

深層学習における画像識別率の学習パラメータ依存性評価

1160021 氏原 収悟(密山研究室)

1. はじめに

深層学習は、コンピュータの性能向上や新規アルゴリズムの発見により、急速に発展している。しかし、深層学習には膨大な演算量を必要とする。そこで、本研究では深層学習の画像識別処理を対象に調査する、また、必要な演算量を減らし、高い画像識別処理能力を持つニューラルネットワークの構成を検討するために、画像識別率に対するニューラルネットワークの学習パラメータの依存性を定量的に評価する。

2. 評価環境

本研究では、ニューラルネットワークの構成 (Network In Network [1] の構成) として Chainer1.5 [2] に用意されている nin モデルを用いる (図 1)。nin モデルの構成を入力から出力までの順に説明する。入力層から入力された画像データに対して、Mlpconv 層におけるフィルタ処理によって特徴を抽出する。続く最大プーリングによってある領域内から最大の値を取る。この Mlpconv 層と最大プーリングを交互に合計三回繰り返す。その後 Mlpconv 層を介して、大域平均プーリングで次元毎の平均の値を取り、最後に出力層で尤度を出力する。

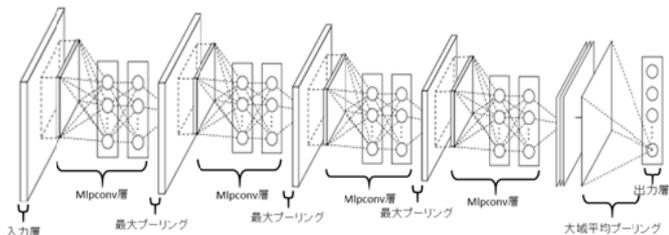


図 1 ニューラルネットワークの構成 [2]

学習に用いる画像 (学習画像) として Caltech101[3]を用いる。カテゴリ数を 100 に抑えるため、BACKGROUND_Google と Faces のカテゴリを除外する。図 2 に示す 10 枚の motorbike のテスト画像を用いて学習による識別率を評価する。



図 2 識別率確認のためのテスト画像

3. 実験結果

テスト画像の識別を行った結果、表 1 に示すように尤度が毎回異なる値となった。このため、各画像を 5 回ずつ識別させ、その 5 回の尤度の合計が最も大きい値のカテゴリを出力とした。

表 1 画像識別結果

	1回目	2回目	3回目	合計
motorbikes	93.8%	gramophone 71.4%	motorbikes 100.0%	motorbikes 361.7%
gramophone	6.2%	motorbikes 28.4%	- 0.0%	gramophone 138.0%
-	0.0%	butterfly 0.1%	- 0.0%	butterfly 0.1%

活性化関数 5 種類、学習回数 3 通り、学習画像数 3 通りのパラメータを変化させた時の識別率を図 3, 4, 5 に示す。学習回数が 200 回の場合、最も優れている活性化関数は leaky relu 関数であった。学習回数が 500 回の場合、学習画像数が 3,000 枚以上では leaky relu 関数と relu 関数であり、3,000 枚以下では clipped relu 関数であった。学習回数が 1,000 回の場合、学習画像数が 6,000 枚では clipped relu 関数を除く活性化関数は識別率 100%であったが、3,000 枚以下では clipped relu 関数が優れていた。

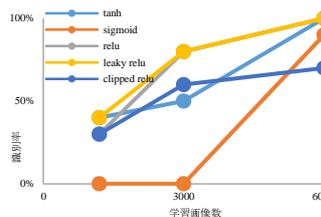


図 3 学習回数 200 回の識別率

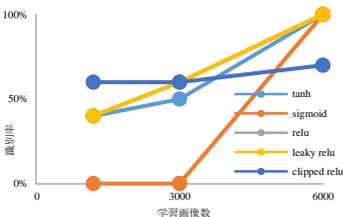


図 4 学習回数 500 回の識別率

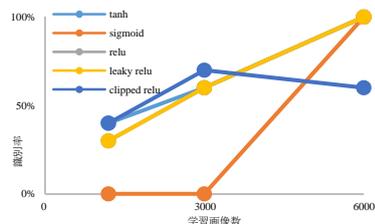


図 5 学習回数 1,000 回の識別率

4. 考察

本来、ニューラルネットワークにはランダムな関数は存在しないため、毎回異なる尤度を出力することはないはずである。しかし、本研究では 3 章で述べたように、尤度の値が識別を行うたびに变化した。その原因として 2 つ考えられる。第 1 に、学習時に使用される関数が認識を行うときにも使用されていたためだと考えられる。第 2 に、本研究で使用した nin モデルに画像を入力するときランダム関数を用いているため、このランダム関数が原因だと考えられる。

実験結果から、学習回数より学習画像数を増やすと識別率が高くなった。そのため、学習画像数は学習回数より識別率に対する依存性は高いと考えられる。

5. 結論

高い識別処理能力を持ち、演算量が少ないニューラルネットワークの構成を検討するために、ニューラルネットワークのパラメータとして、活性化関数、学習回数、学習画像数を変化させ、識別率を評価した。

今後の課題として、現在変化させていないパラメータとして、ニューラルネットワークの層構成とノード数を変化させ、識別率を確認する。

参考文献

[1] Min Lin, Qiang Chen, and Shuicheng Yan, "Network In Network," arXiv: 1312.4400, Mar. 2014.
 [2] PFN, "Chainer - A flexible framework of neural networks v1.6.0", Jan. 2016.
 [3] California Institute of technology, "Caltech101," April. 2006.