［ディープラーニング活用事例と使いこなしの勘所］ 画像処理分野

2 ラーメン画像からの全店舗識別

基般

土井賢治 | ヤフー（株）

深層学習による画像のクラス分類精度が人間を超えた

近年，深層学習によりコンピュータによる画像認識の精度が著しく向上しており，2015年のILSVRC☆1において，画像のクラス識別タスクで深層学習を利用したモデルが人間の認識精度を上回ったといわれている．

本稿では，深層学習の画像識別分野への適用事例として，筆者がラーメン二郎の画像から店舗を識別するモデルを作成した際の具体的な作業項目や勘所を紹介する．

ラーメン二郎とは関東圏を中心に約40店舗を展開する人気のラーメン店で，店舗ごとの盛り付けに個性的な特徴がある．常連の中には画像を見ただけで店舗を識別できる人もいる．

学習データの収集と分類

クローラー☆2を独自に開発し，Web上からラーメン二郎画像を収集した☆3．

犬や猫といった画像を収集する場合は，検索エンジン等でキーワード検索した画像を収集してそのまま学習に利用することもできるが，本事例では，ラーメン画像からラーメン二郎の店舗を識別したいため，画像収集時にどの店舗のラーメン画像なのかという点も考慮する必要がある．そこで，画像収集時にデータに付与されているコメントやタグ等をあわせて収集し，その内容をもとに店舗ごとに分類しながら画像を収集した．最終的にラーメン二郎40店舗を対象に総計で約8万枚程度のデータセットを構築した．

画像のクレンジング

収集した画像には店舗外観や自撮り等，ラーメンがまったく写っていないものも多数含まれている．ラーメンが写っている画像だけを対象にモデルの学習を行いたいため，学習を始める前に対象外または重複している画像の削除，分類誤りの確認等，データセットのクレンジングを行った．

最初に目視にてラーメンがまったく写っていない画像を削除し，その後，同一データを検出（画像データのMD5ハッシュ値を比較）し削除する．この際，同一データが異なる2つのクラス（店舗）にそれぞれ収集されている場合には，少なくとも片方が収集時の分類ミスであると考えられるため削除しておく必要がある．本事例においては，簡易的に複数クラスに重複して含まれる画像をすべて削除することで対処した．

また，画像の再圧縮やリサイズ処理により同一データではなくなった画像に対しても重複検出しておきたい．そのために，知覚ハッシュ関数☆4の一種であるphash☆5を全画像に対して計算し，ハッシュのハミング距離（どれだけ値が近いかの指標）をもとに画像の重複および分類ミスが疑われる画像を検出し削除した．

最後に，あらためて全画像を店舗ごとに目視確認し，これまでの工程で発見できていなかったラーメン以外の画像や分類ミス画像を削除した．

☆1 ILSVRC（世界的な画像認識コンペティション）：http://www.image-net.org/challenges/LSVRC/

☆2 Webを自動巡回して，文書や画像を収集するプログラム．

☆3 クローラー等でWebから画像データを収集する際には，対象サイトやサービスの利用規約を確認の上，収集先のサーバ等へ負荷をかけないよう注意．

☆4 知覚ハッシュ関数（perceptual hashing）：人間の感覚で似ている画像では近い値を生成するハッシュ関数．

☆5 phash：http://www.phash.org/

これらのクレンジング作業により，最終的に40クラス（店舗），約6万枚のデータセットとなった．

データセットの分割（学習，評価，テスト）

モデル学習後に識別精度を評価するために，データセットを事前に分割しておく．手法としては，学習データとは別に評価データを分けておくホールドアウト法を採用し，さらに学習済みモデルの汎化能力(未知のデータに対する識別能力）を確認するためテストデータも分割しておく．

40店舗の各店舗ごとの画像枚数については，1,000枚を下回るものが4店舗あり，最少で600枚から最大で3,000枚と偏りがあるが，評価用データとテストデータについては，それぞれ各店舗80枚の計3,200枚（全体の約5%）とし，残りを学習データとした．

モデルの学習

学習にあたっては，まず利用するモデルを決める必要がある．ユーザ自身でモデルを設計することも可能であるが，本事例では，2015年のILSVRCで優勝したResNetとこれをベースにしたSE-ResNeXt，そしてInception-V3を利用した．また，識別性能をより向上させる目的で，ファインチューニングやデータ拡張といった手法も実施した．これら手法は，深層学習フレームワークを活用することで自ら実装しなくても利用可能である（本事例ではApache MXNet[☆6]を利用）．

ファインチューニング

ファインチューニングとは，学習済みモデルのパラメータを初期値に利用することで学習済みモデルの汎化能力を引き継いでモデルを再学習する手法である．比較的少ない画像データでも良い精度が得られることが多く，試してみる価値は大いにある．

多くのフレームワークで，ILSVRCの題材であるImageNet（1,000クラス識別）データセットで学習済みのモデルが公開されており，本事例でもImageNetで学習済みのモデルを利用している．

データ拡張

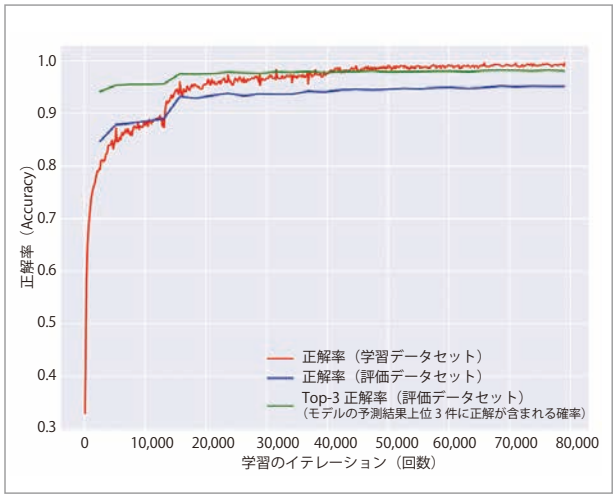
データ拡張とは，学習データに各種画像変換処理を行いデータ量を増やす手法である．本事例では，学習データに対して，トリミング，左右反転，色合い・輝度の増減，回転といった変換をランダムに行っている．フレームワークの機能を活用すると，画像を事前に水増しするのではなく，学習中にリアルタイムに画像を変換して利用することができる．

モデルアンサンブル

複数のモデルの識別結果の平均値を採用することで精度が向上する場合がある．1つのモデルが誤識別してもほかの複数のモデルが正解していれば，全体としては正しい識別結果となり識別性能が向上する．

学習の進捗

学習は，バッチサイズ（1イテレーションあたりにまとめて学習する画像枚数）を20とし，79,050イテレーション学習した（30エポック相当：データセット全体を30回分学習）．図-1に学習時の正解率の推移を示す．



■図-1　正解率（Accuracy）の推移

☆6　Apache MXNet : https://mxnet.apache.org/

学習したモデルの評価

クラス識別精度の主な評価指標としては，予測結果が正解している割合である正解率，正解と予測した件数のうち実際に正解している割合である適合率，正解のうち正解を正しく予測できたものの割合である再現率，という 3 つの尺度がある．

適合率と再現率はトレードオフの関係にあり，両者の調和平均をとった F- 値もよく利用される．本稿では，上記 4 つの尺度で評価を行った．

学習結果の各エポック時点のモデルごとに評価用データの正解率を確認し，正解率最大のモデルをテストデータで評価した（アンサンブルは本事例で作成した 3 種類のモデルの平均値を採用）．

結果は表 -1 のとおりである．

■表 -1　ラーメン二郎店舗識別モデルの識別精度（40 店舗の平均値）

モデル	正解率	適合率	再現率	F- 値
Inception-V3	0.9550	0.9557	0.9550	0.9550
ResNet（152 層）	0.9641	0.9651	0.9641	0.9641
SE-ResNeXt（50 層）	0.9709	0.9714	0.9709	0.9710
アンサンブル（上記 3 モデルの平均）	0.9772	0.9776	0.9772	0.9772

図 -2 のように，混同行列を確認することも重要である．

最も適合率が低い店舗（本事例では小岩店）でも 91.25%（73/80 枚）と想定を上回る結果となった．仮に極端に識別精度の低いクラスがあれば，データを追加したり，識別誤りとなった画像に共通の特徴がないか等を確認してみることも重要である．

さらなる識別精度の向上に向けて

本稿では，ラーメン二郎を題材に画像識別モデル作成の具体的な作業フローを解説した．

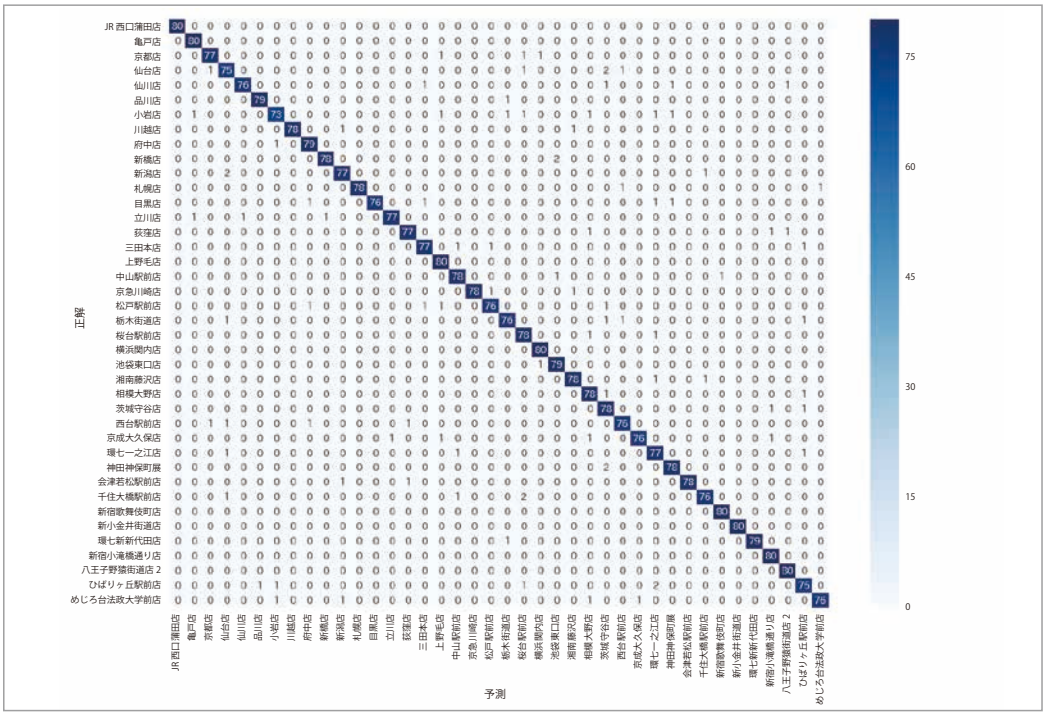
近年，ディープラーニングによる画像識別手法は日進月歩であり，今後もさらなる精度向上につながる手法が開発されることが期待される．

これら手法も踏まえ，画像分類に興味を持った読者が今後より良いモデルを作成するにあたり，本稿がその一助となれば幸いである．

（2018 年 8 月 1 日受付）

■土井賢治　knjcode@gmail.com

2007 年広島大学大学院工学研究科情報工学専攻修了．現在，ヤフー（株）勤務．データサイエンスによる自社サービスの改善業務に従事．



■図 -2　混同行列
（店舗ごとの予測数および正解数）