深層学習

渡辺太郎

taro.watanabe at nict.go.jp

https://sites.google.com/site/alaginmt2015/

kbest翻訳候補

機械翻訳について勉強したい。

	$\log Pr(\boldsymbol{\phi}\|\boldsymbol{e})$	$\log Pr(\boldsymbol{e})$	$\log Pr(\boldsymbol{f},\boldsymbol{\alpha}\|\boldsymbol{\phi})$	
I want to study about machine translation	-2	-3	-4	-9
I need to master machine translation	-3	-4	-4	-11
machine translation want to study	-2	-5	-1	-8
I don't want to learn anything	-5	-2	-3	-10

0.5×-2	0.4×-3	0.2×-4	-3.0
0.5×-3	0.4×-4	0.2×-4	-3.9
0.5×-2	0.4×-5	0.2×-1	-3.2
0.5×-5	0.4×-2	0.2×-3	-3.9

重み付けにより並び替え

重み付け

$$\hat{\boldsymbol{e}} = \arg\max_{\boldsymbol{e}} \overset{0.2}{Pr(\boldsymbol{f}, \boldsymbol{\alpha} | \boldsymbol{\phi}, \boldsymbol{e})} \; \overset{0.5}{Pr(\boldsymbol{\phi} | \boldsymbol{e})} \; \overset{0.4}{Pr(\boldsymbol{e})}$$

- より一般化:

$$\begin{aligned} \hat{\boldsymbol{e}} &= \arg\max_{\boldsymbol{e}} \frac{\sum_{\boldsymbol{d}} \exp\left(\boldsymbol{w}^{\top} \boldsymbol{h}(\boldsymbol{f}, \boldsymbol{d}, \boldsymbol{e})\right)}{\sum_{\boldsymbol{e}', \boldsymbol{d}'} \exp\left(\boldsymbol{w}^{\top} \boldsymbol{h}(\boldsymbol{f}, \boldsymbol{d}', \boldsymbol{e}')\right)} \\ &\approx \arg\max_{\langle \boldsymbol{e}, \boldsymbol{d} \rangle} \boldsymbol{w}^{\top} \boldsymbol{h}(\boldsymbol{f}, \boldsymbol{d}, \boldsymbol{e}) \end{aligned}$$

(Och and Ney, 2002)

最適化 = 最適なwを識別学習

なぜ深層学習?

日本	|||	a japanese	|||	0.272727	0.283379	0.00953895	0.00411563
日本	|||	all over japan	|||	0.222222	0.541131	0.00317965	3.72914e-07
日本	|||	around japan	|||	0.05	0.541131	0.00158983	0.000337958
日本	|||	as some japanese	|||	1	0.283379	0.00158983	5.45274e-07
日本	|||	japan ,	|||	0.0897436	0.541131	0.0111288	0.0225871
日本	|||	japan :	|||	1	0.541131	0.00158983	0.00131649
日本	|||	japan	|||	0.396648	0.541131	0.338633	0.463146
日本	|||	japanese ,	|||	0.0769231	0.283379	0.00158983	0.00557971
日本	|||	japanese	|||	0.242553	0.283379	0.09062	0.114411

素性を試行錯誤で開発

⇩

素性を自動的に学習

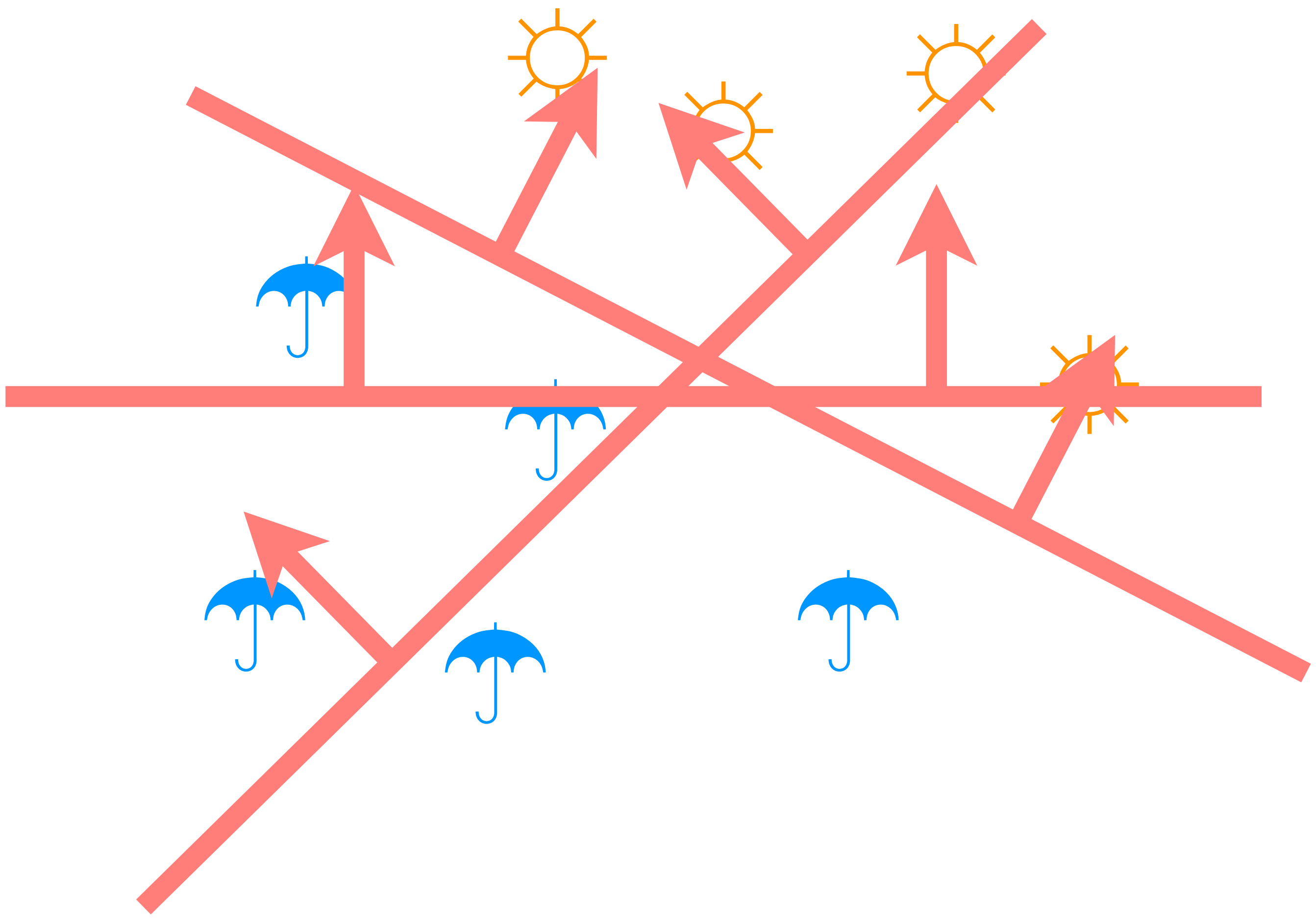
機械翻訳への貢献

NIST MT12 Test		
	Ar-En	**Ch-En**
	BLEU	BLEU
OpenMT12 - 1st Place	49.5	32.6
OpenMT12 - 2nd Place	47.5	32.2
OpenMT12 - 3rd Place	47.4	30.8
...	...	...
OpenMT12 - 9th Place	44.0	27.0
OpenMT12 - 10th Place	41.2	25.7
Baseline (w/o RNNLM)	48.9	33.0
Baseline (w/ RNNLM)	49.8	33.4
+ S2T/L2R NNJM (Dec)	51.2	34.2
+ S2T NNLTM (Dec)	52.0	34.2
+ T2S NNLTM (Resc)	51.9	34.2
+ S2T/R2L NNJM (Resc)	52.2	34.3
+ T2S/L2R NNJM (Resc)	52.3	34.5
+ T2S/R2L NNJM (Resc)	52.8	34.7
"Simple Hier." Baseline	43.4	30.1
+ S2T/L2R NNJM (Dec)	47.2	31.5
+ S2T NNLTM (Dec)	48.5	31.8
+ Other NNJMs (Resc)	49.7	32.2

(Devlin et al., 2014)

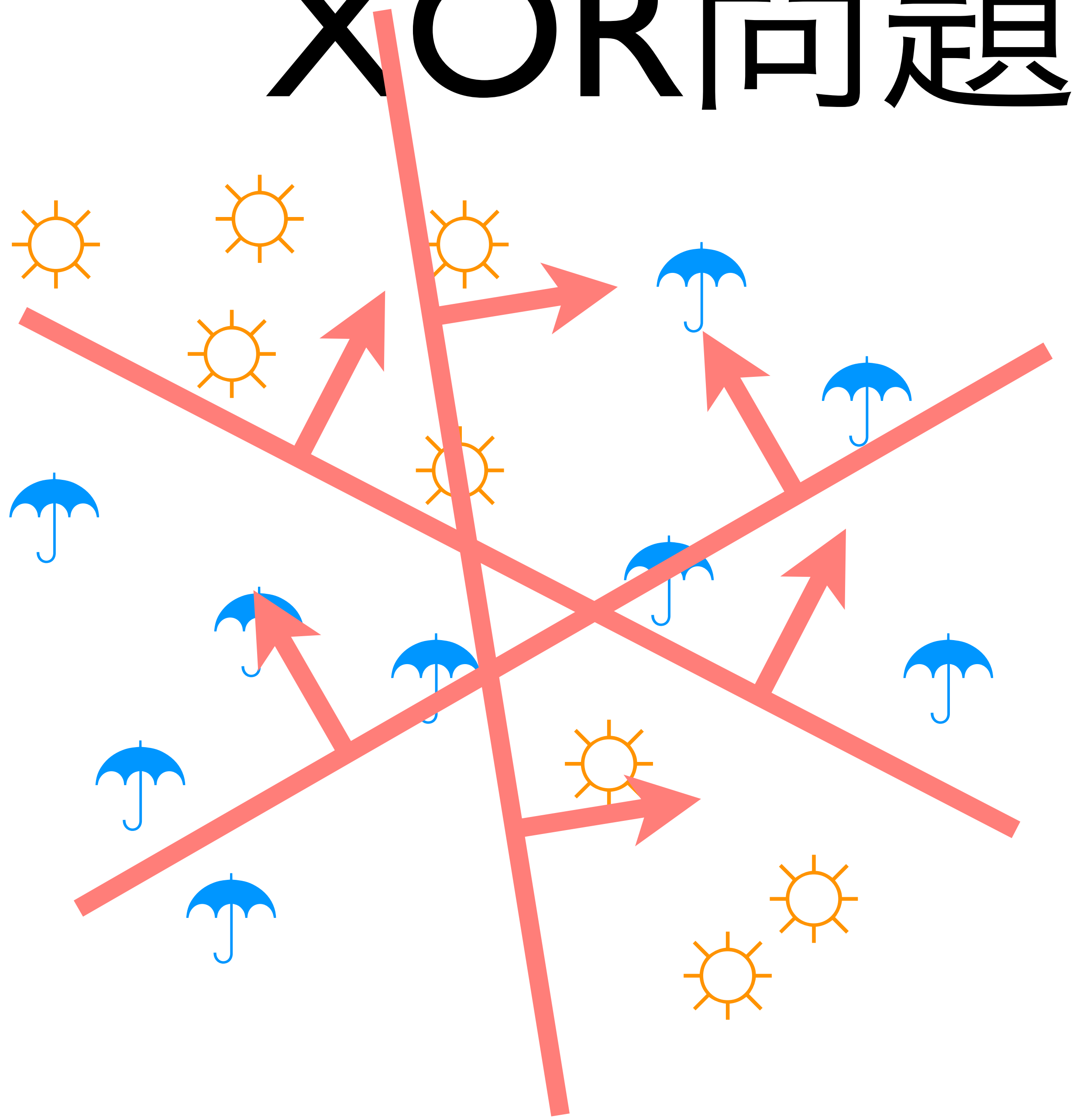
背景

線形分類



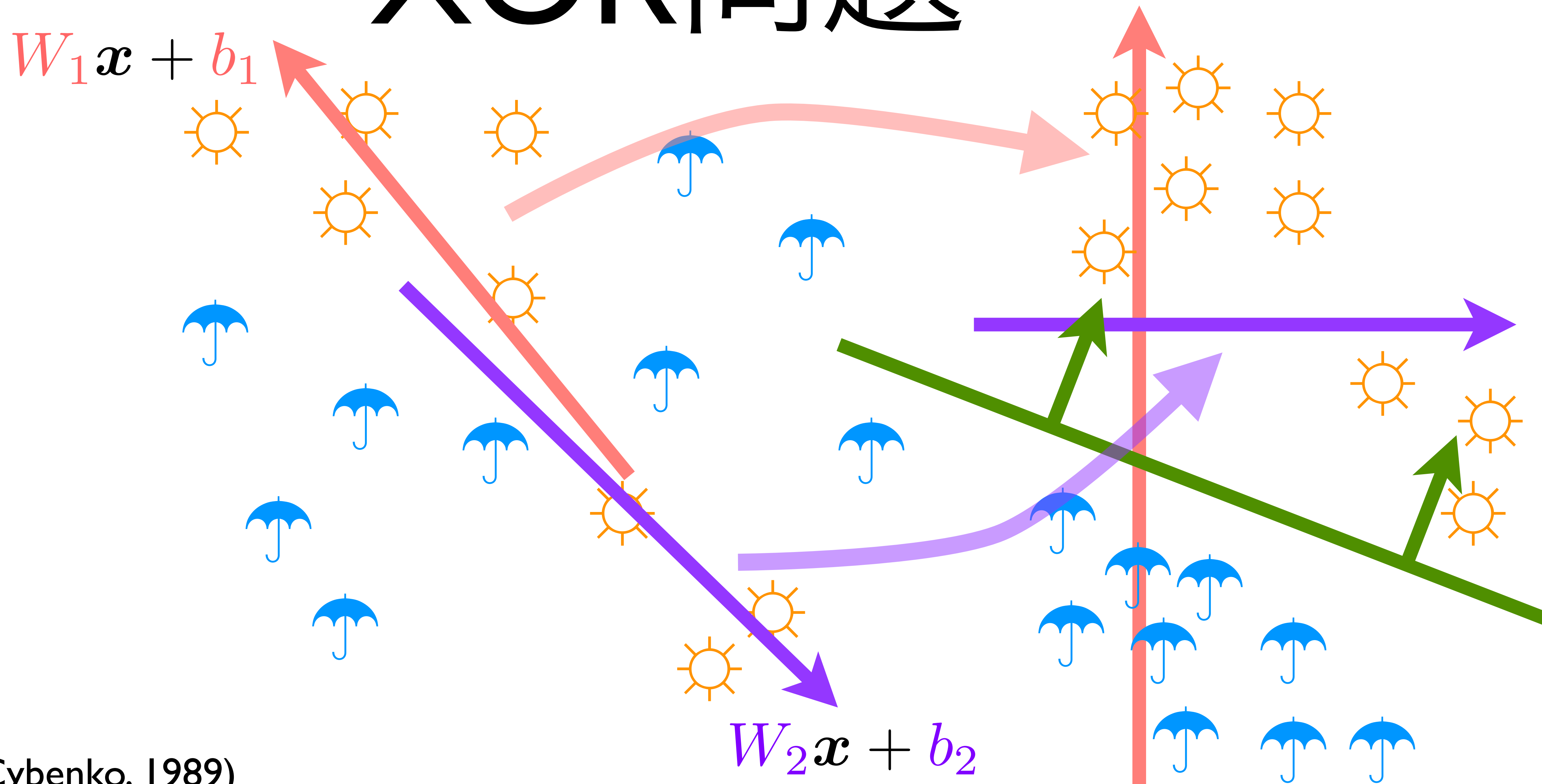
- 線形モデルで識別可能: $y = \mathrm{sign}(W\boldsymbol{x} + b)$

XOR問題



- 線形モデルで解けない

XOR問題



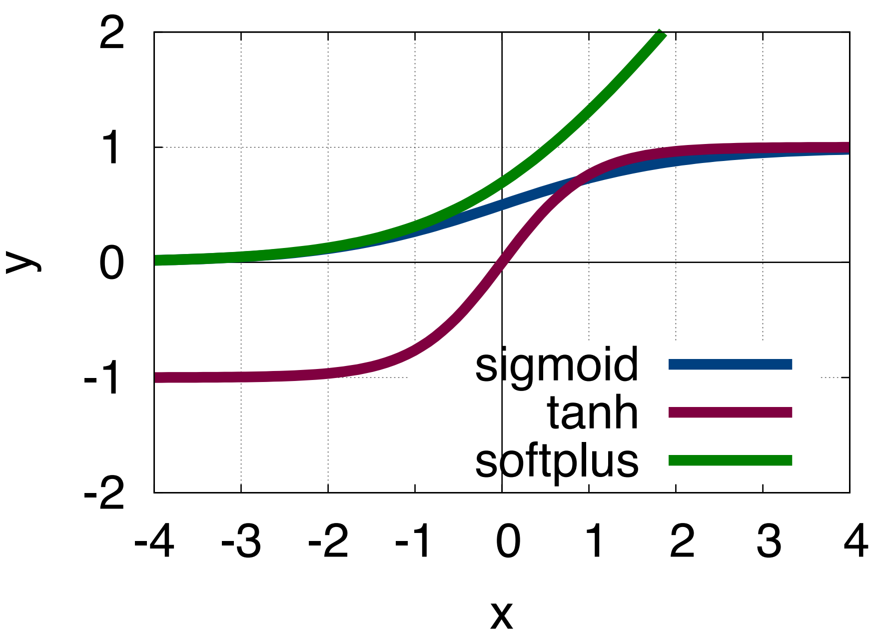
(Cybenko, 1989)

- 複数の座標変換 + 線形分類

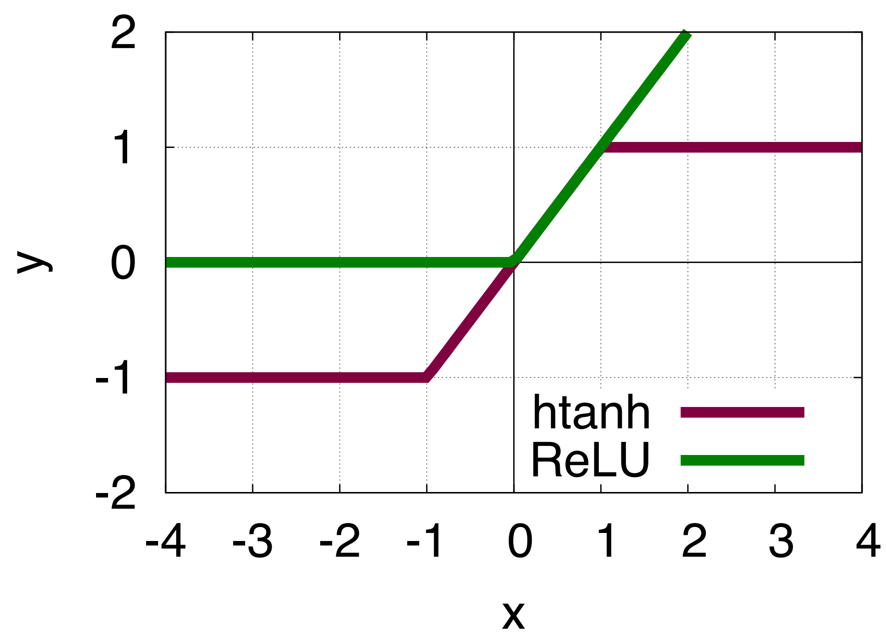
活性化関数

$$y = \mathrm{sign}\left(W_3 \begin{bmatrix} f(W_2\boldsymbol{x} + b_2) \\ f(W_1\boldsymbol{x} + b_1) \end{bmatrix} + b_3 \right)$$

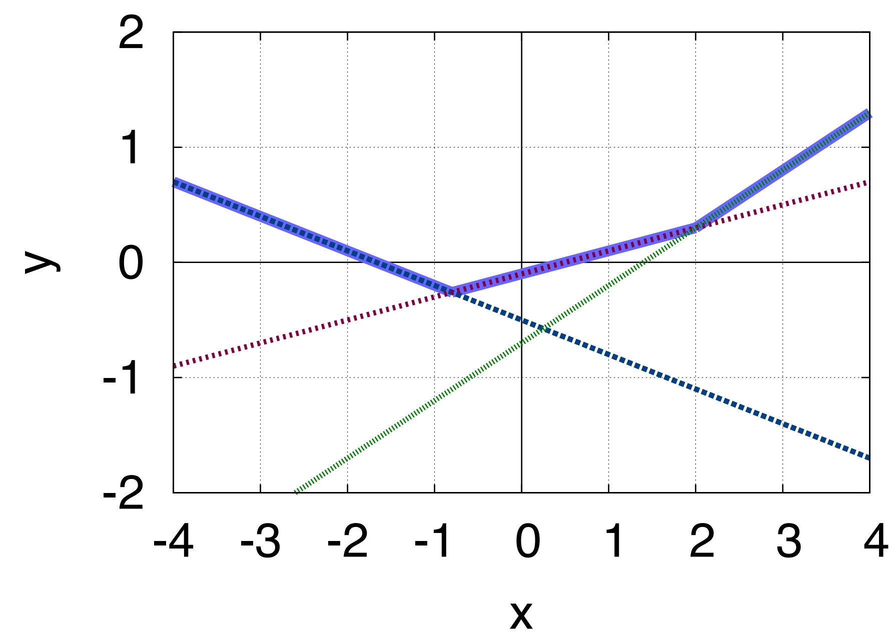
活性化関数



sigmoid/tanh/
softplus



htanh/ReLU

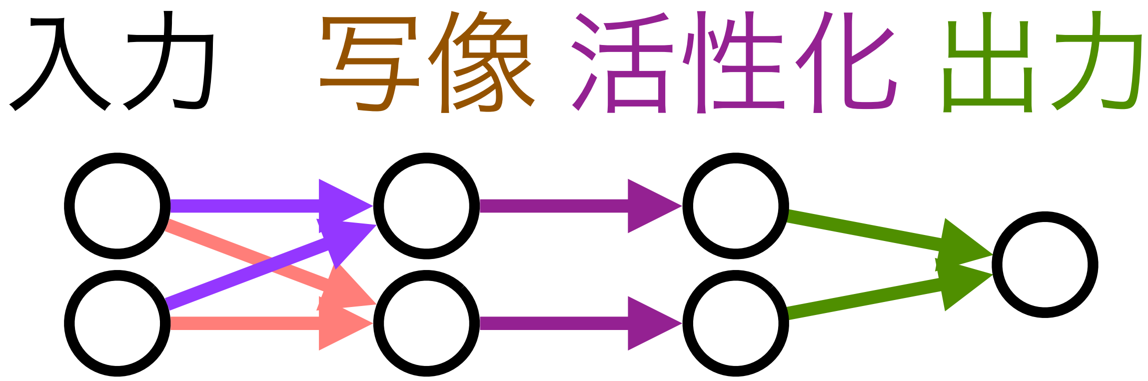


maxout

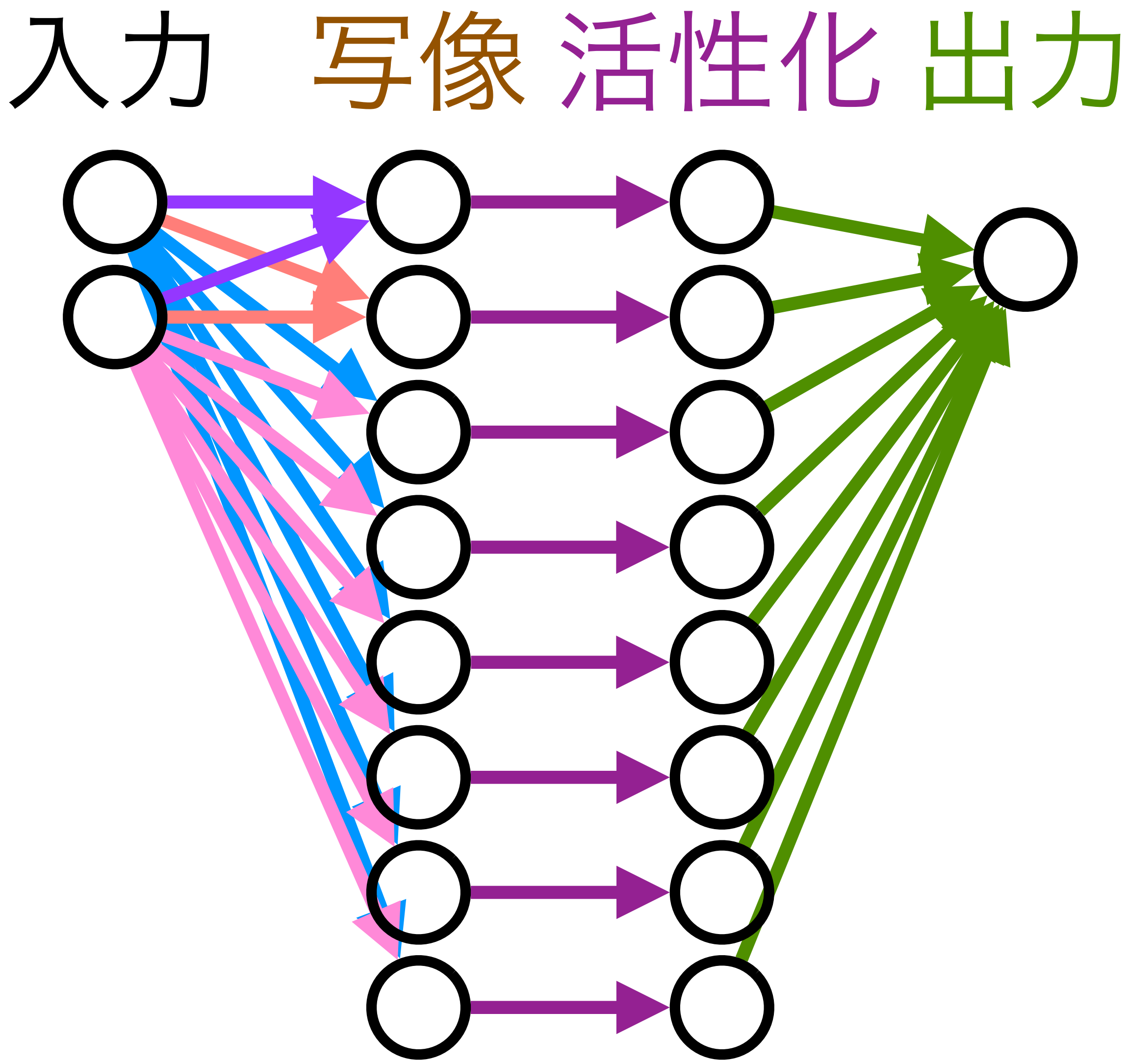
(Goodfellow et al., 2013)

- 非線形な変換による柔軟な設計

ニューラルネットワーク

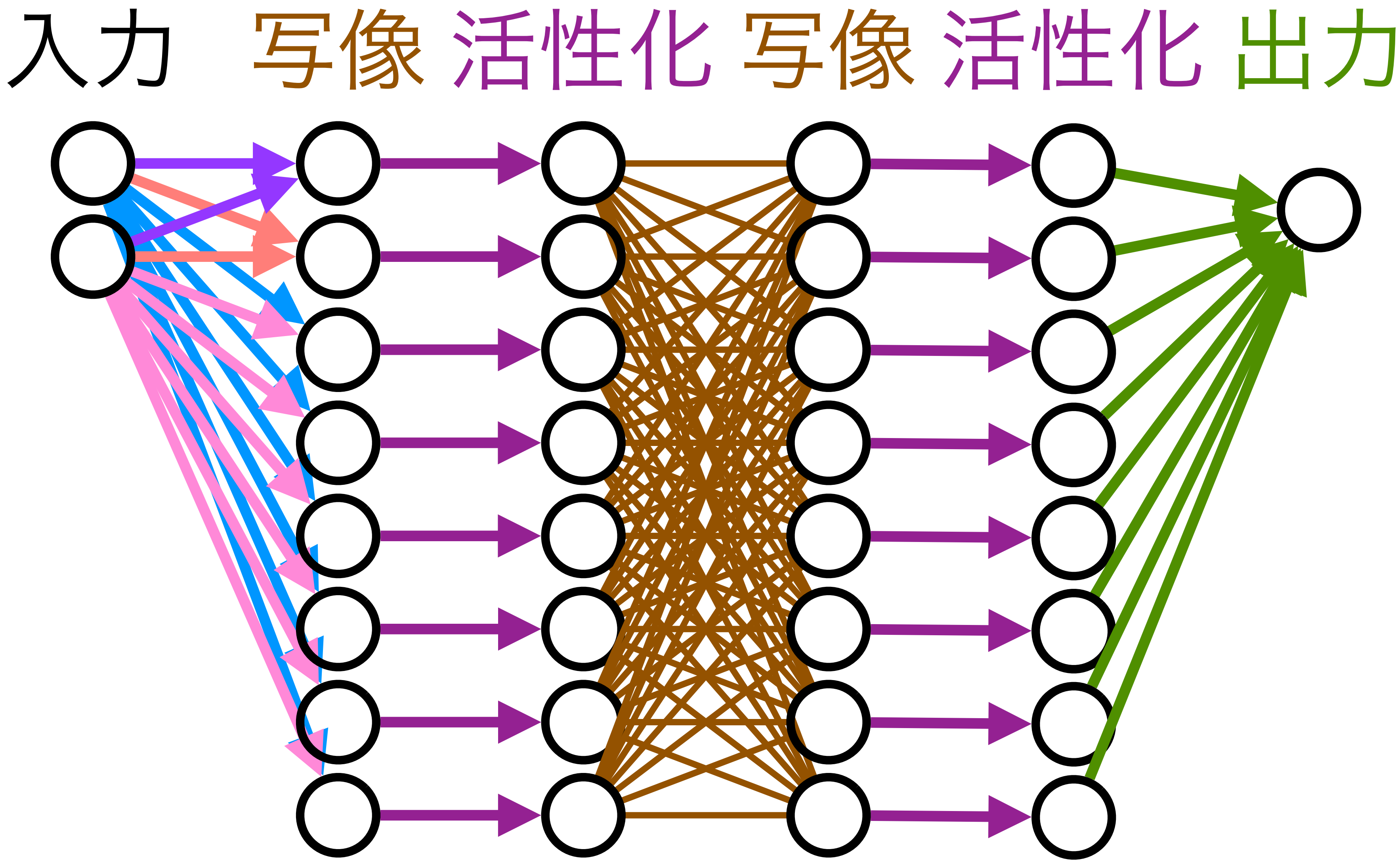


ニューラルネットワーク



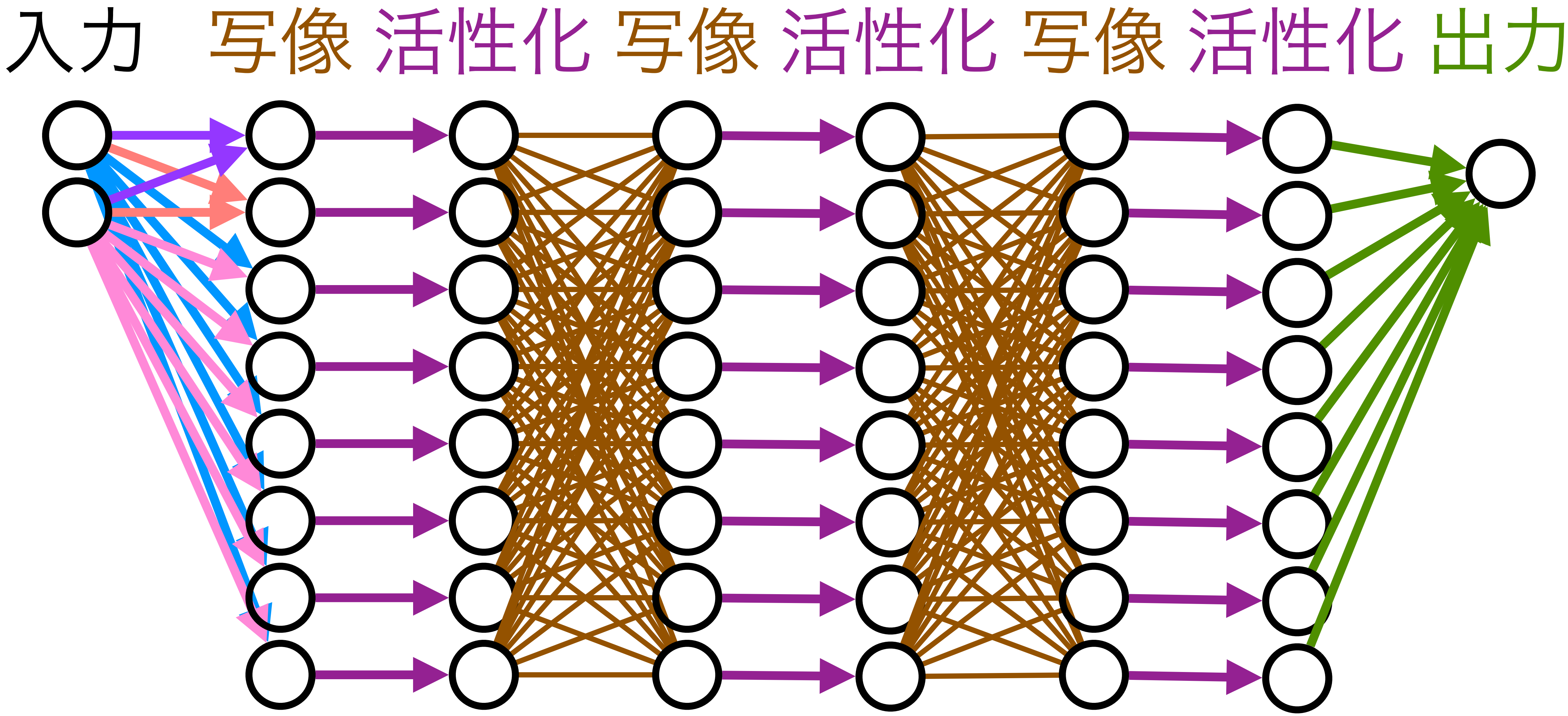
多次元化

ニューラルネットワーク



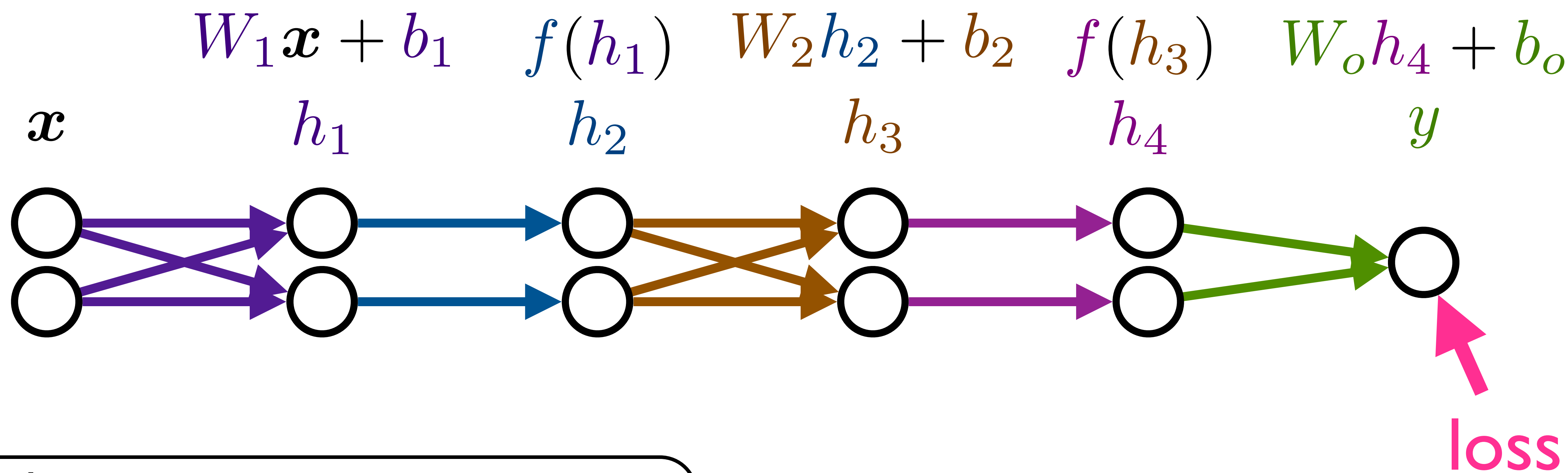
多次元化+多層化

ニューラルネットワーク



多次元化+多層化+多層化

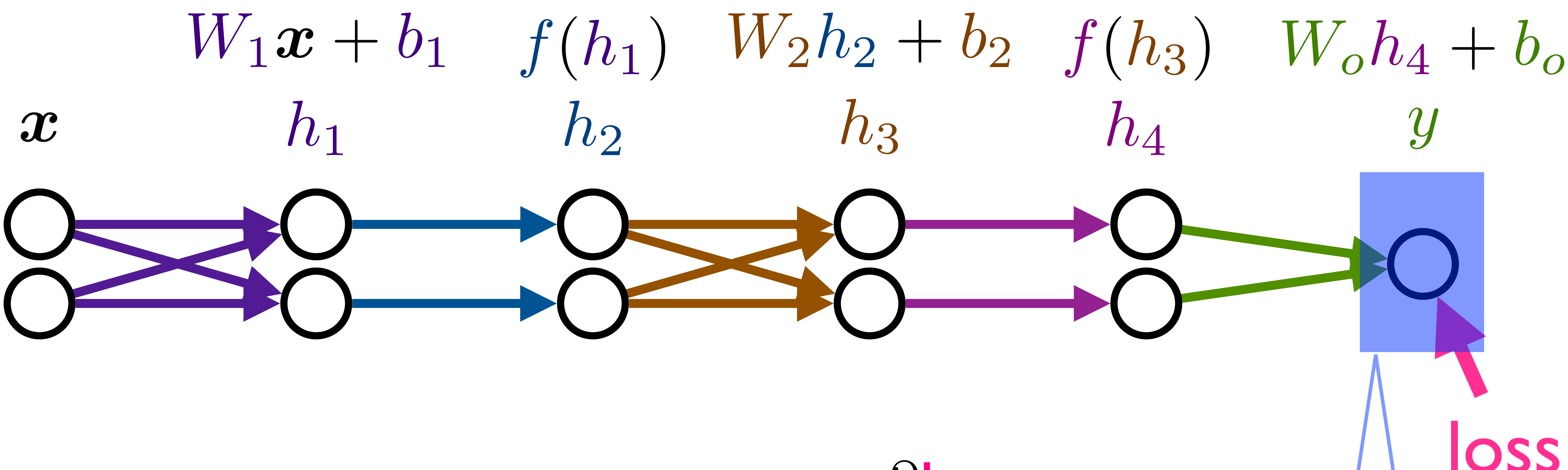
学習



晴れなら1、曇なら-1

- 学習データ(x, label)のxからネットワークを計算
- 損失を計算: 例、hinge損失: max(0, 1 - label * y)

最適化



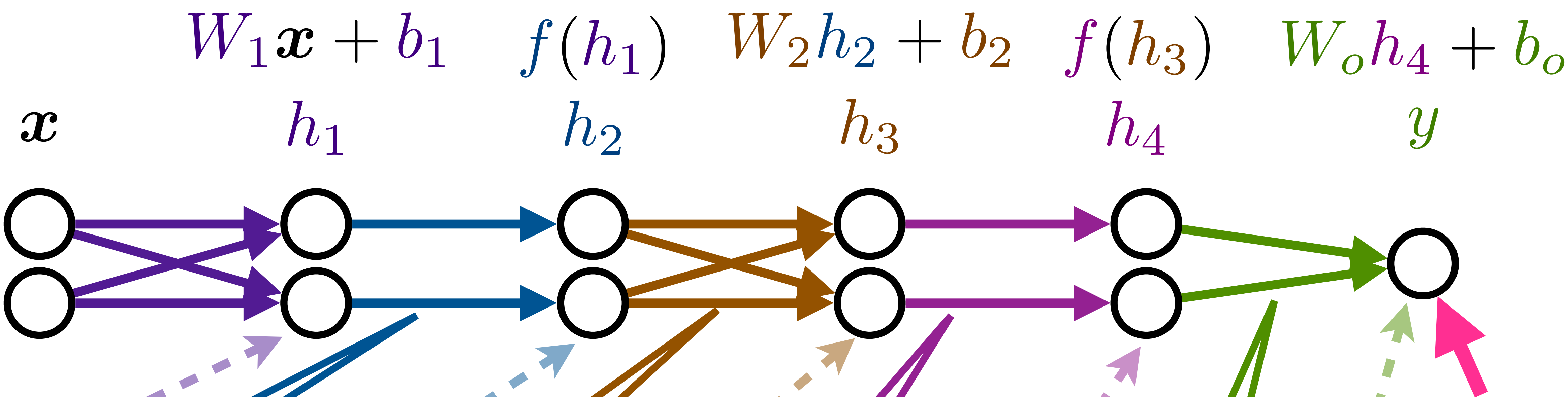
$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \frac{\partial \text{loss}}{\partial \theta}$$

わかっているのは、ここだけ。

- 例: t番目の学習データが与えられた時、

  SGDで更新: $\Theta = \{W_1, b_1, W_2, b_2, W_o, b_o\}$

伝搬



$$\delta_1 = \boxed{\frac{\partial h_2}{\partial h_1}}\delta_2 = f'(h_1)\delta_2$$

$$\delta_2 = \boxed{\frac{\partial h_3}{\partial h_2}}\delta_3 = W_2^\top\delta_3$$

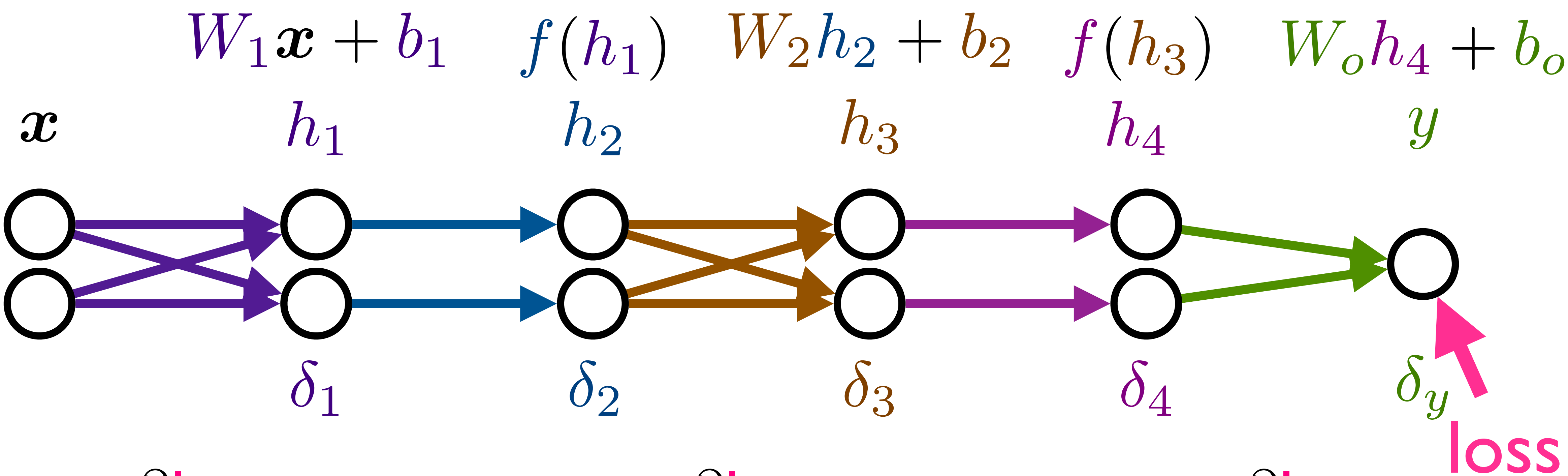
$$\delta_3 = \boxed{\frac{\partial h_4}{\partial h_3}}\delta_4 = f'(h_3)\delta_4$$

$$\delta_4 = \boxed{\frac{\partial y}{\partial h_4}}\delta_y = W_o^\top\delta_y$$

$$\delta_y = \frac{\partial \text{loss}}{\partial y} = -\text{label}$$

- 損失は一番上の層で計算: 下層へ伝搬

勾配

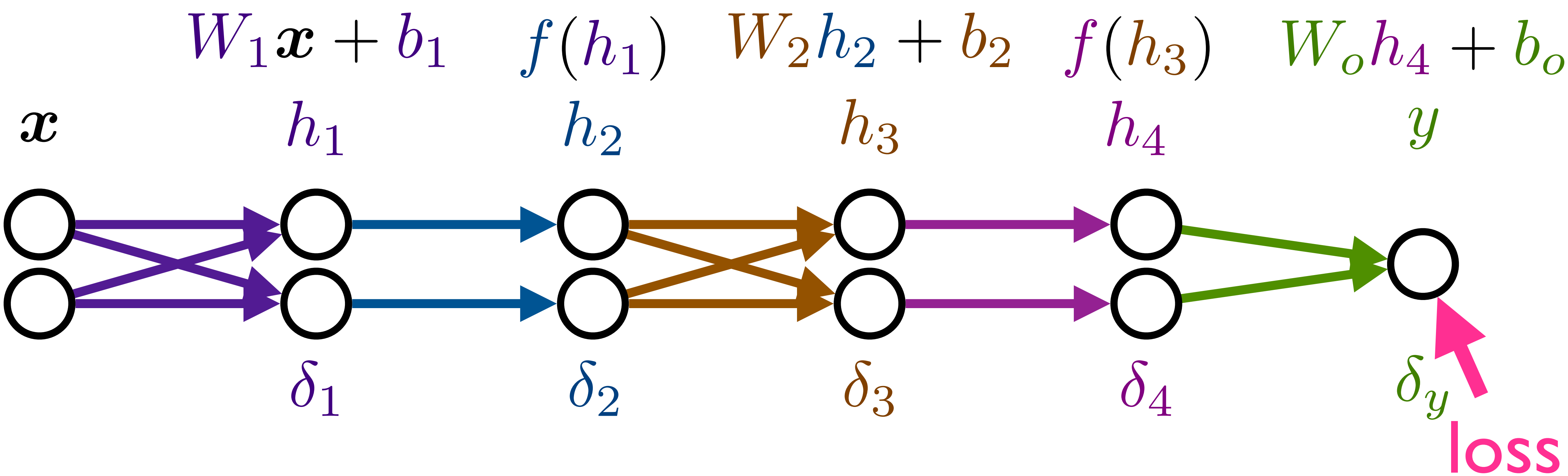


$$\frac{\partial \text{loss}}{\partial W_1}=\delta_1\boldsymbol{x}^\top \qquad \frac{\partial \text{loss}}{\partial W_2}=\delta_3 h_2^\top \qquad \frac{\partial \text{loss}}{\partial W_o}=\delta_y h_4^\top$$

$$\frac{\partial \text{loss}}{\partial b_1}=\delta_1 \qquad \frac{\partial \text{loss}}{\partial b_2}=\delta_3 \qquad \frac{\partial \text{loss}}{\partial b_o}=\delta_y$$

- 各層のδから各層毎にパラメータの勾配を計算

更新



$$W_2^{(t+1)} \leftarrow W_2^{(t)} - \frac{\partial \text{loss}}{\partial W_2} \quad W_2^{(t+1)} \leftarrow W_2^{(t)} - \frac{\partial \text{loss}}{\partial W_2} \quad W_o^{(t+1)} \leftarrow W_o^{(t)} - \frac{\partial \text{loss}}{\partial W_o}$$

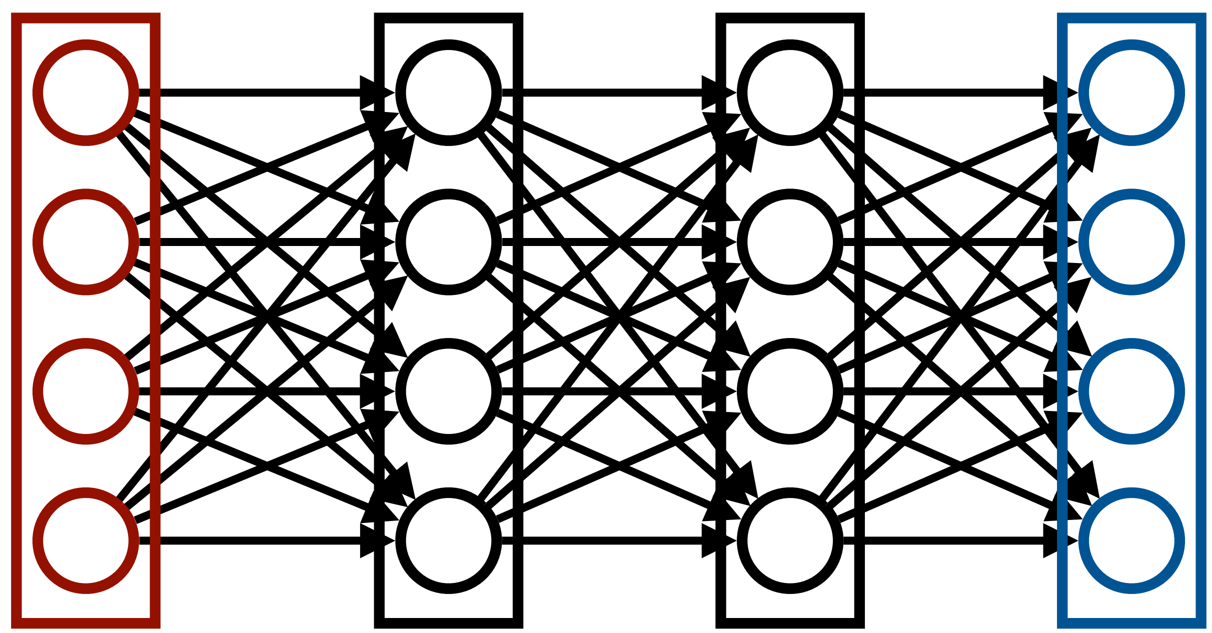
$$b_2^{(t+1)} \leftarrow b_2^{(t)} - \frac{\partial \text{loss}}{\partial b_2} \quad b_2^{(t+1)} \leftarrow b_2^{(t)} - \frac{\partial \text{loss}}{\partial b_2} \quad b_o^{(t+1)} \leftarrow b_o^{(t)} - \frac{\partial \text{loss}}{\partial b_o}$$

- 各パラメータを更新

注

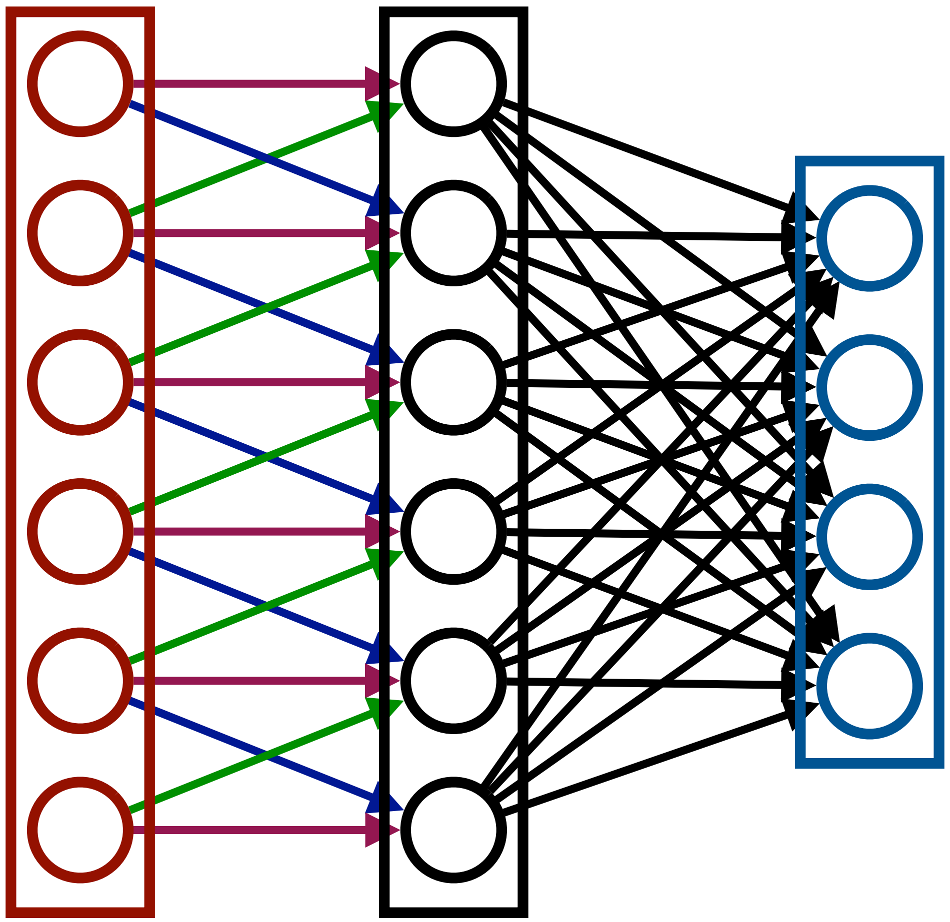
- SGDの代わりに: AdaGrad、AdaDelta、Adam
- 過学習しやすい: DropOut、正則化、事前学習
- 学習に時間が掛かる: オンライン学習、mini-batch、GPU

Feed-Forward NN



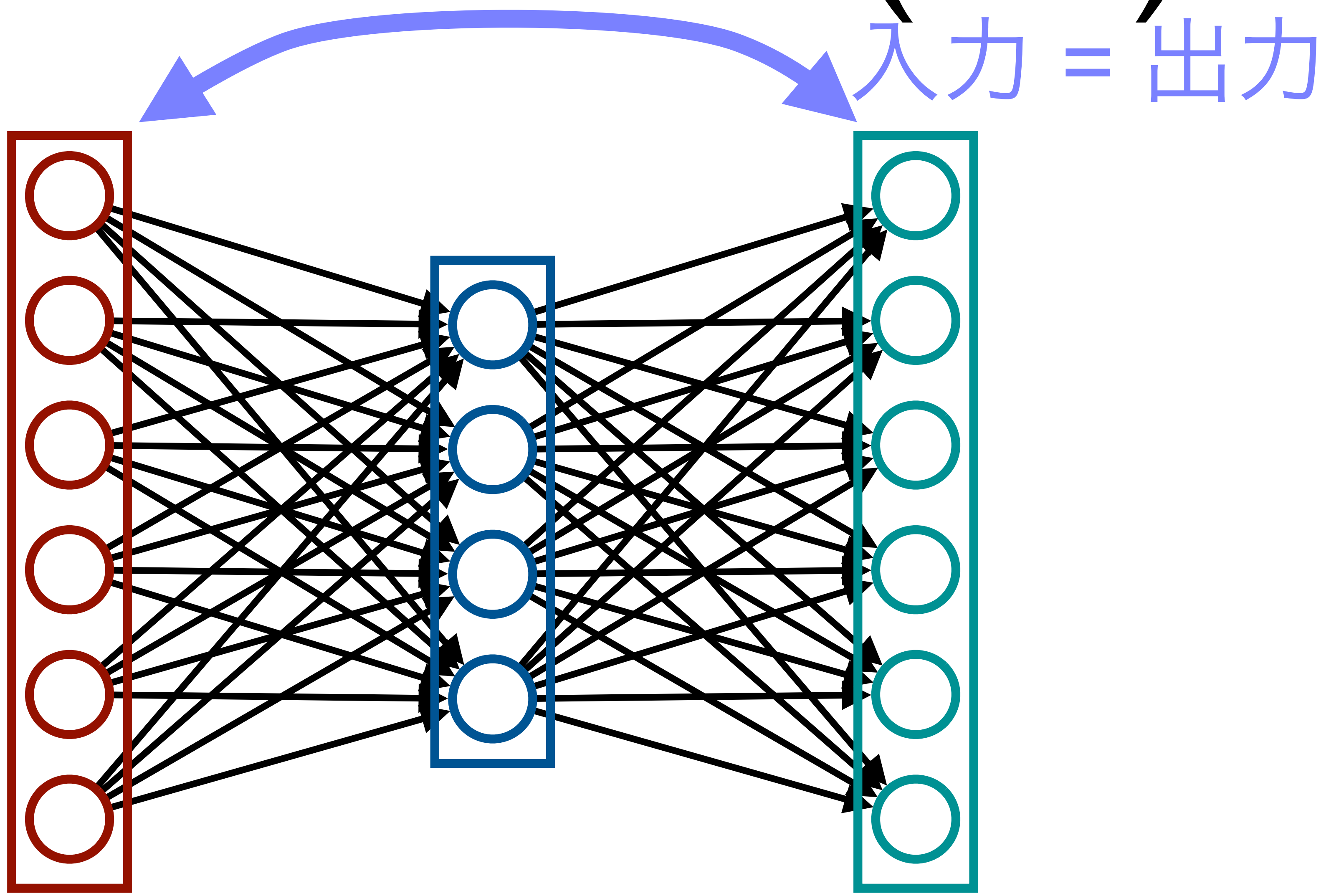
- いわゆる(?)ニューラルネットワーク
- 注意: 簡潔にするため活性化関数を省略

Convolution



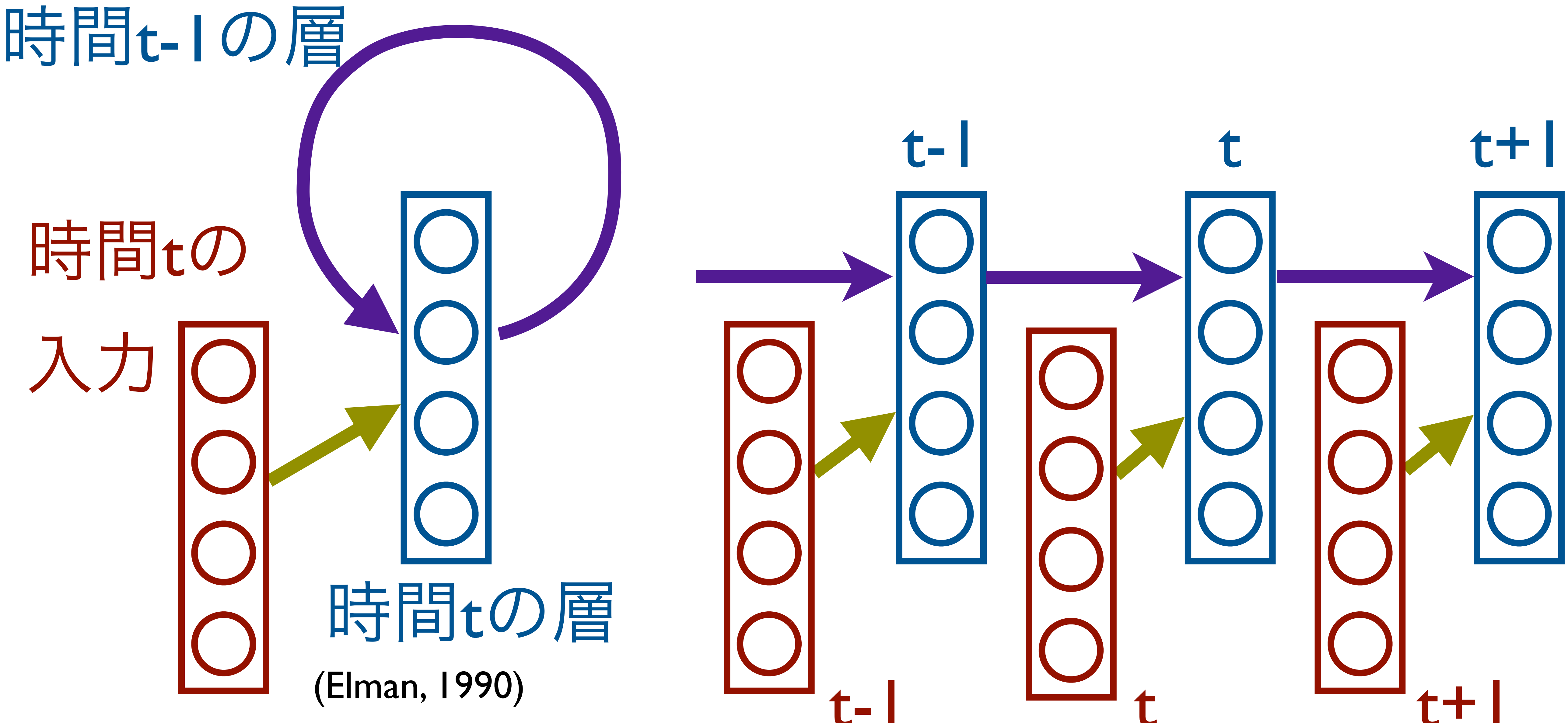
- 局所的に各層を接続(同じ向きでパラメーターを共有)+次元数を減らす(pooling)

Autoencoder (AE)



- 隠れ層から入力を再現するように学習
- 教師なし学習が可能

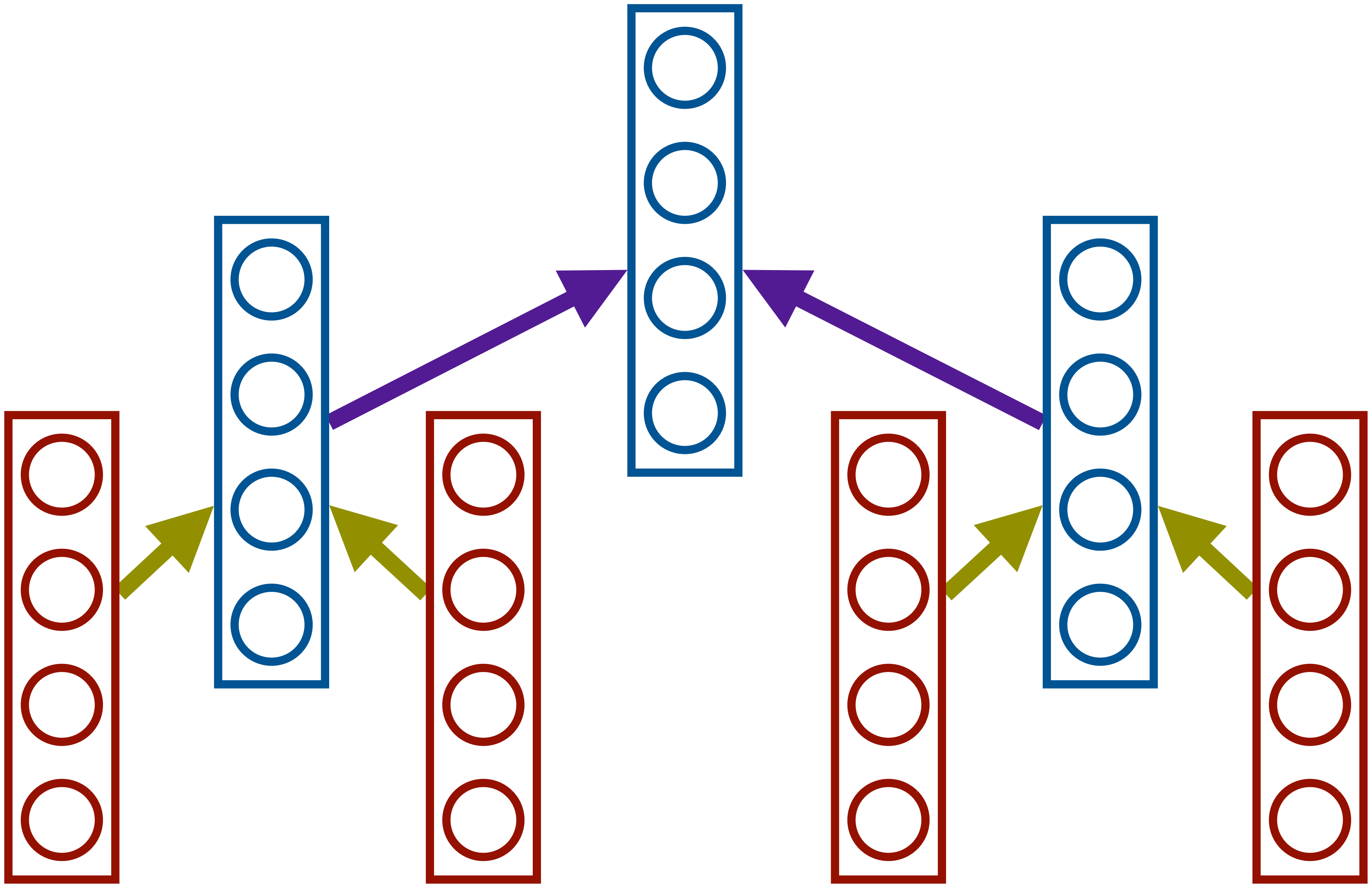
Recurrent NN



(Elman, 1990)

- 系列をモデル化
- 注意: 簡潔にするため細かい接続を無視

Recursive NN



(Pollack, 1990)

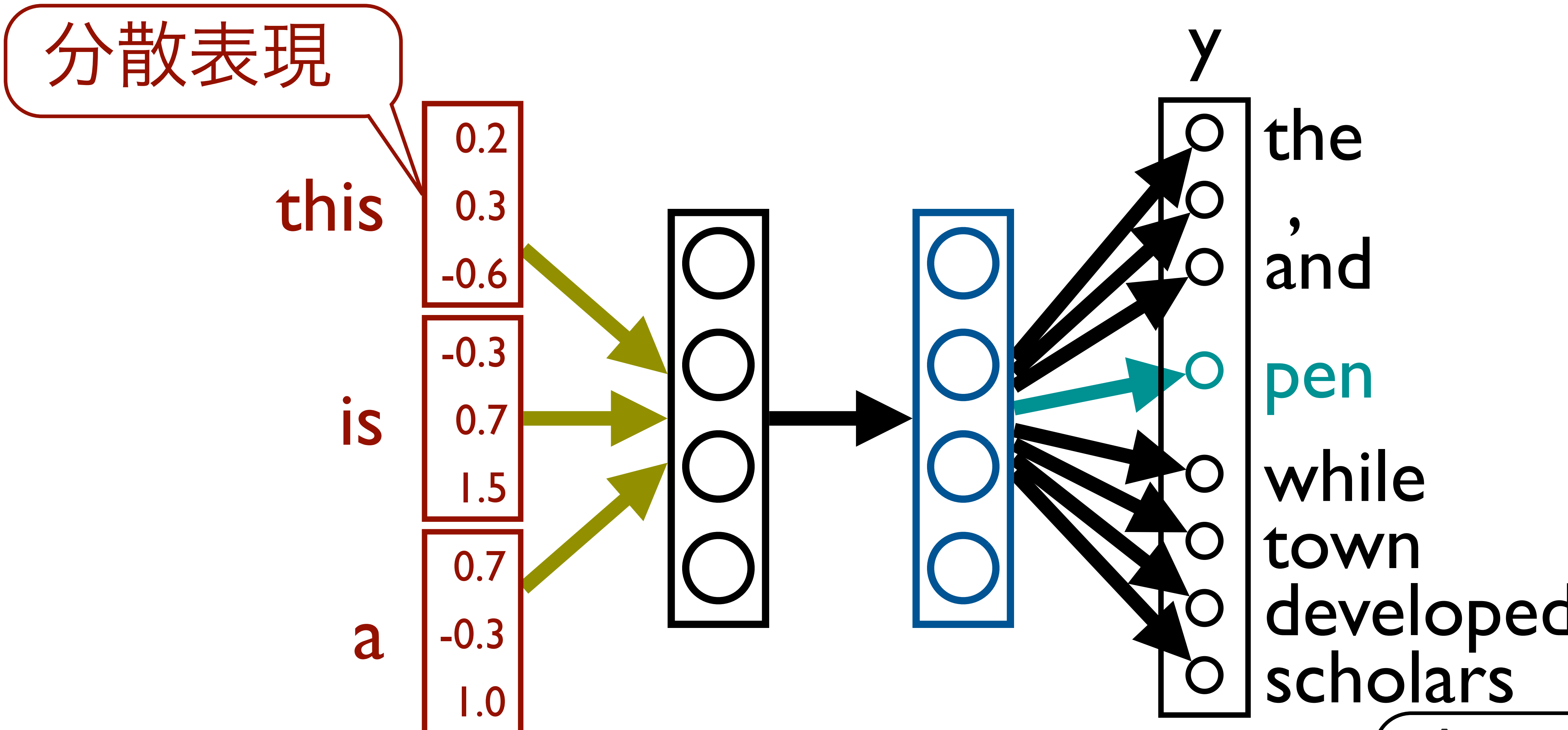
- 任意の構造(例、木構造)をモデル化

機械翻訳への応用

- 言語モデル
- 翻訳モデル
- 単語アライメント

言語モデル

FFNN 言語モデル



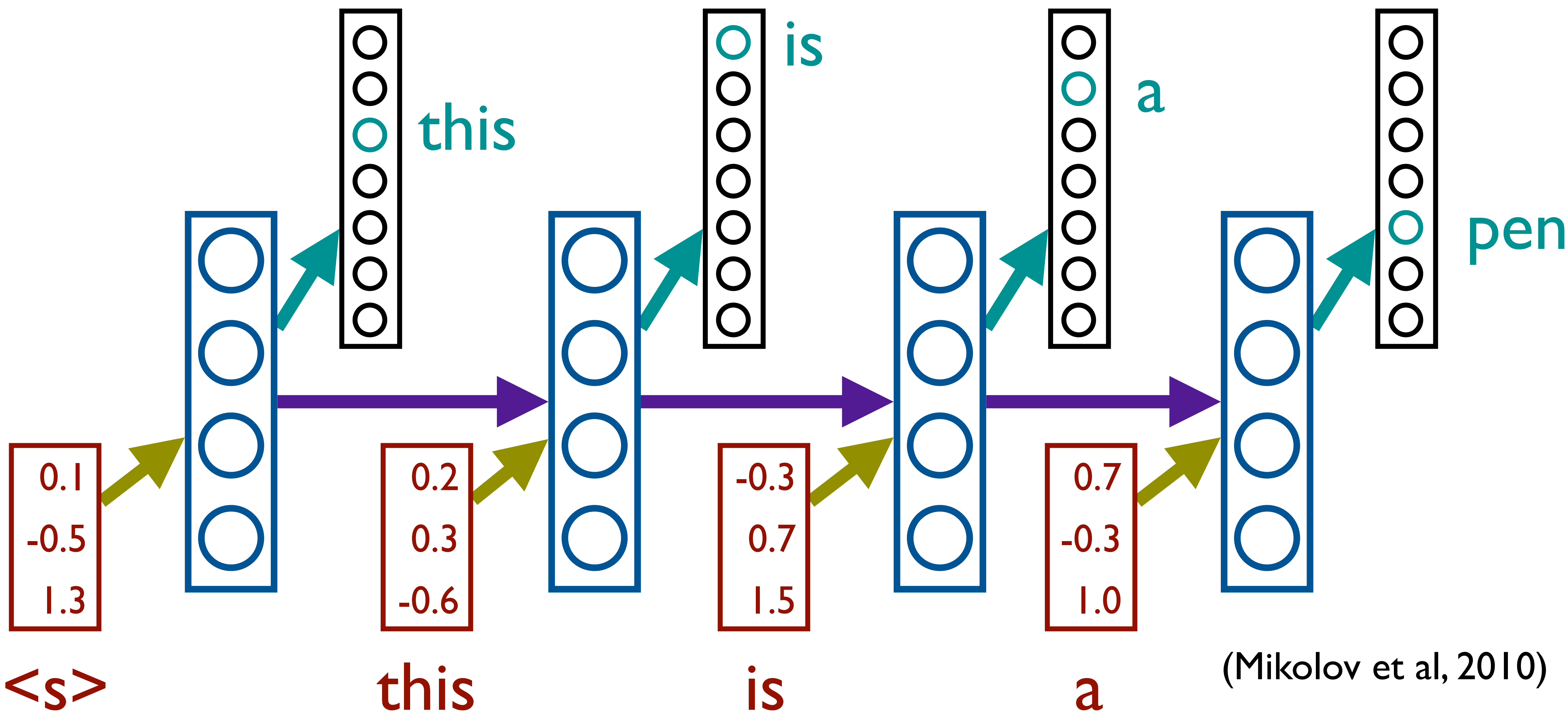
(Schwenk, 2007)

- softmaxによる出力層

$$Pr(\text{pen}|\text{this is a}) = \frac{\exp(y_{\text{pen}})}{\sum_{w \in \mathcal{V}} \exp(y_w)}$$

全ての単語についてΣを計算

Recurrent NN 言語モデル

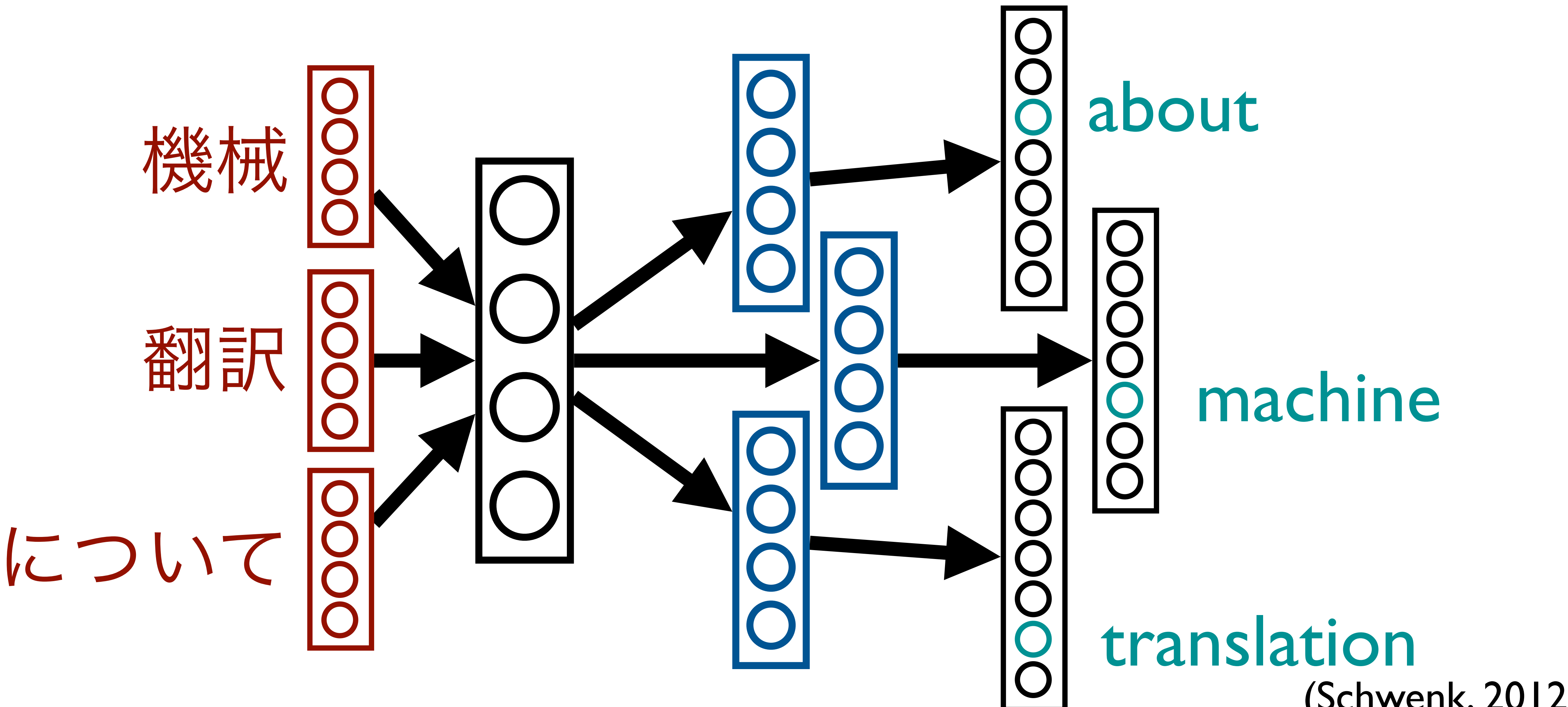


(Mikolov et al, 2010)

- 文全体のコンテキストを隠れ層で表現
- 近似的にデコーダの素性として使用可能

翻訳モデル

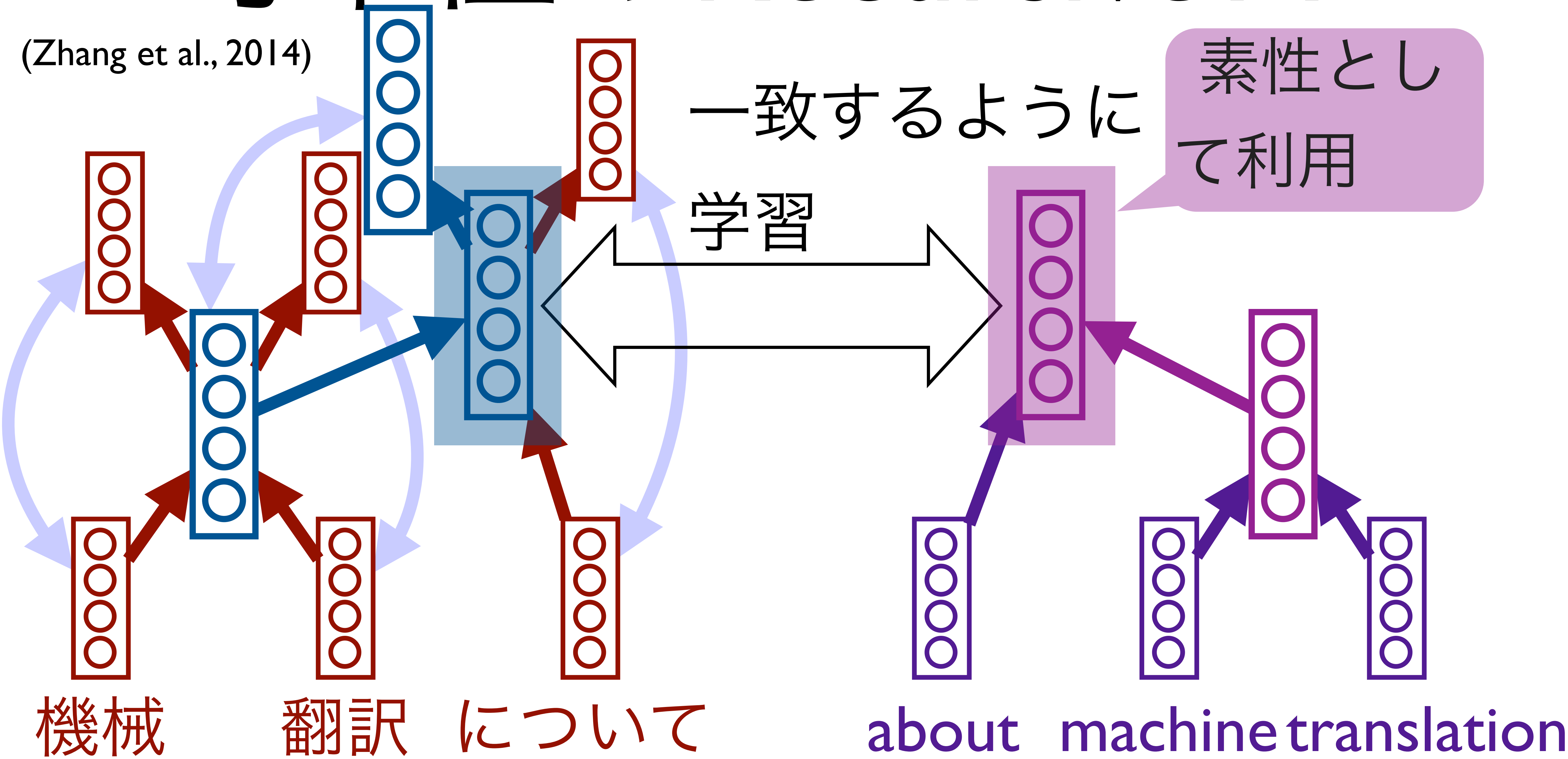
句単位のFFNN



- FFNN言語モデルと同様な学習法
- 問題点: 従来法により句を抽出するため、改善は少ない

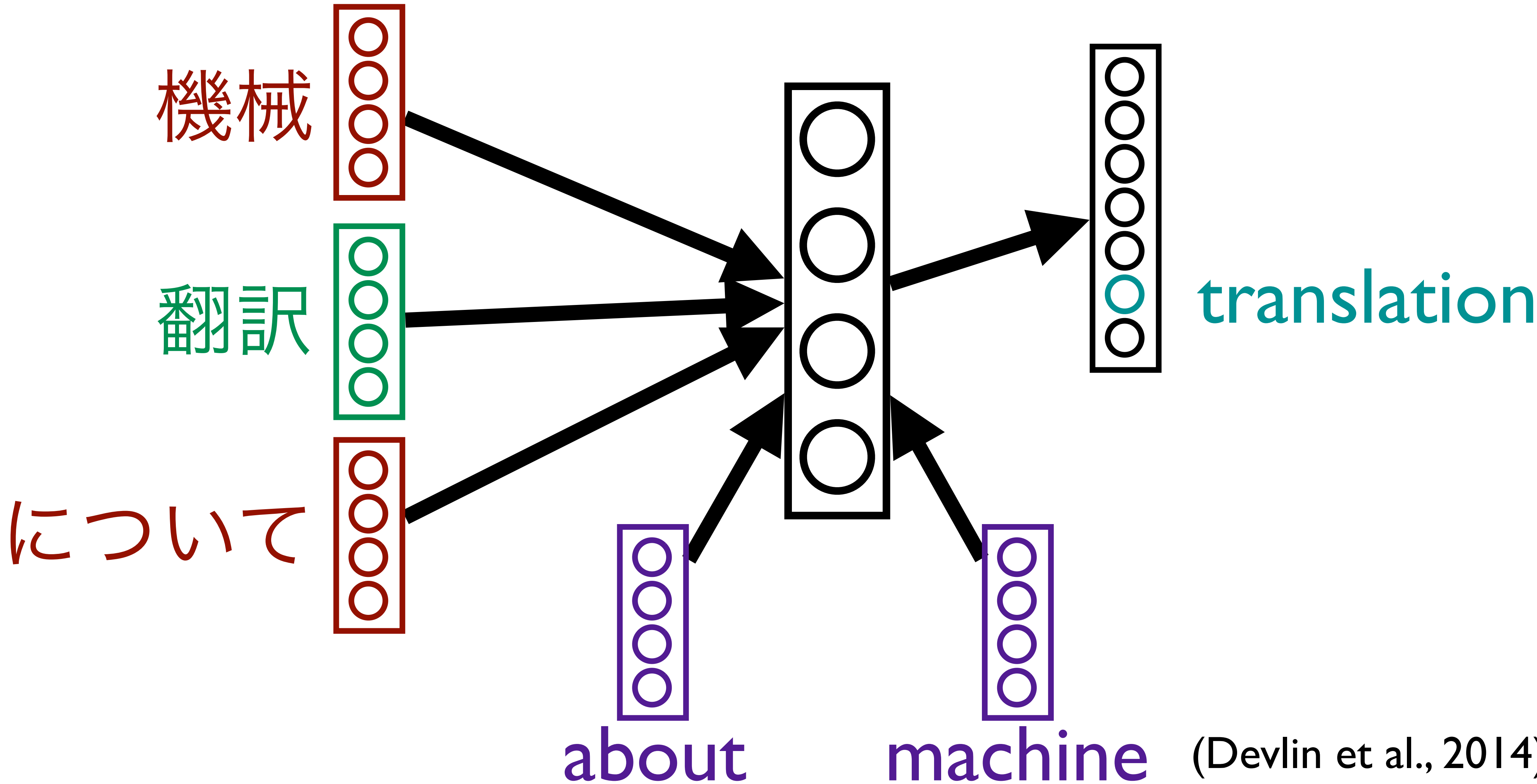
句単位のRecursive AE

(Zhang et al., 2014)



- 各言語独立にRecursive AEのエラーが最小になる木構造を推定

単語単位のFFNN



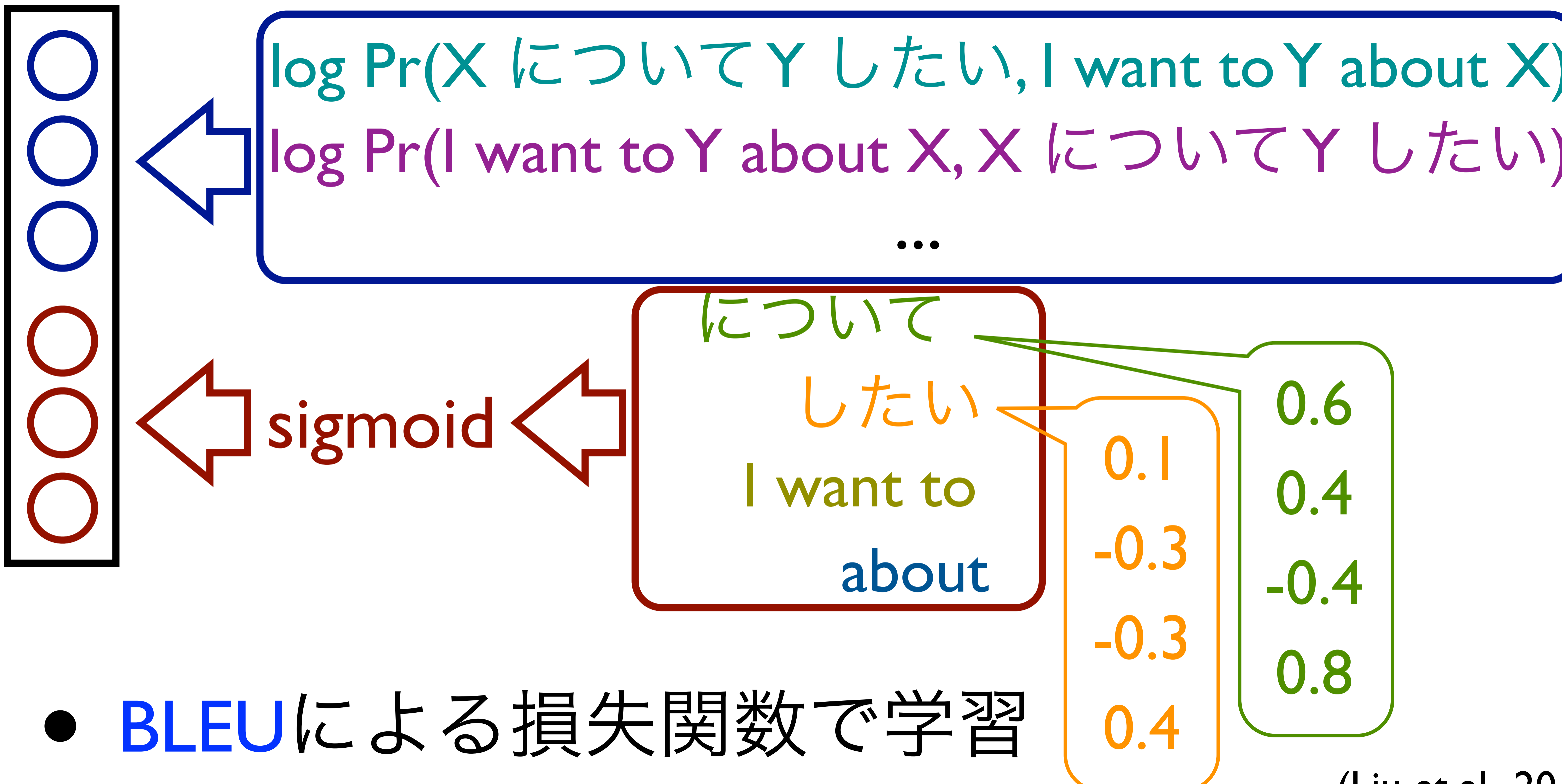
(Devlin et al., 2014)

$Pr($translation|about machine, 機械翻訳について$)$

- 単語アライメント単位の計算により、句単位の制約を排除

addNNによるモデル

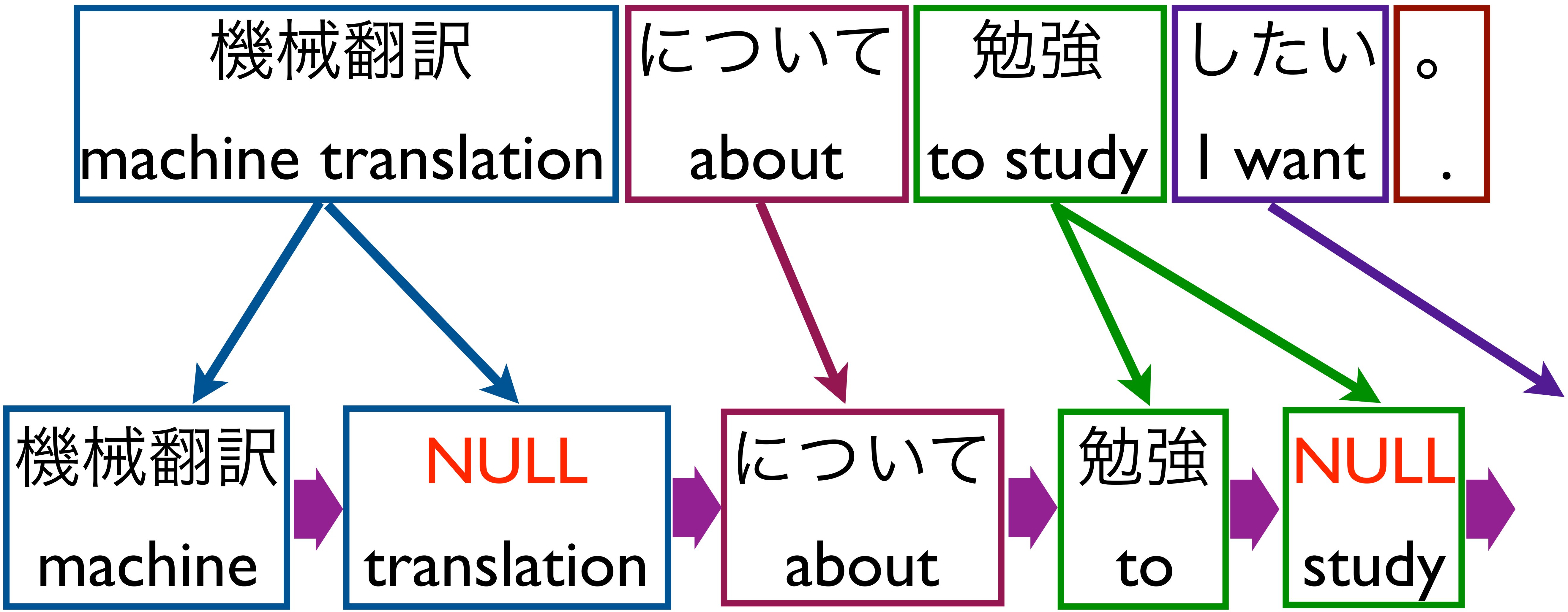
h(X について Y したい, I want to Y about X) =



- BLEUによる損失関数で学習

(Liu et al., 2013)

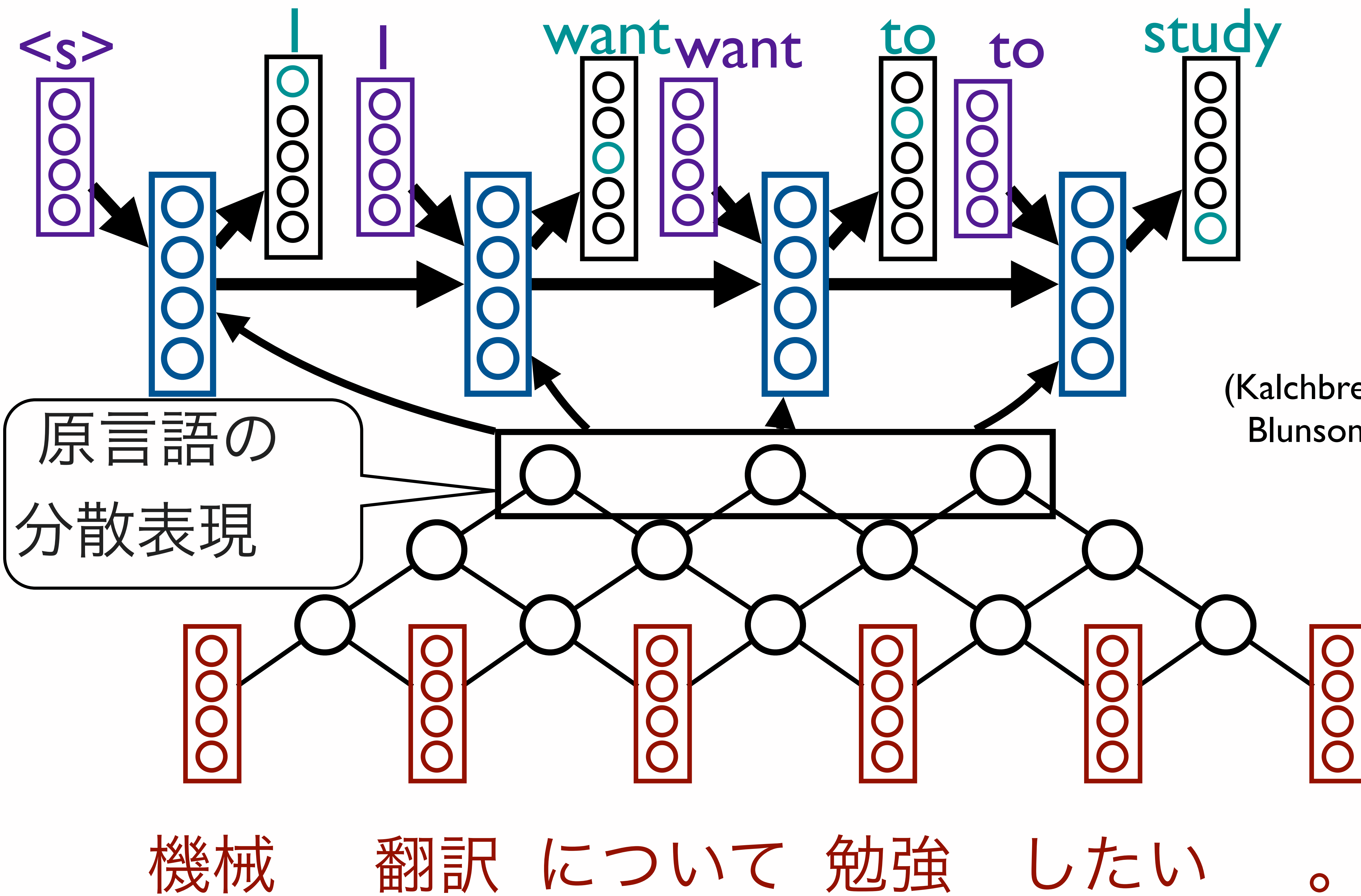
句によるRNN



(Wu et al., 2014)

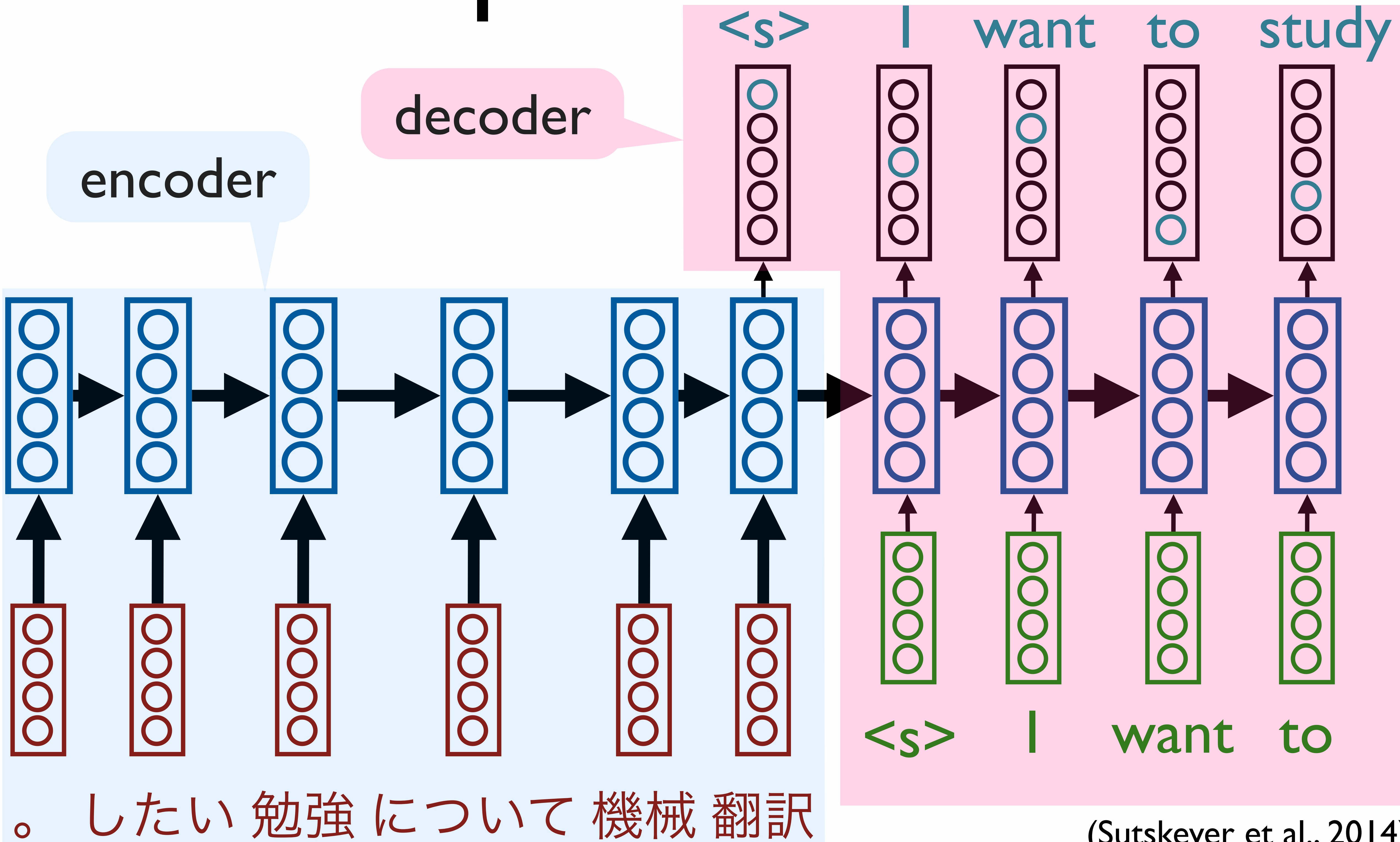
- 句単位のモデルを単語単位に展開

Convolution+RNN



(Kalchbrenner and Blunsom, 2013)

Sequenceモデル

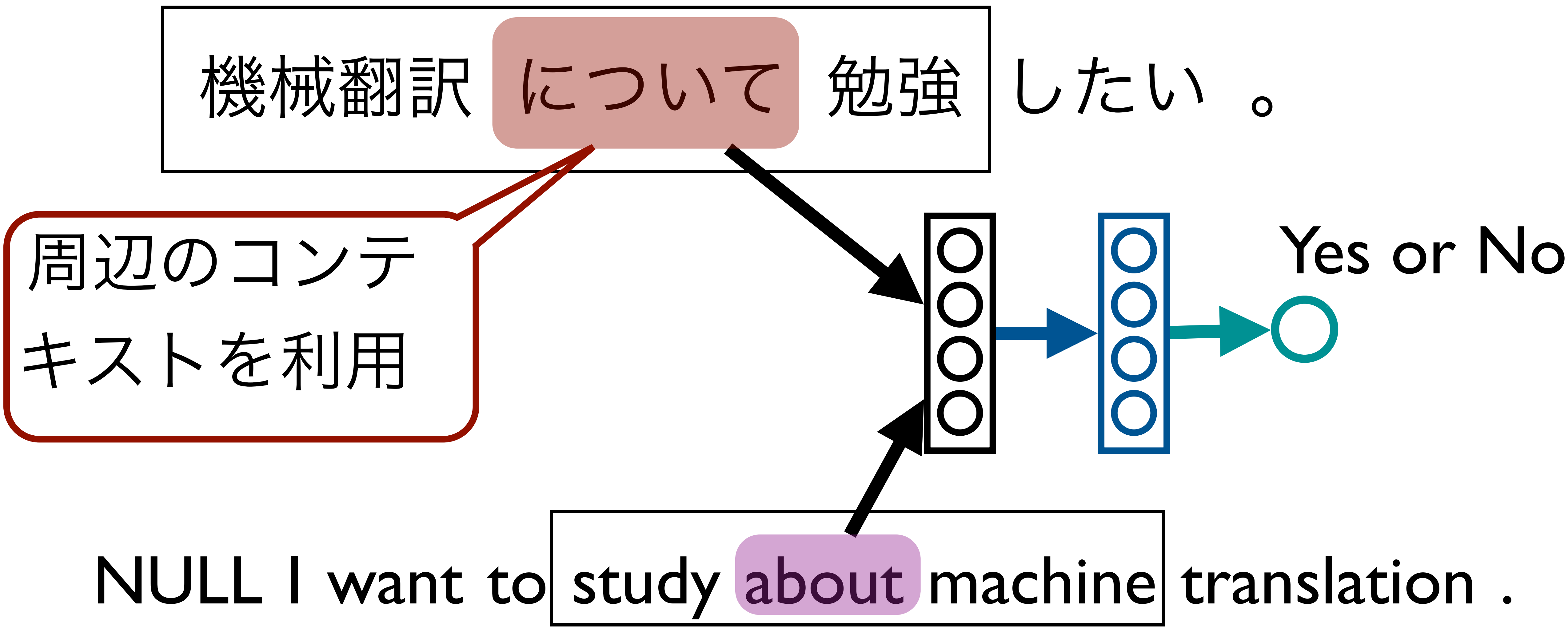


(Sutskever et al., 2014)

単語アライメント

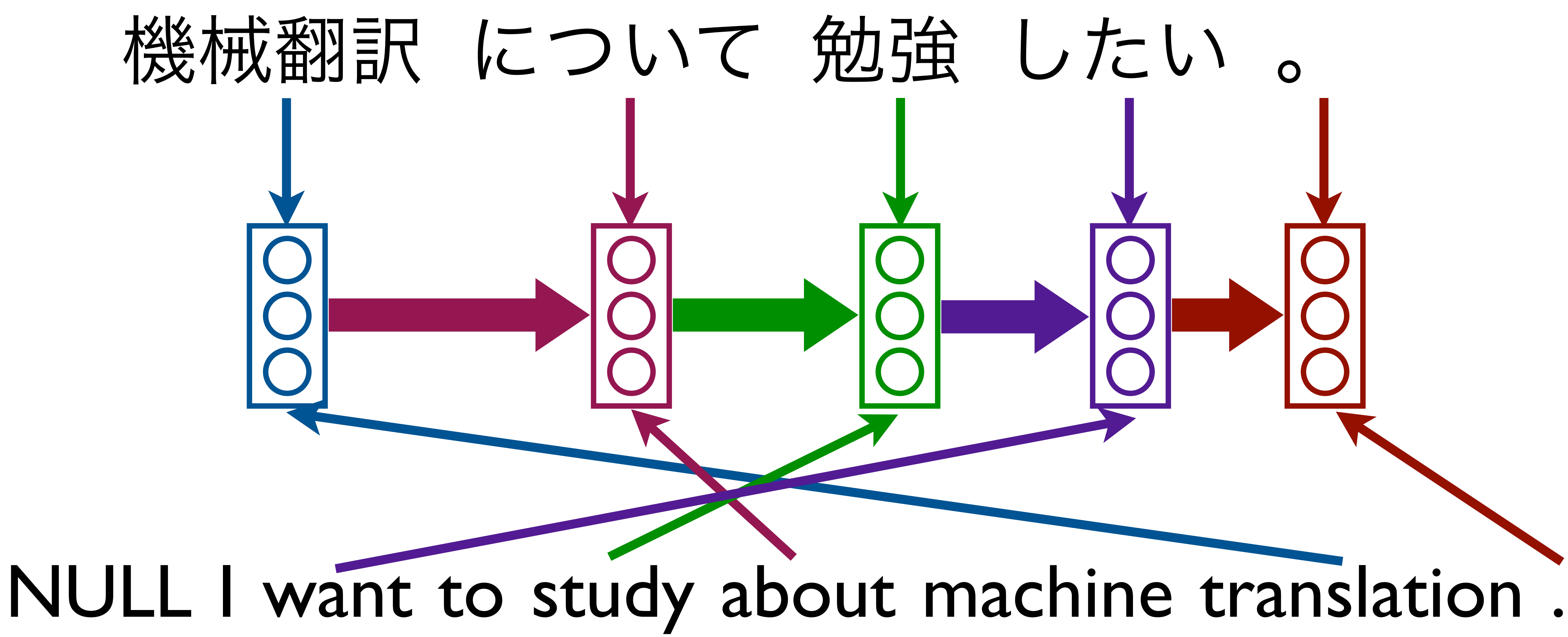
単語アライメント: FFNN

(Yang et al., 2013)



- 単語アライメントが付与されたデータから学習

単語アライメント: RNN



(Tamura et al., 2014)

- 単語アライメントの全ての履歴を表現 + noise contrastive estimateによる教師無し学習

まとめ

- ニューラルネットワークの機械翻訳への応用
    - 言語モデル、翻訳モデル、単語アライメント
- 始まったばかりなのに、数多くの研究
    - 全てを列挙できない: 似たり寄ったり
- 今後に期待

参考文献

- G. Cybenko. 1989. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314.
- Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380, Baltimore, Maryland, June. Association for Computational Linguistics.
- Jeffrey L. Elman. 1990. Finding structure in time. *Cognitive Science*, 14(2):179–211.
- Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza Aaron C. Courville, and Yoshua Bengio. 2013. Maxout networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 1319–1327.
- Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing pages 1700–1709, Seattle, Washington, USA, October. Association for Computational Linguistics.

参考文献

- lemao liu, Taro Watanabe, Eiichiro Sumita, and Tiejun Zhao 2013. Additive neural networks for statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 791–801, Sofia, Bulgaria, August. Association for Computational Linguistics.
- Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. 2010. Recurrent neura network based language model. In INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010 pages 1045–1048.
- Jordan B. Pollack. 1990. Recursive distributed representations. *Artificial Intelligence*, 46(1-2):77–105, November.
- Holger Schwenk. 2007. Continuous space language models *Computer Speech and Language*, 21(3):492–518, July.
- Holger Schwenk. 2012. Continuous space translation models for phrase-based statistical machine translation. In *Pro ceedings of COLING 2012: Posters*, pages 1071–1080, Mumbai, India, December. The COLING 2012 Organizing Committee.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proc NIPS*, Montreal, CA.

参考文献

- Akihiro Tamura, Taro Watanabe, and Eiichiro Sumita. 2014 Recurrent neural networks for word alignment model In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) pages 1470–1480, Baltimore, Maryland, June. Association for Computational Linguistics.
- Youzheng Wu, Taro Watanabe, and Chiori Hori. 2014 Recurrent neural network-based tuple sequence model for machine translation. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1908–1917, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.
- Nan Yang, Shujie Liu, Mu Li, Ming Zhou, and Nenghai Yu 2013. Word alignment modeling with context dependent deep neural network. In *Proceedings of the 51st Annual Meet ing of the Association for Computational Linguistics (Volume 1 Long Papers)*, pages 166–175, Sofia, Bulgaria, August. Association for Computational Linguistics.
- Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. 2014. Bilingually-constrained phrase embeddings for machine translation. In *Proceedings of the 52nd Annual Meet ing of the Association for Computational Linguistics (Volume 1 Long Papers)*, pages 111–121, Baltimore, Maryland, June. Association for Computational Linguistics.