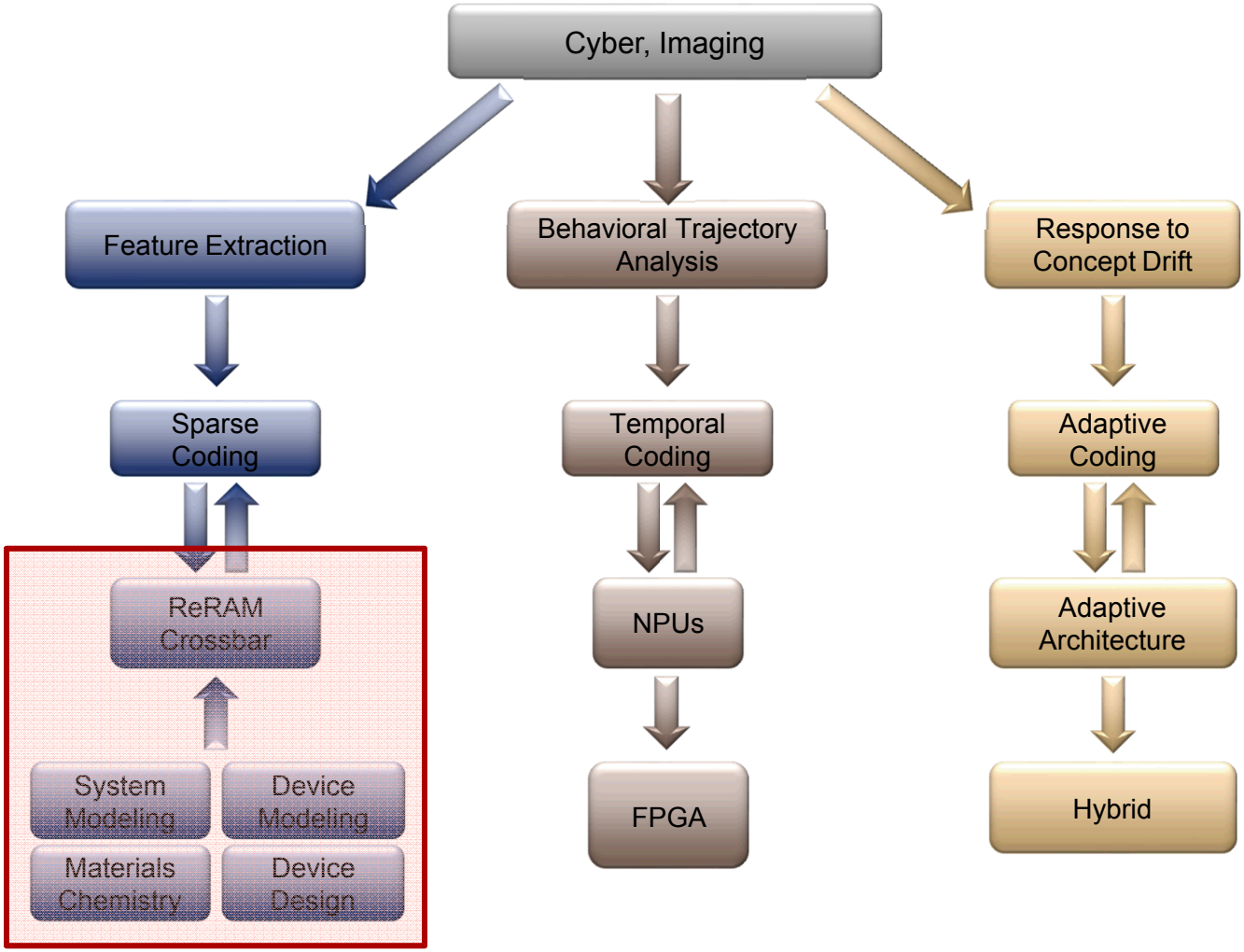
Example 2D flow-field (t_0 an t_1)



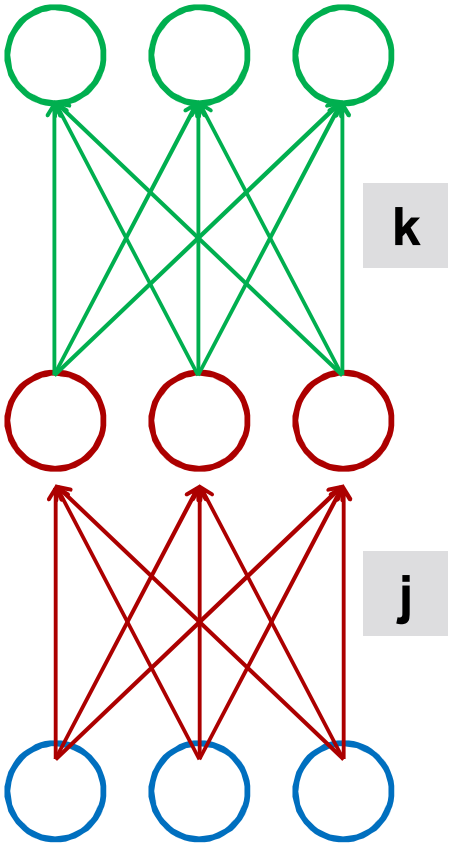


Severa et al, accepted ICRC 2016

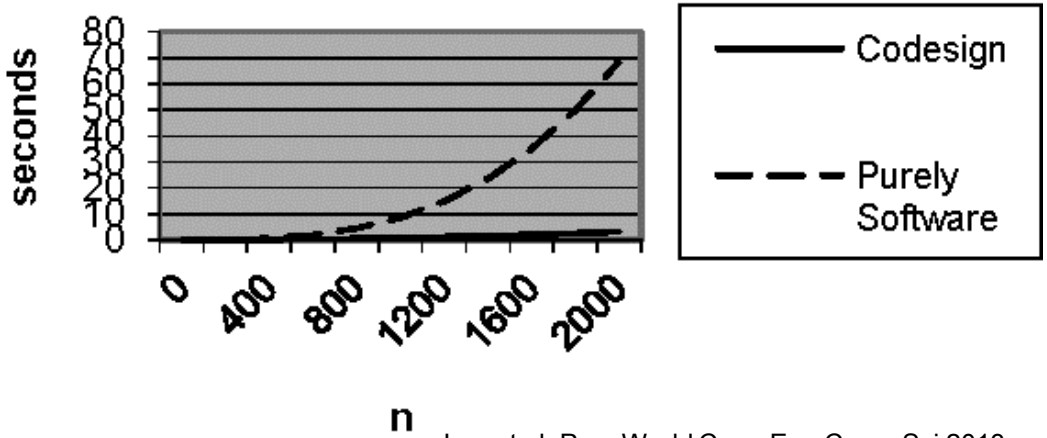
Learning crossbar performance assessment



Use hardware acceleration (existing or novel technology) to speed-up data processing in neural-inspired algorithms



Agarwal et al, IJCNN 2016



Lee et al. Proc World Cong Eng Comp Sci 2013

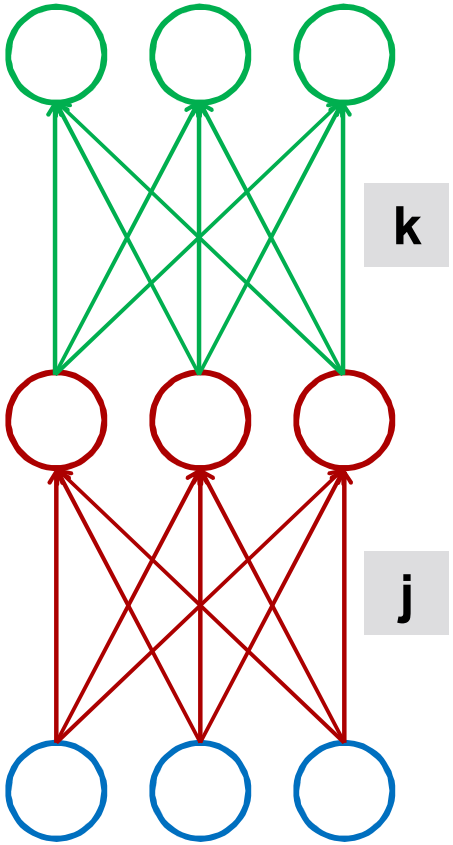
Configuration	# of chips	Chip area (mm^2)	% active	Power (W)	**Power eff. over Xeon**
	Example 1: 25,600 neurons 100,000 iterations/s				
Memristor Analog (config 4)	1	5.9	38.6%	0.07	**234,859**
Memristor Digital (config 5)	1	18.2	89.6%	0.62	**16,968**
SRAM (config 6)	1	29.1	89.6%	1.13	**8,215**
NVIDIA M2070	12	529.0	99.2%	2700.00	**6**
Intel Xeon X5650	179	240.0	99.9%	17005.00	**1**

Merolla, P. A. *et al. Science* 345, 668-673 (2014)

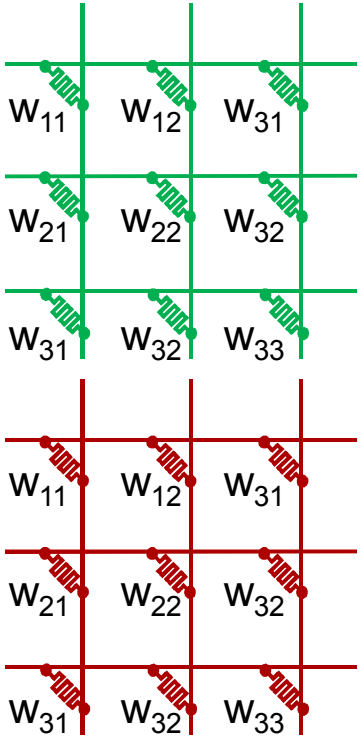
T. Taha, et al., Proc. IEEE Intl. Joint Conf. on Neural Networks, 2013.

Translating non-spiking neural network algorithms into hardware

Network



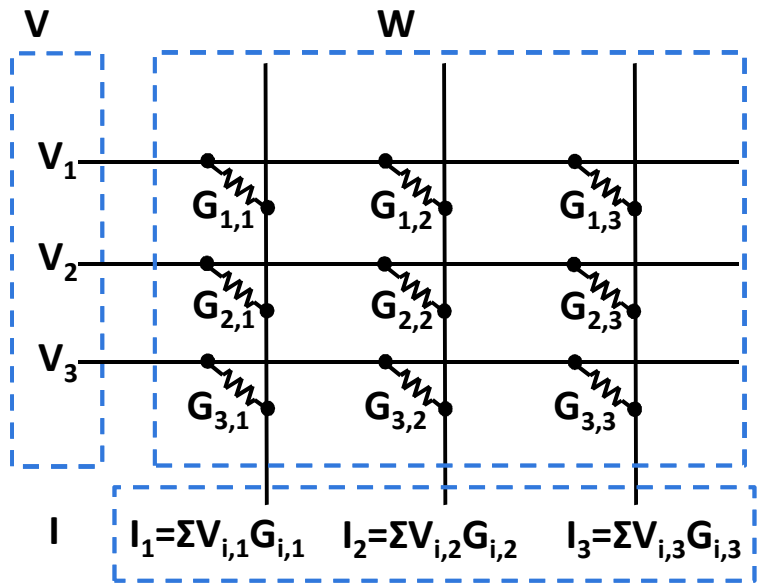
RRAM analog crossbar



$$V^T W = I$$

$$\begin{bmatrix} V_1 & V_2 & V_3 \end{bmatrix} \begin{bmatrix} W_{1,1} & W_{1,2} & W_{1,3} \\ W_{2,1} & W_{2,2} & W_{2,3} \\ W_{3,1} & W_{3,2} & W_{3,3} \end{bmatrix} =$$

$$\begin{bmatrix} I_1=\Sigma V_{i,1}W_{i,1} & I_2=\Sigma V_{i,2}W_{i,2} & I_3=\Sigma V_{i,3}W_{i,3} \end{bmatrix}$$

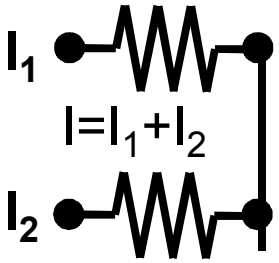


Agarwal et al, IJCNN 2016

Use resistive memory elements for low-power local computation

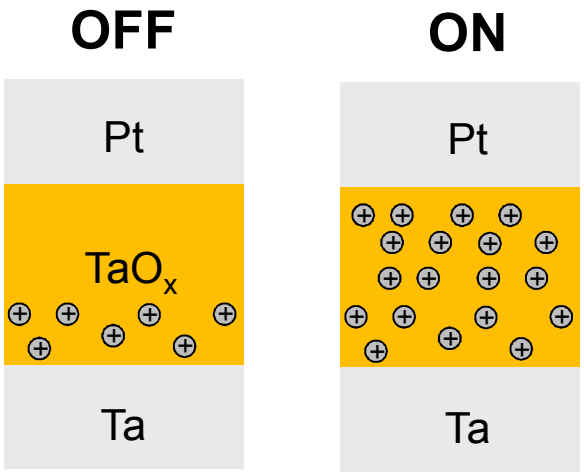
$V = I \times R$

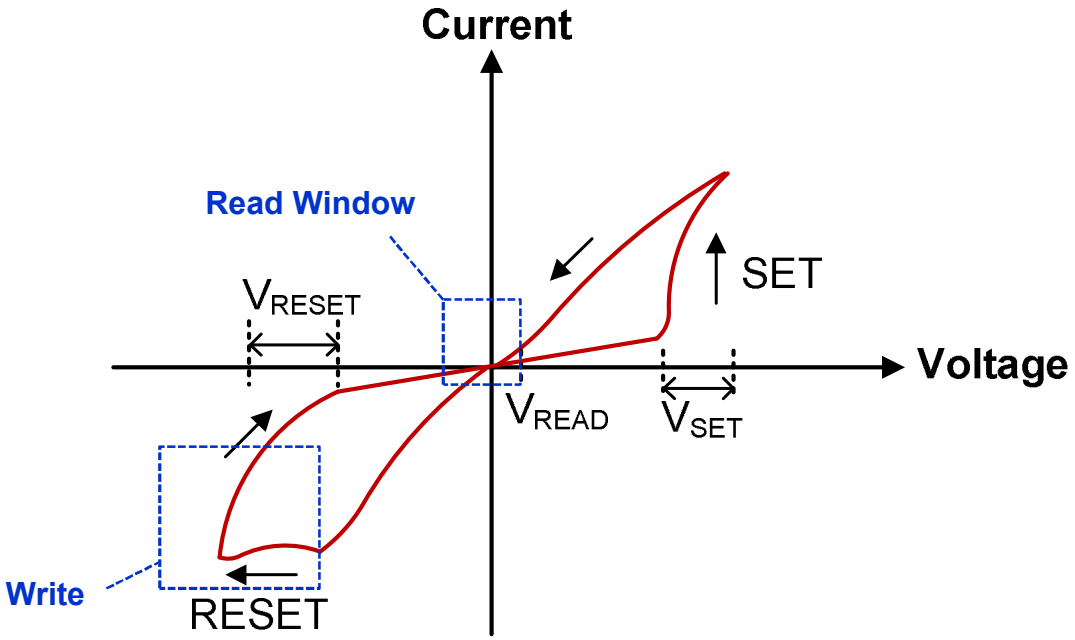
$I = G \times V$



A resistive memory or ReRAM is a programmable resistor

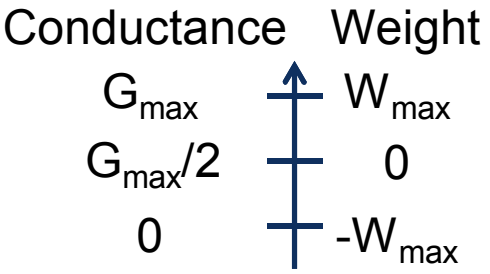
- apply small V to read G
- apply large V to change G



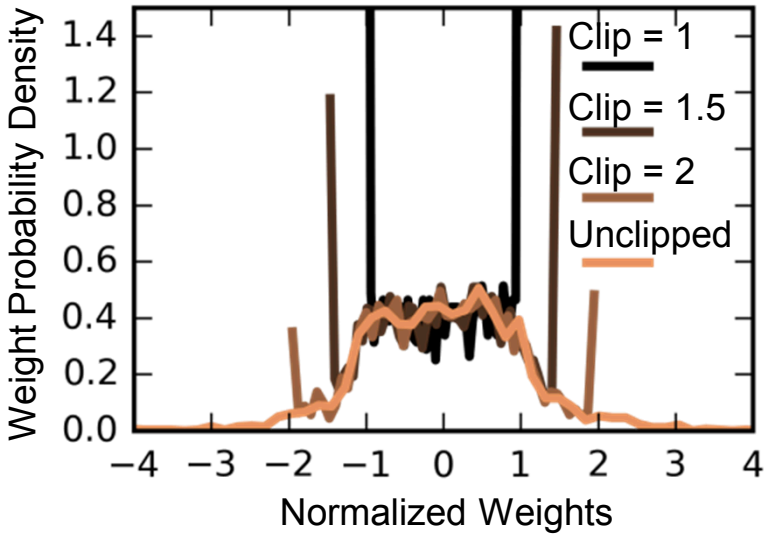


Mapping resistive memory devices to neural algorithm weights

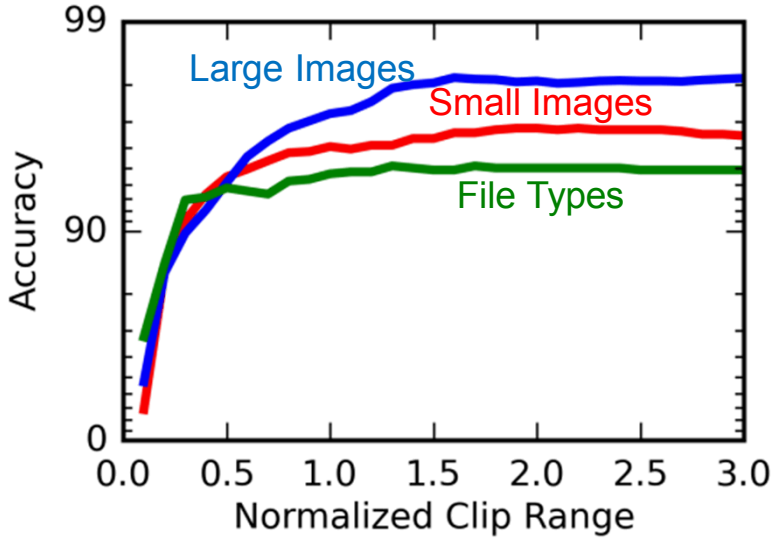
Data set	# Training Examples	# Test Examples	Network Size
UCI Small images	3,823	1,797	64×36×10
File types	4,501	900	256×512×9
MNIST large images	60,000	10,000	784×300×10



Small Images, First Layer

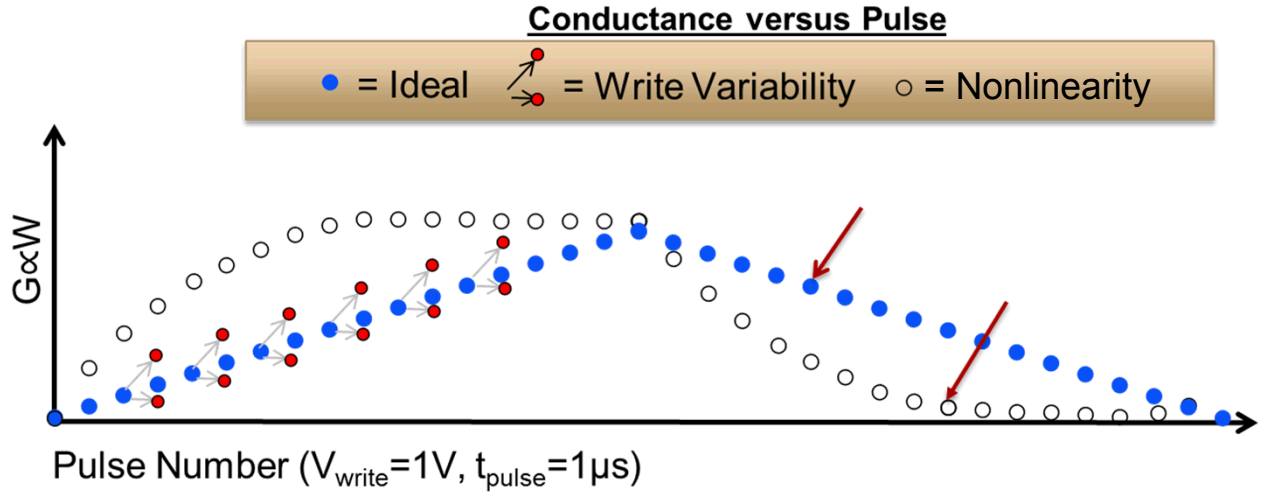


Weight Range Clipping



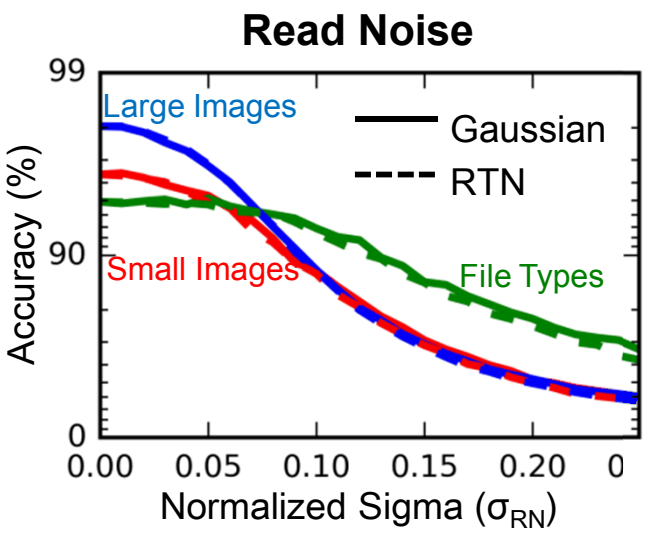
Agarwal et al, IJCNN 2016

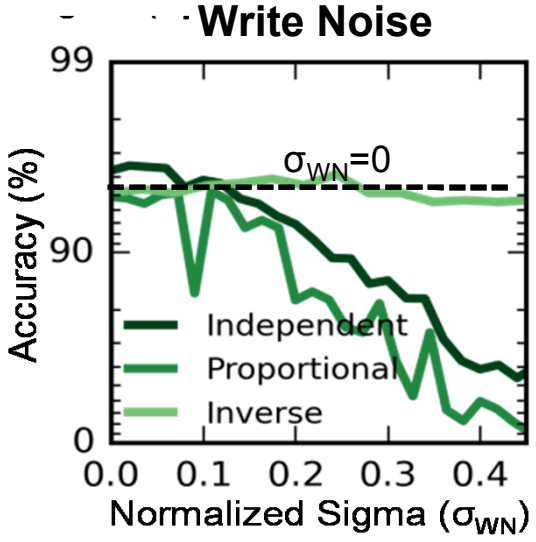
Resistive memory device characteristics are non-ideal

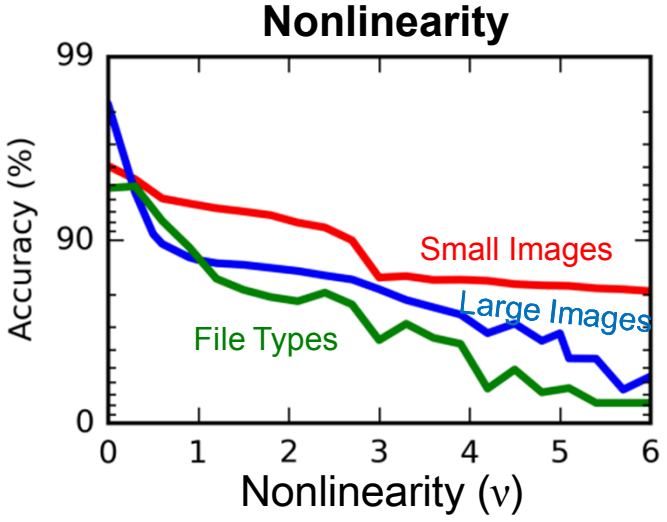


High training accuracy requires:

- Low Write Variability
- Low Write Nonlinearity
- Low Asymmetry
- Low Read Noise



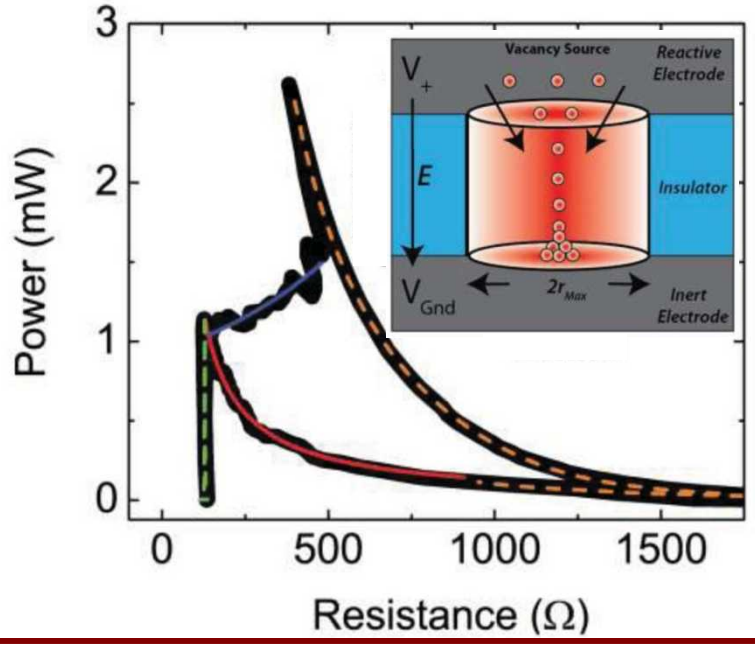




Agarwal et al, IJCNN 2016

Designing, modeling, and fabricating devices with improved neural algorithm characteristics

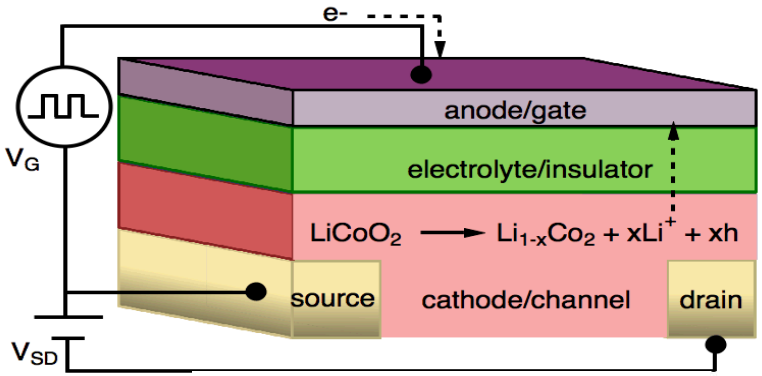
ON switching, ON state, OFF switching, OFF state

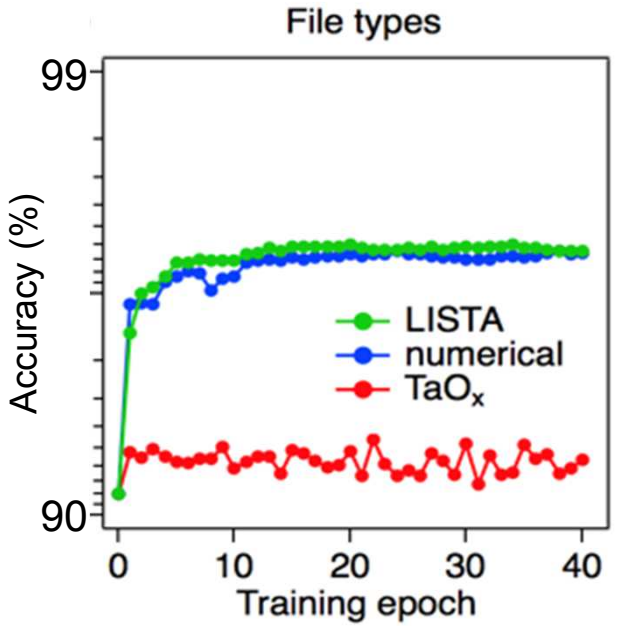


Filament surface temperature (T_s):

$$T_s = T_{RT} + \sigma V^2 \frac{d_E}{2k_E d_o}\left[1 - \frac{k_E}{k_F}\frac{r_F^2}{4d_E d_o}\right]$$

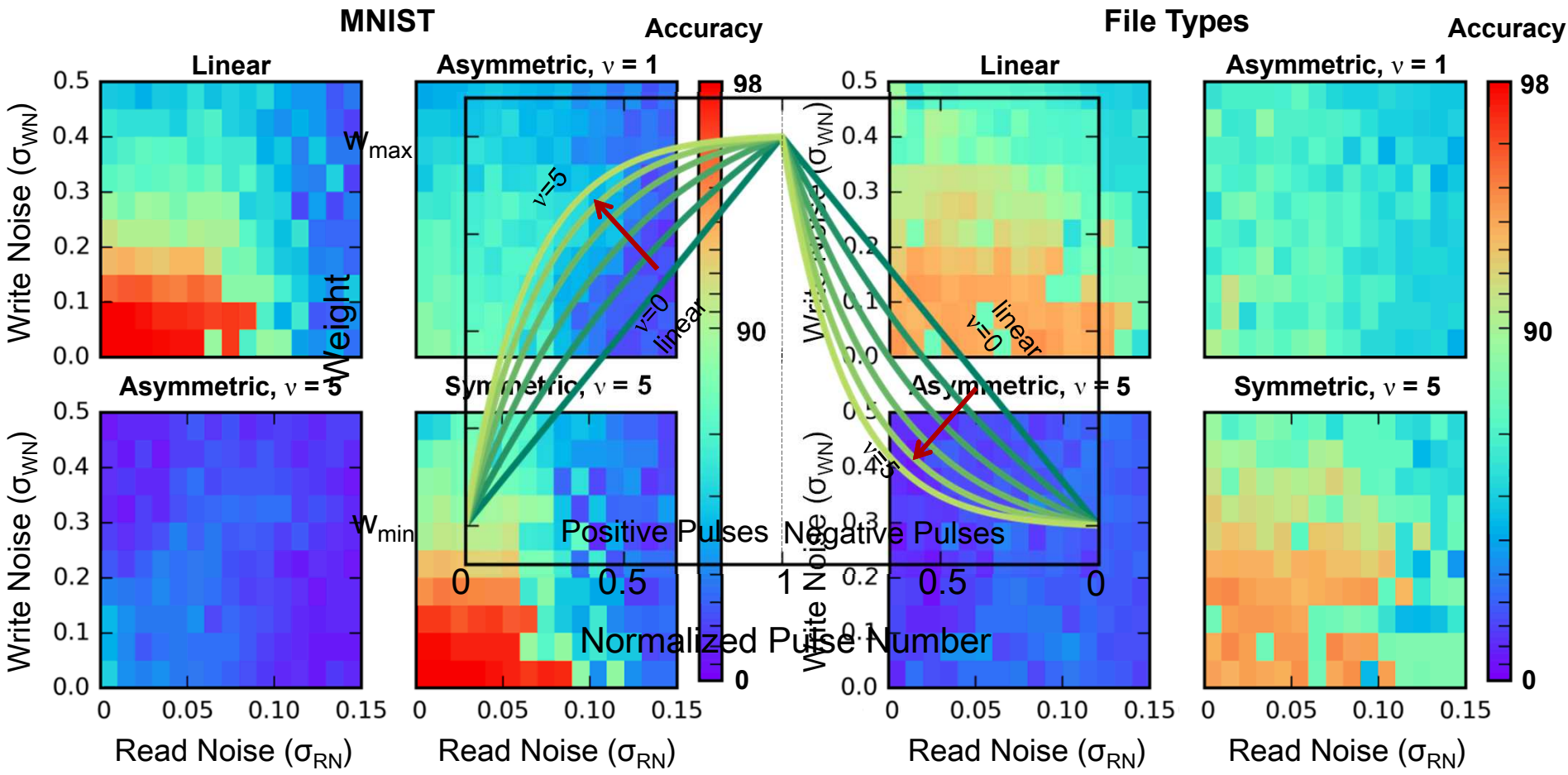
Mickel, Lohn, James, and Marinella, Adv Mater, 26, 4486, 2014
Landon et al., APL 2015, 107, 023108





Fuller et al., Adv Mater 2016, in press

Combined impact of device non-idealities on algorithm performance



Thanks for your time!

Questions?