title: Deep Learning author: Yann LeCun, Yoshua Bengio, and Geoffrey Hinton journal: Nature year: 2015 volume: 512 pages: 436-444 doi: 10.1038/nature14539

深層学習 (ディープラーニング Deep Learning)

要旨

深層学習 (ディープラーニング) は，複数の処理層で構成された計算モデルに，複数の抽象度を持つデータの表現を学習させる手法である。 この手法は，音声認識，視覚物体認識，物体検出などのほか，創薬やゲノミクスなどのさまざまな分野で，最先端の技術を劇的に向上させている。 深層学習は，**誤差逆伝播法 (バックプロパゲーション) アルゴリズム** を用いて，各層の表現を計算するために使用される機械の内部パラメータを，前層の表現からどのように変更すべきかを示すことで，大規模なデータセットの複雑な構造を発見する。 **深層畳み込みネットワーク**は，画像，ビデオ，音声，聴覚の処理に飛躍的な進歩をもたらし，**リカレントネットワーク** はテキストや音声などの連続したデータに光を当てる。

機械学習技術は， 現代社会のさまざまな場面で活躍している。 ウェブ検索，ソーシャルネットワーク上のコンテンツフィルタリング，eコマースサイトでのレコメンドなど，現代社会のさまざまな場面で機械学習技術が活用されており，カメラやスマートフォンなどのコンシューマ製品にも搭載される場合が増えている。 機械学習システムは，画像内の物体を識別したり，音声をテキストに変換したり，ニュース項目や投稿，商品をユーザの興味に合わせてマッチングさせたり，関連性の高い検索結果を選択したりするために使用される。 これらの応用事例では，深層学習 (ディープラーニング) と呼ばれる技術が利用されることが多くなっている。

従来の機械学習技術では，自然界のデータを生のまま処理するには限界があった。 何十年もの間，パターン認識システムや機械学習システムを構築するためには，慎重なエンジニアリング (訳注:特徴工学とも呼ばれる) と，画像の画素値などの生データを，学習サブシステム (多くの場合，分類器) が入力パターンを検出または分類できるような適切な **内部表現** または **特徴ベクトル** に変換する **特徴抽出器** を設計するため，かなりの分野の専門知識が必要であった。

表現学習 とは，機械に生データを入力し，検出や分類に必要な表現を自動的に発見することができる一連の手法である。 深層学習法は，複数のレベルの表現を持つ表現学習法であり，あるレベルの表現 (生の入力から始まる) を，より高い，やや抽象的なレベルの表現に変換する，単純だが非線形のモジュールを組み合わせることで得られる。 このような変換を十分に行うことで，非常に複雑な機能を学習することができる。 分類課題では，表現の上位層は，識別に重要な入力の側面を増幅し，無関係な変動を抑制する。 例えば，画像は **画素値** の配列であり，表現の第1層で学習された特徴は，画像内の特定の方向や位置におけるエッジの有無を表している。 第 2 層では，エッジの位置のわずかな違いに関わらず，エッジの特定の配置を検出することで，モチーフを検出する。 第 3 層では，モチーフをより大きな組み合わせにして，身近な物体のパーツに対応させ，後続層ではそのパーツの組み合わせとして物体を検出する。 深層学習で重要なのは，これらの特徴量の層を人間の技術者が設計するのではなく，一般的な方法でデータから学習することである。 一般的な学習方法を用いて，データから学習される。

深層学習は，長年にわたって人工知能コミュニティの最善の試みに抵抗してきた問題を解決する上で大きな進歩を遂げている。 深層学習は，高次元データの複雑な構造を発見するのに非常に優れていることが判明しており，科学，ビジネス，政府の多くの領域に適用されている。 画像認識 (1-4) や音声認識 (5-7) の記録を塗り替えただけでなく，潜在的な薬物分子の活性予測 (8)，粒子加速器データの分析 (9,10)，脳回路の再構築 (11)，非コード DNA 変異が遺伝子発現や疾患に及ぼす影響の予測 (12,13) などでも，他の機械学習技術を凌駕している。 さらに驚くべきことに，深層学習は自然言語理解のさまざまな課題 (14)，特にトピック分類，感情分析，質問応答 (15)，言語翻訳 (16,17) において極めて有望な結果を出している。

深層学習は，人手をほとんど必要としないため，計算量やデータ量の増加を容易に利用することができ，近い将来，さらに多くの成功を収めることができると考えられる。 現在，ディープニューラルネットワークのために開発されている新しい学習アルゴリズムとアーキテクチャは，この進歩をさらに加速させるだろう。

1. 教師あり学習 (Supervised learning)

機械学習の最も一般的な形態は，ディープかどうかに関わらず，教師あり学習である。 例えば，家，車，人，ペットが写っている画像を分類するシステムを作りたいとする。 まず，家，車，人，ペットの画像の大規模なデータセットを収集し，それぞれにカテゴリのラベルを付ける。 学習の際には，機械に画像を見せて，各カテゴリごとに 1 つの得点というような，ベクトルという形で出力する。 目的のカテゴリーが，全カテゴリー中で最も高い得点を持つようにしたいが，学習前にそれが実現する可能性は低い。 そこで，出力された得点と目的の得点パターンとの誤差 (または距離) を測定する目的関数を計算する。 そして，機械はこの誤差を減らすために，内部の調整可能なパラメータを変更する。 これらの調整可能なパラメータは，しばしば **重み** と呼ばれ，機械の入出力機能を定義する「つまみ」と見なすことができる実数である。 一般的な深層学習システムでは，これらの調整可能な重みが何億個もあり，機械を訓練するためのラベル付けされた例が何億個もある。

重みベクトルを適切に調整するために，学習アルゴリズムは，各重みについて，その重みをわずかに増加させた場合に誤差がどの程度増加または減少するかを示す **勾配ベクトル** を計算する。 そして，重みベクトルは，勾配ベクトルとは逆方向に調整される。

すべての学習例で平均化された目的関数は，重み値の高次元空間における一種の丘陵地帯と見なすことができる。 負の勾配ベクトルは，この風景の中で最も急降下する方向を示しており，出力誤差が平均的に小さくなる最小値に近づく。

実際には，ほとんどの実務者が，**確率的勾配降下法 (SGD)** と呼ばれる手順を使用している。 SDG とは，いくつかの例の入力ベクトルを表示し，出力と誤差を計算し，それらの例の平均勾配を計算し，それに応じて重みを調整するというものである。 この処理は，目的関数の平均が減少しなくなるまで，学習セットからの多くの小さな例のセットに対して繰り返される。 この方法は，小数例のデータセットが全例データセットの平均勾配のノイズの任意の推定値を与えるため，確率的と呼ばれる。 この単純な手順は，はるかに精巧な最適化技術と比較すると，通常，驚くほど早く良い重みセットを見つけることができる (18)。 学習後，**テスト (検証) データセット** と呼ばれる別の例のデータセットでシステムの性能を測定する。 これは，機械の一般化能力，つまり，学習時には見たことのない新しい入力に対して，適切な答えを出す能力を検証するためのものである。

現在，実用化されている機械学習の多くは，手作業で作成した特徴量の上に **線形分類器** を使用している。 2 クラスの線形分類器は，**特徴ベクトル** 成分の加重和を計算する。 この加重和がある閾値以上であれば，入力は特定のカテゴリに属するものとして分類される。

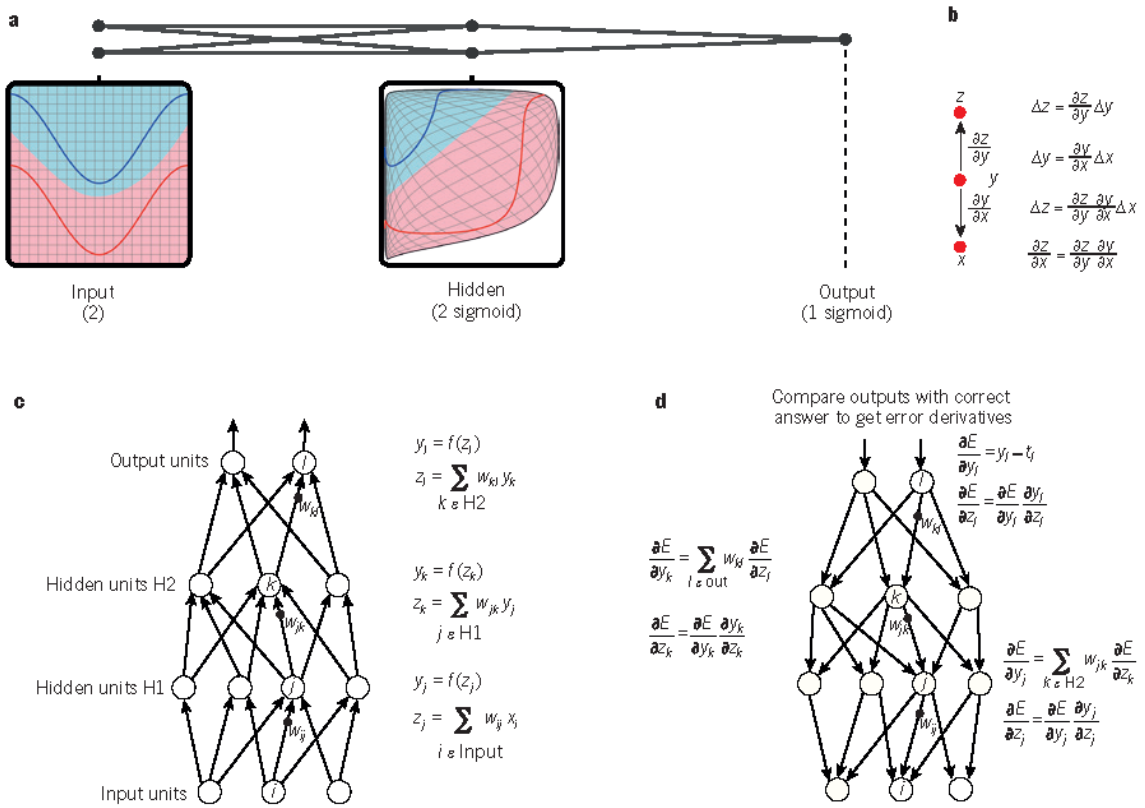


図 1 多層ニューラルネットワークと誤差逆伝播

a 多層構造のニューラルネットワーク(連結されたドットで示す) は，入力空間を歪めて，データのクラス (赤と青の線上にある例) を線形分離可能にすることができる。 入力空間の規則的なグリッド (左) が，隠れユニットによってどのように変換されるか (中) に注目。 これは 2 つの入力ユニット，2 つの隠れユニット，1 つの出力ユニットを持つ例であるが，物体認識や自然言語処理に用いられるネットワークには，数万から数十万のユニットが含まれている。 C. Olah (http://colah.github.io/) の許可を得て転載。

b 微分の連鎖則 (訳注: 高等学校の数学の用語で 合成関数の微分公式 のこと)は，2 つの小さな効果 (x の y に対する微少変化と，y の z に対する変化) がどのように構成されるかを教えてくれる。 x の微少な変化 Δx は， $\partial y/\partial x$ をかけられることによって，まず y の小さな変化 Δy に変換される (つまり，偏微分の定義)。 同様に，Δy の変化は z の変化 Δz を生み出す。 一方の式を他方の式に代入すると，微分の連鎖則 (Δx が $\partial y/\partial x$ と $\partial z/\partial x$ の積の乗算によって Δ に変わる) が得られる。 この法則は x, y, z がベクトルの場合にも有効である (導関数はヤコビ行列となる)。

c 2 つの隠れ層と 1 つの出力層を持つニューラルネットにおいて， 勾配を逆伝播するモジュールを構成する順方向パスの計算に使用される方程式。 各層では，まず， 各ユニットへの全入力 z を計算する。 これは，下層のユニットの出力を加重加算したものである。 次に，非線形関数 $f(\cdot)$ を z に適用して，ユニットの出力を得る。 簡略化のため，バイアス項は省略してある。 ニューラルネットワークで使われる非線形関数には，近年よく使われる **ReLU (整流線形ユニット)** $f(z) = \max(0, z)$ のほか，**ハイパーボリックタンジェント**，$f(z) = (\exp(z) - \exp(-z))/(\exp(z) + \exp(-z))$ や **ロジスティック関数** $f(z) = 1/(1 + \exp(-z))$ などの一般的な **シグモイド** (訳注: シグモイドとは S 字状の と言った意味合い) がある。

d 逆向パスの計算に使用される方程式。 各隠れ層では，各ユニットの出力に関する誤差導関数を計算する。 これは，上位層のユニットへの全入力に関する誤差導関数の加重和である。 次に，出力に対する誤差導関数に $f(z)$ の勾配を乗じることで， 入力に対する誤差導関数に変換する。 出力層では，ユニットの出力に関する誤差微分を，コスト関数を微分することで計算す

る。 これにより，ユニット l のコスト関数が $0.5(y_l - t_l)$ (2) の場合，$y_l - t_l$ となり，t_l は目標値である。 $\partial E/\partial z_k$ がわかれば，下位層のユニット j からの接続の重み w_{jk} の誤差微分は，ちょうど $y_j \partial E/\partial z_k$ となる。

1960 年代以降，線形分類器は入力空間を超平面で区切られた半空間という非常に単純な領域にしか切り分けられないことがわかっていた (19)。 しかし，画像認識や音声認識のような問題では，入出力関数は，物体の位置や向き，照明の変化，音声ピッチやアクセントの変化などの無関係な入力の変化には鈍感である一方，特定の微細な変化 (例えば，白いオオカミとサモエドと呼ばれるオオカミに似た白い犬種の違い) には非常に敏感である必要がある。 画素レベルでは，2 頭のサモエドが異なるポーズや環境で撮影された画像は互いに大きく異なるが，サモエドとオオカミが同じ姿勢で同じような背景で撮影された 2 つの画像は互いによく似ていることがある。 線形分類器やその他の「浅い」分類器は，前者 2 つの画像を同じカテゴリに分類する一方で，後者 2 つの画像を区別することはできない。 このため，浅い分類器には，**選択性ー不変性ジレンマ** を解決する優れた特徴抽出器が必要になる。 つまり，識別に重要な画像の側面には選択性があり，動物のポーズなどの無関係な側面には不変性がある表現を生成する必要がある。 分類器をより強力にするためには，**カーネル法** のように汎用の非線形特徴を用いることができるが (20)，**ガウシアンカーネル** で生じるような汎用の特徴では，学習例から遠く離れたところで学習者をうまく一般化することができない(21)。 従来手法では，優れた特徴抽出器を手作業で設計していたが，これにはかなりのエンジニアリング技術と専門領域の知識が必要である。 しかし，汎用の学習手順を用いて良い特徴を自動的に学習することができれば，このようなことはすべて避けることができる。 これが深層学習の最大の利点である。

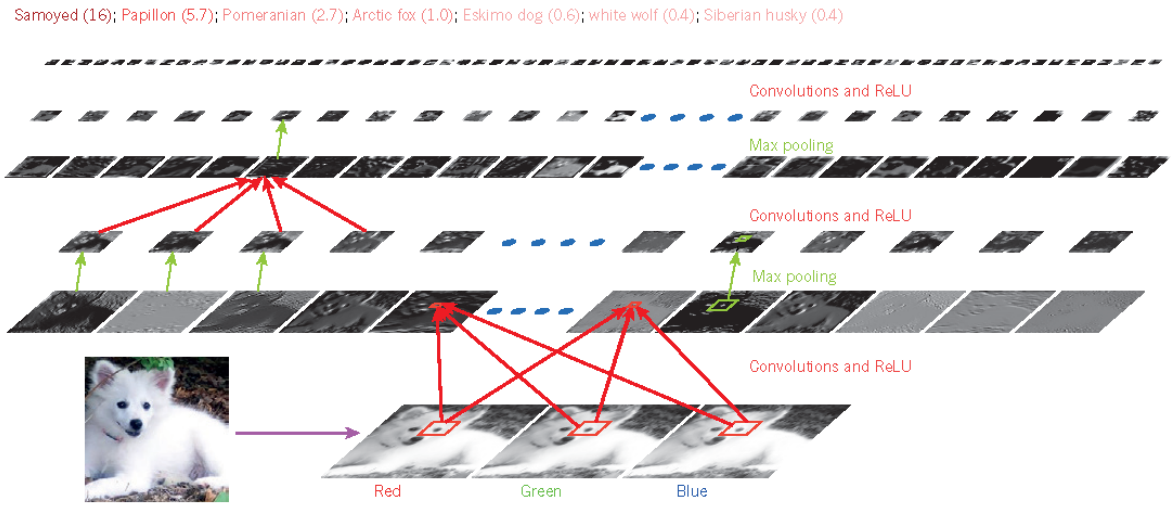


図 2 畳み込みネットワークの内部
典型的な畳み込みネットワークアーキテクチャの各層 (水平方向) の出力 (フィルタではない) を，サモエドの画像 (左下: 右下 は RGB (赤，緑，青) の入力 に適用したもの。 各矩形の画像は，画像の各位置で検出された，学習された特徴の 1 つに対する出力に対応する特徴地図である。 情報はボトムアップで流れ，下位の特徴は方向性のあるエッジ検出器として機能し，出力の各画像クラスに対してスコアが計算される。 図中の ReLU は 整流線形ユニット を示す。

深層学習アーキテクチャは，単純なモジュールの多層スタックであり，すべて (またはほとんど) のモジュールが学習の対象となり，多くのモジュールが非線形の入出力マッピングを計算する。 スタックの各モジュールは，入力を変換して，表現の選択性と不変性の両方を高める。 複数の非線形層，例えば 5〜20 層の深さがあれば，システムは入力に対して非常に複雑な機能を実装することができる。 この機能は，サモエドと白い狼を区別するような微細な詳細特徴に敏感であると同時に，背景，ポーズ，照明，周囲の対象などの無関係な大きな変化には影響を受けない。

2. 多層アーキテクチャを訓練するための誤差逆伝播

パターン認識の黎明期 (22, 23) から，研究者の目的は，手作業で作成した特徴量を学習可能な多層ネットワークに置き換えることだったが，そのシンプルさにもかかわらず，1980 年代半ばまで，この解決策は広く理解されていなかった。 結論から言うと，多層アーキテクチャーは単純な確率的勾配降下法で学習できる。 モジュールが入力と内部の重みの比較的滑らかな関数である限り，誤差逆伝播法を用いて勾配を計算することができる。 この方法が可能であること，そしてそれが機能することは 1970 年代から 1980 年代にかけて，いくつかの異なるグループによって独自に発見された (24-27)。

多層モジュールの重みに対する目的関数の勾配を計算する誤差逆伝播法は，導関数の連鎖法則を応用したものに他ならない。 ここで重要なのは，あるモジュールの入力に対する目的関数の導関数 (または勾配) は，そのモジュールの出力 (または次のモジュールの入力) に対する勾配から逆算することで計算できるということである (図1)。 誤差逆伝播法の方程式を繰り返し適用して， 全モジュールに勾配を伝搬させることができる。 最上位の出力 (ネットワークが予測を生成する場所) から始まり，最下部 (外部入力が入力される場所) に至るまで，勾配を伝搬させる。 これらの勾配が計算されると，各モジュールの重みに対する勾配を簡単に計算することができる。

深層学習の多くのアプリケーションでは，固定サイズの入力 (例えば，画像) を固定サイズの出力 (例えば，いくつかのカテゴリーのそれぞれに対する確率) にマッピングすることを学習する **前向き (フィードフォワード)ニューラルネットワーク** アーキテクチャ (図1) が使用されている。 ある層から次の層に移る際には，一連のユニットが，前層からの入力の加重和を計算し，その結果を非線形関数に通す。 現在，最もよく使われている非線形関数は，半波整流器 $f(z) = \max(z, 0)$ である整流線形ユニット(ReLU)である。 過去数十年のニューラルネットでは $\tanh(z)$ や

$1/(1+\exp(-z))$ など，より滑らかな非線形関数が使用されていたが，ReLUは通常，層数の多いネットワークでは学習速度が速く，教師なしの事前学習なしで深い教師付きネットワークの学習が可能である(28)。 入力層や出力層に属さないユニットを従来は **隠れユニット** と呼んでいた。 隠れ層は，入力を非線形に歪ませ，最後の層でカテゴリーが線形に分離できるようにすると考えることができる (図1)。

1990 年代後半，ニューラルネットと **誤差逆伝播法(バックプロパゲーション)** は，機械学習のコミュニティからは見放され，コンピュータビジョンや音声認識のコミュニティからは無視されていた。 少ない予備知識で有用な多段の特徴抽出器を学習することは不可能であると広く考えられていた。 特に，単純な勾配降下法では，劣悪なローカルミニマム，つまり，わずかな変化でも平均誤差を減らすことができないような重み設定に陥ってしまうと考えられていた。

実際には，大規模なネットワークでは貧弱な **局所最小 (ローカルミニマム)** が問題になることはほとんどない。 初期条件にかかわらず，システムはほとんど常に非常に似た品質の解に到達する。 最近の理論的，経験的な結果は，一般的に局所的な最小値は深刻な問題ではないことを強く示唆している。 それどころか，勾配がゼロで，表面がほとんどの次元で上にカーブし，残りの次元で下にカーブするサドルポイントが，組み合わせ的に多数詰め込まれている (29,30)。 分析の結果，下向きに曲がる方向がわずかしかないサドルポイントが非常に多く存在するが，ほとんどすべてのサドルポイントで目的関数の値が非常に似通っていることがわかったようだ。 したがって，アルゴリズムがこれらのサドルポイントのどれに引っかかるかはあまり重要ではない。

ディープフィードフォワードネットワークへの関心は，2006 年頃，カナダ高等研究所 (CIFAR) が集めた研究者グループによって復活した (参考文献31〜34)。 この研究者たちは，ラベル付きデータを必要とせずに特徴検出器の層を作成できる教師なし学習法を導入した。 特徴検出器の各層を学習する目的は，下位層にある特徴検出器の活動 (または生の入力) を再構成またはモデル化できるようにすることである。 この再構築の目的を用いて，徐々に複雑になる特徴検出器の層をいくつか「事前学習」することで，深層ネットワークの重みを適切な値に初期化することができる。 その後，出力ユニットの最終層をネットワークの最上部に追加し，標準的なバックプロパゲーションを用いて深層システム全体を微調整することができる (33-35)。 これは，手書きの数字を認識したり，歩行者を検出したりする際に，特にラベル付けされたデータの量が非常に限られている場合に，非常にうまく機能した (36)。

この事前学習法を音声認識に初めて適用したのは，プログラムの作成が容易で (37)，ネットワークの学習速度を 10 倍から 20 倍に高めることができる高速な GPU (Graphics Processing Unit) の登場であった。 2009 年には，音波から抽出した係数の短い時間窓を，窓の中心にあるフレームが表すさまざまな音声の断片の確率にマッピングするために，この手法が使用された。 このシステムは， 少ない語彙を使用する標準的な音声認識ベンチマークで記録的な結果を出し (38)， すぐに大規模な語彙の課題で記録的な結果を出すように開発された (39)。 2012 年には，2009 年に開発されたディープネットのバージョンが，多くの主要な音声認識グループ (6) によって開発され，すでに Android 携帯電話に搭載されている。 データセットが小さい場合，教師なしの事前学習を行うことで，過学習を防ぐことができる (40)。 これにより，ラベル付けされた例の数が少ない場合や，「ソース」課題の例はたくさんあるが，「ターゲット」課題の例はほとんどないという転送設定の場合に，一般化が大幅に向上する。 深層学習が回復すると，事前学習の段階は小さなデータセットにしか必要ないことがわかった。

しかし，ディープフィードフォワードネットワークの中には，隣接する層間に完全な接続性があるネットワークよりも，はるかに簡単に学習でき，一般化できる特定のタイプがあった。 それは，畳み込みニューラルネットワーク (ConvNet) である (41,42)。 ConvNet は，ニューラルネットワークが人気を失っていた時期に多くの実用的な成功を収め，最近ではコンピュータビジョンのコミュニティで広く採用されている。

3. 畳み込みニューラルネットワーク

ConvNets は，複数の配列で構成されるデータを処理するように設計されている。 例えば，カラー画像は，3 つの色チャンネルの **画素 (ピクセル) 強度** を含む 3 枚の 2 次元配列で構成されている。 多くのデータモダリティは，複数の配列の形をしている。 言語を含む信号や系列は 1 次元，画像やオーディオのスペクトログラムは 2 次元，ビデオやボリューム画像は 3 次元である。 ConvNets には，自然な信号の特性を利用した 4 つの重要なアイデアがある。 **局所的な接続**，**重み共有**，**プーリング**，多数の層の使用である。

典型的な ConvNet アーキテクチャ (図2) は，一連のステージとして構成されている。 最初の数ステージは，2 種類の層で構成されている。 畳み込み層とプーリング層である。 畳み込み層ユニットは，特徴地図で構成されており，各ユニットは，フィルターバンクと呼ばれる重みセットを介して，前層の特徴マップのローカルパッチに接続されている。 この局所的な重み付けの和の結果は，ReLU などの非線形性に通される。 特徴地図の全ユニットは，同じフィルターバンクを共有する。 層内の異なる特徴地図は，異なるフィルターバンクを使用する。 このようなアーキテクチャを採用した理由は 2 つある。 まず，画像などの配列データでは，局所的な値の集まりは相関性が高く，局所的に特徴的なモチーフが形成されていることが多く，これを容易に検出することができる。 2 つ目は，画像などの信号の局所的な統計量は，場所に依存しないということである。 つまり，画像のある部分に現れるモチーフは，どこにでも現れる可能性がある。 そのため，異なる場所にあるユニットが同じ重みを共有し，配列の異なる部分で同じパターンを検出するというアイデアが生まれた。 数学的には，特徴地図が行うフィルタリング操作は離散的な畳み込みであり，それが名前の由来となっている。

畳み込み層の役割は，前層の特徴の局所的な結合を検出することだが，プーリング層の役割は，意味的に類似した特徴を 1 つにまとめることである。 モチーフを形成する特徴の相対的な位置は多少異なることがあるため，各特徴の位置を粗視化することで，モチーフを確実に検出することができる。 一般的なプーリングユニットは，1 つの特徴地図 (またはいくつかの特徴地図) 内のユニットのローカルパッチの最大値を計算する。 隣接するプーリングユニットは，1 行または 1 列以上シフトしたパッチから入力を受けることで，表現の次元を下げ，小さなシフトや歪みに対する不変性を作り出す。 畳み込み，非線形，プーリングを 2 段または 3 段重ね，さらに畳み込み層と完全連結層を重ねる。 ConvNet での勾配のバックプロパゲーションは，通常のディープネットワークと同様に簡単で，すべてのフィルターバンクのすべての重みを学習することができる。

深層ニューラルネットワークは，自然界の信号の多くが，下位の特徴を組み合わせることで上位の特徴が得られる「構成的階層」であるという性質を利用している。 画像では，エッジの局所的な組み合わせがモチーフを形成し，モチーフがパーツに集まり，パーツがオブジェクトを形成する。 音声やテキストにおいても，音から音声，音素，音節，単語，文へと同様の階層が存在する。 プーリングにより，前層の要素の位置や見え方が変わっても，表現はほとんど変わらない。

ConvNet の畳み込み層とプーリング層は，視覚神経科学における単純細胞と複雑細胞という古典的な概念に直接インスパイアされたものであり (43)，全体のアーキテクチャは，視覚野の腹側経路における **LGN-V1-V2-V4-IT** 階層を彷彿とさせるものである (44)。(訳注: LGN は外側膝状体，V1 は第 1 次視覚野，V2 は第 2 次視覚野，V4 は第4 次視覚野，IT は下側頭葉の意味) ConvNet モデルとサルに同じ絵を見せると，ConvNetの高レベルユニットの活性化は，サルの下頭側頭葉皮質にある 160 個のニューロンのランダムセットの分散の半分を説明する (45)。 ConvNet の祖先は **ネオコグニトロン (neocognitron)** (46) で，アーキテクチャは似ているが，バックプロパゲーションのような **エンドツーエンド** の教師あり学習アルゴリズムは持っていなかった。 時間遅延ニューラルネットと呼ばれる原始的な 1 次元の ConvNet は，音素と単純な単語の認識に使用された (47,48)。

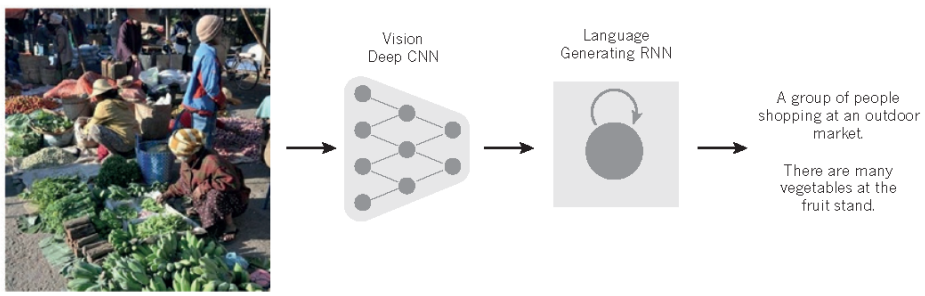
畳み込みネットワークの応用例は，1990 年代初頭に数多くあり，音声認識 (47) や文書読解のための時間遅延ニューラルネットワークに始まった (42)。 この文書読解システムでは，言語制約を実装した確率モデルと共同で学習した畳み込みネットワークを使用していた。 1990 年代後半には，このシステムは米国の全小切手の 10% 以上を読み取っていた。 ConvNet をベースにした光学式文字認識や手書き認識のシステムは，後に Microsoft 社によって多数導入された (49)。 また ConvNet は，顔や手を含む自然画像中の物体検出 (50,51) や 顔認識 (52) のために，1990 年代初頭に実験された。

4. ディープ畳み込みネットワークによる画像理解

2000 年代初頭から，ConvNets は，画像中の物体や領域の検出，領域切り分け，認識に適用され，大きな成功を収めてきた。 これらは，交通標識の認識 (53)，生物学的画像の切り分け (54)，特にコネクティオミクス (55)，自然画像中の顔，字，歩行者，人体の検出 (36,50,51,56-58) など，ラベル付きデータが比較的豊富に存在する課題であった。 最近の ConvNets の実用的な成功例としては，顔認識が挙げられる (59)。

重要なのは，画像を画素レベルでラベリングできることであり，これは自律移動ロボットや自動運転車などの技術に応用できるだろう (60, 61)。 Mobileye 社や NVIDIA 社などの企業は，このような ConvNet ベースの手法を，自動車用の次期ビジョンシステムに採用している。 また，自然言語理解 (14) や音声認識などのアプリケーションも重要になってきている (7)。

このような成功にもかかわらず，ConvNets はコンピュータビジョンや機械学習の主流から見放されていたが，2012 年に ImageNet コンテストが開催された。 深層畳み込みネットワークを，ウェブから収集した約 100 万枚の画像と 1,000 種類のクラスを含むデータセットに適用したところ，競合する最良のアプローチのエラー率をほぼ半減させるという素晴らしい結果が得られた (1)。 この成功は，**GPU**, **ReLU**, **ドロップアウト (dropout)** (62) と呼ばれる新しい正則化技術，および既存の学習例を変形させてより多くの学習例を生成する技術を効率的に使用したことによるものである。 この成功は，コンピュータビジョンに革命をもたらした。 ConvNets は現在，ほとんどすべての認識・検出課題で主流となっており (4,58,59,63-65), いくつかの課題では人間の性能に近づいている。 最近の見事なデモンストレーションでは，ConvNets とリカレントネットモジュールを組み合わせて，画像のキャプションを生成している (図3)。




A woman is throwing a **frisbee** in a park.


A **dog** is standing on a hardwood floor.

A **stop** sign is on a road with a mountain in the background


A little **girl** sitting on a bed with a teddy bear.


A group of **people** sitting on a boat in the water.


A giraffe standing in a forest with **trees** in the background.

図3：画像からテキストへ。 ディープ畳み込みニューラルネットワーク (CNN) がテスト画像から抽出した表現を追加入力として，リカレントニューラルネットワーク (RNN) が生成した脚注。 参考文献からの許可を得て再掲(102)。 RNN に，入力画像の異なる場所に注意を向ける機能を持たせた場合 (中段と下段)。 RNN が各単語を生成する際に (太字)，入力画像内の異なる場所に注目する能力を与えた場合 (中段と下段，明るいパッチほど注目された)，RNN はこれを利用して画像からキャプションへのより良い「翻訳」を実現することがわかった(86)。

最近の ConvNet アーキテクチャは，10〜20 層の ReLU，数億の重み，数十億のユニット間接続を備えている。 このような大規模なネットワークの学習は，わずか 2 年前には数週間かかっていたが，ハードウェア，ソフトウェア，アルゴリズム の並列化の進歩により，学習時間は数時間に短縮された。

ConvNet ベースの視覚システムの性能は，Google, Facebook, Microsoft, IBM, Yahoo!, Twitter, Adobe などのほとんどの主要テクノロジー企業や，急速に増加しているスタートアップ企業が研究開発プロジェクトを開始し， ConvNet ベースの画像理解製品やサービスを展開している。

ConvNet は，チップやフィールド・プログラマブル・ゲート・アレイ への効率的なハードウェア実装に容易に従うことができる (66,67)。 NVIDIA, Mobileye, Intel, Qualcomm, Samsung などの多くの企業が，スマートフォン，カメラ，ロボット，自動運転車などの実時間視覚アプリケーションを実現するために，ConvNet チップを開発している。

5. 分散表現と言語処理

深層学習理論によると，深層ネットワークは，分散表現を使用しない古典的な学習アルゴリズムと比較して，2 つの異なる指数的な利点を持っている (21)。 これらの利点はいずれも構成力から生じるもので，基礎となるデータ生成分布が適切な構成構造を持つことに依存する (40)。 まず，分散表現を学習することで，学習した特徴量の値の，学習時に見られた組み合わせ以外の新たな組み合わせに一般化することができる (例えば n 個の二値特徴量で 2^n 通りの組み合わせが可能) (68,69)。 第二に，ディープネットワークで表現の層を構成することで，別の指数的な利点 (70) が得られる可能性がある (深さに対して指数的)。

多層ニューラルネットワークの隠れ層は，ネットワークの入力を，ターゲットとなる出力を予測しやすいように表現することを学習する。 このことは，多層ニューラルネットワークを訓練して，以前の単語の局所的な文脈から，連続した次の単語を予測することでよくわかる (71)。 文脈中の各単語は 1 対 N のベクトルとしてネットワークに提示される。 つまり，1 つの成分が 1 の値を持ち，残りは 0 である (訳注: ワンホット表現，あるいは，ワンホットベクトルと呼ばれる)。 第 1 層では，各単語が異なるパターンの活性化，すなわち単語ベクトルを作り出す (図4)。 言語モデルでは，ネットワークの他の層は，入力された単語ベクトルを，予測された次の単語のための出力単語ベクトルに変換するように学習される。 このネットワークは，記号の分散表現を学習する際に初めて実証されたように，それぞれが単語の個別の特徴と解釈できる多くの活性成分を含む単語ベクトルを学習する (27)。 これらの意味的特徴は，入力に明示的に存在するものではなかった。 これらは，入力記号と出力記号の間の構造化された関係を複数の「微小規則」に分解するのに適した方法として，学習手続きによって発見された。 単語ベクトルの学習は，単語の配列が実際のテキストの大規模なコーパスから来ており，個々の微小規則が信頼できない場合にも非常に有効であることが判明した (71)。 例えば，ニュース記事中の次の単語を予測するように学習した場合，火曜日と水曜日の学習した単語ベクトルは，スウェーデンとノルウェーの単語ベクトルと同様に酷似している。 このような表現は，その要素 (特徴) が相互に排他的ではなく，多くの構成が観測されたデータの変動に対応することから，**分散表現** と呼ばれる。 これらの単語ベクトルは，専門家が事前に決めたものではなく，ニューラルネットワークが自動的に発見した学習済みの特徴で構成されている。 テキストから学習された単語のベクトル表現は，現在，自然言語の応用事例で非常に広く使用されている (14,17,72-76)。

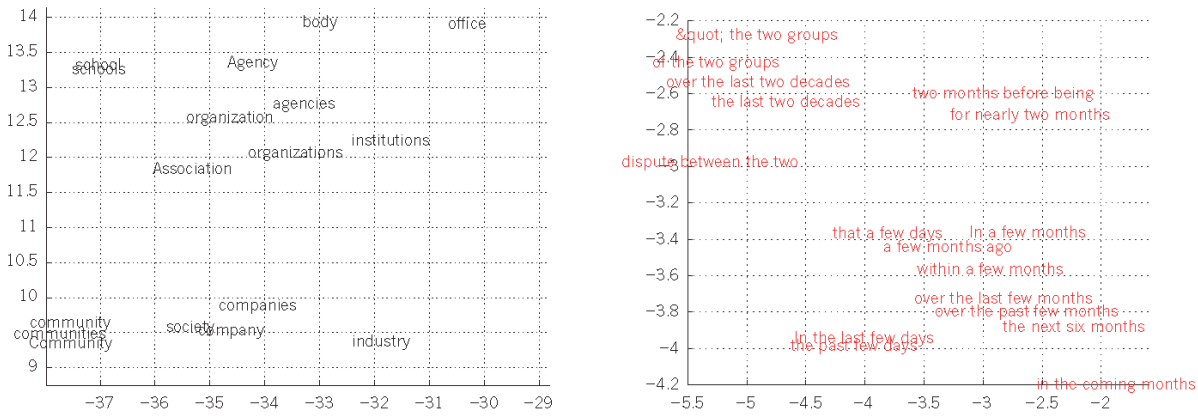


図 4 学習した単語ベクトルの可視化。

左 言語をモデル化するために学習した単語表現を，**t-SNE** アルゴリズム (1) を用いて非線形に 2 次元に投影して可視化した図 (03)。 **右** 英語からフランス語への符号化器・復号化器リカレントニューラルネットワークによって学習された文節を 2 次元で表現したもの (75)。 意味的に類似した単語や単語の並びが，近くの表現にマッピングされていることがわかる。 単語の分散表現は，バックプロパゲーションを用いて，各単語の表現と，一連の単語の次の単語 (言語モデリングの場合) や翻訳された単語系列全体 (機械翻訳の場合) などの目標量を予測する関数を共同で学習することで得られる (18,75)。

表現の問題は，論理を重視した認知パラダイムと，ニューラルネットワークを重視した認知パラダイムとの間の議論の中心となっている。 論理に基づくパラダイムでは，シンボルの実体は，他のシンボルの実体と同一か非同一かのどちらかであるという特性しかないものである。 記号を使って推論するためには， 慎重に選ばれた推論規則の中で， 記号を変数に結びつけなければならない。 対照的に，ニューラルネットワークは，大きな活性値ベクトル，大きな重み行列，スカラー非線形性を用いて，楽な常識的推論を支える高速な「直感的」推論を行う。

ニューラル言語モデル (71) が登場する以前，言語の統計的モデリングの標準的なアプローチは，分散表現を利用していなかった。 それは N 個までの長さの短い記号列 (N-gram と呼ばれる) の出現頻度をカウントすることに基づいていた。 N-gram の 数 は V^N (V は語彙数) のオーダーであり，一握りの単語以上の文脈を考慮するには，非常に大きな学習コーパスが必要になる。 一方，ニューラル言語モデルは，各単語を実数値の特徴量のベクトルと関連付けることで，意味的に関連する単語がそのベクトル空間の中で互いに近くなるため，一般化することができる (図4)。

6. リカレントニューラルネットワーク

バックプロパゲーションが導入された当初は，リカレントニューラルネットワーク (RNN) の学習に使用するのが主流であった。 音声や言語など，連続した入力を伴う課題には，RNN を使う方がよい場合が多い (図5)。 RNN は，入力系列を 1 要素ずつ処理し，隠れユニットに，系列の過去の全要素の履歴に関する情報を暗黙のうちに含む「状態ベクトル」を保持する。 異なる離散的な時間ステップにおける隠れユニットの出力を，深い多層ネットワークの異なるニューロンの出力であるかのように考えると (図 5 の右)，RNN の学習にバックプロパゲーションを適用する方法が明らかになる。

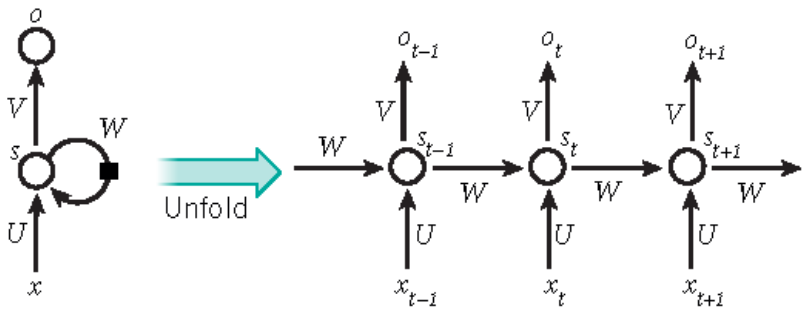


図 5 **リカレントニューラルネットワークとその前進計算に関わる計算の時間的展開を示したもの**
人工ニューロン (例えば，ノード s の下にグループ化された隠れユニットで，時刻 t に 値 s_t を持つ) は，前時間ステップで他のニューロンからの入力を得ている (これは，左の 1 時間ステップの遅延を表す黒い四角で表現されている)。 このようにして，リカレントニューラルネットワークは，要素 x_t の入力系列を要素 o_t の出力系列にマッピングすることができ，各 o_t は ($t' \leq t$の場合) すべての前の $x_{t'}$ に依存する。 各時間ステップで同じパラメータ (行列 U, V, W) が使用される。 ネットワークが一連の出力 (例えば単語) を生成し，それぞれが次の時間ステップの入力として使用されるようなバリエーションを含め，他の多くのアーキテクチャが可能である。 バックプロパゲーションアルゴリズム (図1) は，右図の展開されたネットワークの計算グラフに直接適用することができ，すべての状態 s_t とすべてのパラメータに関する全誤差 (例えば，正しい出力系列スを生成する対数確率) の微分を計算することができる。

RNN は非常に強力な動的システムだが，逆伝播された勾配が時間ステップごとに大きくなったり小さくなったりするため，多くの時間ステップで爆発したり消滅したりすることが多く，RNN の学習には問題があることがわかっている (77, 78) (訳注: これらを，**勾配爆発問題** および **勾配消失問題** という)。

RNN のアーキテクチャ (79,80) や学習方法 (81,82) の進歩により，RNN は，文章中の次の文字 (83) や連続する単語 (75) を予測するのに非常に適していることがわかっているが，より複雑な課題にも使用できるようになっている。 例えば，英語の文章を 1 単語ずつ読んだ後，英語の「符号化器」ネットワークを訓練して，隠れユニットの最終的な状態ベクトルが，その文章で表現されている思考をうまく表現できるようにすることができる。 この思考ベクトルは，共同で学習したフランス語の「復号化器」ネットワークの初期隠れ状態として (あるいは追加入力として) 使用することができる。 この分布から特定の最初の単語が選択され，復号化器ネットワークの入力として提供されると，翻訳の 2 番目の単語の確率分布が出力され，終端文字が選択されるまで繰り返される (17,72,76)。 この処理では，英語の文に依存した確率分布に従ってフランス語の単語の系列が生成される。 このように，かなり素朴な方法で機械翻訳を行うことで， 瞬く間に最先端の機械翻訳に対抗できるようになった。 このことから，文章を理解するためには，推論ルールを使用して操作される内部の記号表現のようなものが必要なのかどうかについて，重大な疑問が生じる。 これは，日常的な推論には多くの類似性が同時に存在し，それぞれが結論の妥当性に寄与するという見解とより相性が良い (84,85)。

フランス語の文章の意味を英語の文章に翻訳する代わりに，画像の意味を英語の文章に「翻訳」することを学習することができる (図 3)。 ここでの符号化器は，深層 ConvNet で，最終隠れ層で画素を活性値ベクトルに変換する。 復号化器には，機械翻訳やニューラル言語モデリングに用いられるような RNN を使用している。 最近，このようなシステムへの関心が高まっている (文献 86 に記載の例を参照)。

RNN は，時間的に展開すると (図5)，すべての層が同じ重みを共有する，非常に深いフィードフォワードネットワークと見なすことができる。 RNN の主な目的は **長距離依存** 的な依存関係を学習することだが，理論的にも経験的にも，情報を非常に長く保存することを学習するのは難し

いことがわかっている (78)。

この問題を解決するために，ネットワークに明示的な記憶を持たせることが考えられる。 この種の最初の提案は，特別な隠れユニットを使用する **長短期記憶 (LSTM)** ネットワークで，その自然な動作は入力を長期間記憶することである (79)。 メモリセルと呼ばれる特別なユニットは，アキュムレータやゲートリーキーニューロンのような働きをする。 しかし，この自己接続は，メモリの内容を消去するタイミングを決定することを学習する別のユニットによって乗算的にゲートされる。

その後，LSTM ネットワークは，特に時間ステップごとに複数の層を持つ場合，従来の RNN よりも効果的であることが証明され (87)，音響から転写の文字の並びまで，音声認識システム全体の構築が可能になった。 また，機械翻訳で優れた性能を発揮する符号化器と復号化器のネットワークには，現在，LSTM ネットワークやそれに関連する形態のゲートユニットが使用されている (17,72,76)。

この 1 年間，複数の著者が RNN をメモリモジュールで拡張するさまざまな提案を行ってきた。 提案には，RNN が読み書きできる「テープのような」メモリでネットワークを拡張する「**ニューラルチューリング機械 Neural Turing Machine**」(88) や，通常のネットワークを一種の連想メモリで拡張する「メモリネットワーク」(89) などがある。 メモリネットワークは，標準的な質問応答のベンチマークで優れた性能を発揮している。 記憶は，ネットワークが後に質問に答えるように要求されるストーリーを記憶するために使用される。

単なる記憶にとどまらず，通常は推論や記号の操作が必要な作業にも，ニューラルチューリングマシンやメモリーネットワークが使われている。 ニューラルチューリングマシンは「アルゴリズム」を教えることができる。 例えば，各シンボルがリスト内での優先順位を示す実数値を伴うソートされていない系列を入力として，ソートされたシンボルのリストを出力することを学習することができる (88)。 メモリネットワークは，テキストアドベンチャーゲームのような設定で世界の状態を把握し，物語を読んだ後，複雑な推論を必要とする質問に答えられるように学習できる (90)。 あるテスト例では，ネットワークは「指輪物語」の 15 文章版を見せられ，「フロドは今どこにいる？」などの質問に正しく答える (89)。

7. 深層学習 (ディープラーニング) の未来

教師なし学習（91-98）は，深層学習への関心を復活させる触媒効果があったが，その後，純粋な教師付き学習の成功の影に隠れてしまった。 今回のレビューでは注目していないが，長期的には教師なし学習の重要性が増してくると予想している。 人間や動物の学習は，ほとんどが教師なしで行われる。 我々は，すべての物体の名前を教えてもらうのではなく，世界を観察することで世界の構造を発見する。

人間の視覚は，小さくて高解像度の焦点と，大きくて低解像度の周囲を使って，知的で課題に特化した方法で視神経配列を順次サンプリングする能動的な処理である。 今後の視覚の進歩は，エンドツーエンドで学習され，強化学習を用いてどこを見るべきかを決定する ConvNets と RNN を組み合わせたシステムによってもたらされるものと期待している。 深層学習と強化学習を組み合わせたシステムはまだ初期段階にあるが，すでに分類課題では受動的視覚システム (99) を凌駕し，さまざまなビデオゲームのプレイを学習する際にも素晴らしい結果を出している（100）。

自然言語理解もまた，深層学習が今後数年間で大きな影響を与えることが期待される分野である。 RNN を使って文章や文書全体を理解するシステムは，一度に一つの部分だけを選択的に処理する戦略を身につけることで，より優れたものになると期待している (76, 86)。

最終的に，人工知能の大きな進歩は，表現学習と複雑な推論を組み合わせたシステムによってもたらされるだろう。 音声認識や手書き認識には，深層学習と単純な推論が古くから用いられてきたが，記号表現のルールベースの操作を大きなベクトルに対する演算に置き換えるには，新しいパラダイムが必要である (101)。