

ディープラーニングフレームワーク Caffeの分散処理に向けた一検討

一瀬 絢衣[†]

竹房 あつ子[‡]

中田 秀基[‡]

小口 正人[†]

[†]お茶の水女子大学

[‡]産業技術総合研究所

1. はじめに

近年インターネット上の情報量の増大やクラウドコンピューティングの普及により、ライフログの取得やそのデータの蓄積が容易になった。その結果監視カメラなどのセンサを用いたライフログの利用も普及してきている。ここで、一般家庭にサーバやストレージを設置して解析までの処理を行うことは困難であるが、映像をそのまま送信しクラウドで処理する場合、プライバシーや各種センサとクラウド間のネットワーク帯域の問題が生じてしまう。

本研究では、データ量の削減とプライバシーの保護を目的とし、ディープラーニングのフレームワークであるCaffe [1]を分散環境へ適用させることを考える。ここで、処理を分散する際のデータ量の削減は識別率を下げてしまうと考えられるため、本稿ではパラメータを変化させてデータ量を削減させた際の識別率を調査した。

2. ディープラーニング

ディープラーニングとは、人間の脳の神経回路がもつ仕組みを模した情報処理システムであるニューラルネットワークの中で、識別を行う中間層を多層化したものを用いた機械学習を指す。中間層が複数になっていることにより何段階かで認識を繰り返し、色や形状、全体像など複数の特徴を抽出してより正確な識別が可能となっている。

そのフレームワークであるCaffeは、Convolutional Architecture for Fast Feature Embeddingの略であり、Berkeley Vision and Learning Centerが中心となって進めている。C++で実装されており、GPUに対応していることから高速な処理が可能であること、学習済みネットワークモデルが提供されていて簡単に実験を行うことができるという特徴がある。

Caffeは、ディープラーニングの中でも畳み込みニューラルネットワークと呼ばれる構造になっている。畳み込みニューラルネットワークは主に画像認識に応用されるネットワークである。通常畳み込みとプーリングという画像処理の基本的な演算を行う層がペアで複数回繰り返されたあと、全結合層が配置される構造になっている。畳み込み層では入力画像に対しフィルタを適用し、フィルタをずらしながら各重なりで両者の積和計算を行うことによってフィルタが表す特徴的な濃淡構造を画像から抽出する。プーリング層では画像上で正方領域をとり、この中に含まれる画素値を使って一つの画素値を求め、畳み込み層で抽出された特徴の位置感度を若干低下させる。

3. 分散ディープラーニングフレームワーク

本研究では、図1で示すフレームワークを提案する。畳み込みニューラルネットワークに新たな層を定義してネットワークを分離し、クライアント側、クラウド側を用いた分散処理を実現する。分散処理を行うことにより、映像や画像そのものでなく特徴量をクラウドに送るためプライバシーが確保され、またデータ量を小さくしてから送ることによりネットワーク帯域を考慮した処理が可能になると考えられる。

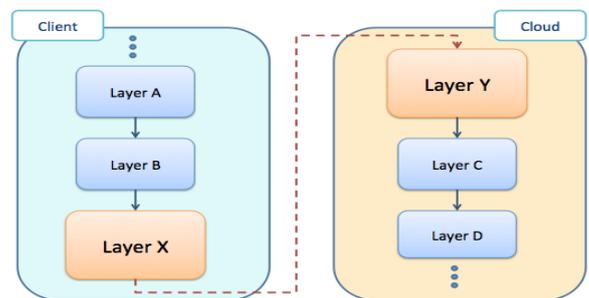


図1: 分散ディープラーニングフレームワーク

4. 基礎実験

4.1 データセット

実験では、CIFAR-10とSTL-10の2種類のデータセットを用いた。

CIFAR-10とは32×32画素の画像が10種類のクラスに分類されているデータセットであり、Caffeで学習済みネットワークモデルが提供されているデータセットの一つである [2]。このネットワークモデルにおいて各層で定義されているパラメータは表1のようにになっている。

Caffeにおいてデータはバッチサイズ、チャンネル数、画像の大きさの4次元配列でデータが格納されており、チャンネル数は直前の畳み込み層におけるフィルタ数と一致する。提供されているネットワークモデルのデフォルトの値では、始めは(100×3×32×32)byteのデータ量からconv1層を通り(100×32×32×32)byteへ、ストライドを2に定義しているpool1層を通り(100×32×16×16)byteへと変化しており、pool3層までは最初のデータ量よりも大きくなっていることがわかる (図2)。

STL-10は64×64画素の画像が10種類のクラスに分類されているデータセットである [3]。本研究ではCIFAR-10形式に変換し、CIFAR-10のネットワークモデルを使って識別を行った。

4.2 実験概要

畳み込み層におけるフィルタ数を変化させることによってクライアントからサーバへ送る際のデータ量を小さくすることを検討する。その際、データ量を小さくすると識別率が低くなってしまうことが考えられるため、フィルタ数を

Examination for the destributed processing of deep learning framework Caffe

[†] Ayae Ichinose, Masato Oguchi

[‡] Atsuko Takefusa, Hidemoto Nakada

Ochanomizu University (†)

National Institute of Advanced Industrial Science and Technology (AIST) (‡)

表 1: パラメータ

n : num_output	フィルタの数
p : pad	パディングの幅
k : kernel_size	フィルタの大きさ
s : stride	フィルタの適用位置の間隔

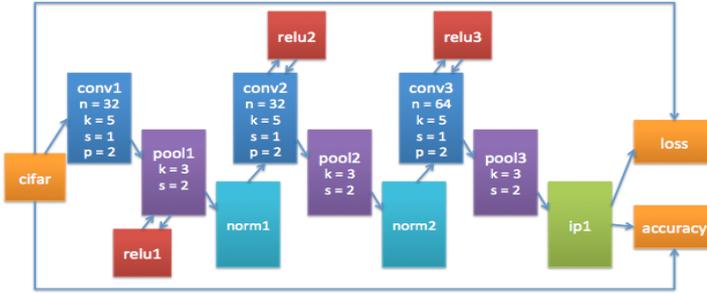


図 2: CIFAR-10 識別用ネットワークモデル

変化させて識別率がどう変化するかを調査するため、以下の2つの実験を行った。実験環境は表2に示す。

実験1 pool1層の直後にクラウドへ送ることを想定し、その直前のconv1層のフィルタ数を1からデフォルト値の32まで変化させ、pool1層直後のデータ量と識別率の相関を示した。conv2層、conv3層のフィルタ数はそれぞれデフォルト値の32、64に固定した。

実験2 pool2層の直後にクラウドへ送ることを想定し、その直前のconv2層のフィルタ数を1からデフォルト値の32まで変化させ、pool2層直後のデータ量と識別率の相関を示した。conv1層、conv3層のフィルタ数はそれぞれデフォルト値の32、64に固定した。

表 2: 実験環境

OS	Ubuntu 14.04LTS
CPU	Intel(R) Xeon(R) CPU W5590 @3.33GHz (8コア) × 2ソケット
Memory	8Gbyte

4.3 実験結果

実験1の結果を図3、実験2の結果を図4に示す。CIFAR-10、STL-10ともに少ないデータ量で識別率が収束していることが確認でき、フィルタ数を削減しても高い識別率を保てるということがわかった。識別率は、デフォルト値ではCIFAR-10は0.7811、STL-10は0.6187であった。

実験1では、データ量が512000byteでCIFAR-10は0.7335、STL-10は0.5748を計測した。これは画像をそのまま送信する場合と比較して3分の2のデータ量となっている。

実験2では、データ量が25600byteでCIFAR-10は0.7335、STL-10は0.5664を計測した。これは識別率を大幅に下げないままデータ量はさらに小さく6分の1となっているが、クライアント側でconv2層までの処理を行わなければならないため、クライアントで処理に時間がかかることも考えられる。

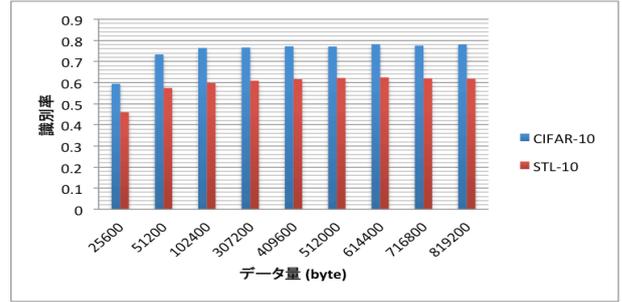


図 3: conv1層のフィルタ数を変化させた場合の識別率

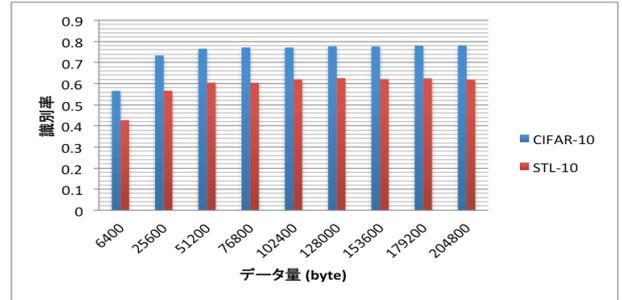


図 4: conv2層のフィルタ数を変化させた場合の識別率

5. まとめと今後の課題

本研究ではディープラーニングのフレームワークであるCaffeを分散環境へ適用させることにより、プライバシーやネットワーク帯域を考慮したセンサデータ解析処理を検討した。その際のデータ量の削減と識別率のトレードオフを探るため実験を行い、畳込み層におけるフィルタ数を変えることにより、識別率を大幅に下げることなくクライアントからクラウドへ送る際のデータサイズを小さくできることが確認できた。

今後の課題としては、Caffeを拡張して分散処理を可能にし、分散環境においてネットワーク帯域を考慮した処理時間と識別率の評価を行う。

参考文献

- [1] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R. B., Guadarrama, S. and Darrell, T.: Caffe: Convolutional Architecture for Fast Feature Embedding, *CoRR*, Vol. abs/1408.5093 (2014).
- [2] Alex, K., Nair, V. and Hinton, G.: The CIFAR-10 dataset. <https://www.cs.toronto.edu/~kriz/cifar.html> (accessed December 27, 2015).
- [3] Coates, A., Lee, H. and Ng, A. Y.: An Analysis of Single Layer Networks in Unsupervised Feature Learning. <https://cs.stanford.edu/~acoates/stl10/> (accessed December 27, 2015).