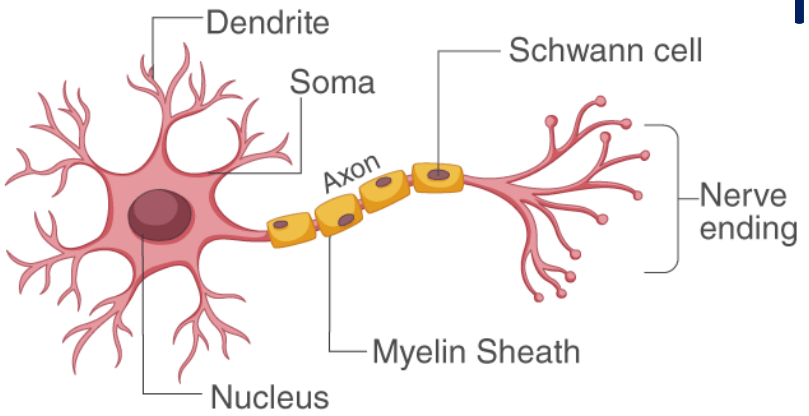
13th International Conference
Dependable Systems, Services and Technologies
Greece, Athens, October 13-15, 2023

# Tsetlin Machines: Stepping Towards Energy-efficient, Explainable and Dependable AI

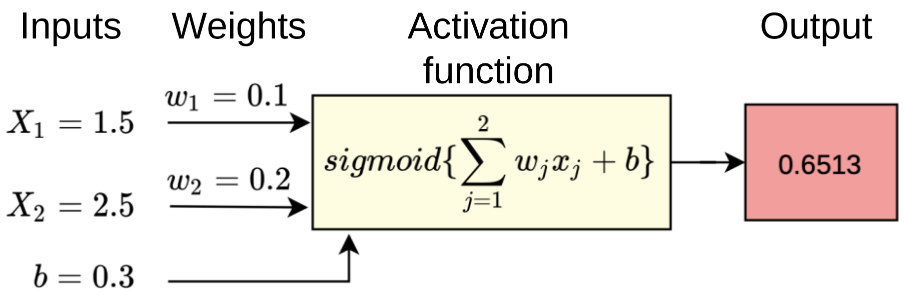

Prof. Alex Yakovlev

Newcastle University


# Artificial DNN: Architecture and Features (recap)



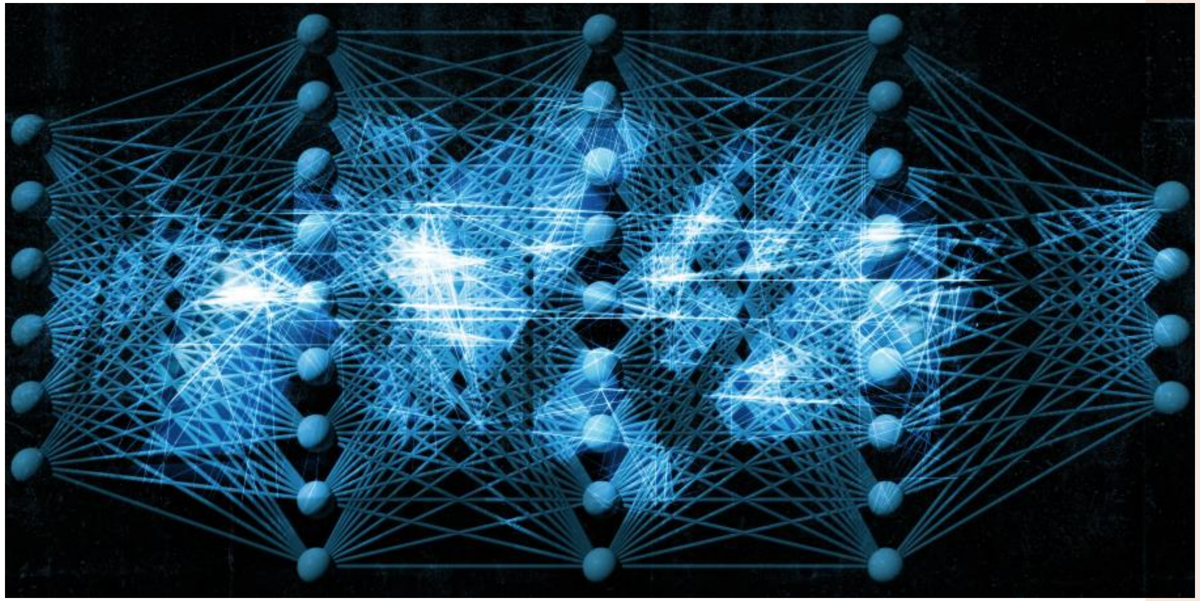

## Neuron model:




## Features:

- Number of weights to fit: $X \times HL_1 \times \ldots \times HL_k \times Y$
- Operations: Arithmetic (+,-,/,*)
- Operands: Float numbers
- Learning algorithm: Back propagation

## Network architecture:



# Artificial DNN: Major limitations (recap)

- **Black box** 〉〉〉 **Lack of explainability** 〉〉〉 **Vulnerability**

- **Too many hyperparameters**

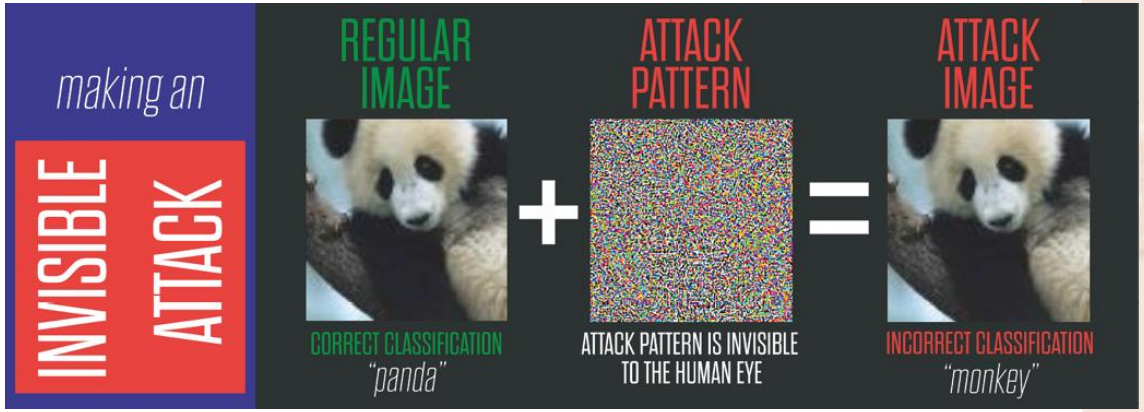
  DNN can have anywhere from a few to a **few hundred** hyperparameters

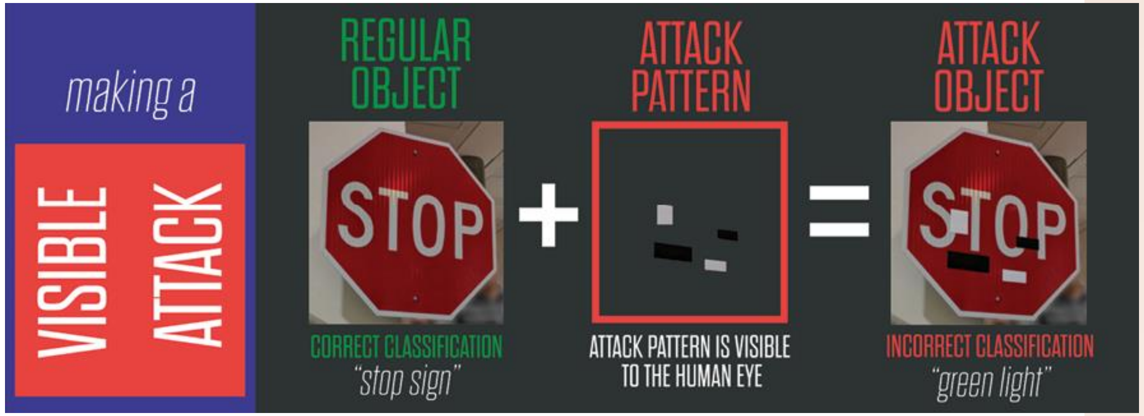
- **Hardware/energy demanding**

  ChatGPT-3 training: **1.287 GWh**

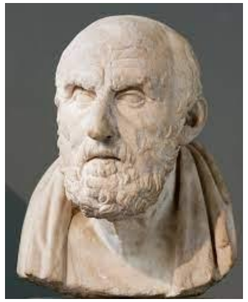
  ChatGPT-3 inference: **936 MWh**

- **Not suitable for on-chip/ at-the-edge learning**

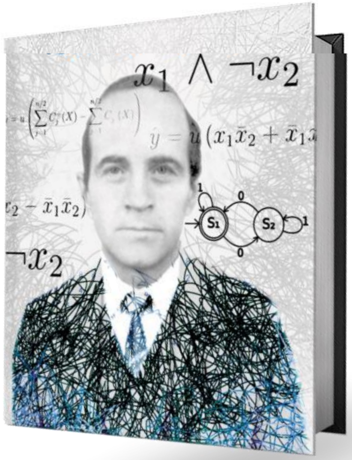


# Tsetlin Machine: Logic-based AI (2018)

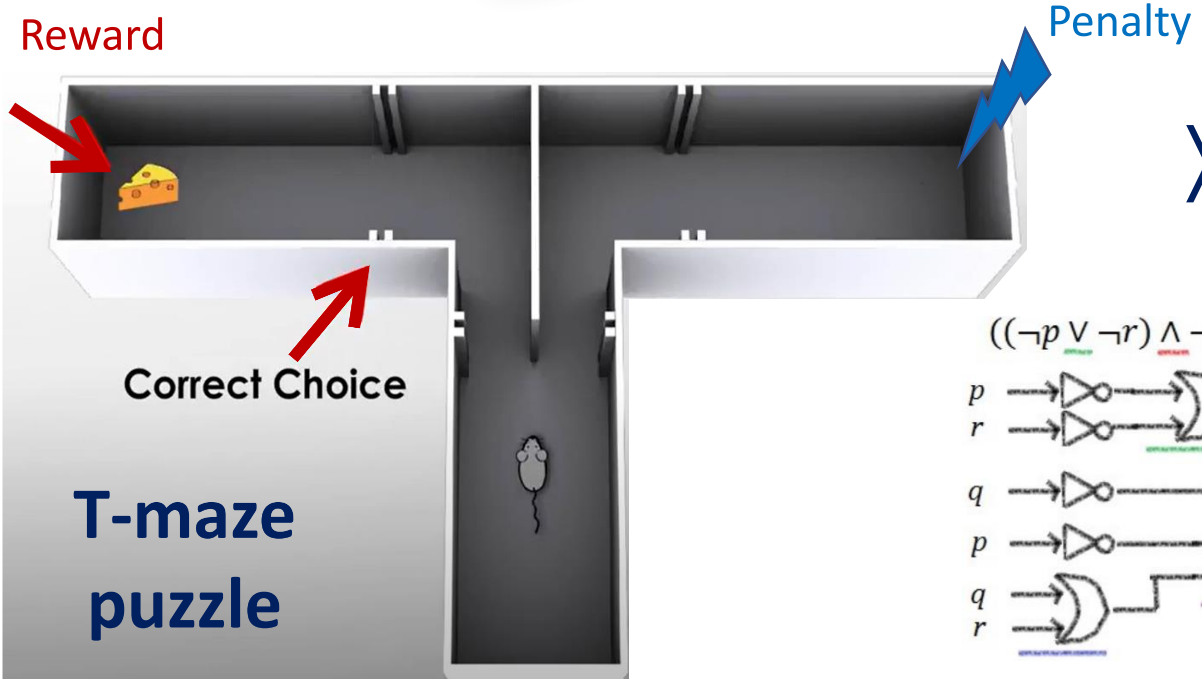


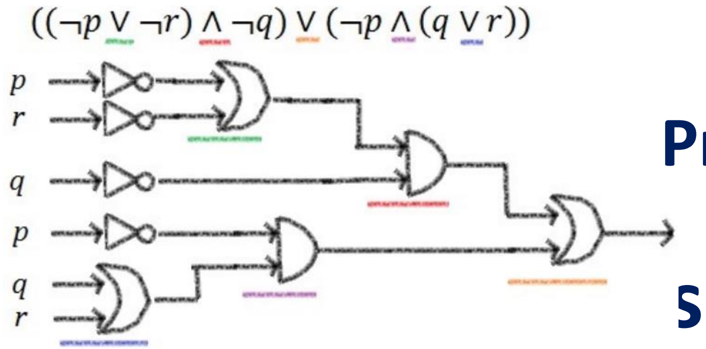
+



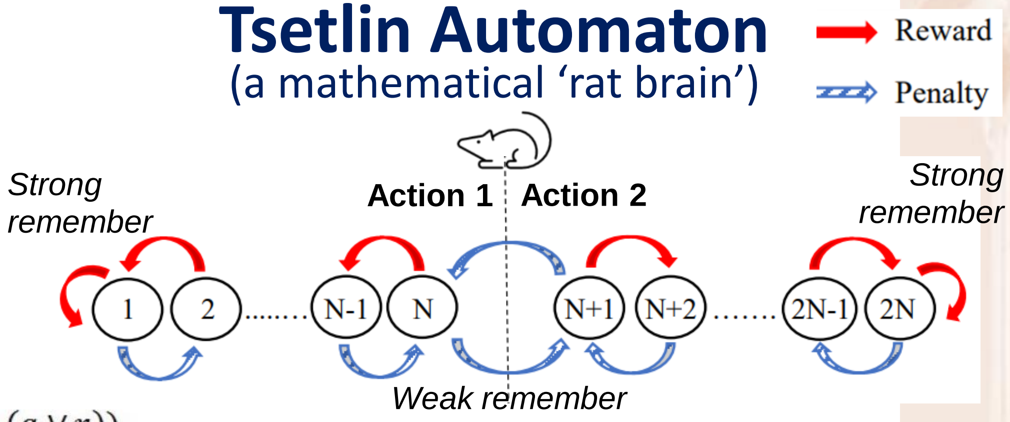

→

- Chrysippus
  a Greek Stoic philosopher (c. 279 – c. 206 BC)
- Prof. Mikhail Tsetlin
  a Soviet mathematician and physicist who worked on cybernetics (22/09/1924 – 30/05/1966)
- Prof. Ole-Christoffer Granmo
  a Norwegian computer scientist, director at the Centre for Artificial Intelligence Research (CAIR) at the University of Agder (born 26/08/1974)




**Tsetlin Automaton**
(a mathematical 'rat brain')




**+**

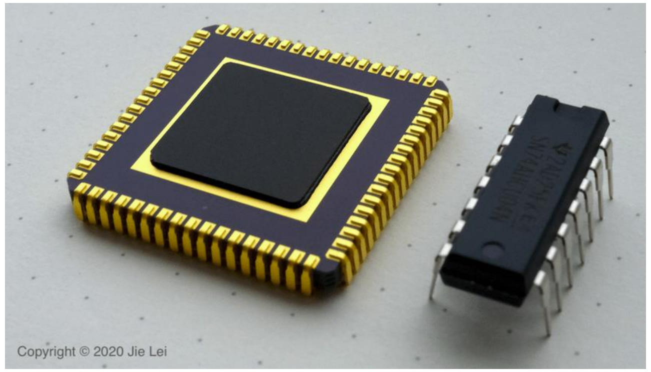
**Propositional Logic**

**=**

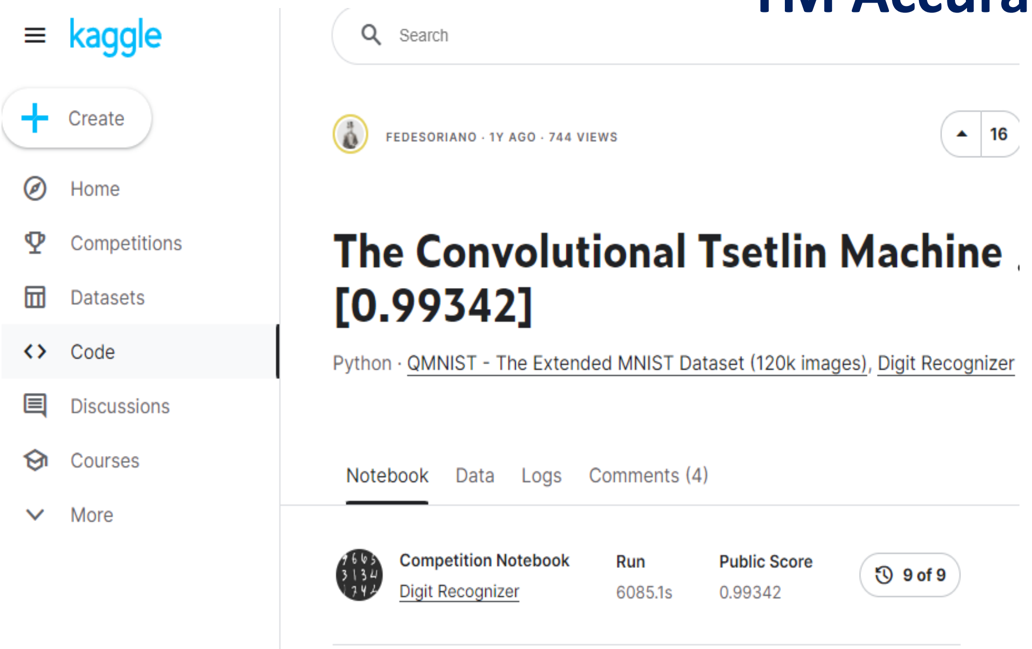
**setlin Machine AI**

# Tsetlin Machine: Features and Benefits

- **Accuracy:** Competitive to ANN
- **Performance:** Nearly hardware (1 learning step ≈ 1 clock cycle)
- **Energy consumption:** 10 TOPs/J, operating at 33MHz
- **Operations:** Logic(AND, NOT), Sum, Voting
- **Operands:** Boolean, Integer
- **Hardware resources:** $2X \times C \times Y$ TAs
- **Explainability:** Yes
- **Highly parallel:** Yes

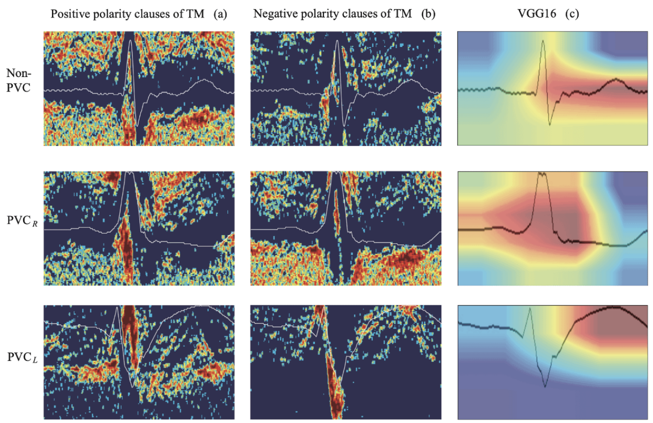



## TM Accuracy:




| Model | 2D N-XOR | MNIST | K-MNIST | F-MNIST |
|---|---|---|---|---|
| 4-Nearest Neighbour [5, 41] | 61.62 | *97.14* | *91.56* | 85.40 |
| SVM [5] | 94.63 | *98.57* | *92.82* | *89.7* |
| Random Forest [22] | 70.73 | *97.3* | - | *81.6* |
| Gradient Boosting Classifier [41] | 87.15 | *96.9* | - | *88.0* |
| Simple CNN [5, 22] | 91.05 | *99.06* | *95.12* | *90.7* |
| BinaryConnect [6] | - | *98.99* | - | - |
| FPGA-accelerated BNN [20] | - | *98.70* | - | - |
| Logistic Circuit (binary) [22] | - | *97.4* | - | *87.6* |
| Logistic Circuit (real-valued) [22] | - | *99.4* | - | *91.3* |
| PreActResNet-18 [5] | - | *99.56* | *97.82* | *92.00* |
| ResNet18 + VGG Ensemble [5] | - | *99.60* | *98.90* | - |
| TM | 99.12 | 98.57 | 92.03 | 90.09 |
| CTM (Mean) | $99.99 \pm 0.0$ | $99.33 \pm 0.0$ | $96.08 \pm 0.01$ | $91.18 \pm 0.01$ |
| CTM (95 %ile) | 100.0 | 99.38 | 96.25 | 91.39 |
| CTM (Peak) | 100.0 | 99.40 | 96.31 | 91.50 |

# Tsetlin Machine: Applications




- **Medical research**
  - ECG classification
  - Patient allergies detection in electronic health records

- **IoT and embedded systems**
  - Intrusions and anomaly detection
  - Drones activity classification

- **Cybersecurity**
  - Detection of network attacks
  - Spam detection
  - Fake news detection

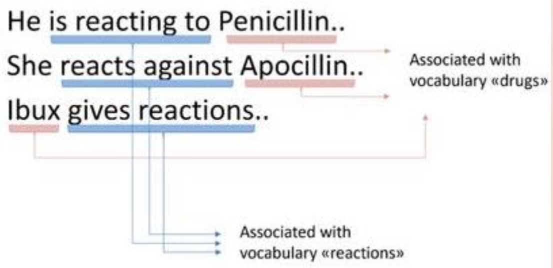
…

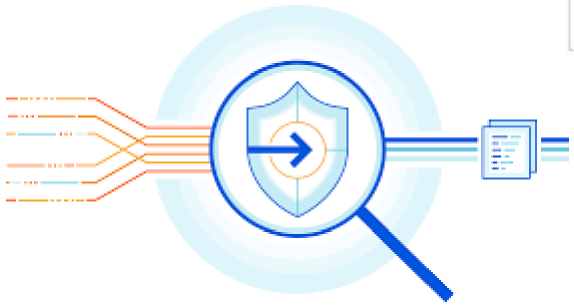
**Interpretability**

**At-the-edge inference**

**Low power**

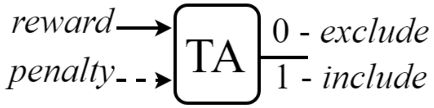
**Noise resilience**

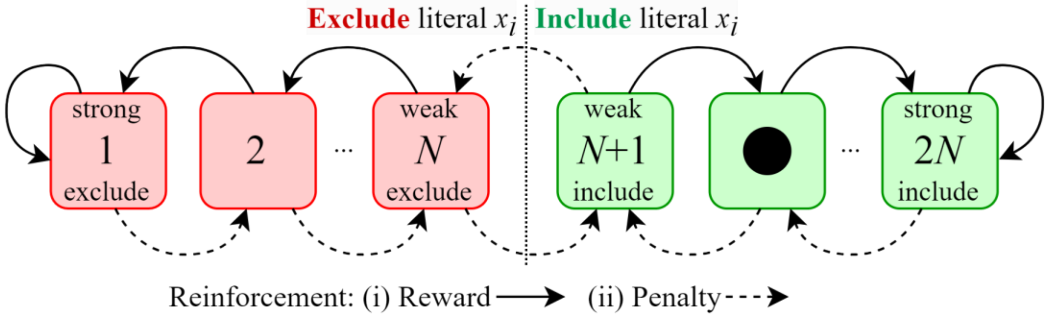



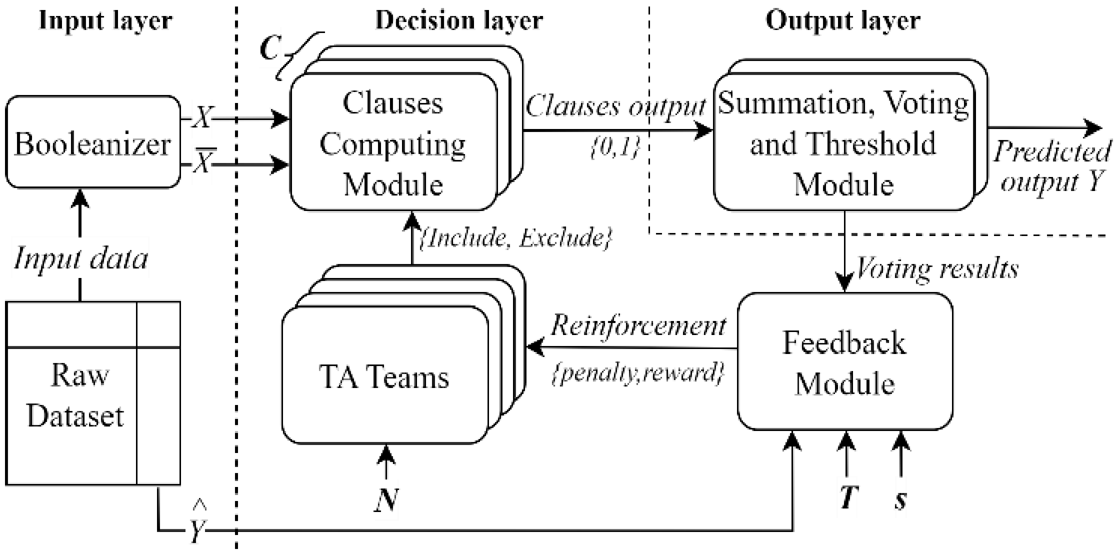
# Tsetlin Machine: Architecture

- **Tsetlin Automaton (TA):**

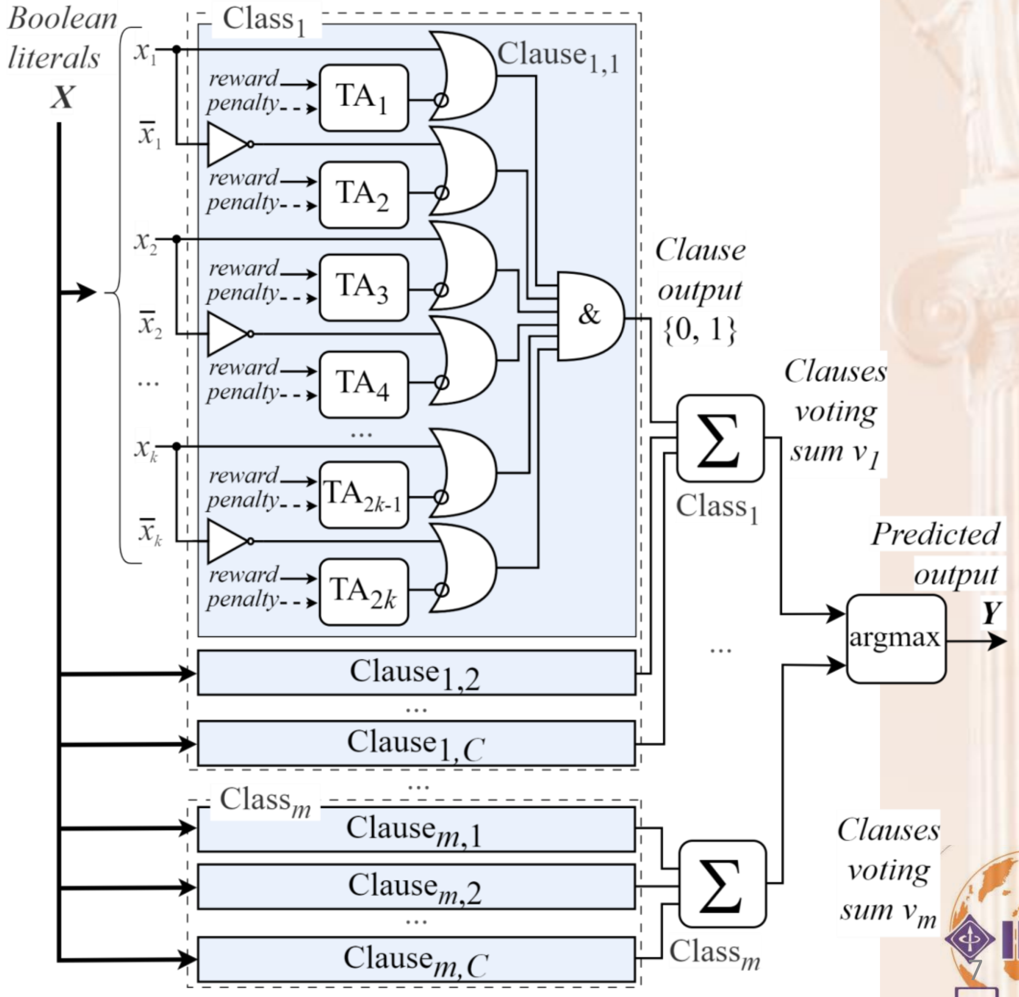



- **TM Architecture:**

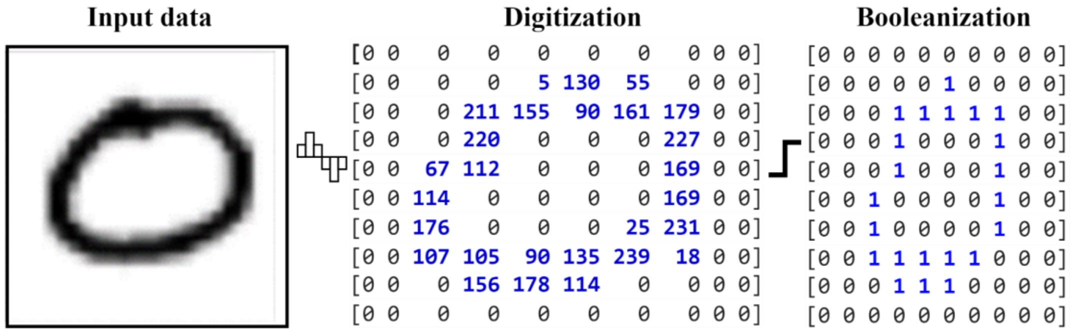



- **Clause Compute Logic:**

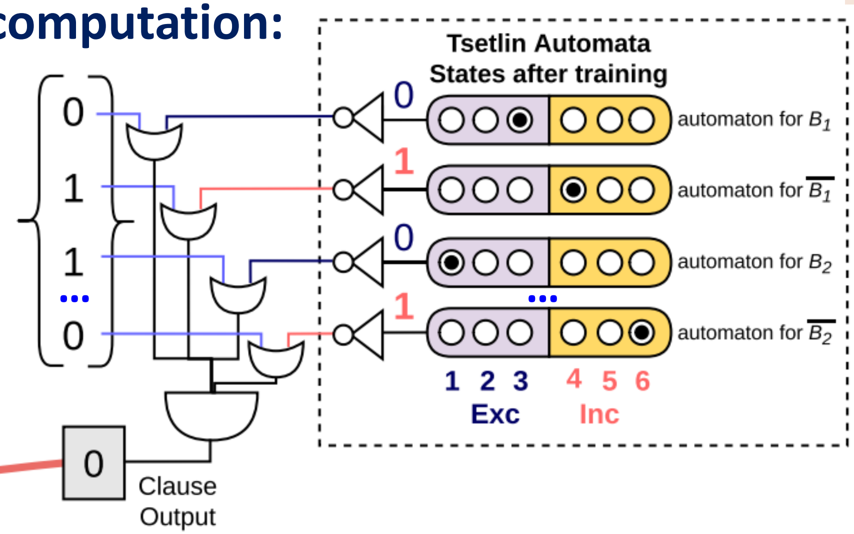


# Tsetlin Machine: Inference

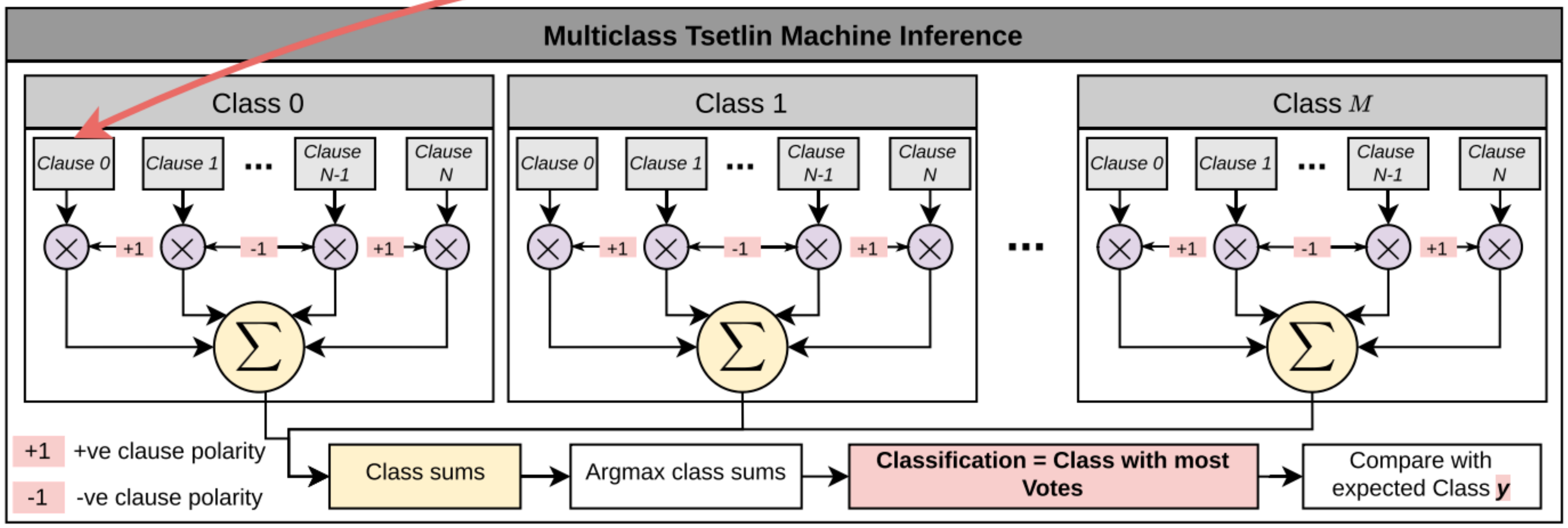
## Data booleanization:



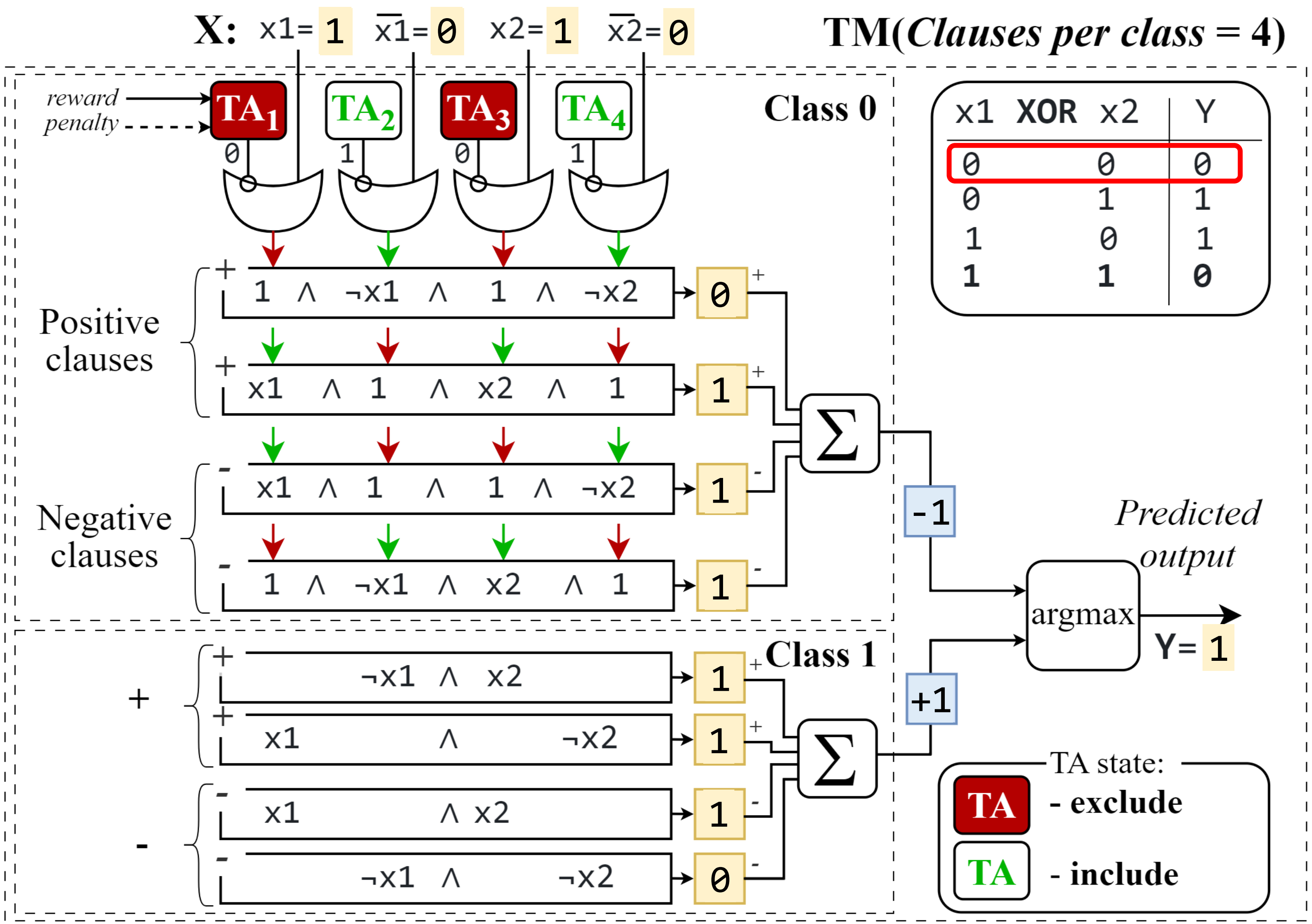

## Clause computation:




## Inference:



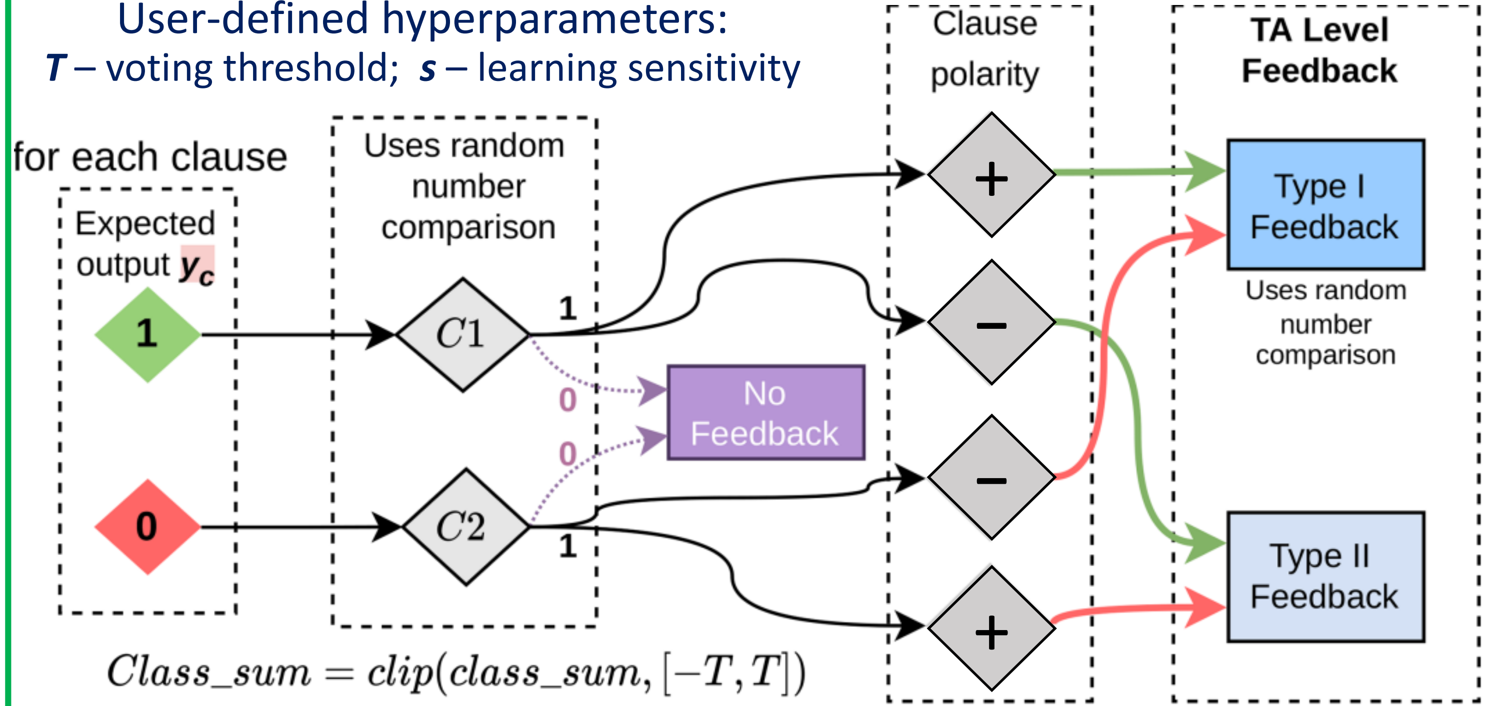
# Tsetlin Machine: Inference Example (XOR)



# Tsetlin Machine: Training and Feedback

## Stochastic Reinforcement Rules:

User-defined hyperparameters:
***T*** – voting threshold; ***s*** – learning sensitivity




$$Class\_sum = clip(class\_sum, [-T, T])$$

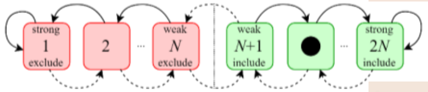
$$C1 = \left(rand() \leq \frac{T - Class\_sum}{2 \times T}\right)$$

$$C2 = \left(rand() \leq \frac{T + Class\_sum}{2 \times T}\right)$$

$$S1 = \left(rand() \leq (\frac{1}{s})\right)$$

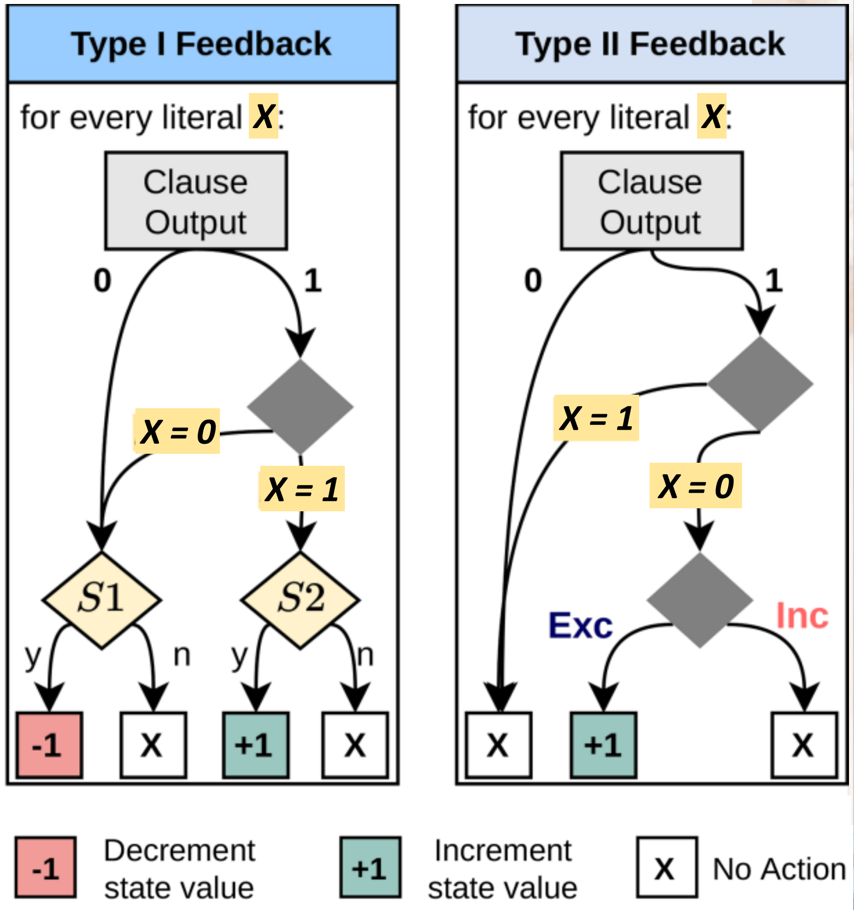
$$S2 = \left(rand() \leq (1 - \frac{1}{s})\right)$$

## TA Feedback:




Combat False Negatives

Combat False Positives



# TM: Natural Explainability (example)

- Task[1]:
  Determining whether a patient is allergic to drugs or not based on EHRs

- Dataset:
  20000 clinical notes belonging to patients who were admitted to the hospital for orthopedic surgical procedures performed between January 1, 2014 and December 31, 2015. Based on the annotated allergy information, each of the patient cases was manually assigned either the category Allergy or the category No Allergy.

- TM:
  - Features: 38 876
  - Clauses per class: 500
  - TA states: 100
  - s-value: 3.0
  - Voting margin, T: 25
  - Epochs: 2

- Accuracy: 79.1%

- Output (Proposition logic clauses)

| Clauses |
|---|
| "Penicillin allergy" |
| "urticaria" |
| "Cortisone" AND "subcutaneous" AND |
| "Ventoline" and "performance ability" AND "work pressure" AND "triggered by effort" AND "grass" |
| "antibodies" AND "allergic" AND "circulatory" |
| "grass" AND "asthmatic" AND "short of breath" AND "itchy skin" AND "symptom picture" AND "Cetirizin" AND "hyperventilating" AND … |

[1]*G. T. Berge et al.*: *Using the Tsetlin Machine to Learn Human-Interpretable Rules for High-Accuracy Text Categorization*, IEEE Access, Vol. 7, pp. 115134-115146, 2019

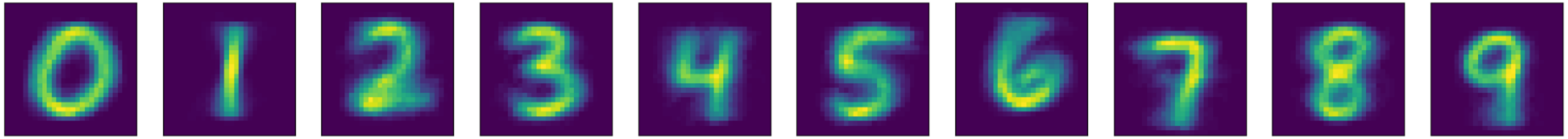
# Tsetlin Machine: Hyperparameters

## DNN vs TM

| DNN | TM | Hyperparameter optimization models |
|---|---|---|
| **Architectural**<br>○ Number of hidden layers<br>○ Number of neurons in each layer<br>○ Drop-outs layers<br>○ Neuron's activation function<br>○ … | ○ Number of clauses per class, ***C***<br>○ Number of TA states, $\boldsymbol{2N=2^n}$ | $\uparrow C \Rightarrow \uparrow Accuracy$<br>$n$=6..8 (good enough) |
| **Algorithmic**<br>○ Optimisation algorithm<br>○ Learning rate<br>○ Loss function<br>○ Regularizati…<br>○ Model…<br>○ … | ○ Voting Margin, ***T***<br>○ Probability of reward/penalty, ***s*** | $T(C) \approx \sqrt{\frac{C}{2}}$ Jagiellonian compromise for qualified majority in the Penrose's square root voting system<br>$s(C) = k1 \cdot \ln(k2 \cdot C) < T(C)$<br>~ Relationship between the crowd density and its velocity (crowd dynamics) |
| **Training**<br>○ Da… …ain, val, test)<br>○ Nu… …er of training epochs<br>○ Batch size<br>○ … | ○ Number of training epochs, ***E*** | $\uparrow E \Rightarrow \uparrow Accuracy,$<br>*no overfitting* |

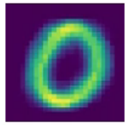
**Major issue** – lacks the hyperparameter optimization models

# Tsetlin Machine: Demo

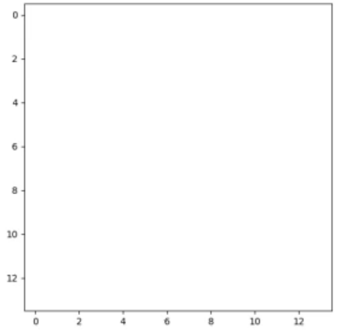


**Handwritten Digits Recognition**
**(MNIST dataset)**

# Clauses voting FOR the certain class:

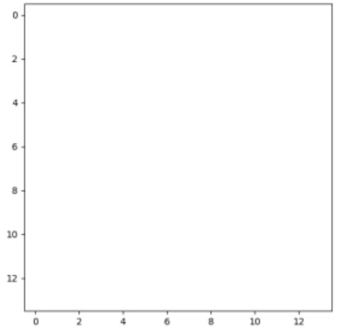


Clauses sum

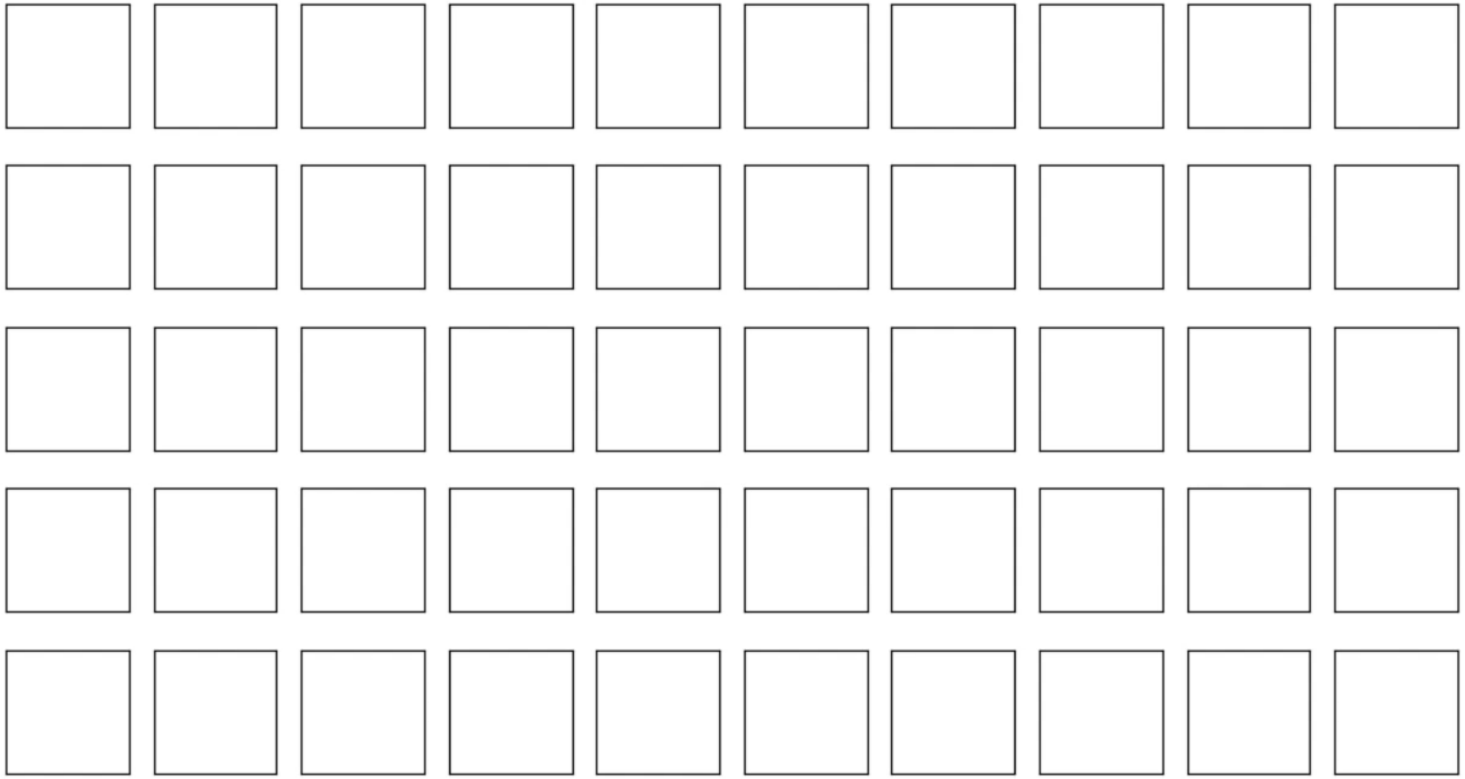



■ include $X_i$

■ include $\overline{X_i}$

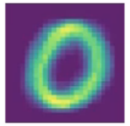


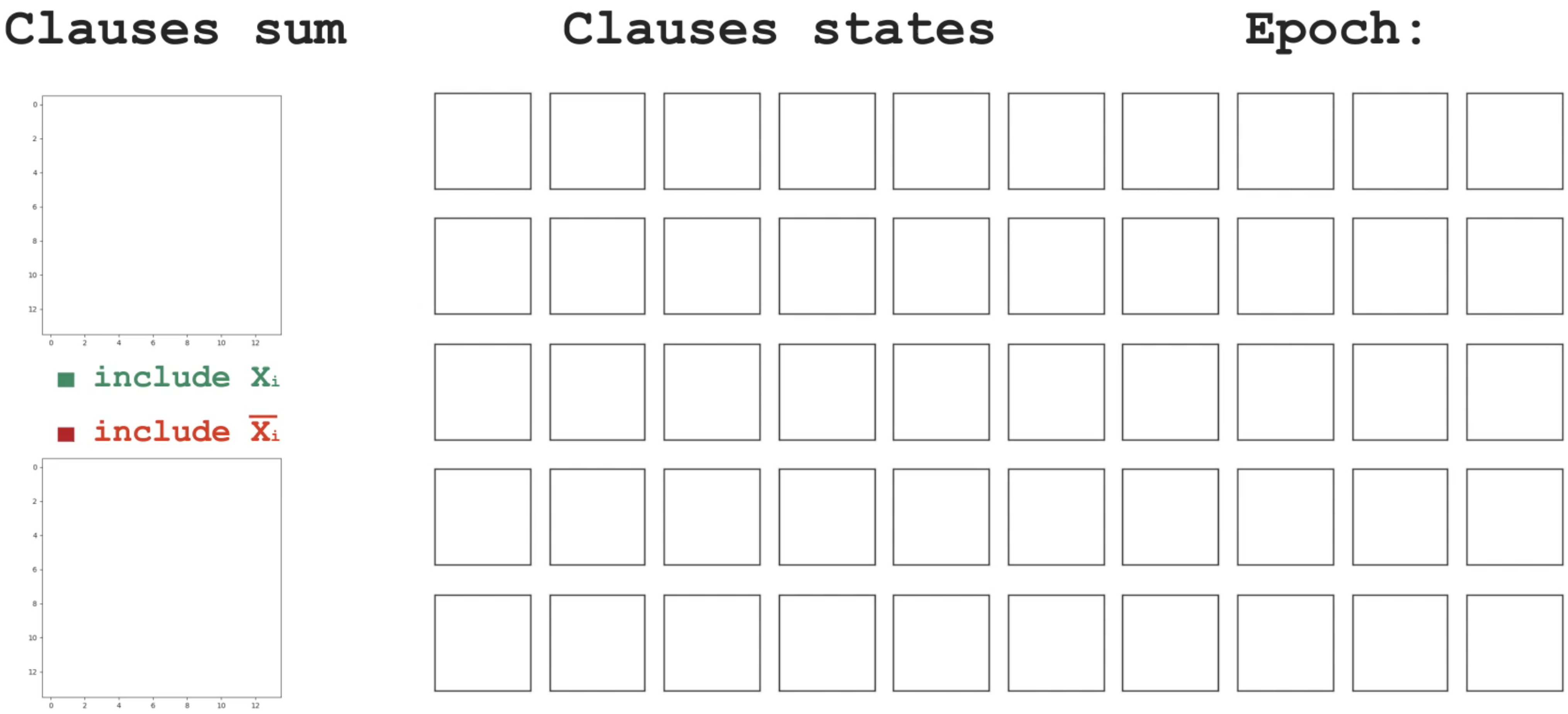

Clauses states



Epoch:

# Clauses voting AGAINST the certain class:





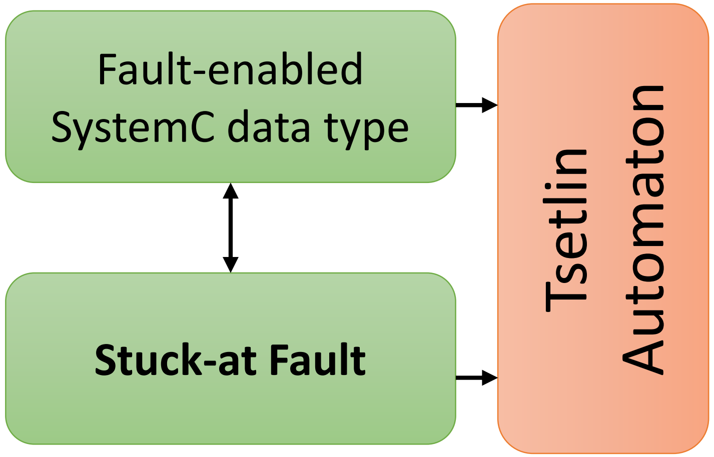
# Learned digit patterns

# TM Dependability: TAs natural fault mitigation

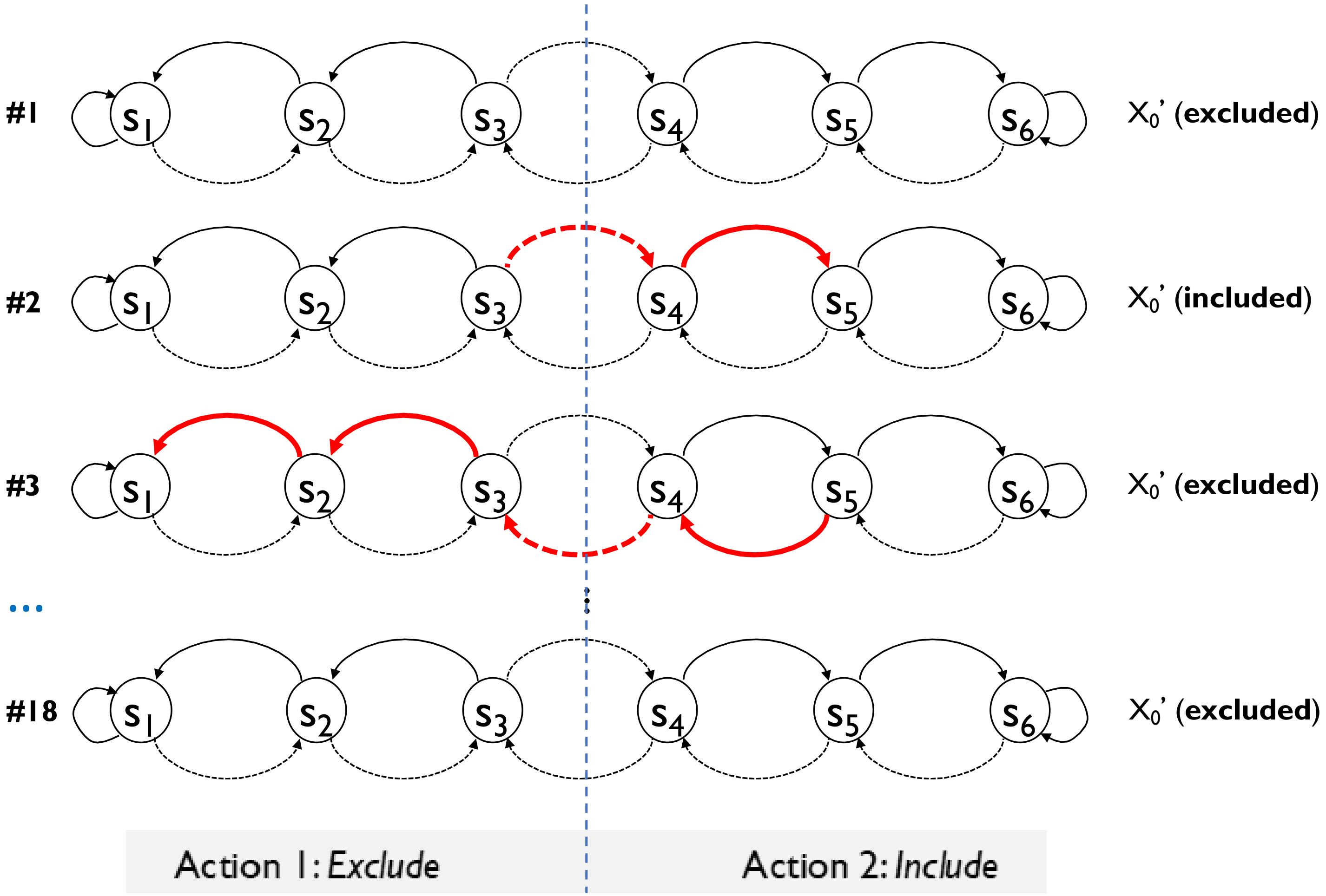
## 2-input XOR example

| $X_0$ | $X_1$ | Y | Y’ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |

## TA’s **stuck-at fault** mitigation implementation

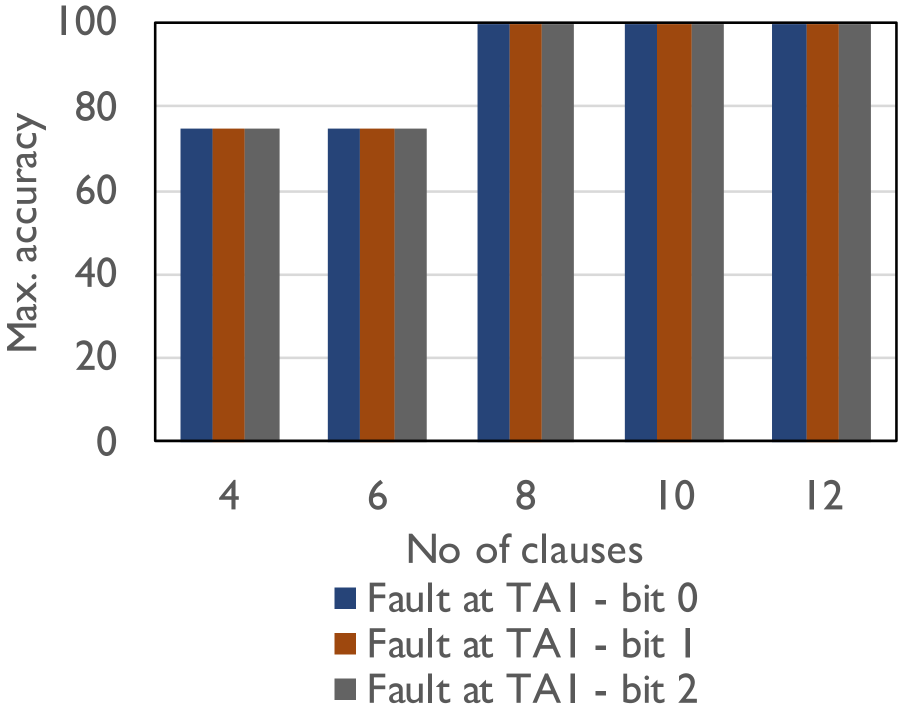



## Fault in TA1: stuck-at 1 at bit 0

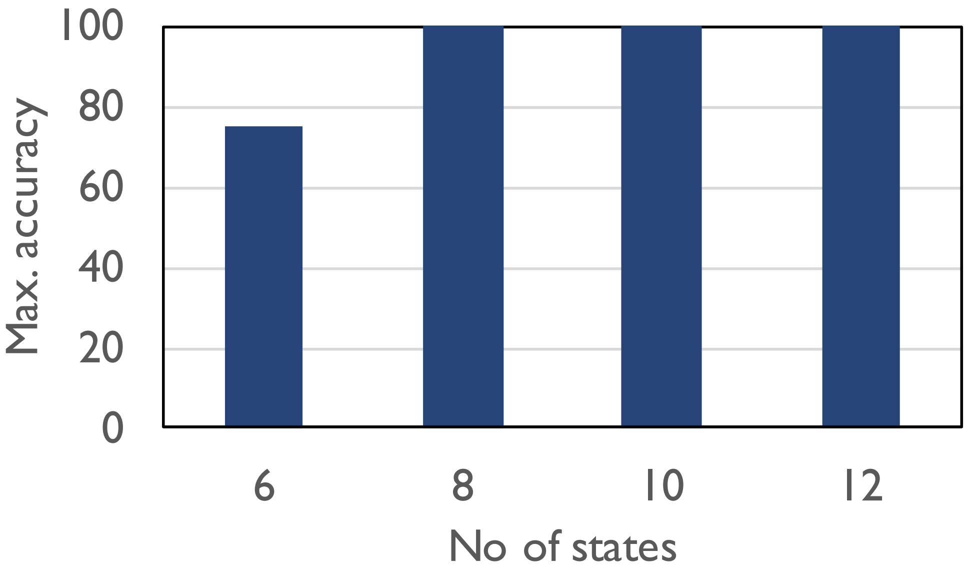


# Mitigating faults by natural redundancy

**More clauses** allow for higher accuracy
(with even fewer no of learning steps)




**More TA states** allow for more accuracy
(with higher no of learning steps)




XOR example

*Yakovlev et al.: IOLTS2020; IBM Neuroscience 2021*

# Summary

- Tsetlin Machine (TM) is a new logic-based machine learning algorithm
- TM originates from the study of animals' ability to learn and simulates collective behaviour of learning automata
- TM provides a higher level of intelligence abstraction compared to ANN - an ensemble of Tsetlin automata ('rat brains'), each making a discrete decision
- TM uses stochastic reinforcement to update learning automata states; its convergence and reproducibility has been proven mathematically
- TM has only 4 hyperparameters to turn which optimum can be found using explainable mathematical models
- TMs provides competitive accuracy and supports natural explainability
- TM can be natively converted to computer hardware with great productivity and energy-efficiency

# Thank you for your attention

Prof Alex Yakovlev
**µSystems**