Investigations of Hierarchical Temporal Memory

Ezra Aylaian

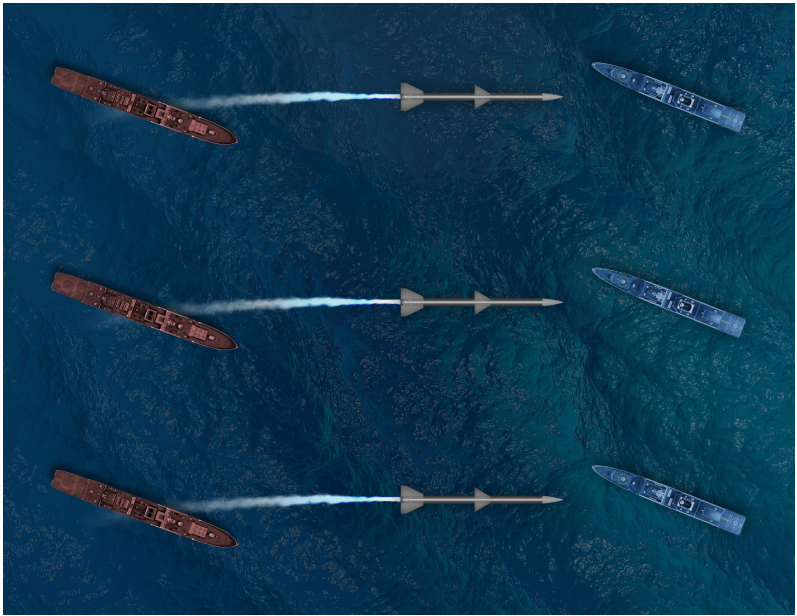
Maritime Force Engagement Control Group
Air and Missile Defense Sector
Johns Hopkins University Applied Physics Laboratory

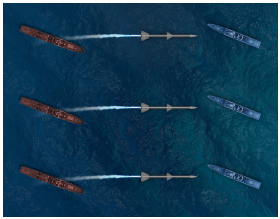
Department of Mathematics
University of Maryland

Joint work with Thomas Corcoran and Samim Manizade

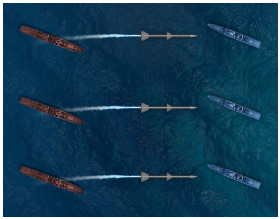
Nebraska Conference for Undergraduate Women in Mathematics
January 21st, 2023

Table of Contents



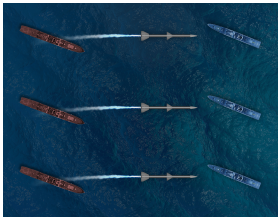


Raid 1: Benign

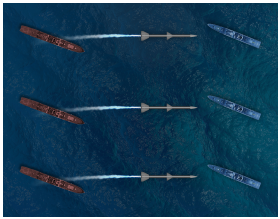


Raid 1: Benign

⋮

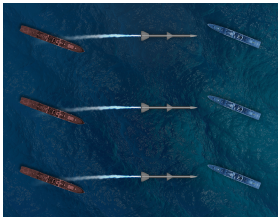


Raid 6: Benign

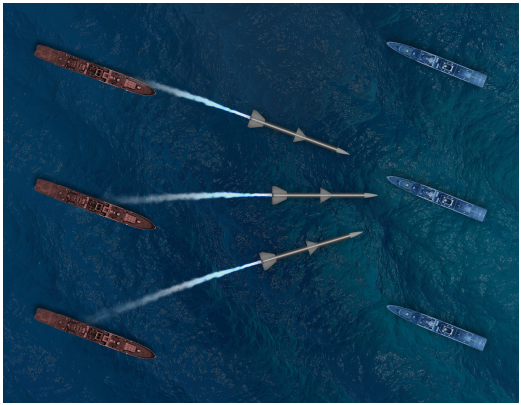


Raid 1: Benign

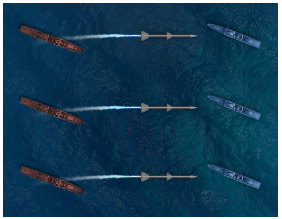
⋮



Raid 6: Benign

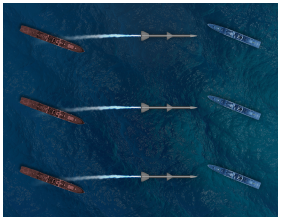


Raid 7: Anomalous

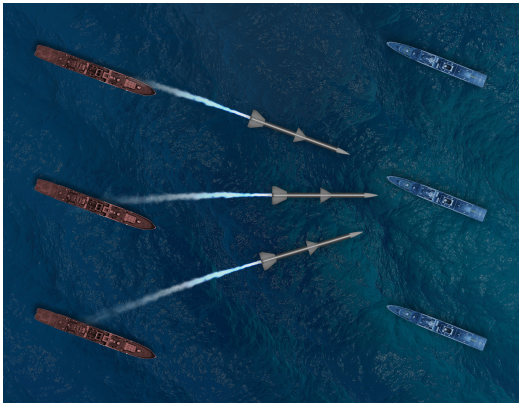


Raid 1: Benign

⋮



Raid 6: Benign



Raid 7: Anomalous

How can we detect a sudden change in tactics?

Table of Contents

Boolean Algebras

Definition: Boolean Algebra

A boolean algebra is a set B along with binary operations $\wedge$ and $\vee$ and unary operation $\neg$ such that:

- $\wedge$ and $\vee$ are associative, commutative, and distribute over each other,
- $a \vee (a \wedge b) = a$ and $a \wedge (a \vee b) = a$,
- $\exists 0, 1 \in B$ such that $a \vee 0 = a$ and $a \wedge 1 = a$,
- $a \vee \neg a = 1$ and $a \wedge \neg a = 0$.

Boolean Algebras

Definition: Boolean Algebra

A boolean algebra is a set B along with binary operations $\wedge$ and $\vee$ and unary operation $\neg$ such that:

- $\wedge$ and $\vee$ are associative, commutative, and distribute over each other,
- $a \vee (a \wedge b) = a$ and $a \wedge (a \vee b) = a$,
- $\exists 0, 1 \in B$ such that $a \vee 0 = a$ and $a \wedge 1 = a$,
- $a \vee \neg a = 1$ and $a \wedge \neg a = 0$.

A homomorphism of Boolean algebras is a map $f : B \to B'$ such that

$$f(a \wedge b) = f(a) \wedge f(b), \quad f(a \vee b) = f(a) \vee f(b), \quad f(0) = 0, \quad f(1) = 1.$$

Boolean algebras with Boolean homomorphisms form a category.

Boolean Algebras

Example

The two-element Boolean algebra has 0 and 1 as its only elements. We interpret 0 as false, 1 as true, and $\wedge$, $\vee$, and $\neg$ as and, or, and not.

Boolean Algebras

Example

The two-element Boolean algebra has 0 and 1 as its only elements. We interpret 0 as false, 1 as true, and $\wedge$, $\vee$, and $\neg$ as and, or, and not.

Definition: Direct Product

Given Boolean algebras $\{B_\alpha\}_{\alpha \in \mathcal{A}}$, we define the direct product $\prod_{\alpha \in \mathcal{A}} B_\alpha$ to be the Boolean algebra with the Cartesian product as the underlying set and with operations defined componentwise.

Boolean Algebras

Example

The two-element Boolean algebra has 0 and 1 as its only elements. We interpret 0 as false, 1 as true, and $\wedge$, $\vee$, and $\neg$ as and, or, and not.

Definition: Direct Product

Given Boolean algebras $\{B_\alpha\}_{\alpha \in \mathcal{A}}$, we define the direct product $\prod_{\alpha \in \mathcal{A}} B_\alpha$ to be the Boolean algebra with the Cartesian product as the underlying set and with operations defined componentwise.

For the rest of the talk, let B be the two-element Boolean algebra. Elements of the direct product $B^n := \prod_{i=1}^{n} B$ can be classified by how many of their components have ones. Specifically, let B^n_w be elements of B^n with w ones, then $B^n = \coprod_{w=0}^{n} B^n_w$.

Boolean Algebras

Example

The two-element Boolean algebra has 0 and 1 as its only elements. We interpret 0 as false, 1 as true, and $\wedge$, $\vee$, and $\neg$ as and, or, and not.

Definition: Direct Product

Given Boolean algebras $\{B_\alpha\}_{\alpha\in\mathcal{A}}$, we define the direct product $\prod_{\alpha\in\mathcal{A}} B_\alpha$ to be the Boolean algebra with the Cartesian product as the underlying set and with operations defined componentwise.

For the rest of the talk, let B be the two-element Boolean algebra. Elements of the direct product $B^n := \prod_{i=1}^n B$ can be classified by how many of their components have ones. Specifically, let B^n_w be elements of B^n with w ones, then $B^n = \coprod_{w=0}^n B^n_w$. In the HTM terminology, if $w \ll n$, then elements of B^n_w are called **Sparse Distributed Representations (SDRs)** of size n and sparsity w/n.

What is Hierarchical Temporal Memory (HTM)?

- HTM is a recent biologically-plausible alternative to neural networks created by Numenta based on a series of conjectures about the structure of the neocortex (the Thousand Brains Theory).

What is Hierarchical Temporal Memory (HTM)?

- HTM is a recent biologically-plausible alternative to neural networks created by Numenta based on a series of conjectures about the structure of the neocortex (the Thousand Brains Theory).
- HTM can learn online, which means that it learns the scenario and generates results in real time with no prior training.

What is Hierarchical Temporal Memory (HTM)?

- HTM is a recent biologically-plausible alternative to neural networks created by Numenta based on a series of conjectures about the structure of the neocortex (the Thousand Brains Theory).
- HTM can learn online, which means that it learns the scenario and generates results in real time with no prior training.

- The **encoder** maps the input space into B^n in a way that preserves semantic structure. The **Spatial Pooler (SP)** learns to represent the encoded SDR at a fixed sparsity while preserving semantic structure. The **Temporal Memory (TM)** learns to predict which components will be ones in the next SP output given the previous SP outputs and gives an anomaly score based on how unaccurate it was.

Table of Contents

Methodology

Question: How should we assess how useful HTM is?

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

1. **Noise resilience:** the ability to operate effectively on noisy data

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

1. **Noise resilience:** the ability to operate effectively on noisy data
2. **Selective attention:** the ability to ignore irrelevant data

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

1. **Noise resilience:** the ability to operate effectively on noisy data
2. **Selective attention:** the ability to ignore irrelevant data
3. **Studiousness:** the utility of the paradigmal particularities of the learning algorithms

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

1. **Noise resilience:** the ability to operate effectively on noisy data
2. **Selective attention:** the ability to ignore irrelevant data
3. **Studiousness:** the utility of the paradigmal particularities of the learning algorithms
4. **Ease-of-use and readiness:** the ability to be easily operationalized

Methodology

Question: How should we assess how useful HTM is?

Properties that we want HTM to have:

1. **Noise resilience:** the ability to operate effectively on noisy data
2. **Selective attention:** the ability to ignore irrelevant data
3. **Studiousness:** the utility of the paradigmal particularities of the learning algorithms
4. **Ease-of-use and readiness:** the ability to be easily operationalized
5. **Experimental plausibility:** the ability to perform well on a simple experiment

Noise Resilience

The ability to operate effectively on noisy data

To test the noise resilience of HTM's spatial pooler, we used well-known MNIST digit classification problem, which contains 60,000 handwritten grayscale 28x28 images of digits for training and 10,000 for testing [4]. Noise was added to the digits as follows:

```
# image is a numpy array, noise_level is an integer
def add_noise(image, noise_level):
        noise = np.rint(np.random.normal(scale=noise_level, size=(28,28)))
        return np.clip(np.absolute(image + noise), 0, 255)
```

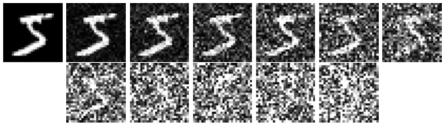
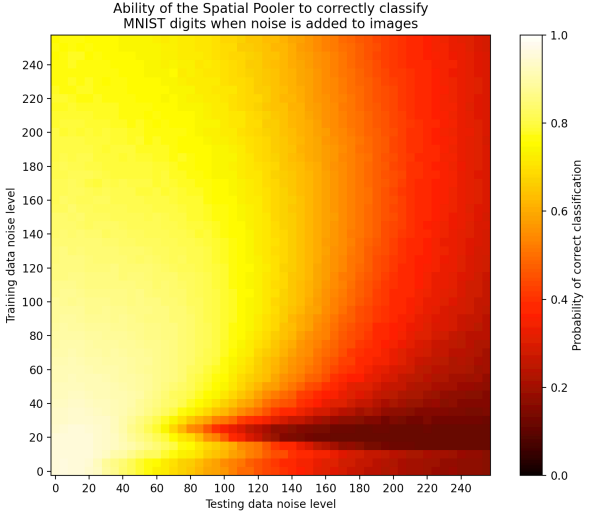


Figure 1: A handwritten MNIST digit with noise levels 0, 20, 40, 60, ..., 220.



Selective Attention

The ability to operate effectively in the presence of irrelevant data

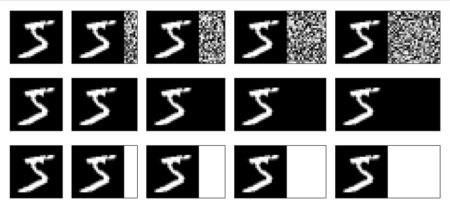


Figure 2: A handwritten MNIST digit with 0, 7, 14, 21, and 28 noisy, black, and white irrelevant columns added.

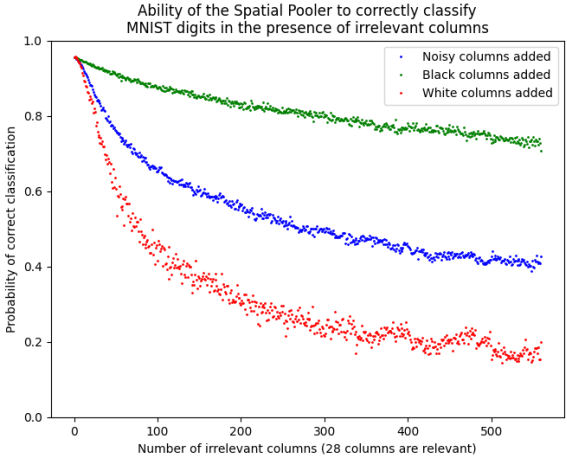
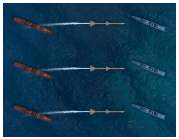
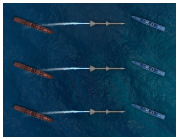


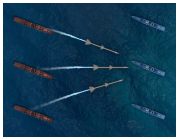
Table of Contents



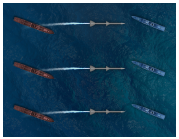
Raid 1: Benign



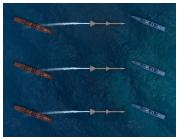
Raid 6: Benign



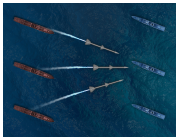
Raid 7: Anomalous



Raid 1: Benign

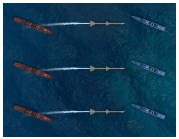


Raid 6: Benign

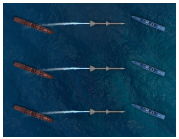


Raid 7: Anomalous

- Each missile's position is encoded as an SDR.



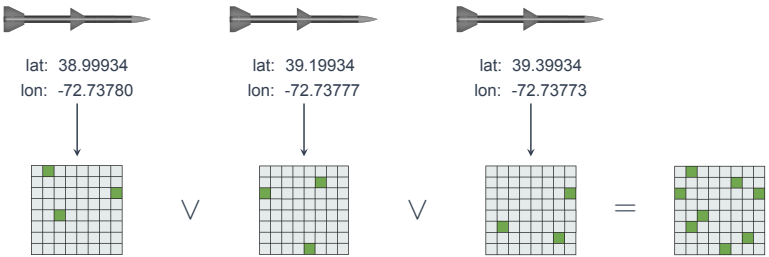
Raid 1: Benign

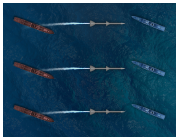


Raid 6: Benign

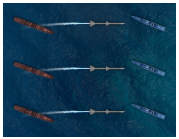
Raid 7: Anomalous

- Each missile's position is encoded as an SDR. The missile SDRs are ∨ed together to get the scenario SDR.





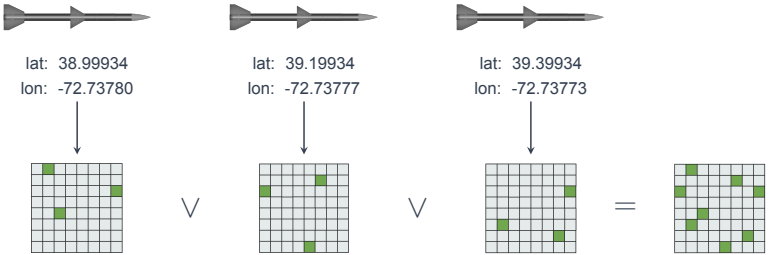
Raid 1: Benign



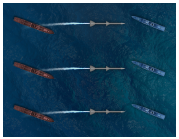
Raid 6: Benign

Raid 7: Anomalous

- Each missile's position is encoded as an SDR. The missile SDRs are ∨ed together to get the scenario SDR.

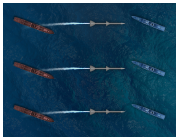


- We run the scenario SDRs through the HTM pipeline to get an anomaly probability associated with each timestep.



Raid 1: Benign

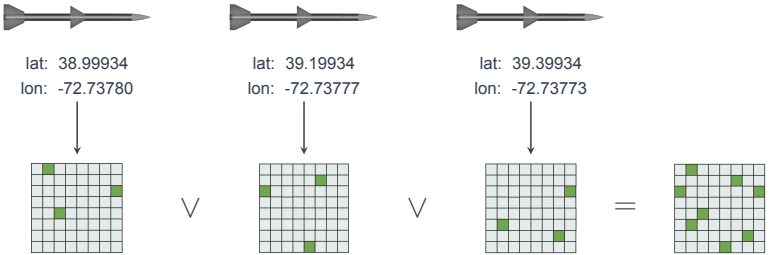
⋮



Raid 6: Benign

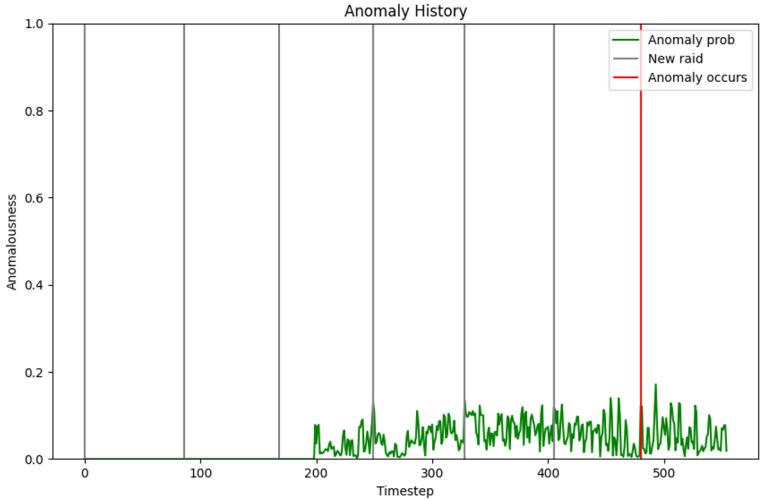
Raid 7: Anomalous

- Each missile's position is encoded as an SDR. The missile SDRs are ∨ed together to get the scenario SDR.

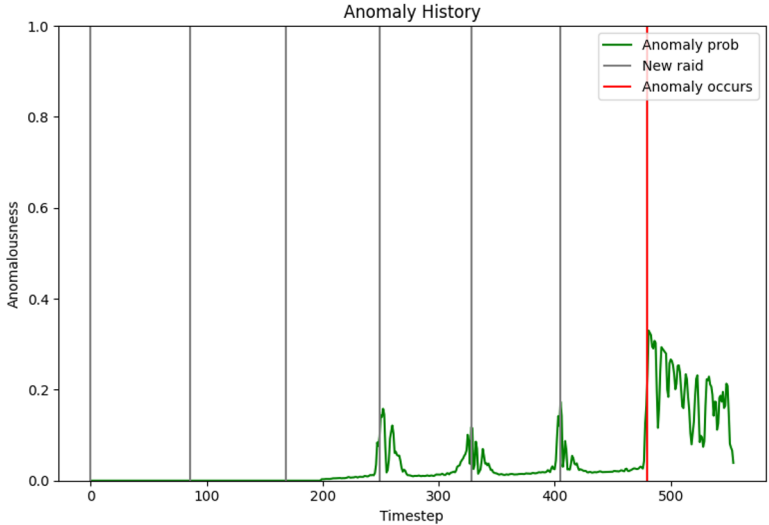


- We run the scenario SDRs through the HTM pipeline to get an anomaly probability associated with each timestep.
- Success is achieved if the anomaly probability spikes when the anomalous raid begins.

Results Without Hyperparameter Optimization



Results With Hyperparameter Optimization



Summary

How useful is HTM?

Summary

How useful is HTM? Is HTM...

1. Noise resilient?

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective?

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious?

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious? Yes, features online learning

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious? Yes, features online learning
4. Ready to operationalize?

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious? Yes, features online learning
4. Ready to operationalize? No, low TRL

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious? Yes, features online learning
4. Ready to operationalize? No, low TRL
5. Experimentally plausible?

Summary

How useful is HTM? Is HTM...

1. Noise resilient? Yes, when trained on noisy data
2. Attention selective? Not great
3. Studious? Yes, features online learning
4. Ready to operationalize? No, low TRL
5. Experimentally plausible? Performed decently well in a very simple scenario

JOHNS HOPKINS

APPLIED PHYSICS LABORATORY