Intelligence and Machines

Creating Intelligent Machines by Modeling the Neocortex

Simons Institute
VOTC
May 30, 2013
Jeff Hawkins

Numenta
GROK

1) Discover operating principles of neocortex

2) Build systems based on these principles

Intelligence and Machines: agenda

- Brief history of machine intelligence
- Define machine intelligence
- Neocortical principles
- State of the art: Grok
- Future of intelligent machines

Alan Turing

"Computers are universal machines" 1935+

"Human behavior as test for machine intelligence" 1950

VOL. LIX. No. 236.] [October, 1950

MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

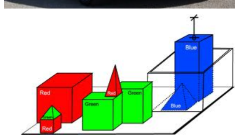
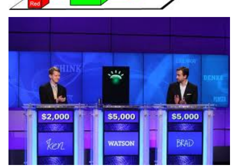
BY A. M. TURING

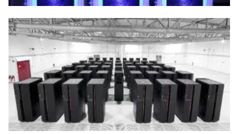
Artificial Intelligence - no neuroscience

AI Projects/Techniques

- 4CAPS
- ACT-R
- PreAct
- Apex
- Asimo
- CALO, DARPA
- CHREST
- CLARION
- CoJACK
- Copycat
- Cyc
- Deep Blue
- DUAL
- EPIC
- Expert systems
- CogAff schema
- FORR
- Global Workspace Theory
- Mycin
- Open mind common sense
- PRODIGY
- R-CAST
- SHRDLU
- Soar
- Watson





Major AI Initiatives

- MIT AI Lab
- 5th Generation Computing Project
- DARPA Strategic Computing Initiative
- DARPA Grand Challenge

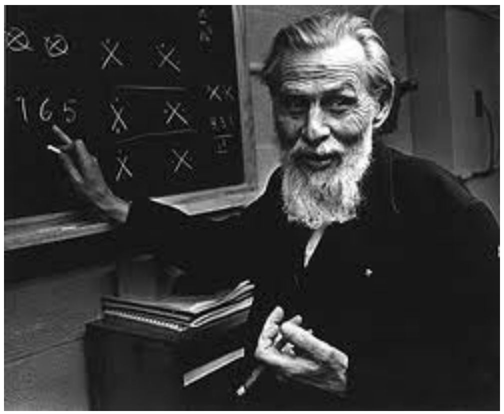
Pros:
- **Good solutions**

Cons:
- **Task specific**
- **Limited or no learning**

Warren McCulloch, Walter Pitts

"Neurons as logic gates" 1943

"Neural networks for computation"

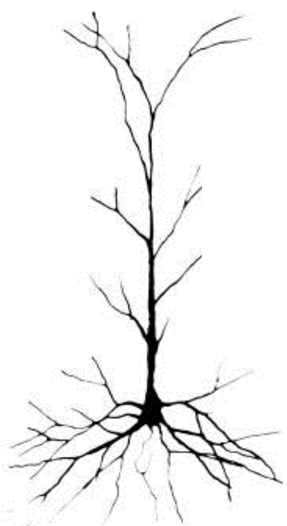


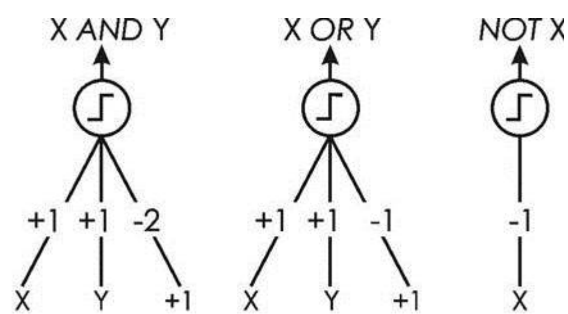
A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY

WARREN S. MCCULLOCH AND WALTER PITTS

FROM THE UNIVERSITY OF ILLINOIS, COLLEGE OF MEDICINE, DEPARTMENT OF PSYCHIATRY AT THE ILLINOIS NEUROPSYCHIATRIC INSTITUTE, AND THE UNIVERSITY OF CHICAGO

Because of the "all-or-none" character of nervous activity, neural events and the relations among them can be treated by means of propositional logic. It is found that the behavior of every net can be described in these terms, with the addition of more complicated logical means for nets containing circles; and that for any logical expression satisfying certain conditions, one can find a net behaving in the fashion it describes. It is shown that many particular choices among possible neurophysiological assumptions are equivalent, in the sense that for every net behaving under one assumption, there exists another net which behaves under the other and gives the same results, although perhaps not in the same time. Various applications of the calculus are discussed.

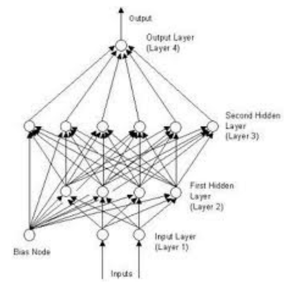


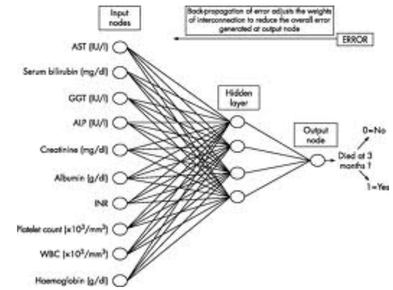


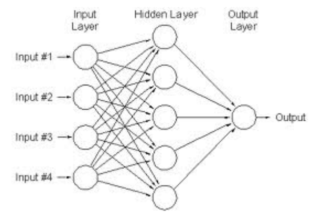
Artificial Neural Networks – almost no neuroscience

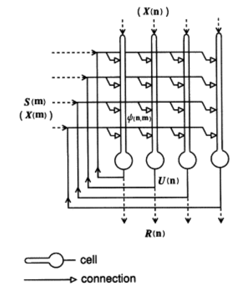
- Back propagation
- Boltzman machines
- Hopfield networks
- Kohonen networks
- Parallel Distributed Processing

- Machine learning
- Deep Learning







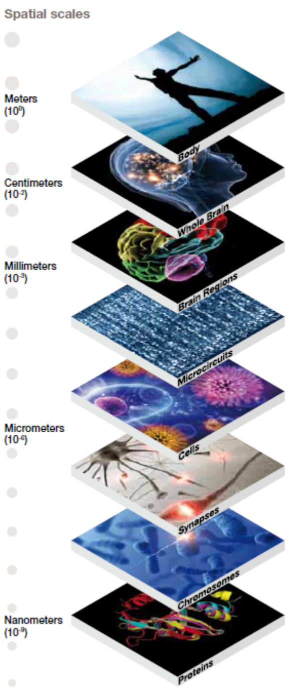


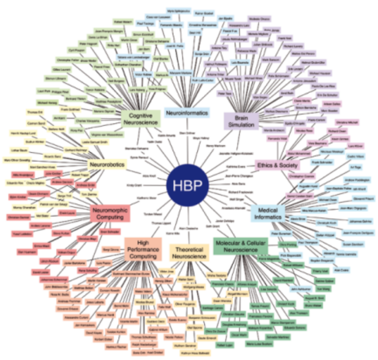
Pros:
- Good classifiers
- Learning systems

Cons:
- Limited
- Not brain like

Whole Brain Projects – maximal neuroscience

- **Human Brain Project**





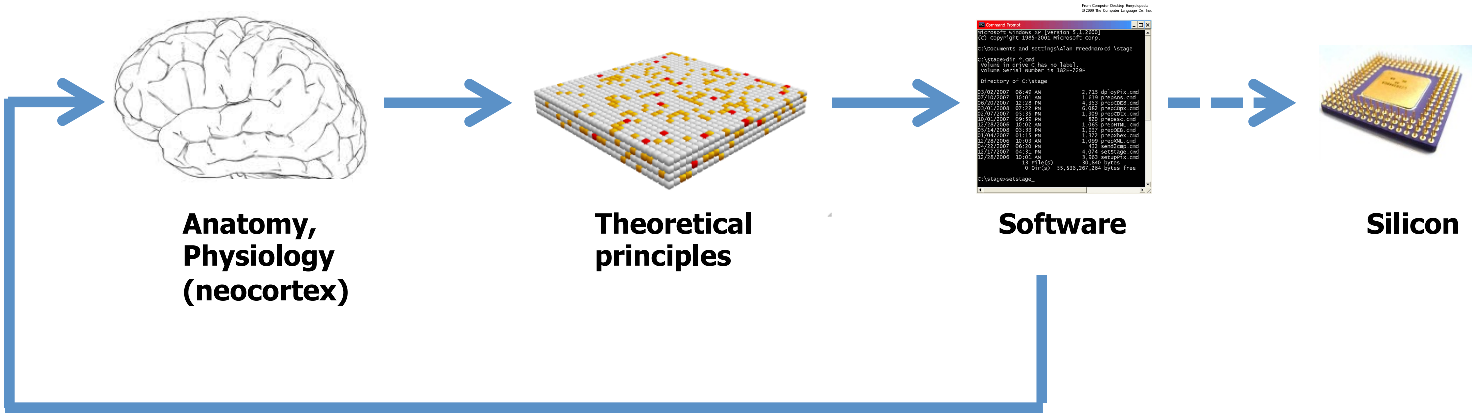
- **BRAIN Initiative**

No theory
No attempt at Machine Intelligence

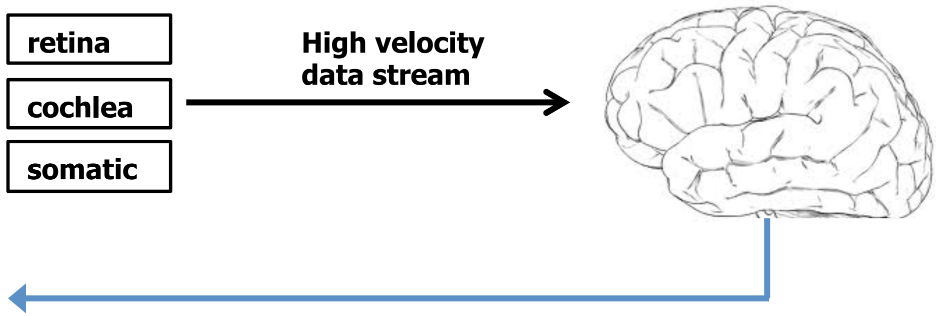
The “Just Right” Approach

1) Discover operating principles of neocortex

2) Build systems based on these principles



The neocortex is a memory system.



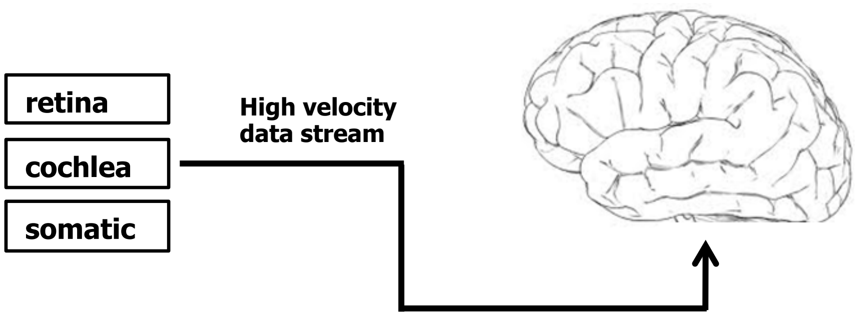
The neocortex learns a model from sensory data

- predictions
- anomalies
- actions

The neocortex learns a sensory-motor model of the world

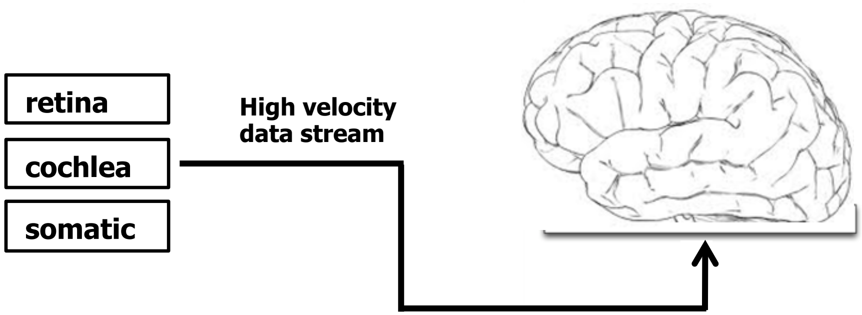
Intelligence is the ability to learn a sensory-motor model of the world.

Principles of Neocortical Function



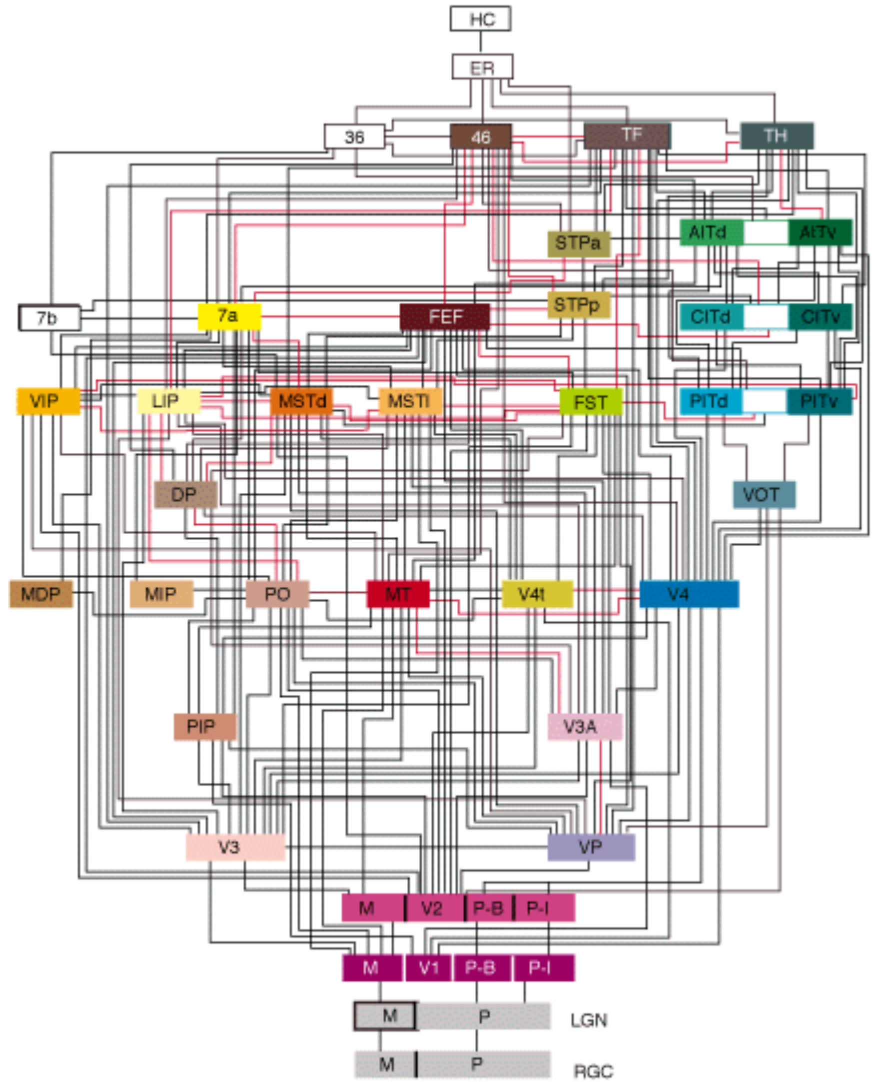
1) On-line learning from streaming data

Principles of Neocortical Function

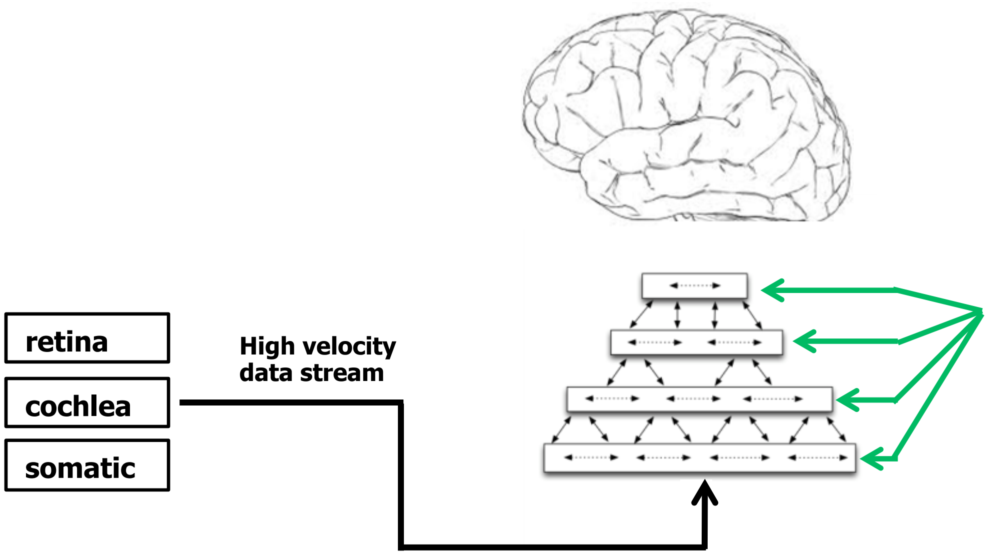


1) On-line learning from streaming data

2) Hierarchy of memory regions


HC
ER
36
46
TF
TH
STPa
AITd
AITv
7b
7a
FEF
STPp
CITd
CITv
VIP
LIP
MSTd
MSTl
FST
PITd
PITv
DP
VOT
MDP
MIP
PO
MT
V4t
V4
PIP
V3A
V3
VP
M
V2
P-B
P-I
M
V1
P-B
P-I
M
P
LGN
M
P
RGC

Principles of Neocortical Function

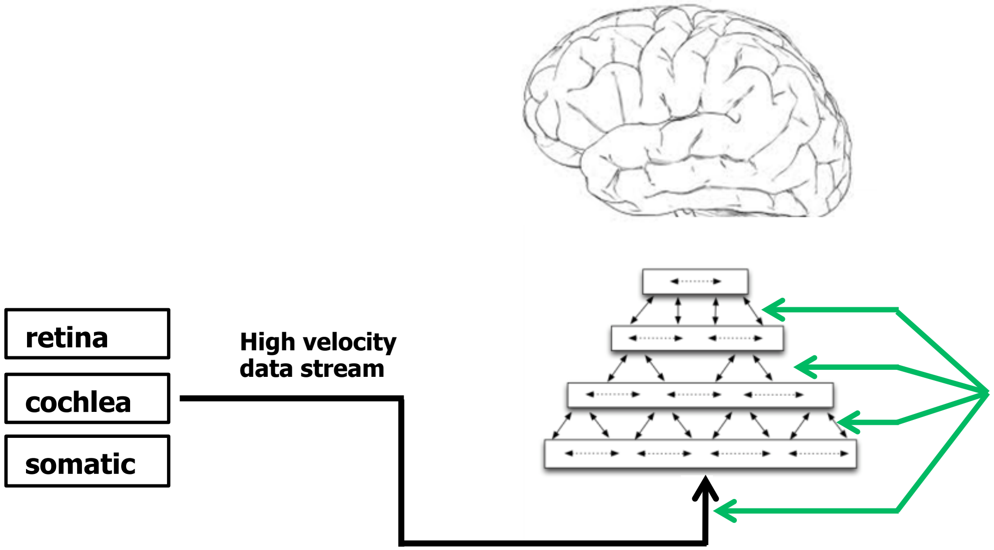


1) On-line learning from streaming data

2) Hierarchy of memory regions

3) Sequence memory
- inference
- motor

Principles of Neocortical Function



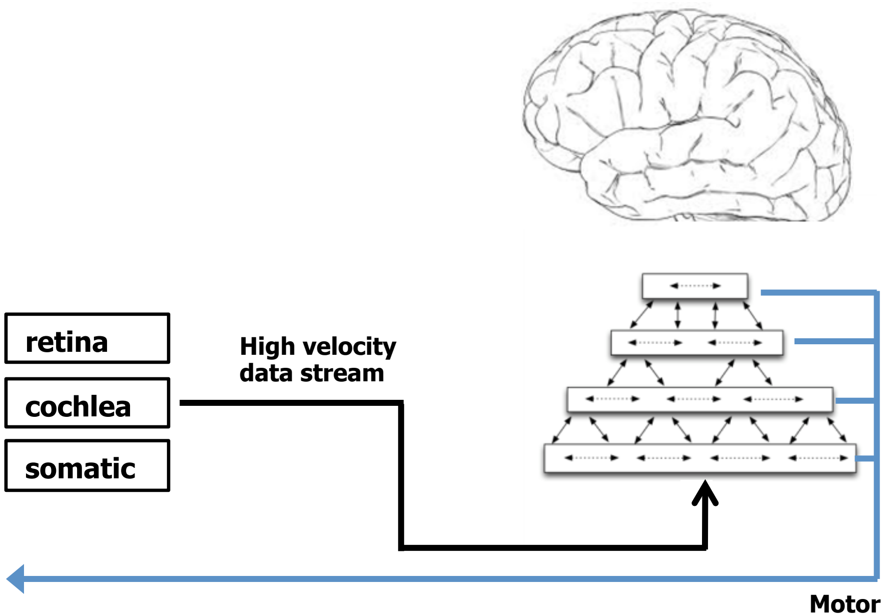
1) On-line learning from streaming data

2) Hierarchy of memory regions

3) Sequence memory

4) Sparse Distributed Representations

Principles of Neocortical Function



1) On-line learning from streaming data

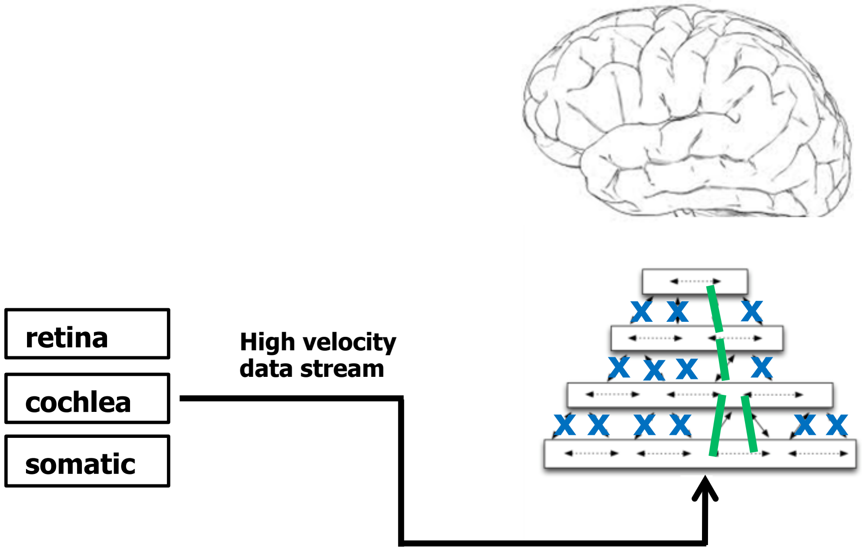
2) Hierarchy of memory regions

3) Sequence memory

4) Sparse Distributed Representations

5) All regions are sensory and motor

Principles of Neocortical Function



1) On-line learning from streaming data

2) Hierarchy of memory regions

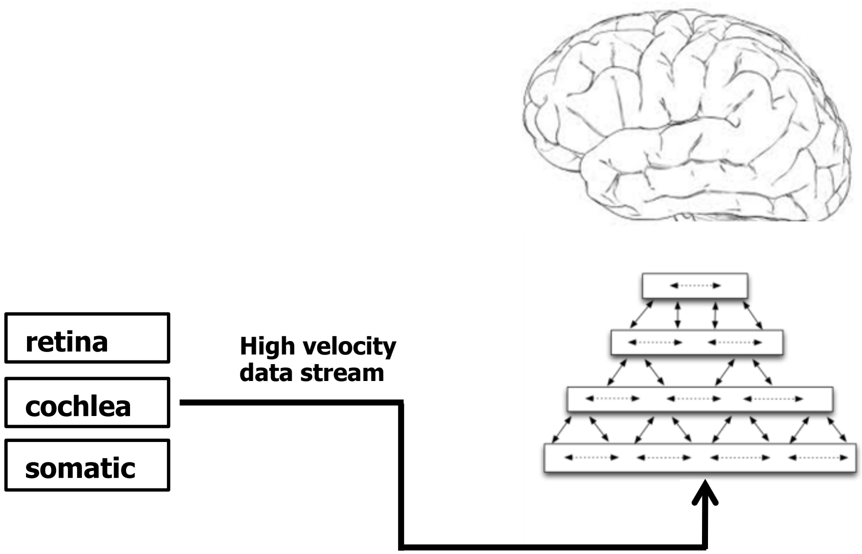
3) Sequence memory

4) Sparse Distributed Representations

5) All regions are sensory and motor

6) Attention

Principles of Neocortical Function



1) On-line learning from streaming data

2) Hierarchy of memory regions

3) Sequence memory

4) Sparse Distributed Representations

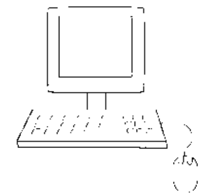
5) All regions are sensory and motor

6) Attention

These six principles are necessary and sufficient for biological and machine intelligence.

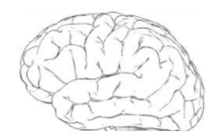
- All mammals from mouse to human have them
- Not necessary: language, human-like emotions, physical body

Dense Representations



- Few bits (8 to 128)
- All combinations of 1's and 0's
- Example: 8 bit ASCII
  01101101 = m
- Individual bits have no inherent meaning
- Representation is assigned by programmer

Sparse Distributed Representations (SDRs)



- Many bits (thousands, cell activity)
- Few 1's mostly 0's
- Example: 2,000 bits, 2% active
  010000000000000000010000000000000000000000000000000000000100000…………01000
- Each bit has semantic meaning
- Meaning of each bit is learned, not assigned

SDR Properties

1) **Similarity**:
shared bits = semantic similarity

01000000000000000000100000000000000000000000000000000000000010000............01000
00000010000000000000100000000000000000000001000000000000000010000............00000

2) **Store and Compare**:
store indices of active bits

0100000000000000000100000000000001000000000000000000000000010000............01000

Indices
1
2
3
4
5
|
40

subsampling is OK

0100000000000000000100000000000001000000000000000000000000010000............01000

Indices
1
2
|
10

3) **Union membership**:

1) 0100000000000000000100000000000000000000000000000000000010000............01000
2) 000000000000000001000000000000000001000000000000000000000000000............00001 2%
3) 0000100000000000000000000000010000000000000000000000100000000000............00000
....
10) 00000000000000100000001000000000000000000000000010000000000000............00000

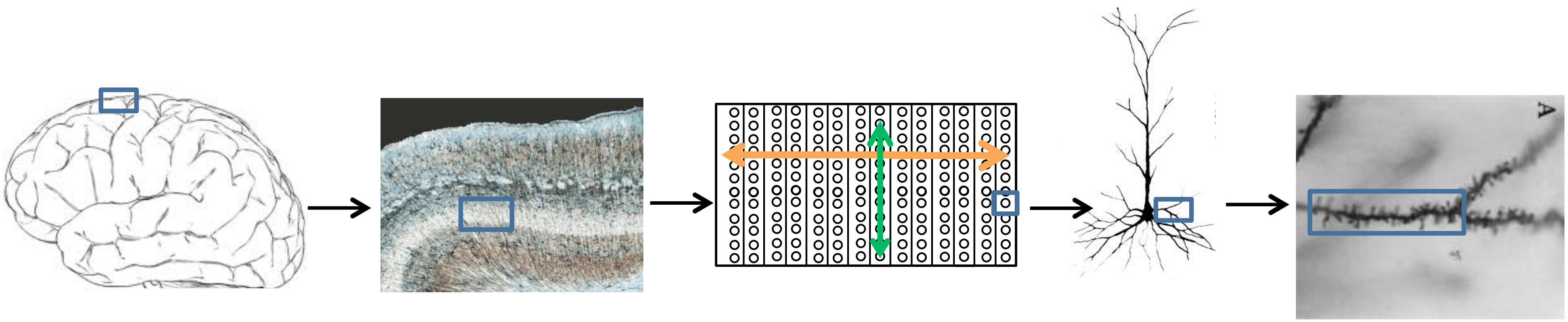
Union → 010010000000001001010010000010000010000000000010100000100000............01001 20%

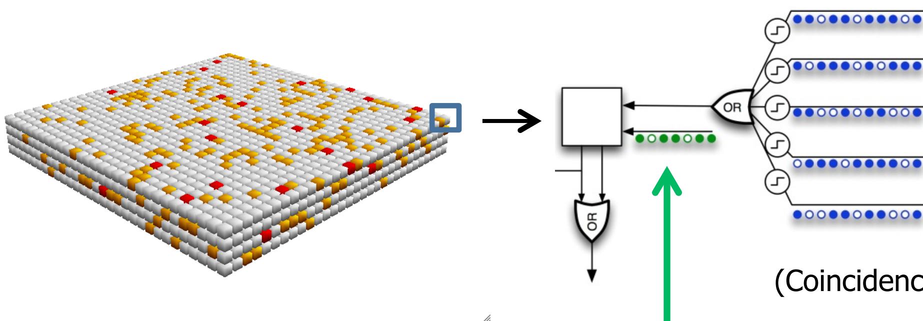
Is this SDR a member? → 00001000000000000000000000000100000000000000000000001000000000............00000

Sequence memory: Theoretical constraints

- Discover temporal structure in large arrays of sensory bits
- Respect topology
- Learn online
- Input patterns will be noisy
- Input patterns may never repeat exactly
- Learn high order sequences
- Make multiple simultaneous predictions
- Detect anomalies
- Generalize as memory fills

Sequence Memory



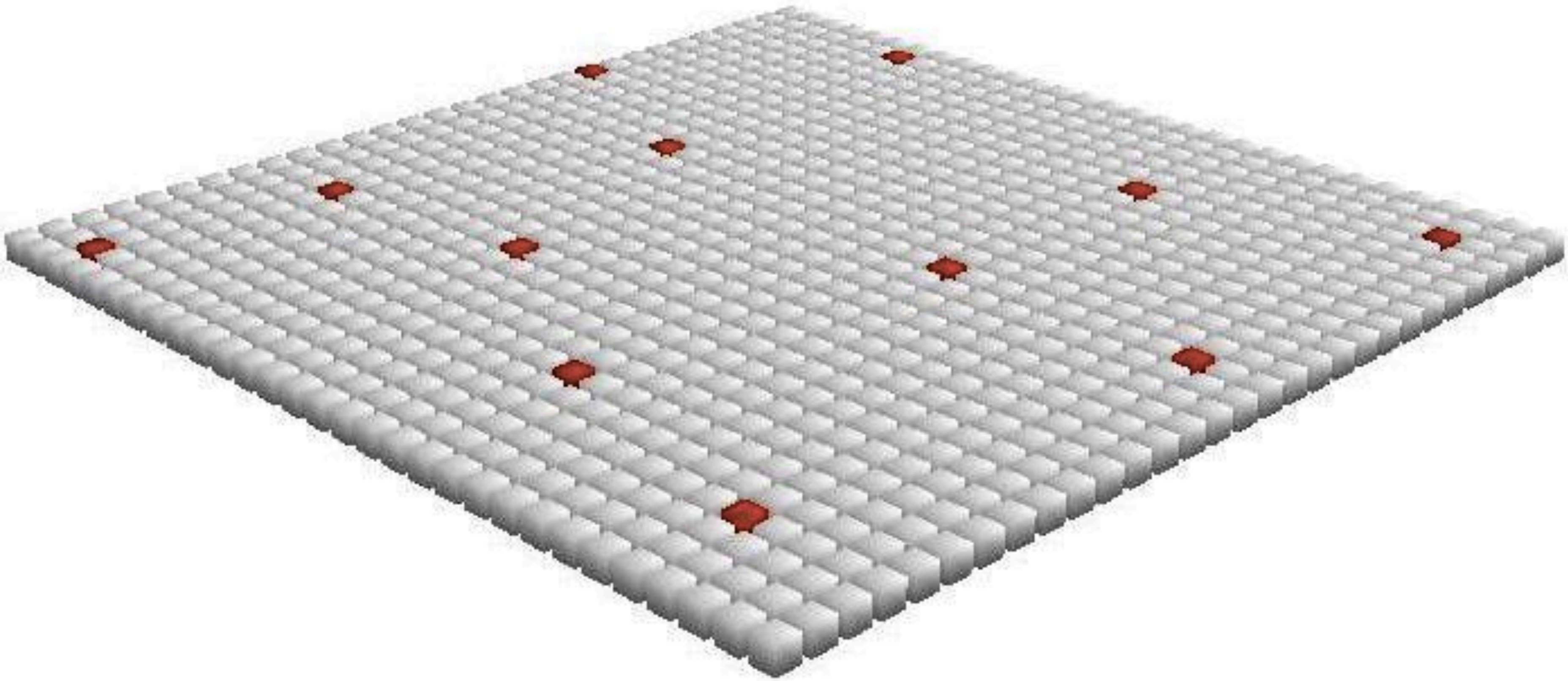


Predict activity in temporal context

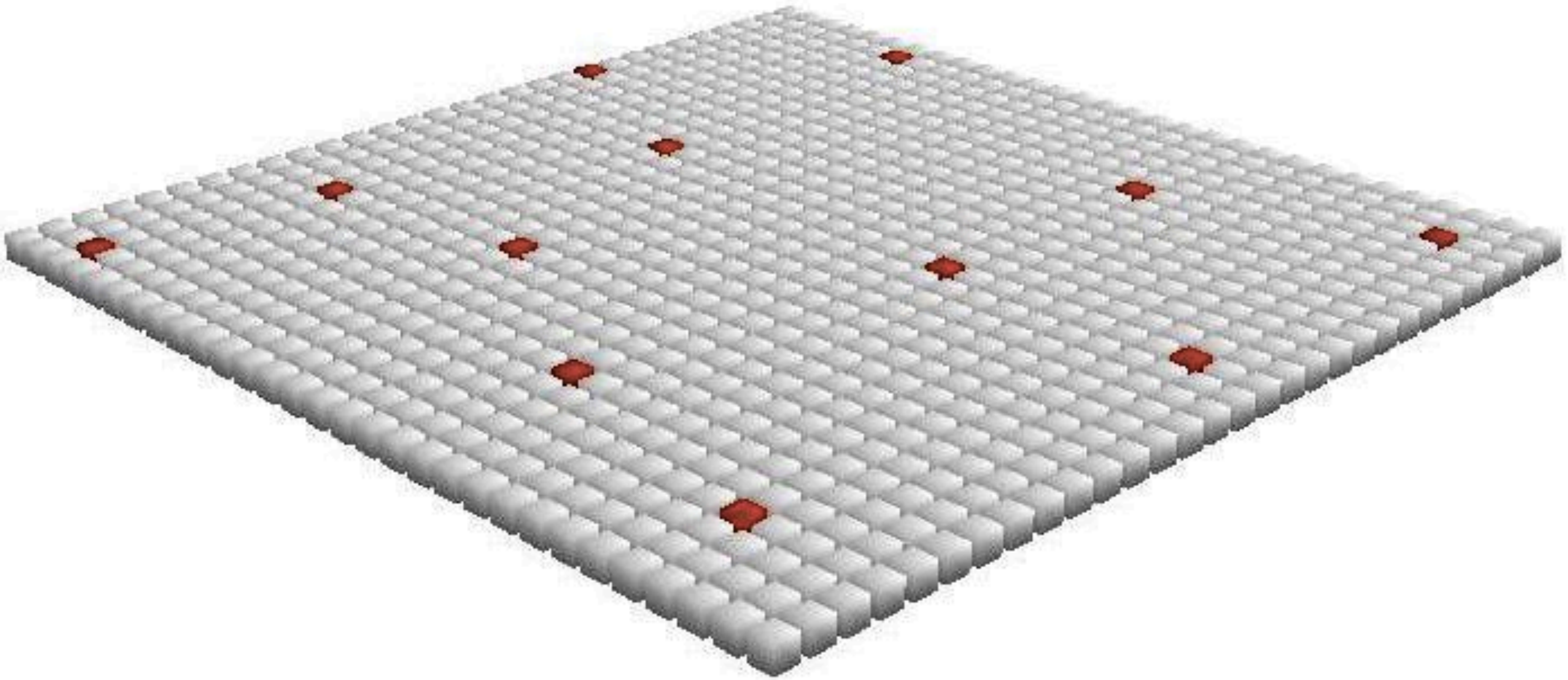
(Coincidence detectors)

Feedforward response property of cell

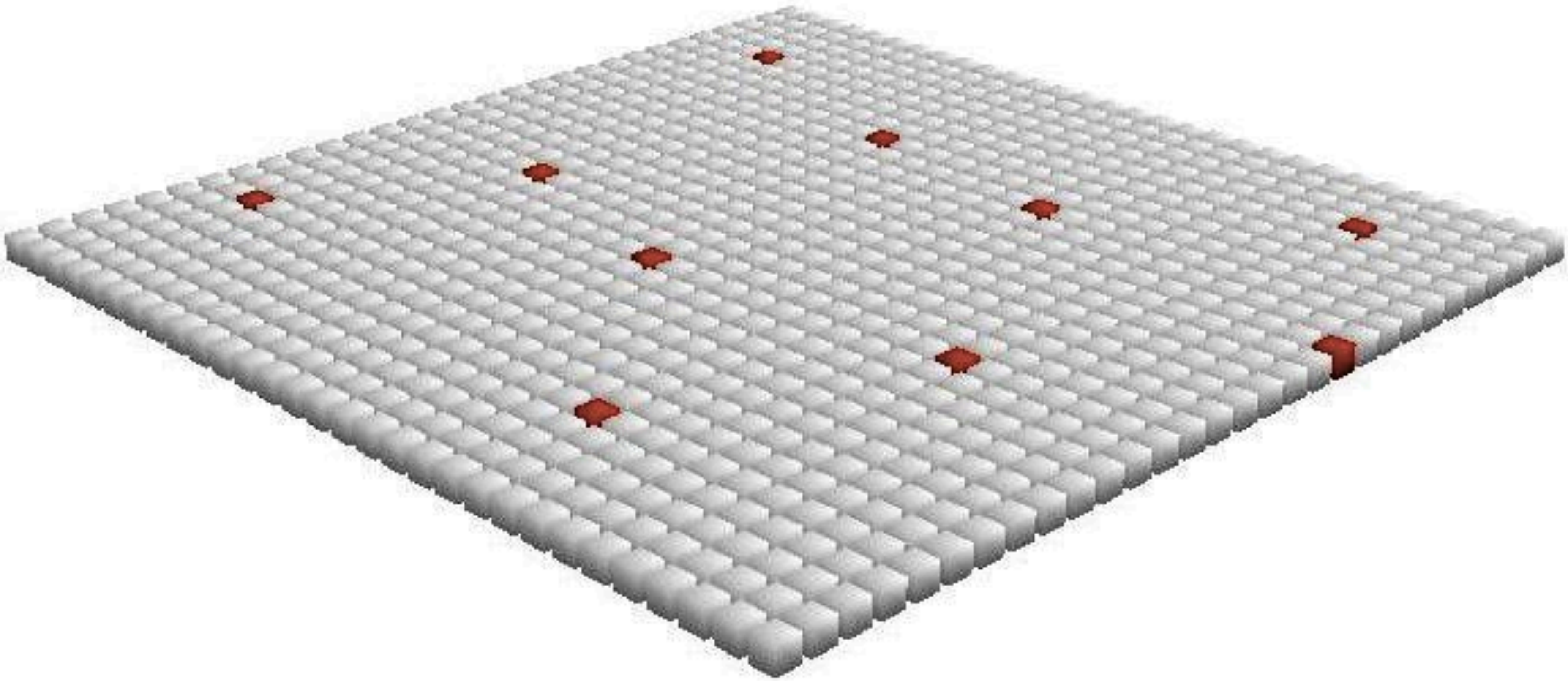
Each cell is one bit in our Sparse Distributed Representation



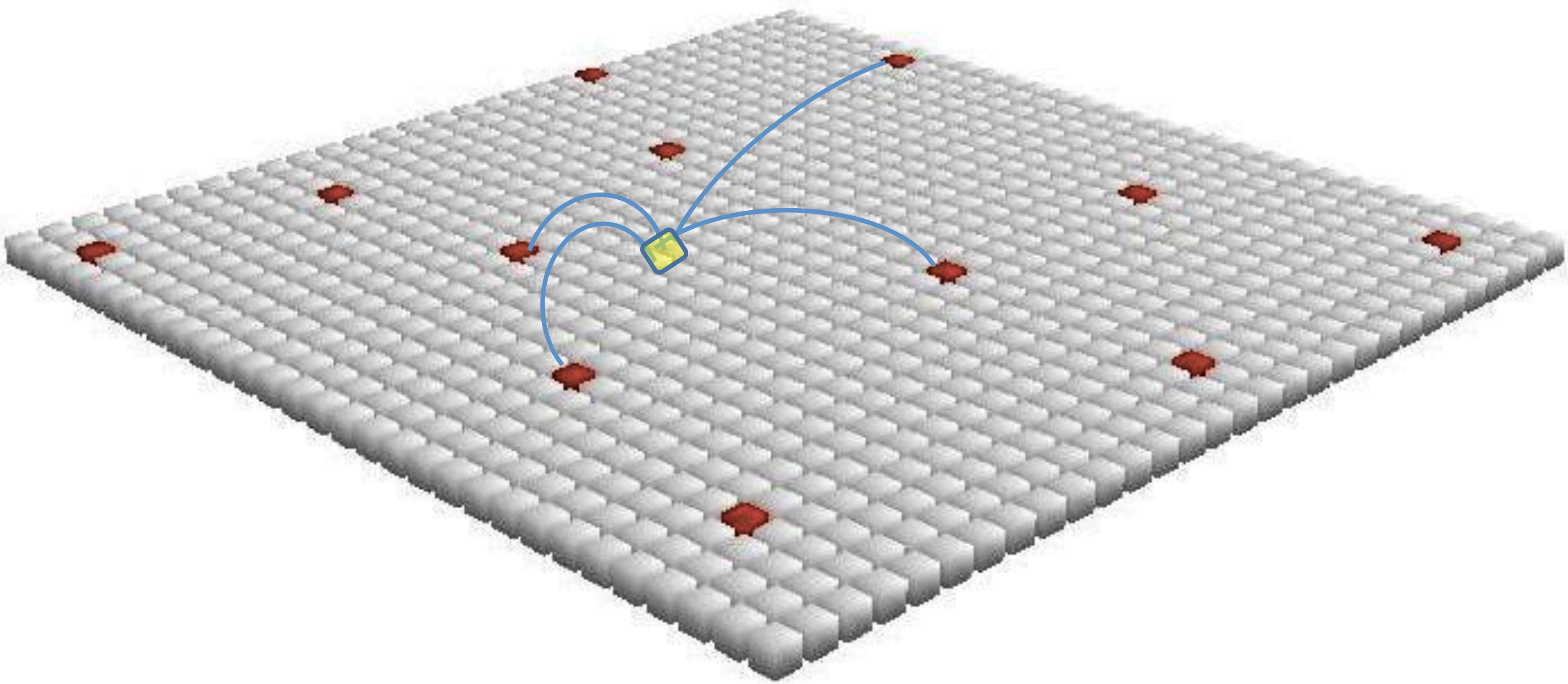
SDR (time =1)



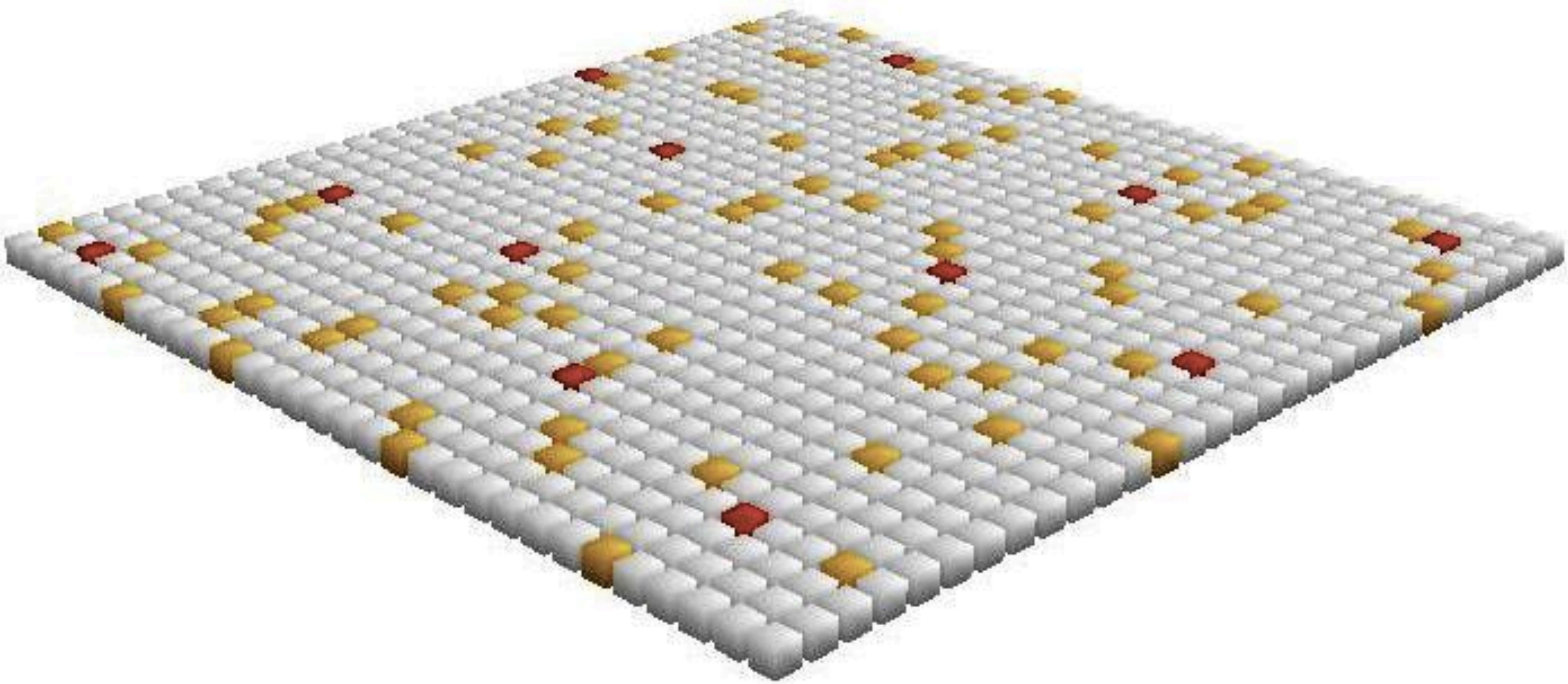
SDR (time =2)



Cells form connections to subsample of previously active cells.
Predicts its own future activity.

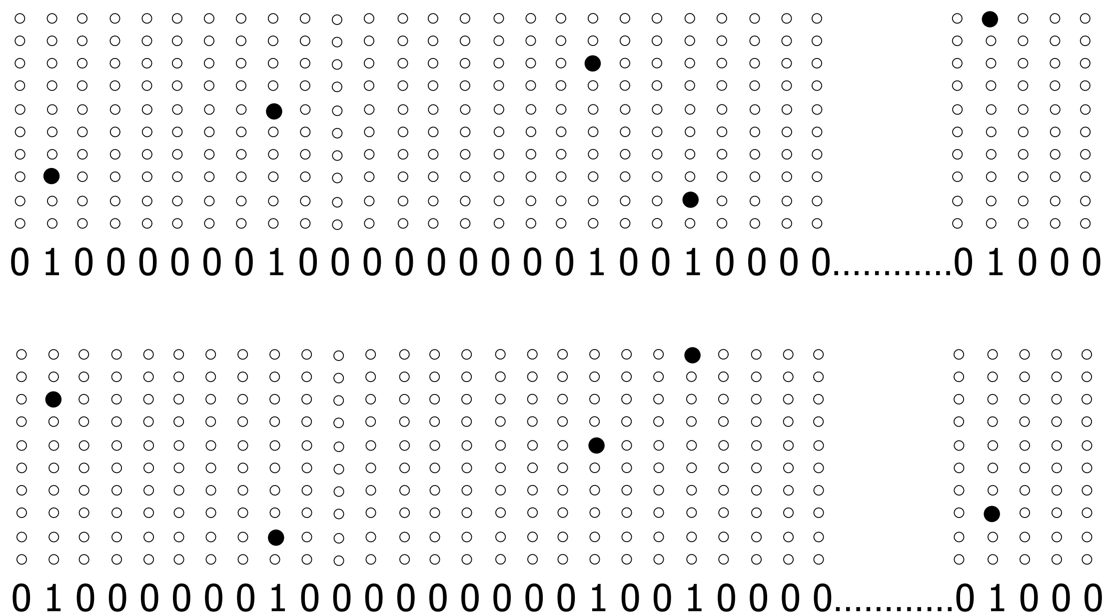


Multiple Predictions Can Occur at Once



With one cell per column, 1st order memory
We need a high order memory

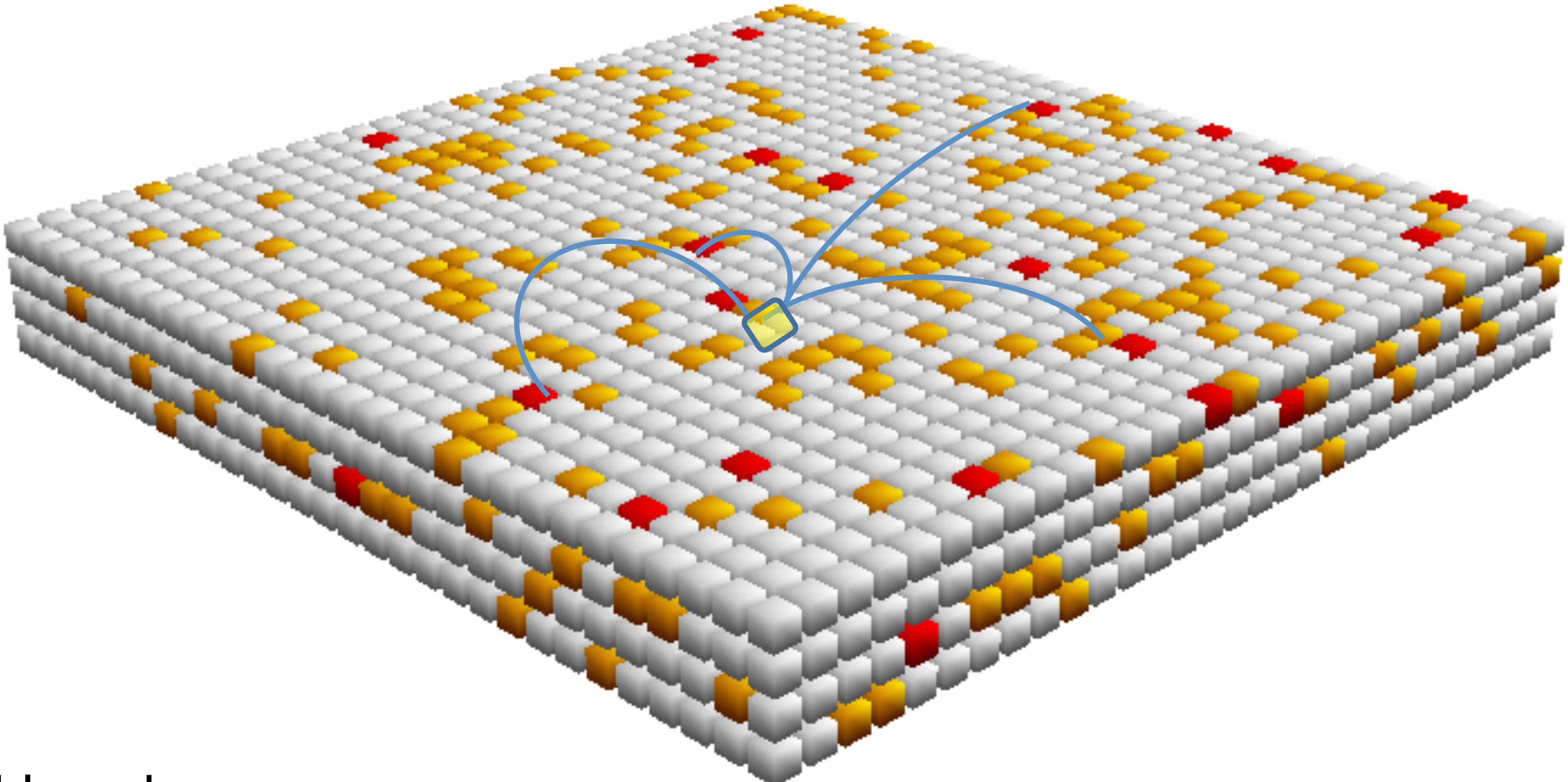
High Order Sequence Memory



40 active columns, 10 cells per column

= 10^{40} ways to represent the same input in different contexts

Variable Order Sequence Memory

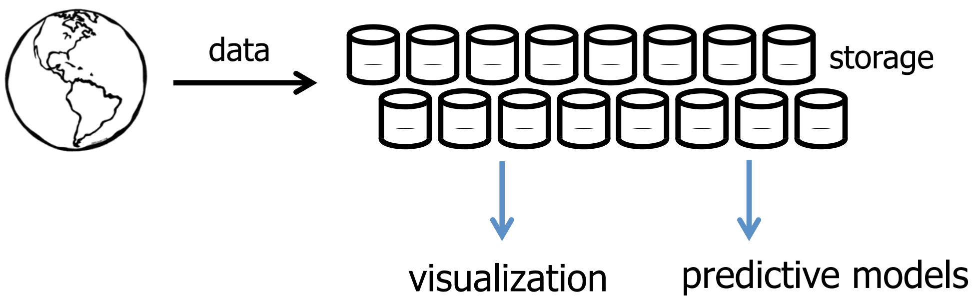


Variable order sequence memory
Multiple simultaneous predictions
High capacity
Distributed, fault tolerant
Semantic generalization

Details at Numenta.com

Predictive Analytics

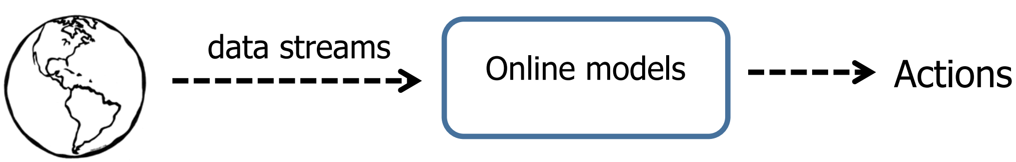
Today



Challenges

- Data prep
- Model obsolescence
- People

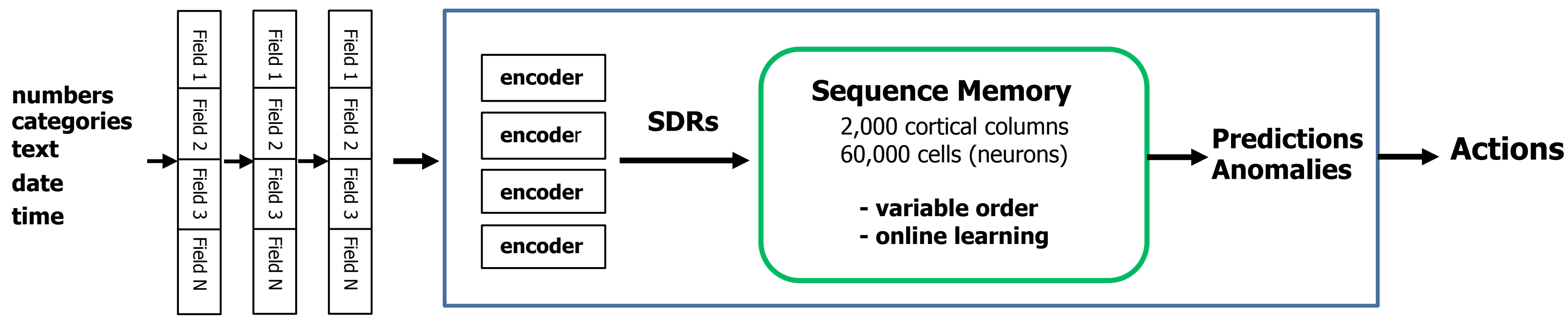
Tomorrow: (M2M, Internet of Things, Industrial Internet)



Key criteria

- Automated model creation
- Continuous learning
- Temporal and spatial models

Grok: A Engine for Acting on Data Streams



User

Access data stream

Define problem
- what to predict
- how often
- how far in advance

Grok

Creates models

Learns continuously

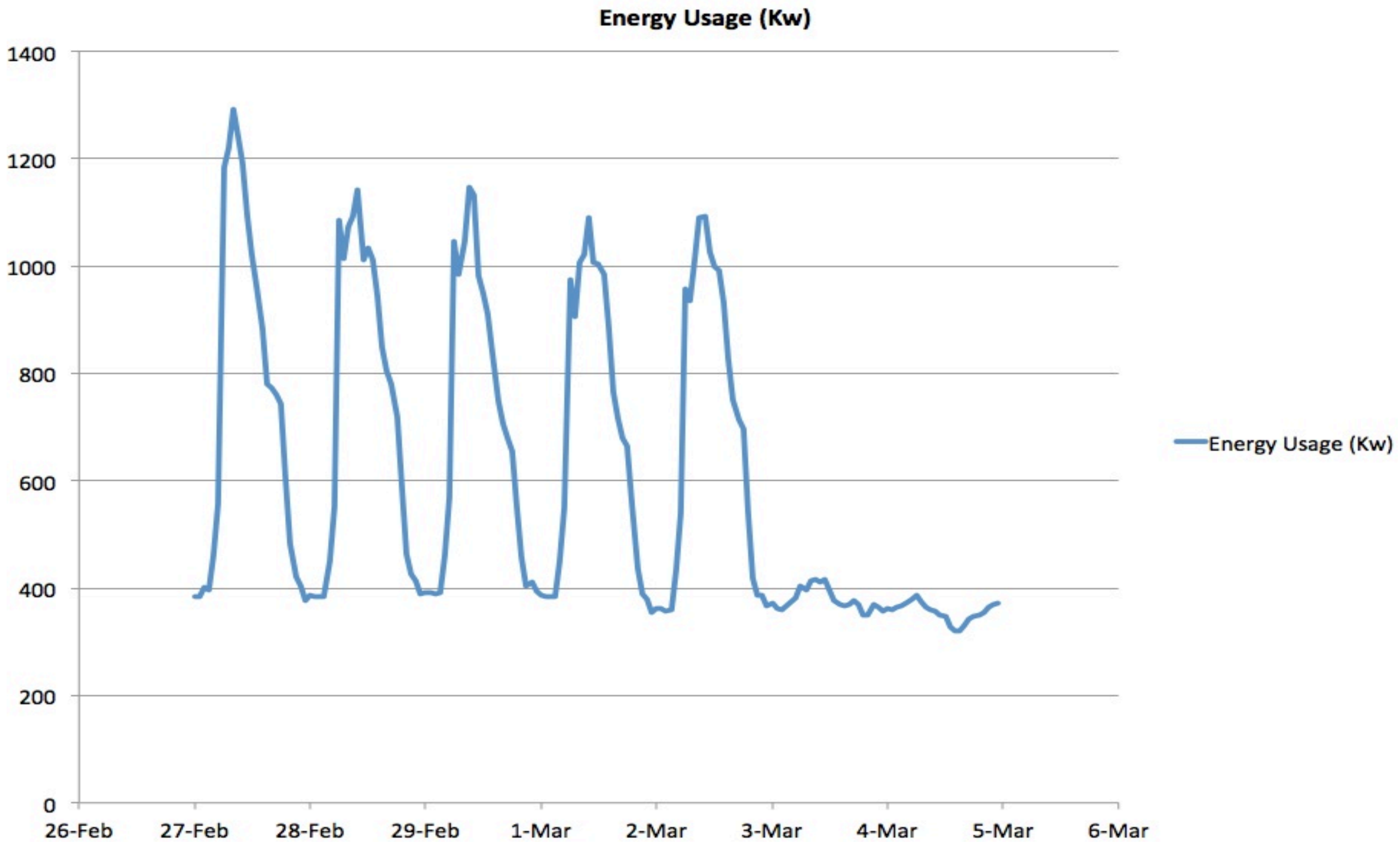
Finds spatial/temporal patterns

Outputs
- predictions
  with probabilities

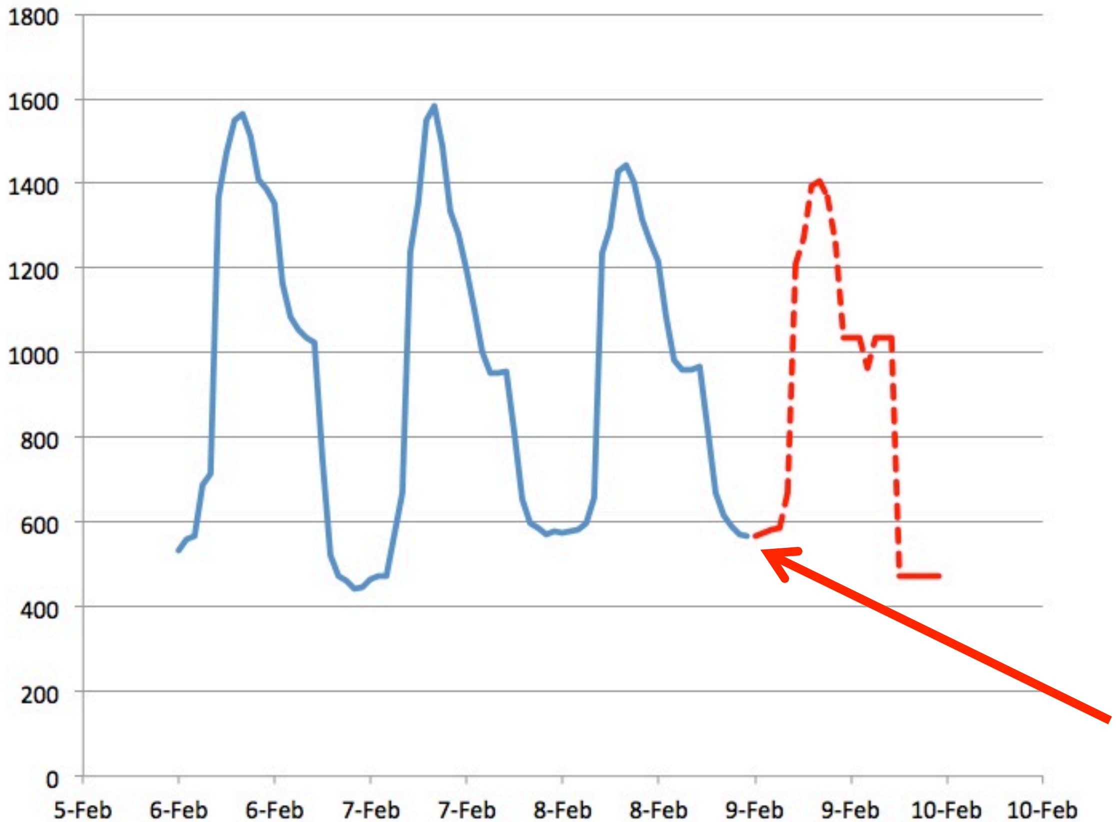
Customer areas

Energy pricing
Energy demand
Product forecasting
Ad network return
Machine efficiency
Anomaly detection
Server loads

Factory Energy Profile

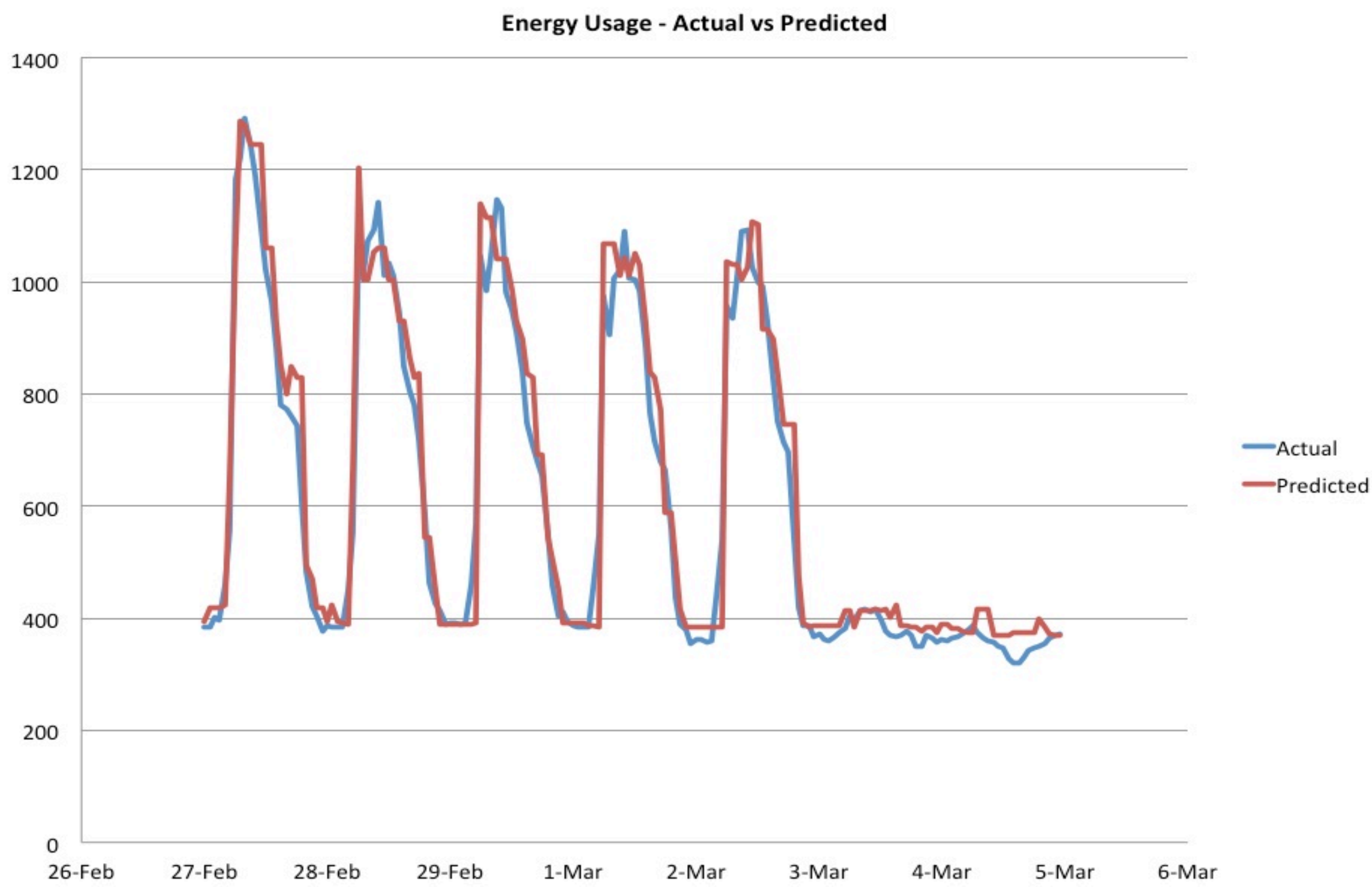


Customer need

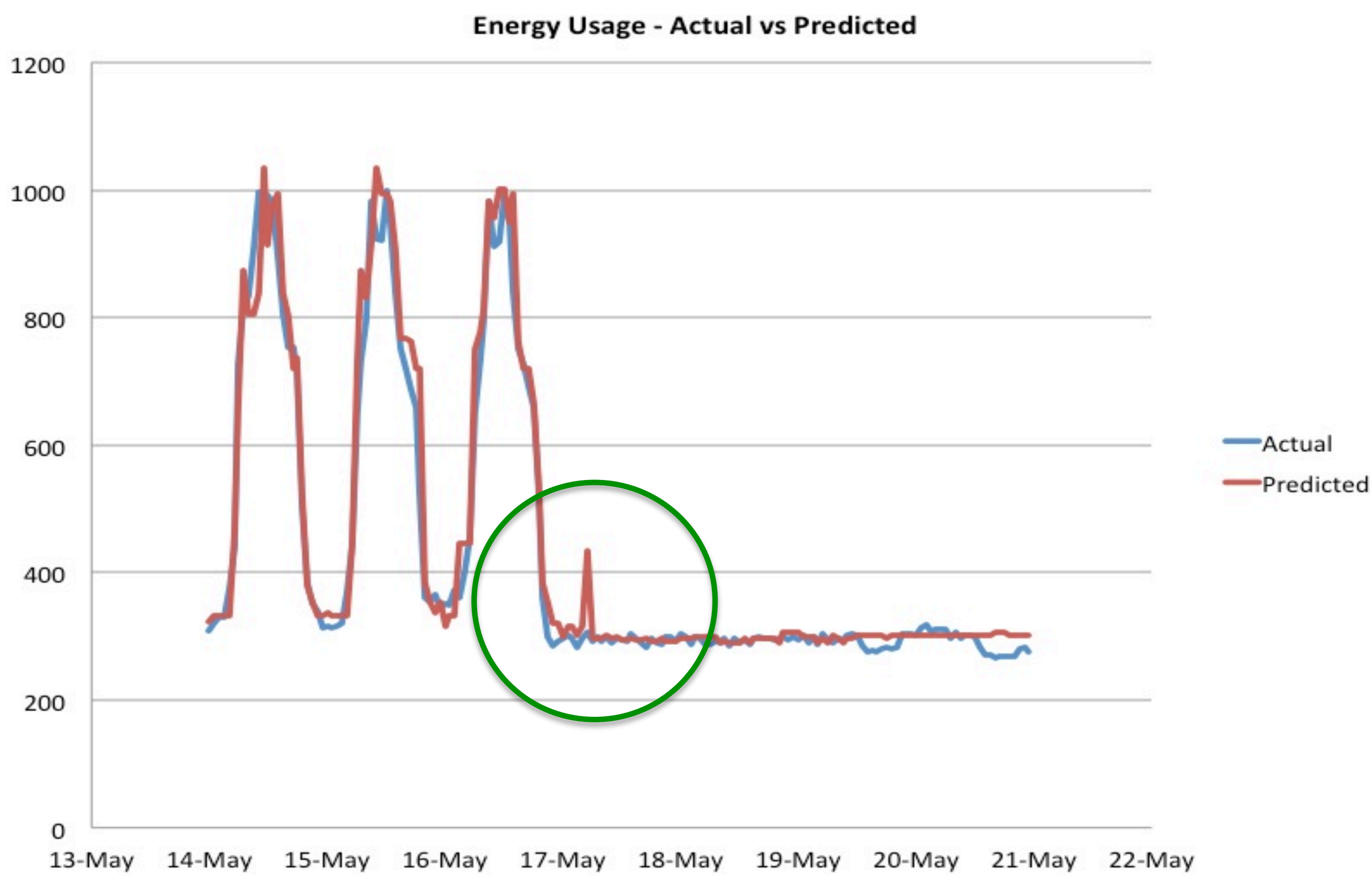


At midnight, make predictions for next 24 hours

Predictions and Actuals



Predictions and Actuals II

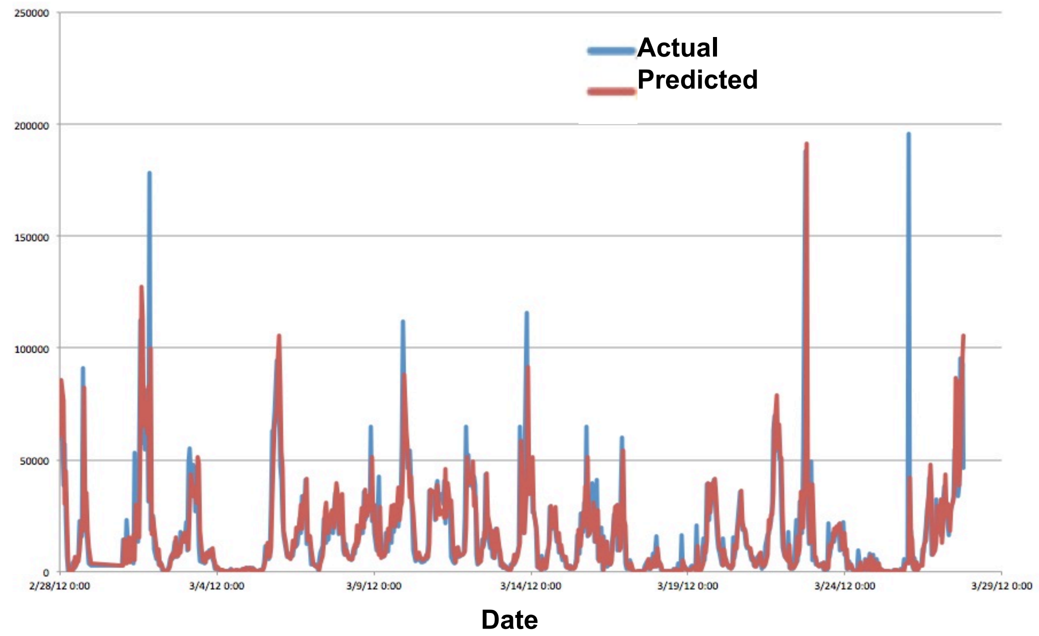


Managing Server Capacity

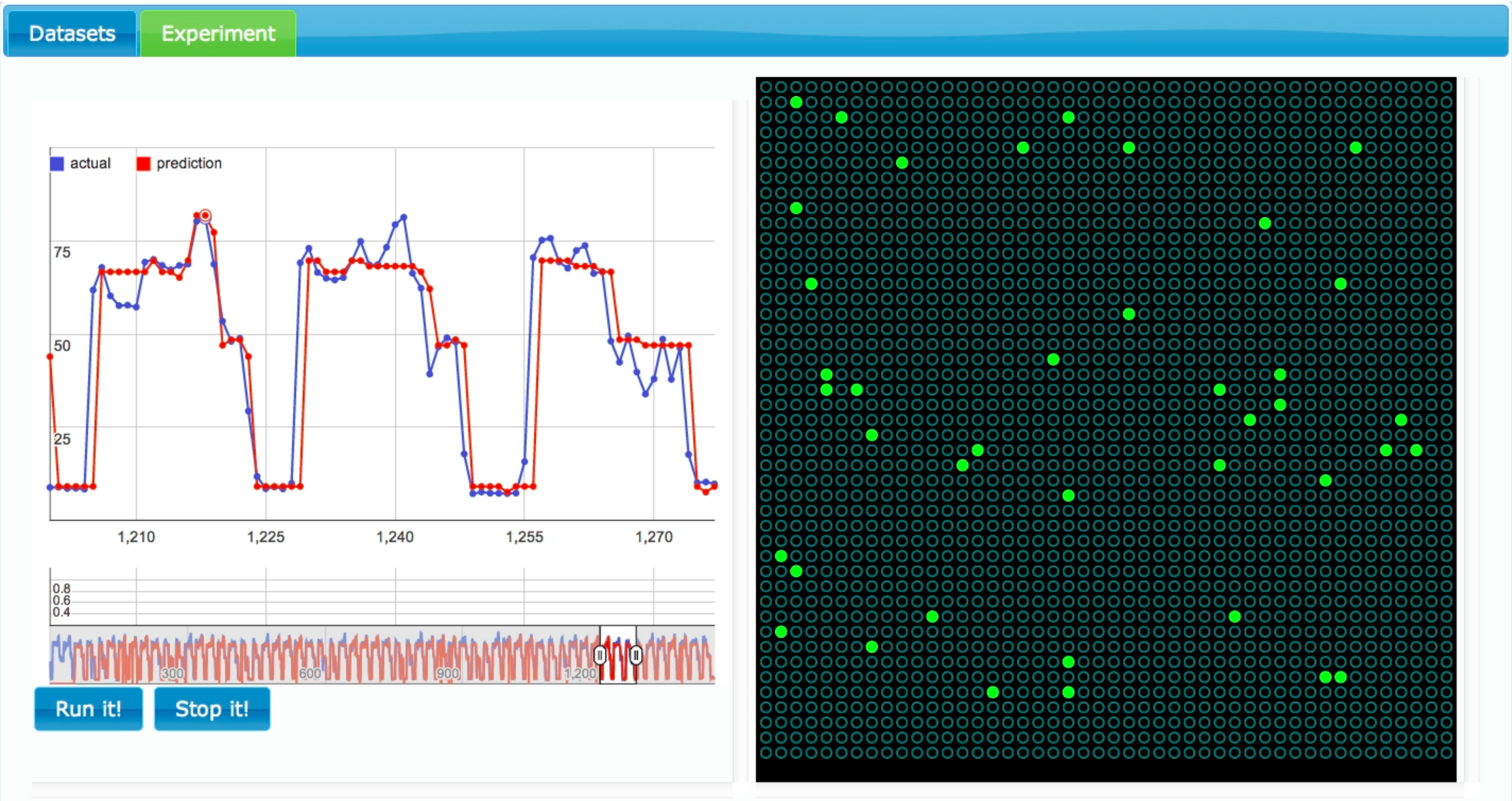
Grok used to predict server demand

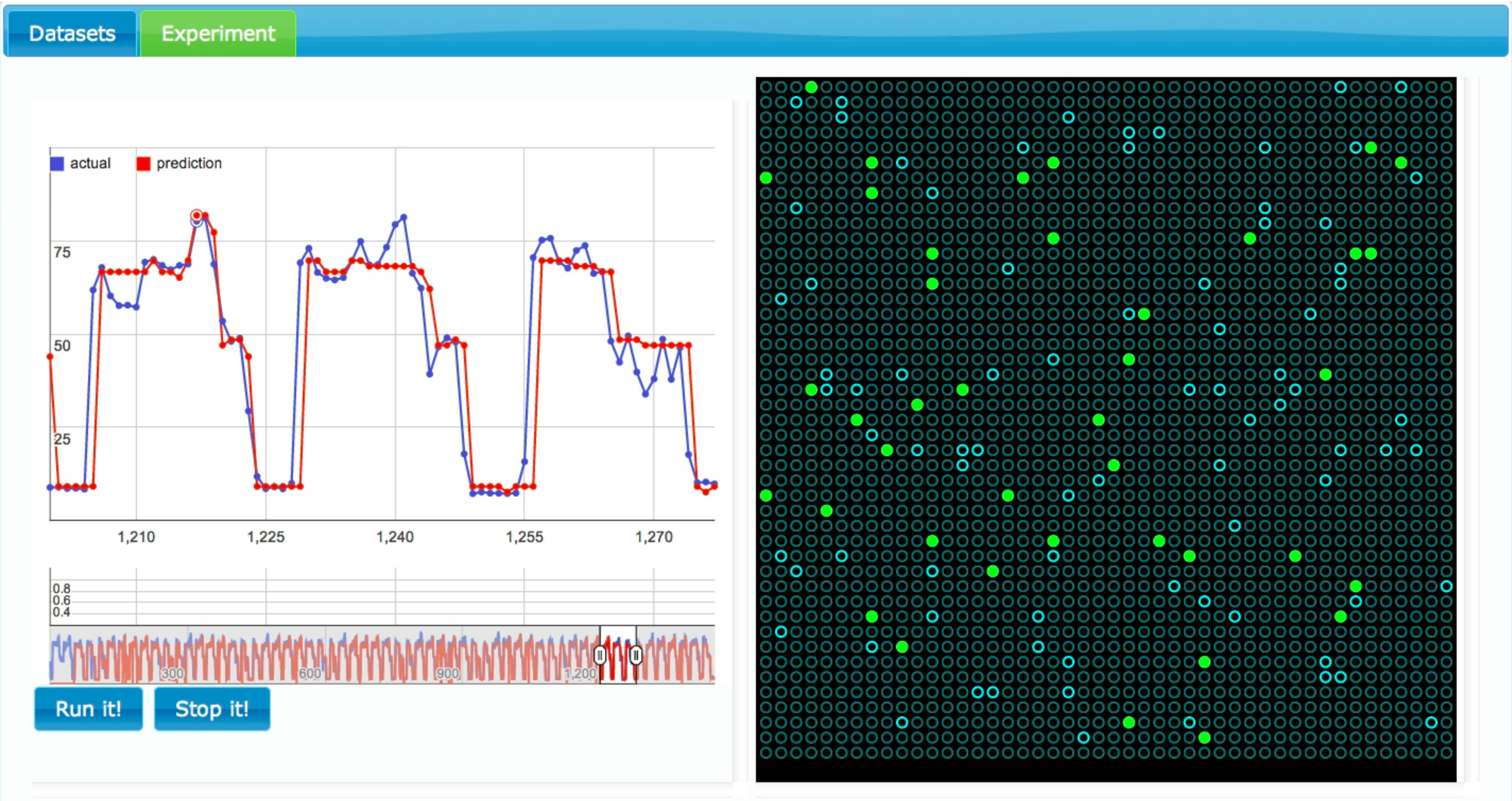
Used to provision instances ahead of time

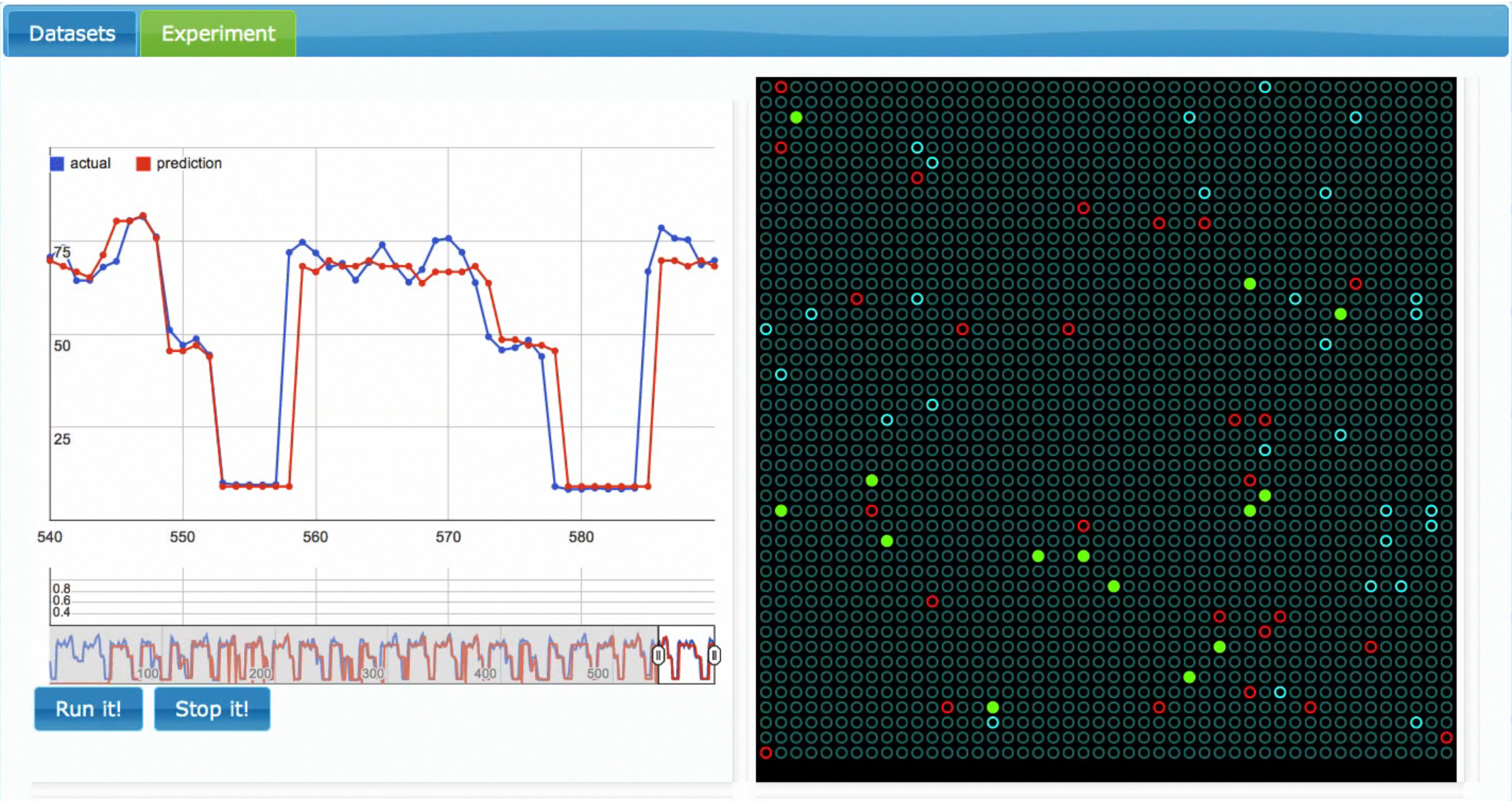
Results show approximately 15% reduction in AWS cost



Incoming server demand, Actual vs. Predicted


Datasets
Experiment
actual
prediction
75
50
25
1,210
1,225
1,240
1,255
1,270
0.8
0.6
0.4
300
600
900
1,200
Run it!
Stop it!


Datasets
Experiment
actual
prediction
75
50
25
1,210
1,225
1,240
1,255
1,270
0.8
0.6
0.4
300
600
900
1,200
Run it!
Stop it!

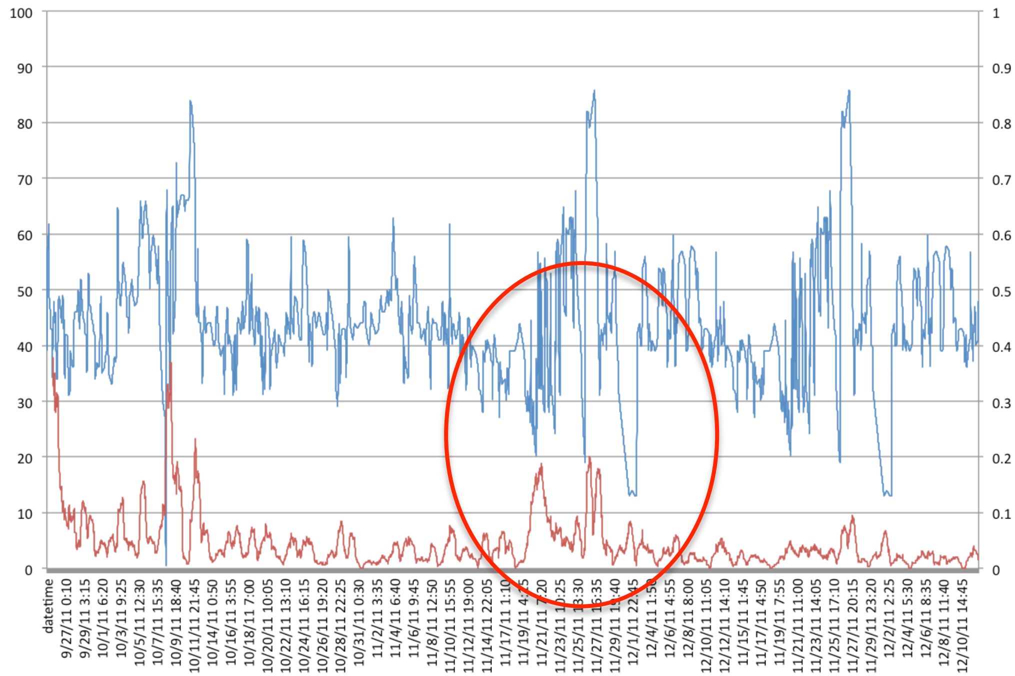

Datasets
Experiment
actual
prediction
75
50
25
540
550
560
570
580
0.8
0.6
0.4
100
200
300
400
500
Run it!
Stop it!

Predictive Maintenance

Grok used to detect anomalies in gear bearing temperature

Can detect anomalies based on temporal characteristics

Can be used to proactively optimize maintenance schedules



Gear bearing temperature & Grok Anomaly Score

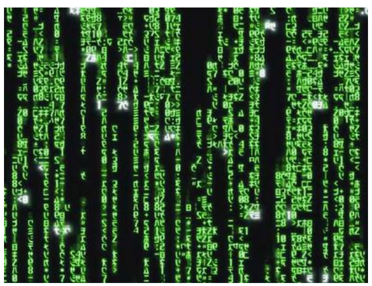
Open Source Project

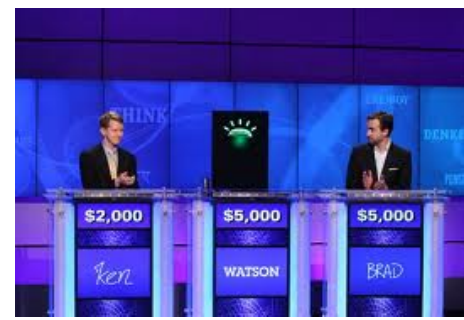
Academic and industrial interest in algorithms
Seven volunteer translations of white paper
Multiple independent implementations

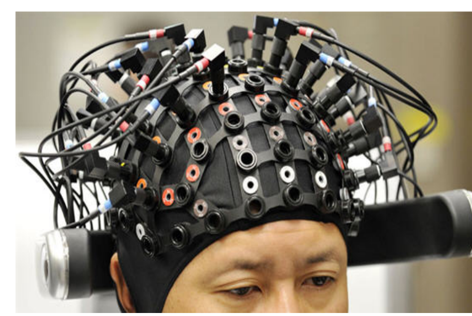
Open source of algorithm source code
"NuPIC"
GPLv3
Launch July 2013 OSCon
Prelaunch hackathon June 21

Goal: Catalyst for machine intelligence

Future of Machine Intelligence



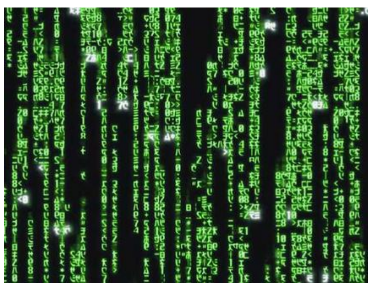


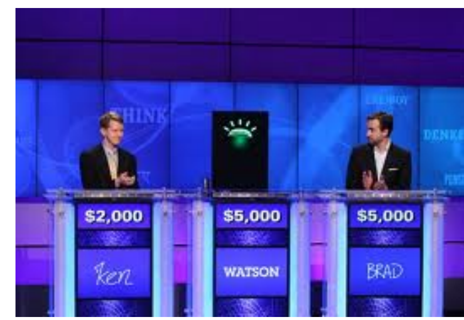


Future of Machine Intelligence

Definite

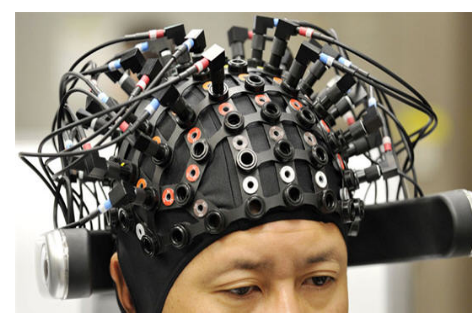
- Faster, Bigger
- Super senses
- Fluid robotics
- Distributed hierarchy





Maybe

- Humanoid robots
- Computer/Brain interfaces for all

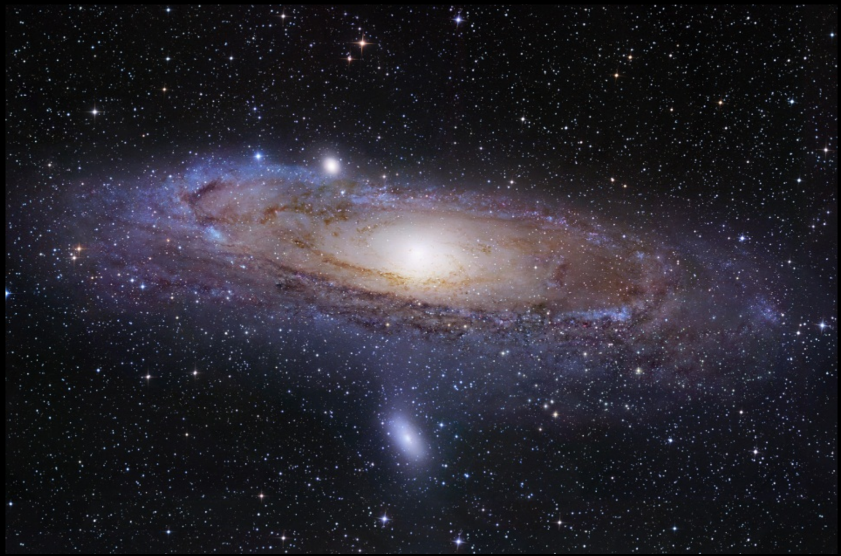


Not

- Uploaded brains
- Evil robots
- Friendly uses only

Why Machine Intelligence?

Live better



Learn more

Thank You