Kaggle Kuzushiji Recognition(くずし字認識) 手法概要

土井 賢治

2019.11.11 日本文化とAIシンポジウム2019

1. 自己紹介

名前：土井 賢治
所属：ヤフー株式会社

knjcode

・2018.4〜 ヤフー@福岡 天神オフィス
・機械学習、ディープラーニングを活用して、
　ヤフオク!やPayPayフリマのサービス改善に取り組む
・趣味でラーメン二郎の店舗識別bot(@jirou_deep)を作ってます

2. コンペ参加のきっかけ

・2019.7.11に「Google Developers ML Summit」に登壇した際、
同じ場所で開催されていた「Solve with AI」での KuroNet の発表を
見て興味を持つ（web記事にてKaggleコンペ開催を知る）

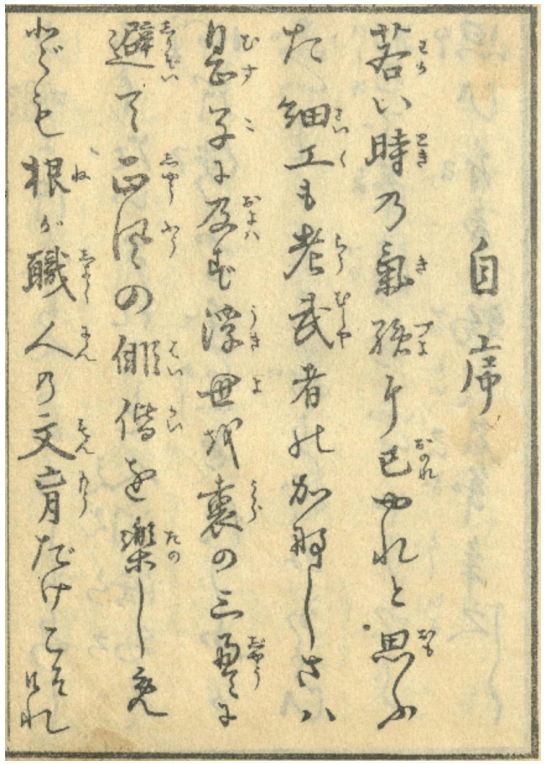
・技術的な興味と文化的にも貢献できると思い参加
  ・数年間の機械学習、ディープラーニングの経験（特に画像分野を多く扱ってきた）を試したい
  ・OCRシステムを作ってみたい
  ・くずし字を読める人の少なさや翻刻されていない大量の古典籍が存在していることを知った

・Kaggleは以前に一度だけ同僚と参加（サポート的役割）
今回のコンペが始めての本格&ソロ参加

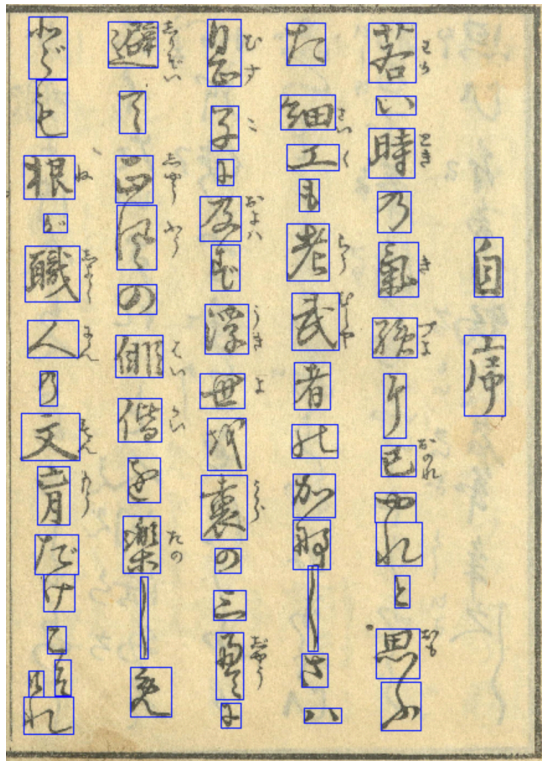
3. くずし字認識コンペ概要

入力画像の文字の位置(中心座標)および文字種を認識するタスク

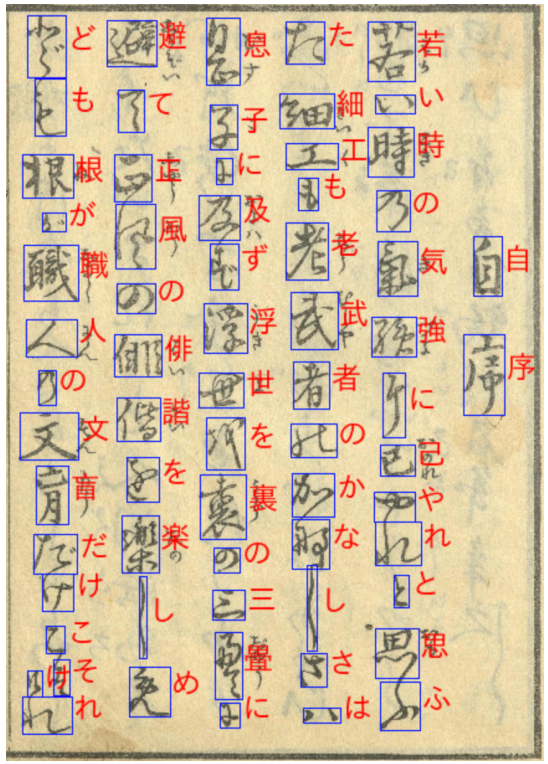
入力画像 (input)



文字の位置の認識
(detection)



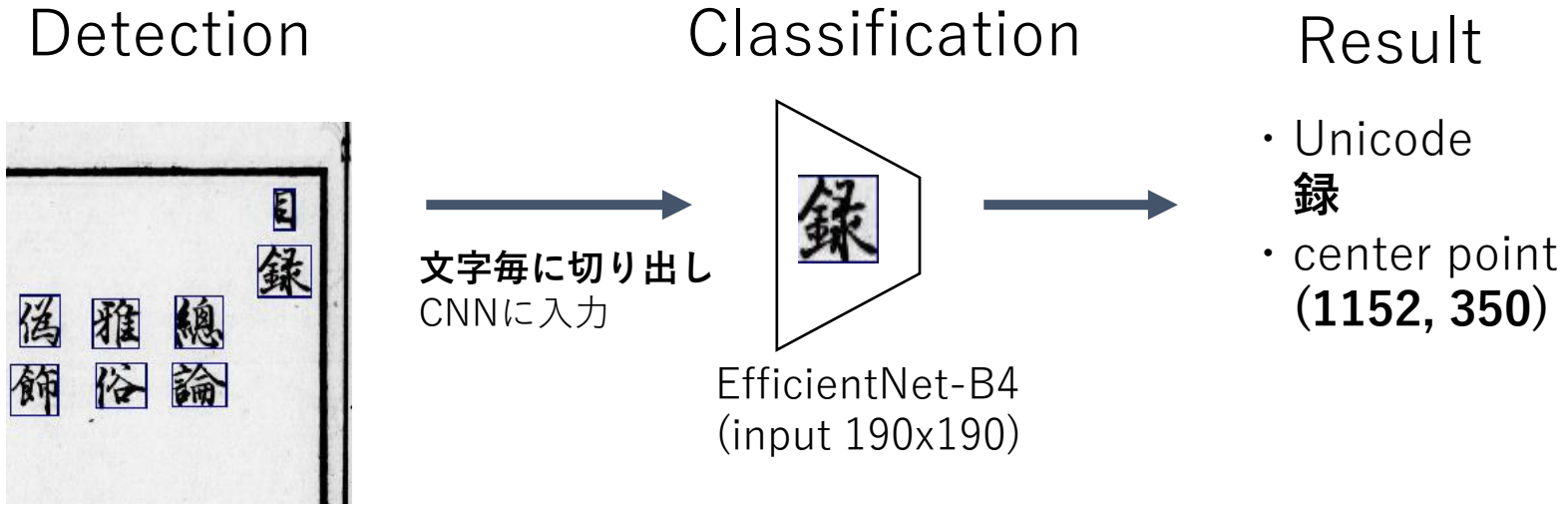
文字の種類の識別
(classification)



コンペにおいては、detectionとclassificationを同時に行う手法もあれば個別に行う手法もあった

4. 私のアプローチ (概要)

・文字の位置判定と文字種の識別を独立に実施(2-stage approach)
  ・Detection: (Faster R-CNN)
    画像中の物体検出用の手法を用いて文字の**位置のみ**を判定
  ・Classification: 畳込みニューラルネット(CNN)で文字種を判定



最終的に2つのDetectionモデルを
アンサンブルした結果を利用

最終的に5つの分類モデルを
アンサンブルした結果を利用

※：アンサンブルとは、複数のモデルの予測結果の多数決や平均値を算出し、予測精度を改善する手法

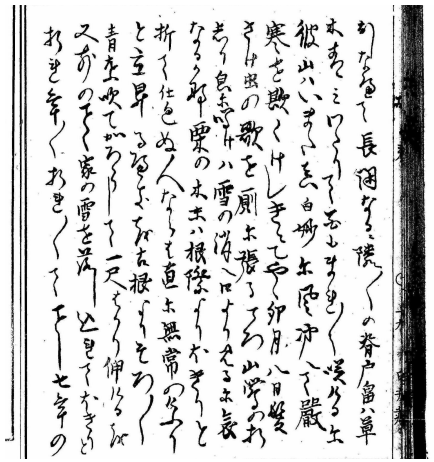
5.1. アルゴリズムの工夫点

・**データ拡張 (Random Grayscale)**

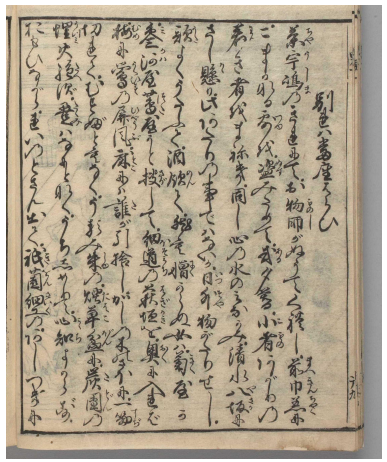
学習データには**フルカラーの写真**もあれば**モノクロ写真**もある

特定の文字、例えば「録」がモノクロ写真に**「のみ」**出現する場合
ニューラルネットワークが**文字種とモノクロ写真という特徴を**
関連付けて学習してしまう可能性がある(のをモノクロ化して防ぐ)

（フルカラー写真中の「録」を誤認識しやすくなる）



モノクロの学習データ



フルカラーの学習データ

学習時にランダムに
モノクロ化して利用

モノクロ化した学習データ

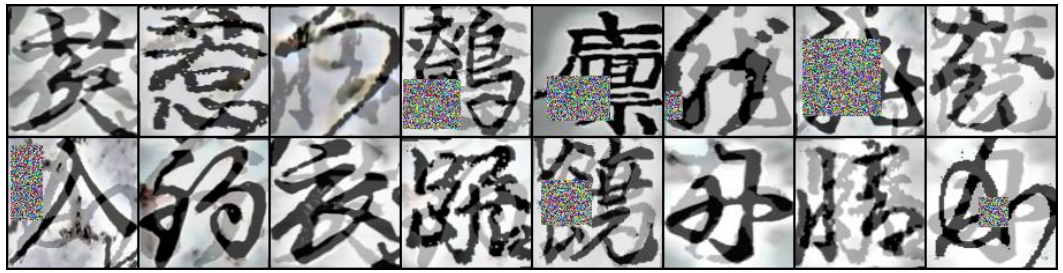
5.2. アルゴリズムの工夫点

- データ拡張 (mixup, ICAP, Random Erasing)
  - mixup, ICAP

    複数の文字を組み合わせ(2文字のブレンドや4文字の切り貼り)し学習データとして利用
  - Random Erasing

    画像の一部を欠損(ノイズで埋める)させて学習

人間には非常に読みづらい(読めない)がモデルの識別精度は向上



mixup(2文字をブレンド) + Random Erasig の例

ICAP(4文字を切り貼りして1枚の画像に)の例

※：ICAPはRICAPというデータ拡張手法を参考に独自に実装した手法です

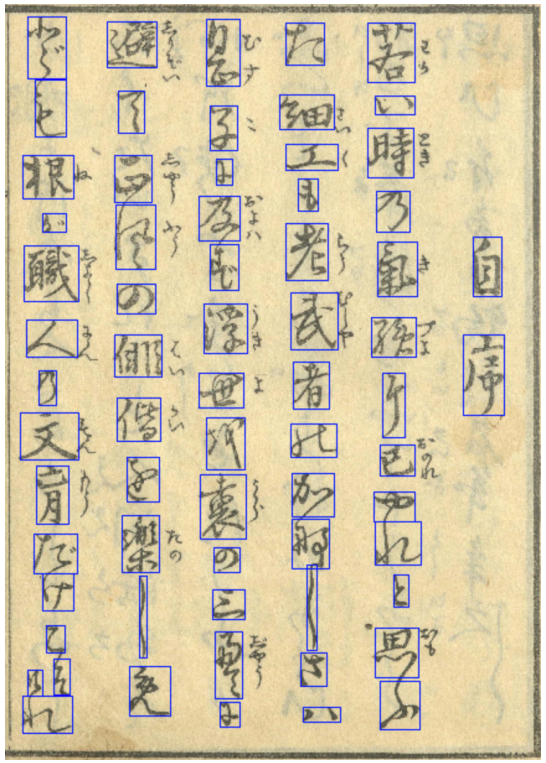
5.3. アルゴリズムの工夫点

・FalsePositive Predictor（誤検出予測器）
　取り組みの当初、ルビの取り扱いに苦戦

くずし字に付いている
ルビは認識対象外

ルビと同じ形であって
も、サイズや位置に
よっては認識対象

モデルの予測結果
青枠：正解　赤枠：不正解

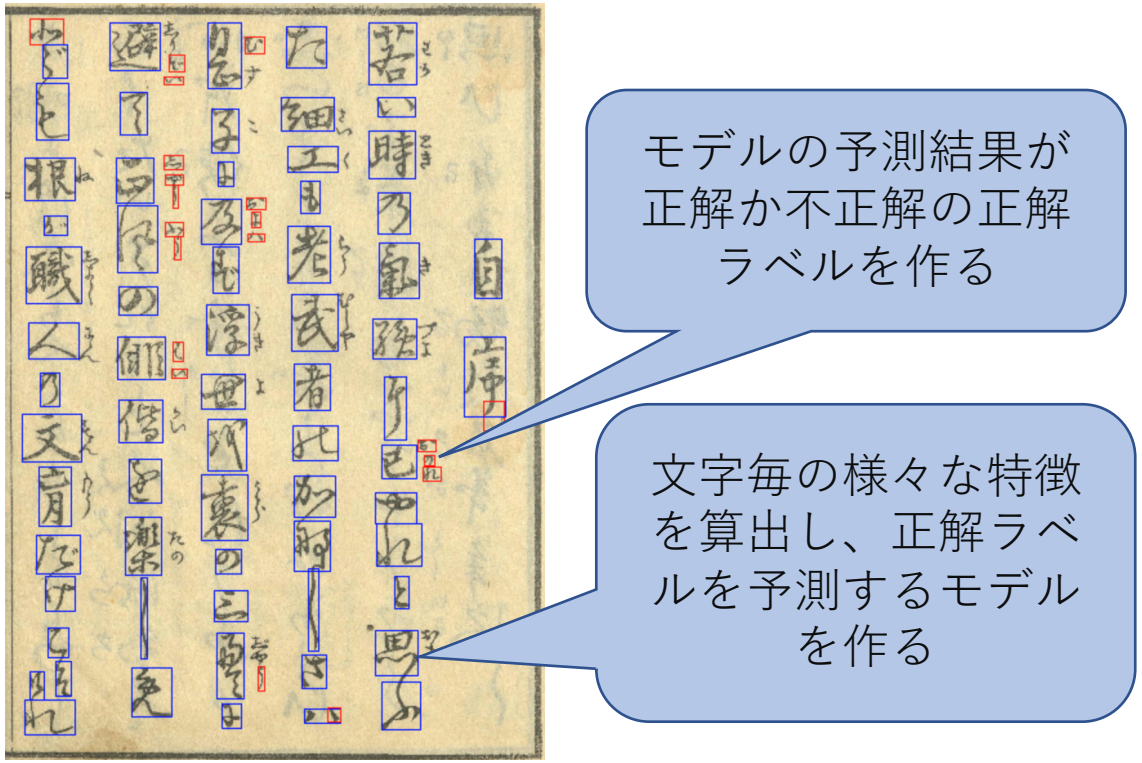


正解データ

5.4. アルゴリズムの工夫点

・学習データの予測については正解データで答えあわせできる

文字の位置・サイズ、他の文字との幾何学的な関係(サイズの比率、隣の文字までの距離、角度、etc)をもとに、対象の文字が誤検出か否か判定するモデルを学習

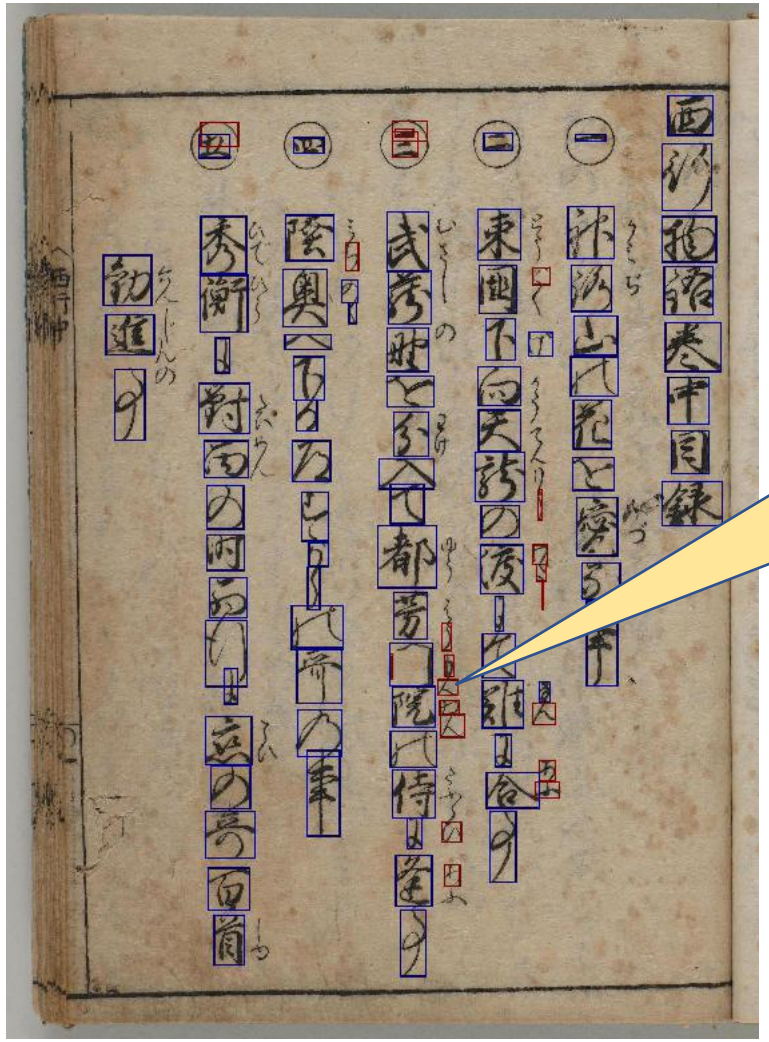


モデルの予測結果が正解か不正解の正解ラベルを作る

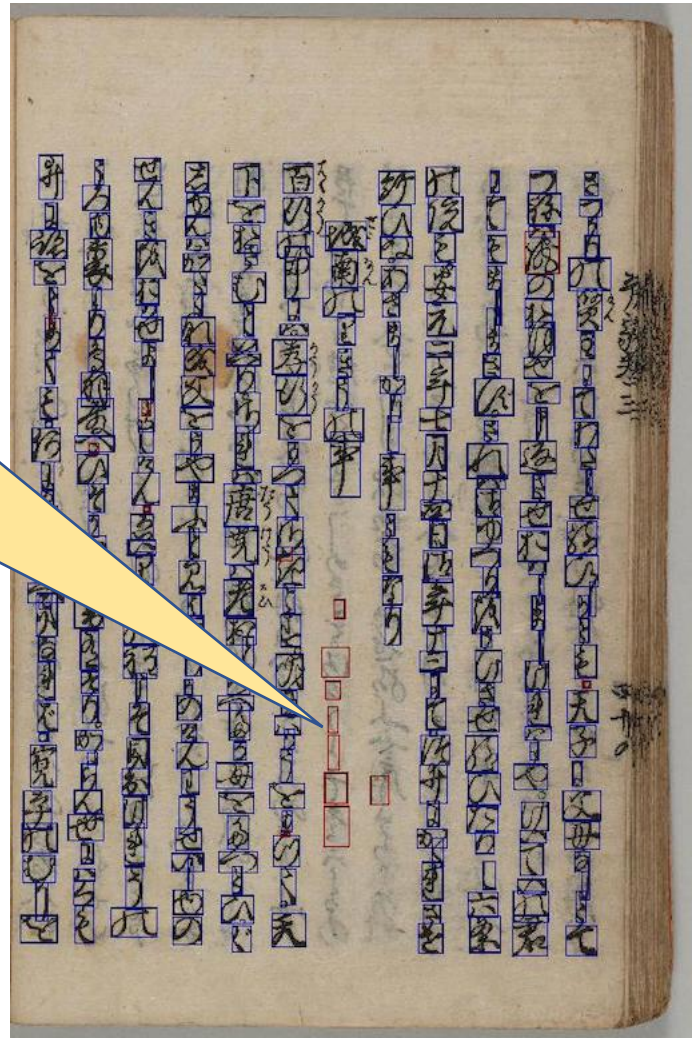
文字毎の様々な特徴を算出し、正解ラベルを予測するモデルを作る

5.5. 誤検出予測の適用例

・**赤枠**が誤検出と予測された文字



ルビの検出だけでなく、
紙の裏側から透けた
文字の検出等にも役立った
(誤検出予測でスコアが
約0.002～0.004向上)



6. 気づきと学び

・モデルの精度がスコアとして定量的に可視化されランキングされる
　・コンペならではの面白さ
　・上位を目指して様々なアイデアを試すきっかけに
・データ拡張の効果
　・人間にとって識別しづらい手法も意外に効果がある
・誤検出予測
　・ルビ検出についてはhand-craftedなアルゴリズムも作ったが、
　　機械学習で作成したモデルのほうが精度が良かった

・OCRシステムは精度が99%超えても実用的でなかったりするが、
　読める人が少なく、まだ読まれていない古典籍が大量にある
　くずし字という領域に機械学習・ディープラーニングを適用するという
　今回の取り組みやコンペは非常に意義あるものと感じました

7. 手法の詳細について

下記URLにて詳細な解法を紹介しています

手法の概要 (Kaggle Discussion)
3rd place solution overview: 2-stage + FalsePositive Predictor
https://www.kaggle.com/c/kuzushiji-recognition/discussion/113049

解法詳細、ソースコードおよび学習済みモデル
https://github.com/knjcode/kaggle-kuzushiji-recognition-2019

8. 参考文献

- AIで日本史研究者やマニアが狂喜乱舞する「くずし字」の翻訳ツールが開発
  https://pc.watch.impress.co.jp/docs/news/1195499.html
- mixup: Beyond Empirical Risk Minimization
  https://arxiv.org/abs/1710.09412
- Random Erasing Data Augmentation
  https://arxiv.org/abs/1708.04896
- Data Augmentation using Random Image Cropping and Patching for Deep CNNs
  https://arxiv.org/abs/1811.09030