

Python で体験する 深層学習

—Caffe, Theano,
Chainer, TensorFlow—

博士(文学) 浅川 伸一 著

コロナ社

ま え が き

恩師小嶋謙四郎先生は「生きることは表現することだ」と書いた¹⁾。出来の悪い弟子は「なに」を「どう」表現すれば「生きる」ことになるのか、師の意図を汲みかねて依然として途方に暮れたままである。だが、こういう形でしかできない表現があると考えた。作家の土屋つかさ先生から本書のヒントをいただいた。放課後だけでも魔術師になりたいと願うが、その淡い努力が本書である。

本書を上梓するにあたって、この分野のトップカンファレンスの一つである NIPS (neural information processing system) 2007 の講演でボトー (I. Bottou, 確率的勾配降下法の提唱者, 本書で取り上げた) の「コンピュータ社会のメタファ」を紹介したい。

社会には2種類のコンピュータ(計算者)が存在する。生産者(作成機, メーカー)は、仕事をして所得を得る。生産者の活動によりデータが生まれる。一方, 思考家(思考機, シンカー)は収益を増大させるため, データを分析することで競争優位性を見出す。コンピュータの台数が増大しても, 生産者/思想家比は一定である。データは生産者数に比例して増大するが, 思考家の数はデータの増加速度以上には増えない。すなわち

1. 学習に利用可能なコンピュータ資源はデータの増量速度以上に速くならない。
2. データマイニングのコストは収益を越えることができない。
3. 知的生命はデータ系列から学習する。
4. 機械学習のアルゴリズムの多くは, データの増加速度以上の処理容量を必要とする。

本書は知的な機械, あるいは思考家, をつくるために必要な思考のための道具

を提供することを目指した。生産者を駆使して、現在手に入る材料から、競争優位性を獲得することを意図している。材料の中には、他の思考家がどのように思考しているのかも含まれる。思考家が必要ありと判断した場合、その思考家の思考過程を追体験する再現可能性を示そうと試みた。

本書のサポートページも参照していただきたい。編集の労をとってくださったコロナ社にお礼を申し上げる。コロナ社のご尽力なくして陽の目を見ることがなかった。文章の内容についての責任はすべて筆者にあるが、もし本書がなんらかの役に立つ内容であるとすれば、それはコロナ社のおかげである。東京電機大学高橋達二先生、甲野祐先生、高橋ゼミナールの皆様には本書のキッカケとなった話題を提供させていただいた。草稿に目を通していただいた亀田雅之さん、岩井健二さん、青木瑠璃さんにはお世話になった。末筆ながら、岩船幸代氏の変わらぬご助力に感謝申し上げる。

2016年6月

浅川 伸一

† <http://www.cis.twcu.ac.jp/~asakawa/2015deep-corona/>

目 次

1. はじめに

1.1 深層学習の現状	2
1.2 日常の中の深層学習	5
1.3 実践による理解	8
1.4 本書で使用した環境	10
1.5 必要な予備知識	10
1.6 取り上げた話題	11
1.7 理論および技術の展開	11
1.8 本書で用いた処理系, ツール, パッケージ	13
1.8.1 Caffe	13
1.8.2 Chainer	14
1.8.3 TensorFlow	14
1.8.4 Theano	14
1.9 本書の構成	14
1.10 数 学 表 記	15
章 末 問 題	16

2. Python

2.1 動作環境	18
2.2 Python2 か Python3 か	18

2.3 参 考 情 報	19
2.4 コーディングスタイル	20
2.5 Python と NumPy 略説	21
2.5.1 基本データ型	22
2.5.2 コ ン テ ナ	22
2.5.3 関 数	28
2.5.4 ク ラ ス	29
2.5.5 NumPy	30
2.6 Python による深層学習フレームワーク	39
2.6.1 Caffe	40
2.6.2 Theano	48
2.6.3 Chainer	58
2.6.4 LSTM とゲート付き再帰ユニット	61
2.6.5 TensorFlow	63
2.7 scikit-learn	66
章 末 問 題	67

3. ニューラルネットワークの基盤となる考え方

3.1 ニューロンモデル	68
3.1.1 生物学的ニューロンモデル	69
3.1.2 人工ニューロンのモデル	73
3.2 パーセプトロン	76
3.3 バックプロパゲーション	79
3.3.1 バックプロパゲーションの問題点	81
3.3.2 中間層の意味	86
3.4 多 層 化	89

3.5	リカレントニューラルネットワーク	90
3.5.1	単純再帰型ニューラルネットワーク	91
3.5.2	リカレントニューラルネットワークの学習	92
3.5.3	系列予測と系列生成	95
章末問題		98

4. 深層学習理論

4.1	畳み込みニューラルネットワーク	99
4.1.1	LeNet	103
4.1.2	一般化行列積 (GEMM)	109
4.1.3	AlexNet	119
4.1.4	畳み込みニューラルネットワークの諸技法	122
4.2	LSTM	125
4.2.1	CEC の呪い	126
4.2.2	勾配消失問題, 勾配爆発問題	126
4.2.3	ゲート付き再帰ユニット	135
4.2.4	双方向再帰モデル: BRNN	136
4.2.5	LSTM の変種	138
4.3	確率的勾配降下法	140
4.3.1	勾配降下法: GD	141
4.3.2	ニュートン法: 2GD	141
4.3.3	確率的勾配降下法: SGD	141
4.3.4	2階確率的勾配降下法: 2SGD	142
4.3.5	AdaGrad	144
4.3.6	AdaDelta	145
4.3.7	Adam	147

4.3.8 RMSprop	149
4.3.9 Nesterov	150
章 末 問 題	151

5. 深層学習の現在

5.1 GoogLeNet	152
5.2 VGG	154
5.3 SPP	156
5.4 ネットワーク・イン・ネットワーク	158
5.5 残差ネット ResNet	159
5.6 画像からの領域切り出し	162
5.7 R-CNN	165
5.8 Fast R-CNN	166
5.9 Faster R-CNN	169
5.10 リカレントニューラルネットワーク言語モデル	172
5.10.1 従 来 法	172
5.10.2 リカレントニューラルネットワーク言語モデル上の意味空間 ..	175
5.10.3 スキップグラム (word2vec)	178
5.10.4 ニューラルネットワーク機械翻訳	180
5.10.5 ニューラル言語モデル	183
5.10.6 注 意 の 導 入	185
5.10.7 プログラムコード生成	186
5.10.8 ニューラルチューリングマシン	186
5.10.9 構 文 解 析	189
5.10.10 音 声 認 識	190
5.10.11 リカレントニューラルネットワーク関係の実装サイト	191

6. 深層学習の展開

6.1	ニューラル画像脚注づけ	192
6.1.1	MS COCO	193
6.1.2	グーグルの方法	195
6.1.3	スタンフォード大の方法	197
6.1.4	UC バークリー校の方法	199
6.1.5	ニューラル画像脚注づけへ注意の導入	199
6.2	強化学習によるゲーム AI	201
6.2.1	ボードゲーム	201
6.2.2	強化学習	203
6.2.3	TD 学習	204
6.2.4	TD バックギャモン	205
6.2.5	Q 学習	207
6.2.6	ディープ Q 学習ネットワーク	208
6.2.7	ロボット制御	208
6.3	メモリネットワーク	210
6.4	強化学習ニューラルチューリングマシン	213
6.5	顔情報処理	213
6.5.1	従来法	213
6.5.2	深層学習による顔情報処理	218
	章末問題	230

7. おわりに

7.1	工学と哲学の狭間にある尊厳	231
-----	---------------	-----

7.2 変容する価値と社会	233
章 末 問 題	235
付 録	236
A.1 画像処理基本と用語	236
A.1.1 SIFT	237
A.1.2 HOG	239
A.1.3 Bag-of-Words	239
A.1.4 スーパーピクセルズ	240
A.1.5 グラフ理論による領域分割	240
A.1.6 MCG	242
A.2 ミコロフのリカレントニューラルネットワーク言語モデル	242
A.3 自然言語処理における指標	242
A.3.1 TD-IDF	242
A.3.2 パ ッ ケ ー ジ	243
A.3.3 BLEU	243
A.3.4 F-値	243
A.3.5 パープレキシティ	243
A.3.6 METEOR	244
A.3.7 ROUGE-L	244
A.3.8 CIDEr	244
A.3.9 バックオフ平準化	245
A.3.10 トークナイザ	245
A.3.11 WER	245
A.4 ヴィオラ・ジョーンズ法による顔領域の切り出し	246
引 用 文 献	253
索 引	272

5

深層学習の現在

ここでは、LeNet の深化型継承モデル（deeper successors, 命名は Long らによる⁶⁷⁾）である GoogLeNet, VGG, SPP, ResNet を概説する。

5.1 GoogLeNet

GoogLeNet は、ILSVRC2014 への登録チーム名であり、LeNet⁷⁰⁾ へのオマージュであると書いてある³⁾。ILSVRC2014 の当時最高性能（SOTA）であるが、単純な CPU パワーに頼るのではなく、後述の R-CNN を組み込むことでパラメータ数の減少と成績向上とを実現した。GoogLeNet の概要を図 5.1 に示した。

図 5.1 の楕円で囲まれた部分が図 5.2 のインセプションモジュールであり、

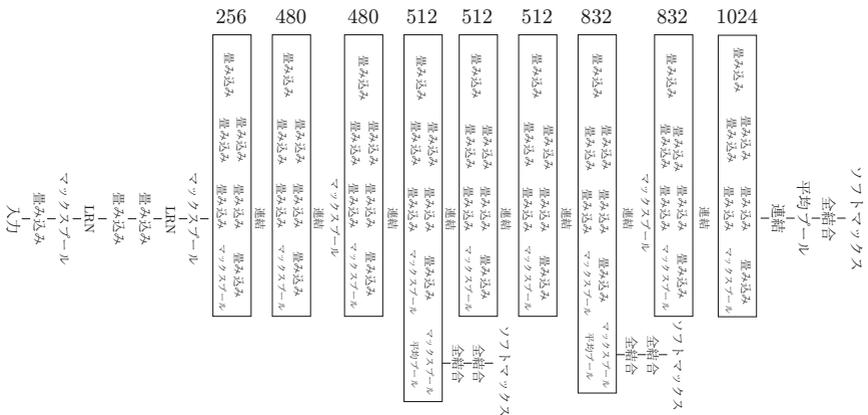


図 5.1 GoogLeNet の構成（文献 3）の図 3 を改変）

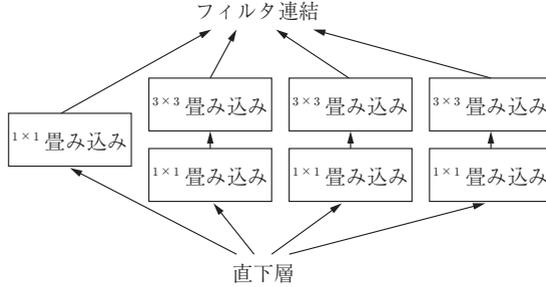


図 5.2 GoogLeNet のインセプションモジュール
(文献 3) の図 2 を改変)

楕円上の数字がモジュールの個数を記してある[†]。深層学習では性能と規模との間にトレードオフが発生する。ネットワークの深さ（層数）と幅（層内ユニット数）を増やせば性能は向上する可能性がある。しかし、大規模データ、大規模ネットワークによる解には以下のような問題が発生する。

- (1) パラメータ数の増大により過学習の恐れがある。
- (2) 計算コストが増大する。

CPU や GPU の利用環境が整備されてきたが、今後さらに、画像認識を越えて、動画、意図、感情、信念、意思、意味などの高次認知機能を考慮すると、さらなる計算コストの削減が必要であろう。GoogLeNet では、パラメータ数を減らす目的もあり、インセプション構造が採用された（図 5.2）。

図 5.2 ではインセプションモジュール内で畳み込み演算が連結されている。これにより次元削減と投射の減少による疎性結合とを実現させた。素朴にサイズの異なる核関数とマックスプーリングの結果を高次層へ送るよりも、構造をつくり込んだほうが総結合数は減少する。GoogLeNet をまとめると以下のようになる。

- (1) 平均的なプーリング層のカーネル関数の大きさは 5×5 で、ストライド間隔は 3 であった。

[†] 原語は inception だが、2010 年に公開されたハリウッド映画 Inception の邦題が「インセプション」である（クリストファー・ノーラン監督・脚本・製作。レオナルド・デカプリオ主演）。本書でも日本語に訳さなかった。

- (2) サイズ 1×1 のカーネル関数の活性化関数は、整流線形ユニットである。
- (3) 完全結合層では 1024 ユニットが使われ、出力関数は整流線形ユニットである。
- (4) ドロップアウト層では確率 0.7 で出力を脱落させた。
- (5) ソフトマックスによる分類層では、1000 カテゴリーの分類が行われた。
- (6) 学習は非同期確率的勾配降下法 (stochastic gradient descent) で、ミニバッチのサイズは 256、モーメント係数 0.9、重み崩壊係数は 5×10^{-4} である。
- (7) ドロップアウト率は 0.5 である。
- (8) データ拡張 (data augmentation) は AlexNet と同じで、 256×256 の原画像から 224×224 の領域を切り出し、拡大して用いた。
- (9) 37 万回の繰り返しで終了 (74 エポック)。訓練終了までに 2~3 週間を費やした。

GoogLeNet の情報

- (1) 論文：<http://arxiv.org/abs/1409.4842>
- (2) スライド：<http://www.image-net.org/challenges/LSVRC/2014/slides/GoogLeNet.pptx>
- (3) 動画：http://youtu.be/ySrj_G5gHWI
- (4) コード：<http://vision.princeton.edu/pvt/GoogLeNet/code/>
- (5) 画像 (ImageNet) による訓練済みネットワーク：<http://vision.princeton.edu/pvt/GoogLeNet/ImageNet/>
- (6) 場所 (Places) による訓練済みネットワーク：<http://vision.princeton.edu/pvt/GoogLeNet/Places/>

5.2 VGG

VGG⁴⁾ は ILSVRC2014 に参加したチーム名である。ローカライゼーション課題で 1 位、分類課題では 2 位であった⁴⁾。VGG は 16 層から 19 層の畳み込

ミニニューラルネットワークで多層化の可能性を追求した。このため、他のパラメータをすべて固定し、畳み込みの核関数の幅は 3×3 のみであった。入力画像は 224×224 ピクセルの RGB 画像に固定し、前処理は各画素の RGB 値から平均値を引いただけである。画像の各点は 2 次元格子状に配置されるので、畳み込み演算の最小単位は 1×1 である。この場合近傍を考慮しない線形フィルタと同義である。一方画素の 8 近傍を走査する 3×3 は実質的な空間フィルタとしては最小のフィルタである。VGG のフィルタサイズは 3×3 に固定された。畳み込み演算のフィルタ間隔 (stride, ストライド) は 1 画素であり、畳み込み層への入力における空間充填 (spatial padding) も 1 画素であった。空間充填は 3×3 の空間フィルタに対して後処理において空間解像度を保証する効果がある。マックスプーリングは 2×2 画素に対して実行され、ストライドは 2 であった。VGG はこのように局在し、単純な構成で畳み込み層を積み上げ、3 層から 5 層おきにマックスプーリング層を挟んで、最上位の 4 層は全結合、最終層はソフトマックス層で認識に至る。VGG では、畳み込み層数を A から E まで 5 種類のアーキテクチャと LRN を介在させた、A-LRN の全 6 ネットワークを採用した。

VGG と GoogLeNet (分類課題で当時最高性能 SOTA を示した) とは独立に開発された。しかし、同様の発想で 22 層、畳み込み演算のフィルタ (特徴) の幅は 1×1 , 3×3 , 5×5 である。したがって類似したネットワーク構成である。学習時の重み崩壊係数 (L2 ペナルティ) は 5×10^{-4} であり、全結合層のドロップアウト率は 0.5 であった。

このような単純なネットワークを積み上げて多層化した VGG と GoogLeNet は、人間の成績に肉薄した。上位 5 カテゴリーを挙げる分類課題では GoogLeNet の誤判別率 6.7%, VGG は 6.7%, 人間は 5.1% である⁶²⁾。その後、人間超え 4.94% の論文がアーカイブ (<http://arxiv.org/>) に掲載された¹²⁶⁾。

VGG の情報

- (1) プロジェクトページ: http://www.robots.ox.ac.uk/~vgg/research/very_deep/

- (2) 論文：<http://arxiv.org/abs/1409.1556>
- (3) スライド：http://www.image-net.org/challenges/LSVRC/2014/slides/VGG_ILSVRC_2014.pdf
- (4) 動画：<https://www.youtube.com/watch?v=j1jIoHN3m0s>

5.3 SPP

SPPはILSVRC2014に参加したマイクロソフトアジア (MSRA) のチームである。彼らのモデル名はSPP (spatial pyramid pooling, 空間ピラミッドプーリング)-netである⁵⁾。対象検出で2位, 画像分類で3位であった。SPPの概略を図5.3に示した。GoogLeNetのインセプションモジュール(図5.2)との違いは, GoogLeNetが空間的に局在化したモジュールである一方で, SPP-netはAlexNet(図4.21)の最終畳み込み層(5層)と全結合層(6層)との間にSPP(図5.3)を挿入した構成である。第5層の出力を分割したプーリング結果を第6層への入力信号としている。特徴地図は原理的に位置情報を伝達しないが, 局在化した情報であるので束ねれば位置情報の概要は伝達可能である。SPPはこの局在化した特徴地図がもつ位置情報が解釈可能となるようなプーリングを行ったとの見方もできる。SPPの介在によって, 小さな位置のズレではなく, 対象が画像上のどの位置にあっても頑健な認識が可能となる。

図5.4左はフォークリフトが縦列駐車している画像である。仮に下位層においてフォークリフト認識のための十分な特徴抽出が行われているのであれば, SPPにとって, フォークリフトの縦列駐車は有利な証拠が重なっていることになる。一方, 図5.4右はペーパーナイフが正解である。1. 万年筆, 2. ボールペン, 3. 金槌, などと認識し, 類似した特徴をもつ物体と誤認識した。特徴地図の情報を空間的に束ねるSPPの操作が, 逆に類似特徴をもつ物体との重複により誤認識率を高めてしまう例である。

索引

【あ】		【お】		局所反応正規化	
アーキテクチャ	68	オートエンコーダ	102		119, 120, 223
赤池情報量基準	79	オーバーラッププーリ		虚誤率	251
アクター	205	ング	119, 120	【く】	
アクタークリティック		オッカムのカミソリ	79	クイックソート	21
モデル	205	重み減衰法	61	クイックプロップ	146
アフィン変換	219	重み崩壊	124	空間異方性	113
アマゾンメカニカルターク		重み崩壊法	61	空間ピラミッドプーリング	
	194				156
アルファベータ法	203	【か】		GoogLeNet	103, 152
AlexNet	223	過学習	123	繰り返し	54
【い】		核関数	104	グリッド細胞	113
イジケビッチモデル	72	学習係数	78	【け】	
一般化行列積	33, 109	学習済み畳み込みニューラ		形式ニューロン	74
一般化デルタ則	79	ルネットワークモデル		系列生成課題	96
イテレータ	26		168	系列予測課題	95
異同判断	214	確率的勾配降下法	140, 141	ゲート付き再帰ユニット	135
因果律	136	隠れマルコフモデル	172	結合係数	77
インセプション	113, 153	活性化関数	74	結合係数行列	104
【う】		ガボール関数	105	結合ベジアンモデル	221
ヴァイオラ・ジョーンズ法		完全結合層	111	検出率	251
	217	【き】		【こ】	
【え】		キーポイント記述子	238	交差検定	124
エコーステートネット		キーワード付き引数	28	格子状 LSTM	139
ワーク	136	帰還信号	69	恒等写像	160
枝刈り	124	脚注作成	134	勾配クリップ	127, 130
N-gram	182	逆行性健忘	82	勾配降下法	141
エネルギー関数	100	強化学習	203	勾配消失問題	81, 126
F-値	243	教師あり学習	69	勾配爆発問題	81, 126
		教師なし学習	69	誤差逆伝播	79
		共分散行列	214	固有値	214
		共有変数	48		

- 【に】
 ニュートン法 141
 ニューラル画像脚注づけ 134, 192
 ニューラルチューリングマシン 186
 ニューラル TD ギャモン 206
 ニューラルネットワーク機械翻訳 180
- 【ね】
 ネオコグニトロン 99
 ネットワーク・イン・ネットワーク 158
- 【は】
 バークリー校視覚学習センター 14, 40
 バーセプトロン ———の学習則 76, 78
 パープレキシティ 243
 ハールの基底関数 246
 背側径路 69
 排他的論理和 88
 ハイパータングジェント関数 75
 配列指定子 33
 バックオフ平準化 245
 バックプロパゲーション 79
 バッケージ 18
 バッチ正規化 160
 破滅的干渉 81, 82
 破滅的忘却 82
- 【ひ】
 ビームサーチ 196
 表情認識モデル 226
- 【ふ】
 ファイナルファンタジー 212
 部 位 221
- フィードフォワード 69
 フィッシャー顔 214, 215
 フィッシャー情報行列 141
 フィッシャーの線形判別分析 214
 プール型配列指定子 35
 フェイスネット 223
 FORTRAN 119
 深 さ 115
 腹側径路 69
 物体仮説 164
 プラトン問題 97
 プルトフォース 202
 フレームワーク 18
 ブロードキャスト 37
 プロトバッファ 44
 プロブ 42
- 【へ】
 平 均 41
 平均情報量 243
 平均精度 163
 ベイズ情報量規準 79
 ヘシアン行列 140, 143
 ヘッセ行列 12
 ベンジオのモデル 172
 変分リカレントニューラルネットワーク 138
- 【ほ】
 方向付き勾配のヒストグラム 239
 ホジキン・ハックスリーモデル 72
 ホップフィールドモデル 102
 ポリシー 205
 ボルツマンマシン 102
- 【ま】
 マイクロソフト COCO 193
 マッカロック・ピッツモデル 73
 マックスプーリング 122
- 丸暗記 202
- 【み】
 ミコロフのモデル 174
- 【め】
 メモリネットワーク 210
- 【も】
 モーメント法 144
 モデル選択基準 79
- 【や】
 ヤコビアン 127
 ヤコビ行列 56
- 【ゆ】
 有限マルコフ決定過程 204
 有向非環グラフ 189
 ユニバーサル関数 38
 指輪物語 210
- 【ら】
 ラピッチモデル 70
 ラムダ学習 205
 ラムダ式 25
 ランドマーク 221
- 【り】
 リカーシブニューラルネットワーク 125, 189
 リカレントニューラルネットワーク 90
 リカレントニューラルネットワーク言語モデル 242
 リスト 22, 23
 リニア 61
 領域切り出し畳み込みニューラルネットワーク 162
 領域提案ネットワーク 169
- 【る】
 ルカン 100, 103

<p>【れ】</p> <p>レイヤ 42</p> <p>レーベンシュタイン距離 245</p>	<p>【ろ】</p> <p>ロジスティック関数 74</p> <p>【わ】</p> <p>word2vec 63, 67, 178</p>	<p>1-of-k 表現 89, 96</p> <p>ワンホットベクトル 61, 89, 92, 96</p>
--	---	---

<p>【A】</p> <p>action unit 227</p> <p>AdaBoost 247</p> <p>AdaDelta 144, 145</p> <p>AdaGrad 144</p> <p>Adam 144, 147</p> <p>affine transform 219</p> <p>AIC 79</p> <p>Akaike information criterion 79</p> <p>AlexNet 119, 223</p> <p>alpha-beta method 203</p> <p>Amazon Mechanical Turk 194</p> <p>AMT 194</p> <p>anisotropy 113</p> <p>architecture 68</p> <p>attention 185</p> <p>AU 227</p> <p>AUDN 226</p> <p>autism 213</p> <p>【B】</p> <p>backward 59</p> <p>back-propagation through time 93</p> <p>bag of words 179</p> <p>bag-of-words 239</p> <p>basic linear algebra sub-programs 110</p> <p>batch normalization 160</p> <p>Bayesian information criterion 79</p>	<p>beam search 196</p> <p>BIC 79</p> <p>biological neuron model-ing 69</p> <p>BLAS 110</p> <p>BLEU 243</p> <p>blob 42</p> <p>BOW 179, 239</p> <p>BPTT 92, 93, 125</p> <p>BRNN 136</p> <p>brute force 202</p> <p>BVLC 14, 40</p> <p>【C】</p> <p>Caffe 13, 40</p> <p>caffemodel 17</p> <p>catastrophic forgetting 82</p> <p>catastrophic interference 81, 82</p> <p>CEC 126</p> <p>——の呪い 126</p> <p>Chain 58, 59</p> <p>Chainer 14, 58</p> <p>chainer.Variable 59</p> <p>CIDeR 244</p> <p>classification 214</p> <p>CNN 99</p> <p>comprehension 24</p> <p>conditional random fields 172</p> <p>connection weight 77</p> <p>convolution 105</p> <p>convolutional neural networks 99</p>	<p>credit assignment problem 81</p> <p>CRF 172</p> <p>cross validation 124</p> <p>【D】</p> <p>DAG 189</p> <p>DDFD 224</p> <p>DeepID 220</p> <p>deep dense face detector 224</p> <p>depth 115</p> <p>directed acyclic graph 189</p> <p>DOG 237</p> <p>dorsal stream 69</p> <p>DQN 208</p> <p>dropout 119</p> <p>【E】</p> <p>early stopping 124</p> <p>echo state network 136</p> <p>eigenface 214</p> <p>EM アルゴリズム 108</p> <p>ESN 136</p> <p>exclusive OR 88</p> <p>exponential moving average 148</p> <p>【F】</p> <p>F-値 243</p> <p>FaceNet 223</p> <p>face in the wild 213</p> <p>facial action coding system 227</p>
--	--	--

FACS	227	histograms of oriented	
false correct	251	gradient	239
Faster R-CNN	169	HMM	172
Fast R-CNN	166	HOG	115, 239
feed back	69		
feed forward	69		
finite Markov decision process	204		
finite MDP	204		
Fisher face	214		
Fisher's linear discriminant analysis	214		
FIW	213		
FLD	214		
FORTRAN	119		
forward	58		
FTRL	144		
F measure	243		
		[I, J, K]	
		identification	214
		ILSVRC	2
		im2col	109
		inception	113
		Integrate-and-fire	69
		iterator	26
		Jacobian	127
		keyword arguments	28
		KL 情報量	243
		[L]	
		labeled faces in the wild	213
		lambda expression	25
		latent Dirichlet allocation	214
		layer	42
		LDA	214
		learning coefficient	78
		LeCun	100, 103
		LeNet	103
		leveldb	41
		LFW	213
		linear	61
		linear perceptron	77
		linear separability	87
		Link	58, 59
		lmdb	41
		local response normalization	120
		long recurrent convolutional networks	199
		long-term dependency	126
		loss function	102
		LRCN	199
		LRN	119, 120, 223
		LSTM	125
		[M]	
		mAP	163
		max pooling	122
		MCG	242
		MDL	79
		mean average precision	163
		MemNN	210
		memory neural networks	
			210
		METEOR	244
		Microsoft Common Objects in Context	193
		minimal description length	79
		model zoo	40
		MS COCO	193
		multiscale combinatorial grouping	242
		[N]	
		NAG	144
		Neocognitron	99
		Nesterov	144
		network-in-network	158
		neural image captioning	192
		neural machine translation	180
		neural Turing machines	186
		NMT	180
		NumPy	30
		N-gram	182
		[O]	
		objectness	164
		object hypothesis	164
		Occam's razor	79
		one hot vector	61
		Optimizer	58, 59
		optimizers	60
		overfitting	123
		over-learning	123
		[G]	
Gabor function	105		
gated recurrent unit	135		
GD	141		
GEMM	33, 109		
gemm	110		
generalized delta rule	79		
general matrix to matrix multiplication	109		
global average pooling	158		
GoogLeNet	103, 152		
gradient clip	127		
gradient exploding problem	81		
gradient vanishing problem	81		
GRU	135		
GRU	63		
		[H]	
Harr basis function	246		
Hessian matrix	12		
hidden Markov model	172		

	[P]	RTRL	92, 94, 125	<code>theano.tensor</code>	49
PCA	214	R-CNN	165	the selectivity—invariance dilemma	90
perceptron	76			topology	68
perplexity	243	[S]		<code>translate.py</code>	65
Plato’s problem	97	scale invariant feature transform	237	tuple	27
poverty of stimulus	97	scikit-learn	66	Turing machine	125
PPL	243	selective search	163		
principal component analysis	214	self feedback	69	[U]	
prosopagnosia	213	SGD	141	<code>udaptex</code>	48
pruning	124	<code>shape</code>	30	<code>ufunc</code>	38
PTB データセット	61	SIFT	115, 237	unsupervised learning	69
<code>pycaffe</code>	17	simple recurrent neural networks	91	<code>update</code>	59
Python	17	skip-gram	178	<code>updates</code>	54
	[Q]	slice	33		
Q 学習	207	SMT	181	[V]	
quick sort	21	sparse coding	82	ventral stream	69
	[R]	spatial pyramid pooling	156	verification	214
RBM	102	spiking neuron modeling	69	VGG	154
real time recurrent learning	94	SPP	156	VJ 法	217
RecNN	189	SPP-net	156		
rectified linear unit	75	statistical machine translation	181	[W]	
recurrent	69	stride	115	weight	77
recurrent neural networks	90	supervised learning	69	<code>WeightDecay</code>	61
recursive neural networks	125			weight decay	124
regions with CNN	165	[T]		WER	245
regularization	124	TD 学習	204	word error rate	245
ReLU	75	TD 誤差	206, 207	<code>word2vec</code>	63, 67
ResNet	159	temporal-difference learning	204	<code>word2vec</code>	178
restricted Boltzmann machine	102	ing	204	[X]	
retrograde amnesia	82	tensorboard	63	<code>xavier</code>	44
RMSprop	144, 149	TensorFlow	14, 63	XOR	88
RNN	125	term frequency	242		
rote learning	202	TF	242	[Z]	
ROUGE-L	244	TF-IDF	242	zero-padding	115
RPN	169	Theano	14, 48		
		<code>theano.function</code>	48, 49	[数字]	
		<code>theano.grad</code>	48, 51	2 階確率的勾配降下法	142
		<code>theano.nnet</code>	48	2 値型配列指定子	35
		<code>theano.scan</code>	48, 54	2GD	141
		<code>theano.shared</code>	48	2SGD	142

— 著者略歴 —

- 1980年 千葉県銚子市立銚子高等学校卒業
1985年 早稲田大学第一文学部心理学専修卒業
1988年 早稲田大学大学院博士前期課程修了（心理学専攻）
1994年 早稲田大学大学院博士後期課程単位取得満期退学（心理学専攻）
1994年 東京女子大学情報処理センター助手
現在に至る
2003年 博士（文学）（早稲田大学）

Pythonで体験する深層学習

— Caffe, Theano, Chainer, TensorFlow —

Practical Python Recipes of Deep Learning

— Caffe, Theano, Chainer and TensorFlow — © Shin-ichi Asakawa 2016

2016年8月18日 初版第1刷発行

★

検印省略

著者 あさ かわ しん いち
浅 川 伸 一
発行者 株式会社 コロナ社
代表者 牛来真也
印刷所 三美印刷株式会社

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社

CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844・電話 (03) 3941-3131 (代)

ホームページ <http://www.coronasha.co.jp>

ISBN 978-4-339-02851-5 (金) (製本: 愛千製本所)

Printed in Japan



本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられております。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めておりません。

落丁・乱丁本はお取替えいたします