

# 自然言語処理における ディープラーニングの発展

坪井 祐太

テキストを扱う自然言語処理においてもディープラーニングが大きな注目を集めている。機械学習課題としての自然言語処理の特徴とよく使われているニューラルネットワーク構造に焦点を合わせて解説する。

キーワード：自然言語処理, ディープラーニング, ニューラルネットワーク

## 1. 導入

自然言語処理 (Natural Language Processing; NLP) は、人間が日常的に使っている言葉をコンピュータで処理する技術の総称である。画像認識や音声認識と同様に、コンピュータで自然言語を扱うことは古くから人工知能研究で掲げられてきた目標の一つであった。また、インターネットの発展により近年では大量のテキストデータが利用可能になっており、企業での研究活動も活発である。

自然言語処理でも機械学習が盛んに取り入れられているためタスクごとに特徴量の設計をする必要があり、他の分野と同様に自然言語処理分野でもディープラーニング (Deep Learning) による特徴量の自動獲得が期待されている。しかし、画像認識や音声認識での成功を受けて大きな注目を集めているが、適切な適用方法が研究されている最中であり基礎技術として確立していると言うにはまだ早いと言えよう。

他の成功分野と比べてディープラーニングの活用が遅れている原因として、入力データの質の違いにあると筆者は考えている。画像認識や音声認識の入力データが連続値で構成されているのに対して、自然言語処理の入力データは離散値で構成されている点が大きく異なる。たとえば、画像のRGB形式は連続値で近い値は近い色を表す。一方、テキストデータは「あ」、「い」…などの文字 (記号) で構成されており、文字間の近さは定義されていない。さらに、一般的にはテキストは一方向に読んでいくため、記号列が基本的な入力データ

構造となる。そのため、自然言語処理では、入力データの特徴から a) 疎な表現の組み合わせをコンパクトに表現することと、b) 可変長の入力に対する固定長のベクトル表現獲得が重要になる。次節ではこれらの課題をくわしく説明する。

## 2. 自然言語処理における課題

自然言語処理では入力データが記号であるため特徴量の設計に人間の直感が働きやすく、疎ベクトルで記号を表現することが一般的である。たとえば、各単語を次元とした特徴空間を考え、ある文書に含まれる単語の次元のみに1、それ以外に0を割り当てた2値ベクトルで文書を表現する。異なり単語数が次元数に相当し、一方で一つの文書に含まれる単語数はわずかであるため、各文書のベクトル表現では1となる要素はわずかである。ただし、この形式では文書内の単語の順番を無視した表現 (Bag of words 表現) となっているが、たとえば“White House”のように単語の接続と個々の単語で意味が異なることを表現できない。このような現象に対処するために2単語以上の接続に次元を割り当てた特徴空間を構成することがある。しかし、複数の単語の組み合わせで特徴空間を構成することは指数的に次元数が増えてしまい、訓練データで観測できる確率が減る問題がある。同様に、処理時間も指数的に増えてしまうため、むやみやたらに明示的な組み合わせ特徴量を追加することはできない。一方、隠れ層の一つしか持たないニューラルネットワークにとって指数的なパラメータ数が必要になる関数を、深い隠れ層を使うことで線形のパラメータ数で表現できることが示されている [1]。明示的に組み合わせ特徴空間を構成する代わりに、深いニューラルネットワークの表現能力を生かして、深い隠れ層によって同等の特徴量

つばい ゆうた

IBM 東京基礎研究所

〒135-8511 東京都江東区豊洲 5-6-52

yutat@jp.ibm.com

間をコンパクトに表現することが期待されている。

また、自然言語処理の入力となる文や文書は文字数や単語数が事例ごとに異なり、基本的に入力は可変長である。一方で、機械学習アルゴリズムで入力長に対して汎化するためには異なる長さの入力を共通の特徴空間に射影する、つまり固定長のベクトル表現に変換することが必要になる。これまでは、文から主語と述語だけを取り出して特徴とする、あるいは2文の共通部分を見つける（アラインメント処理）、などタスクごとの試行錯誤によって固定長のベクトル表現を得ていた。しかし、単純な Bag of words 表現より優れた固定長ベクトル表現を設計することすら容易ではなく、多くの試行錯誤を必要とした。そこで、ディープラーニングによる可変長入力に対する特徴空間の自動獲得が期待されている。

さらには、ディープラーニングはこれまでの自然言語処理の枠組みを大きく変える可能性も秘めている。自然言語処理では複雑な言語現象を扱うために、いくつもの処理をつなげたパイプラインで構成されることが一般的である。基盤処理として単語分割・品詞タグ付け・固有表現抽出・構文解析などがあり、機械翻訳・自動要約・質問応答などの応用ではこれらの解析結果を入力として用いてきた。基盤処理の出力例として、英文 “Neural networks improve the accuracy.” に対する品詞タグ付けと構文解析（係り受け構造解析および句構造解析）、および応用例として翻訳出力を図1に示す。これらの基盤処理はそれぞれ自身だけで独立したタスクとして研究されてきた。品詞タグ付けは単語の品詞を予測するタスク、係り受け構造解析は単語間の依存関係を示す木構造を予測するタスク、句構造解析は名詞句 (NP) や動詞句 (VP) などの再帰構造を予測するタスクである。品詞によって文の内容を示す単語（名詞や動詞など）と機能語（冠詞など）を区別することができ、また構文解析によって長文でも述語項の関係（AがBをCする）などを抽出することが容易になる。自然言語処理における基盤処理は、多くの応用で共通してテキスト入力を構造化し特徴を抽出するために使われてきた。一方、画像認識においてはさまざまな特徴抽出手法による前処理が未加工の RGB 形式を入力としたディープラーニングに置き換えられた [2]。同様に、自然言語処理における基盤処理を明示的に行うことなしに、ディープラーニングによって直接応用問題を解くことが期待されている。

現在のディープラーニングでは、タスクに合ったネットワーク構造を見つけることが重要な課題となってい

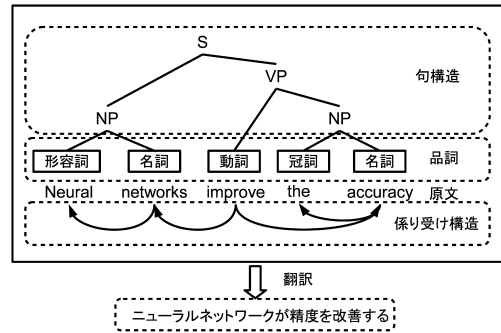


図1 自然言語処理における基盤処理と翻訳の例

る。次節からはニューラルネットワークの構成種類ごとに自然言語処理への適用手法を解説する。

### 3. リカレントニューラルネットワーク

#### 3.1 可変長入力モデル

リカレントニューラルネットワーク (Recurrent Neural Networks; RNN) は、長さ  $T_X$  の入力ベクトル列  $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{T_X})$  から長さ  $T_Y$  の出力列  $Y = (y_1, y_2, \dots, y_{T_Y})$  を予測する関数を次のようにモデル化する。

$$\hat{y}_t = o(\mathbf{w}^o \top \mathbf{h}_t^l),$$

$$\mathbf{h}_t^l = f^l(\mathbf{W}^l[\mathbf{h}_{t-1}^{l-1}; \mathbf{h}_{t-1}^l]),$$

ただし、 $\{l | 1 \leq l \leq L\}$  は隠れ変数ベクトル  $\mathbf{h} \in \mathbb{R}^{N^l}$  の階層を示し、 $l = \theta$  のときは  $\mathbf{h}_t^\theta = \mathbf{x}_t$  とする。  $\mathbf{W}^l$  は  $N^l \times (N^{l-1} + N^l)$  のパラメータ行列、  $\mathbf{w}^o$  は予測  $\hat{y}$  を出力する予測関数  $o$  のパラメータ、  $f$  はシグモイド関数などの非線形関数である。また、先端と終端の隠れ変数ベクトル  $\mathbf{h}_0$  と  $\mathbf{h}_{T+1}$  もパラメータである。RNN は前の隠れ変数ベクトルと下層の隠れ変数ベクトル（または入力ベクトル）から現在の隠れ変数ベクトルを推定するモデルとなっており、  $\mathbf{h}_t$  には1から  $t$  までの可変長入力が自然と記憶されている。また、通常のフィードフォワードネットワークの階層に加えて、時間方向にも深いモデルとなっていることに特徴がある。

たとえば、文の途中まで入力したときの次の単語の分布を予測するためにRNNが用いられている [3]。このモデルでは、システムが出力した文の各単語の確率を推定し掛け合わせることで文全体の自然言語らしさを評価できるため言語モデルと呼ばれ、機械翻訳などで文を生成するときに使用される。原理的には前の文脈すべてをRNNによって記憶したうえで次の単語を予測することができる。このタスクでは  $x$  および  $y$  が単語に相当し、  $\mathbf{W}^1$  の  $x$  に対応するパラメータを単語

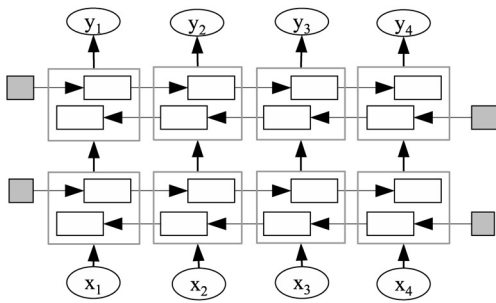


図2 2階層の隠れ層を持つ双方向 RNN の例

のベクトル表現と考えることができる。文献 [4] では、言語モデルの性能だけでなく、学習で得られた単語のベクトル表現で単語間の類似度を測ることで興味深い結果を報告している。たとえば、単語のベクトル表現のコサイン類似度の評価で、単語 “king” と “woman” に近く “man” から遠い単語として “queen” が選ばれるなど、人にも直感的な関係が自動的に学習できていることが示されている。

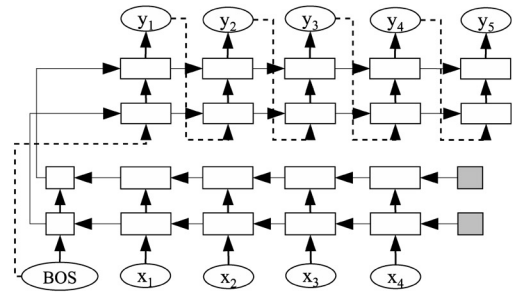
RNN は  $t = 1$  から前向きに走査するだけでなく、 $t = T_x$  から後向きに走査するモデルも考えられ、二つの走査の向きを組み合わせさせた双方向 (Bi-directional) RNN もよく用いられる。

$$\begin{aligned} \hat{y}_t &= o(\mathbf{W}^o[\vec{h}_t^L; \overleftarrow{h}_t^L]), \\ \vec{h}_t^L &= f^L(\overrightarrow{\mathbf{W}}^L[\vec{h}_{t-1}^L; \overleftarrow{h}_t^L; \vec{h}_{t-1}^L]), \\ \overleftarrow{h}_t^L &= f^L(\overleftarrow{\mathbf{W}}^L[\overleftarrow{h}_{t-1}^L; \vec{h}_t^L; \overleftarrow{h}_{t+1}^L]), \end{aligned}$$

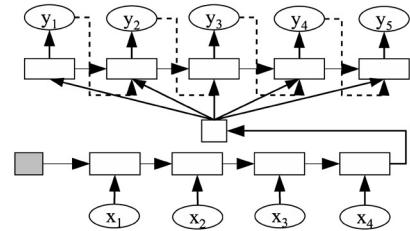
ただし、 $\overrightarrow{\mathbf{h}}, \overrightarrow{\mathbf{W}}$  は前向きの、 $\overleftarrow{\mathbf{h}}, \overleftarrow{\mathbf{W}}$  は後向きの隠れ変数ベクトルとパラメータを示す。2階層の隠れ層を持つ双方向 RNN の例を図2に示す。前向き・後向きの初期状態をグレーの四角で表し、それ以外の隠れ変数ベクトルを白抜ききの四角で示す。

### 3.2 入出力の長さが異なるモデル

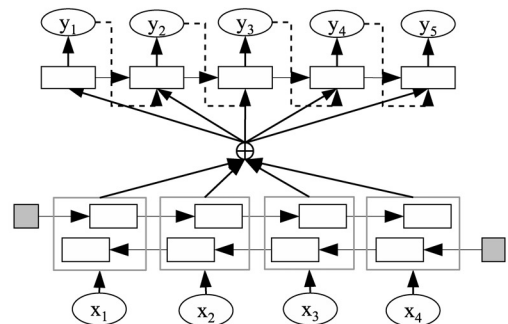
自然言語処理で、特に注目を集めているのは、言語生成にも RNN を適用する手法である。機械翻訳や自動要約のようなタスクでは出力も自然言語の文や文書であるため可変長である。つまり、入力長と出力長がそろっていない ( $T_x \neq T_y$ )。しかし、既存の RNN では入力長と出力長がそろっていることを仮定していた。たとえば、文献 [5] では、双方向 RNN を使って機械翻訳の際には  $T_x = T_y$  となるように前処理でダミー単語を挿入して入力長と出力長をそろえている。このような手法では長さを合わせる前処理に依存してしまうが、最近では前処理が不要となる手法がいくつか提案されている [6~8]。



(a) エンコーダの終端をデコーダの先端に接続 [6]



(b) エンコーダの終端をデコーダの各時刻に接続 [7]



(c) エンコーダの各時刻を集約してデコーダの各時刻に接続 [8]

図3 入力列から出力列を生成する RNN

可変長入出力を実現する手法の共通点は、入力を変数表現に変換するエンコーダ (encoder) と、可変長の出力をするデコーダ (decoder) を接続した構成である。エンコーダは RNN を使って入力文を記憶した固定長のベクトル表現に変換する。デコーダは入力文を表すエンコーダ出力を参照し、1) 一つ前の隠れ変数ベクトルと 2) 一つ前の予測結果の二つを入力とする RNN で、言語モデルと同様に逐次的に次の単語の予測を繰り返すことでテキストを生成する。なお、デコーダは終わりを示す特別な記号 (End of Sentence; EOS) を出力することで出力長を決定する。この構成では、デコーダには各  $t$  で教師信号 (正しい単語) が直接与えられるが、エンコーダにはバックプロパゲーションでのみ教師信号が与えられる。デコーダとエンコーダとの長期の依存関係を学習することは困難であるため、Long

Short-Term Memory (LSTM) [9] や Gated Recurrent Unit (GRU) [7] などの長期の記憶ユニットを用いた RNN が用いられている。一方で、エンコーダによる文全体の表現のデコーダでの利用方法に各手法の違いがあり、図 3 にそれぞれの構成を示す。

文献 [6] では、後向き走査の RNN エンコーダの終了時のベクトルを RNN デコーダの初期状態として前向き走査を行う構成となっている。図 3(a) は隠れ層が 2 層の例で、図中の“BOS”は文頭をあらわす特殊な記号である。なお、元論文では入力列を逆順にすると表現されているが、RNN の後向き走査と同等である。一方、文献 [7] と文献 [8] では、RNN エンコーダの出力を RNN デコーダのすべての時刻  $t$  で入力している。つまり、エンコーダが表現した文全体の情報をデコーダの各時刻の予測において参照する。文献 [7] では、前向き走査の RNN エンコーダの終了時のベクトルをすべての  $t$  で RNN デコーダの入力としている (図 3(b))。また、文献 [8] では双方向 RNN エンコーダの隠れ変数ベクトルの重み付き線形和をすべての  $t$  で RNN デコーダの入力としている (図 3(c))。なお、RNN エンコーダの隠れ変数ベクトルごとの重みも学習されるため、現時点での予測対象に関連ある入力列の一部を表す選択的な記憶ユニットを用いているとも言える。

これらの手法は提案されたばかりで、まだ優劣は明らかにはなっていない点に注意されたい。しかし、可変長の入出力を可能にした点で自然言語処理での幅広い応用が考えられ、拡張した手法が今後も提案されることになるだろう。また、画像を入力としたニューラルネットワークにエンコーダを変更することで画像を説明する文を生成する手法 [10, 11] や RNN デコーダで木構造を生成して構文解析に適用する手法 [12] も提案されている。

#### 4. 畳み込みニューラルネットワーク

畳み込みニューラルネットワーク (Convolutional Neural Networks; CNN) は入力領域内の場所の不変性を学習可能であり、特に画像処理において成功を収めている手法である。画像と違ってテキストは 1 次元の記号列であるため、窓幅  $w$  の単語列を表す局所的な特徴をオーバーラップさせたネットワーク構造が用いられる：

$$z_t^l = \mathbf{W}^l [h_{t-\lfloor w/2 \rfloor}^{l-1}; \dots; h_t^{l-1}; \dots; h_{t+\lfloor w/2 \rfloor}^{l-1}],$$

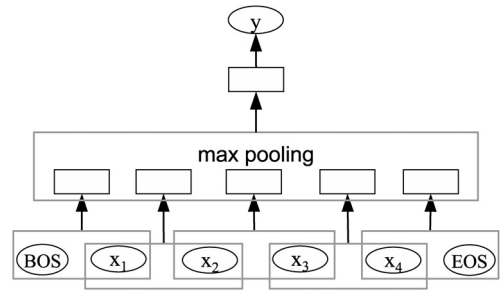


図 4 畳み込み層と 1 層の全結合隠れ層による CNN 例: 窓幅は 2 単語で、最大値プーリングを用いて可変長入力に対して固定長ベクトルを出力することができる

ただし、 $z_t$  は局所特徴、 $h_t^0 = x_t$  である。

また、可変長のテキスト入力を固定長ベクトルで表現するために、複数の局所特徴の最大値のみを上層に持ち上げる最大値プーリング (max-pooling) が用いられることが多い。最大値プーリングでは隠れ変数ベクトル  $h_i$  は  $t$  の  $i$  番目の局所特徴  $z_{t,i}$  を  $t$  に対して最大値をとったものとなる：

$$h_i^l = \max(\{z_{t,i}^l\}_{t=1}^T).$$

最大値プーリングを用いることで、入力長に合わせて局所特徴の数が変わっても全体を表現した固定長のベクトル表現に変換できる (図 4)。

文献 [13] は、複数の基盤処理タスク (品詞タグ付け、句構造チャンキング、固有表現抽出、意味ラベル付与タスク) に CNN を適用し、すべてのタスクで当時の最高性能にかなり接近した性能が得られることを示した。重要な局所特徴量を文中から選択して、文全体を考慮した予測ができたことによると考えられる。また、複数のタスクの教師信号を使って共通の隠れ変数ベクトルを学習することによって各タスクの性能が上がることも示した。このことからニューラルネットワークが何らかの普遍的な言語的特徴量を自動で学習できている可能性が示された。文献 [14] では、CNN で文全体を固定長ベクトルで表現しその他の局所的な情報と合わせて 2 語の関係の分類に適用している。また、文献 [15] では質問応答システムで CNN で質問文全体を固定長ベクトルで表現し回答との近さを測るために利用している。

CNN の構成もさまざまなものが提案されている。入力となる単語は高次元の疎ベクトルではなく低次元に射影した密ベクトルで表現されることが多い。その単語のベクトル表現は大量のデータで教師なし学習をし

たほうがよいことが報告されている [16]. 一方で, 単語のベクトル表現も文字単位から CNN を使って構成することで, 活用形が多い言語の処理やインフォーマルな記述が多いソーシャルネットワークテキストに頑健であることが報告されている [17, 18]. また, 文献 [19] では, 深い畳み込み層を設計する際に, 上位  $k$  個の値を出現位置順に上層に持ち上げる  $k$  最大値プーリング ( $k$ -max-pooling) を提案している.  $k$  個の局所特徴出現位置が保持される点で最大値プーリングとは異なる. “with...that” など接続しない構文的に意味のある表現に反応する特徴量が上位層で観測されたことが報告されている. さらに, 元文を CNN で固定ベクトル表現に変換し, RNN (3 節) で訳文を出力する機械翻訳手法も提案されている [20].

## 5. 再帰ニューラルネットワーク

再帰ニューラルネットワーク (Recursive Neural Networks) は, 3 節の RNN を系列から DAG (Directed Acyclic Graph) 構造に拡張したものであり, 構造に関して深いモデルとなっている. 自然言語処理では図 1 に示したような構文木構造を使ってネットワーク構造が決められることが多い [21]. 図 5 は, 図 1 の句構造を元に再帰ニューラルネットワークを定義した例である. 2 分木を仮定し, 隠れ変数ベクトル  $h$  の左の子と右の子を  $h_L, h_R$  とすると, 隠れ変数ベクトル  $h$  は次のように計算される:

$$h = f(W[h_L; h_R]).$$

木構造の各隠れ変数ベクトルはその子孫である要素の情報を集約していると考えられ, 木構造に即した入力範囲をベクトルで表現することが可能になる. つまり, 再帰ニューラルネットワークを使うことで, 入力文全体だけでなく, 構文的に意味のある単位の固定長のベクトル表現を得ることができる. 文献 [22] では再帰ニューラルネットワークを使って構文木構造に即した句のレベルで好評・不評を判定することができることを報告している.

文献 [23] では質問応答システムで質問文を再帰ニューラルネットワークを使って表現し, 該当する回答を分類している. また, 画像と文を共通の特徴空間であらわすことができたことも報告されている [24]. さらに, 文献 [25] では奥行き方向にも階層を持った深い再帰ニューラルネットワークを提案している (図 6). なお, 入力に対する構文木は基盤技術である構文解析器を用

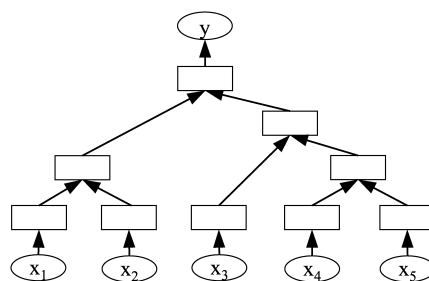


図 5 再帰ニューラルネットワークの例

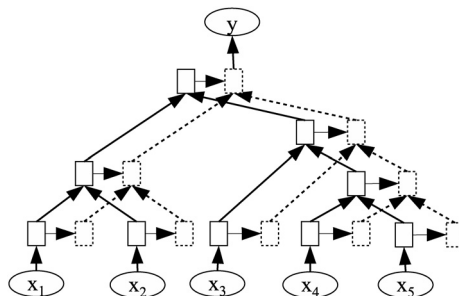


図 6 2 層の深い再帰ニューラルネットワークの例

いて予測したものを利用するが, 自然言語の構文解析自身も自明なタスクではなく解析誤りは発生する.

## 6. フィードフォワードニューラルネットワーク

フィードフォワードニューラルネットワーク (Feed-forward Neural Networks) は最もシンプルなネットワーク構造である:

$$\hat{y} = o(\mathbf{w}^{\top} \mathbf{h}^L),$$

$$\mathbf{h}^l = f^l(\mathbf{W}^l \mathbf{h}^{l-1}).$$

本節以前に取り上げたネットワーク構造と異なり, 隠れ変数ベクトル同士のつながりに分岐がない. 自然言語処理の特徴である可変長の入力に自然に対応した方法ではないが, 入力範囲を固定することで適用が可能になる. 参照できる文脈が限られていることは短所であるが, 既存の自然言語処理手法で使われている線形分類器をニューラルネットワークに置き換えることで適用できるため利用しやすい. また, 既存の疎な特徴ベクトルを入力とした線形分類器と密な特徴ベクトルを入力としたフィードフォワードニューラルネットワークを組み合わせることで, 既存手法を改善することも報告されている [26, 27].

3 節でも触れた言語モデルの構築にフィードフォワードニューラルネットワークを利用した研究には文献 [28]

があり、ニューラルネットワークによる言語モデル研究の発端となった。次の単語を予測するために数単語前のみをニューラルネットワークの入力とする手法で、固定長の翻訳元単語列と翻訳先単語列を入力として機械翻訳にも拡張されている [29]。また、単語の組み合わせを明示的に高次元特徴空間で表現して線形分類する手法に比べて、ニューラルネットワークで暗黙に組み合わせを考慮することで高精度かつ高速に分類できることが報告されている [30]。

## 7. おわりに

構成の異なるニューラルネットワークがどのように自然言語処理で利用されているかを解説した。3節で取り上げたRNNは記憶容量が十分であれば入力全体を表現でき、また可変長の自然言語を出力することもできるため、自然言語処理に適用する際に最も汎用性の高いネットワーク構造であると考えられる。4節で取り上げたCNNは最大値プーリングにより一部の局所特徴だけを選択しているため、一部の情報だけが重要でノイズを除去する必要のあるタスクに向いていると考えられる。5節で取り上げた再帰ニューラルネットワークはネットワーク構造の特定を構文解析に依存しているため汎用性に劣るが、言語学の知識が反映された構造を使うことができるためタスクによっては高い性能を示すと考えられる。また、6節で取り上げたフィードフォワードニューラルネットワークは線形分類器を使った既存手法を置き換える形で広く使われるようになると考えられる一方で、非線形分類器への置き換えという意味では近年高速化が実現されているランダム射影に基づくカーネル法 [31, 32] も再考に値すると考える。

日本語での解説としては本稿以外に文献 [33] がある。また、機械翻訳におけるディープラーニング適用についての日本語解説には文献 [34] がある。ディープラーニングの自然言語処理への適用は発展途上で今後大きなブレイクスルーが見つかり、まったく異なる手法が一般的になる可能性がある。今後の発展に期待したい。

## 参考文献

- [1] O. Delalleau and Y. Bengio, “Shallow vs. deep sum-product networks,” In *Advances in Neural Information Processing Systems (NIPS)*, pp. 666–674, 2011.
- [2] A. Krizhevsky, I. Sutskever and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1097–1105, 2012.
- [3] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký and S. Khudanpur, “Recurrent neural network based language model,” In *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1045–1048, 2010.
- [4] T. Mikolov, W. Yih and G. Zweig, “Linguistic regularities in continuous space word representations,” In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 746–751, 2013.
- [5] M. Sundermeyer, T. Alkhoul, J. Wuebker and H. Ney, “Translation modeling with bidirectional recurrent neural networks,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 14–25, 2014.
- [6] I. Sutskever, O. Vinyals and Q. V. V. Le, “Sequence to sequence learning with neural networks,” In *Advances in Neural Information Processing Systems (NIPS)*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger (eds.), pp. 3104–3112, 2014.
- [7] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, “Learning phrase representations using RNN encoder–decoder for statistical machine translation,” In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014.
- [8] D. Bahdanau, K. Cho and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” 2014. arXiv:1409.0473.
- [9] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, **9**(8), pp. 1735–1780, 1997.
- [10] O. Vinyals, A. Toshev, S. Bengio and D. Erhan, “Show and tell: A neural image caption generator,” 2014. arXiv:1411.4555.
- [11] A. Karpathy and L. Fei-Fei, “Deep visual-semantic alignments for generating image descriptions,” 2014. arXiv:1412.2306.
- [12] O. Vinyals, L. Kaiser, T. Koo, S. Petrov, I. Sutskever and G. Hinton, “Grammar as a foreign language,” 2014. arXiv:1412.7449.
- [13] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. P. Kuksa, “Natural language processing (almost) from scratch,” *Journal of Machine Learning Research*, **12**, pp. 2493–2537, 2011.
- [14] D. Zeng, G. Zhou and J. Zhao, “Relation classification via convolutional deep neural network,” In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pp. 2335–2344, 2014.
- [15] W. Yih, X. He and C. Meek, “Semantic parsing for single-relation question answering,” In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 643–648, 2014.
- [16] Y. Kim, “Convolutional neural networks for sentence classification,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, 2014.
- [17] C. N. dos Santos and B. Zadrozny, “Learning character-level representations for part-of-speech tagging,” In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1818–1826, 2014.

- [18] C. N. dos Santos and M. Gatti, “Deep convolutional neural networks for sentiment analysis of short texts,” In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pp. 69–78, 2014.
- [19] N. Kalchbrenner, E. Grefenstette and P. Blunsom, “A convolutional neural network for modelling sentences,” In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 655–665, 2014.
- [20] N. Kalchbrenner and P. Blunsom, “Recurrent continuous translation models,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1700–1709, 2013.
- [21] R. Socher, “Recursive deep learning for natural language processing and computer vision,” PhD thesis, Stanford University, 2014.
- [22] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. Manning, A. Ng and C. Potts, “Recursive deep models for semantic compositionality over a sentiment treebank,” In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 1631–1642, 2013.
- [23] M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher and H. Daumé III, “A neural network for factoid question answering over paragraphs,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 633–644, 2014.
- [24] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning and A. Y. Ng, “Grounded compositional semantics for finding and describing images with sentences,” *Transactions of the Association for Computational Linguistics*, pp. 207–218, 2014.
- [25] O. Irsoy and C. Cardie, “Deep recursive neural networks for compositionality in language,” In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2096–2104, 2014.
- [26] J. Ma, Y. Zhang, T. Xiao and J. Zhu, “Tagging the web: Building a robust web tagger with neural network,” In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 144–154, 2014.
- [27] Y. Tsuboi, “Neural networks leverage corpus-wide information for part-of-speech tagging,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 938–950, 2014.
- [28] Y. Bengio, R. Ducharme, P. Vincent and C. Janvin, “A neural probabilistic language model,” *Journal of Machine Learning Research*, **3**, pp. 1137–1155, 2003.
- [29] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz and J. Makhoul, “Fast and robust neural network joint models for statistical machine translation,” In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1370–1380, 2014.
- [30] D. Chen and C. Manning, “A fast and accurate dependency parser using neural networks,” In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 740–750, 2014.
- [31] Q. V. Le, T. Sarlócs and A. J. Smola, “Fastfood—computing Hilbert space expansions in loglinear time,” In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 244–252, 2013.
- [32] N. Pham and R. Pagh, “Fast and scalable polynomial kernels via explicit feature maps,” In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 239–247, 2013.
- [33] D. Bollegala, “自然言語処理のための深層学習,” *人工知能*, **29**(2), pp. 195–201, 2014.
- [34] 渡辺太郎, “深層学習による機械翻訳,” *情報・システムサイエティ*, **19**(3), pp. 8–9, 2014.