Hierarchical Temporal Memory

Biological And Machine Intelligence

Felix Karg

August 31, 2019

LessWrong Community Weekend 2019

Content

Disclaimer:

Disclaimer: I don't really know what I'm talking about.

Epistemic status

Epistemic status

- Evolving theories

Epistemic status

- Evolving theories
- Hypotheses partially verified

Epistemic status

- Evolving theories
- Hypotheses partially verified
- Theories are constantly being updated

Epistemic status

- Evolving theories
- Hypotheses partially verified
- Theories are constantly being updated
- This is the newest information regarding this theory

Content

Tests for Intelligence

Tests for Intelligence

- Turing test

Tests for Intelligence

- Turing test
- 'IQ' tests

Tests for Intelligence

- Turing test
- 'IQ' tests
- Problme solving tests

Tests for Intelligence

- Turing test
- 'IQ' tests
- Problme solving tests
- Car driving skills

Tests for Intelligence

- Turing test
- 'IQ' tests
- Problme solving tests
- Car driving skills
- ...

Tests for Intelligence

- Turing test
- 'IQ' tests
- Problme solving tests
- Car driving skills
- ...

But dogs, monkeys and dolphins fail them.

Tests for Intelligence

- Turing test
- 'IQ' tests
- Problme solving tests
- Car driving skills
- ...

But dogs, monkeys and dolphins fail them.

Focusing on human-like performance is **limiting.**

Intelligence - Definition

Intelligence - Definition

Intelligence: The degree of flexibility in both learning and behaviour [1].

Intelligence - Overview

Intelligence - Overview

Might not be best at specific task.

Intelligence - Overview

Might not be best at specific task.

But can do a lot of different tasks quite well.

Intelligence - Overview

Might not be best at specific task.

But can do a lot of different tasks quite well.

$\rightarrow$ General solution.

Content

The Human Brain in Numbers

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)
Neurons in cerebellum	69 billion (80%)

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)
Neurons in cerebellum	69 billion (80%)
Rel. size of cerebellum	10% of brain

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)
Neurons in cerebellum	69 billion (80%)
Rel. size of cerebellum	10% of brain
Neurons in celebral cortex	16 billion (19%)

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)
Neurons in cerebellum	69 billion (80%)
Rel. size of cerebellum	10% of brain
Neurons in celebral cortex	16 billion (19%)
Rel. size of cerebral cortex	82% of brain

The Human Brain in Numbers

Neurons in brain (total)	86 billion (100%)
Neurons in cerebellum	69 billion (80%)
Rel. size of cerebellum	10% of brain
Neurons in celebral cortex	16 billion (19%)
Rel. size of cerebral cortex	82% of brain
Neurons in brain stem	1 billion (1%)

Data from [2].

The Human Brain

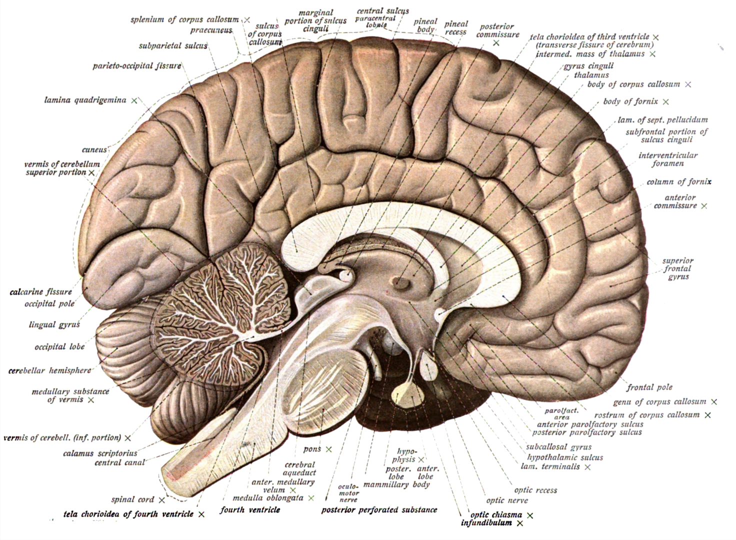
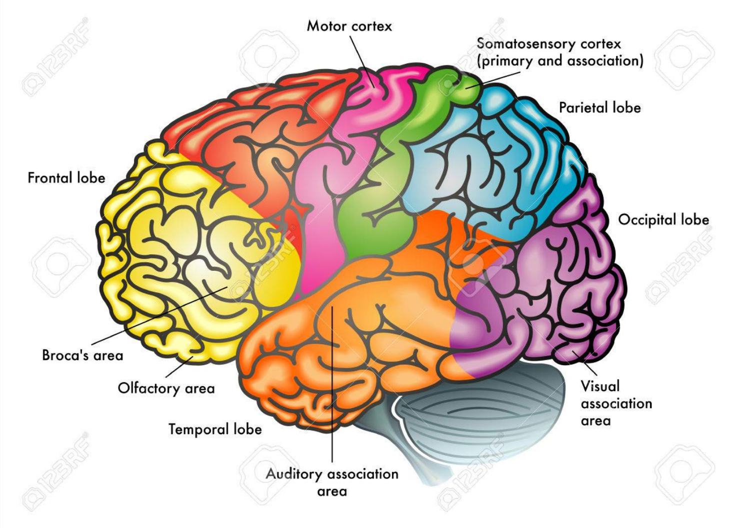


Image from [3].

The Human Brain - Different Areas



Roberto Biasini © 123RF.com

Cortical Column

“There is nothing visual about the visual cortex, and nothing auditory about the auditory cortex“

- Vernon Mountcastle

Cortical Column

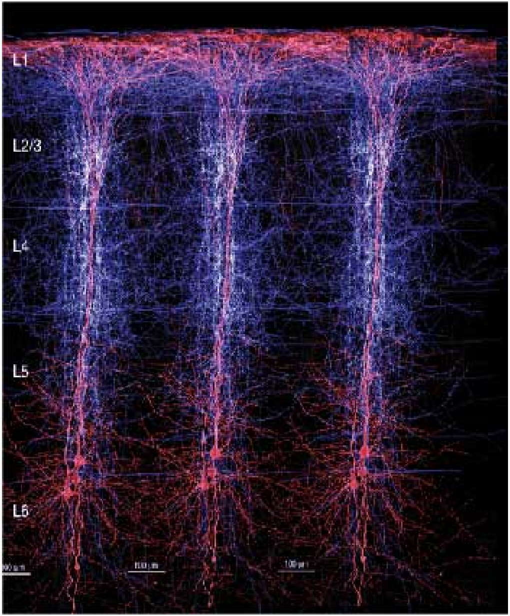


Image from [4].

Cortical Column

- Everywhere in the Brain

Cortical Column

- Everywhere in the Brain
- 80-120 up to 200-400 Neurons

Cortical Column

- Everywhere in the Brain
- 80-120 up to 200-400 Neurons
- Smallest symbol unit

Cortical Column

- Everywhere in the Brain
- 80-120 up to 200-400 Neurons
- Smallest symbol unit
- Activity has meaning

Neuron - Number of Connections

Neuron - Number of Connections

Min. n. of connections	1’000
Avg. n. of connections	7’000
Max. n. of connections	10’000

Neuron - Number of Connections

Min. n. of connections	1'000
Avg. n. of connections	7'000
Max. n. of connections	10'000
Firing Rate	20-250 Hz (453 Hz [5])

Connection data from [2] and firing rate from [6].

Neuron - Spike Frequencies

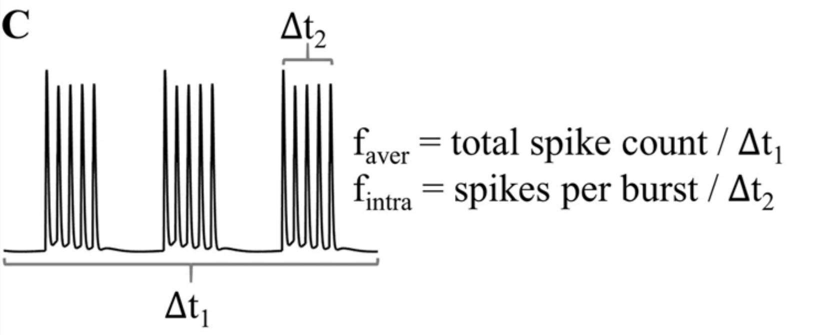


Image adapted from [7].

Neuron - Overview

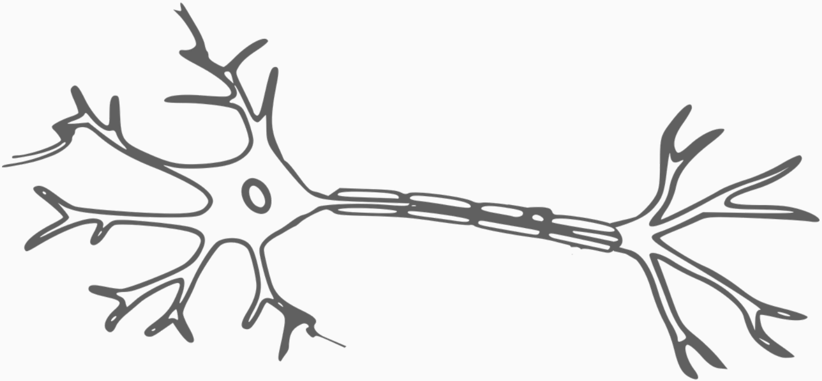


Image from [8].

Content

What is HTM?

What is HTM?

- Biologically constrained **Theory of Intelligence**

What is HTM?

- Biologically constrained **Theory of Intelligence**
- First described in "On Intelligence"

What is HTM?

- Biologically constrained **Theory of Intelligence**
- First described in "On Intelligence"
- **Based on Neuroscience** of the brain

What is HTM?

- Biologically constrained **Theory of Intelligence**
- First described in "On Intelligence"
- **Based on Neuroscience** of the brain

$\rightarrow$ Learning Algorithms

What is HTM?

- Biologically constrained **Theory of Intelligence**
- First described in "On Intelligence"
- **Based on Neuroscience** of the brain

→ Learning Algorithms (of the brain)

Not Included in HTM

Not Included in HTM

- Firing rythms

Not Included in HTM

- Firing rythms
- Emotions

Not Included in HTM

- Firing rythms
- Emotions
- Basic Behaviours

Not Included in HTM

- Firing rythms
- Emotions
- Basic Behaviours
- Sleep (yet)

The brain as Prediction Machine

The brain as Prediction Machine

- Prediction of future sensory input

The brain as Prediction Machine

- Prediction of future sensory input
- 'Anticipating' events

The brain as Prediction Machine

- Prediction of future sensory input
- 'Anticipating' events
- multiple connected regions

The brain as Prediction Machine

- Prediction of future sensory input
- 'Anticipating' events
- multiple connected regions
- Invariant representations

The brain as Prediction Machine

- Prediction of future sensory input
- 'Anticipating' events
- multiple connected regions
- Invariant representations
- Hierarchies of Concepts

The brain as Prediction Machine

- Prediction of future sensory input
- 'Anticipating' events
- multiple connected regions
- Invariant representations
- Hierarchies of Concepts
- A sense of location

Attributes of HTM Algorithms

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams
- robust against noise

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams
- robust against noise
- can learn multiple patterns at once

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams
- robust against noise
- can learn multiple patterns at once
- suited for prediction, anomaly detection, classification

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams
- robust against noise
- can learn multiple patterns at once
- suited for prediction, anomaly detection, classification
- tested and implemented in software

Attributes of HTM Algorithms

- can store, learn, infer and recall higher-order sequences
- learns unsupervised time-based patterns in unlabeled data on continuous streams
- robust against noise
- can learn multiple patterns at once
- suited for prediction, anomaly detection, classification
- tested and implemented in software
- commercially used (anomaly detection, NLP)

The role of Time

Crucial for learning, inference and prediction.

The role of Time

Crucial for learning, inference and prediction.

- Inference is hard on static information

The role of Time

Crucial for learning, inference and prediction.

- Inference is hard on static information
- Predictions are somewhat inherently time-based

The role of Time

Crucial for learning, inference and prediction.

- Inference is hard on static information
- Predictions are somewhat inherently time-based
- Learning is hard without feedback

Content

Content

Why Hierarchy?

Why Hierarchy?

If there is a connection cost, hierarchies are more efficient [9].

Why Hierarchy?

If there is a connection cost, hierarchies are more efficient [9].

Especially when tasks change regularly.

Why Hierarchy? II

Why Hierarchy? II

- Reduced Training Time

Why Hierarchy? II

- Reduced Training Time
- Reduced Memory Usage

Why Hierarchy? II

- Reduced Training Time
- Reduced Memory Usage
- Introduce Generalizations

Why Hierarchy? II

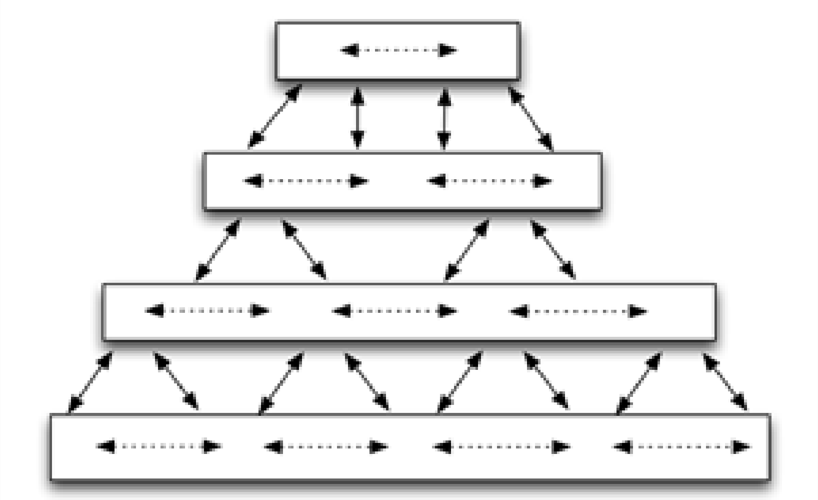
- Reduced Training Time
- Reduced Memory Usage
- Introduce Generalizations
- Learned patterns are recombined at higher levels

Why Hierarchy? II

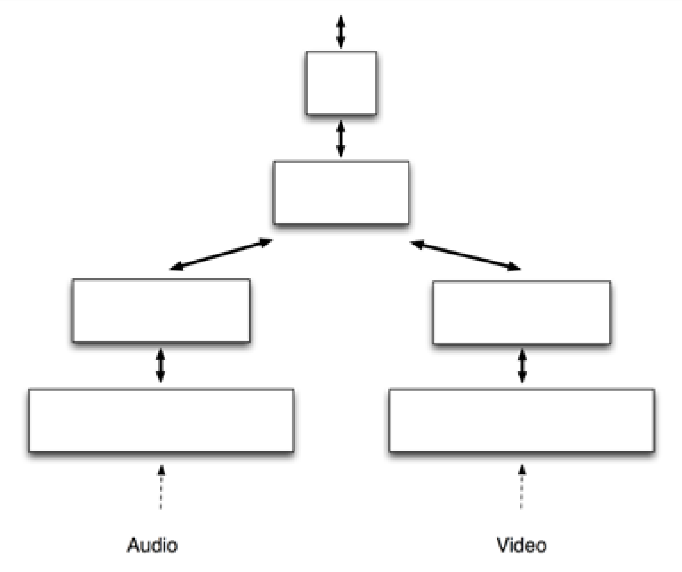
- Reduced Training Time
- Reduced Memory Usage
- Introduce Generalizations
- Learned patterns are recombined at higher levels
- Transfer Learning

What Hierarchy

What Hierarchy



Example Application



How Many Levels?

How Many Levels?

- They always learn the best representation

How Many Levels?

- They always learn the best representation
- Tradeoff between depth and layer size

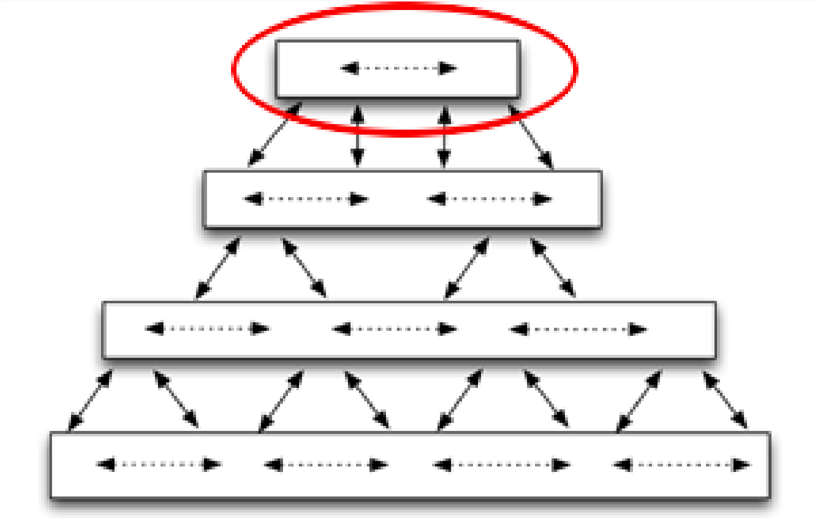
How Many Levels?

- They always learn the best representation
- Tradeoff between depth and layer size
- Simple problems can be solved with one region

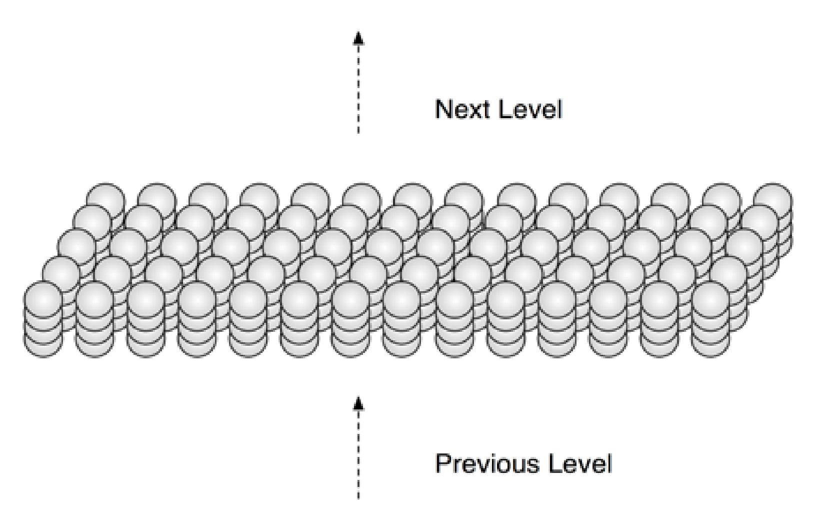
Content

Region - Introduction

Region - Introduction



Region - Details



Region - Attributes

Region - Attributes

- All Regions do basically the same

Region - Attributes

- All Regions do basically the same
- Based on Biological Regions in the Brain

Region - Attributes

- All Regions do basically the same
- Based on Biological Regions in the Brain
- HTM Regions are similar to Layer 3 of the Neocortex

Region - Attributes

- All Regions do basically the same
- Based on Biological Regions in the Brain
- HTM Regions are similar to Layer 3 of the Neocortex
- Can do Inference and Prediction even on complex data

Content

Data Saving - Computer Science Solution

What is 01100101?

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans
  (`False, True, True, False,...`)

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans
  `(False, True, True, False,...)`
- Integer (`101`)

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans
  `(False, True, True, False,...)`
- Integer (`101`)
- Float (`3328.0`)

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans
  (`False, True, True, False,...`)
- Integer (`101`)
- Float (`3328.0`)
- (Byte-) String (`'e'`)

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans (`False, True, True, False,...`)
- Integer (`101`)
- Float (`3328.0`)
- (Byte-) String (`'e'`)
- Pointer to something else

Data Saving - Computer Science Solution

What is `01100101`? Could be either one of:

- Booleans (`False, True, True, False,...`)
- Integer (`101`)
- Float (`3328.0`)
- (Byte-) String (`'e'`)
- Pointer to something else
- Part of some other Datastructure

Biological observation:

We use only part of our brain!

Sparse Distributed Representation - Introduction

Sparse Distributed Representation - Introduction

- Datastructure of the brain

Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)

Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)
- Distributed (clusters are somewhat rare)

Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)
- Distributed (clusters are somewhat rare)
- Inhibitory mechanisms

Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)
- Distributed (clusters are somewhat rare)
- Inhibitory mechanisms
- Neuron states actually have 'meaning'

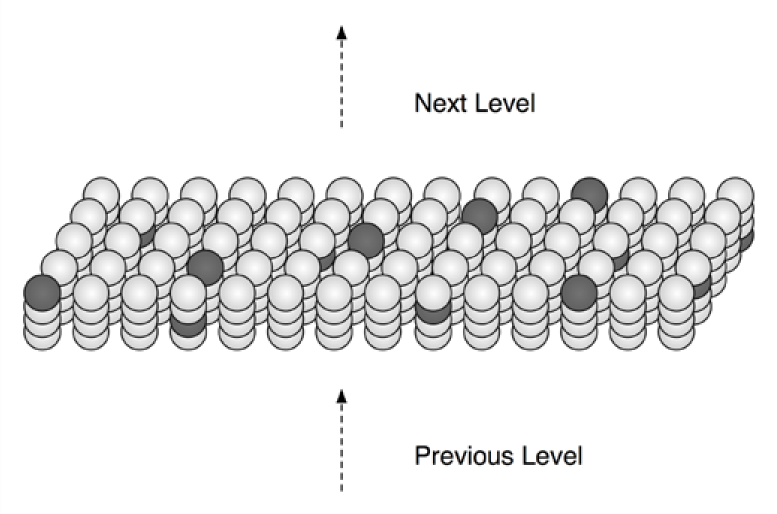
Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)
- Distributed (clusters are somewhat rare)
- Inhibitory mechanisms
- Neuron states actually have 'meaning'
- Combined, they give context as well

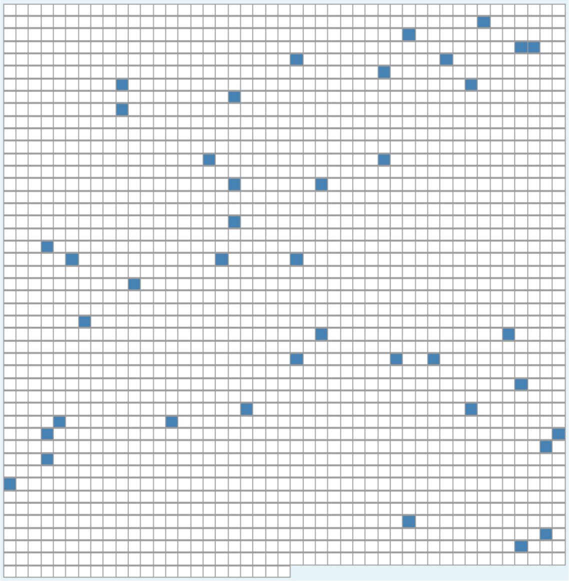
Sparse Distributed Representation - Introduction

- Datastructure of the brain
- Sparse (around 2% are active)
- Distributed (clusters are somewhat rare)
- Inhibitory mechanisms
- Neuron states actually have 'meaning'
- Combined, they give context as well
- Many mechanisms in the brain would not work otherwise

Sparse Distributed Representation - Example



Sparse Distributed Representation - Example



Live Demo!

Sparse Distributed Representation - Live Demos

Sparse Distributed Representation - Live Demos

- Ep2/Capacity

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling
- Ep4/Classification

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling
- Ep4/Classification
- Ep4/Union

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling
- Ep4/Classification
- Ep4/Union
- Ep5/Scalar Encoding

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling
- Ep4/Classification
- Ep4/Union
- Ep5/Scalar Encoding
- Ep6/Date Encoding

Sparse Distributed Representation - Live Demos

- Ep2/Capacity
- Ep2/Matching (Noise resistency)
- Ep3/Subsampling
- Ep4/Classification
- Ep4/Union
- Ep5/Scalar Encoding
- Ep6/Date Encoding
- Ep5/RDSE - Number Encoding

Encoders - Conclusion

Encoders - Conclusion

- Semantically similar data should result in SDRs with overlapping active bits.

Encoders - Conclusion

- Semantically similar data should result in SDRs with overlapping active bits.
- The same input should always produce the same SDR as output.

Encoders - Conclusion

- Semantically similar data should result in SDRs with overlapping active bits.
- The same input should always produce the same SDR as output.
- The output should have the same dimensionality (total number of bits) for all inputs.

Encoders - Conclusion

- Semantically similar data should result in SDRs with overlapping active bits.
- The same input should always produce the same SDR as output.
- The output should have the same dimensionality (total number of bits) for all inputs.
- The output should have similar sparsity for all inputs and have enough one-bits to handle noise and subsampling.

Encoders - Conclusion

- Semantically similar data should result in SDRs with overlapping active bits.
- The same input should always produce the same SDR as output.
- The output should have the same dimensionality (total number of bits) for all inputs.
- The output should have similar sparsity for all inputs and have enough one-bits to handle noise and subsampling.

Cited from [1].

HTM - Pipeline

Image adapted from [10].

HTM - Pipeline

Image adapted from [10].

- Data

HTM - Pipeline

Image adapted from [10].

- Data
- SDR Encoded Data

HTM - Pipeline

Image adapted from [10].

- Data
- SDR Encoded Data
- Spatial Pooler

HTM - Pipeline

Image adapted from [10].

- Data
- SDR Encoded Data
- Spatial Pooler
- Temporal Pooler

HTM - Pipeline

Image adapted from [10].

- Data
- SDR Encoded Data
- Spatial Pooler
- Temporal Pooler
- Classifier

HTM - Pipeline

Image adapted from [10].

- Data
- SDR Encoded Data
- Spatial Pooler
- Temporal Pooler
- Classifier
- Prediction/Classification

Content

Content

Learning

Learning

- Learning is purely statistical

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns
- Regions themselves are limited

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns
- Regions themselves are limited
- Automatically adjusts to size of allocated Memory

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns
- Regions themselves are limited
- Automatically adjusts to size of allocated Memory
- Automatic On-Line learning

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns
- Regions themselves are limited
- Automatically adjusts to size of allocated Memory
- Automatic On-Line learning
- Takes longer to learn high-level concepts with lower levels missing

Learning

- Learning is purely statistical
- Looking for Spatial and Temporal Patterns
- Regions themselves are limited
- Automatically adjusts to size of allocated Memory
- Automatic On-Line learning
- Takes longer to learn high-level concepts with lower levels missing
- Only a precursor for inference and prediction

Inference

Inference

- Matching previously learned sequences

Inference

- Matching previously learned sequences
- Example: recognizing a melody

Inference

- Matching previously learned sequences
- Example: recognizing a melody
- There are only novel experiences

Inference

- Matching previously learned sequences
- Example: recognizing a melody
- There are only novel experiences
- Partial SDR matches suffice

Prediction

Prediction

- Matching stored sequences

Prediction

- Matching stored sequences
- Can be thought of to be similar to a markov chain

Prediction

- Matching stored sequences
- Can be thought of to be similar to a markov chain
- Takes up a considerable amount of memory

Prediction

- Matching stored sequences
- Can be thought of to be similar to a markov chain
- Takes up a considerable amount of memory
- Integral to how the brain works

Prediction - Key Properties

Prediction - Key Properties

- Continuity

Prediction - Key Properties

- Continuity
- Occurs everywhere

Prediction - Key Properties

- Continuity
- Occurs everywhere
- Context sensitivity

Prediction - Key Properties

- Continuity
- Occurs everywhere
- Context sensitivity
- Stability

Prediction - Key Properties

- Continuity
- Occurs everywhere
- Context sensitivity
- Stability
- Anomaly Detection

Prediction - Key Properties

- Continuity
- Occurs everywhere
- Context sensitivity
- Stability
- Anomaly Detection
- Noise robustness

Content

Spatial Pooler - Introduction

Spatial Pooler - Introduction

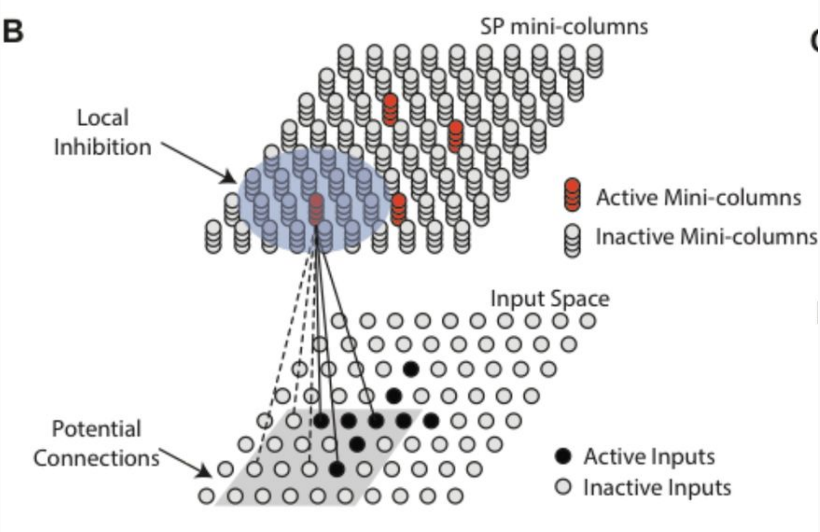


Image adapted from [10].

Spatial Pooler - Connection details

Show `Ep8/Learning Rules`!

Spatial Pooler - Connection details

Show `Ep8/Learning Rules`!

- Many Connections

Spatial Pooler - Connection details

Show `Ep8/Learning Rules!`

- Many Connections
- Only Columns with highest overlap scores continue

Spatial Pooler - Connection details

Show `Ep8/Learning Rules!`

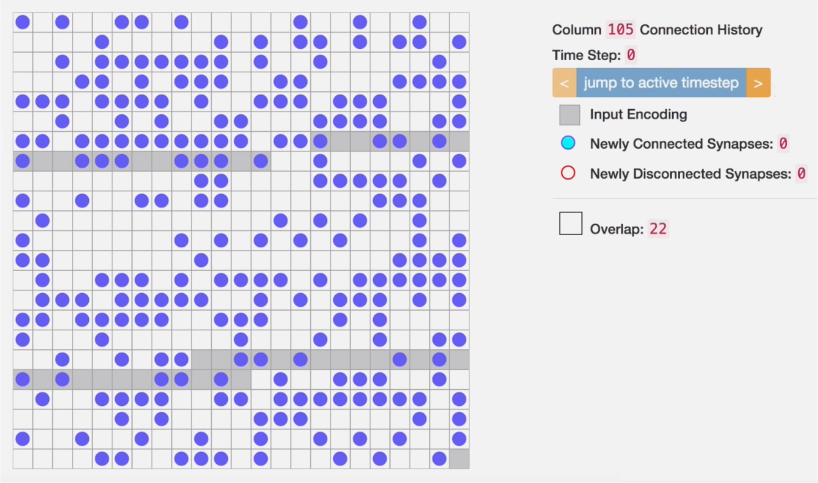
- Many Connections
- Only Columns with highest overlap scores continue
- Everyone else gets inhibited

Spatial Pooler - Connection details

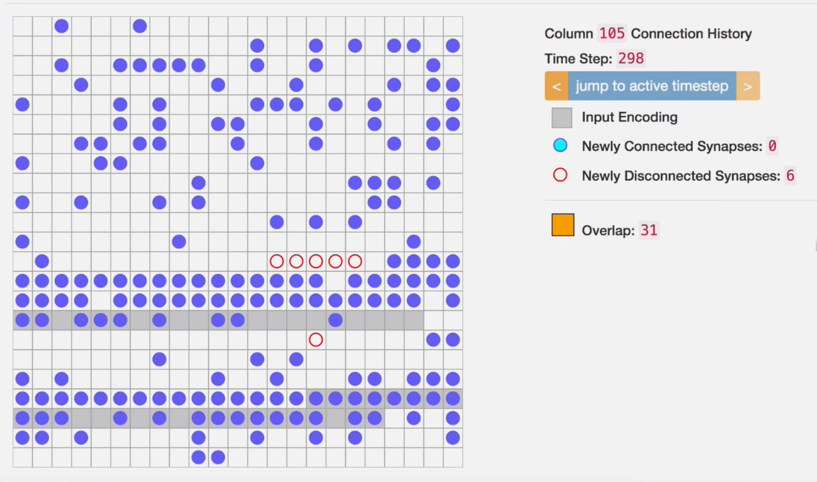
Show `Ep8/Learning Rules`!

- Many Connections
- Only Columns with highest overlap scores continue
- Everyone else gets inhibited
- Next: Updating Permanence Values

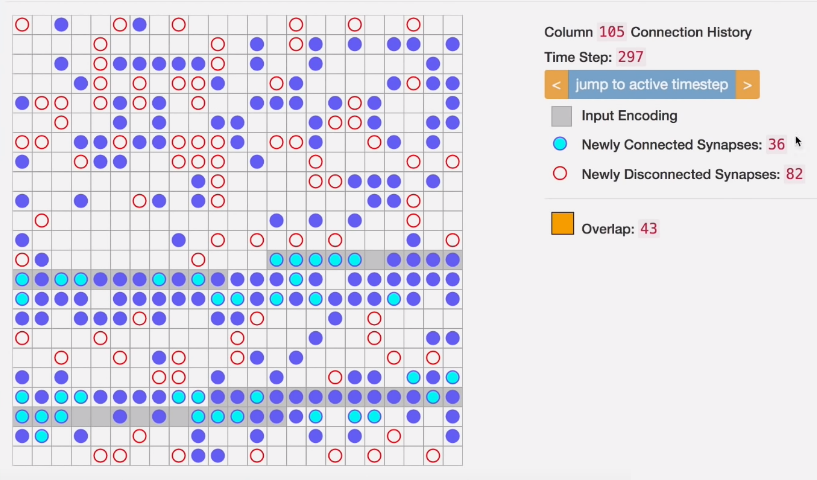
Spatial Pooler - Learning Details i



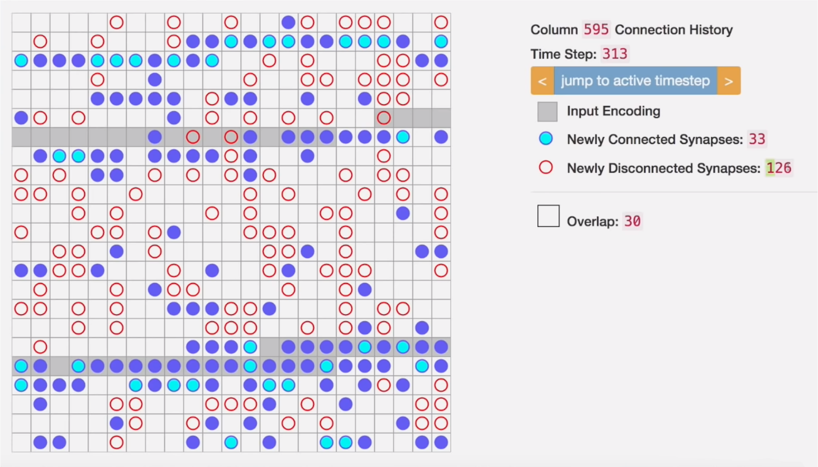
Spatial Pooler - Learning Details ii



Spatial Pooler - Learning Details iii

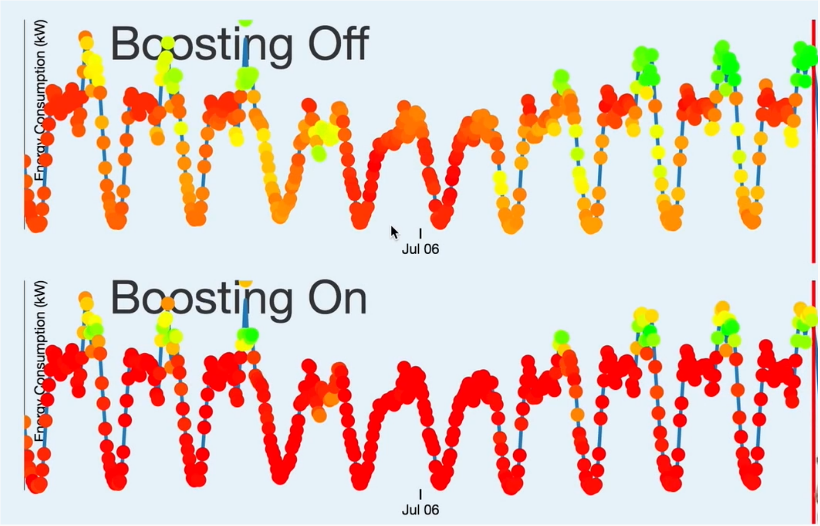


Spatial Pooler - Learning Details iv

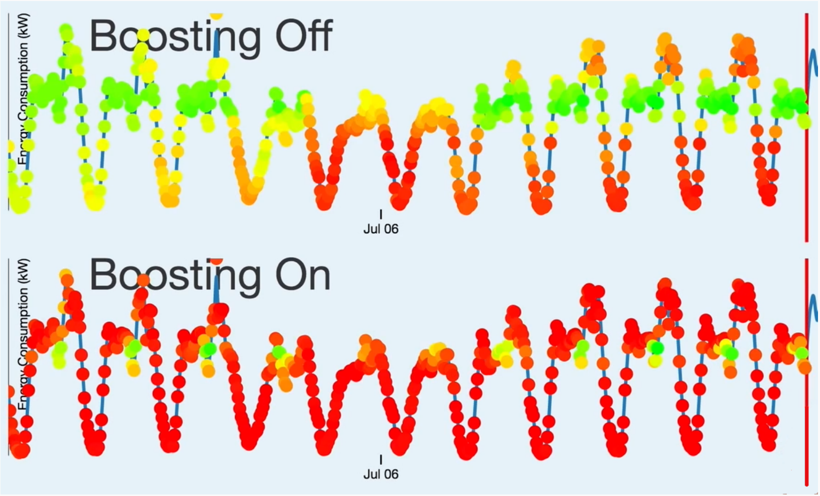


Spatial Pooler - Boosting i

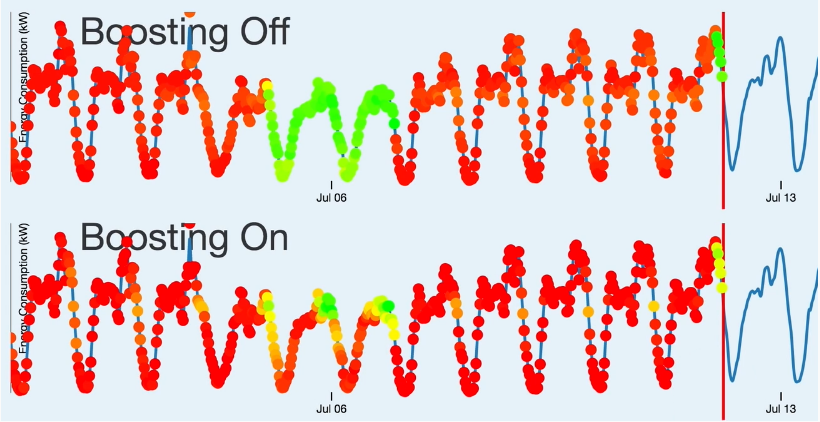
Spatial Pooler - Boosting ii



Spatial Pooler - Boosting iii



Spatial Pooler - Boosting iv



Spatial Pooler - Parameters

- Algorithm Structure (receptive field)

Spatial Pooler - Parameters

- Algorithm Structure (receptive field)
- Inhibition

Spatial Pooler - Parameters

- Algorithm Structure (receptive field)
- Inhibition
- Learning rates

Spatial Pooler - Parameters

- Algorithm Structure (receptive field)
- Inhibition
- Learning rates
- Column Activity

Spatial Pooler - Phases

Spatial Pooler - Phases

1. Initializing with random variables

Spatial Pooler - Phases

1. Initializing with random variables
2. Compute overlap scores (+Boost)

Spatial Pooler - Phases

1. Initializing with random variables
2. Compute overlap scores (+Boost)
3. Inhibition

Spatial Pooler - Phases

1. Initializing with random variables
2. Compute overlap scores (+Boost)
3. Inhibition
4. Updating Permanence values

Spatial Pooler - Phases

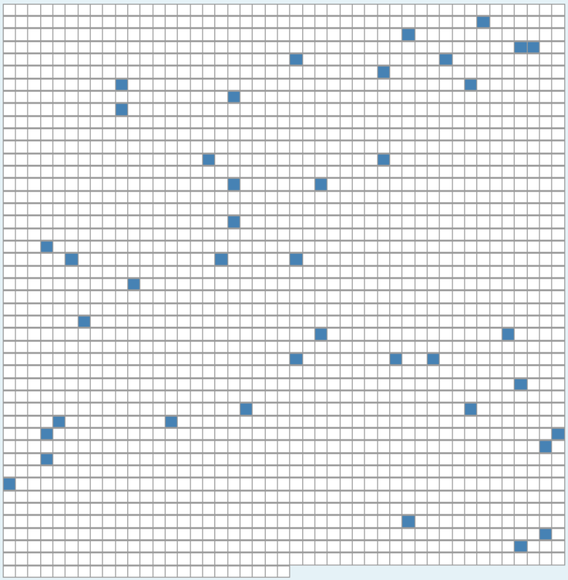
1. Initializing with random variables
2. Compute overlap scores (+Boost)
3. Inhibition
4. Updating Permanence values
5. Repeat from step 2 with new input

Content

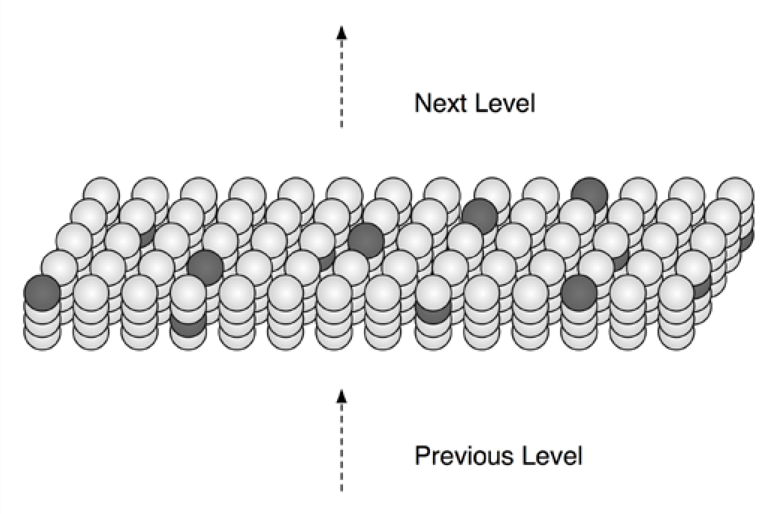
Temporal Pooler - Pipeline

Image adapted from [10].

Temporal Pooler - Introduction



Temporal Pooler - Introduction



Temporal Pooler - Steps

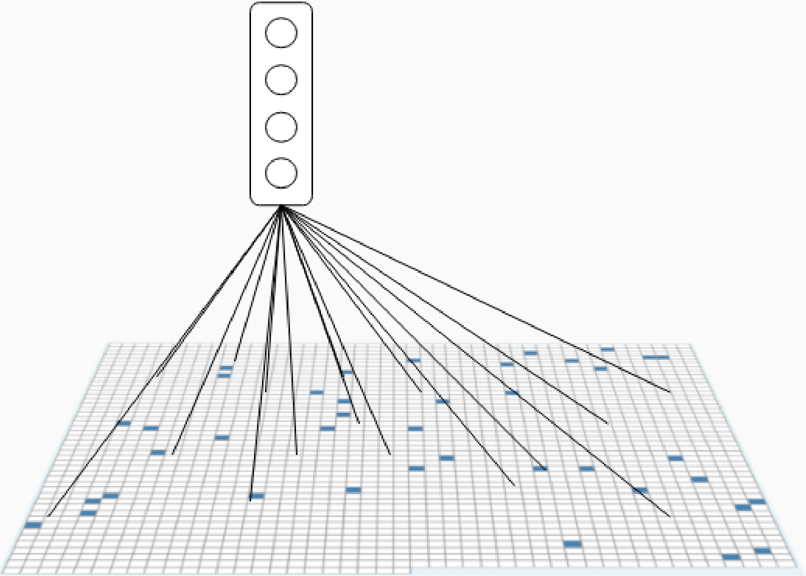
Temporal Pooler - Steps

1. Form representation in context of previous states

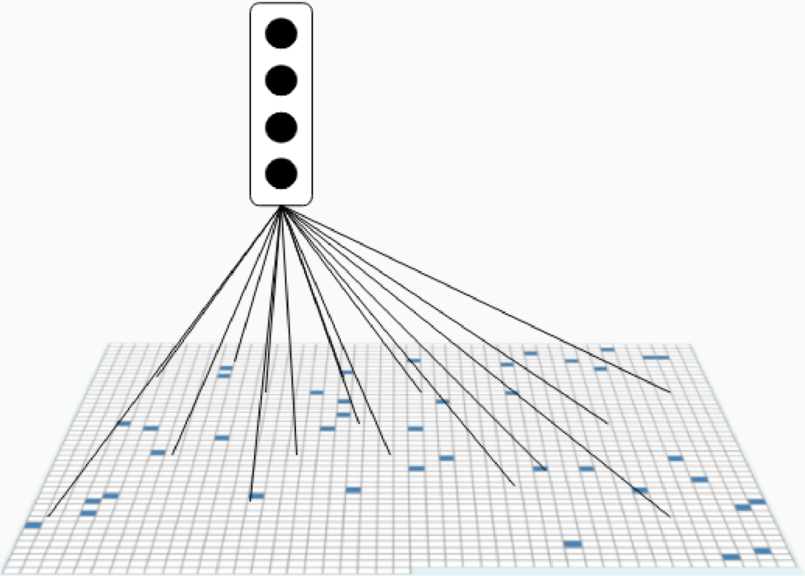
Temporal Pooler - Steps

1. Form representation in context of previous states
2. Form predictions based on previous inputs

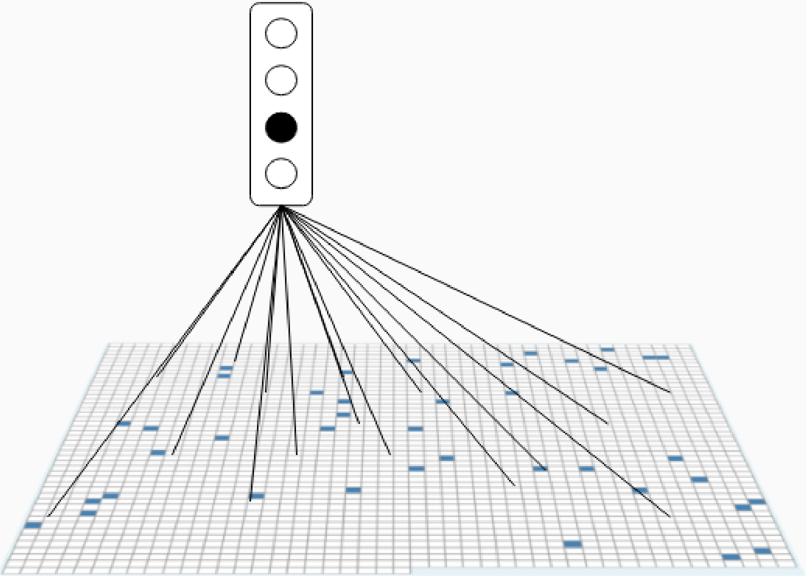
Temporal Pooler - Selecting Winner Cells



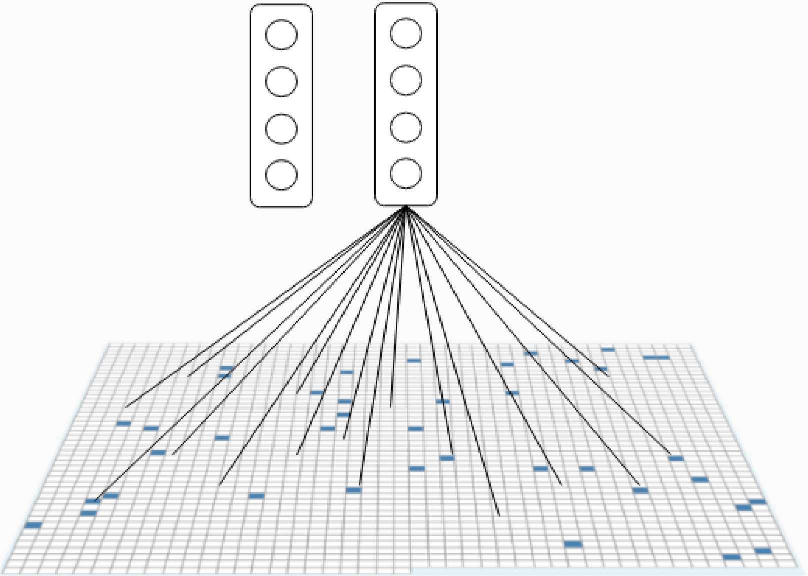
Temporal Pooler - Selecting Winner Cells



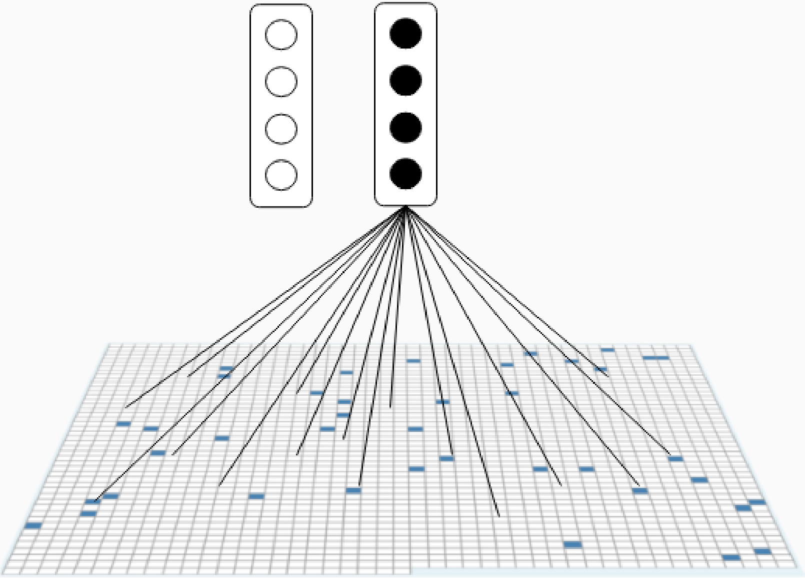
Temporal Pooler - Selecting Winner Cells



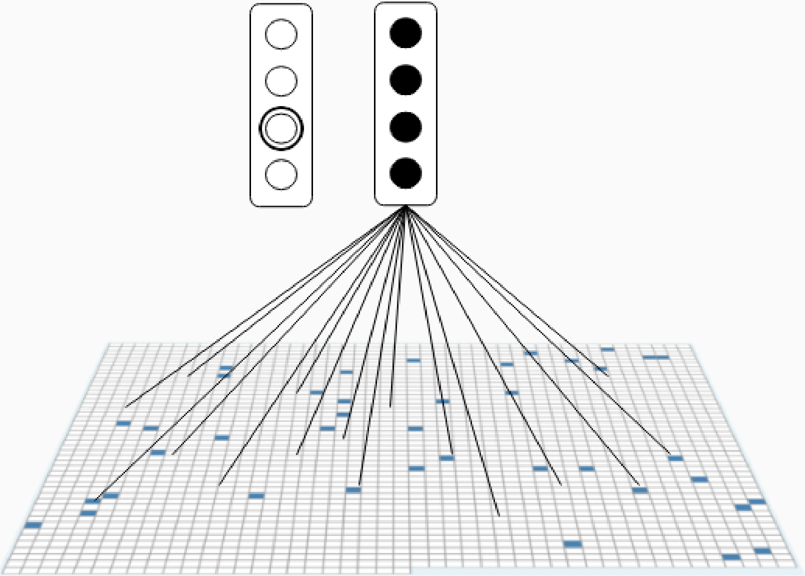
Temporal Pooler - Selecting Winner Cells



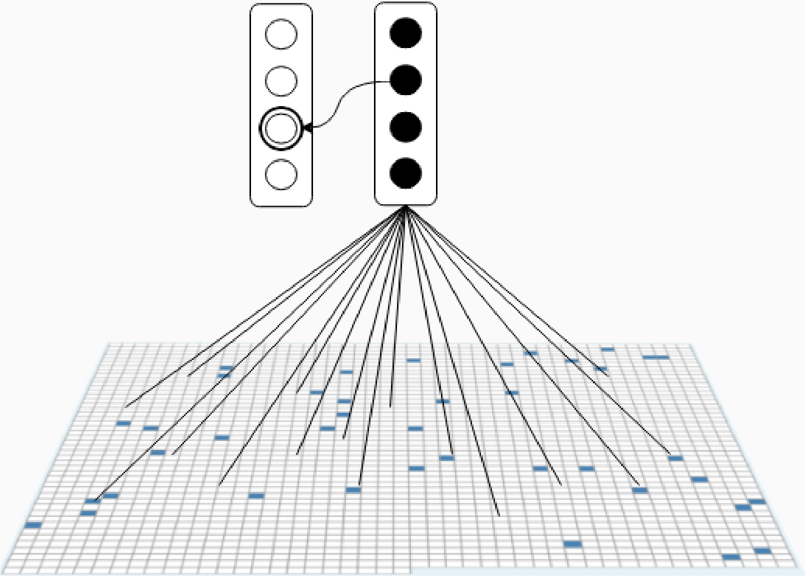
Temporal Pooler - Selecting Winner Cells



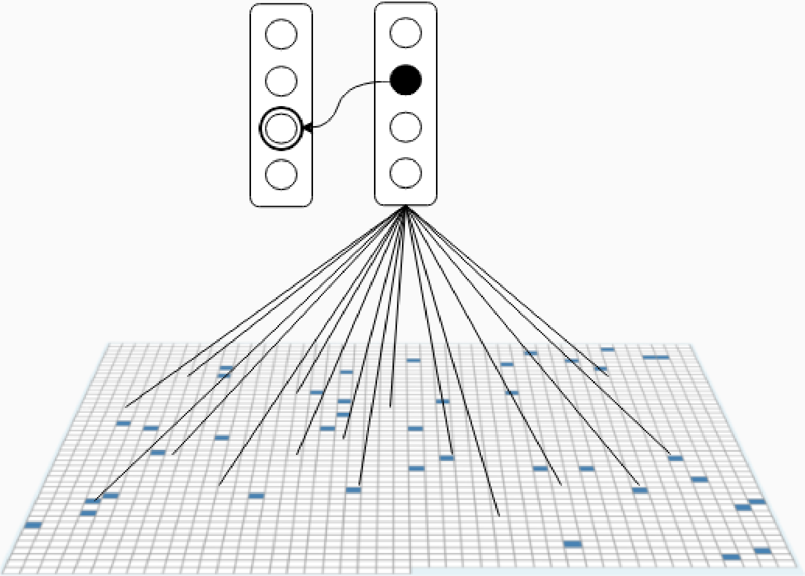
Temporal Pooler - Selecting Winner Cells



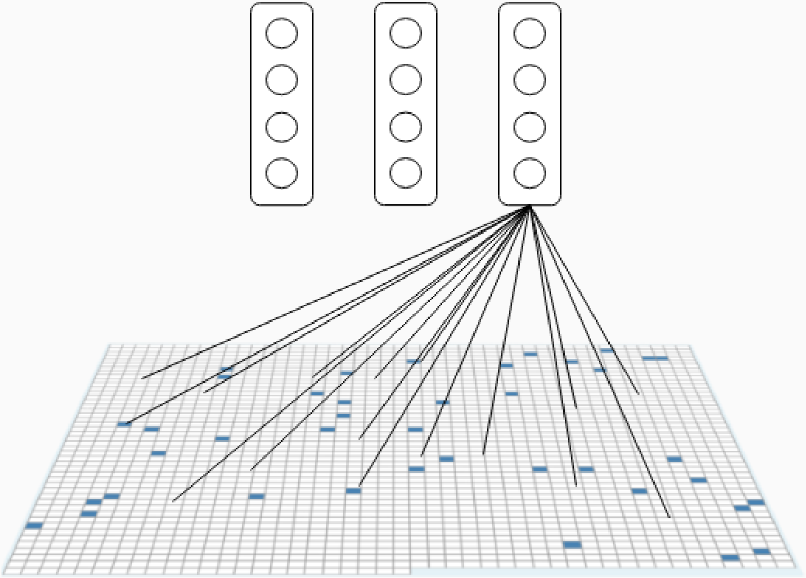
Temporal Pooler - Selecting Winner Cells



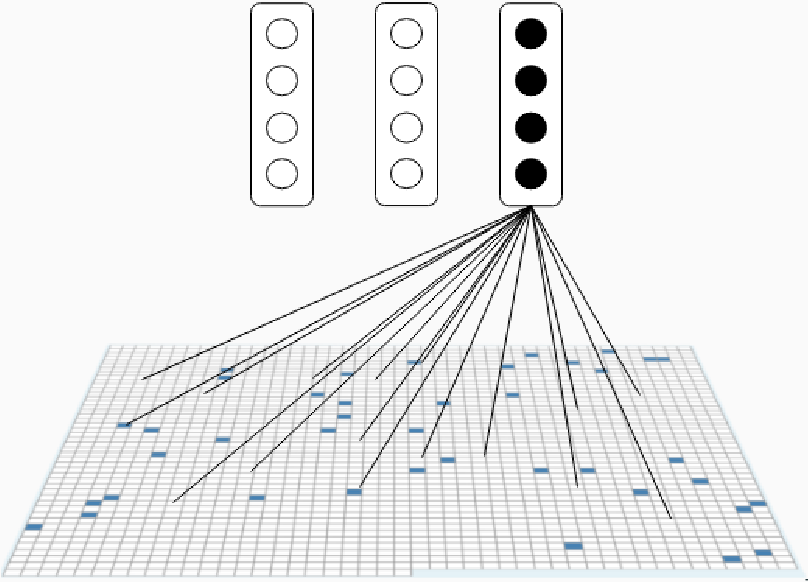
Temporal Pooler - Selecting Winner Cells



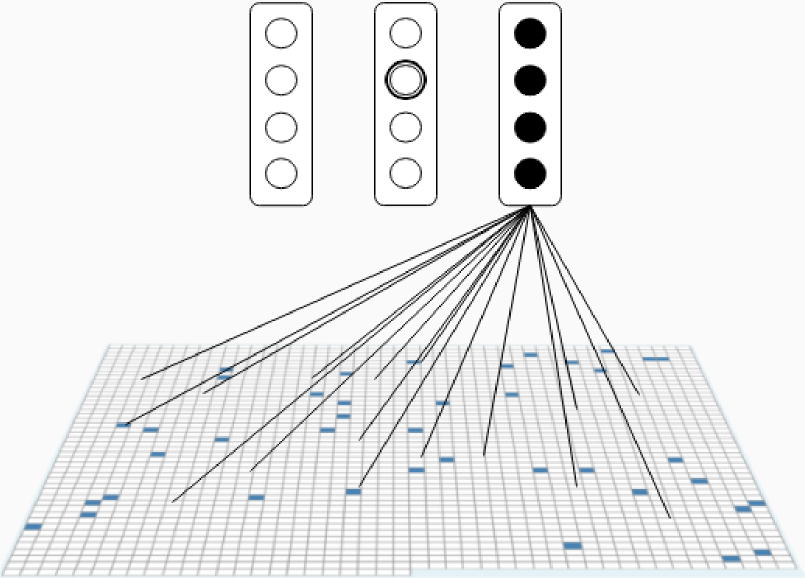
Temporal Pooler - Selecting Winner Cells



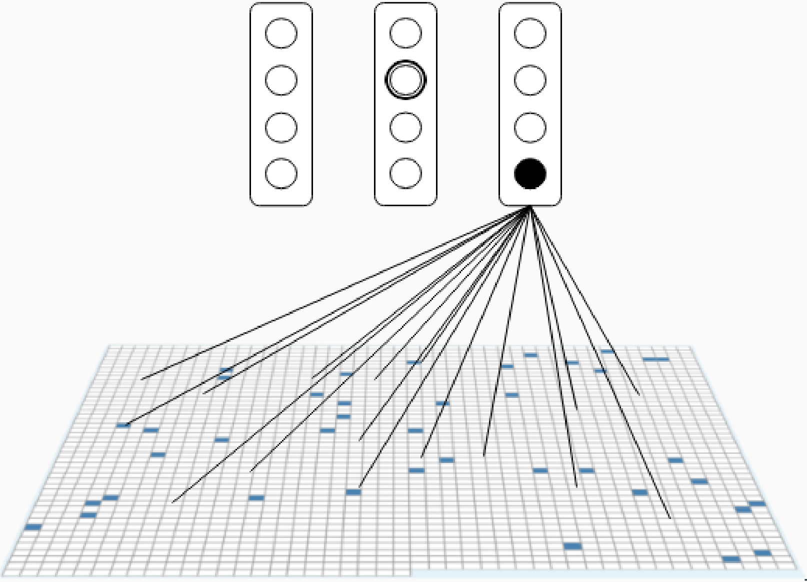
Temporal Pooler - Selecting Winner Cells



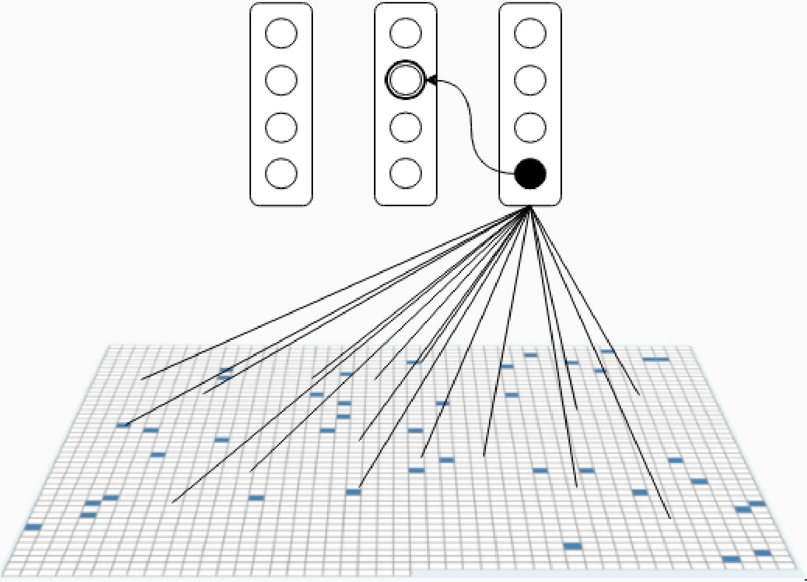
Temporal Pooler - Selecting Winner Cells



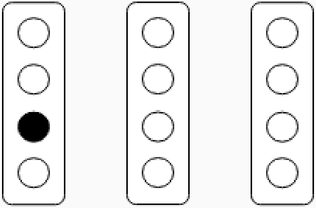
Temporal Pooler - Selecting Winner Cells



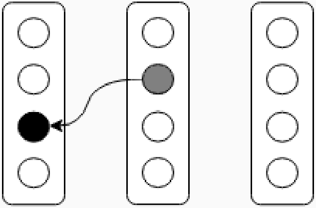
Temporal Pooler - Selecting Winner Cells



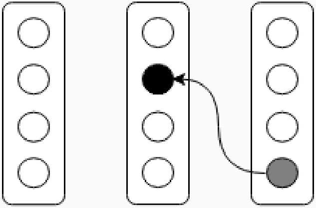
Temporal Pooler - Selecting Winner Cells



Temporal Pooler - Selecting Winner Cells



Temporal Pooler - Selecting Winner Cells



Temporal Memory - Example I

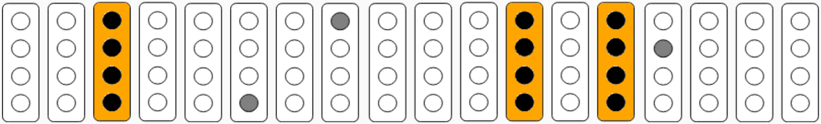
We are very busy

Temporal Memory - Example I

We are very busy

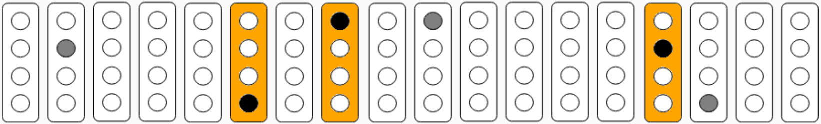
You are very knowledgeab

Temporal Memory - Example I



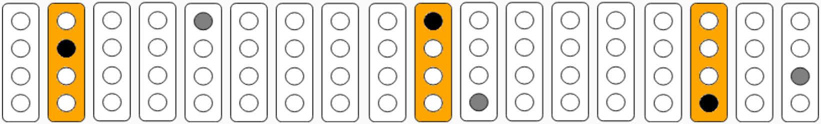
We

Temporal Memory - Example I



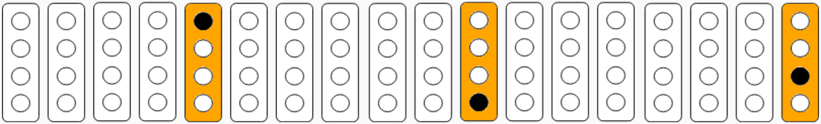
We are

Temporal Memory - Example I



We are very

Temporal Memory - Example I



We are very busy

Temporal Memory - Example I

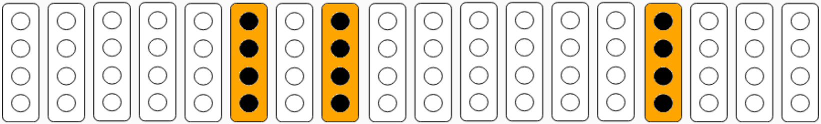
We are very busy

You are very knowledgeab

Temporal Memory - Example I

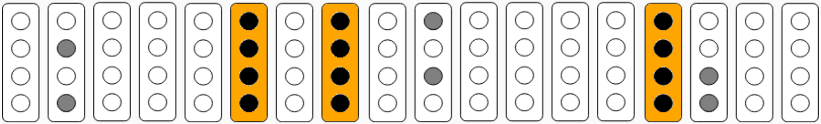
We are very busy

You are very knowledgeab



are

Temporal Memory - Example I

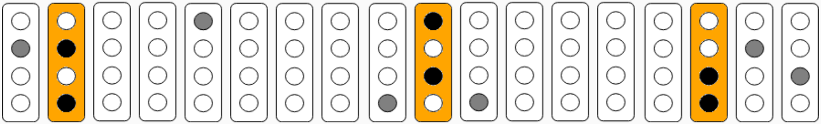


are

Temporal Memory - Example I

We are very busy

You are very knowledgeab



very

Temporal Memory - Advanced

Temporal Memory - Advanced

- Within a region

Temporal Memory - Advanced

- Within a region
- But also across

Temporal Memory - Advanced

- Within a region
- But also across
- Up and Down

Temporal Memory - Advanced

- Within a region
- But also across
- Up and Down
- There are much more connections DOWN than UP

Temporal Memory - Advanced

- Within a region
- But also across
- Up and Down
- There are much more connections DOWN than UP
- In fact, about 90% go either sideways or DOWN

Temporal Memory - Example II

Temporal Memory - Example II

- I **ate** a pear

Temporal Memory - Example II

- I **ate** a pear
- I have **eight** pears

Temporal Memory - Example II

- I **ate** a pear
- I have **eight** pears

- I ...

Temporal Memory - Example II

- I **ate** a pear
- I have **eight** pears

- I ...
- I have ...

Temporal Memory - Example II

- I **ate** a pear
- I have **eight** pears

- I ...
- I have ...

Temporal predictions add to the threshold for the spatial pooler!

Temporal Memory - SDRs Advanced II

Q: If you have an SDR with 10 000 Cells and 200 active, how much difference would saving only 20 of them make?

Q: If you have an SDR with 10 000 Cells and 200 active, how much difference would saving only 20 of them make?

A: Due to the property of SDRs, it is *very* unlikely that they activate in a totally unrelated pattern.

Content

Recap

Let us go over all of the content once more.

Content

Content

Open Questions

- Neuron fire frequency

Content

Sources i

The slides are online: `https://github.com/fkarg/things-to-talk-about/blob/master/htm/main.pdf`

Drop me a mail: fkarg10@gmail.com

Sources ii

- J. Hawkins, S. Ahmad, S. Purdy, and A. Lavin, "Biological and machine intelligence (bami)." Online release 0.5 (March 2017), 2017.
- S. Herculano-Houzel, "The human brain in numbers: a linearly scaled-up primate brain," *Frontiers in human neuroscience*, vol. 3, p. 31, 2009.
- D. J. Sobotta, "An anatomical illustration from sobotta's human anatomy 1908." `https://upload.wikimedia.org/wikipedia/commons/e/ea/Sobo_1909_624.png`, 1908.

Licensed under CC BY-SA 3.0; Accessed 2019-08-17.

B. B. Project, "Cortical column." https://www.mada.org.il/brain/tools-e.html, 2012.
Accessed 2019-08-18.

B. Wang, W. Ke, J. Guang, G. Chen, L. Yin, S. Deng, Q. He, Y. Liu, T. He, R. Zheng, *et al.*, "Firing frequency maxima of fast-spiking neurons in human, monkey, and mouse neocortex," *Frontiers in cellular neuroscience*, vol. 10, p. 239, 2016.

Sources iv

- A. Impacts, "Neuron firing rates in humans." `https://aiimpacts.org/rate-of-neuron-firing/`, 2015.
  Accessed 2019-08-18.
- G. Yi and W. M. Grill, "Average firing rate rather than temporal pattern determines metabolic cost of activity in thalamocortical relay neurons," *Scientific reports*, vol. 9, no. 1, p. 6940, 2019.

Sources v

- Pixabay.com, "Brain neuron nerves." `pixabay.com`. **Pixabay License (free for commercial and noncommercial use, no attribution required), Accessed 2019-08-21.**
- H. Mengistu, J. Huizinga, J.-B. Mouret, and J. Clune, "The evolutionary origins of hierarchy," *PLoS computational biology*, vol. 12, no. 6, p. e1004829, 2016.
- Y. Cui, S. Ahmad, and J. Hawkins, "The htm spatial pooler—a neocortical algorithm for online sparse distributed coding," *Frontiers in computational neuroscience*, vol. 11, p. 111, 2017.

Sources vi

H. Gray, “Gray754.”
https://en.wikipedia.org/wiki/File:Gray754.png,
1918.
Accessed 2019-08-24.

End

Cortical Column

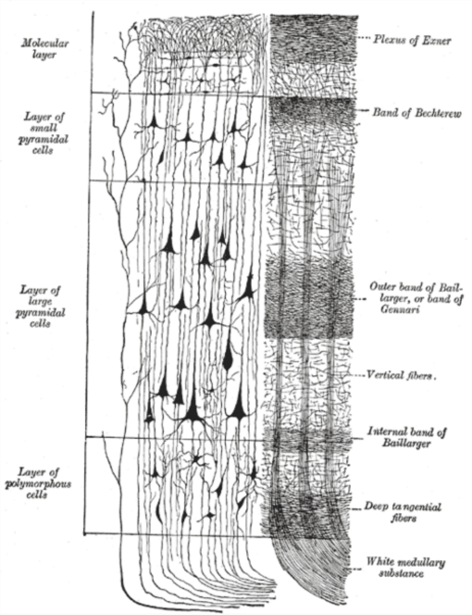


Image from [11]

Cortical Column II