6S-01

深層学習モデルにおける学習プロセスの可視化方法の提案と評価

稲垣　遥太[†1]　　青山　幹雄[†1]

南山大学大学院　理工学研究科　ソフトウェア工学専攻[†1]

1　研究背景と課題

深層学習には，学習モデル(以下，モデル)が入力から出力を導出するプロセスが不透明であるというブラックボックス問題がある．そのため，モデルのプロセスの可視化が求められる．本稿では，モデルのパラメータや特徴マップの変化に着目し，その可視化を課題とする．

2　関連研究

機械学習の解釈可能性に関する議論[1]や，深層学習モデルのネットワークの学習を説明するための特徴量可視化(Feature Visualization)の研究がある[2]．

3　アプローチ

本研究では深層学習モデルとして CNN を対象とし，学習によって変化するモデルのパラメータに着目する．学習プロセスにおけるモデルのスナップショットを取得し，一連の学習プロセスデータを作成する．作成したデータを用いて，学習前後のパラメータの変化や特徴マップの変化を可視化することで，学習プロセスやニューラルネットワークのユニットの理解の支援を可能にする．

4　可視化方法

4.1 可視化プロセス

提案する可視化プロセスを示す．

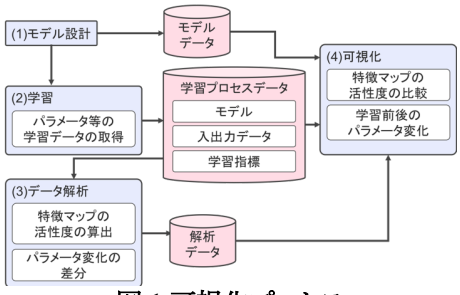


図 1 可視化プロセス

(1) モデルの設計

ニューラルネットワークのレイヤ構成やユニット数，学習率，バッチ数等を定義する．

(2) 学習

設計したモデルの学習を行う．バッチ処理毎にモデルのスナップショットを取得する．

(3) データ解析

学習プロセスデータを基に，可視化のために必要なデータを作成する．

(4) 可視化

作成した解析データ等を可視化し，モデルに対する知見を獲得する．

4.2 学習前後のパラメータの差分を可視化

学習プロセスにおける 2 点のスナップショットを選択し，各パラメータの差分を取得し，パラメータの変化を算出する．算出した値が 0 のときに白色，減少を赤色，増加を青色として，ヒートマップとして可視化する．

4.3 特徴マップの活性度の比較を可視化

モデルが正しく予測したデータを基に，各出力ラベルにおける特徴マップの活性度を比較する．これによって，フィルタが特徴量を抽出できているかどうか，フィルタがどのような役割を果たしているかを明らかにする．

各出力ラベルの対象とするフィルタの特徴マップの活性度を出力するプロセスを図 2 に示す．N 個の入力データに対して特徴マップを出力し，特徴マップのパラメータの総和の平均を求め，これを特徴マップの活性度とする．また，他の出力ラベルの特徴マップの活性度を比較するために，特徴マップの総和の平均を入力のパラメータの総和の平均で割った値を求める(式 1)．これにより，各入力による特徴マップの活性度のばらつきを小さくする．

$$(\text{比較のための特徴マップの活性度}) = \frac{(\text{特徴マップの活性度})}{(\text{入力パラメータの総和の平均})} \quad (式1)$$

各特徴マップの活性度の比較ができるように，出力ラベルを軸に，求めた値を棒グラフで可視化する．

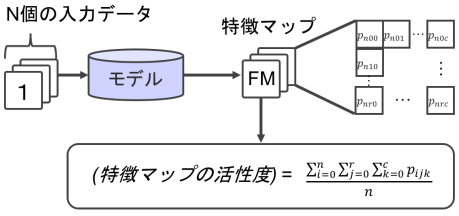


図 2 特徴マップの活性度の算出

5　プロトタイプ

5.1 アーキテクチャ

プロトタイプのアーキテクチャを示す．

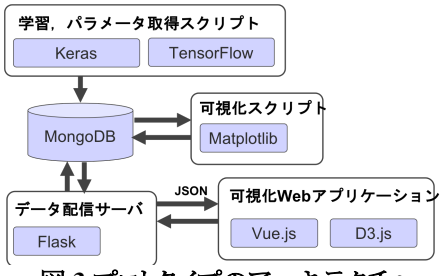


図 3 プロトタイプのアーキテクチャ

(1) 学習，パラメータ取得スクリプト

モデルの学習とバッチ処理毎にモデルのスナップショットを取得しデータベースを作成する．

(2) 可視化スクリプト

A Visualization Method of Learning Process for Deep Learning
†1 Yota Inagaki, Mikio Aoyama, Graduate School of Science and Engineering, Nanzan University.

Python の可視化ライブラリである Matplotlib を用いて，特徴マップの活性度の比較に関する可視化をする．

(3) 可視化アプリケーション

モデルのパラメータの値や，学習前後のパラメータの値の差分を可視化する．パラメータ等のデータを JSON で配信するデータ配信サーバと JavaScript のアプリケーションで構成する．

5.2 実行環境

本研究における実行環境を表 1 に示す．

表 1 実行環境

学習	Python 3.6	Keras 2.2.2	TensorFlow 1.11
データ配信	Python 3.6	Flask 1.0.2	
Web App	Vue.js 2.5.17	D3.js 4.2.2	
DB	MongoDB 4.0		

6 例題への適用と評価

手書き数字のデータセット，MNIST を対象として，以下のモデル構成を用いて学習プロセスデータを作成する．

モデルの構成例を示す．

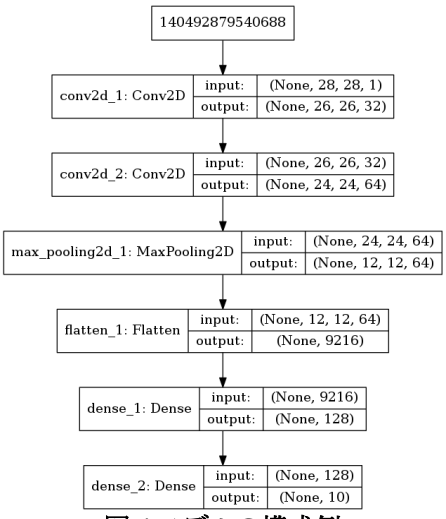


図 4 モデルの構成例

6.1 学習前後のパラメータの差分を可視化

1層目の畳み込み層のフィルタのパラメータと，学習前後の増減を可視化した(図 5)．パラメータの初期値が正のものは学習によって値が増加し，逆に負のものはパラメータが減少する傾向が確認された(図 6)．

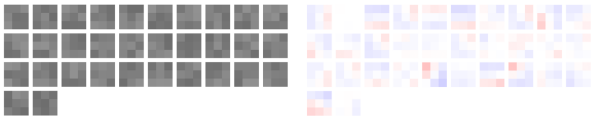


図 5 畳み込み層のフィルタのパラメータ(左)と学習前後の増減(右)

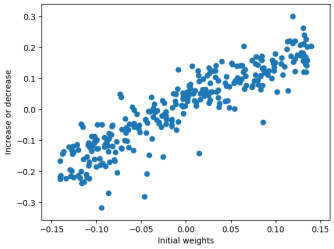
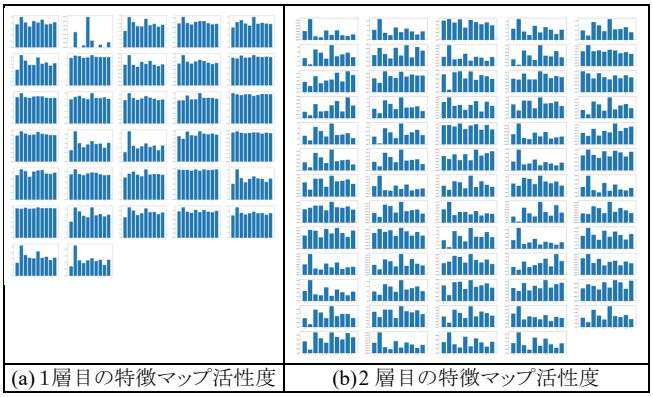


図 6 学習前後のパラメータ増減の散布図

6.2 特徴マップの活性度の比較を可視化

畳み込み層の 1 層目，2層目のトク著マップの活性度の比較を可視化した(図 7, **Error! Reference source not found.**)．横軸がラベル(0 から 9)，縦軸が活性度を示す．

1 層目を見ると，特徴マップの活性度がほとんど同じものが多いことがわかる．2層目では，1層目と比較して特徴マップの活性度に差が表れているものが増えており，特徴量をうまく抽出できている可能性を窺わせる．



(a) 1層目の特徴マップ活性度　(b)2 層目の特徴マップ活性度

図 7 畳込み層の特徴マップの活性度の比較

7 考察

(1) 学習前後のパラメータ増減の可視化

学習プロセスにおける 2 点のパラメータの重みの差分を計算し，可視化した．これにより，初期パラメータの重みが学習によるパラメータの増減に影響を与えることを明らかにした．

(2) 畳み込み層のフィルタの役割理解の支援

特徴マップの活性度を比較する可視化方法により，分類するラベルごとの活性度の違いを可視化した．これにより，活性度の違いから特徴マップを算出するフィルタの役割を理解するのに有用であると考える．

8 まとめ

本研究では，深層学習モデルの持つパラメータに着目し，学習プロセスの可視化方法を提案した．MNIST を用いた例題へと適用し，提案方法の有効性を評価した．

参考文献

[1] Z. C. Lipton. The Mythos of Model Interpretability, Jun. 2016, pp. 1-9, arXiv:1606.03490.

[2] C. Olah, et al., Feature Visualization, Nov. 2017, https://distill.pub/2017/feature-visualization/.