Vol.2020-MPS-129 No.2
2020/7/27

近代書籍用文字認識のための低出現頻度文字クローリング

藤﨑菜々美[†1]　竹本有紀[†1]　石川由羽[†2]　髙田雅美[†1]　城和貴[†1]

概要：本稿では，近代書籍用文字認識のための学習データ用低出現頻度文字画像をクローリングによって収集する手法を提案する．近代書籍用文字認識の精度向上のために，学習データである文字画像の収集を支援する Web アプリケーションが開発されたが，文字種を満遍なく収集するには効率的ではない．それは，文字種によって出現頻度に大きな差があり，既存の手法では出現頻度が低い文字種の収集が難しいためである．そこで，本稿では近代書籍のクローリングを行い，求める低出現頻度の文字画像を探索し，収集する Web アプリケーションを開発する．そして，Web アプリケーションの動作結果と，クローリングで行われる黒画素率による文字画像の選定処理の結果を示し，提案した手法の有用性を確認する．

キーワード：Web アプリケーション，近代書籍，文字認識，低出現頻度文字種

Crawling Low Appearance Frequency Characters Images for Early-Modern Japanese Printed Character Recognition

NANAMI FUJISAKI[†1]　YUKI TAKEMOTO[†1]　YU ISHIKAWA[†2]
MASAMI TAKATA[†1]　KAZUKI JOE[†1]

1. はじめに

明治から昭和初期にかけて刊行された書籍は近代書籍と呼ばれている．この近代書籍は，国立国会図書館デジタルコレクション[1]上で著作権の保護期間が終了した書籍を対象に無償で公開されている．インターネットにアクセス可能であれば，誰でも書籍ページの閲覧とダウンロードができる．そのため，近代の研究において貴重な学術資料として利用されている．しかしながら，近代書籍はすべて画像データで公開されており，書籍の本文はテキストデータ化されていない．それゆえ，近代書籍を参照する際に，書籍の本文内の文字列検索を行うことができない．資料検索の利便性を向上させるために，近代書籍のテキストデータ化を目指している．

現在の一般的な書籍や文書画像は，光学文字認識（Optical Character Recognition，OCR）ソフトウェアで自動的にテキスト抽出が可能である．しかし，近代書籍文字に対しては既存の OCR では正確な認識が難しい．それは，近代書籍は活版印刷であり，文字フォントが規格化されていないためである．そのため，近代書籍は出版者によって書体が異なる．また，同じ出版者であっても出版された年代によっては文字の書体が異なっている．そこで，近代書籍に特化した文字認識の研究が行われている．

近代書籍の文字認識では，畳み込みニューラルネットワーク（Convolutional neural network，CNN）[2]が採用されている[3]．CNN の学習データには，JIS 第一水準の漢字 2,678 種類に対して，現在使われている印字用フォント 20 種類と，近代書籍文字フォント 6 出版者分が用いられており，認識率は約 97%となっている．近代書籍文字フォントは，近代書籍から抽出された文字画像である．実用化に向けてさらに認識率を向上させるためには，近代書籍文字フォントを増やさなければならない．それには，学習データとなる文字画像を，１つの文字種に対して複数の出版年代，出版者から収集する必要がある．また，文字種自体の数も充実させなければならない．JIS 第一水準と第二水準を合わせた 6,355 種のうち，出現頻度上位 3,000 種程度を学習データにすることができれば，認識精度を大幅に向上させることができると見込まれている．未収集である残り 500 種程度の文字種の獲得も，認識率向上には必要不可欠である．そこで，近代書籍の文字画像を効率良く収集するための Web アプリケーションの開発が行われている[4]．この Web アプリケーションによって，膨大な数の近代書籍に対し，手作業よりも短時間かつ正確に文字画像の収集を行うことができる．しかしながら，この Web アプリケーションによる手法では，文字種を網羅して収集するには効率的ではない．それは，書籍の文章中に出現する文字は文字種によって出現頻度の差があるためである．出現頻度が低い文字種は，学習データが必要とする数を満たすまでに莫大な時間がかかってしまうことが予想される．学習データ数が不十分である低出現頻度文字種を獲得するためには，低出現頻度文字種に着目し，それに特化した収集を行う必要がある．そこで，本稿では，国立国会図書館で公開されている近代書籍を探索し，求める文字種を発見するというクローリングによる収集手法を提案する．

†1 奈良女子大学
†2 滋賀大学

本稿の構成を示す．まず，2 章で既存の手法である近代書籍用文字認識の学習データ収集支援 Web アプリケーションと，文字種の出現頻度について述べる．3 章では，クローリングによる学習データ収集方法を提案する．4 章では開発したクローリングアプリケーションの動作結果と黒画素率による文字画像の選定処理の結果を示す．

2. 近代書籍用文字認識の学習データ収集

2.1 学習データ収集支援 Web アプリケーション

近代書籍用文字認識の認識率を向上させるためには，学習データとなる文字画像を，できるだけ多くの出版年代，出版者から収集する必要がある．そこで，文字収集の効率化を目的とした Web アプリケーションが提案されている．Web アプリケーションによる収集作業の手順を以下に示す．

① 認識を行う書籍を入力
② 書籍画像の前処理
③ 文字認識
④ ユーザによる認識結果のテキスト修正
⑤ 文字画像をデータベースに登録

初めに，ユーザは文字認識を行う書籍を入力する．次に，入力された書籍画像に対して，前処理を行う．前処理によって抽出された文字画像に対して文字認識を行う．そして，認識の結果であるテキストデータをユーザに表示する．ユーザは結果を確認し，誤認識があれば修正する．修正を終えたら，文字データベースに文字画像とそのテキストデータを登録する．

Web アプリケーションの開発が行われ，収集作業への活用が期待されたが，実際に Web アプリケーションを実行させた際に，問題点が 2 点生じている．1 つは，各機能を 1 つにまとめて実行すると動作が不安定になる点である．Web アプリケーションは機能ごとに分担して開発が進められたが，個々が作成した各機能には統一性が欠けていたことが確認されている．機能単位では動作する一方で，各機能を 1 つに結合した Web アプリケーションでは実用的なパフォーマンスが得られない．もう 1 つの問題は，学習データの文字データベースが不完全なものであったという点である．Web アプリケーションでは，明確なデータベースの定義が決められていないまま文字の登録が行われている．これらの問題点を解決するために，Web アプリケーションの再開発が行われている．各機能の設計と実装方法を見直し，Web アプリケーションの動作の安定性とユーザインターフェースが改良されるように開発する．

この Web アプリケーションによる手法では，ユーザが書籍を指定し，認識結果に誤認識があれば修正をする．そのため，指定した書籍内の文字を正確に収集するには有用なものであるといえる．しかし，この手法のみでは多種多様な学習データを収集することが困難である．それは，出現頻度が高い文字の学習データの数は充実していく一方で，出現頻度が低い文字の学習データの数は極端に増えないためである．文字の出現頻度には，文字種によってかなりの差があることを 2.2 節で述べる．

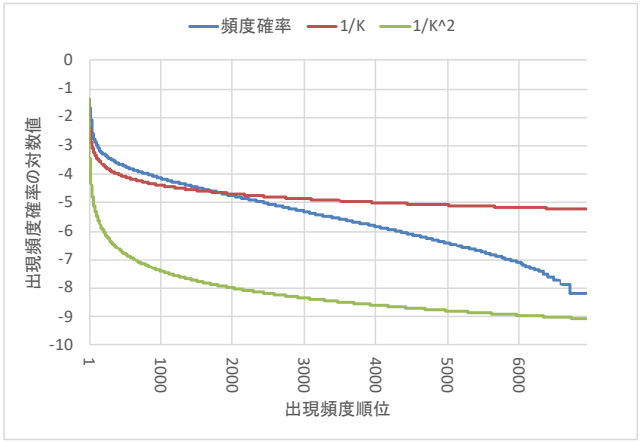

図 1 青空文庫の文字の頻度確率と $1/n, 1/n^2$ との比較

2.2 低出現頻度文字種

本節では，青空文庫[5]で公開されている書籍の文字の出現頻度確率とジップの法則[6]のグラフを比較することで，近代書籍における文字の出現頻度の差を確認する．ジップの法則とは，単語の出現頻度を多い順に並べ替えたとき，出現頻度が n 番目に高い単語が全体に占める割合は n 分の 1 に比例する法則である．ジップの法則を満たすと，出現頻度が上位のものに比べ，下位になるほど出現頻度が極端に少なく，発見が困難である．図 1 の青，赤，緑のグラフはそれぞれ，文字の出現頻度確率の散布図，n 分の 1 のグラフ，n^2 分の 1 のグラフである．n 分の 1 のグラフはジップの法則を表し，n^2 分の 1 のグラフはジップの法則よりさらに低い確率を表す．縦軸は頻度確率の対数値，横軸は出現頻度順位を示す．図 1 より，出現頻度 2000 位までは青と赤のグラフがほぼ同じ推移になっていることが確認できる．つまり，青空文庫の文字の出現頻度は，上位 2000 種においてジップの法則を満たしている．さらに，出現頻度が 2000 位より下位になると，n 分の 1 よりも低く，n^2 分の 1 に近づいていくことが確認できる．この結果より，ジップの法則に従っている出現頻度の上位 2000 種では，出現頻度上位の文字種に比べ，下位のほとんどの文字種の獲得がしにくいといえる．さらに，出現頻度が 2000 位以下の文字種の出現確率はジップの法則よりも低い．よって，出現頻度が 2000 位以下の文字種の獲得はさらに困難であることが予想できる．

3. 低出現頻度文字クローリング

学習データ数が不十分である低出現頻度の文字種を効率

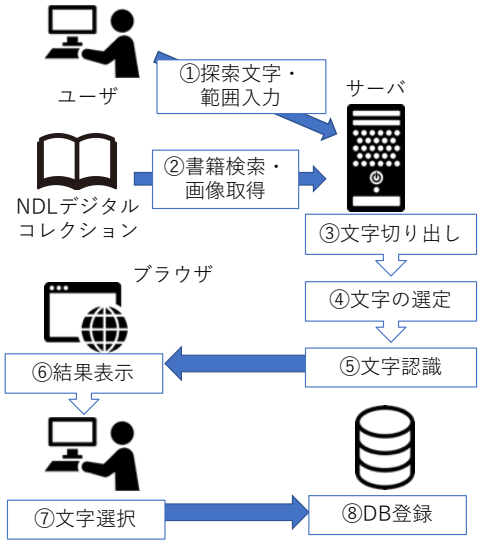

図 2　クローリングによる文字収集の手順

良く収集する方法として，クローリングによる収集を提案する．また，その手法を実装した Web アプリケーションを開発する．ここでのクローリングとは，国立国会図書館デジタルコレクション上の近代書籍を巡回し，求める文字画像を抽出することである．この開発では，Web アプリケーションフレームワークに Django[7], データベースソフトウェアに PostgreSQL[8]を使用する．図 2 はクローリングによる学習データ収集作業の手順である．ユーザは Web アプリケーションにアクセスし，探索する文字と，クローリングを行う範囲となる書籍の出版年と出版者を入力する．入力内容がサーバに送信されると，サーバは API を利用して検索条件を満たす書籍情報を取得する．その後，該当する書籍のページ画像を国立国会図書館デジタルコレクションからダウンロードする．そして，ダウンロードした書籍画像から文字の切り出しを行う．1 字ずつ切り出された文字画像には，文字認識を行う前に黒画素率による文字画像の選定処理を行う．出現頻度の高い文字種は，ひらがなや簡単な漢字が多く，画数が少ない．そのため，文字画像中の黒画素の割合が少ない文字が比較的よく見られる．一方で，出現頻度の低い文字種は，画数が多い複雑な漢字がよく見られる．ゆえに，黒画素の割合が高いと考えられる．これを利用して，切り出した各文字画像中の画素を調べ，黒画素率が低い文字画像は切り捨てる．低出現頻度文字である可能性が高い文字画像だけを文字認識の対象とする．選ばれた文字画像は文字認識用のサーバに送信され，文字認識が実行される．認識が終了すると，確信度が高い文字の JIS コードが複数個返される．サーバは，文字認識の結果が探索対象の文字に当てはまる文字画像をブラウザに送信する．受け取ったブラウザは，入力される文字ごとに探索結果の文字画像を Web ページに表示する．ユーザは表示された文字画像を確認して，正しく認識されている文字画像を選択する．ユーザによる選択が終わると，選択された文字画像

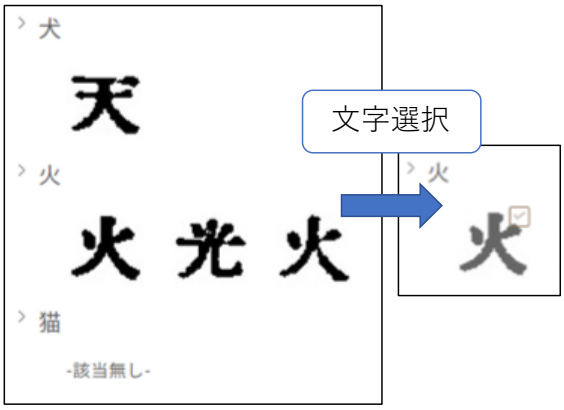

図 3　認識結果の表示と文字画像の選択

がサーバに送信される．その文字画像と認識結果の JIS コードは文字データベースに登録される．

4. 動作結果

本章では，クローリングによる収集手法の有用性を実証する．そのために，Web アプリケーションのフロントエンドにおける動作結果を示す．また，クローリングの処理の 1 つである黒画素率による文字選定処理が低出現頻度文字種に有効であることを示すために，文字画像の黒画素率を集計した結果を述べる．動作確認は，Django に付属している開発用サーバを立ち上げ，ローカル環境下で行う．

フロントエンドでのユーザ操作による Web アプリケーションの動作結果を示す. 動作の流れは以下の通りである. 初めに, ユーザは入力ページにアクセスし, 探索する文字, 出版年，出版者をフォームに入力する．入力後に，送信ボタンをクリックすると，入力ページから探索結果を表示するページに遷移する．結果表示ページには，処理の進捗状況を示す進捗バーが表示される．図 3 の左図は結果表示ページ上の探索結果の表示部分である．クローリングはページ単位で処理されるため，発見された文字画像から順にブラウザ上に表示される．サーバとブラウザ間の通信はバックグラウンドで行われる．そのため，クローリングの処理中でも，ユーザは認識結果の選択操作を行うことが可能である．結果が表示され次第，ユーザは表示された文字画像を確認する．ユーザは求める文字の文字画像があれば，その文字画像をクリックする．クリックすると，図 3 の右図のように，文字画像自体の透明度が下がり，画像右上にチェックボックスが表示される．これにより，選択済みであることが視覚的に差別化される．指定した範囲のクローリングがすべて終了すると，結果を表示するページに送信ボタンが出現する．また，探索の結果，求める文字が発見されなかった場合は”該当無し”と表示される．ユーザはすべての選択作業を終えたら，送信ボタンをクリックする．

文字画像の黒画素率を取得し，集計した結果を述べる．今回用いる入力データは，現在使われているフォントであ

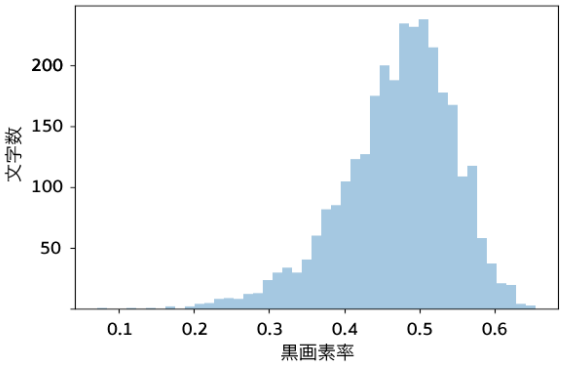

図 4　明朝体の文字画像の黒画素率の分布

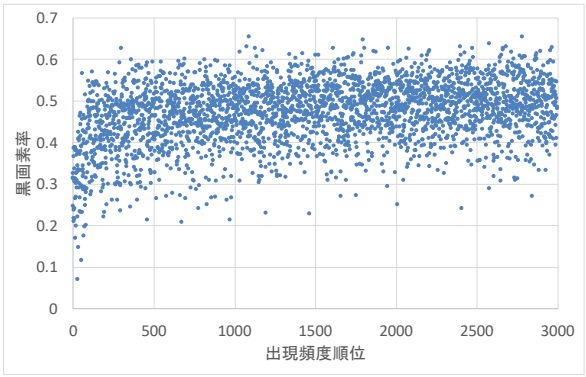

図 5　文字の出現頻度順位ごとの黒画素率の分布

る HG 明朝 E の文字画像である．また，文字種は JIS 第一水準のひらがなと漢字 3011 字である．各文字画像の黒画素率を取得し，集計した結果から分布を作成する．図 4 は作成した文字画像の黒画素率のヒストグラムである．横軸は文字画像の黒画素率，縦軸は文字の個数である．図 4 より，明朝体における黒画素率ごとの文字の度数分布は正規分布に近いことが確認できる．近代書籍文字においても同様の分布になることが見込まれる．そのため，クローリングにおける文字の選定処理では，図 4 のような度数分布から黒画素率の閾値を決定する．ここで，2.2 節で述べた，青空文庫の書籍で用いられる文字の出現頻度ごとの黒画素率の分布を調査する．図 5 は，文字の出現頻度ごとの HG 明朝 E の黒画素率の分布である．図 5 より，出現頻度が上位の文字画像は黒画素率の低い文字が多いことが確認できる．また，出現頻度上位 1000 文字程度までは，黒画素率の分布範囲が上がっている一方で，出現頻度順位が 1000 字より下位の文字は，黒画素率の分布はほぼ一定である．結果より，出現頻度上位 1000 字程度は，黒画素率によって求めることが可能であるといえる．よって，文字認識処理前の黒画素率による文字の選定処理は有効であることが確認できる．このとき，フォントによっては文字の形状や細さが異なるため，文字画像の黒画素率の分布はフォントによって変化することが予想される．そのため，十分な文字画像の数が集まり次第，出版者・出版年代別で黒画素率の分布を作成する．それによって，より最適な閾値を求めることができると考えられる．

5. まとめ

本稿では，近代書籍の低出現頻度文字の獲得のために，クローリングによる文字収集手法を提案する．この手法では，国立国会図書館デジタルコレクションで公開されている近代書籍のページ画像を取得してクローリングを行い，文字を探索する．さらに，ユーザによって，探索結果の文字が求める文字であるか照合される．そして，ユーザによって選択された文字がデータベースに登録される．この手法を実装した Web アプリケーションを開発し，動作テストを行う．動作の結果，入力された文字に対して，一連のクローリングの処理が実行され，正確な学習データとして文字をデータベースに登録するという目的が果たされている．従来の Web アプリケーションよりも求める文字の収集が効率的であることが実証されている．クローリングでは，収集した文字画像の黒画素率による認識対象の文字画像を選定する処理を実装する．現在のフォントを用いて黒画素率の分布を作成した結果，出現頻度が高い文字は黒画素率が低いことが確認できる．よって，低出現頻度文字を探索するクローリングにおいて，黒画素率による文字の選別処理は有効であることが証明される．今後はアプリケーションの実稼働に向けて，文字切り出しと文字認識処理の精度向上を目指す．

謝辞　本研究は MEXT 科研費 JP20H04483 の助成を受けたものです．

参考文献

[1] “国立国会図書館デジタルコレクション”. http://dl.ndl.go.jp/ , (参照 2020-06-26).
[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proc. of the IEEE, pages 2278–2324, (1998).
[3] Suzuka Yasunami, Norie Koiso, Yuki Takemoto, Yu Ishikawa, Masami Takata and Kazuki Joe. “Applying CNNs to Early-Modern Printed Japanese Character Recognition”. The 2019 International Conference on Parallel and Distributed Processing Techniques and Applications(PDPTA2019), pp.189-195 (2019).
[4] Kazumi Kosaka, Taeka Awazu, Yu Ishikawa, Masami Takata and Kazuki Joe. “An Effective and Interactive Training Data Collection Method for Early-Modern Japanese Printed Character Recognition”. The 2015 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA2015), Vol. II , pp.276-282 (2015).
[5] “青空文庫”. https://github.com/aozorabunko/aozorabunko, (参照 2020-06-26).
[6] Zipf,G. K.(1949): Human Behavior & The Principle of Least Effort, An Introduction to Human Ecology, Addison-Wesley Press Inc.
[7] “The Web framework for perfectionists with deadlines | Django”. https://www.djangoproject.com , (参照 2020-06-26).
[8] “PostgreSQL”. https://www.postgresql.org , (参照 2020-06-26).