ディープラーニングを用いた画像認識におけるエポック数と識別精度の関係についての研究

國井 悠貴、金澤 遼、渡邉 洋美*
茨城県立日立第一高等学校 〒317-0063 茨城県日立市若葉町 3-15-1
* Corresponding author. *e-mail* address: watanabe-hiromi@※※※
(※※※ = mail.ibk.ed.jp)
(2022 年 8 月 8 日 受付 ; 2022 年 8 月 24 日 受理)

Abstract

学習用データの学習回数 (エポック数) を一定以上増加させると、過学習が発生する。過学習を回避する方法として、Dropout や正規化が挙げられる。本研究ではディープラーニングによる画像認識においてエポック数を増やしたときに過学習を起こす兆候とエポック数の打ち切りに適切な学習回数についてロジスティック関数を用いて考察できることが示唆された。

1 緒言

機械学習とは、人工知能が与えられたデータを繰り返し学習することによって、ルールやパターンを認識する仕組みのことである。機械学習の一種であるディープラーニングは AlexNet の開発によって注目されはじめた。AlexNet とは畳み込みニューラルネットワーク（CNN）を応用した仕組みの一つであり、サポートベクトルマシンなどの従来のアルゴリズムを採用した仕組みに比べ非常に高い精度を持っている。その AlexNet がきっかけとなりディープラーニングは近年より注目され、顔認証システムや自動運転技術などに広く活用されている。ディープラーニングにおいて認識精度は重要な要素である。そのため、多くのデータをより速く学習することが重要である。

ディープラーニングにおいて認識精度の向上には学習するデータ数を増やすことや、別の学習済みプログラムを転移させる転移学習を行うことが有効である。さらに学習するデータ数が一緒であっても学習用データの学習回数（エポック数）を増やすことも有効である。しかしながら、一定のエポック数までは認識精度は向上するが、ある一定の値を超えるとその精度は飽和してしまう。さらに、エポック数を増やすことにより、学習したデータに適応した学習ばかりが過剰に進んでしまい未知のデータに対して推定する性能が下がってしまう「過学習」が起こりやすくなる。また、エポック数を増やすことで実際に運用する際のコストの面にも大きな負担となる。したがって、十分な精度が得られる最小のエポック数を見極めることが重要である。

そこで本研究では、単純な認識技術でも行える筆跡鑑定に着目し、ディープラーニングを用いた画像認識において過学習が起きる兆候とエポック数の打ち切りをする学習回数について考察した。

2 本研究でおこなったディープラーニングの概要

2.1 畳み込みニューラルネットワーク [1)]

今回使用したプログラムは、ディープラーニングでの画像認識の手法で一般的になっている畳み込みニューラルネットワーク（CNN）と呼ばれる仕組みを使用した。

CNN は畳み込み層、プーリング層、全結合層と呼ばれる層からなり、プログラムに入力された画像データは、これらの層によってそれぞれ処理される。プログラムに入力された画像は、色を元に画像の各位置が数値化され、行列に変換される。行列となったデータは最初に畳み込み層を通る。畳み込み層では入力された行列と学習に必要なフィルターと呼ばれる行列を利用して畳み込み演算を行う（Fig.1）。

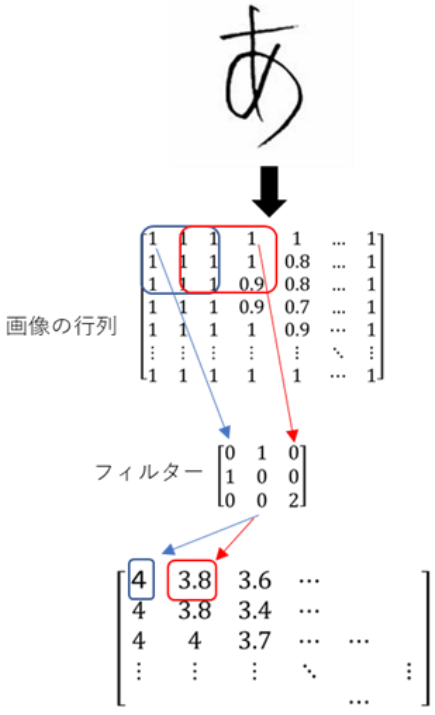

Fig.1：畳み込み層

畳み込み層を終えた行列はReLU関数

$$f(x)=\begin{cases}0, x<0\\ x, x\geqq 0\end{cases}$$

により負の数は0にされ、プーリング層へ入る。プーリング層は画像の解像度を低くする層である。プーリングにはいくつかの種類があるが、今回使用した最大値プーリング（max pooling）と呼ばれる方法は一般的に使われているものであり、小行列内の要素の最大値のみ残す処理が行われ、画像の位置ずれを起こしても影響のない仕組みになっている（Fig.2）。

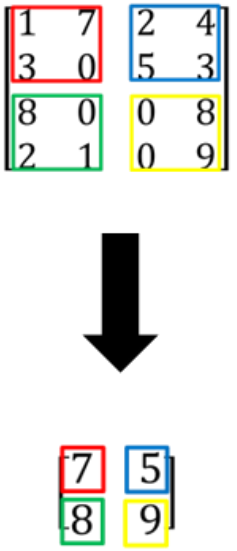

Fig.2：プーリング層

ディープラーニングでは畳み込み層とプーリング層を複数設定する。今回は8層設定した（Fig.3）。

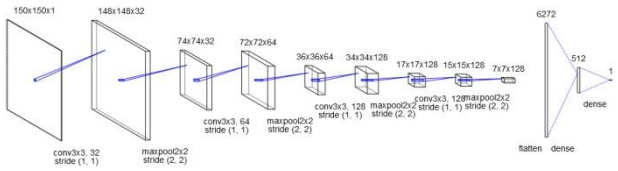

Fig.3：畳み込みとプーリングによる行列のサイズ

これらの層の処理が終了した後、行列データを一つの列に並び変え（一次元化）、全結合層と呼ばれる層で重み付き和とバイアスの総和を計算し、シグモイド関数を使用することによって0～1の数字を出力し、プログラムから予測したい画像である確率（予測値）を得ることができる。（Fig.4）。

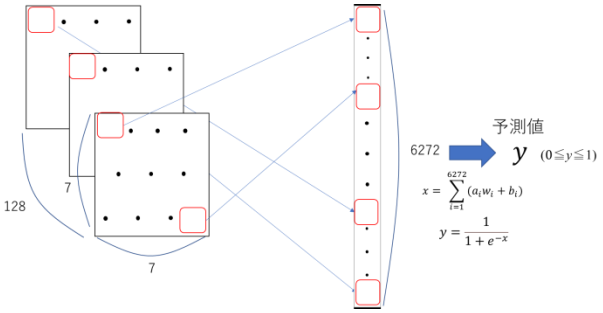

Fig.4：全結合層

最終的に出力された予測値と正解の値のズレを損失関数と呼び、損失関数を下げるように畳み込み層のフィルターの要素（重み）を調整することによってAIが学習したと考えることができる。

正解値 t は

$$t=\begin{cases}1\left(\text{識別対象以外の筆跡である場合}\right)\\ 0\left(\text{識別対象の筆跡である場合}\right)\end{cases}$$

とした。損失関数にもいくつかの種類があるが、本研究では交差エントロピー誤差

$$\mathrm{E}=-\sum_{k}t_k\log y_k$$

を使用した。

Eの最小値をとる重みの値は、第i層における重みをw_iとすると $\frac{\partial E}{\partial w_i}$ によって求めることができ、

$$\frac{\partial E}{\partial w_i}=\frac{\partial E}{\partial y}\frac{\partial y}{\partial w_i}$$

であるから、前層から重みの最適化ができる誤差逆伝播法を用いた。

3 CNNによる筆跡鑑定

3.1 収集データ

識別対象の筆跡と協力者7人の筆跡の訓練データ（学習に用いるためのデータ、トレーニングデータとも）とテストデータ（モデルの精度を評価するために識別させる用のデータ）の数はTable1のようにした。

Table1：収集したデータの数

訓練データ	
識別対象の筆跡	105個
識別対象以外の筆跡	15個×7人
テストデータ	
識別対象の筆跡	50個
識別対象以外の筆跡	10個×5人

画像データのノイズを無くすために、App Store で提供されているスキャンスマホアプリ「iScanner」を用いて画素数 150×150 の筆跡を収集した。筆跡は全てひらがなの「あ」1 文字に統一した。

Fig.5 : 収集した画像データ（左が識別対象の筆跡、右が識別対象以外の筆跡）

3.2 開発環境

プログラミング言語は機械学習関連のライブラリが豊富な Python(3.7.12)、開発環境は構築が簡単でありクラウド上で GPU の提供を行っている Google Colaboratory を使用した。また、フレームワークにはディープラーニング用のライブラリの一つである Keras[2] を使用した。

4 結果及び考察

エポック数が 1～100 におけるテストデータのうち、正確に識別できた割合（識別精度）は Fig.6 のようになった。エポック数 25 までは識別精度の向上が見られたが、それ以降は 82.3%～87.4%でエポック数を増やしても精度は頭打ちになった。

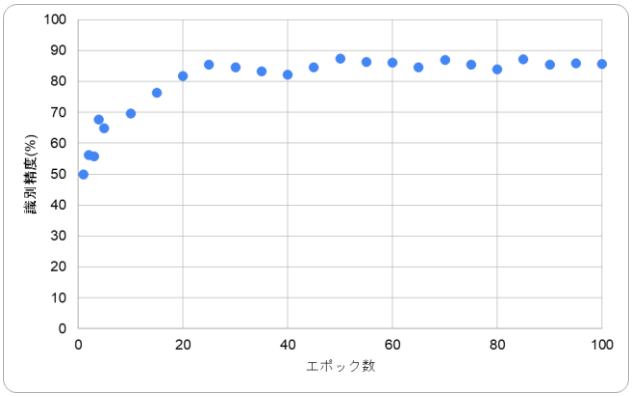

Fig.6 : エポック数と識別精度の関係

ここで、必要最低限のエポック数を求めるために、識別精度とエポック数の関係を対数プロットしてロジスティック回帰することができた（Fig.7）。

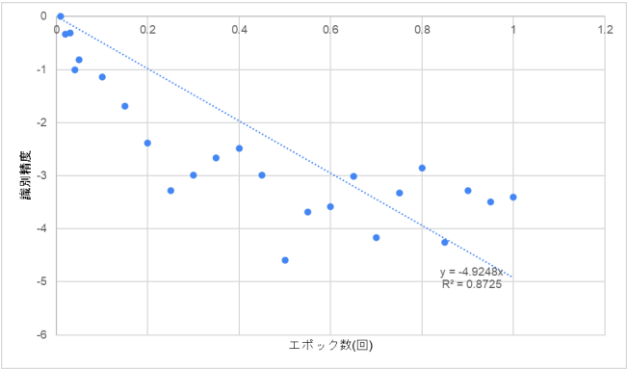

Fig.7 : エポック数と識別精度のロジスティック回帰

5 結論

エポック数を増やしたところ、識別精度の停滞が起こった。識別精度はロジスティック関数に従ったため、ロジスティック関数を利用して適切なエポック数での打ち切りを可能にすることが示唆された。

References

1) 斎藤 康毅、“ゼロから作る Deep Learning”、オライリージャパン (2016).
2) Francois Chollet、“Python と keras によるディープラーニング”、株式会社クイープ (2018).