F-029

機械学習によるくずし字の文字切り分けに向けた検討
Examination for character separation of Kuzushiji by machine learning

村井 健† 市川 周一†
Ken Murai Shuichi Ichikawa

1. はじめに

近年，古文書をデジタルアーカイブ化して公開することが進められている．古文書は「くずし字」で書かれており，文字の区切りが不明確な形で表現されているため，一般の利用者は理解することができない．また，多くの古文書は手書きであり，個人の癖なども強いため，手書き文字認識技術による翻刻の完全自動化は現在も困難である [1].

くずし字の一例として，文字と文字が連なっている連綿体という字体が存在する．連綿体の特徴として，自動での切り分けが難しく，1 文字ずつのくずし字 (主にデータセット) は手作業で切り分けられている．本研究の目的は，手作業で行われている切り分け作業を機械学習で実現させることである．切り分けた領域（候補）が文字として認識可能であれば，その領域指定は適切であると考えられるので，切り分け自動化には前提として高精度の文字認識が必要であると考えられる．

ここでは，中間報告として

1. データセット作成
2. 作成したデータセットを用い，文字認識の精度評価

以上 2 項目を報告する．

2. データセットの作成

開発環境を表 1 に示す．Google Colabratory を使用しているのは，GPU が使用でき，Google Drive へのアクセスが容易なためである．言語には Python を用い，Pytorch による機械学習を行う．

表 1 開発環境

プラットフォーム	Google Colabratory
言語	Python 3.8
機械学習フレーム	Pytorch 1.7.1
画像処理フレーム	OpenCV-Python 4.1.2
使用する画像	人文学オープンデータ共同利用センターくずし字データセット[2] 4,328 文字種 1,086,326 文字の一部

切り分けをする画像・精度を比較できるようにするために，既存のデータセットから新たなデータセットを作成する．データセット作成プログラムでは，人文学オープンデータ共同利用センターの日本古典籍くずし字データセット [2] における作品コード(国文研書誌 ID)，Unicode に当てはまる 1 文字の画像を取り込み，画像処理を行い出力する．

今回は Kuzushiji – MNIST (KMNIST) [3] で使用するひらがな 10 文字種 (お・き・す・つ・な・は・ま・や・れ・を) を用いることにした．

指定した画像は Google Colaboratory で Google Drive 上に保存してある日本古典籍くずし字データセットから取り込む．日本古典籍くずし字データセットには，作品ごとに各画像のデータ(Unicode，Image(作品コード・ページ)，X・Y 座標，幅・高さ)がまとまった csv ファイルがある．そこから Unicode の一致した部分のみファイル名を指定し，画像の読み込みをする．その際，画像はグレースケールで読み込み，KMNIST の基準に合わせるように 28 × 28 にリサイズする．

その後，文字種ごとのフォルダ“label 0，label 1… ”を作成，同じ文字種で文字ごとに通し番号(0000,0001,…) をつけて jpg として出力を行った．

実際のプログラムの流れは図 1 に示し，そのときに出力された「お」の画像の一部を図 2 に示す．

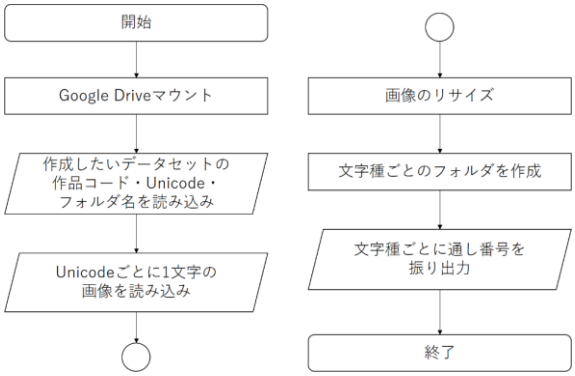

図 1 データセット作成プログラムの流れ

図 2 出力画像(「お」の場合)

3. くずし字認識プログラム

データセット作成プログラムで出力したデータセットを用いてくずし字認識をしていく．画像は Google Drive 上に保存してあるデータセットから読み込み，学習させる．プログラムは「現場で使える！PyTorch 開発入門 深層学習モデルの作成とアプリケーションへの実装」[4]を参考にした．

次に，図 3 に示す学習のネット構造について述べる．2 層の Convolution Neural Network (CNN) で 1 × 28 × 28 →

† 豊橋技術科学大学 Toyohashi University of Technology

$32 \times 8 \times 8$ → $64 \times 4 \times 4$ へ畳み込み，FlattenLayer で $conv\ size = 64 \times 4 \times 4 = 1024$ にする．各層では活性化関数 ReLU を使用している．その後，2 層の Multilayer perceptron (MLP) で 10 種類に分類して出力をしている．また，過学習の抑制をするために CNN の 2 層と 1 層目の MLP で Dropout 0.25 とし，イテレーション数は 16，エポック数は 20 とした．

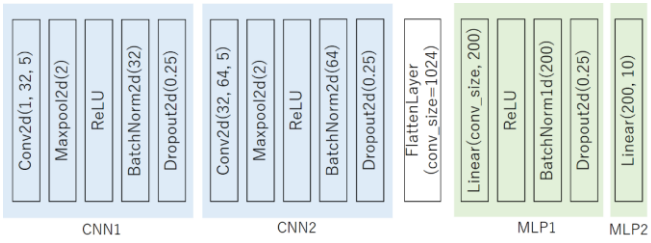

図 3 今回使用したくずし字認識プログラムのネット構造

4. 結果と今後の課題

3. データセット作成プログラムで出力した，ひらがな 10 文字種(お・き・す・つ・な・は・ま・や・れ・を)のデータセットを用い，4. くずし字認識プログラムで学習した結果を示す．トレーニング画像には「虚南留別志(うそなるべし)」と「鼎左秘録(ていさひろく)」のひらがな 10 文字種，テスト画像に「当世料理(とうせいりょうり)」のひらがな 10 もお字種を使用した．トレーニング画像は 1119 枚，テスト画像は 505 枚となっている．このときの学習結果を図 4 に示す．

この結果から，図 4 上のトレーニング損失 train_loss から学習回数が 10 回ほどで収束していることが分かった．また，図 4 下からトレーニングデータでの精度 train_acc はほぼ 100%となっている一方で，テストデータでの精度 test_acc は 70%程度となっていた．

今後の課題は以下の通りである．まず，テストデータでの認識精度が 70%程度と低いことがあげられる．原因としては，データセットの文字数が少ないこと，ネット構造の改良が必要なことが考えられる．また，現在はひらがな 10 文字種しか扱っていないため，文字種の追加が必要である．その後，文字の切り出しに向けたプログラム作成を進めていくことになる．

5. おわりに

本研究では，文字の切り分け精度を検証するための文字認識の精度と，文字認識に使用したデータセット作成プログラムについて示した．トレーニングデータでの認識精度が 100%であったのに対し，テストデータでの認識精度が 70%程度では文字の切り出しの評価をできるまでに至っていないことが分かった．今後，文字認識精度を向上させるとともに，文字種の追加をする必要がある．そして，文字の切り出しに向けたプログラム作成を進めていく．

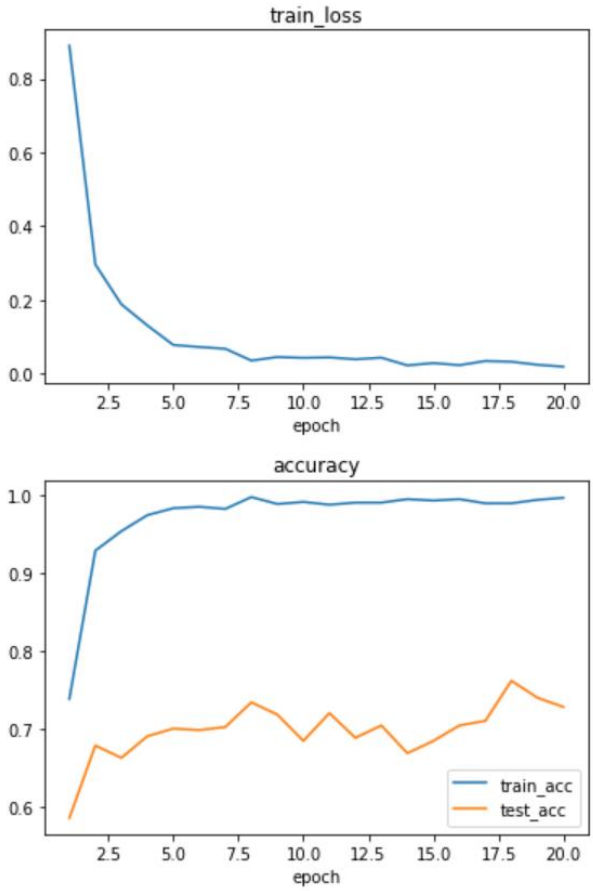

図 4 トレーニング損失 train_loss (上)と文字認識精度 accuracy (下)

謝辞

本研究の一部は JSPS 科研費 20K11733 によるものである．

参考文献

[1] 山本純子，大澤留次郎：”古典籍翻刻の省力化：くずし字を含む新方式 OCR 技術の開発”，情報管理，2015，58 巻，11 号，p. 819 - 827， DOI:10.1241/johokanri.58.819，(2015)
[2] 人文学オープンデータ共同利用センター：日本古典籍くずし字データセット，入手先<http://codh.rois.ac.jp/char-shape/book/> (参照 2021-6-11)
[3] 人文学オープンデータ共同利用センター：KMNIST データセット，入手先<http://codh.rois.ac.jp/kmnist/> (参照 2021-6-11)
[4] 杜世橋(著)：現場で使える！PyTorch 開発入門 深層学習モデルの作成とアプリケーションへの実装， pp.64-71，(2018)