

知能システム学特論レポート

(DL2 班) Caffe on Ubuntu

2015 年 7 月 27 日

1 報告者

15344203	有田 裕太
15344206	緒形 裕太
15344209	株丹 亮
12104125	宮本 和

2 進行状況

- ドロップアウトの理論について
- Caffe を用いてドロップアウトを試してみた
- アニメキャラクター画像認識時の中間層の出力
- データセットを増やした, 学習パラメータの調整

3 理論研究

3.1 ドロップアウト

ドロップアウトは, 多層ネットワークのユニットを確率的に選別して学習する方法である. ネットワークの学習過程と学習後の推論過程を, それぞれ以下のように修正する.

学習時は, 中間層の各層と入力層のユニットを決まった割合 p でランダムに選出し, それら以外を無効化, つまりそもそも存在しないかのように扱う. そして選出したユニットのみからなる仮のネットワークを, いつも通り最適化する. 学習終了後の推論時には, すべてのユニットを使って順伝播計算を行う. ただしドロップアウトで無効化の対象とした層のユニットはすべて一律に出力を p 倍する.

以上の手順からなるドロップアウトの狙いは, 学習時にネットワークの自由度を強制的に小さくし, 過適合を避けることである. またこうすることは, ドロップアウト時のネットワークを多数独立に訓練し, 推論時にそれらの結果を平均するのと同じ効果があると考えられている. 複数のネットワークの平均をとると推論の精度が一般に向上することが知られており, ドロップアウトはこれと同じ効果をより小さな計算コストで得ていると解釈できる.

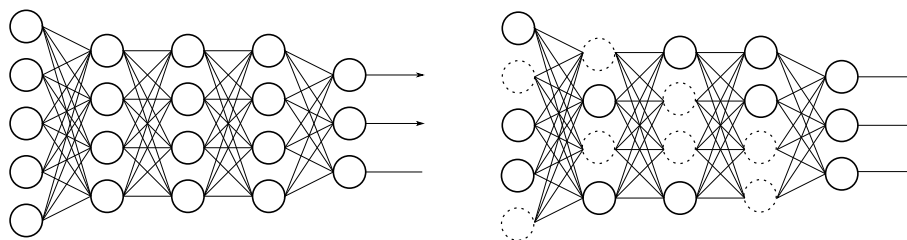
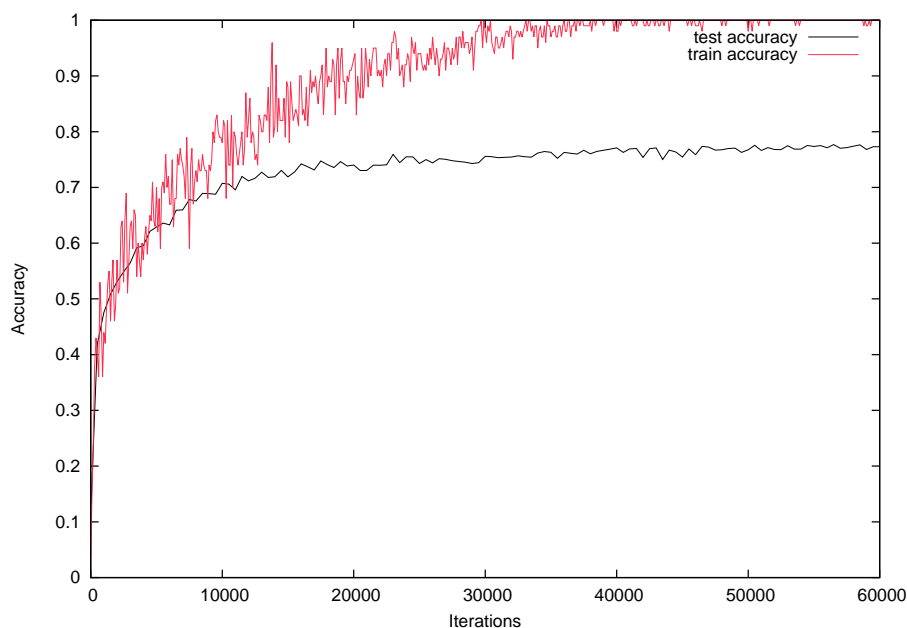
Fig.1 ドロップアウト ($p=0.5$ 程度)

Fig.2 過学習の傾向

4 プログラミング

4.1 Caffe のドロップアウトを試す

今まではアニメーション（ラブライブ！）を切り出して学習を行って、性能評価を行った．そこで実在する人物の識別を行うためのデータセットとして、韓国の女性アイドルグループ（少女時代）のメンバーの識別を試みた．アニメのキャラクター識別では学習の結果を解析したが良好な精度で学習ができ、過学習の傾向は見られなかった．しかし少女時代のメンバーのクラスタリングでは用いたデータセットがアニメキャラクターの識別と比較して髪の色、光の当たり方、顔の向き等の条件が一定になりづらく学習の精度も低い．このときの精度に関する結果を Fig. 2 に示す．ただしこの結果はアニメキャラクターの学習でも同様に用いた Cifar10 のモデルで行っている．Fig. 2 より過学習の傾向が見られ、訓練データに関しての精度は 1 に収束しているが、テストデータに関しては精度は 77% 程度である．

したがって前節で説明したドロップアウトを導入し、過学習の回避を試みる．ドロップアウトのユニットを

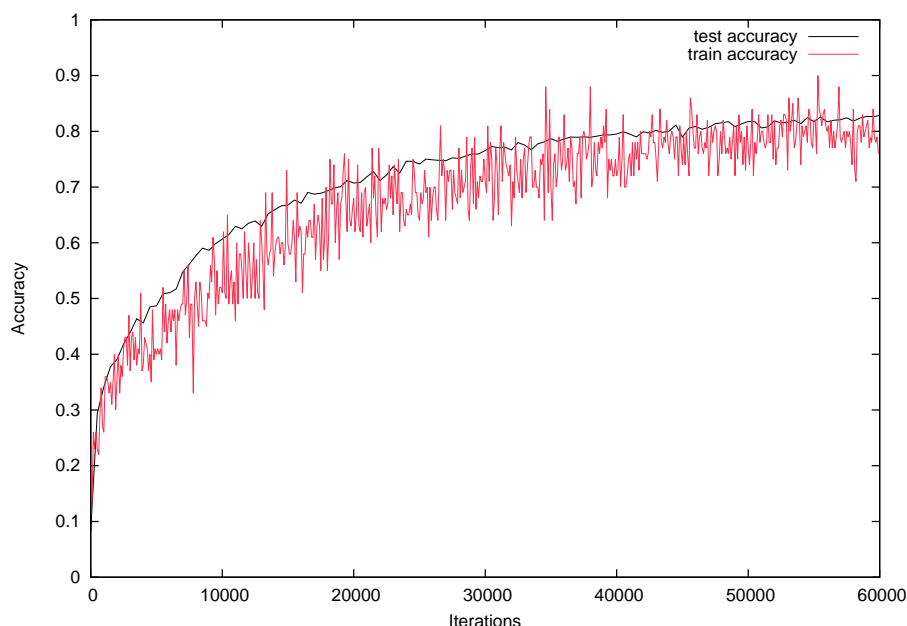


Fig.3 ドロップアウトを追加し、データセットを増やす工夫を行った結果

追加するだけでなく、過学習が起きる原因となる学習データの不足が考えられるため単純にデータセットを増やす以外の方法で改善を行った。その改善方法は入力として使うデータセットをランダムにクロップし入力データとして学習を行う方法と、入力データをランダムで左右反転させる方法である。これらの工夫によって得られた結果を Fig. 3 に示す。Fig. 3 より訓練データにおける精度とテストデータにおける精度の乖離がなくなり過学習を抑えることができています。また最終的なテストデータに関する精度は 83% と通常の Cifar10 のモデルを使用した場合よりも精度向上が見られた。

4.2 アニメキャラクター画像認識時の中間層出力

以前の発表で、猫のサンプル画像を用いて画像認識を行った際の中間層の出力を試みたが、今回は現在行っているアニメキャラクター（ラブライブ！）の画像認識の際の中間層を出力した。Fig.4 に識別する画像、Fig.5 に入力画像、Fig.6 に第一層目のフィルタ、Fig.7 に第一層目の出力結果を示す。

.prototex ファイルを用いてキャラクター識別を行うため、 32×32 のサイズに入力画像を変換する。フィルタを見てもわかるように、caffe における画像認識において色も重要な判断要素であると言える。前回の発表にもあったように、髪の色が似ているアニメキャラクターは誤認識が起こっていた。このような誤認識を防ぐためには、形状の微妙な特徴の違いに注目する工夫が必要である。

4.3 データセットの強化

アニメキャラクター（ラブライブ！）の識別で制度改善のためにデータセットを増やして再度学習を行った。また、学習の繰り返し回数も増やした。

前回と同様に従来のニューラルネットワークを用いて簡易的にクラスタリングを行った後、手動で修正を



Fig.4 識別する画像

行った。その結果、各キャラクターに 1000 枚程度データセットを追加した。したがって、合計で 9 名のキャラクターそれぞれに約 3000 枚の画像を用意した。また、学習を行うにあたり、学習の繰り返し回数を前回の 4000 回から 60000 回に変更した。

学習を行った結果を Fig.8 に示す。Test Accuracy は最終的に 95.04% となった。

今回作成した識別器を使った識別結果と前回の識別器を用いた識別結果を Fig.9 と Fig.10 に示す。誤認識が改善したことが分かる。



Fig.5 入力画像



Fig.6 一層目のフィルタ

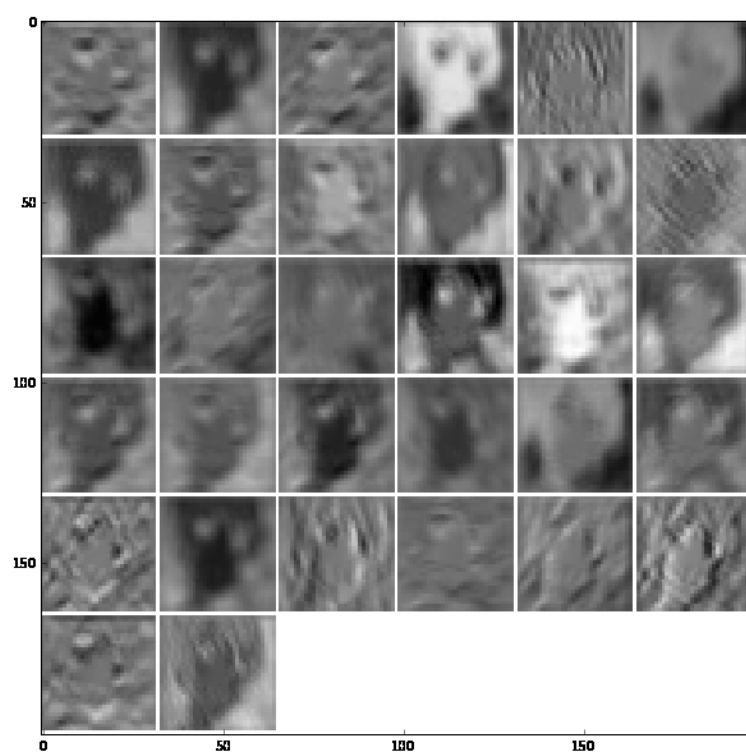


Fig.7 第一層目の出力

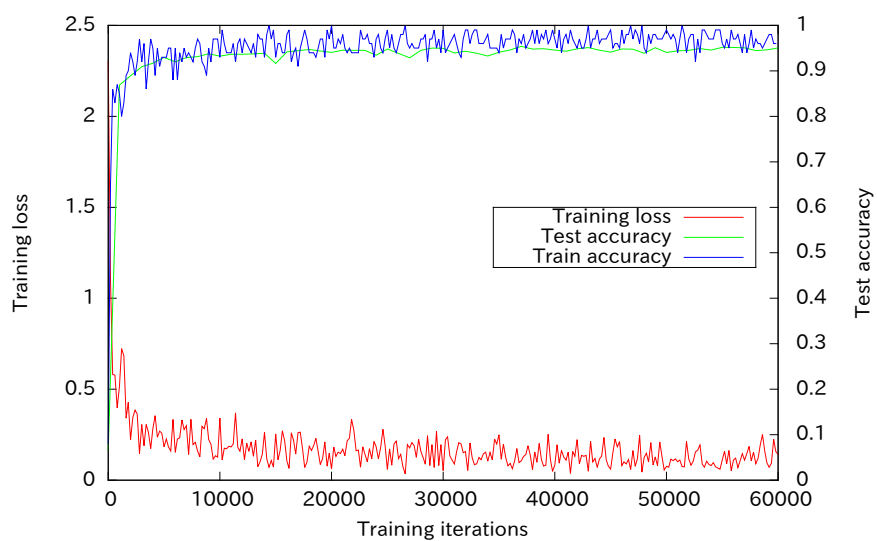


Fig.8 データセットを増やして学習を行った結果



Fig.9 今回作成した識別器を用いた識別結果 (Full Solver)



Fig.10 前回作成した識別器を用いた識別結果 (Quick Solver)