



# 自然言語処理における Deep Learning

---

東北大学大学院情報科学研究科

岡崎 直観 ([okazaki@ecei.tohoku.ac.jp](mailto:okazaki@ecei.tohoku.ac.jp))

<http://www.chokkan.org/>  
@chokkanorg



# 自然言語処理とは

- 言葉を操る賢いコンピュータを作る
  - 応用: 情報検索, 機械翻訳, 質問応答, 自動要約, 対話生成, 評判分析, SNS分析, ...
  - 基礎: 品詞タグ付け (形態素解析), チャンキング, 固有表現抽出, 構文解析, 共参照解析, 意味役割付与, ...
- 多くのタスクは「入力 $x$ から出力 $\hat{y}$ を予測」

$$\hat{y} = \operatorname{argmax}_{y \in Y} P(y|x)$$

※確率ではないモデルもあります

# 単語列からラベル: $\hat{y} = \operatorname{argmax}_{y \in Y} P(y|x)$

$x$ : 単語列

$P(y|x)$

$\hat{y}$

*The movie is the best I've ever seen!*



*The movie is coming soon on cinemas.*



*This movie is rubbish!!!*



評判分析の例:   

- モデル: ナイーブ・ベイズ, パーセプトロン, ロジスティック回帰, サポート・ベクトル・マシン

# 単語列から系列ラベル: $\hat{y} = \operatorname{argmax}_{y \in Y^m} P(y|x)$

(入力) *In March 2005, the New York Times acquired About, Inc.*

(品詞)	IN	NNP	CD	DT	NNP	NNP	NNP	VBD	NNP	NNP

(句)	O	B-NP	I-NP	B-NP	I-NP	I-NP	I-NP	B-VP	B-NP	B-NP
-----	---	------	------	------	------	------	------	------	------	------

(翻訳) 2005年 3月 , ニューヨーク・タイムズ は About 社 を 買収 した .

(対話) I heard Google and Yahoo were among the other bidders.

- モデル : 隠れマルコフモデル, 条件付き確率場, 符号・復号
- 探索法 : 点予測, 動的計画法, ビーム探索, ...

# 単語列から木構造: $\hat{y} = \operatorname{argmax}_{y \in \text{Gen}(x)} P(y|x)$

Stack

Queue

[ ROOT ] [ Economic news had little effect on financial markets . ]

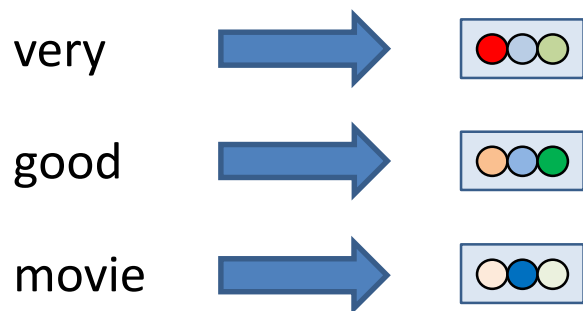
- モデル：確率的文脈自由文法，条件付き確率場
- 探索法：Shift-Reduce法，Eisner法，CKY法，最小全域木

# 深層学習ブームの幕開け (2012年頃)

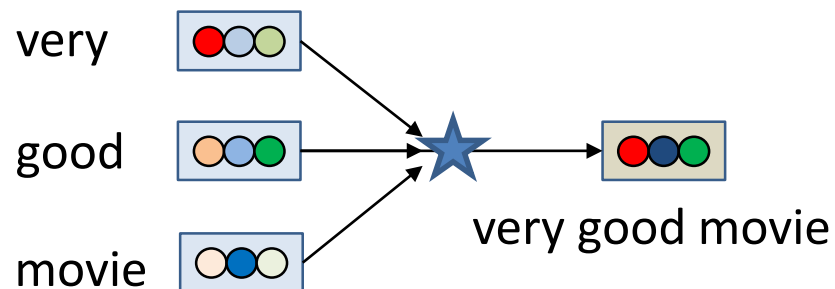
- 画像認識でブレークスルー
  - エラー率が10%以上減少 (ILSVRC 2012)
- 当初, 言語処理での衝撃は限定的
  - 文字や単語などの明確な特徴量があったため?
  - 最近ではDNNが様々なタスクで最高性能を達成
- 深層学習と言語処理の関わりは昔からあった
  - Neural Language Model (Bengio+ 03)
  - SENNA (CNNで汎用NLP) (Collobert+ 08)

# 言語処理における深層学習の進展

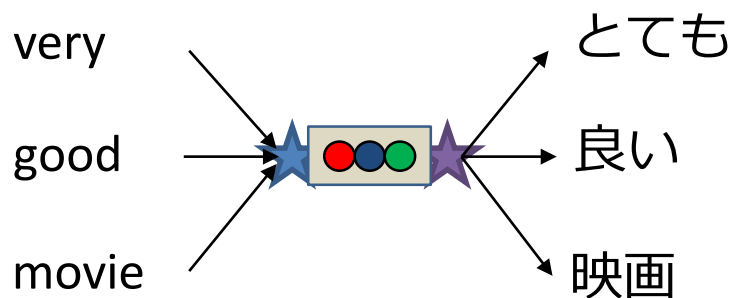
## • 単語の分散表現



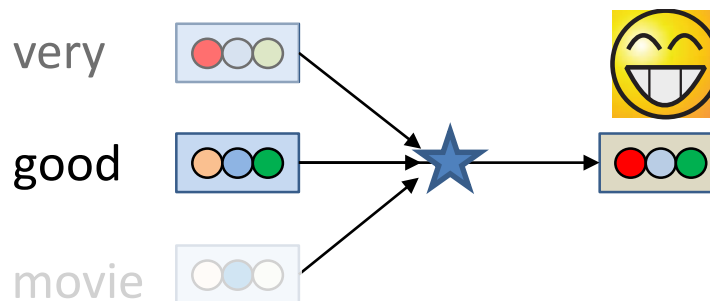
## • 分散表現の合成



## • エンコーダ・デコーダ

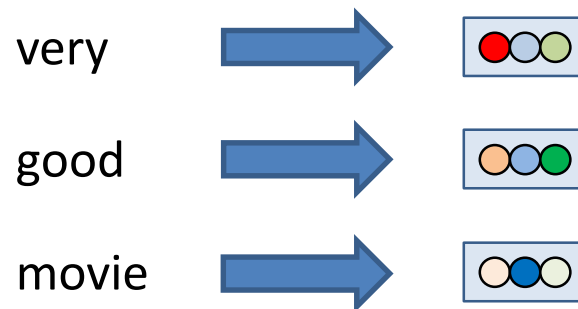


## • アテンション



# 単語の分散表現

(単語の意味をNNに埋め込む)





# 分散表現 (Hinton+ 1986)

- 局所表現 (local representation)

- 各概念に1つの計算要素 (記号, ニューロン, 次元) を割り当て



バス



萌えバス



- 分散表現 (distributed representation)

- 各概念は複数の計算要素で表現される
- 各計算要素は複数の概念の表現に関与する

←----- ニューロンの興奮パターン  
≡ベクトル表現



バス



トラック



萌え



萌えバス



<http://ja.wikipedia.org/wiki/富士急山梨バス> <http://saori223.web.fc2.com/>

# Skip-gram with Negative Sampling (SGNS)

(Mikolov+ 13)

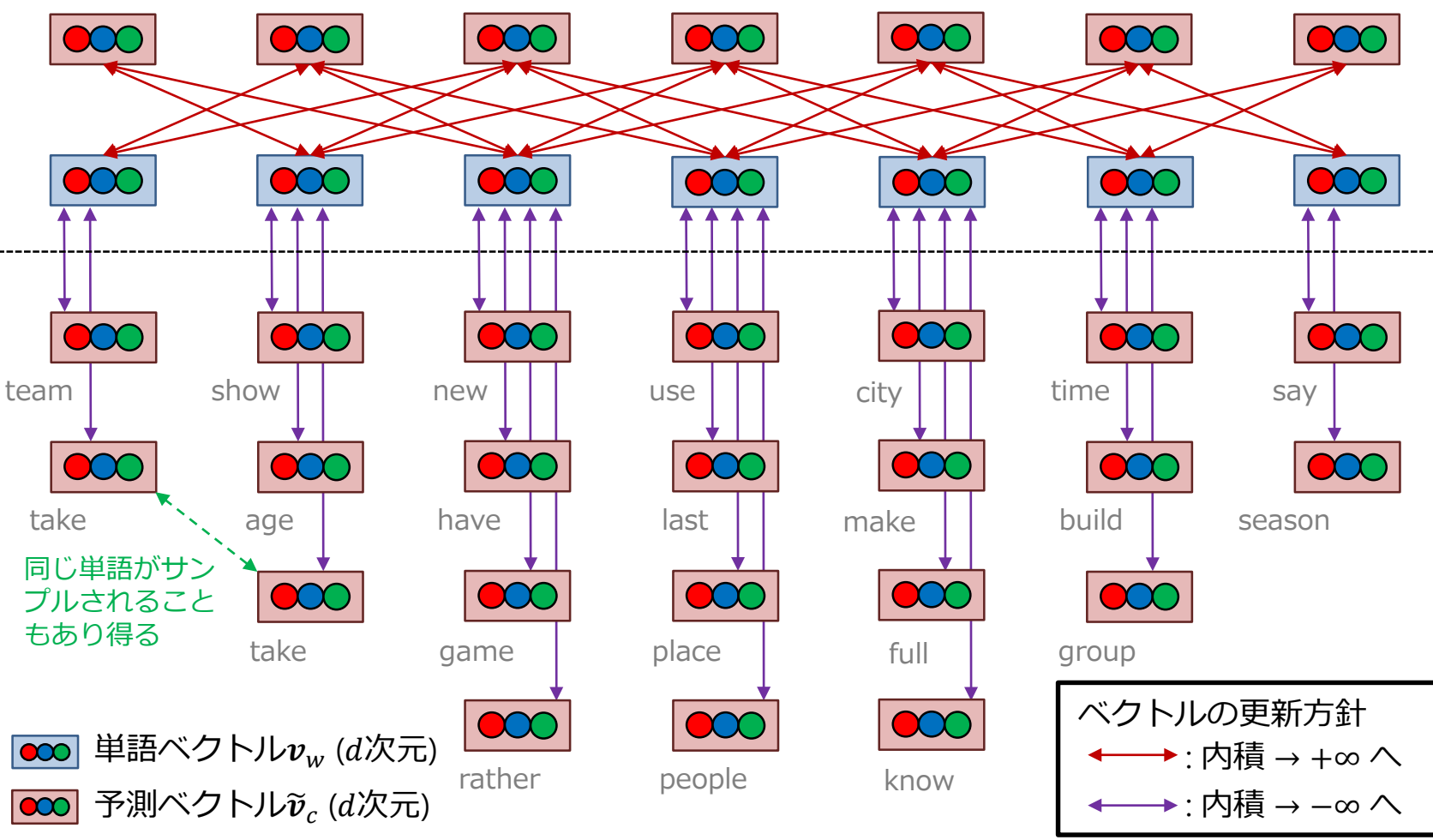
(文脈幅  $h = 2$ , 負例サンプル数  $k = 1$  の場合の例)

コーパス

pubs offer draught beer, cider, and wine


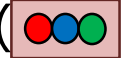
この文脈語を予測するよ  
うに更新

単語の分布から  
サンプリングし、  
これらが予測  
されないように更新  
(負例)



# ベクトルの更新方法 (確率的勾配降下法)

- 初期化:


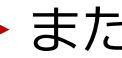


- $t \leftarrow 0$
- 単語ベクトル  :  $[0,1]$ の乱数で初期化
- 文脈ベクトル  : 0で初期化



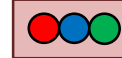
- 学習データの先頭から末尾の単語まで...

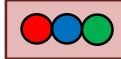
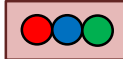

- $t \leftarrow t + 1$

$\alpha_0$ : 初期学習率 (例えば0.025)  
 $T$ : 単語の総出現回数

- 学習率  $\alpha = \alpha_0 \left(1 - \frac{t}{T+1}\right)$  を計算

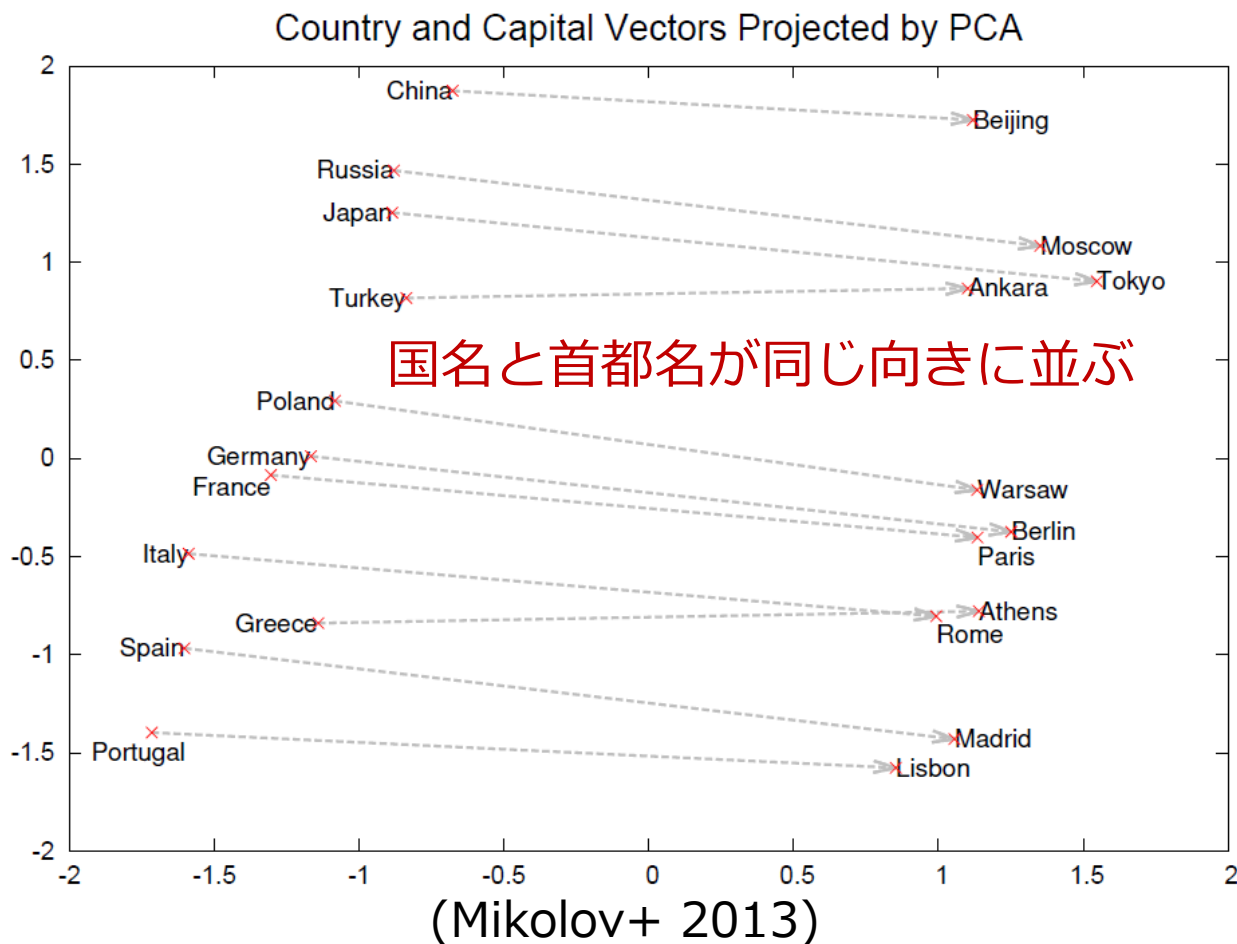
- その単語  と  または  で結ばれる  に関して
  - $g = \begin{cases} 1 - \sigma(\text{blue box} \cdot \text{red box}) & \text{red arrow (内積} \rightarrow +\infty \text{にしたい)とき} \\ \sigma(\text{blue box} \cdot \text{red box}) & \text{purple arrow (内積} \rightarrow -\infty \text{にしたい)とき} \end{cases}$

-   $\leftarrow$    $+ \alpha g$  

-   $\leftarrow$    $+ \alpha g$  

# SGNSで学習した分散表現は加法構成性を持つ

- 有名な例:  $\overrightarrow{\text{king}} - \overrightarrow{\text{man}} + \overrightarrow{\text{woman}} \approx \overrightarrow{\text{queen}}$



# アナロジータスクでの評価

Table 4: Comparison of publicly available word vectors on the Semantic-Syntactic Word Relationship test set, and word vectors from our models. Full vocabularies are used.

Model	Vector Dimensionality	Training words	Accuracy [%]		
			Semantic	Syntactic	Total
Collobert-Weston NNLM	50	660M	9.3	12.3	11.0
Turian NNLM	50	37M	1.4	2.6	2.1
Turian NNLM	200	37M	1.4	2.2	1.8
Mnih NNLM	50	37M	1.8	9.1	5.8
Mnih NNLM	100	37M	3.3	13.2	8.8
Mikolov RNNLM	80	320M	4.9	18.4	12.7
Mikolov RNNLM	640	320M	8.6	36.5	24.6
Huang NNLM	50	990M	13.3	11.6	12.3
Our NNLM	20	6B	12.9	26.4	20.3
Our NNLM	50	6B	27.9	55.8	43.2
Our NNLM	100	6B	34.2	<b>64.5</b>	50.8
CBOW	300	783M	15.5	53.1	36.1
Skip-gram	300	783M	<b>50.0</b>	55.9	<b>53.3</b>

Mikolov+ (2013)

Semanticの例: Athens Greece Tokyo Japan

Syntacticの例: cool cooler deep deeper

# GloVe (Pennington+ 2014)

(最小二乗法による単語ベクトルの学習)

AdaGrad  
(SGD)で学習

単語の総数  $V$       単語*i*と単語*j*の共起頻度  $m_{i,j}$

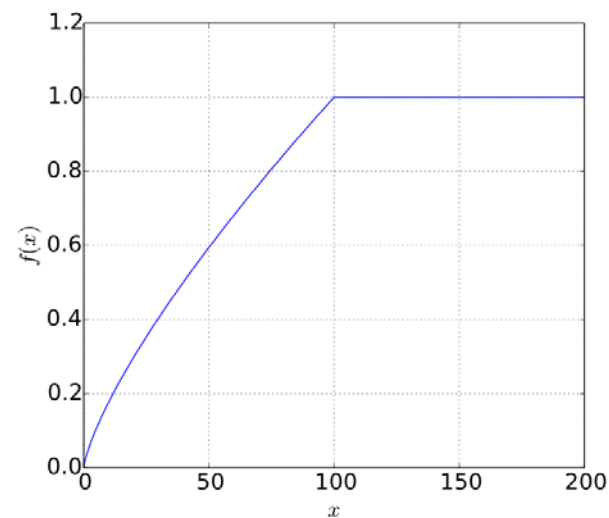
目的関数: 
$$J = \sum_{i,j=1}^V f(m_{i,j}) (\mathbf{v}_i^T \tilde{\mathbf{v}}_j + b_i + \tilde{b}_j - \log m_{i,j})^2$$

1系統	単語 <i>i</i> のベクトル	単語 <i>i</i> のバイアス項
2系統	文脈 <i>j</i> のベクトル'	単語 <i>j</i> のバイアス項'

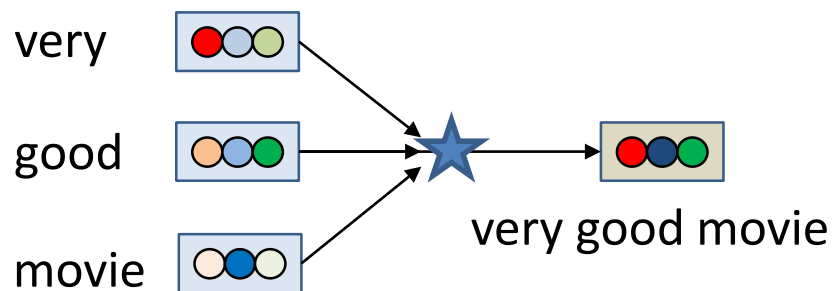
※各単語に対してパラメタが2系統あるのはSGNSと同様. 本研究は単語*i*のベクトルを最終的に $(\mathbf{v}_i + \tilde{\mathbf{v}}_i)$ とする (精度が向上する)

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & (\text{if } x < x_{\max}) \\ 1 & (\text{otherwise}) \end{cases}$$

$x_{\max} = 100, \alpha = 0.75$  の場合  $\rightarrow$



# 分散表現の合成 (文の意味をNNで計算)



# 句や文の分散表現をDNNで計算する

- 単語の分散表現が有用性が実証された！
- 次は句や文の分散表現を獲得したい
- **構成性の原理**に基づき，単語の分散表現から句や文の分散表現を合成する
  - 句や文の意味は，その構成要素の意味とその合成手続きから計算できる



# 句ベクトルの合成 (Mitchell+ 2010)

- 構成性の原理に基づき一般的な式を導入

$$\mathbf{p} = f(\mathbf{u}, \mathbf{v}, R, K)$$

- $\mathbf{u}, \mathbf{v}$ : 2つの単語に対応するベクトル
- $f$ : ベクトルから句ベクトルを合成する関数
- $R$ :  $\mathbf{u}$ と $\mathbf{v}$ の間の文法的な関係 (Partee 1995)
- $K$ : 合成に必要な背景知識 (Lakoff 1977)
- ※ 実際に実験した式の一般性はかなり狭い

# 句ベクトル合成の実験結果

(スピーアマンの順位相関係数; 単語ベクトルはlogを取らないPMI)

Model	Function	JJ-NN	NN-NN	VB-NN
Additive	$p_i = u_i + v_i$	.36	.39	.30
Kintsch	$p_i = u_i + v_i + n_i$	.32	.22	.29
Multiplicative	$p_i = u_i \cdot v_i$	.46	.49	.37
Tensor product	$p_{i,j} = u_i \cdot v_j$	.41	.36	.33
Circular convolution	$p_i = \sum_j u_i \cdot v_{(i-j) \bmod n}$	.09	.05	.10
Weighted additive	$p_i = \alpha u_i + \beta v_i$	.44	.41	.34
Dilation	$p_i = v_i \sum_j u_j u_j + (\lambda - 1) u_i \sum_j u_j v_j$	.44	.41	.38
Head only	$p_i = v_i$	.43	.34	.29
Target unit	$p_i = v_i(t_1 t_2)$	.43	.17	.24
Human		.52	.49	.55

- dilation, multiplicative, (weighted) additiveあたりがよい性能

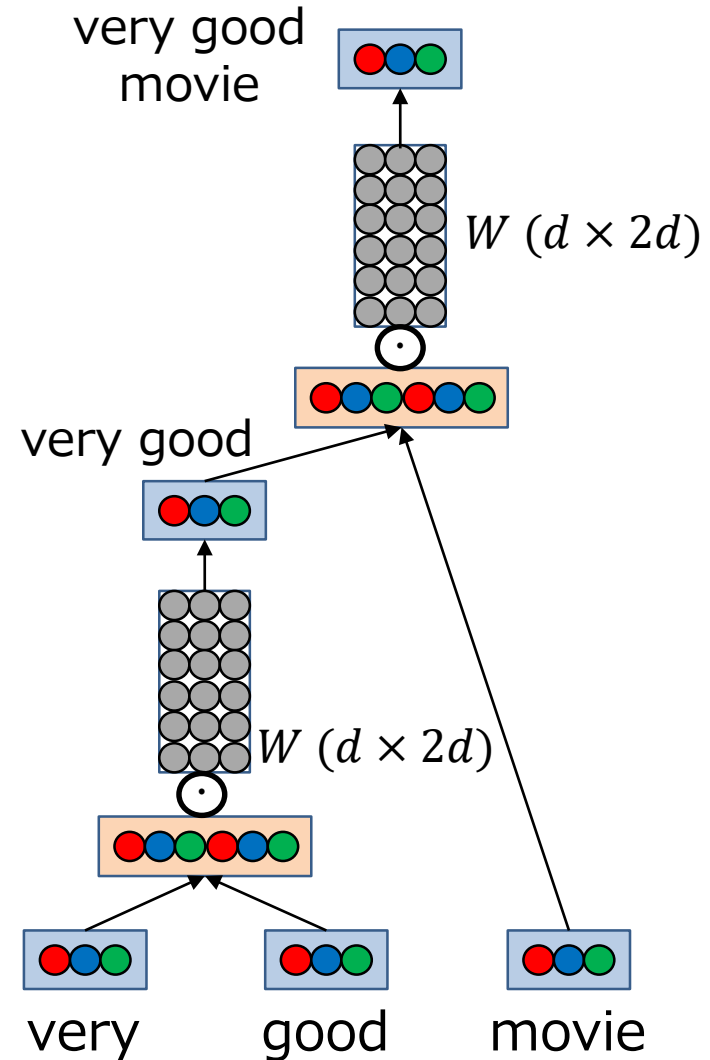
# Recursive Neural Network (RNN)

(Socher+ 2011)

- 句ベクトルを次式で合成

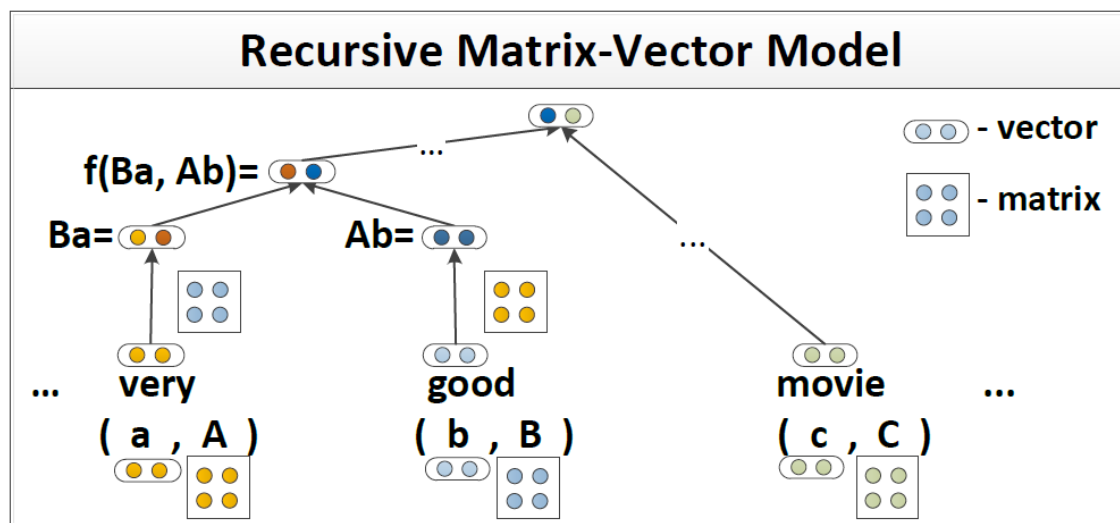
$$\mathbf{p} = f(\mathbf{u}, \mathbf{v}) = g\left(W \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}\right)$$

- $W: \mathbb{R}^{2d} \rightarrow \mathbb{R}^d$ の変換行列 ( $d \times 2d$ )
- $g$ : 活性化関数 ( $\sigma$ や $\tanh$ )
- 文の句構造に従って再帰的に句 (文) ベクトルを計算
- $W$ はオートエンコーダーやタスクでの誤差を用いて学習
- 単語ベクトルも同時に学習
  - ニューラル言語モデル (Collobert+ 2008) 等で初期化



# Matrix-Vector Recursive Neural Network (MV-RNN) (Socher+ 2012)

- 各単語を
  - ベクトル (意味)
  - 行列 (合成方法)で表現



(Socher+ 2012)

- 句のベクトル  $\mathbf{p}$  と行列  $P$  を再帰的に合成していく

$$\mathbf{p} = f_{A,B}(\mathbf{a}, \mathbf{b}) = f(B\mathbf{a}, A\mathbf{b}) = g\left(W \begin{bmatrix} B\mathbf{a} \\ A\mathbf{b} \end{bmatrix}\right)$$

$$P = f_M(A, B) = W_M \begin{bmatrix} A \\ B \end{bmatrix}$$

- 句の評価極性や関係ラベルを教師信号として学習

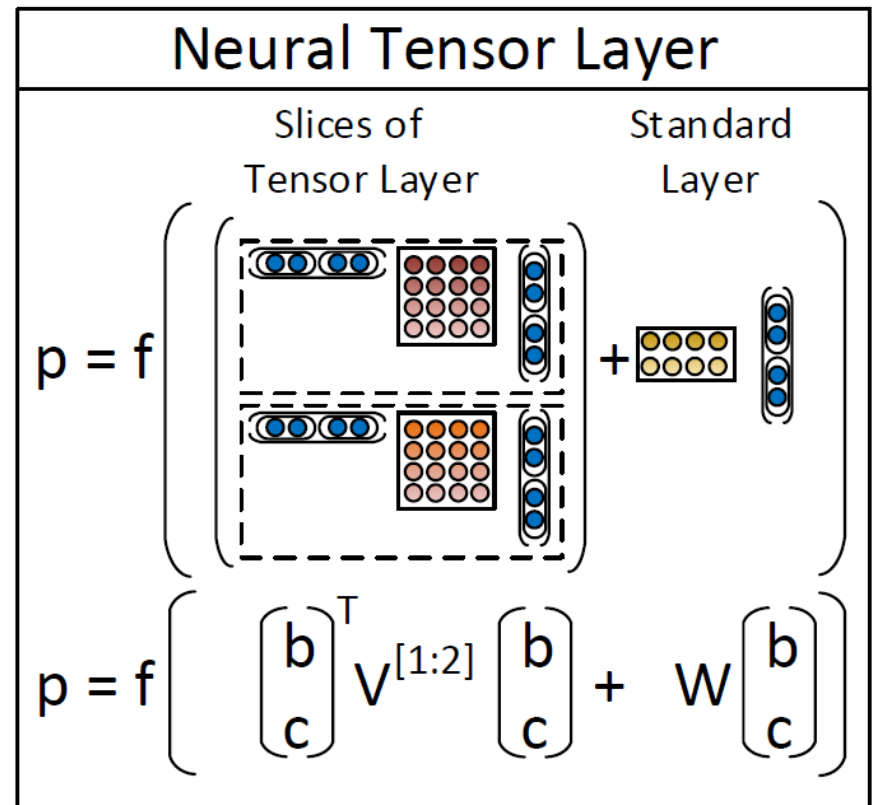
$$\mathbf{y}_p = \text{softmax } W_{\text{label}} \mathbf{p}$$

# Recursive Neural Tensor Network

(Socher+ 2013)

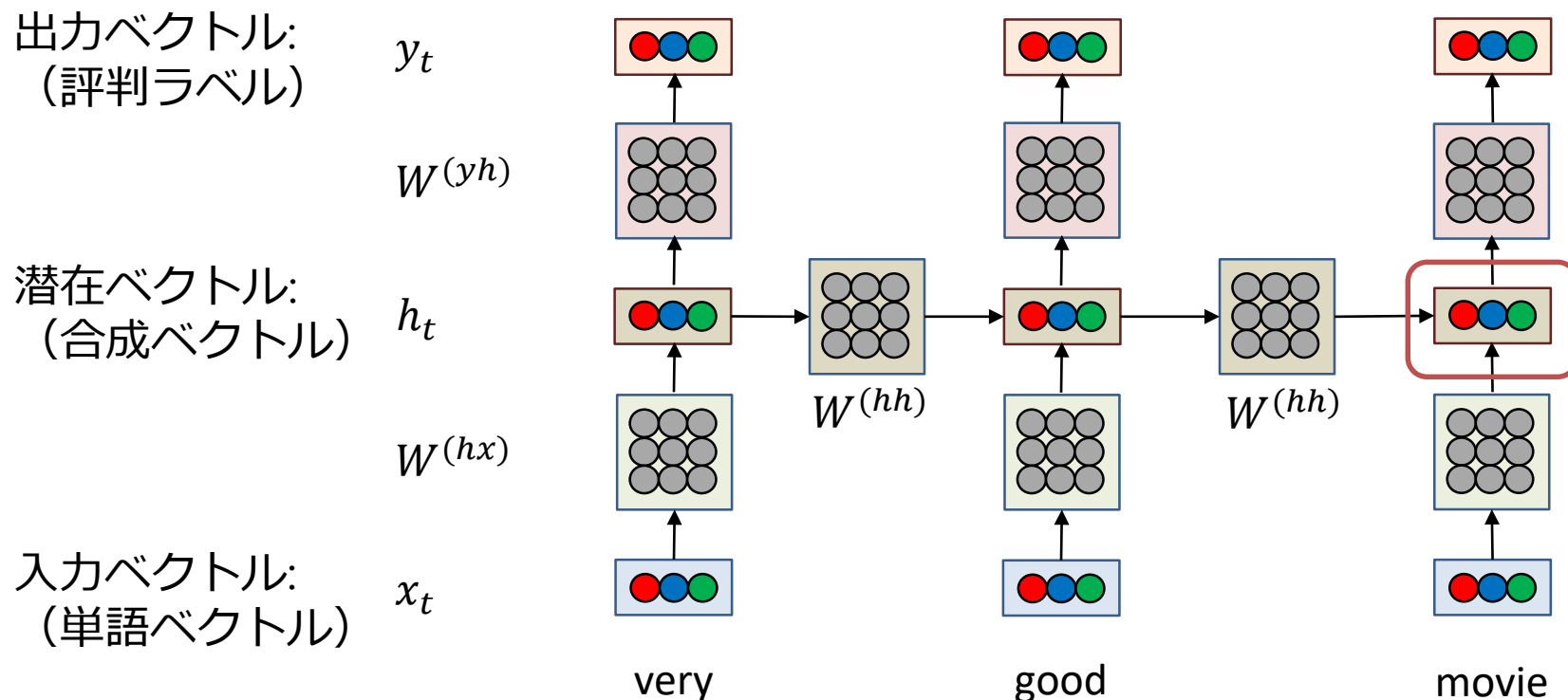
- MV-RNNは全ての単語が行列を持つので，学習するパラメータが多すぎる
- テンソルで単語ベクトルを行列に変換してから，単語ベクトルとの積を計算

Model	Fine-grained		Positive/Negative	
	All	Root	All	Root
NB	67.2	41.0	82.6	81.8
SVM	64.3	40.7	84.6	79.4
BiNB	71.0	41.9	82.7	83.1
VecAvg	73.3	32.7	85.1	80.1
RNN	79.0	43.2	86.1	82.4
MV-RNN	78.7	44.4	86.8	82.9
RNTN	<b>80.7</b>	<b>45.7</b>	<b>87.6</b>	<b>85.4</b>



(Socher+ 2013)

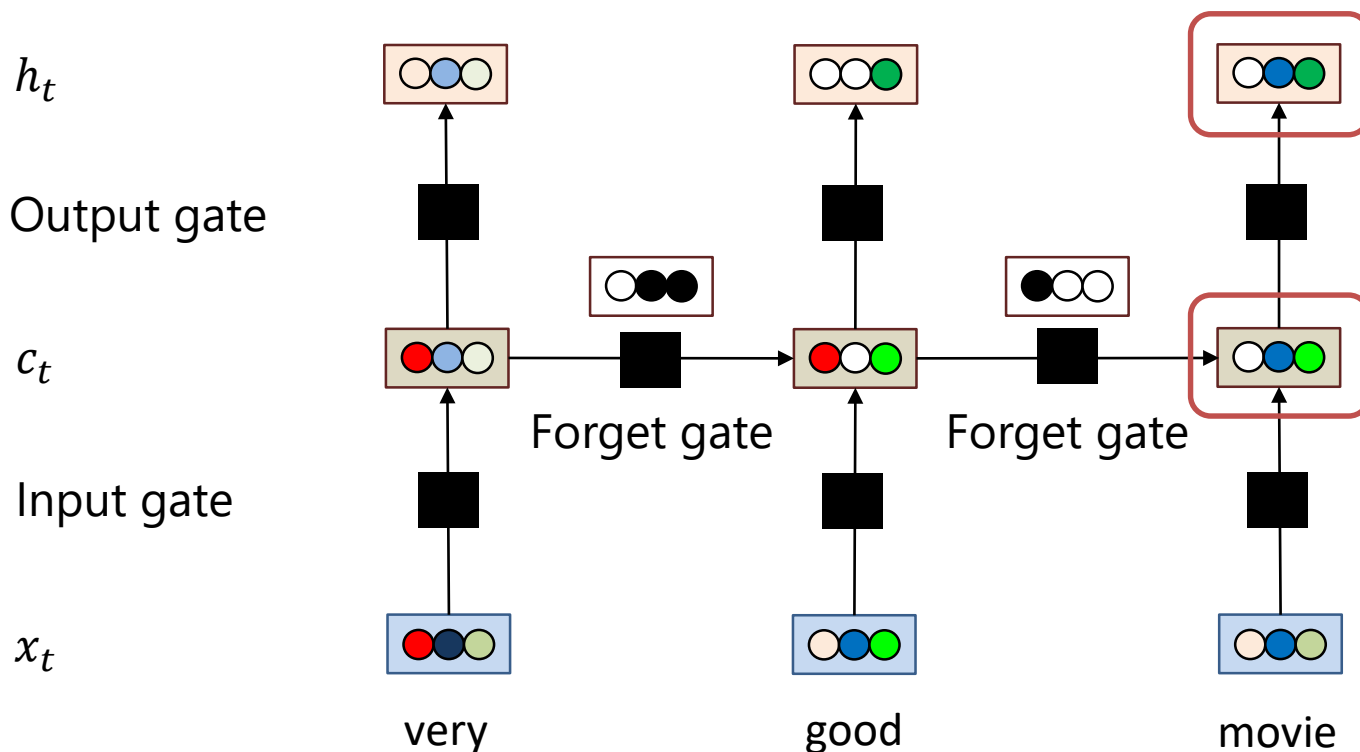
# Recurrent Neural Network (RNN) (Sutskever+ 11)



$$\text{潜在ベクトル: } h_t = \sigma(W^{(hx)}x_t + W^{(hh)}h_{t-1} + b^{(h)})$$

$$\text{出力ベクトル: } y_t = \sigma(W^{(yh)}h_t + b^{(y)})$$

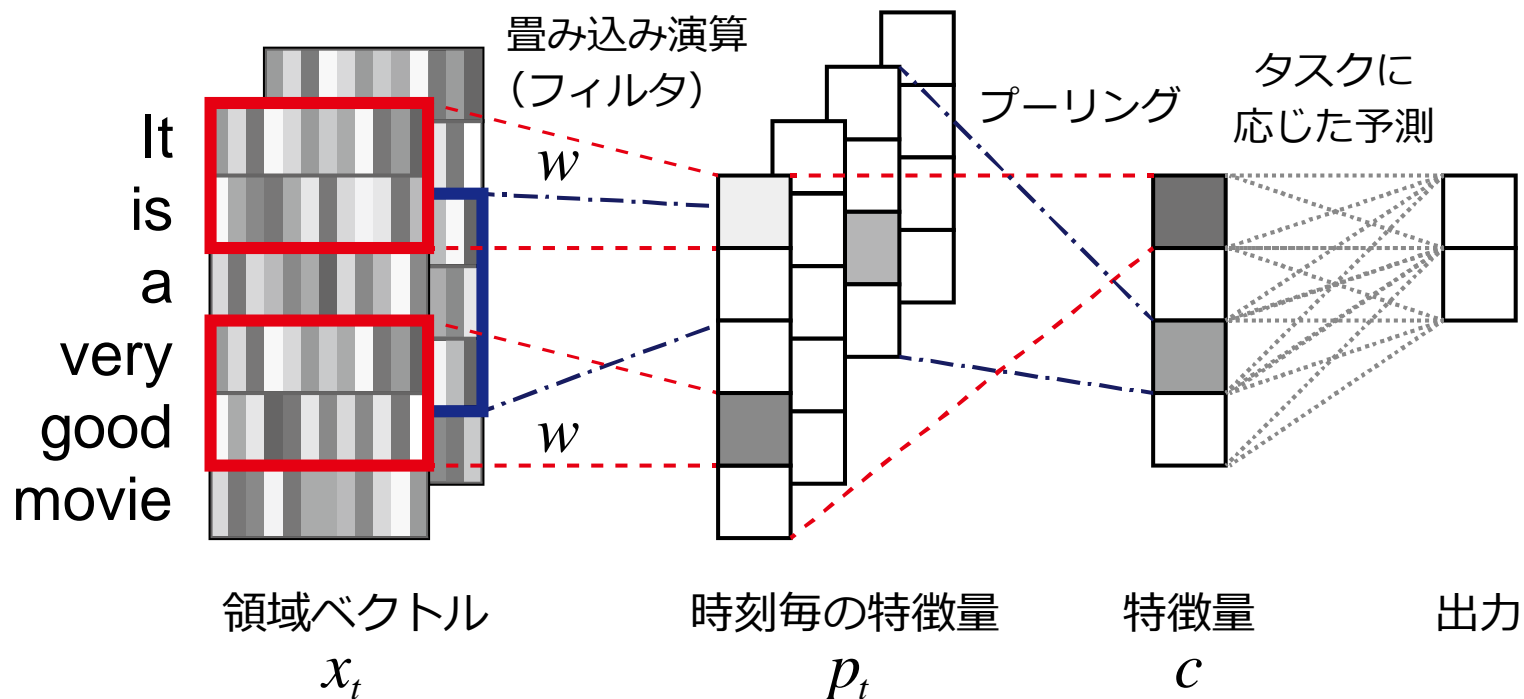
# Long Short-Term Memory (LSTM) (Graves 13)



- 各ゲートはマスクの役割を担う（ベクトルの要素ごとの積）
- 各ゲートのマスクパターンは入力 $x_t$ , 記憶 $h_{t-1}$ , 出力 $h_{t-1}$ などで自動制御
- 長い系列での誤差逆伝搬時の勾配消失をゲートで防止する（→長期依存の保存）
- LSTMの代わりにGated Recurrent Unit (GRU) も用いることも多い

# Convolutional Neural Network (CNN)

(Kim 14)



- 領域ベクトル:  $x_{t:t+\delta} = x_t \oplus x_{t+1} \oplus \dots \oplus x_{t+\delta-1}$
- 時間毎の特徴量:  $p_t = g(w \cdot x_{t:t+\delta} + b)$
- 特徴量 (maxプーリング):  $c = \max_{1 < t < T - \delta + 1} p_t$

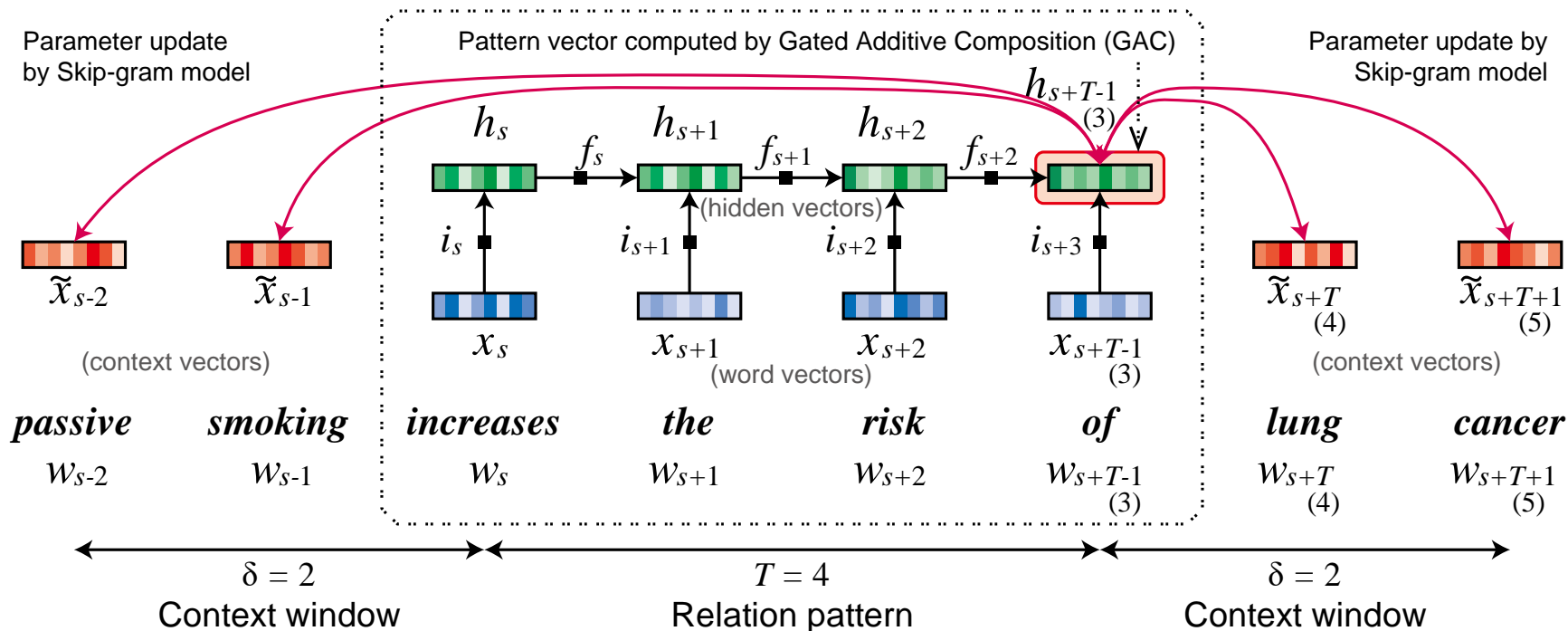


# Stanford Sentiment Treebank での性能

Method	Fine-grained	Binary
RAE (Socher et al., 2013)	43.2	82.4
MV-RNN (Socher et al., 2013)	44.4	82.9
RNTN (Socher et al., 2013)	45.7	85.4
DCNN (Blunsom et al., 2014)	48.5	86.8
Paragraph-Vec (Le and Mikolov, 2014)	48.7	87.8
CNN-non-static (Kim, 2014)	48.0	87.2
CNN-multichannel (Kim, 2014)	47.4	<b>88.1</b>
DRNN (Irsoy and Cardie, 2014)	49.8	86.6
LSTM	46.4 (1.1)	84.9 (0.6)
Bidirectional LSTM	49.1 (1.0)	87.5 (0.5)
2-layer LSTM	46.0 (1.3)	86.3 (0.6)
2-layer Bidirectional LSTM	48.5 (1.0)	87.2 (1.0)
Dependency Tree-LSTM	48.4 (0.4)	85.7 (0.4)
Constituency Tree-LSTM		
– randomly initialized vectors	43.9 (0.6)	82.0 (0.5)
– Glove vectors, fixed	49.7 (0.4)	87.5 (0.8)
– Glove vectors, tuned	<b>51.0</b> (0.5)	88.0 (0.3)

Tai+ (2015)

# 関係パターンの意味合成 (Takase+ 16)

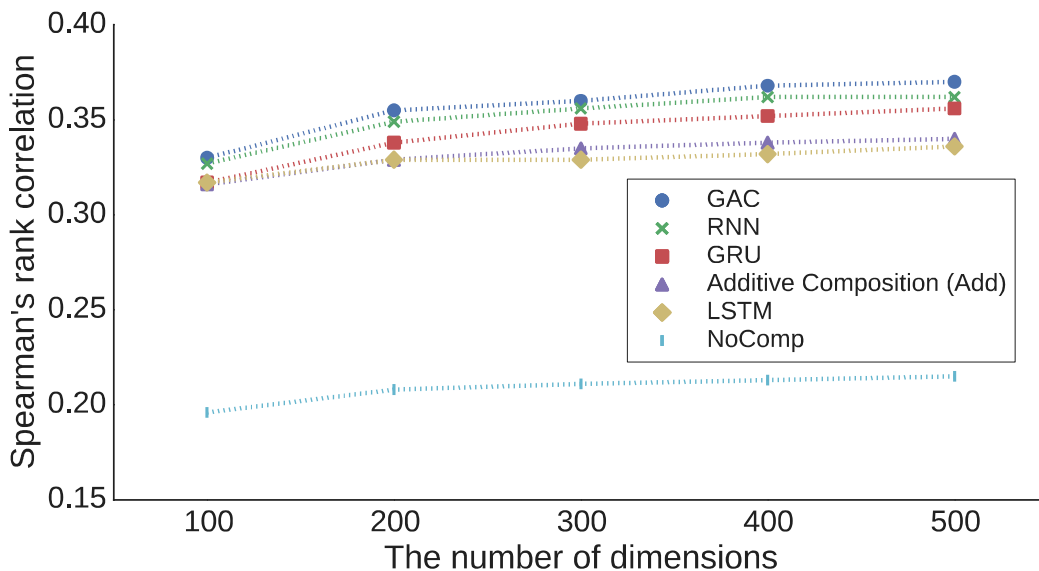


Input gate: 
$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1})$$

Forget gate: 
$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1})$$

Gated composition: 
$$h_t = g(f_t \odot h_{t-1} + i_t \odot x_t)$$

# 関係パターンの意味合成 (Takase+ 16)



状況	現在	後続の表現
Input-open	reimburse payable	for in
Input-close	a a	charter member of valuable member of
Forget-open	be be	eligible to participate require to submit
Forget-close	coauthor capital	of of

- 句の意味合成のための高精度かつ軽量な手法を提案
- 関係パターンの分散表現の評価データを構築
- 学習した関係パターンの分散表現が関係知識抽出に貢献することを実証

# 平均による句ベクトル近似の理論解析 (Tian+ 2017)

- 単語ベクトルの一般形として次式を考える

$$m_{w,c} = \gamma \cdot (F(P(c|w)) - \alpha(c) - \beta(w))$$

- PPMI, Skip-gram, GloVeはこの形で表される
- 句 $t_1 t_2$ のベクトルは単語 $t_1, t_2$ のベクトル平均で近似

$$\frac{1}{2}(\mathbf{v}_{t_1} + \mathbf{v}_{t_2})$$

$t_1$ が出現した後,  
 $t_2$ が続かない確率

- 句ベクトル近似の誤差のバウンドは,

$t_2$ が出現した前に  
 $t_1$ が現れない確率

$$\left\| \mathbf{v}_{t_1 t_2} - \frac{1}{2}(\mathbf{v}_{t_1} + \mathbf{v}_{t_2}) \right\| \leq \sqrt{\frac{1}{2}(\pi_{1 \setminus 2}^2 + \pi_{2 \setminus 1}^2 + \pi_{1 \setminus 2} \pi_{2 \setminus 1})}$$

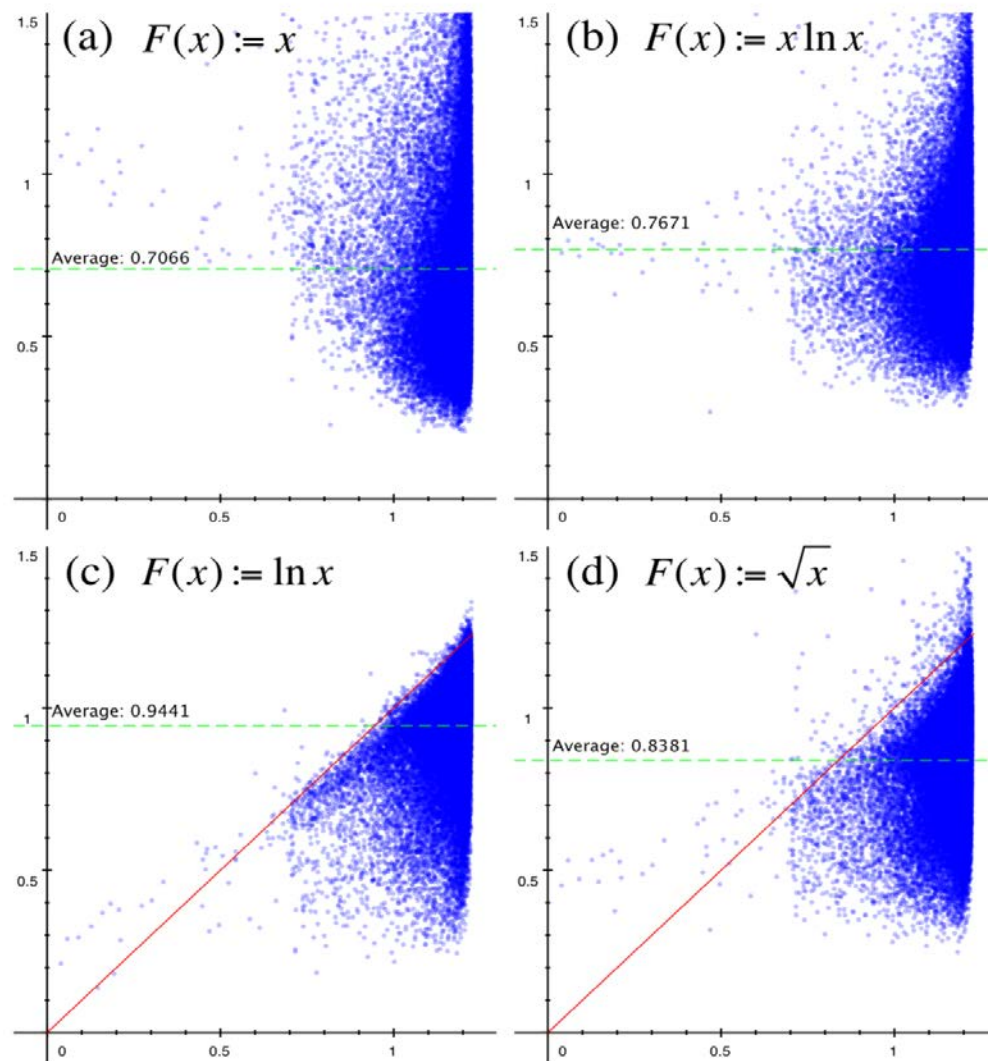
- ただし, 関数 $F(p)$ が満たすべき条件がつく
  - $\log p$ や $\sqrt{p}$ はOKだが,  $p$ や $p \log p$ では成り立たない

# 関数 $F(p)$ による近似誤差の違い (Tian+ 2017)

- 横軸: 句を構成する単語 $t_1, t_2$ のコーレクションの弱さ

$$\sqrt{\frac{1}{2}(\pi_{1\setminus 2}^2 + \pi_{2\setminus 1}^2 + \pi_{1\setminus 2}\pi_{2\setminus 1})}$$

- 縦軸: 実際にコーパスから求めた句ベクトルとの誤差

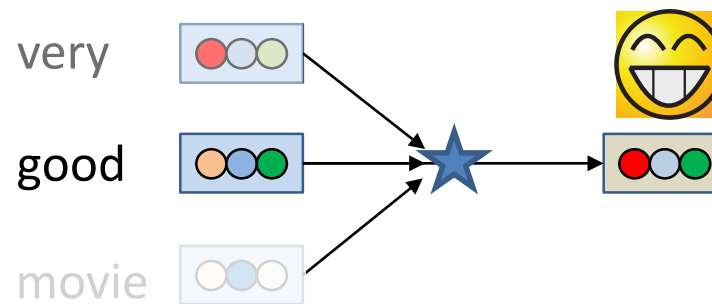
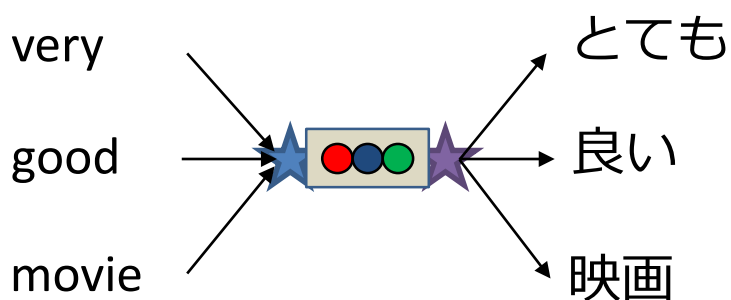


# 符号化・復号化

(分散表現から新たな文を生成)

## アテンション

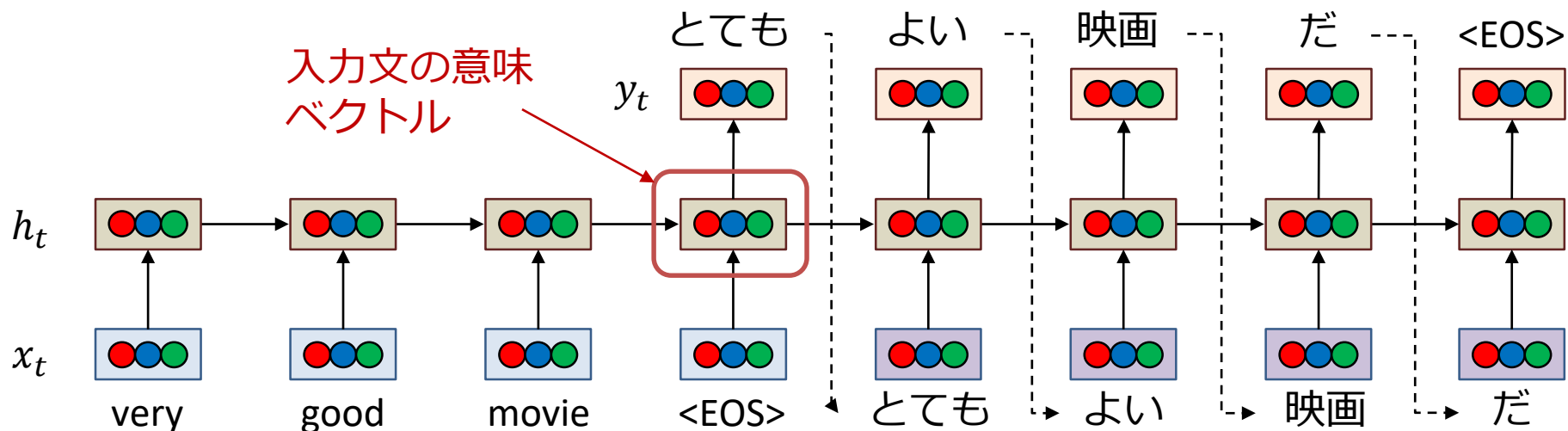
(注視点の自動学習)



# 分散表現から文を生成できるか？

- 単語から句や文の分散表現を合成できた！
- 句や文の分散表現から単語列を取り出せるか？
- 符号化・復号化 (encoder-decoder) モデル
  - 与えられた文の内容をベクトルで表現し, 文を出力する
  - 単語列から単語列を予測できる
- DNNの応用がさらに広がる
  - 機械翻訳 (Sutskever+ 14, Cho+ 14, Luong+ 15)
  - 対話文生成 (Vinyals+ 15)
  - 自動要約 (Rush+ 15, Takase+ 16)
  - 画像説明文の生成 (Vinyals+ 15)
  - 動画説明文の生成 (Laokulrat+ 16)

# Sequence-to-sequence (Sutskever+ 14; Cho+ 14)

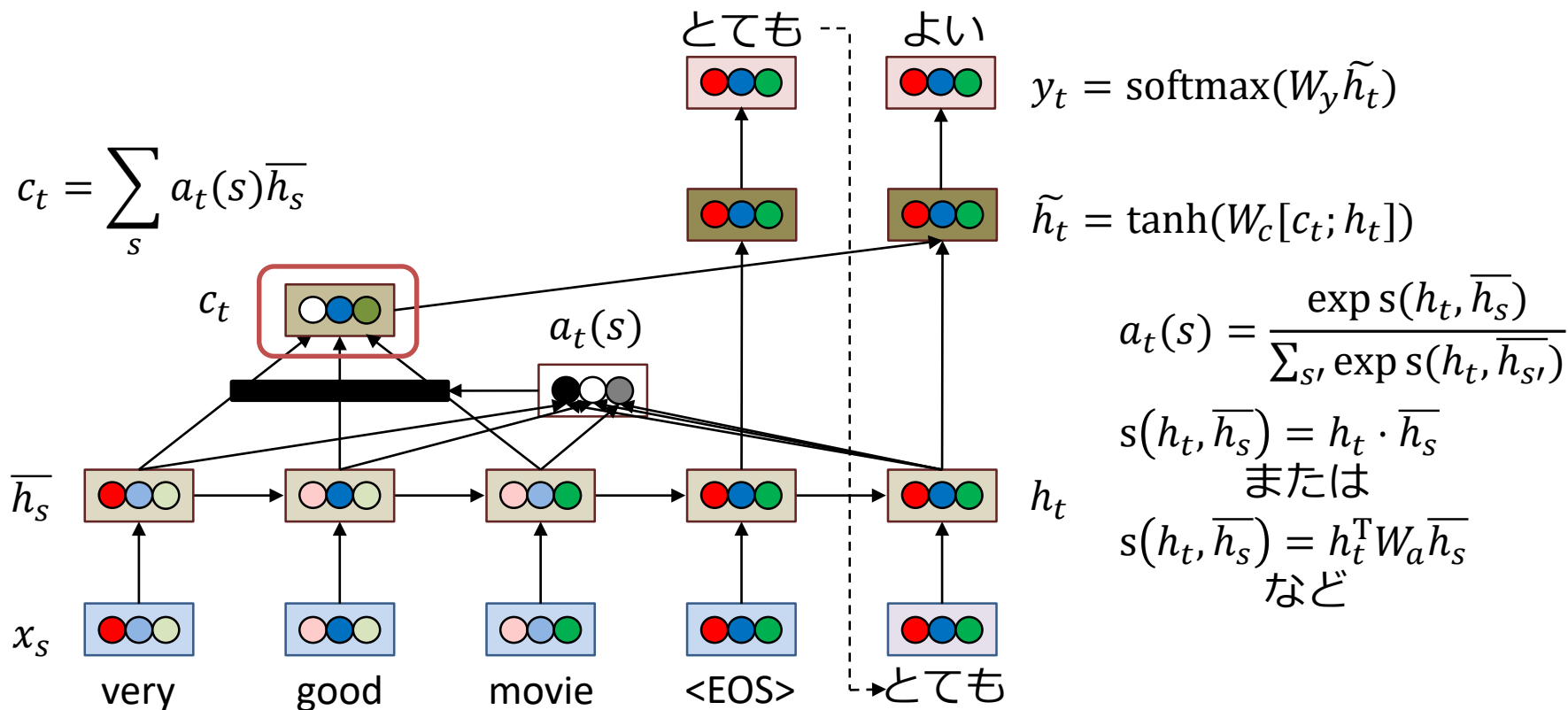


- 英語 ( $x_t$ ) から日本語 ( $y_t$ ) への機械翻訳の例
  - $h_t = \sigma(W^{(hx)}x_t + W^{(hh)}h_{t-1})$
  - $y_t = W^{(yh)}h_t$  ( $y_t$ は出力単語のスコアのベクトル表現)
- 出力単語を入力側に戻すことで, 翻訳履歴を考慮
- 実際にはRNNではなく2-layer LSTM等を用いる

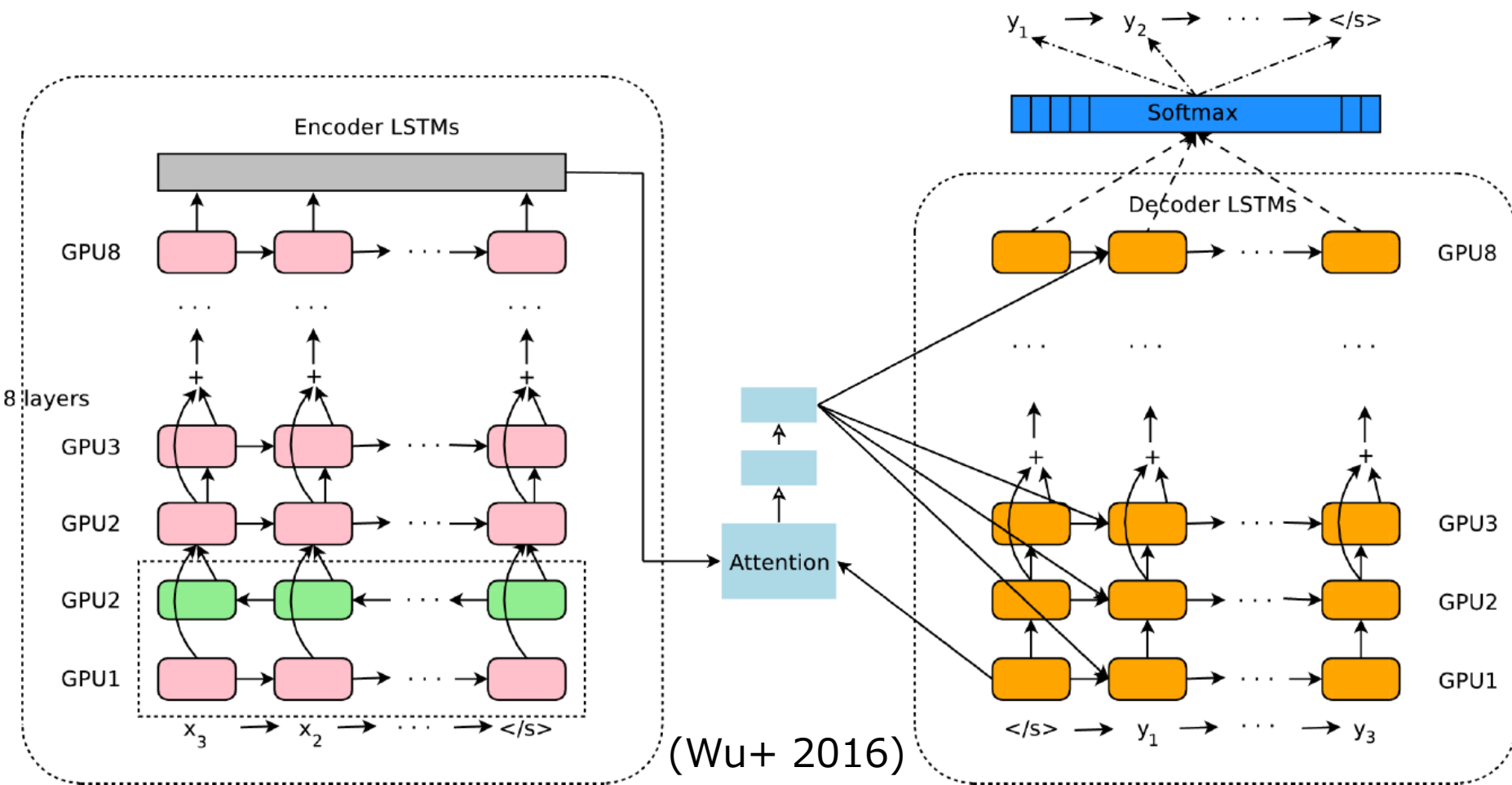


# アテンション (Bahdanau+ 15, Luong+, 15)

- 固定長のベクトルで文全体の意味を表現するには限界がある
- 位置 $t$ の単語を予測するとき、入力のどの単語に着目すべきかの重み $a_t(s)$ を求め、入力単語のベクトル $\bar{h}_s$ の重み付き平均ベクトル $c_t$ を用いて出力単語を生成する



# Googleのニューラル機械翻訳 (Wu+ 2016)



8層のLSTM (residual connection付き), 重みの量子化, Tensor Processing Unit, など

# 対話の生成 (Vinyals+ 15)

- 学習データ: OpenSubtitles
  - 映画の字幕から抽出した対話文 (約6200万文)
- Sequence-to-sequenceで生成した例

**Human:** *hello !*

**Machine:** *hello !*

**Human:** *how are you ?*

**Machine:** *i 'm good .*

**Human:** *what 's your name ?*

**Machine:** *i 'm julia .*

**Human:** *when were you born ?*

**Machine:** *july 20th .*

**Human:** *what year were you born ?*

**Machine:** *1977 .*

**Human:** *where are you from ?*

**Machine:** *i 'm out in the boonies.*

**Human:** *are you a man or a woman ?*

**Machine:** *i 'm a woman .*

**Human:** *why are we here ?*

**Machine:** *i 'm not sure .*

**Human:** *okay, bye !*

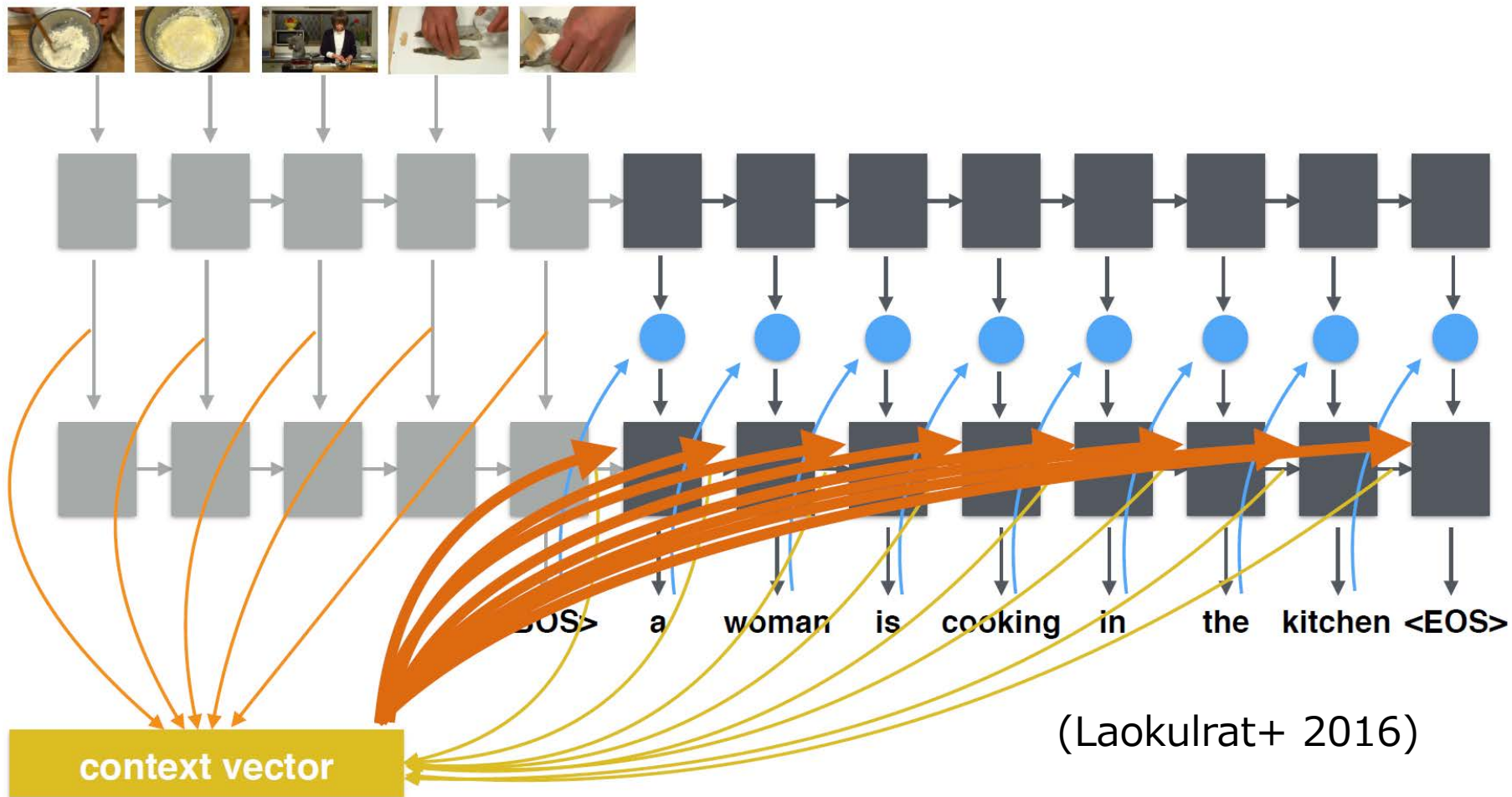
**Machine:** *bye .*

**Human:** *see you later .*

**Machine:** *bye .*

(Vinyals+ 2015)

# 動画記述文生成 (Laokulrat+ 16)



# 文章理解に向けた深層学習

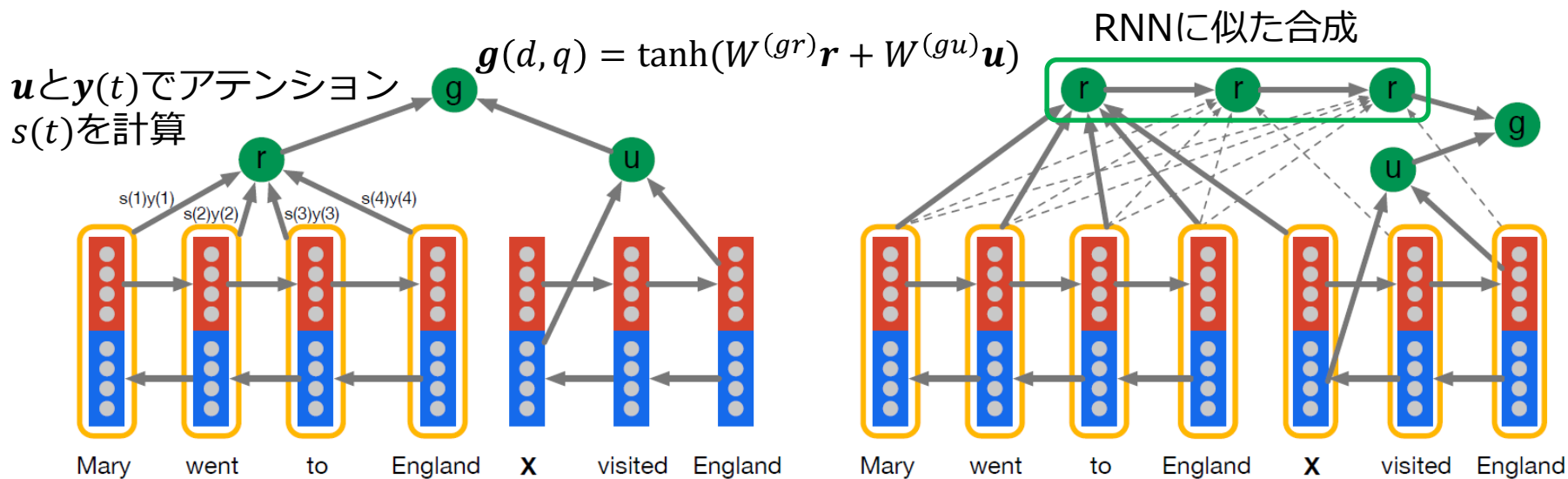
# 読解 (Hermann+ 15)

- 新聞記事と要約から穴埋め問題の訓練データを自動生成
  - 共参照を解消し, エンティティを匿名化 (n-gram等の単純な手法を排除)
  - 記事と質問の文脈を蓄積・処理する能力を測定することを意図
  - 訓練データ数: 約9万記事 (CNN), 約22万記事 (Daily Mail)

Original Version	Anonymised Version
<b>Context</b> The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the “Top Gear” host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisín Tymon “to an unprovoked physical and verbal attack.” ...	the <i>ent381</i> producer allegedly struck by <i>ent212</i> will not press charges against the “ <i>ent153</i> ” host , his lawyer said friday . <i>ent212</i> , who hosted one of the most - watched television shows in the world , was dropped by the <i>ent381</i> wednesday after an internal investigation by the <i>ent180</i> broadcaster found he had subjected producer <i>ent193</i> “ to an unprovoked physical and verbal attack . ” ...
<b>Query</b> Producer X will not press charges against Jeremy Clarkson, his lawyer says.	producer X will not press charges against <i>ent212</i> , his lawyer says .
<b>Answer</b> Oisín Tymon	<i>ent193</i>

(Hermann+ 2015)

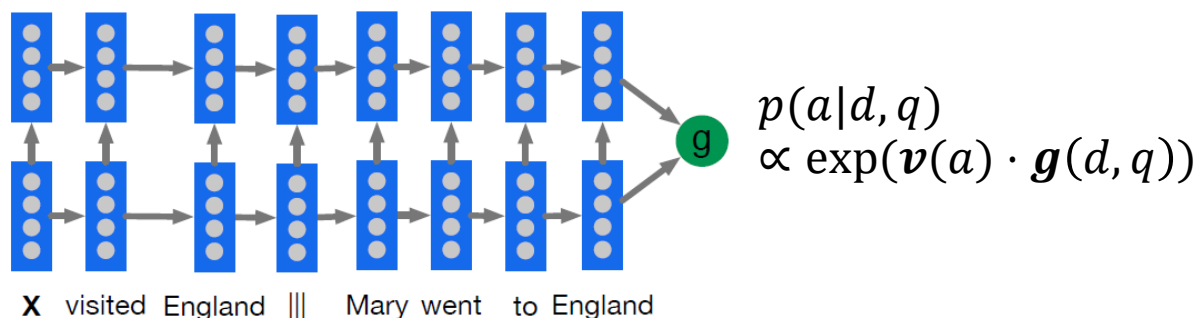
# DNNによる読解 (Hermann+ 15)



(a) Attentive Reader.

(クエリの単語毎にアテンションを張る)

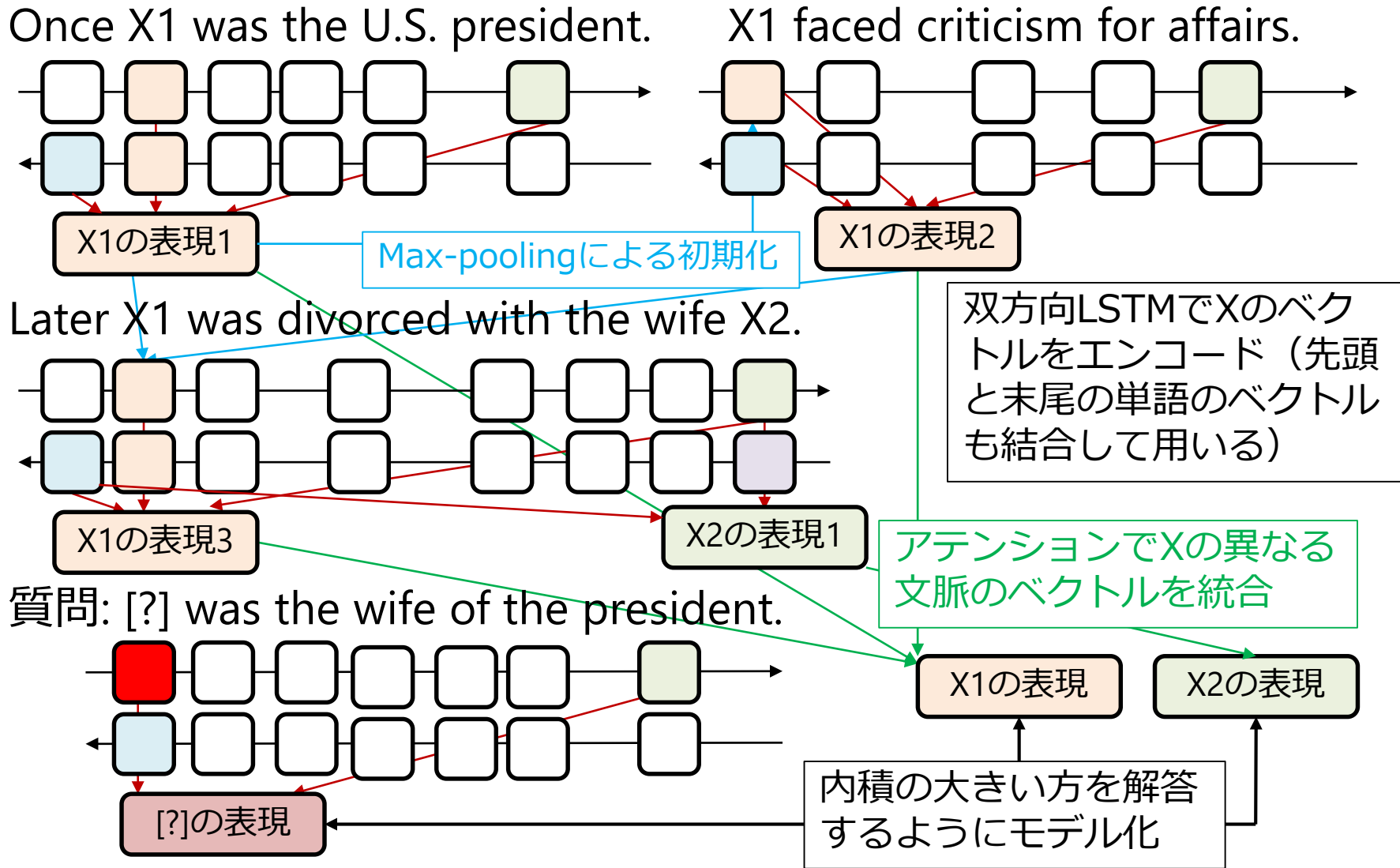
(b) Impatient Reader.



(c) A two layer Deep LSTM Reader with the question encoded before the document.

Hermann+ (2015)

# 動的分散表現による読解 (Kobayashi+ 16)





# CNN/Daily Mailデータの解析 (Chen+ 16)

- Attentive Reader (Hermann+ 15) のアテンションの取り方を変えるだけで性能が大きく向上 ■
- ノイジーなデータのため、**これ以上の性能向上の見込みは低い**
  - 共参照解析誤りと難解な事例が残りの25% ■
- DNNは換言や関連概念の認識が得意 ■
- 文脈の蓄積や処理が必要な事例が少ない ■

Model	CNN		Daily Mail	
	Dev	Test	Dev	Test
Frame-semantic model †	36.3	40.2	35.5	35.5
Word distance model †	50.5	50.9	56.4	55.5
Deep LSTM Reader †	55.0	57.0	63.3	62.2
Attentive Reader †	61.6	63.0	70.5	69.0
Impatient Reader †	61.8	63.8	69.0	68.0
MemNNs (window memory) ‡	58.0	60.6	N/A	N/A
MemNNs (window memory + self-sup.) ‡	63.4	66.8	N/A	N/A
MemNNs (ensemble) ‡	66.2*	69.4*	N/A	N/A
Ours: Classifier	67.1	67.9	69.1	68.3
Ours: Neural net	<b>72.4</b>	<b>72.4</b>	<b>76.9</b>	<b>75.8</b>

Table 2: Accuracy of all models on the *CNN* and *Daily Mail* datasets. Results marked † are from (Hermann et al., 2015) and results marked ‡ are from (Hill et al., 2016). *Classifier* and *Neural net* denote our entity-centric classifier and neural network systems respectively. The numbers marked with \* indicate that the results are from ensemble models.

No.	Category	(%)
1	Exact match	13
2	Paraphrasing	41
3	Partial clue	19
4	Multiple sentences	2
5	Coreference errors	8
6	Ambiguous / hard	17

Table 5: An estimate of the breakdown of the dataset into classes, based on the analysis of our sampled 100 examples from the *CNN* dataset.

Category	Classifier	Neural net
Exact match	13 (100.0%)	13 (100.0%)
Paraphrasing	32 (78.1%)	39 (95.1%)
Partial clue	14 (73.7%)	17 (89.5%)
Multiple sentences	1 (50.0%)	1 (50.0%)
Coreference errors	4 (50.0%)	3 (37.5%)
Ambiguous / hard	2 (11.8%)	1 (5.9%)
All	66 (66.0%)	74 (74.0%)

Table 6: The per-category performance of our two systems.

Chen+ (2016)

## bAbI (Weston+ 16)

- テキストの理解・推論の「単体テスト」を目指して構築されたデータ
  - 文章に関する質問とその答えの形式
  - 人間であれば外部知識無しで簡単に解ける問題
  - シミュレーションによりデータを自動生成
- 20種類のタスクに分けられている
- 限られた語彙 (150語, 4人, 6場所, 3物体)

**Task 1: Single Supporting Fact**

Mary went to the bathroom.  
 John moved to the hallway.  
 Mary travelled to the office.  
 Where is Mary? **A:office**

**Task 2: Two Supporting Facts**

John is in the playground.  
 John picked up the football.  
 Bob went to the kitchen.  
 Where is the football? **A:playground**

**Task 3: Three Supporting Facts**

John picked up the apple.  
 John went to the office.  
 John went to the kitchen.  
 John dropped the apple.  
 Where was the apple before the kitchen? **A:office**

**Task 4: Two Argument Relations**

The office is north of the bedroom.  
 The bedroom is north of the bathroom.  
 The kitchen is west of the garden.  
 What is north of the bedroom? **A: office**  
 What is the bedroom north of? **A: bathroom**

**Task 5: Three Argument Relations**

Mary gave the cake to Fred.  
 Fred gave the cake to Bill.  
 Jeff was given the milk by Bill.  
 Who gave the cake to Fred? **A: Mary**  
 Who did Fred give the cake to? **A: Bill**

**Task 6: Yes/No Questions**

John moved to the playground.  
 Daniel went to the bathroom.  
 John went back to the hallway.  
 Is John in the playground? **A:no**  
 Is Daniel in the bathroom? **A:yes**

**Task 7: Counting**

Daniel picked up the football.  
 Daniel dropped the football.  
 Daniel got the milk.  
 Daniel took the apple.  
 How many objects is Daniel holding? **A: two**

**Task 8: Lists/Sets**

Daniel picks up the football.  
 Daniel drops the newspaper.  
 Daniel picks up the milk.  
 John took the apple.  
 What is Daniel holding? **milk, football**

**Task 9: Simple Negation**

Sandra travelled to the office.  
 Fred is no longer in the office.  
 Is Fred in the office? **A:no**  
 Is Sandra in the office? **A:yes**

**Task 10: Indefinite Knowledge**

John is either in the classroom or the playground.  
 Sandra is in the garden.  
 Is John in the classroom? **A:maybe**  
 Is John in the office? **A:no**

**Task 11: Basic Coreference**

Daniel was in the kitchen.  
 Then he went to the studio.  
 Sandra was in the office.  
 Where is Daniel? A:studio

**Task 12: Conjunction**

Mary and Jeff went to the kitchen.  
 Then Jeff went to the park.  
 Where is Mary? A: kitchen  
 Where is Jeff? A: park

**Task 13: Compound Coreference**

Daniel and Sandra journeyed to the office.  
 Then they went to the garden.  
 Sandra and John travelled to the kitchen.  
 After that they moved to the hallway.  
 Where is Daniel? A: garden

**Task 14: Time Reasoning**

In the afternoon Julie went to the park.  
 Yesterday Julie was at school.  
 Julie went to the cinema this evening.  
 Where did Julie go after the park? A:cinema  
 Where was Julie before the park? A:school

**Task 15: Basic Deduction**

Sheep are afraid of wolves.  
 Cats are afraid of dogs.  
 Mice are afraid of cats.  
 Gertrude is a sheep.  
 What is Gertrude afraid of? A:wolves

**Task 16: Basic Induction**

Lily is a swan.  
 Lily is white.  
 Bernhard is green.  
 Greg is a swan.  
 What color is Greg? A:white

**Task 17: Positional Reasoning**

The triangle is to the right of the blue square.  
 The red square is on top of the blue square.  
 The red sphere is to the right of the blue square.  
 Is the red sphere to the right of the blue square? A:yes  
 Is the red square to the left of the triangle? A:yes

**Task 18: Size Reasoning**

The football fits in the suitcase.  
 The suitcase fits in the cupboard.  
 The box is smaller than the football.  
 Will the box fit in the suitcase? A:yes  
 Will the cupboard fit in the box? A:no

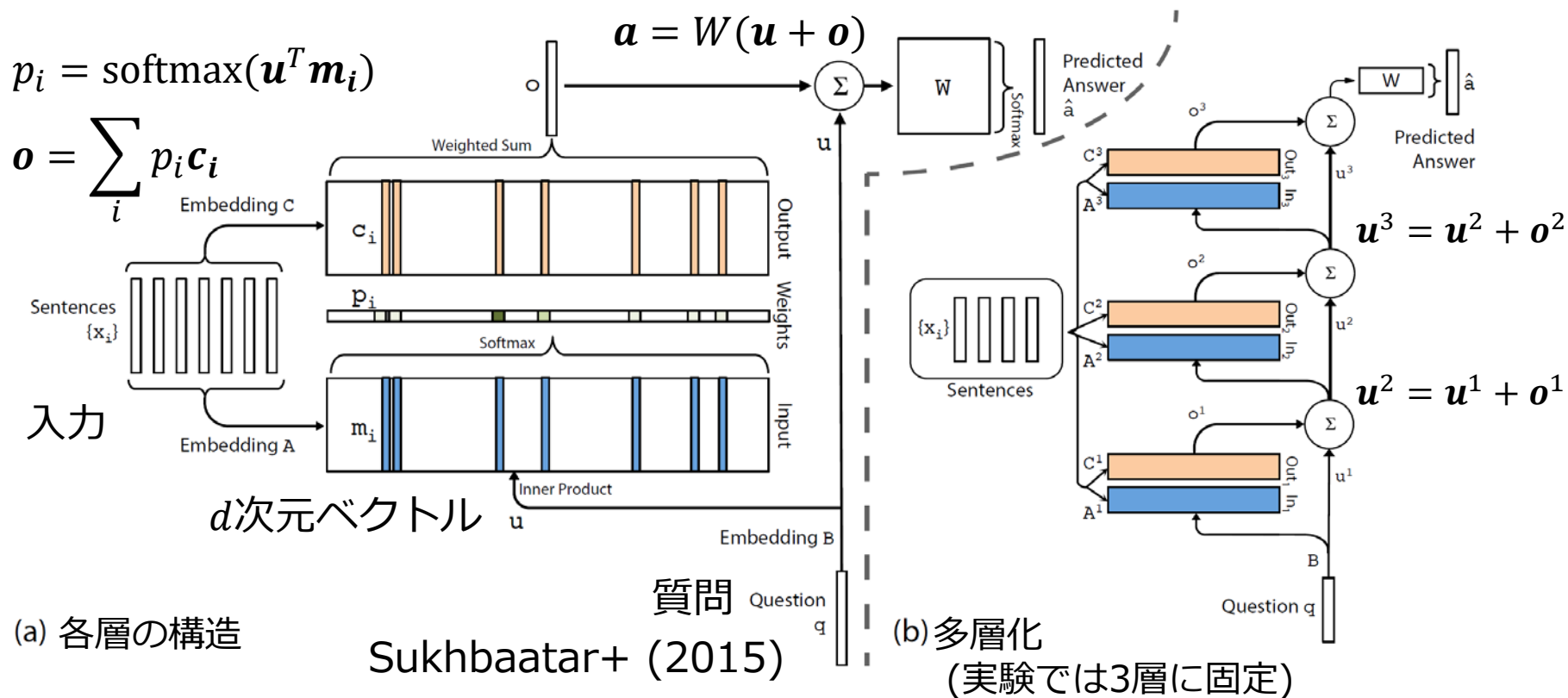
**Task 19: Path Finding**

The kitchen is north of the hallway.  
 The bathroom is west of the bedroom.  
 The den is east of the hallway.  
 The office is south of the bedroom.  
 How do you go from den to kitchen? A: west, north  
 How do you go from office to bathroom? A: north, west

**Task 20: Agent's Motivations**

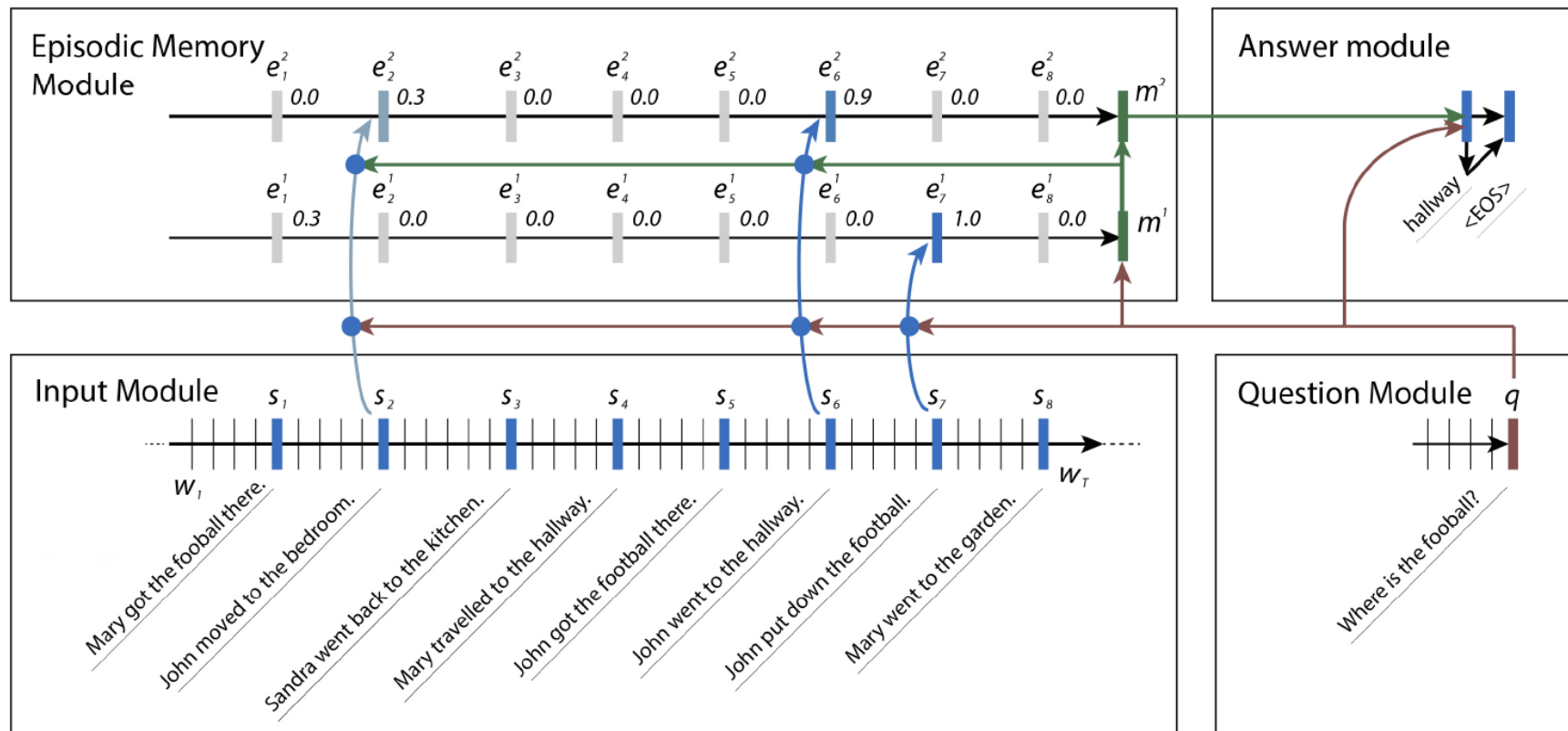
John is hungry.  
 John goes to the kitchen.  
 John grabbed the apple there.  
 Daniel is hungry.  
 Where does Daniel go? A:kitchen  
 Why did John go to the kitchen? A:hungry

# End-to-end Memory Network (Sukhbaatar+ 15)



- Memory Neural Networks (Weston+ 15) の改良版
  - ソフトアテンションによるメモリ読み込み
  - 質問と答えのペアだけで学習が可能

# Dynamic Memory Network (Kumar+ 16)



Kumar+ (2016)

- bAbIの正解率: 93.6% (Memory Networksよりも高い)
- 品詞タグ付けや評判分析でも高性能 ("Ask me anything")

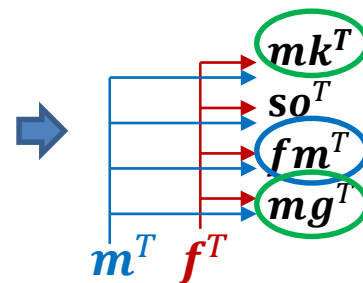
# bAbiデータの解析 (Lee+ 16)

- Tensor Product Representation (TPR) (Smolensky 90) でbAbiタスクを解くと、**ほぼ100%の正解率**

Mary went to the kitchen.  
Sandra journeyed to the office.  
Mary got the football there.  
Mary travelled to the garden.  
Where is the football? (garden)



belong(Mary, kitchen)  
belong(Sandra, office)  
belong(Football, Mary)  
belong(Mary, garden)



TPRによるTask 3の解答例. 文の意味解析結果から所属関係を取り出し, それをベクトルの外積で表現する. 各文の表現に対し, クエリ $f^T$ を左からかけて, 内積が1に最も近い直近の所持者を特定する ( $m^T$ ). Maryは人物なので,  $m^T$ をクエリとして, 同様の処理で直近の所持者 $g^T$ を得る

- bAbiだけでNNの推論能力を測定するのは不適切**
  - エンティティ間の関係が1種類のタスクが多いため
    - Task 17: 東・西・南・北の4種類
    - Task 19: 上・下・左・右の4種類
    - それ以外: 所属関係か所属の推移関係のみで構成されている

MemNNやDMNが苦手  
としているタスク

# まとめ



# まとめ

- 言語処理における深層学習のトレンド
  - 2013年-: 単語の分散表現
  - 2011年-: 句や文などの分散表現の合成
  - 2014年-: 符号化・復号化モデル
  - 2015年-: アテンション, 機械翻訳
  - 2016年-: 質問応答, ストーリーのモデリング
- 深層学習の強み
  - DNNの設計に関する知見が蓄積され, 様々なタスクにおいてend-to-endの学習が成功を収めた
  - 言語処理や画像処理といった分野の垣根を撤廃
  - ソフトウェア基盤が整備され, 研究・実験のサイクルが加速された

# 今後の課題

- 深層学習で起こっているメカニズムの解明
  - タスクを解くことと、単語や文の意味の取り扱いの関連が、はっきりとは分かっていない
- 文脈や情景の理解
  - 深層学習は入力と出力との関係が明確なタスクを得意としている
  - 複数の文や複数の物体を統合した高度な文脈・情景理解はまだまだ難しい
  - 人間の持っている常識的な知識を獲得・活用し、推論を行うメカニズムが必要

# 参考文献 (1/3)

- Bahdanau, D., Cho, K., and Bengio, Y.: Neural machine translation by jointly learning to align and translate, in Proc. of ICLR (2015)
- Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C.: A neural probabilistic language model, Journal of Machine Learning Research, Vol. 3, pp. 1137–1155 (2003)
- Chen, D., Bolton, J., Manning, C. D.: A thorough examination of the CNN/Daily Mail reading comprehension task, in Proc. of ACL, pp. 2358-2367 (2016)
- Cho, K., Merriënboer, van B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation, in Proc. of EMNLP, pp. 1724–1734 (2014)
- Collobert, R. and Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning, in Proc of ICML, pp. 160–167 (2008)
- Graves, A.: Generating sequences with recurrent neural networks, CoRR, Vol. abs/1308.0850, (2013)
- Hermann, K. M., Kočický, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P.: Teaching machines to read and comprehend, in Proc. of NIPS, pp. 1684-1692 (2015)
- Hinton, G., McClelland, J., and Rumelhart, D.: Distributed representations, in Rumelhart, D. E., McClelland, J. L., and Group, P. R. eds., Parallel distributed processing: Explorations in the microstructure of cognition, Vol. I, chapter 3, pp. 77–109, MIT Press, Cambridge, MA (1986)
- Kim, Y.: Convolutional neural networks for sentence classification, in Proc. of EMNLP, pp. 1746–1751 (2014)
- Kobayashi, K., Tian, R., Okazaki, N., Inui, K.: Dynamic entity representation with max-pooling improves machine reading, in Proc. of NAACL (2016)
- Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., Socher, R.: Ask me anything: Dynamic memory networks for natural language processing, in Proc. of ICML (2016)
- Laokulrat, N., Phan, S., Nishida, N., Shu, R., Ehara, Y., Okazaki, N., Miyao, Y., Satoh, S., Nakayama, H.: Generating video description using sequence-to-sequence model with temporal attention, in Proc. of Coling, pp. 44-52 (2016)
- Lee, M., He, X., Yih, W.-T., Gao, J., Deng, L., Smolensky, P.: Reasoning in vector space: An exploratory study of question answering, in Proc. of ICLR (2016)

# 参考文献 (2/3)

- Luong, M.-T., Pham, H., Manning, C. D.: Effective approaches to attention-based neural machine translation, in Proc. of EMNLP, pp. 1412-1421 (2015)
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J.: Distributed representations of words and phrases and their compositionality, in Proc. of NIPS, pp. 3111–3119 (2013)
- Mitchell, J. and Lapata, M.: Composition in distributional models of semantics. *Cognitive Science*, 34:1388–1429 (2010)
- Pennington, J., Socher, R., and Manning, C.: Glove: Global vectors for word representation, in Proc. of EMNLP, pp. 1532–1543 (2014)
- Rush, A. M., Chopra, S., Weston, J.: A neural attention model for sentence summarization in Proc. of EMNLP, pp. 379–389 (2015)
- Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks, in Proc. of NIPS (2015)
- Sutskever, I., Martens, J., and Hinton, G.: Generating text with recurrent neural networks, in Proc. of ICML, pp. 1017–1024 (2011)
- Sutskever, I., Vinyals, O., and Le, Q. V.: Sequence to sequence learning with neural networks, in Proc. of NIPS, pp. 3104–3112 (2014)
- Socher, R., Pennington, J., Huang, E., Ng, A., and Manning, C.: Semi-supervised recursive autoencoders for predicting sentiment distributions, in Proc. of EMNLP, pp. 151-161 (2011)
- Socher, R., Huval, B., Manning, C. and Ng, A.: Semantic compositionality through recursive matrix-vector spaces, in Proc. of EMNLP, pp. 1201-1211 (2012)
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., and Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank, in Proc. of EMNLP, pp. 1631-1642 (2013)
- Tai, K. S., Socher, R., and Manning, C. D.: Improved semantic representations from tree-structured long short-term memory networks, in Proc. of ACL-IJCNLP, pp. 1556–1566 (2015)
- Takase, S., Okazaki, N., Inui, K.: Composing distributed representations of relational patterns, in Proc. of ACL, pp. 2276-2286 (2016)
- Takase, S., Suzuki, J., Okazaki, N., Hirao, T., Nagata, M.: Neural headline generation on abstractive meaning representation, in Proc. of EMNLP (2016)

# 参考文献 (3/3)

- Tian, R., Okazaki, N., Inui, K.: The mechanism of additive composition, Machine Learning Journal, to appear, (2017)
- Vinyals, O., Le, Q. V., A neural conversational model, in Proc. of ICML Deep Learning Workshop, (2015)
- Vinyals, O., Toshev, A., Bengio, S., and Erhan, D.: Show and tell: A neural image caption generator, in Proc. of CVPR (2015)
- Weston, J., Bordes, A., Chopra, S., Rush, A. M., Merriënboer, B. van, Joulin, A., Mikolov, T.: Towards AI-complete question answering: A Set of Prerequisite Toy Tasks, in Proc. of ICLR (2016)
- Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's neural machine translation system: Bridging the gap between human and machine translation, CoRR abs/1609.08144 (2016)