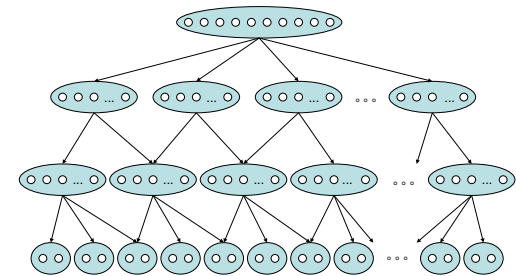


# Deep Restricted Bayesian Network BESOM

2018-02-27

Yuuji Ichisugi

Artificial Intelligence Research Center (AIRC),  
National Institute of Advanced Industrial Science  
and Technology (AIST), Japan



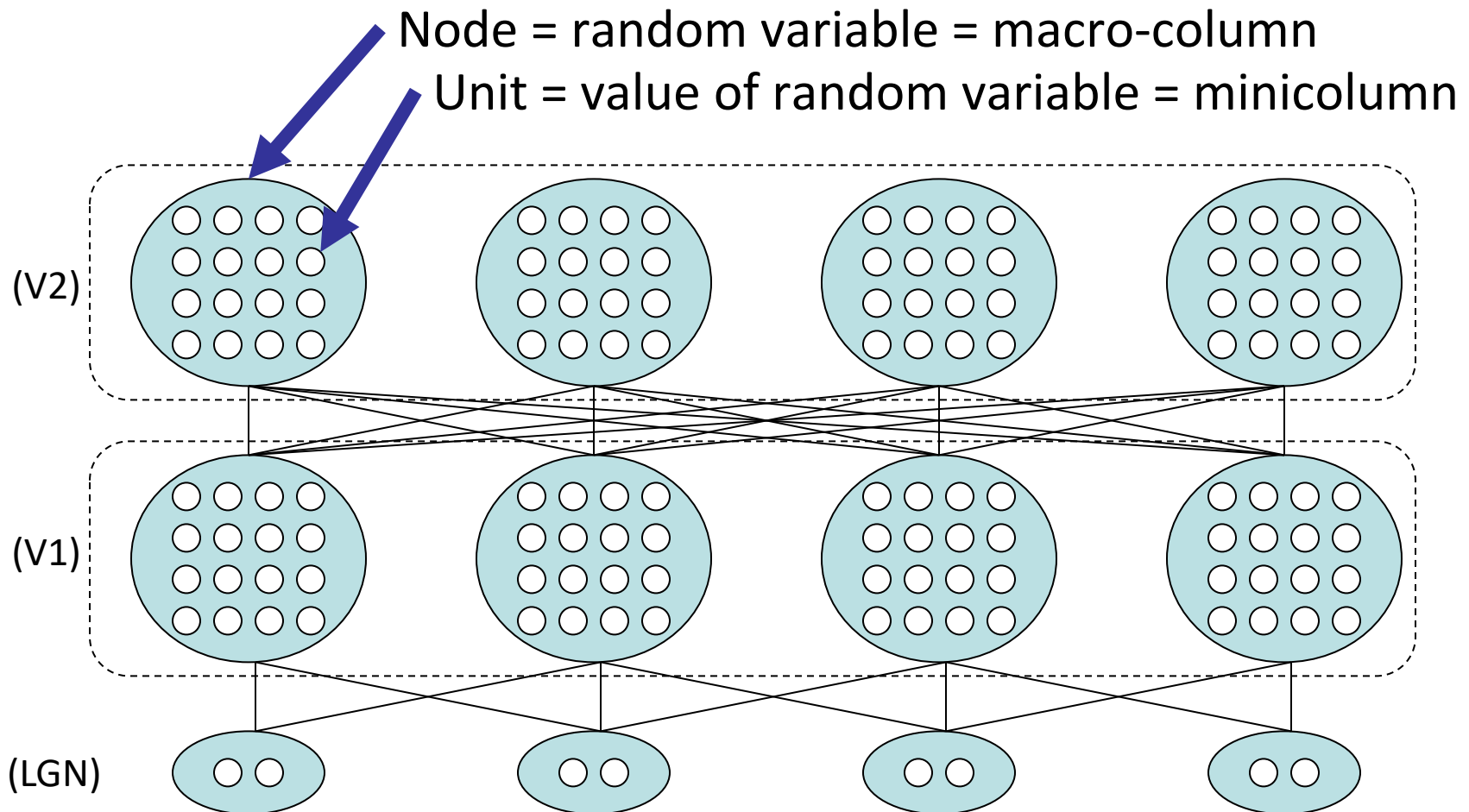
# BESOM (Bidirectional Self Organizing Maps) [Ichisug 2007]

- A computational model of the cerebral cortex
  - **A model of column network**, *not* spiking neurons
- Design goals:
  - **Scalability** of computation
  - **Usefulness** as a machine learning system
  - **Plausibility** as a neuroscientific model
- As a long-term goal, we aim to reproduce functions of such as the visual areas and the language areas using this cerebral cortex model.

# Architecture of BESOM model

**Recognition step:** The entire network behaves like a **Bayesian network**.

**Learning step:** Each node behaves like a **Self-organizing map**.



# Outline

- Bayesian networks and the cerebral cortex
- BESOM Ver.3 and robust pattern recognition
- Toward BESOM Ver.4

# Models of visual cortex based on Bayesian networks

- Various functions, illusions, neural responses and anatomical structure of the visual cortex were reproduced by Bayesian network models.
  - [Tai Sing Lee and Mumford 2003]
  - [George and Hawkins 2005]
  - [Rao 2005]
  - [Ichisugi 2007]
  - [Litvak and Ullman 2009]
  - [Chikkerur, Serre, Tan and Poggio 2010]
  - [Hosoya 2012]
  - ...

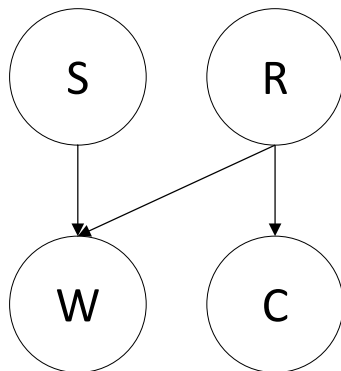
**The visual cortex seems to be a huge Bayesian network with layered structure like Deep Neural Networks.**

# What is Bayesian networks?

## – Efficient and expressive data structure of probabilistic knowledge [Perl 1988]

- Various probabilistic inference can be executed efficiently if a joint probability table can be factored into small **conditional probability tables (CPTs)**.

$$P(S, W, R, C) = P(W | S, R)P(C | R)P(S)P(R)$$



CPTs

P(S=yes)
0.2

P(R=yes)
0.02

S	R	P(W=yes S,R)
no	no	0.12
no	yes	0.8
yes	no	0.9
yes	yes	0.98

R	P(C=yes R)
no	0.3
yes	0.995

# Loopy Belief Propagation

[Murphy, Weiss, Jordan 1999]

- Efficient approximate inference algorithm
  - Iterative algorithm with **local and asynchronous computation**, like brain.
  - Although there is no guarantee of convergence, it is empirically accurate.

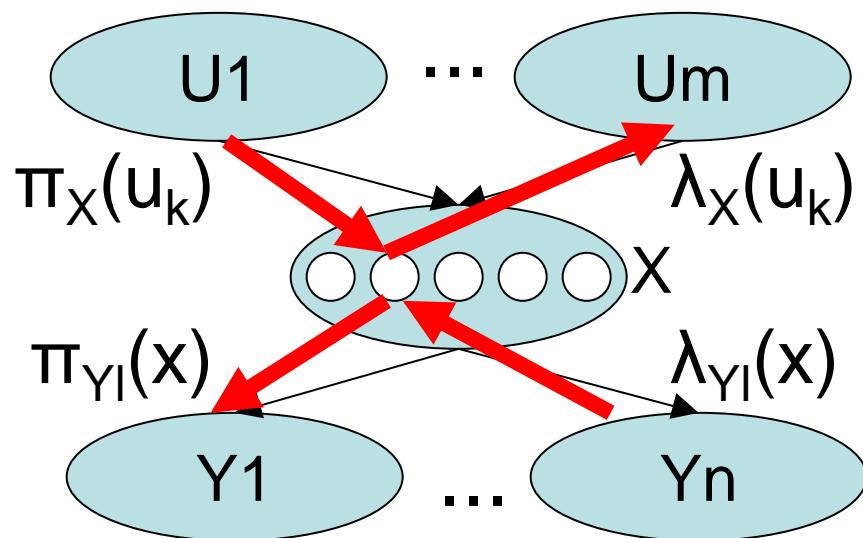
$$BEL(x) = \alpha \lambda(x) \pi(x)$$

$$\pi(x) = \sum_{u_1, \dots, u_m} P(x | u_1, \dots, u_m) \prod_k \pi_X(u_k)$$

$$\lambda(x) = \prod_l \lambda_{Y_l}(x)$$

$$\pi_{Y_l}(x) = \pi(x) \prod_{j \neq l} \lambda_{Y_j}(x)$$

$$\lambda_X(u_k) = \sum_x \lambda(x) \sum_{u_1, \dots, u_m / u_k} P(x | u_1, \dots, u_m) \prod_{i \neq k} \pi_X(u_i)$$



# Belief propagation and micro circuit of cerebral cortex

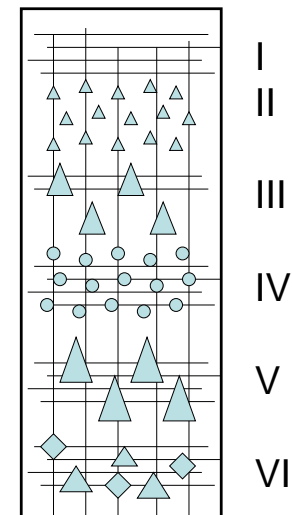
- The similarity between belief propagation and the six-layer structure of the cerebral cortex has been pointed out many times.

[George and Hawkins 2005]

[Ichisugi 2007]

[Rohrbein, Eggert and Korner 2008]

[Litvak and Ullman 2009]





# Approximate Belief Propagation

[Ichisugi 2007]

$$\mathbf{l}_{XY}^{t+1} = \mathbf{z}_Y^t + \mathbf{W}_{XY} \mathbf{o}_Y^t$$

$$\mathbf{o}_X^{t+1} = \bigotimes_{Y \in \text{children}(X)} \mathbf{l}_{XY}^{t+1}$$

$$\mathbf{k}_{UX}^{t+1} = \mathbf{W}_{UX}^T \mathbf{b}_U^t$$

$$\mathbf{p}_X^{t+1} = \sum_{U \in \text{parents}(X)} \mathbf{k}_{UX}^{t+1}$$

$$\mathbf{r}_X^{t+1} = \mathbf{o}_X^{t+1} \otimes \mathbf{p}_X^{t+1}$$

$$Z_X^{t+1} = \sum_i (\mathbf{r}_X^{t+1})_i \quad (= \|\mathbf{r}_X^{t+1}\|_1 = \mathbf{o}_X^{t+1} \bullet \mathbf{p}_X^{t+1})$$

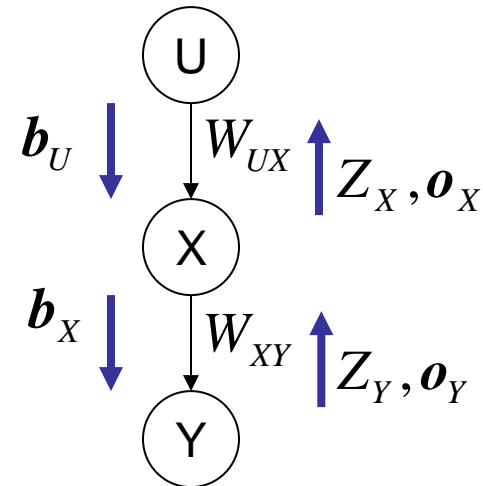
$$\mathbf{z}_X^{t+1} = (Z_X^{t+1}, Z_X^{t+1}, \mathbf{L}, Z_X^{t+1})^T$$

$$\mathbf{b}_X^{t+1} = (1/Z_X^{t+1}) \mathbf{r}_X^{t+1}$$

where  $\mathbf{x} \otimes \mathbf{y} = (x_1 y_1, x_2 y_2, \mathbf{L}, x_n y_n)^T$

Approximates Pearl's algorithm [Pearl 1988] with some assumptions.

$$P(x|u_1, \dots, u_m) = \frac{1}{m} \sum_{k=1}^m P(x|u_k)$$



Yuuji ICHISUGI, "The cerebral cortex model that self-organizes conditional probability tables and executes belief propagation", In proc. of IJCNN2007, Aug 2007.

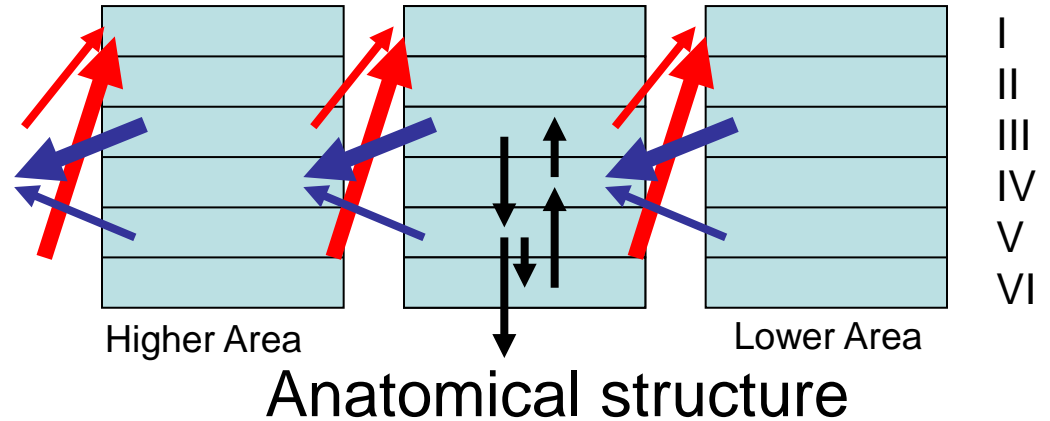
# Similarity in information flow

[Gilbert 1983]

[Pandya and Yeterian 1985]

Gilbert, C.D., Microcircuitry of the visual-cortex, Annual review of neuroscience, 6: 217-247, 1983.

Pandya, D.N. and Yeterian, E.H., Architecture and connections of cortical association areas. In: Peters A, Jones EG, eds. Cerebral Cortex (Vol. 4): Association and Auditory Cortices. New York: Plenum Press, 3-61, 1985.



$$\mathbf{l}_{XY}^{t+1} = \mathbf{z}_Y^t + \mathbf{W}_{XY} \mathbf{o}_Y^t$$

$$\mathbf{o}_X^{t+1} = \bigotimes_{Y \in \text{children}(X)} \mathbf{l}_{XY}^{t+1}$$

$$\mathbf{k}_{UX}^{t+1} = \mathbf{W}_{UX}^T \mathbf{b}_U^t$$

$$\mathbf{p}_X^{t+1} = \sum_{U \in \text{parents}(X)} \mathbf{k}_{UX}^{t+1}$$

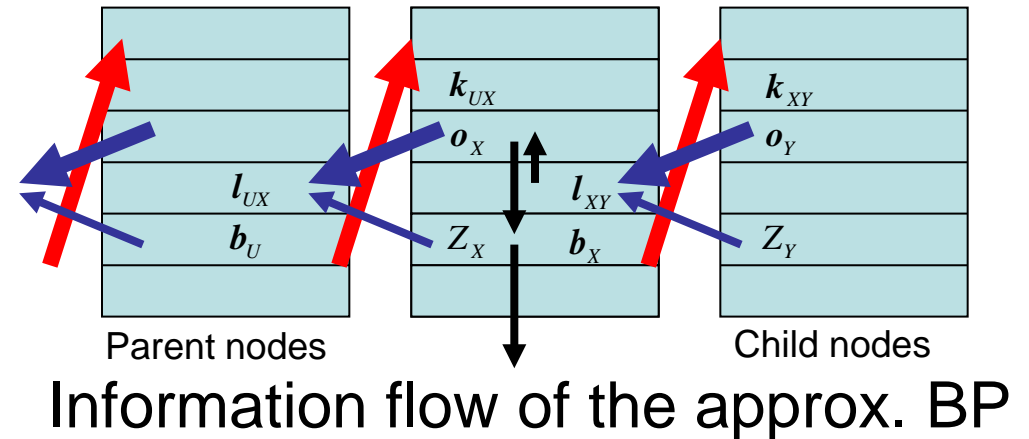
$$\mathbf{r}_X^{t+1} = \mathbf{o}_X^{t+1} \otimes \mathbf{p}_X^{t+1}$$

$$\mathbf{Z}_X^{t+1} = \sum_i (\mathbf{r}_X^{t+1})_i \quad (= \|\mathbf{r}_X^{t+1}\|_1 = \mathbf{o}_X^{t+1} \bullet \mathbf{p}_X^{t+1})$$

$$\mathbf{z}_X^{t+1} = (Z_X^{t+1}, Z_X^{t+1}, \mathbf{L}, Z_X^{t+1})^T$$

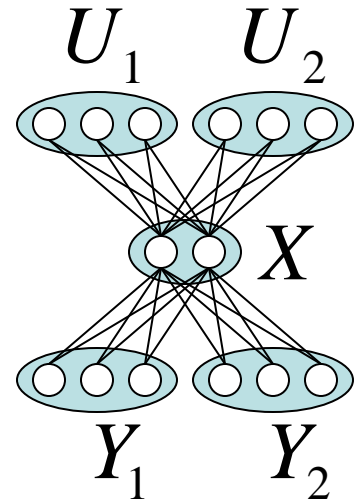
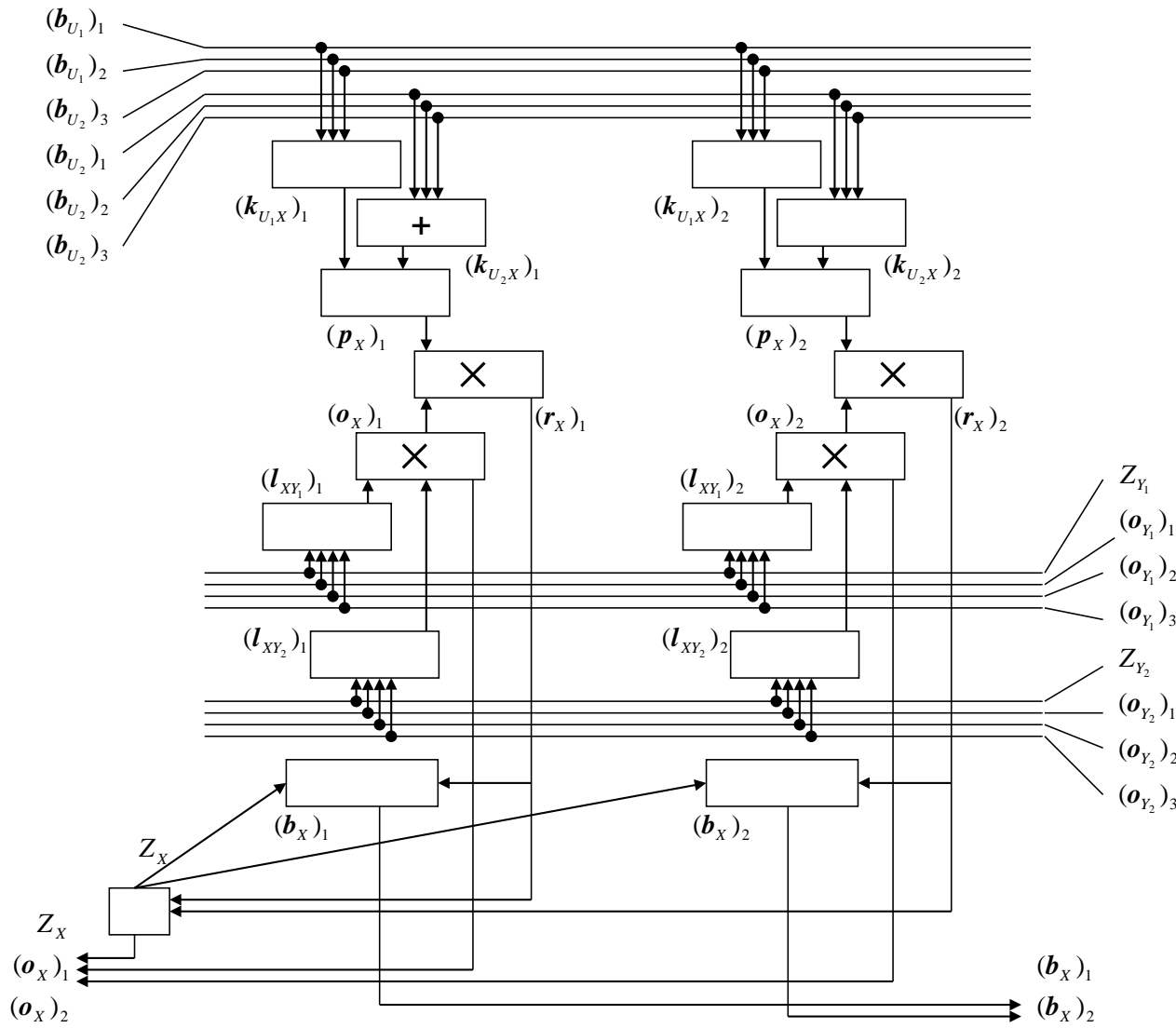
$$\mathbf{b}_X^{t+1} = (1/Z_X^{t+1}) \mathbf{r}_X^{t+1}$$

where  $\mathbf{x} \otimes \mathbf{y} = (x_1 y_1, x_2 y_2, \mathbf{L}, x_n y_n)^T$



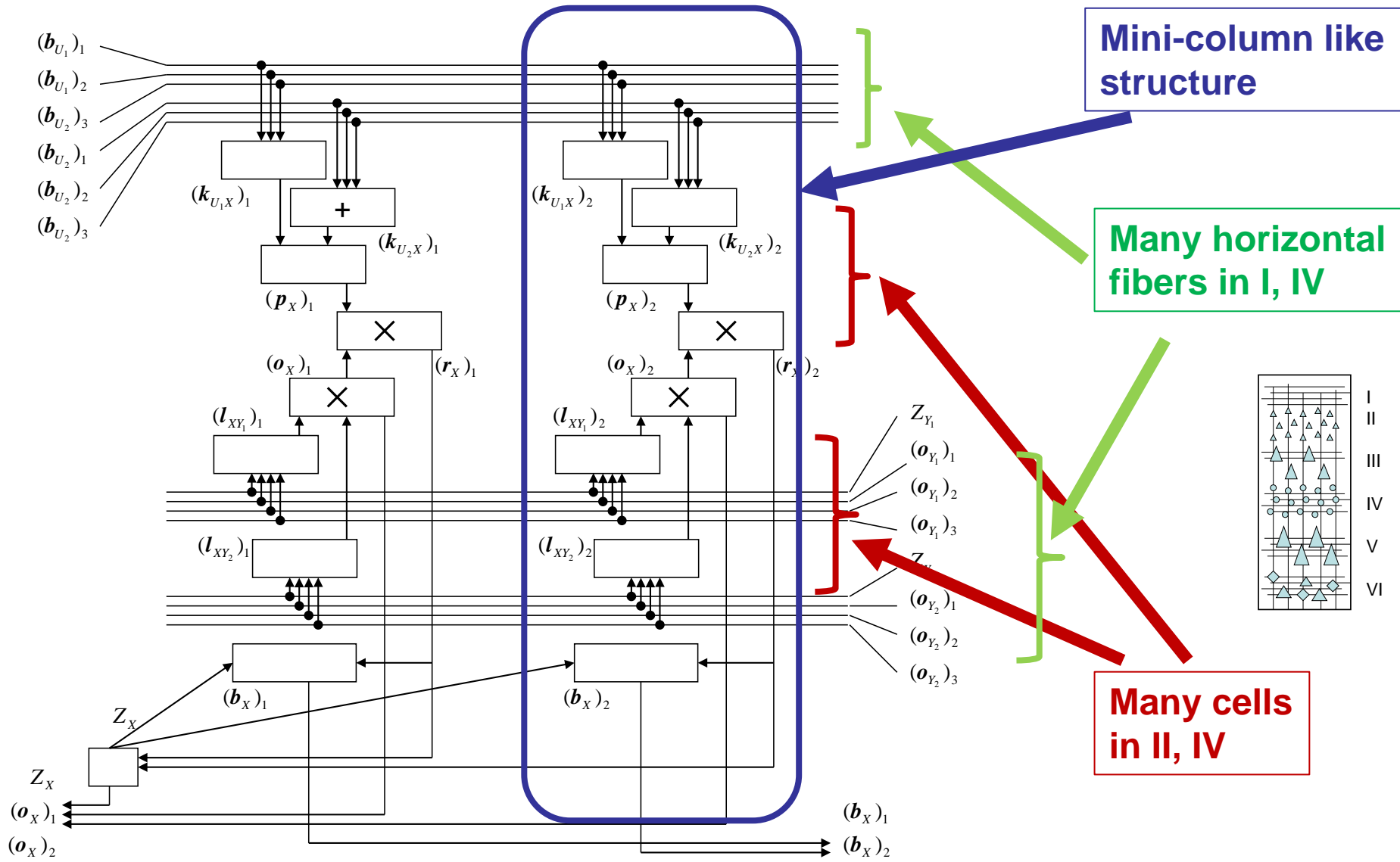
The intermediate variables of this algorithm can be assigned to each layer of the cerebral cortex without contradicting the known anatomical structure.

# Detailed circuit that calculates the approximate BP



The left circuit calculates values of two units,  $x_1$  and  $x_2$ , in node  $X$  in the above network.

# Correspondence with local cortical circuit



# Outline

- Bayesian networks and the cerebral cortex
- **BESOM Ver.3 and robust pattern recognition**
- Toward BESOM Ver.4

# Toward realization of the brain function

- If the cerebral cortex is a kind of Bayesian network, we should be able to reproduce function and performance of it using Bayesian networks.
  - As a first step, we aim to reproduce some part of the functions of the visual areas and the language areas.
  - Although there were some difficulties such as computational cost and local minimum problem, now they have been solved considerably.

# BESOM Ver.3.0 features

- Restricted Conditional Probability Tables:

$$P(x|u_1, \dots, u_m) = \frac{1}{m} \sum_{k=1}^m P(x|u_k)$$

- Scalable recognition algorithm OOBP [Ichisugi, Takahashi 2015]
- Regularization methods to avoid local minima
  - Win-rate and Lateral-inhibition penalty [Ichisugi, Sano 2016]
  - Neighborhood learning

$$\begin{aligned} \lambda_{Y_i}^{t+1}(x) &= \beta_2 \sum_{y_l} \lambda^t(y_l) (\pi^t(y_l) - \kappa_X^t(y_l) + w(y_l, x)) \\ \lambda^{t+1}(x) &= \prod_{i=1}^n \lambda_{Y_i}^{t+1}(x) \\ \pi_{Y_i}^{t+1}(x) &= \beta_1 \rho^{t+1}(x) / \lambda_{Y_i}^{t+1}(x) \\ \kappa_{U_k}^{t+1}(x) &= \sum_{u_k} w(x, u_k) \pi_X^t(u_k) \\ \pi^{t+1}(x) &= \sum_{k=1}^m \kappa_{U_k}^{t+1}(x) \\ \rho^{t+1}(x) &= \lambda^{t+1}(x) \pi^{t+1}(x) \\ BEL^{t+1}(x) &= \alpha \rho^{t+1}(x) \end{aligned}$$

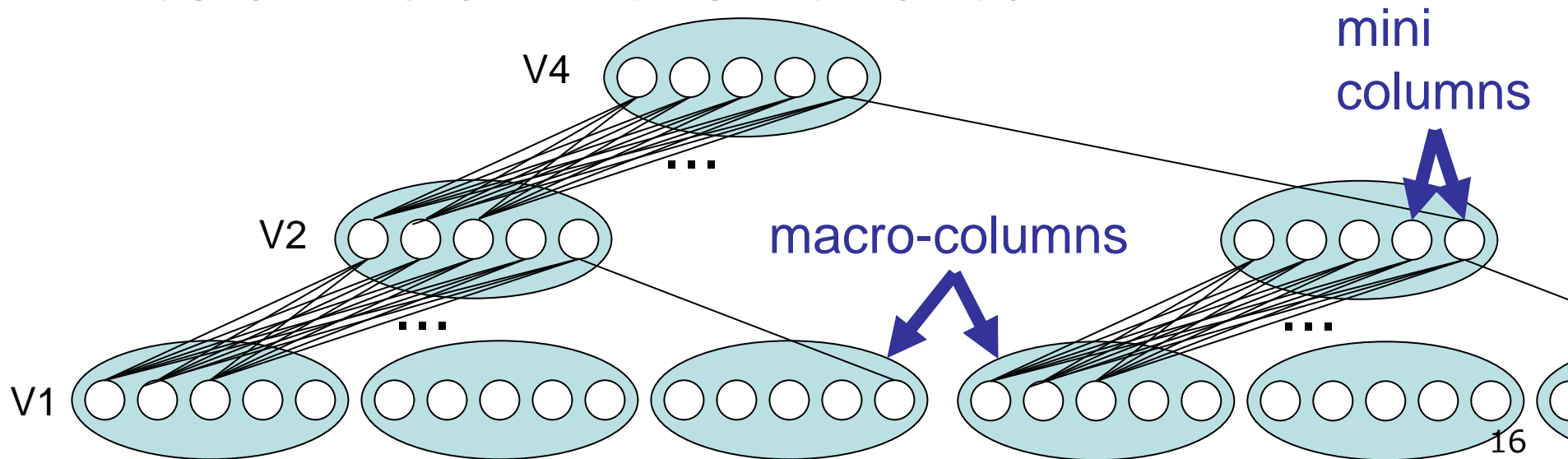
Computational amount of one step of iteration of OOBP is linear to the number of edges of the network.

Yuuji Ichisugi and Naoto Takahashi,  
An Efficient Recognition Algorithm for Restricted  
Bayesian Networks, In proc. of IJCNN 2015.

Yuuji Ichisugi and Takashi Sano,  
Regularization Methods for the Restricted Bayesian  
Network BESOM, In Proc. of ICONIP2016, Part I, LNCS  
9947, pp.290--299, 2016.

The design of BESOM is motivated by two neuroscientific facts.

1. Each macro-column seems to be like a SOM.
2. A macro-column at a upper area receives the output of the macro-columns at the lower area.



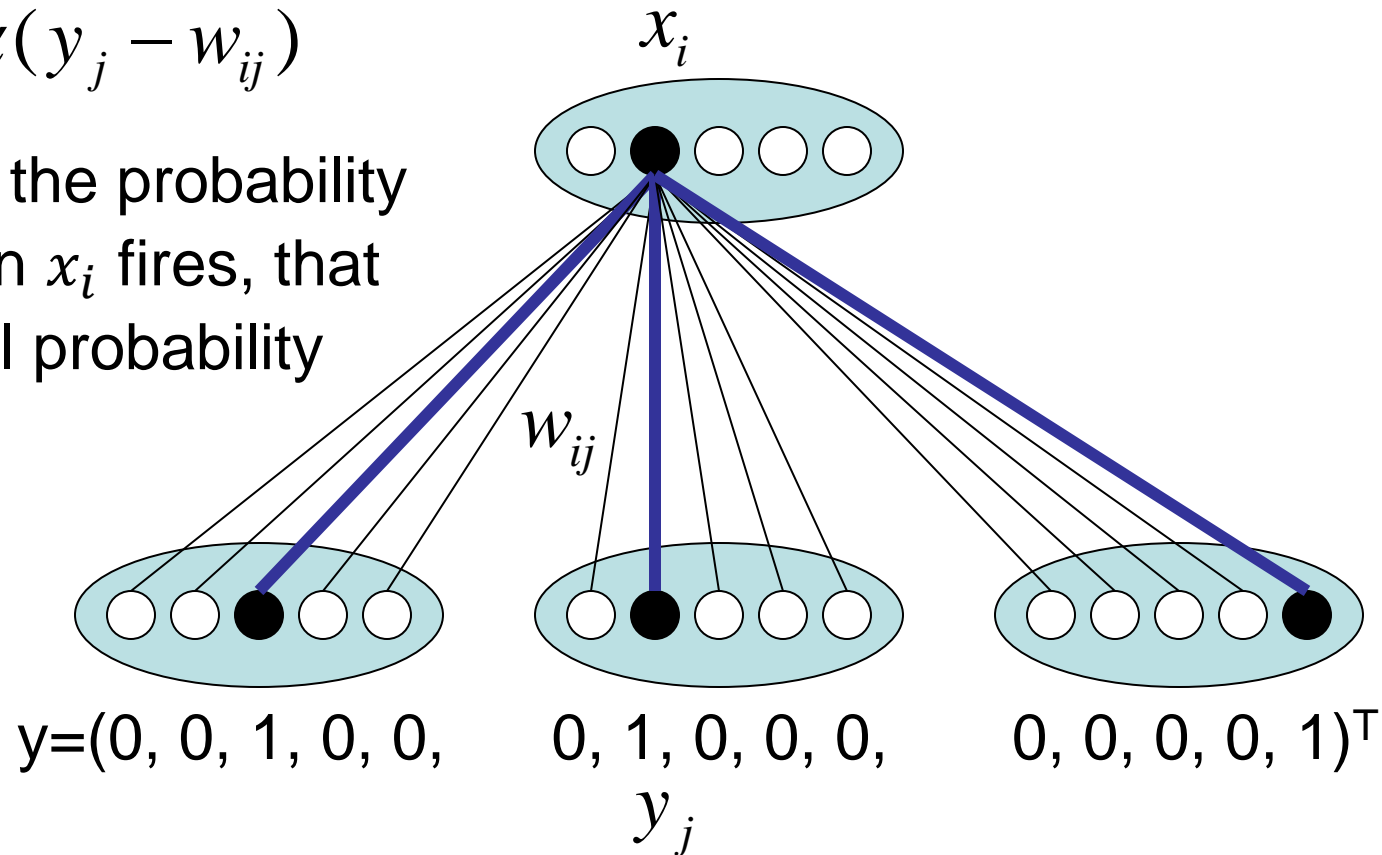


# If a SOM receives input from other SOMs, they naturally become a Bayesian Network

Learning rule (without neighborhood learning)

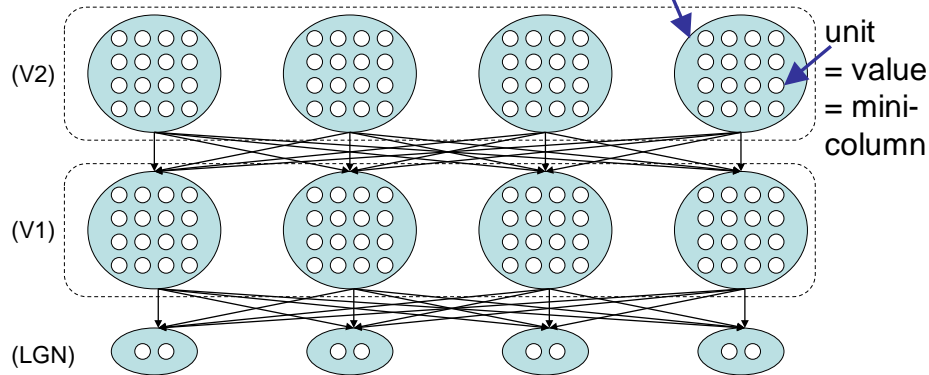
$$w_{ij} \leftarrow w_{ij} + \alpha(y_j - w_{ij})$$

$w_{ij}$  converges to the probability that  $y_j$  fires when  $x_i$  fires, that is, the conditional probability  $P(y_j|x_i)$ .



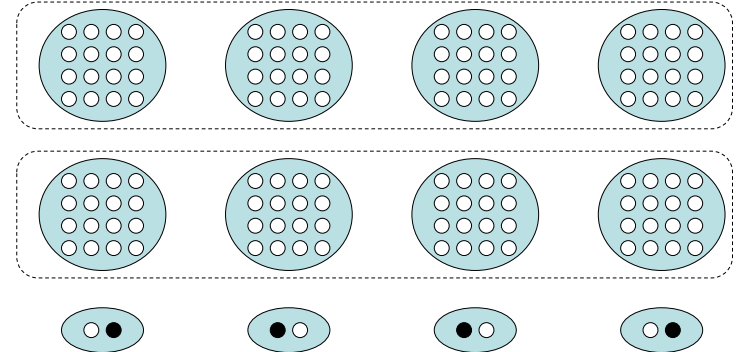
Node = random variable = macro-column

unit  
= value  
= mini-column



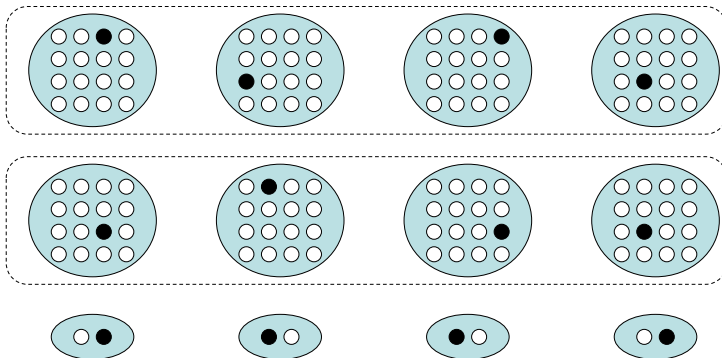
Connection weights  
= Conditional probabilities  
= synapse weights

## Input



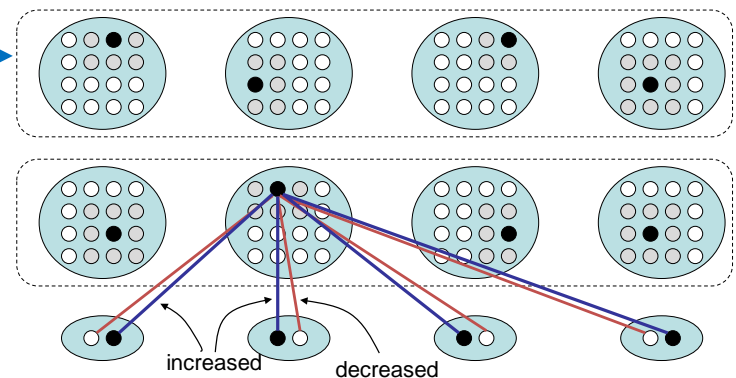
Input (observed data) is given  
at the lowest layer.

## Recognition



Find the values with the highest posterior  
probability. (MAP)

## Learning

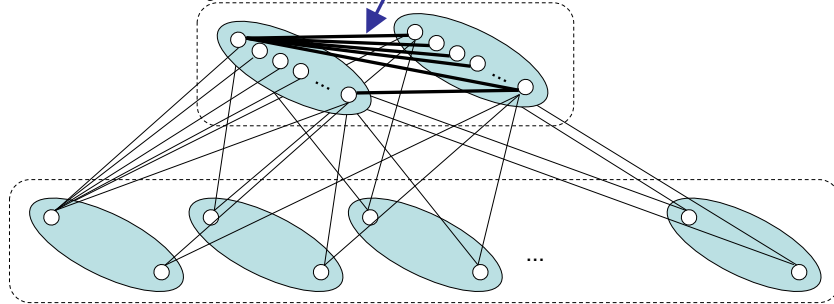


Updates connection weights  
with Hebb's rule.

## Connections of long distance lateral inhibition

Hidden layer

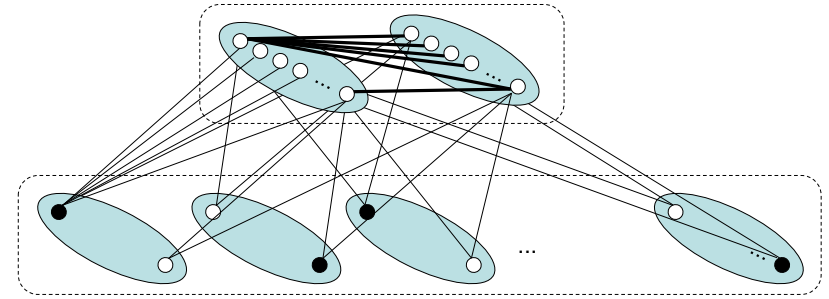
[Ichisugi, Sano 2016]



Input layer

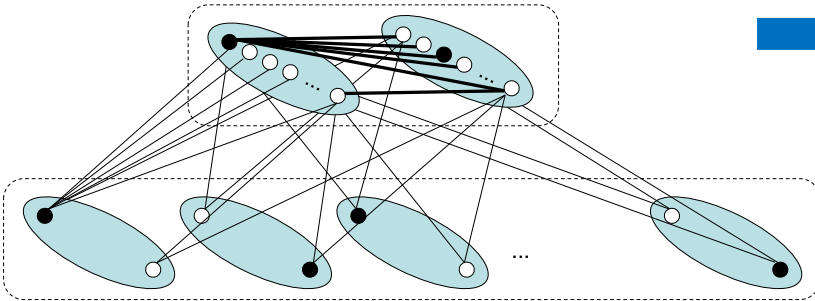
Yuuji Ichisugi and Takashi Sano,  
Regularization Methods for the Restricted Bayesian Network BESOM,  
In Proc. of ICONIP2016, Part I, LNCS 9947, pp.290--299, 2016.

Input



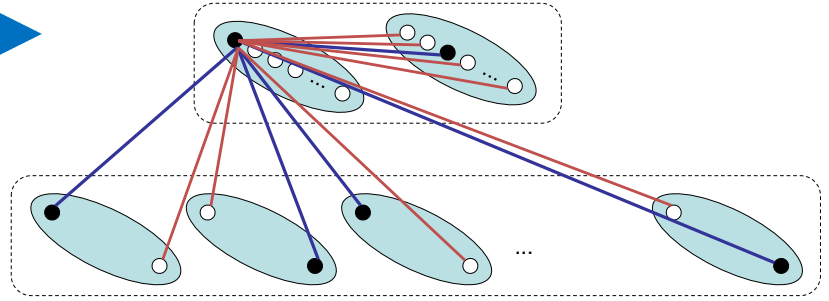
Input (observed data) is given at the lowest layer.

Recognition



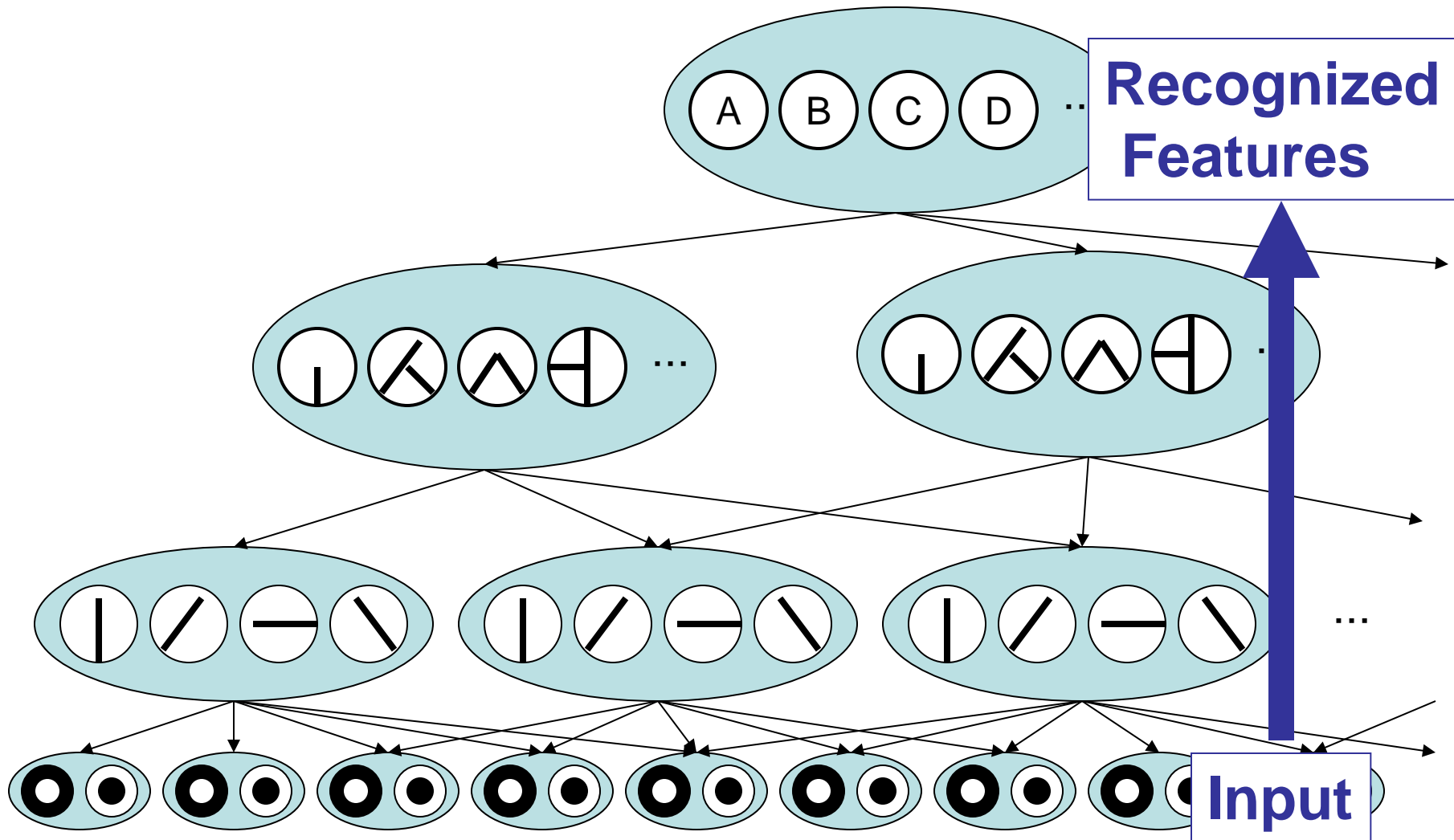
Units that receive strong lateral inhibition are less likely to become winners.

Learning

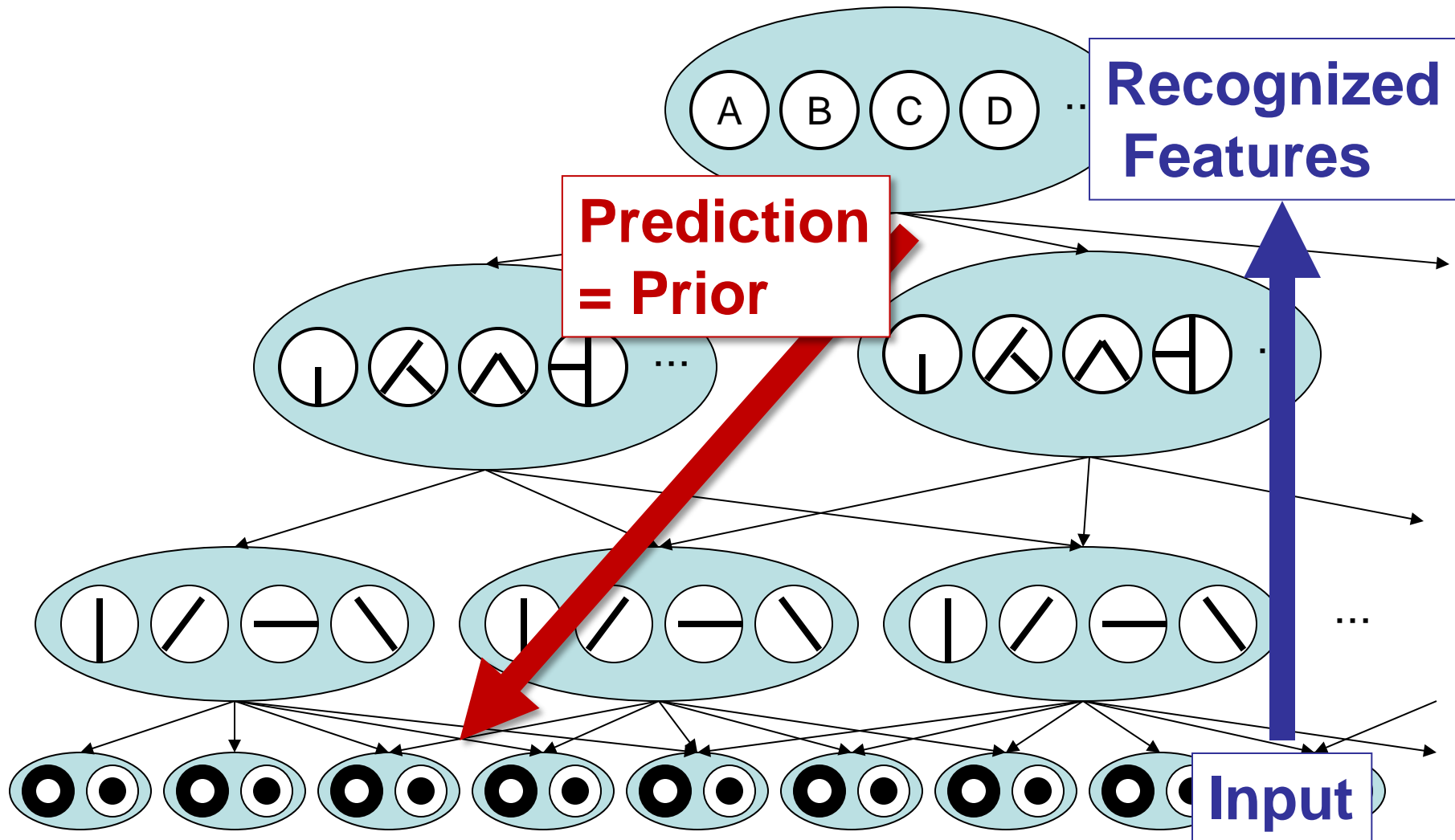


Updates connection weights with Hebb's rule.

BESOM can be used as if  
it were a Deep neural network.

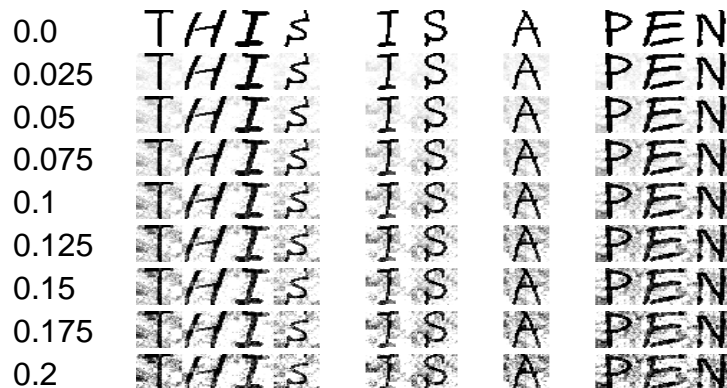
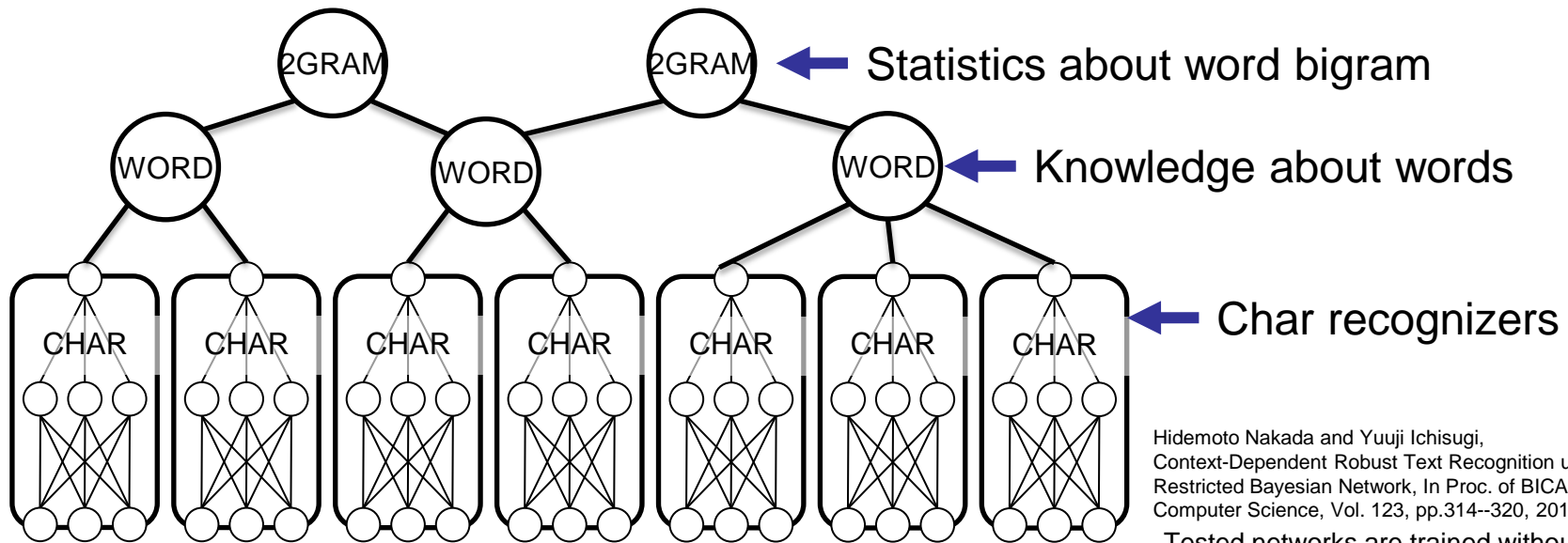


BESOM can be used as if  
it were a *bidirectional* Deep neural  
network.



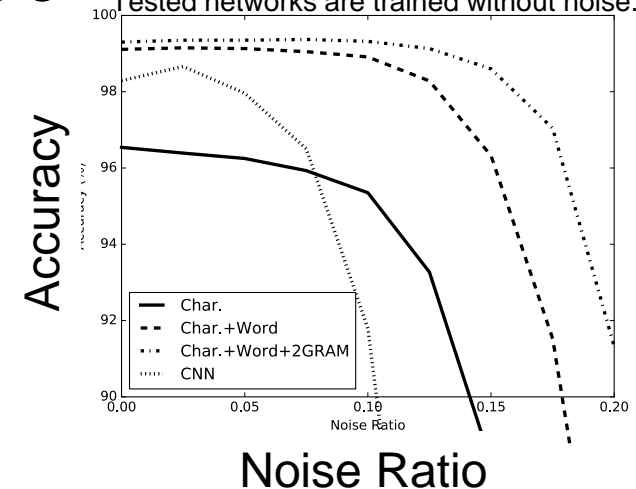
# Robust character recognition utilizing context information

[Nakada, Ichisugi 2018]



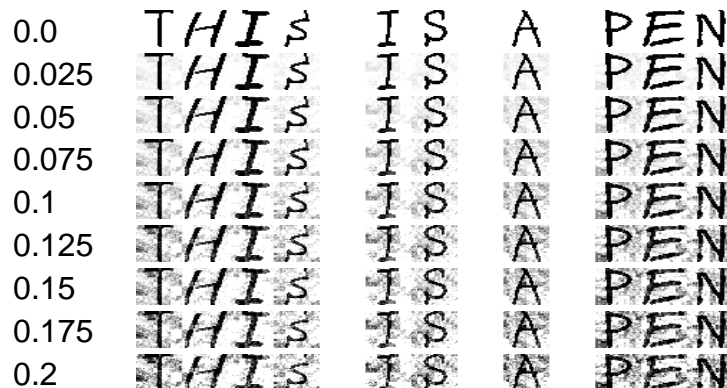
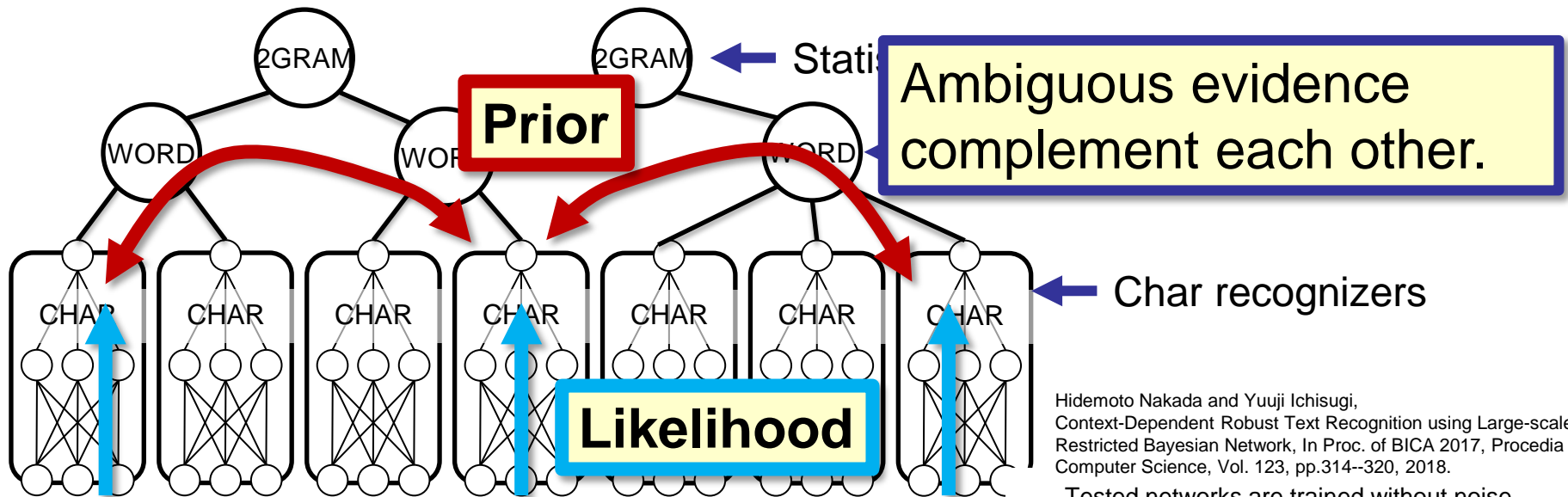
Hidemoto Nakada and Yuuji Ichisugi,  
Context-Dependent Robust Text Recognition using Large-scale  
Restricted Bayesian Network, In Proc. of BICA 2017, Procedia  
Computer Science, Vol. 123, pp.314–320, 2018.

Tested networks are trained without noise.



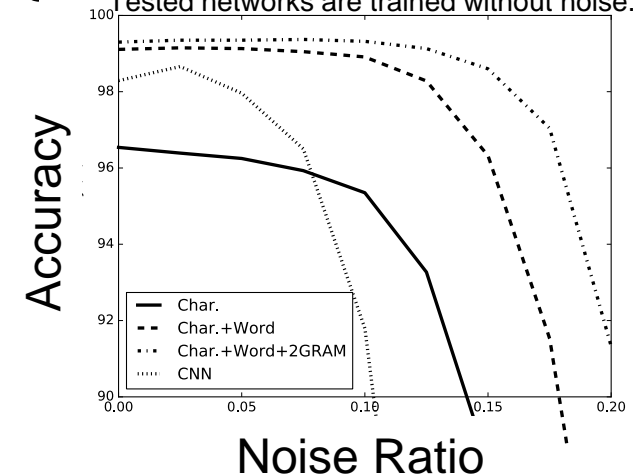
# Robust character recognition utilizing context information

[Nakada, Ichisugi 2018]



Hidemoto Nakada and Yuuji Ichisugi,  
Context-Dependent Robust Text Recognition using Large-scale  
Restricted Bayesian Network, In Proc. of BICA 2017, Procedia  
Computer Science, Vol. 123, pp.314–320, 2018.

Tested networks are trained without noise.



# Outline

- Bayesian networks and the cerebral cortex
- BESOM Ver.3 and robust pattern recognition
- **Toward BESOM Ver.4**



# Problem of BESOM Ver.3

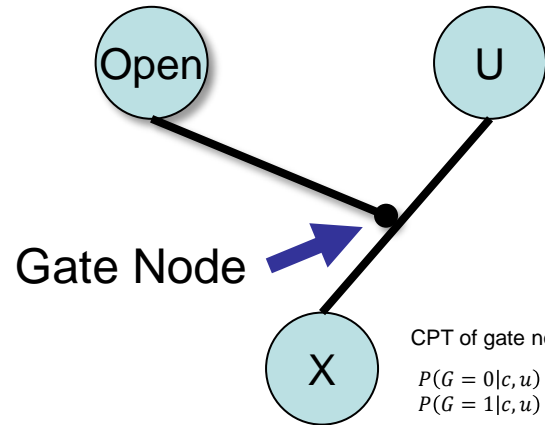
- Recognition and Learning is fast but accuracy is not good enough, probably because conditional probability tables are too restricted.

$$P(x|u_1, \dots, u_m) = \frac{1}{m} \sum_{k=1}^m P(x|u_k)$$

- We are now investigating new conditional probability table models (BESOM Ver.4):
  - Noisy-OR model [Pearl 1988] and **Gate-nodes**.
  - More expressive and fast enough.

# Gate Nodes

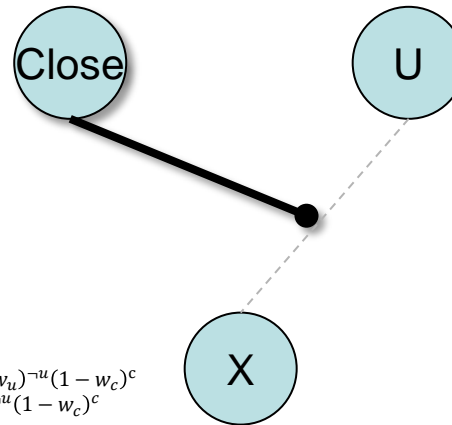
Control Node



CPT of gate nodes

$$P(G = 0|c, u) = 1 - (1 - w_u)^{-u}(1 - w_c)^c$$

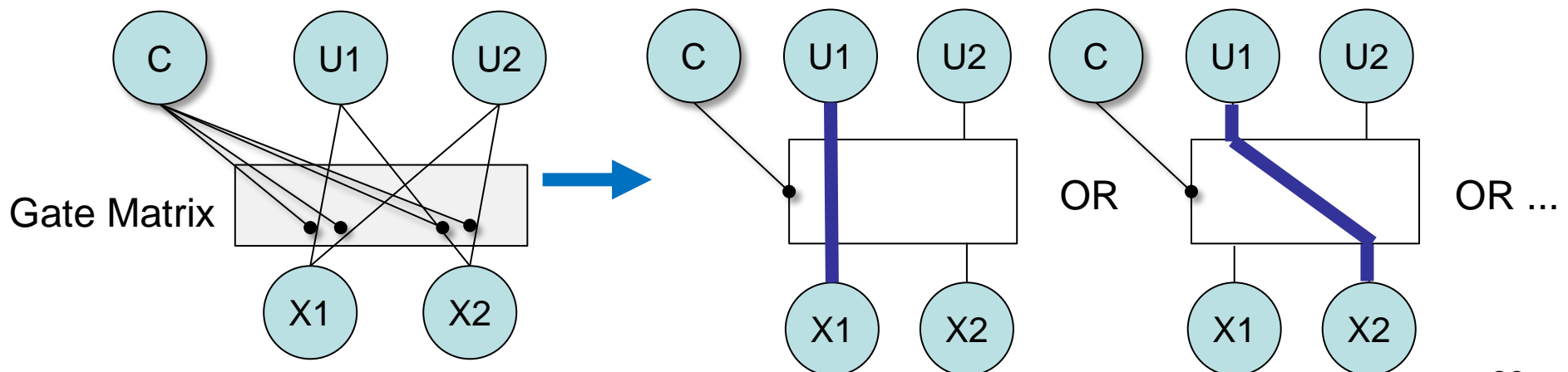
$$P(G = 1|c, u) = (1 - w_u)^{-u}(1 - w_c)^c$$



If Open,  
U and X are connected.  
If Close, disconnected.

Like inhibitory connections  
on dendrites.

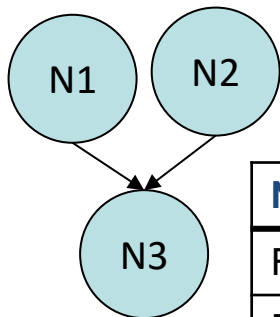
Using matrix of gates, Control node can  
control connections between nodes.



# Prototyping by Quasi Bayesian Networks

- We are designing prototype models of the visual areas and the language areas using *Quasi Bayesian Networks*, which are simplified Bayesian networks that only makes a distinction between zero and non-zero of probabilities.

- Parameter learning is not supported.
- Solutions are found by SAT solver.



N3	N1	N2	P(N3 N1,N2)
False	False	False	0.2
False	True	False	0.3
False	False	True	0
False	True	True	0.9
True	False	False	0.8
...	...	...	...

True CPT

N3	N1	N2	P(N3 N1,N2)
False	False	False	non-zero
False	True	False	non-zero
False	False	True	zero
False	True	True	non-zero
True	False	False	non-zero
...	...	...	...

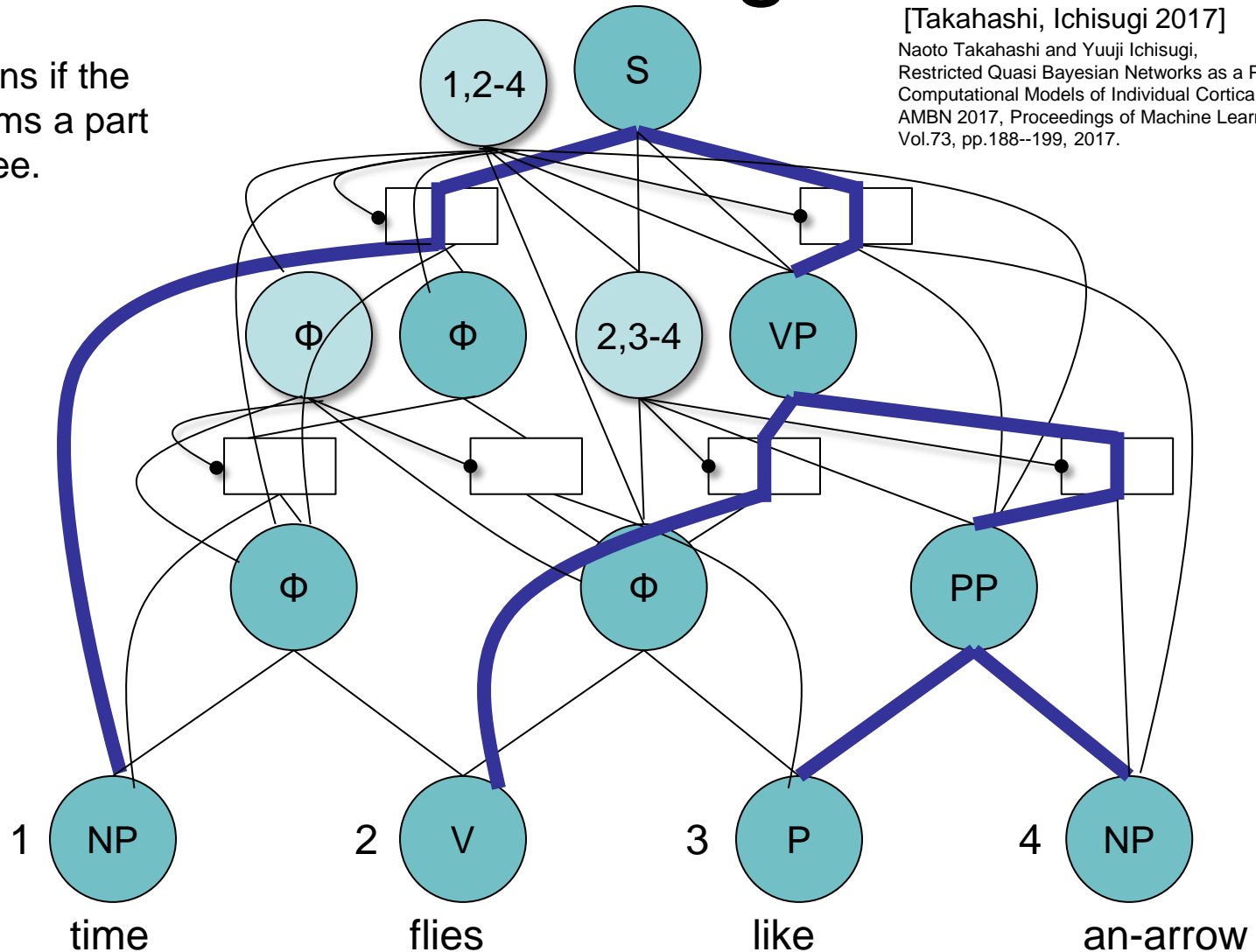
Simplified CPT of  
quasi Bayesian network

# Chart parser for context free grammar

[Takahashi, Ichisugi 2017]

Naoto Takahashi and Yuuji Ichisugi,  
Restricted Quasi Bayesian Networks as a Prototyping Tool for  
Computational Models of Individual Cortical Areas, In Proc. of  
AMBN 2017, Proceedings of Machine Learning Research,  
Vol.73, pp.188--199, 2017.

Each gate opens if the  
connection forms a part  
of the parse tree.

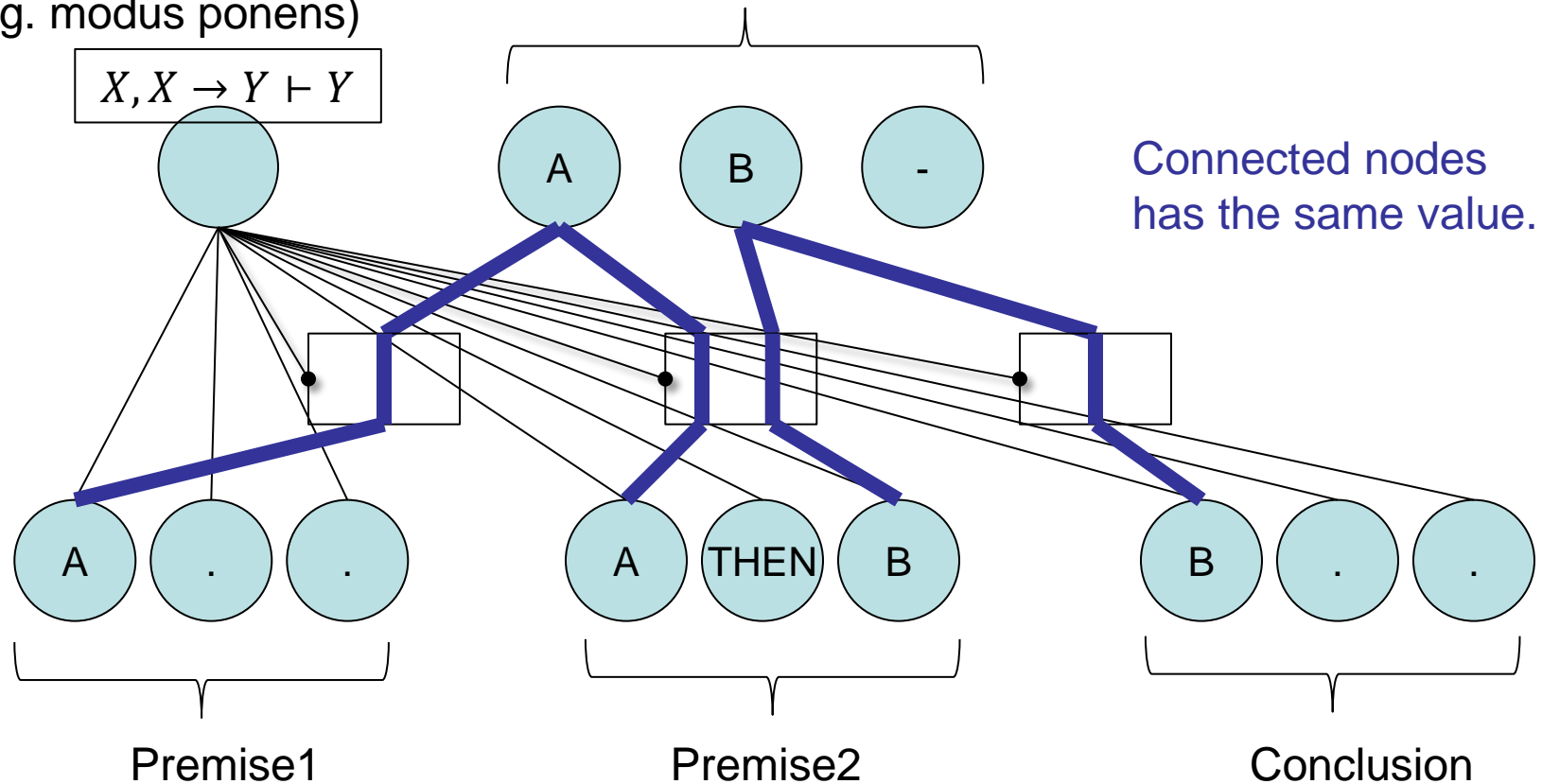


# Prototype of logical inference mechanism

Inference rule  
(e.g. modus ponens)

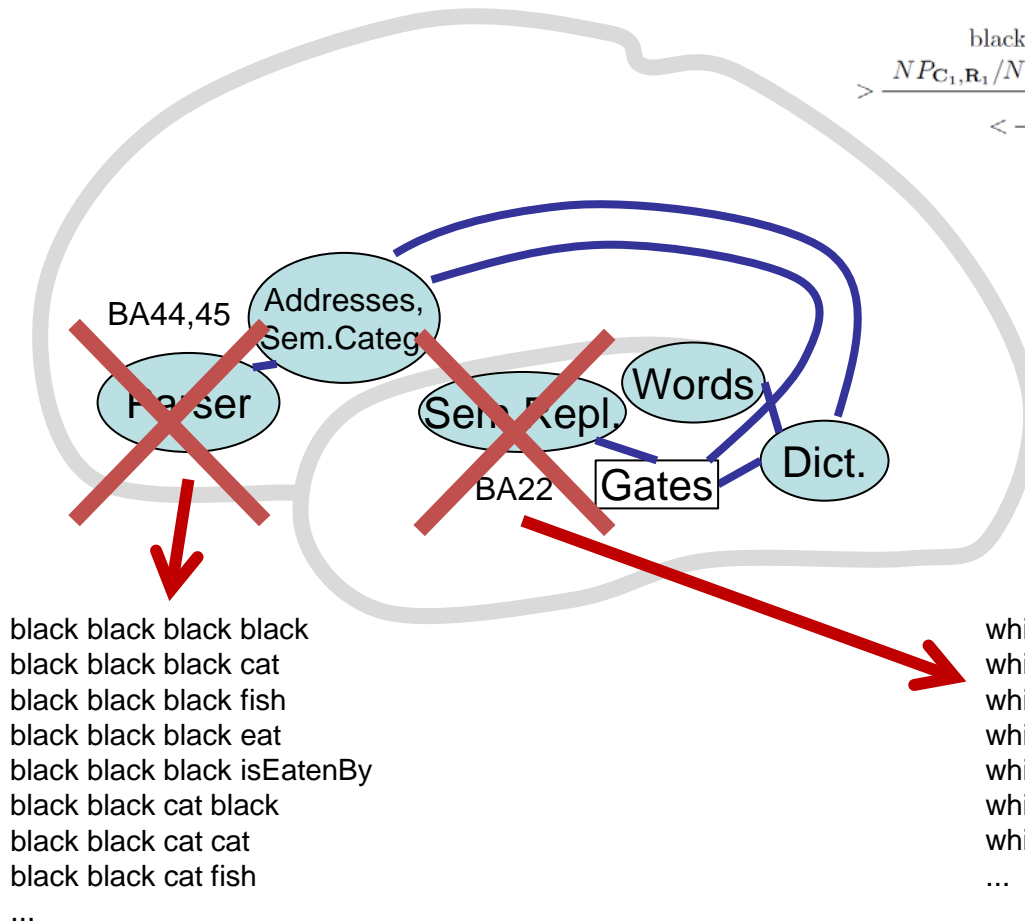
$X, X \rightarrow Y \vdash Y$

Values of unified variables



The network can infer not only a conclusion from the premises, but also the premises from a conclusion. The network will be able to learn inference rules from sample data of premises and conclusions. This mechanism will become key technique to implement unification grammar parser such as CCG.

# A formal model of language areas



$$\begin{array}{c}
 \text{black} \quad \text{cat} \quad \text{eat} \quad \text{fish} \\
 > \frac{NP_{C_1, R_1} / NP_{C_1, R_1} \quad NP_{C_2, R_2}}{NP_{C_1, R_1}} > \frac{(S_{C_3} \setminus NP_{C_3, agent}) / NP_{C_3, patient} \quad NP_{C_4, R_4}}{S_{C_3} \setminus NP_{C_3, agent}} \\
 < \frac{NP_{C_1, R_1}}{S_{C_3}}
 \end{array}$$

- Modified version of Combinatory Categorical Grammar [Steedman 2000].
- This model may be implemented by BESOM Ver.4.
- Destruction of parts of this model reproduces behavior of aphasia.

Sentences are syntactically **incorrect**.  
Selected words are semantically **correct**.

Resembles Broca's aphasia

Sentences are syntactically **correct**.  
Selected words are semantically **incorrect**.

Resembles Werniche's aphasia

# Conclusion

- A BESOM can be used as a *bidirectional* Deep Neural Network.
  - Thanks to the restricted CPT model, recognition and learning algorithms are scalable.
- Using Gate nodes, parser and logical inference mechanism would be implemented. (ongoing project)
- Future work
  - Model of PFC and ACC
  - Sequence learning, short-term memory
  - Large scale implementation