特　集「ニューラルネットワーク研究のフロンティア」

ニューラルネットワークによる構造学習の発展

Advances in Structured Learning by Neural Networks

渡辺　太郎　　グーグル株式会社
Taro Watanabe　　Google Inc.
tarow@google.com

Keywords: natural language processing, machine translation, parsing, neural network, deep learning, structured learning.

1. は　じ　め　に

自然言語で書かれたテキストを処理対象とする自然言語処理の各分野において，ニュラールネットワークにより大幅な性能向上が行われている．例えば，英語の依存構造構文解析において，既存の素性関数をニューラルネットワークに置き換えるだけで，高速かつ従来法を上回る精度を達成している [Chen 14, Weiss 15]．また，機械翻訳において，BBN のシステムが従来法と比較しトップの成績をもつことが報告されている [Devlin 14]．品詞タグ付け，構文解析，意味解析，機械翻訳などの自然言語処理において，古典的に，専門家によるルールや辞書の整備によって複雑なシステムを実現してきたが，メンテナンスのコストや多言語化などの問題があり，容易ではなかった．1990 年代後半から計算機の大幅な性能向上および言語資源が整備されてきたことから，統計的なモデルによるシステムが構築されてきた．ところが，素性関数の開発自体が自明ではなく，ニューラルネットワークを用いた深層学習により素性表現自体を自動化できる，という期待が高まっている [Bengio 13]．

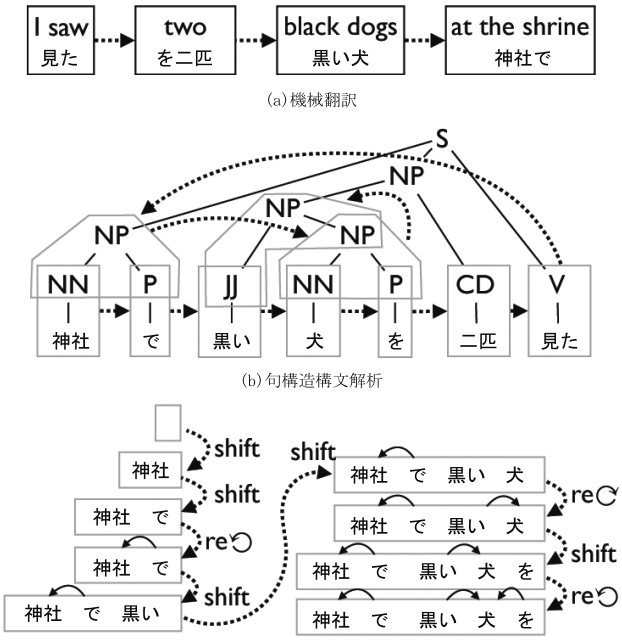
本稿では，特に，構文解析および機械翻訳に着目し，ニューラルネットワークがどのように応用されてきたのかを紹介，また，今後の発展について議論する．この二つの分野は木構造やフレーズペアなどある「構造」を出力とするシステムであり，探索の問題と切って話すことができない．まず，共通する統計モデルを導入し，従来法で幅広く用いられている線形モデルについて解説する（2 章）．その後，3 章にて，自然言語処理以外でも幅広く用いられている非常に簡単なニューラルネットワークのモデルを説明し，それらが構文解析および機械翻訳へとどのように応用されてきたかを紹介する．自然言語の特徴として，自然言語の文は入力長が可変であり，かつ木構造などさまざまな構造を必要とする．このような動的な入力かつ長い履歴を直接反映しつつ，自然言語処理の各アルゴリズムへ直接適用可能なモデルを紹介する（4 章）．以上のモデルは，構文解析や機械翻訳で用いられるアルゴリズムを下敷きとして適用，あるいは開発されたモデルであるが，ニューラルネットワークの隠れ層をそのまま用い，モデルから直接出力を生成する，エンコーダ・デコーダモデルを 5 章で紹介する．

2. 自然言語処理における統計モデル

自然言語処理のタスクは，I 単語から構成される自然言語の入力文 $\boldsymbol{x} = x_1^I = x_1 x_2 \cdots x_I$，$x_i \in \mathcal{X}$ が与えられたときに，ある出力 $\boldsymbol{y}$ を返すシステムとして捉えることができる．例えば，機械翻訳の場合，$\boldsymbol{y}$ は翻訳された目的言語の文であり，構文解析の場合，木構造で表された構文解析結果となる．統計的手法に基づく自然言語処理では，ある一文 $\boldsymbol{x}$ が与えられたとき，$\boldsymbol{y}$ の集合から最も出現確率の高い $\hat{\boldsymbol{y}}$ を探索することで誤りの少ない出力を得る問題と考えられる．

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} Pr(\boldsymbol{y}|\boldsymbol{x}) \tag{1}$$

ここで，$\boldsymbol{y}$ は J ステップから構成される出力 $\boldsymbol{y} = y_1^J = y_1 y_2 \cdots y_J$, $y_j \in \mathcal{Y}$ へと分解可能とする．図 1（a）は日本語の入力文「神社で黒い犬を二匹見た」を英語へと翻訳した例を示している．この例では，7 単語の日本語の入力に対し，“I saw”，“two”，“black dogs”，“at the shrine”といった 4 個のフレーズペアを目的言語の順序で生成している [Koehn 03]．句構造構文解析の場合，図 1（b）のように，“NN →神社”や“NP → NN P”など，複数の終点をもつ 12 個のハイパーエッジから構成されたハイパーグラフで表現される [Klein 04]．CYK などの動的計画法に基づくアルゴリズムの場合，ボトムアップで，左から右へと各ハイパーエッジが生成される．図 1（c）は遷移に基づくアルゴリズムを用いた依存構造構文解析を示している．スタックとキューで構成された探索空間で，例えば，shift アクションによって「神社」や「で」などの各単語がスタックへ追加されている．また，「神社↶で」や「黒い↷犬」のように，スタックのトップの



(a) 機械翻訳

(b) 句構造構文解析

(c) 依存構造構文解析

図 1　自然言語の各タスク

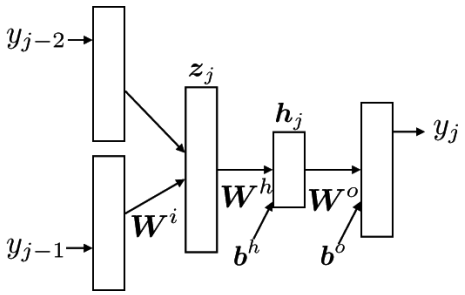


図 2　ニューラルネットワークによる素性表現

要素に対し，re◯や re◯アクションで，左および右へとそれぞれ依存構造を追加する [Nivre 08].

$\boldsymbol{y}$ はある一定の順序で生成されるとすると，式 (1) は各ステップ y_i を順に予測する問題として考えることができる.

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}\in\mathcal{Y}^*} \prod_{j=1}^{J} Pr(y_j|y_1^{j-1}, \boldsymbol{x}) \tag{2}$$

自然言語処理では，入力文はシンボルの系列であり，非常に疎であると同時に可変長であるため統計モデル $Pr(y_j \mid y_1^{j-1}, \boldsymbol{x})$ の設計は自明でない．そこで，経験的にさまざまな素性関数を導入することでさまざまな言語現象を捉えてきた.

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}\in\mathcal{Y}^*} \prod_{j=1}^{J} \frac{\exp(\boldsymbol{w}^{o\top}\boldsymbol{\phi}(y_j, y_1^{j-1}, \boldsymbol{x}))}{\sum_{y'\in\mathcal{Y}} \exp(\boldsymbol{w}^{o\top}\boldsymbol{\phi}(y', y_1^{j-1}, \boldsymbol{x}))} \tag{3}$$

$\boldsymbol{\phi}(\cdot)$ は q 次元の素性関数であり ($\mathcal{Y}\times\mathcal{Y}^{j-1}\times\mathcal{X}^*\to\mathbb{R}^q$)，各素性はベクトル $w^o\in\mathbb{R}^q$ により重み付けられる．重みベクトル w^o はあらかじめ正解ラベルが付与されたデータに対して交差エントロピーや誤差最小化などの目的関数に関して最適化することでパラメータを学習する.

従来法では，$\boldsymbol{y}$ の構成手法および探索手法 (argmax) と同時に，素性関数 $\boldsymbol{\phi}(\cdot)$ の設計および重みベクトル w^o の学習手法の研究開発が行われてきた．例えば，機械翻訳ではフレーズ単位の条件付き確率やフレーズペアの連接確率など数個の素性関数を大規模な対訳データから推定し，翻訳誤りを評価する尺度を直接最適化するように重みベクトルを決定している [Och 03]．句構造構文解析では統語ラベルの細分化を行い，生成確率の推定を行っている [Petrov 06]．遷移に基づく構文解析では各アクションはスタックやキューの状態に基づき，素性テンプレートで生成された二値素性を用いている [Nivre 08]．ある程度自然言語に精通した専門家であれば経験的にどのような素性が性能の向上へと結びつくのかを理解できるが，自明でないような素性も数多くあり，さらに，探索効率とのトレードオフを考慮する必要もある．結局，タスクおよび重みベクトルの学習法に応じて試行錯誤を繰り返す必要があった.

3. ニューラルネットワークの応用

ニューラルネットワークに基づく深層学習は素性関数自体をデータから自動的に学習する，という点で自然言語処理の性能向上に貢献した．具体的には，$\boldsymbol{\phi}(\cdot)$ の各次元の素性はあらかじめ開発者が決定する必要があったが，ネットワークの構造を選択するだけでデータから自動的に学習する．例えば，以下のような非常に簡単なネットワークを想定する.

$$p(y_j|y_1^{j-1}, \boldsymbol{x}) \approx p(y_j|y_{j-2}^{j-1}) \tag{4}$$

$$= \frac{\exp(\boldsymbol{u}^{y_j\top}(\boldsymbol{W}^o\boldsymbol{h}_j + \boldsymbol{b}^o))}{\sum_{y'\in\mathcal{Y}} \exp(\boldsymbol{u}^{y'\top}(\boldsymbol{W}^o\boldsymbol{h}_j + \boldsymbol{b}^o))} \tag{5}$$

$$\boldsymbol{h}_j = f(\boldsymbol{W}^h\boldsymbol{z}_j + \boldsymbol{b}^h) \tag{6}$$

$$\boldsymbol{z}_j = [\boldsymbol{W}^i\boldsymbol{u}^{y_{j-2}}; \boldsymbol{W}^i\boldsymbol{u}^{y_{j-1}}] \tag{7}$$

$W^i\in\mathbb{R}^{q\times|\mathcal{Y}|}$ は，出力シンボル $\boldsymbol{y}$ を対応する q 次元のベクトル表現 (embedding) へと変換するマトリックス，$\boldsymbol{u}^y\in\{0, 1\}^{|\mathcal{Y}|}$ はシンボル $\boldsymbol{y}$ に対応する位置が 1 となる単位ベクトルとし，“;” は $[a\,;\,b]$ のようにベクトルおよびマトリックスの結合として用いる．$z_j\in\mathbb{R}^{2q}$ を入力としてマトリックス $\boldsymbol{W}^h\in\mathbb{R}^{q\times 2q}$ およびバイアス項 $\boldsymbol{b}^h\in\mathbb{R}^q$ から新たに q 次元のベクトル表現を得て，例えば tanh や sigmoid などによる活性化関数 $f(\cdot)$ により非線形な変換を行う．$\boldsymbol{h}_j\in\mathbb{R}^q$ は，$W^o\in\mathbb{R}^{|\mathcal{Y}|\times q}$ で重み付けされた後，バイアス項 $\boldsymbol{b}^o\in\mathbb{R}^{|\mathcal{Y}|}$ を加え，softmax 関数により確率モデルとして変換される．このネットワークの構成は図 2 のように示され，$\boldsymbol{y}$ を単語としたとき，過去の履歴として出力された 2 単語を参照する n-gram 言語モデ

ルとして定式化される [Bengio 03]．あらかじめ決められた次元を入力とし，複数の層を通して最後の出力を得るネットワークであることから，フィードフォーワード型ニューラルネットワーク（Feed-Forward Neural Network：FFNN）と呼ばれる．各ベクトルは層を構成し $\boldsymbol{W}^o$ は出力に一番近いパラメータであることから出力層と呼ばれ，$\boldsymbol{z}_j$ は出力から離れているため一般に隠れ層と呼ばれる．

すべてのパラメータ $\boldsymbol{\theta} = [\boldsymbol{W}^o; \boldsymbol{b}^o; \boldsymbol{W}^h; \boldsymbol{b}^h; \boldsymbol{W}^i]$ は誤差伝播法 [Rumelhart 88] によりデータ $\mathcal{D}$ に対する交差エントロピーを最小化するように学習される*1．

$$\mathcal{L}_{\boldsymbol{\theta}} = \sum_{\boldsymbol{y} \in \mathcal{D}} \sum_{y_j \in \boldsymbol{y}, y' \in \mathcal{Y}} -\delta(y_j, y') \log p(y' | y_{j-2}^{j-1}) \quad (8)$$

従来法では，式（3）のように $\boldsymbol{\phi}(\cdot)$ は固定されており，線形結合した重みベクトル $\boldsymbol{w}^o$ を学習していた．式（5）の $\boldsymbol{W}^o$ だけでなく式（6）のすべてのパラメータを学習することで，素性表現自体を自動推定可能となった．また，例えば 3-gram 言語モデルと比較したとき，$O(|\mathcal{Y}|^3)$ のパラメータが必要であるのに対し，$O(|\mathcal{Y}|)$ へと大幅に削除可能である．ただし，従来法の式（3）のパラメータ学習は凸関数であるのに対し，非線形な最適化問題を解くことになるため，必ずしも最適解を得られるとは限らない．

このフィードフォワード型ネットワークは簡単な構造でありながらモデルとしての表現力は高く，また，隠れ層の数や次元数を工夫することで容易に性能向上を図ることができる．また，既存のシステムにおいて新たな素性関数として容易に追加が可能である [Liu 13]．単語数を制約することで大規模語彙の n-gram 言語モデルを置き換えたり [Schwenk 07]，ノイズ対照推定（Noise Contrastive Estimate：NCE）[Gutmann 12] により大規模データに対して言語モデルを学習することで機械翻訳の性能が向上したことが報告されている [Vaswani 13]．言語モデルの入力層として原言語側のコンテキストを入れることで大幅な翻訳の精度を上げつつ，式（5）の softmax の分母の項を省略したり [Andreas 15]，入力層に近い層において事前計算する，といった工夫で実時間での翻訳を可能とした [Devlin 14]．ほかにも畳込み型ニューラルネットワーク（convolutional neural network）で，原言語側の文全体を表現したり [Meng 15]，テンソル（tensor）により各次元の組合せを学習する [Setiawan 15]，などの工夫が試みられている．単語アライメントで FFNN による長いコンテキストを導入したり [Yang 13]，依存構造構文解析において FFNN による素性表現を用いることにより，既存の素性関数より大幅に性能を向上させつつ処理速度も向上している [Chen 14, Weiss 15].

*1 $\delta(a, b)$ は $a = b$ のとき 1 を返すクロネッカーデルタ関数とする．

4. 動的な構造に基づくニューラルネットワーク

フィードフォーワード型ネットワークは，従来の素性関数をそのまま置き換えることができるが，素性関数の研究開発が次元数を調整する問題へと置き換えられたと考えられる．例えば，言語モデルは，長い履歴を入力した場合，より正確に次の単語を予測可能になり，FFNN の場合，入力する単語の数を多くすれば対応可能である．理想的には，入力文全体を履歴とするネットワークが構築されれば，出力を予測する性能が大幅に高まるであろう．ところが自然言語の入力は可変長であり，長さに応じて入力の次元数が変わる．このため FFNN では入力次元数に応じたモデルを構築するなどの工夫が必要となる．例えば，FFNN に基づく言語モデルでは，最大の履歴数をあらかじめ決定し，より短い履歴のスコアを得る場合，特殊な記号 ⟨null⟩ を詰め込んで入力次元を一定にしている [Vaswani 13]．次元数をあらかじめ決定するのではなく，可変長な入力に対し，柔軟にネットワークを構築可能な枠組みが求められる．

4·1 回帰型ネットワーク

回帰型ニューラルネットワーク（recurrent neural network）は，このような可変長なコンテキストを隠れ層を用いてモデル化する．ある時間 j において，入力 $\boldsymbol{x}_1^j$ が得られたときの出力 $\boldsymbol{y}^j$ を予測するモデルは，次のように表される．

$$p(y_j | x_1^j) = g(\boldsymbol{u}^{y_j \top} (\boldsymbol{W}^o \boldsymbol{h}_j + \boldsymbol{b}^o)) \quad (9)$$

$$\boldsymbol{h}_j = f(\boldsymbol{W}^h [\boldsymbol{h}_{j-1}; \boldsymbol{W}^i \boldsymbol{u}^{x_j}] + \boldsymbol{b}^h) \quad (10)$$

ここで，$\boldsymbol{h}_j \in \mathbb{R}^q$ は隠れ層であり，一つ前の隠れ層 $\boldsymbol{h}^{j-1}$ および現在の入力ベクトル $\boldsymbol{W}^i \boldsymbol{u}^{x_j} \in \mathbb{R}^q$ を用いて計算される．また，$g(\cdot)$ は例えば softmax といった活性化関数とする．ある時間 j における予測は，過去のすべての入力 x_1^j と隠れ層 h_1^j に依存した長いコンテキストを考慮することができる．このネットワーク構造はエルマンネットワーク（Elman network）とも呼ばれ [Elman 90]，図3のように系列を直接表現したネットワークとなっている．[Mikolov 10] は，x_j を直前に出力された単語 y_{j-1} にすることで言語モデルへと応用し，長い履歴を考慮することでパープレキシティを小さくすることができ，音声認識において有効であることを報告している．また，

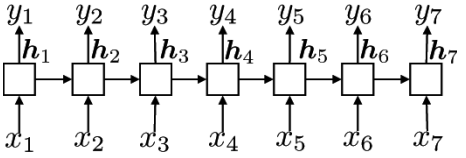


図3 回帰型ニューラルネットワーク

x_j だけでなく，原言語文のベクトル表現を追加することで，言語モデルだけではなく，同時翻訳モデルとして捉えることが可能である [Auli 13, Kalchbrenner 13]．また, [Sundermeyer 14] は，フレーズペア単位に同期させ，原言語と目的言語を同時に入力することで同時翻訳モデルを実現している．[Wu 14] は，予測するシンボル（y_j）をフレーズペアにすることで翻訳モデルとして応用，フレーズペアを最小単位に分割したり，単語の系列へと分割することで，非常に疎なシンボル系列になることを防いでいる．[Tamura 14] は，対訳文の単語アライメントへと応用，NCE による教師なし学習および単語表現に対する制約を加えることで，従来の生成モデルや FFNN [Yang 13] を超える精度を達成している．

FFNN と違い，長いコンテキストを考慮できるものの, 隠れ層 $\boldsymbol{h}_j$ をそのまま探索時の状態として用いるため，効率の良い探索は自明でない．[Auli 14] では，回帰型言語モデルを統計的機械翻訳のデコーダへと組み入れているが，既存のモデルであらかじめ枝刈りされたラティスを再計算することで近似している．また，各ノードですべての状態に対応する隠れ層を記憶するのではなく，スコアの高い隠れ層のみを記憶することで対処している．

回帰型ネットワークを $\boldsymbol{h}_j$ の層とみなすと，隠れ層の数は入力長に比例し，$\boldsymbol{h}_1$ など最初のほうの層は，最後の層に対しての影響が非常に小さくなる．また，誤差逆伝播によりパラメータを学習すると，勾配消失問題と呼ばれ，層を通るたびに誤差が非常に小さくなり，学習が困難になることが知られている．長短期メモリ（Long Short-Term Memory：LSTM）[Hochreiter 97] は，隠れ層と同様にセルと呼ばれる記憶領域を設け，ネットワークの接続にゲートを設定，ゲート自体もセルや隠れ層に応じて自動的に調整することでこれらの問題を回避している．また，セルを排除したより簡単なゲート回帰型ユニット（Gated Recurrent Unit：GRU）[Cho 14] でも同様な精度を達成できることが示された [Chung 14].

4·2 再帰型ネットワーク

自然言語処理では，構文解析などで木構造を用いることが多く，系列に基づく回帰型ネットワークでは直接表現できない．再帰型ニューラルネットワーク（recursive neural networks）は，回帰型ニューラルネットワークを木構造など，有向非巡回グラフ（Directed Acyclic Graph：DAG）へと拡張したものである．図 4 では，二分木のネットワークの例を示しており，ある親ノードを表現した隠れ層を $\boldsymbol{h}_p \in \mathbb{R}^q$ とすると，その左右の子ノードの隠れ層 $\boldsymbol{h}_l$ および $\boldsymbol{h}_r$ を元にして計算される．

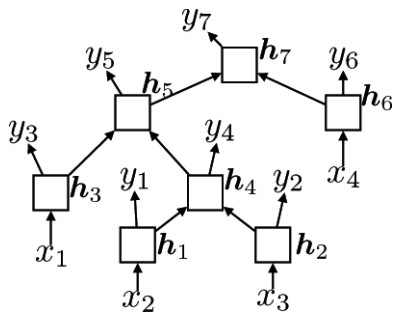


図 4 再帰型ニューラルネットワーク

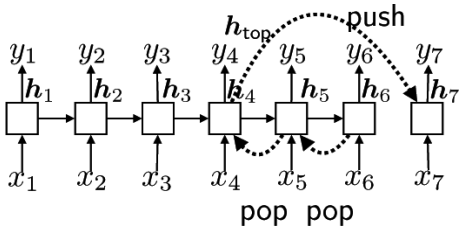


図 5 スタック型ニューラルネットワーク

$$p(y_p|x_{l_p}^{r_p}) = g(\boldsymbol{u}^{y_p\top}(\boldsymbol{W}^o\boldsymbol{h}_p + \boldsymbol{b}^o)) \tag{11}$$

$$\boldsymbol{h}_p = f(\boldsymbol{W}^h[\boldsymbol{h}_l;\boldsymbol{h}_r] + \boldsymbol{b}^h) \tag{12}$$

再帰的な構造により，$\boldsymbol{h}_p$ は，その部分木が被覆する入力ベクトル $\boldsymbol{x}_{l_p}^{r_p}$ を表現したものとなる．このため，回帰型ニューラルネットワークと同様，任意の長さの入力を表現するだけではなく，部分木により入力文の任意の区間の表現を求めることができる．[Socher 12] では，構文解析木を用いて，回帰型ニューラルネットワークを学習し，句単位に評判分析を行えることを報告している．さらに，回帰型ニューラルネットワークに対する LSTM と同様に，再帰型ニューラルネットワークの木構造において LSTM を導入することで，長い履歴を効率良く記憶することができ，性能の向上が報告されている [Tai 15]．[Socher 13] では，パラメータ $\boldsymbol{W}^h$ を子ノードのラベルで細分化することで，パラメータに対して明示的な文法を導入，PCFG の構文解析結果のリランキングで精度を向上できることを示した．[Stenetorp 13] は，依存構造構文解析において，木構造を直接反映した回帰型ニューラルネットワークを導入することで従来法と同等な性能を達成したことを報告している．

4·3 スタック型ニューラルネットワーク

スタック型ニューラルネットワークは回帰型ニューラルネットワークを拡張し，一つ前の隠れ層ではなく，任意の過去の隠れ層からの遷移を可能としたものである [Dyer 15, Watanabe 15]．具体的には，回帰型ニューラルネットワークが時間 j において次の予測 y_j をするとき，$\boldsymbol{h}_{j-1}$ に基づいて新たな隠れ層 $\boldsymbol{h}_j$ を push 操作により追加し（式（10）），式（9）でモデルのスコアが計算される．スタック型ニューラルネットワークでは，push 操作は同じように行われるが，現在のトップの要素を指すポインタ top を導入し，pop 操作にてポインタを

前の要素を指すように移動させる．図5はスタック型ニューラルネットワークの例を示しており，時間jの予測が，topが指している隠れ層$\boldsymbol{h}_{\text{top}}$に基づいて新たに隠れ層$\boldsymbol{h}_j$がpushによって追加され，topが$\boldsymbol{h}_j$を指すように更新される．似たような構造として[Das 92]は回帰的ニューラルネットワークにスタックによる記憶領域を設けているが，スタックのすべての要素を直接参照することができない．明示的なpushおよびpop操作に対し[Grefenstette 15]ではLSTMと同様なベクトル表現によりスタックを実現している．[Dyer 15]は再帰型ニューラルネットワークで求められた部分木のベクトル表現を入力として遷移型依存構造構文解析へと応用している．さらに，スタックとキューとの間で要素を入れ替えるswap操作により交差を許した依存構造構文解析を実現した[Ballesteros 15]．[Watanabe 15]も同様に遷移型句構造構文解析へ応用しているが，[Dyer 15]ではスタックやキューの隠れ層をそれぞれ独立に計算してから結合するのに対し，スタックの隠れ層の計算時に密な結合を行っている．[Le 14]はトップダウンで求められるネットワークを組み合わせた内側外側回帰的ネットワークを提唱，ルートノードからの依存構造をすべて反映したモデル化を可能とした．

5. エンコーダ・デコーダモデル

ニューラルネットワークは，従来の自然言語処理の統計モデルを置き換えるものとして登場してきた．例えば，機械翻訳や構文解析において素性関数の一つとしてFFNNが用いられたり，スタック型ネットワークのように既存の遷移に基づく構文解析アルゴリズムを変更することなくニューラルネットワークが適用されてきた．エンコーダ・デコーダモデルはこれらの手法とは異なる全く新しい考え方を用いており，エンコーダにより入力文を実数値ベクトルで表現し，デコーダで逐次可変長の出力シンボルを生成する[Bahdanau 15, Kalchbrenner 13, Sutskever 14]．機械翻訳へ応用する場合，入力が原言語文となり，出力が目的言語の文となる．エンコーダ・デコーダモデルを図6に示す．エンコーダは入力文$\boldsymbol{x}$に対して回帰型ネットワークあるいはLSTMにより内部表現を得る．

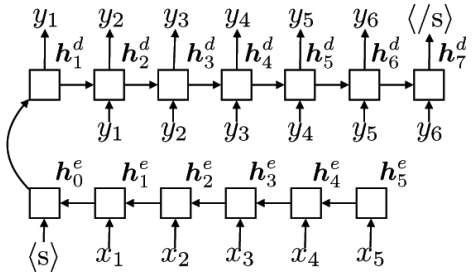


図6 エンコーダ・デコーダモデル

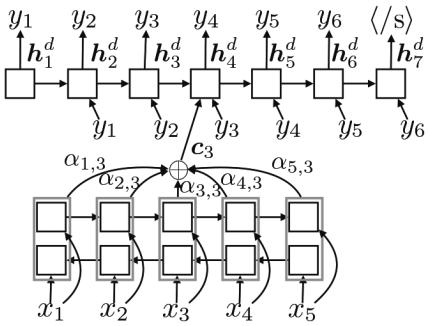


図7 注意モデル

$$\boldsymbol{h}_i^e = f(\boldsymbol{W}^e[\boldsymbol{h}_{i+1}^e; \boldsymbol{W}^{ie}\boldsymbol{u}^{x_i}] + \boldsymbol{b}^e) \tag{13}$$

ここで，エンコーダは遷移に基づく構文解析のキューのように，入力文を文末($\boldsymbol{x}_I$)から文頭($\boldsymbol{x}_1$)へと逆にエンコードし，最後に文頭を示す特殊な記号$\langle\text{s}\rangle$に対応したベクトル表現でエンコードする．デコーダは，最後に得られた隠れ表現$\boldsymbol{h}_0^e = \boldsymbol{h}_1^d$から順番に，モデルスコアを最大化するシンボルを生成し，文末を示す特殊な記号$\langle/\text{s}\rangle$を生成したとき，終了する．

$$\hat{y}_j = \arg\max_{y \in \mathcal{Y}} p(y|\hat{y}_1^{j-1}, \boldsymbol{x}) \tag{14}$$

$$p(y|y_1^{j-1}, \boldsymbol{x}) = g(\boldsymbol{u}^{y\top}(\boldsymbol{W}^o\boldsymbol{h}_j^d + \boldsymbol{b}^o)) \tag{15}$$

$$\boldsymbol{h}_j^d = f(\boldsymbol{W}^d[\boldsymbol{h}_{j-1}^d; \boldsymbol{W}^{id}\boldsymbol{u}^{y_{j-1}}] + \boldsymbol{b}^d) \tag{16}$$

エンコーダ・デコーダモデルの考え方は古くから構文解析へと応用されてきた．回帰型ネットワークにより入力文をエンコードしたあと，デコードときに[Miikkulainen 90, Vinyals 15]は回帰型ネットワーク，また[Berg 92]は再帰型ネットワークにより構文木を出力している．[Mayberry 99]は構文木の出力時に遷移型アルゴリズムを用いている．

5·1 注 意 モ デ ル

エンコーダ・デコーダモデルでは，最初のデコーダ時にはエンコードされた隠れ層を直接反映することができるが，デコードが進むにつれてその影響は小さくなるという欠点があった．[Bahdanau 15]は注意モデル（attention model）を導入，エンコードされたすべての隠れ層を重み付けて足すことでその問題を解決している．具体的には，入力文を両方向にエンコードした双方向再帰型ニューラルネットワークを用いて頑健性を増やす．

$$\overrightarrow{\boldsymbol{h}}_i = f(\overrightarrow{\boldsymbol{W}}^e[\overrightarrow{\boldsymbol{h}}_{i-1}; \boldsymbol{W}^{ie}\boldsymbol{u}^{x_i}] + \overrightarrow{\boldsymbol{b}}^e) \tag{17}$$

$$\overleftarrow{\boldsymbol{h}}_i = f(\overleftarrow{\boldsymbol{W}}^e[\overleftarrow{\boldsymbol{h}}_{i+1}; \boldsymbol{W}^{ie}\boldsymbol{u}^{x_i}] + \overleftarrow{\boldsymbol{b}}^e) \tag{18}$$

各隠れ層からの重み付けを計算し，それを元にして新たにコンテキストベクトル表現を得る．

$$\boldsymbol{c}_j = \sum_{i=1}^{I} \alpha_{i,j} [\overrightarrow{\boldsymbol{h}}_i; \overleftarrow{\boldsymbol{h}}_i] \quad (19)$$

デコーダの隠れ層 $\boldsymbol{h}_j^d$ は，y_{j-1} と $\boldsymbol{h}_{j-1}^d$に加えて $\boldsymbol{c}_j$ から求められ，重み付けパラメータ$\alpha_{i,j}$は，$[\overrightarrow{\boldsymbol{h}}_i; \overleftarrow{\boldsymbol{h}}_i]$ および $\boldsymbol{h}_{j-1}^d$ から計算される．$\alpha_{i,j}$は，j 番目の出力が i 番目の入力との対応付けに関する信頼度と解釈することができ，例えば入力が原言語文で対応する目的言語を生成する翻訳のタスクの場合，単語アライメントの度合いとして捉えられる．

5·2 大規模語彙化

エンコーダ・デコーダモデルで実現されたニューラル翻訳モデルは，既存の機械翻訳と異なり，言語モデルやフレーズテーブルなどを保持する必要がなく，すべてパラメータとして表現される．原言語のすべての単語はメモリが許す限り大規模な語彙を使用することができるが，式（14）のように，デコード時には $\mathcal{Y}$のすべての目的言語の単語を列挙してスコアを最大化する単語を選択し，生成する必要がある．また，パラメータの学習を容易にするため，$g(\cdot)$ の活性化関数として softmax 関数が用いられることから，$\mathcal{Y}$のすべての単語について総和を取る必要があり，大規模な語彙へと対応することが非常に困難であった．このため，頻度の高い語彙集合に絞り，それ以外の単語をすべて UNK などのシンボルへと置き換えることで対処している．

[Luong 15] では，UNK として出力されたシンボルの原言語側の対応付けをヒューリスティックに求め，あらかじめ用意した単語単位の対訳辞書を用いて UNK を単語へと変換している．ここで，注意モデルを用いた場合，あらかじめ単語の対応付けの信頼度が求まるため，簡単に対応付けが計算される．これに対し [Jean 15] では，学習データをサブデータへと分割するが，このとき各サブデータの語彙の異なり数があらかじめ決められた範囲を超えないようにする．サブデータ内では語彙集合が限られるため，容易に学習可能としつつ，結合したモデル全体ではすべての語彙を学習可能となった．さらに，デコード時には，注意モデルの単語アライメントの信頼度を利用し，すべての目的言語の単語を列挙するのではなく，信頼度の高い原言語の単語と共起する，頻度の高い目的言語に制限した．これらの工夫により，既存の句に基づく機械翻訳と同等あるいはそれを超える性能を達成している．

6. 今後の展望

深層学習の自然言語処理分野への応用は，フィードフォーワード型ネットワークを既存のシステムへと適用することから始まった．フィードフォーワード型ネットワークはコンテキストが限られることから，素性の一部として容易に組み入れることができ，かつ従来法で必要不可欠とされる素性の開発，選択の問題から一部解放された．回帰型および再帰型ネットワークによる無限の履歴を考慮したモデルの組み込みは始まったばかりであり，例えば，[Auli 14] のような近似手法，あるいはニューラルネットワークの構造に適したデコードや探索手法の研究開発が一層進むと思われる．

全く新しい，エンコーダ・デコーダの枠組みは，探索空間を単純な線形な空間に制限しつつ，ベクトル表現がどのような表現をもっているのか，その可能性に挑戦している．初期の研究では，エンコーダおよびデコーダともに回帰型ネットワークを直接用いているが，注意モデルによりエンコード時の位置などの構造をデコーダへと反映させる取組みが始まっている．また，スタック型ネットワークは，既存の遷移型構文解析器の動きを直接ネットワークの構造へと反映したものである．今後，エンコーダ・デコーダの枠組みにおいて，言語処理の各タスクに必要な構造を取り入れる，といった研究と同時に，既存の探索アルゴリズムの構造を直接反映したネットワークの構造の研究開発が進められるであろう．

自然言語処理のタスクでは，例えば，品詞タグ付けなど，過去の入力および出力から次のラベルを予測するようなタスクが多く，モデルのパラメータ学習はあらかじめラベルが付与されたデータに対して行われる．ところが，例えば，対訳データに対する単語アライメントの付与など，あらかじめラベルが付与されたデータが存在しないようなタスクが多く，このとき，教師なし学習により自動推定が行われる．ニューラルネットワークを構造の自動推定に用いる研究はあまりない．[Tamura 14] は NCE により単語アライメントを自動推定して，機械翻訳にて推定された単語アライメントの有効性を示している．[Socher 11] は再帰型自己符号化器（recursive autoencoder）[Pollack 90] を用いて，自動的に木構造を推定する手法を提案している．その成果が機械翻訳の並び換えモデル [Li 13, Li 14] やフレーズペアのベクトル表現 [Liu 14, Su 15, Zhang 14] へと利用されているが，自動的に推定された木構造自体を直接利用した研究はない．

構文解析や機械翻訳は，木構造や句単位のアライメントなど構造を出力するタスクと考えられ，そのパラメータを学習する問題は構造学習と呼ばれた．構文解析木や翻訳などの正解ラベルがあったとしても，探索アルゴリズムの制約やヒューリスティックな枝刈りによる探索エラーのために，たとえ正しいモデルパラメータが学習されたとしても正解が得られないことがある．従来法では，実際にデコードしてその誤りを元にパラメータを更新する，といった手法が用いられる [Collins 04]．ところが，ニューラルネットワークでは非線形なモデルのため学習が非常に難しく，例えば，隠れ層までのパラメータを事前に学習し，最後の表層のパラメータを平均化パー

セプトロンで学習している [Weiss 15]．[Watanabe 15] は k-best の出力のうち，誤った出力を重み付けでペナルティを与えることですべてのパラメータを同時に安定して学習できることを示している．

今後，非線形なモデルの構造を利用して，データから複雑な構造を自動的に推定し，かつ，大規模化する探索空間であってもパラメータを学習する研究が一層発展するであろうと思われる．

謝 辞

本稿を完成するにあたり，中川哲治氏から貴重なコメントをいただきました．ここに感謝の意を表します．

◇ 参 考 文 献 ◇

[Andreas 15] Andreas, J. and Klein, D.: When and why are log-linear models self-normalizing?, *NAACL-HLT2015,* pp. 244-249, Denver, Colorado（2015）

[Auli 13] Auli, M., Galley, M., Quirk, C. and Zweig, G.: Joint language and translation modeling with recurrent neural networks, *EMNLP 2013,* pp. 1044-1054, Seattle, Washington, USA（2013）

[Auli 14] Auli, M. and Gao, J.: Decoder Integration and Expected BLEU training for recurrent neural network language models, *ACL 2014,* pp. 136-142, Baltimore, Maryland（2014）

[Bahdanau 15] Bahdanau, D., Cho, K. and Bengio,Y.: Neural machine translation by jointly learning to align and translate, *ICLR 2015*（2015）

[Ballesteros 15] Ballesteros, M., Dyer, C. and Smith, N. A.: Improved transition-based parsing by modeling characters instead of words with LSTMs, *EMNLP 2015,* pp. 349-359, Lisbon, Portugal（2015）

[Bengio 03] Bengio, Y., Ducharme, R.,Vincent, P. and Janvin, C.: A neural probabilistic language model, *J. Machine Learning Research,* Vol. 3, pp. 1137-1155（2003）

[Bengio 13] Bengio,Y., Courville, A. and Vincent, P.: Representation learning: A review and new perspectives, *IEEE Trans. on Pattern Analysis and Machine Intelligence,* Vol. 35, No. 8, pp. 1798-1828（2013）

[Berg 92] Berg, G.: A connectionist parser with recursive sentence structure and lexical disambiguation, *AAAI'92,* pp. 32-37（1992）

[Chen14] Chen, D. and Manning, C.: A fast and accurate dependency parser using neural networks, *EMNLP 2014*, pp. 740-750, Doha, Qatar（2014）

[Cho 14] Cho, K., Merrienboer, van B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y.: Learning phrase representations using RNN encoder, decoder for statistical machine translation, *EMNLP 2014,* pp. 1724-1734, Doha, Qatar（2014）

[Chung 14] Chung, J., Gülçehre, Ç., Cho, K. and Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling, *CoRR,* Vol. abs/1412.3555（2014）

[Collins04] Collins, M. and Roark, B.: Incremental parsing with the perceptron algorithm, *ACL 2004,* pp. 111-118, Barcelona, Spain（2004）

[Das 92] Das, S., Giles, C. L. and Sun, Zheng, G.: Learning context-free grammars: Capabilities and limitations of a recurrent neural network with an external stack memory, *Conf. of the Cognitive Science Society,* pp. 791-795（1992）

[Devlin14] Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R. and Makhoul, J.: Fast and robust neural network joint models for statistical machine translation, *ACL 2014,* pp. 1370-1380, Baltimore, Maryland（2014）

[Dyer 15] Dyer, C., Ballesteros, M., Ling, W., Matthews, A. and Smith, N. A.: Transition-based dependency parsing with stacklong short-term memory, *ACL 2015,* pp. 334-343, Beijing, China（2015）

[Elman 90] Elman, J. L.: Finding structure in time, *Cognitive Science,* Vol. 14, No. 2, pp. 179-211（1990）

[Grefenstette 15] Grefenstette, E., Hermann, K. M., Suleyman, M. and Blunsom, P.: Learning to transduce with unbounded memory, *CoRR,* Vol. abs/1506.02516（2015）

[Gutmann12] Gutmann, M. U. and Hyvärinen, A.: Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics, *J. Machine Learning Research,* Vol. 13, No. 1, pp. 307-361（2012）

[Hochreiter 97] Hochreiter, S. and Schmidhuber, J.: Long short-term memory, *Neural Computation,* Vol. 9, No. 8, pp. 1735-1780（1997）

[Jean15] Jean, S., Cho, K., Memisevic, R. and Bengio, Y.: On using very large target vocabulary for neural machine translation, *ACL 2015,* pp. 1-10, Beijing, China（2015）

[Kalchbrenner 13] Kalchbrenner, N. and Blunsom, P.: Recurrent continuous translation models, *EMNLP 2013,* pp. 1700-1709, Seattle, Washington, USA（2013）

[Klein 04] Klein, D. and Manning, C. D.: Parsing and Hypergraphs, Bunt, H., Carroll, J. and Satta, G., eds., *New Developments in Parsing Technology,* pp. 351-372, Kluwer Academic Publishers, Norwell, MA, USA（2004）

[Koehn03] Koehn, P., Och, F. J. and Marcu, D.: Statistical phrase-based translation, *NAACL'03,* pp. 48-54, Stroudsburg, PA, USA（2003）

[Le 14] Le, P. and Zuidema, W.: The inside-outside recursive neural network model for dependency parsing, *EMNLP 2014,* pp. 729-739, Doha, Qatar（2014）

[Li 13] Li, P., Liu, Y. and Sun, M.: Recursive autoencoders for ITG-based translation, *EMNLP 2013,* pp. 567-577, Seattle, Washington, USA（2013）

[Li 14] Li, P., Liu, Y., Sun, M., Izuha, T. and Zhang, D.: A neural reordering model for phrase-based translation, *COLING 2014,* pp. 1897-1907, Dublin, Ireland（2014）

[Liu 13] Liu, L., Watanabe, T., Sumita, E. and Zhao, T.: Additive neural networks for statistical machine translation, *ACL 2013,* pp. 791-801, Sofia, Bulgaria（2013）

[Liu 14] Liu, S., Yang, N., Li, M. and Zhou, M.: A recursive recurrent neural network for statistical machine translation, *ACL 2014,* pp. 1491-1500, Baltimore, Maryland（2014）

[Luong 15] Luong, T., Sutskever, I., Le, Q., Vinyals, O. and Zaremba,W.: Addressing the rare word problemin neural machine translation, *ACL 2015,* pp. 11-19, Beijing, China（2015）

[Mayberry 99] Mayberry, M. R. and Miikkulainen, R.: SARDSRN: A neural network shift-reduce parser, *IJCAI'99*, pp. 820-827, San Francisco, CA, USA（1999）

[Meng 15] Meng, F., Lu, Z., Wang, M., Li, H., Jiang, W. and Liu, Q.: Encoding source language with convolutional neural network for machine translation, *ACL 2015,* pp. 20-30, Beijing, China（2015）

[Miikkulainen 90] Miikkulainen, R.: A PDP architecture for processing sentences with relative clauses, *COLING'90,* pp. 201-206, Stroudsburg, PA, USA（1990）

[Mikolov 10] Mikolov, T., Karafit, M., Burget, L., Cernock, J. and Khudanpur, S.: Recurrent neural network based language model, *INTERSPEECH 2010*, pp. 1045-1048（2010）

[Nivre 08] Nivre, J.: Algorithms for deterministic incremental dependency parsing, *Computational Linguistics,* Vol. 34, No. 4, pp. 513-553（2008）

[Och03] Och, F. J.: Minimum error rate training in statistical machine translation, *ACL 2003,* pp. 160-167, Sapporo, Japan（2003）

[Petrov 06] Petrov, S., Barrett, L., Thibaux, R. and Klein, D.: Learning accurate, compact, and interpretable tree annotation, *ACL 2006,* pp. 433-440, Sydney, Australia（2006）

[Pollack 90] Pollack, J. B.: Recursive distributed representations,

Artificial Intelligence, Vol. 46, No. 1-2, pp. 77-105（1990）
[Rumelhart 88] Rumelhart, D. E., Hinton, G. E. and Williams, R. J.: Neurocomputing: *Foundations of Research,* chapter Learning Representations by Back-propagating Errors, pp. 696-699, MIT Press, Cambridge, MA, USA（1988）
[Schwenk 07] Schwenk, H.: Continuous space language models, *Computer Speech and Language,*Vol. 21, No. 3, pp. 492-518（2007）
[Setiawan 15] Setiawan, H., Huang, Z., Devlin, J., Lamar, T., Zbib, R., Schwartz, R. and Makhoul, J.: Statistical machine translation features with multitask tensor networks, *ACL 2015,* pp. 31-41, Beijing, China（2015）
[Socher 11] Socher, R., Pennington, J., Huang, E. H., Ng, A. Y. and Manning, C. D.: Semi-supervised recursive autoencoders for predicting sentiment distributions, *EMNLP 2011,* pp. 151-161, Edinburgh, Scotland, UK（2011）
[Socher 12] Socher, R., Huval, B., Manning, C. D. and Ng, A. Y.: Semantic compositionality through recursive matrix-vector spaces, *EMNLP 2012,* pp. 1201-1211, Jeju Island,Korea（2012）
[Socher 13] Socher, R., Bauer, J., Manning, C. D. and Andrew, Y., N.: Parsing with compositional vector grammars, *ACL 2013,* pp. 455-465, Sofia, Bulgaria（2013）
[Stenetorp 13] Stenetorp, P.: Transition-based dependency parsing using recursive neural networks, *Deep Learning Workshop at NIPS 2013,* Lake Tahoe, Nevada, USA（2013）
[Su 15] Su, J., Xiong, D., Zhang, B., Liu,Y., Yao, J. and Zhang, M.: Bilingual correspondence recursive autoencoder for statistical machine translation, *EMNLP 2015,* pp. 1248-1258, Lisbon, Portugal（2015）
[Sundermeyer 14] Sundermeyer, M., Alkhouli, T., Wuebker, J. and Ney, H.: Translation modeling with bidirectional recurrent neural networks, *EMNLP 2014,* pp. 14-25, Doha, Qatar（2014）
[Sutskever 14] Sutskever, I., Vinyals, O. and Le, Q. V.: Sequence to sequence learning with neural networks, Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D. and Weinberger, K. Q., eds., *NIPS 2014,* pp. 3104-3112（2014）
[Tai 15] Tai, K. S., Socher,R. and Manning, C. D.: Improved semantic representations from tree-structured long short-term memory networks, *ACL 2015,* pp. 1556-1566, Beijing, China（2015）
[Tamura 14] Tamura, A., Watanabe, T. and Sumita, E.: Recurrent neural networks for word alignment model, *ACL 2014,* pp. 1470-1480, Baltimore, Maryland（2014）
[Vaswani13] Vaswani, A., Zhao, Y., Fossum, V. and Chiang, D.: Decoding with large-scale neural language models improves translation, *EMNLP 2013,* pp. 1387-1392, Seattle,Washington, USA（2013）
[Vinyals 15] Vinyals, O., Kaiser, L., Koo, T., Petrov, S., Sutskever, I. and Hinton, G.: Grammar as a foreign language, Cortes, C., Lawrence, N., Lee, D., Sugiyama, M. and Garnett, R., eds., *NIPS 2015,* pp. 2755-2763, Curran Associates, Inc.（2015）
[Watanabe 15] Watanabe,T. and Sumita, E.: Transition-based neural constituent parsing, *ACL 2015,* pp. 1169-1179, Beijing, China（2015）
[Weiss 15] Weiss, D., Alberti, C., Collins, M. and Petrov, S.: Structured training for neural network transition-based parsing, *ACL 2015,* pp. 323-333, Beijing, China（2015）
[Wu 14] Wu, Y., Watanabe, T. and Hori, C.: Recurrent neural network-based tuple sequence model for machine translation, *COLING 2014,* pp. 1908-1917, Dublin, Ireland（2014）
[Yang 13] Yang, N., Liu, S., Li, M., Zhou, M. and Yu, N.: Word alignment modeling with context dependent deep neural network, *ACL 2013,* pp. 166-175, Sofia, Bulgaria（2013）
[Zhang 14] Zhang, J., Liu, S., Li, M., Zhou, M. and Zong, C.: Bilingually-constrained phrase embeddings for machine translation, *ACL 2014,* pp. 111-121, Baltimore, Maryland（2014）

2016 年 1 月 18 日　受理

著　者　紹　介

渡辺　太郎

1994 年京都大学工学部情報工学科卒業．1997 年同大学院工学研究科情報工学専攻修士課程修了．2000 年 Language and Information Technologies, School of Computer Science, Carnegie Mellon University, Master of Science 取得．2003 年京都大学大学院情報学研究科知能情報学専攻博士後期課程指導認定退学．2004 年京都大学博士（情報学）．ATR および NTT，NICT にて研究員として務めた後，現在，グーグル株式会社ソフトウェアエンジニア．言語処理や機械学習，特に統計的機械翻訳の研究に従事．