

# Kuzushiji Recognition Challenge

## 8th place solution

日本文化とAIシンポジウム2019 (2019/11/11)

半谷 (t-hanya)

# 自己紹介

- t-hanya / 半谷 敏規 (はんや としのり)
- エンジニア (機械学習 / ソフトウェア)
  - 認識モデル開発 / アプリの設計・開発 / 処理の高速化
- 機械学習歴：
  - 4年前に自宅GPUマシン購入、色々論文を読んで実装して実験
  - 3年前から業務として映像認識技術の開発と実問題への適用
- Kaggle歴：
  - 今回初参加



@hanyacch



t-hanya

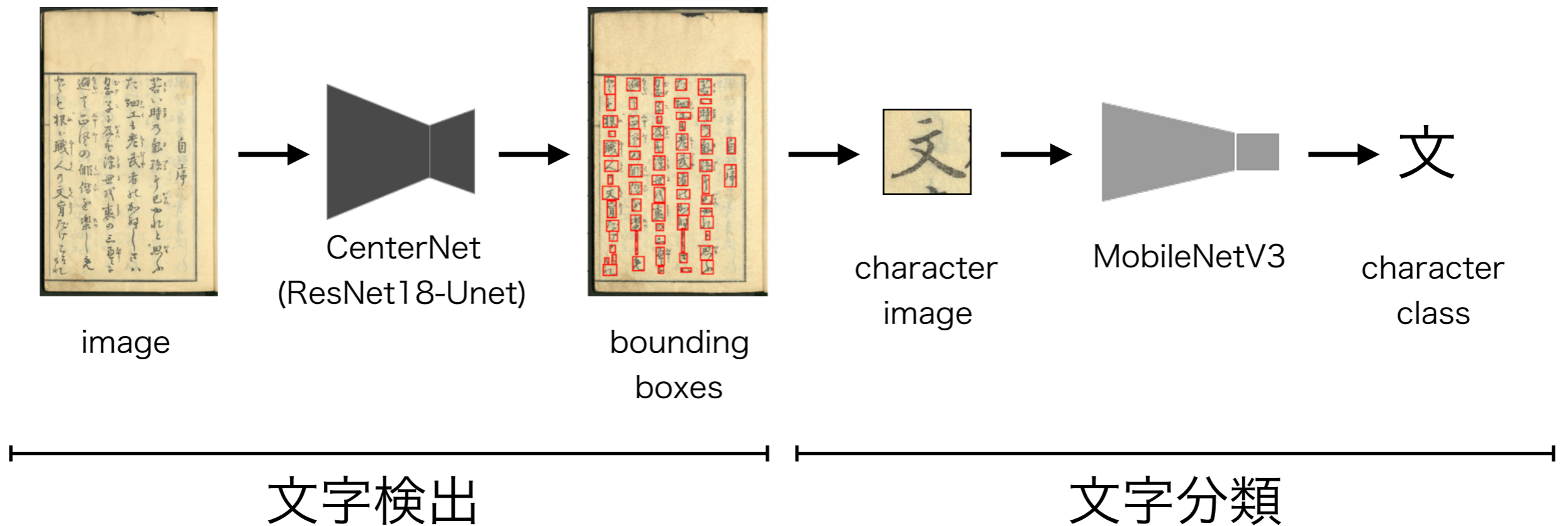
[codecrafthouse.jp](http://codecrafthouse.jp)

# くずし字コンペへの参加

- コンペの目的への共感
  - 膨大な古典籍へのアクセスと日本の文化遺産の継承
  - 読める人が減っている (0.01%) 中での機械学習への期待
- 腕試しがしたい
  - 同じ認識タスク&データで精度を競うような経験がしたい
  - どこまで戦えるか? 今後どんなスキルを鍛えるべきか?
- 賞金欲しい
  - もっと良いGPUが買いたい (今は4年前に買ったGTX970)

# 全体像 – 2段階の認識パイプラインとその狙い –

- 文字領域の検出と1文字ずつの分類の2段階の認識手法
- 文字分類の難易度が高いと考え、独立に工夫を施せるように分割



## CenterNetの選定理由

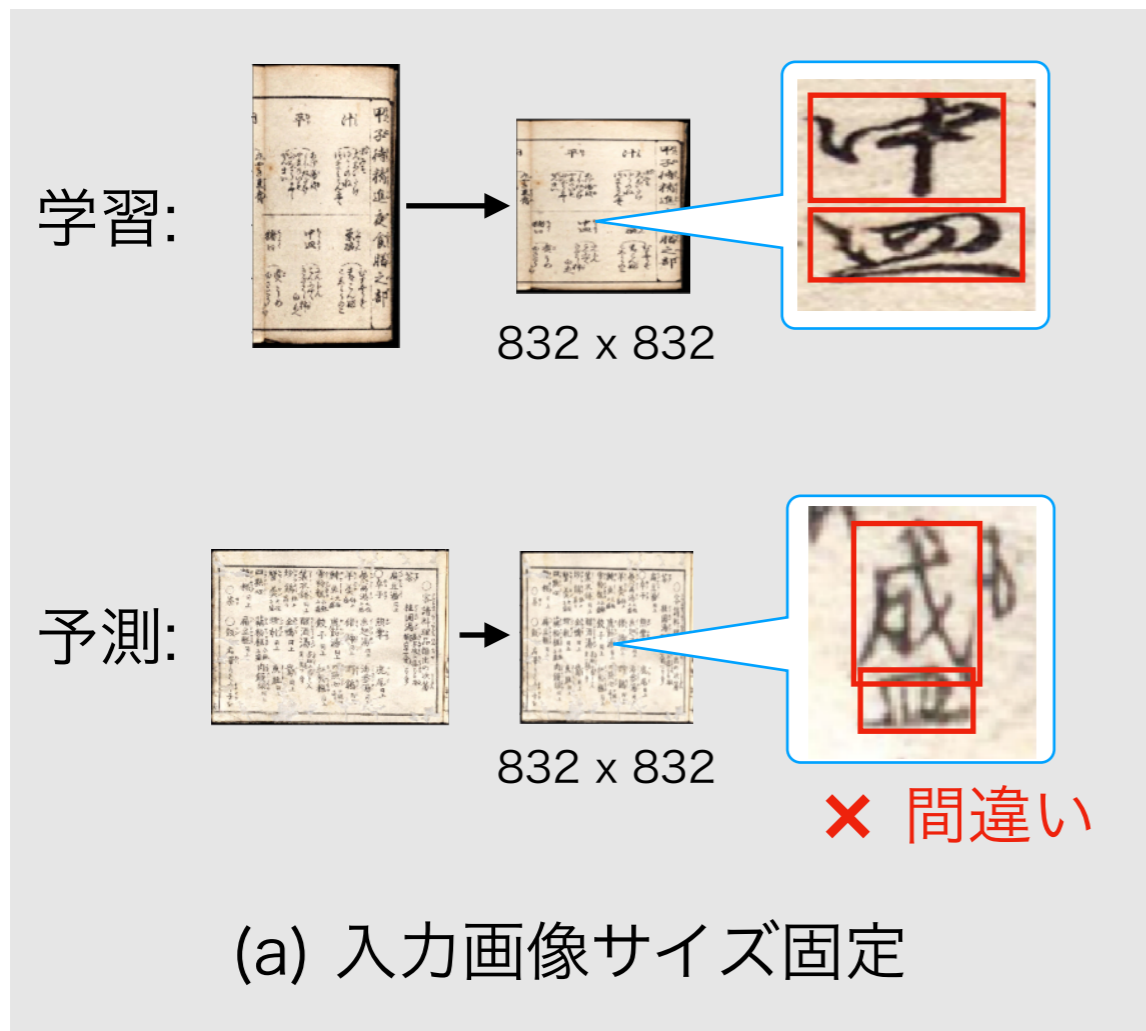
- シンプルで高精度
- 調整パラメータが少ない

## こちらの方が難しそう…

- 超多クラス (4000クラス規模)
- サンプル数の偏り

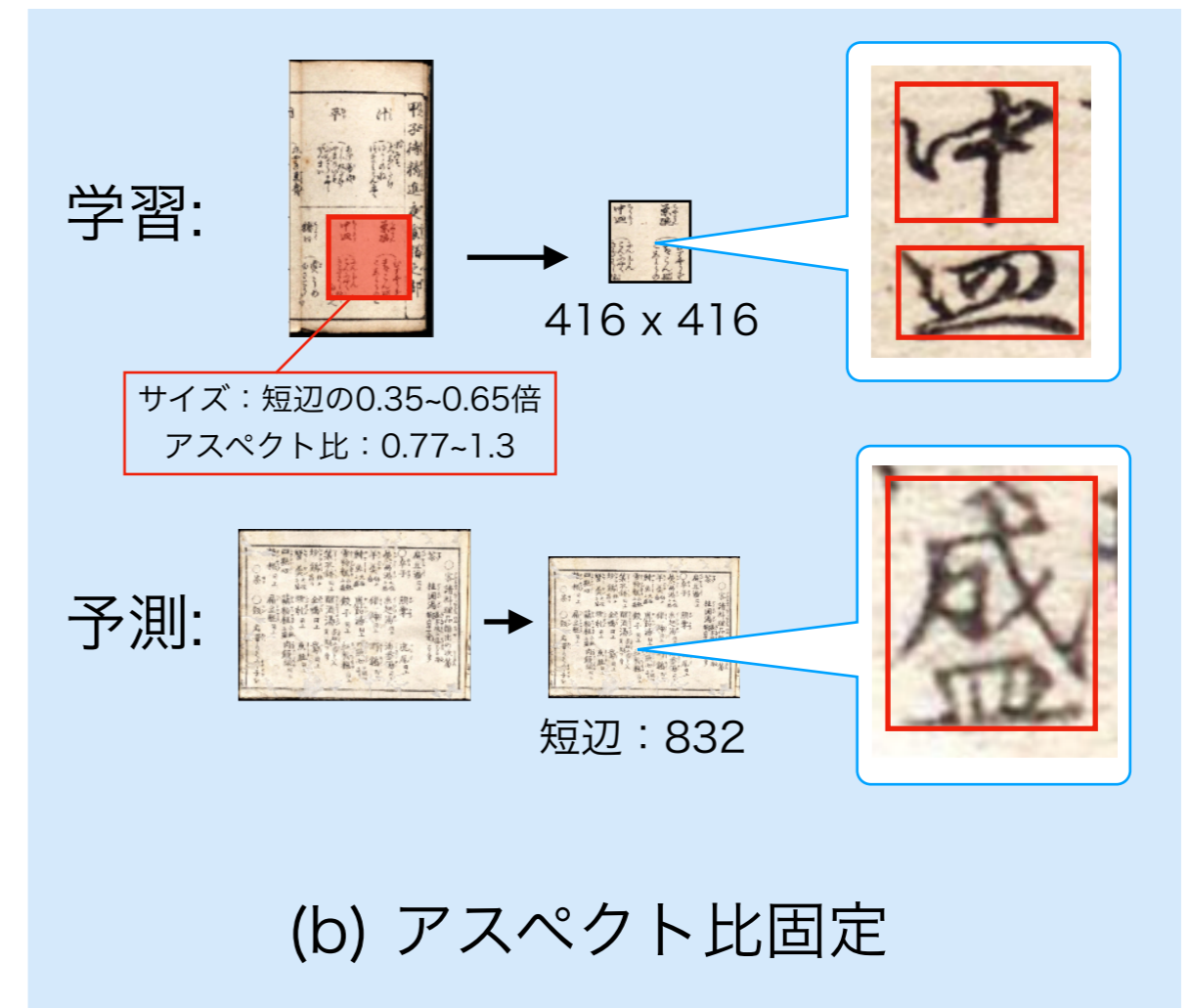
# 文字検出の工夫1 – アスペクト比の扱い –

- ・ 学習時とのアスペクト比の不一致は誤検知の原因になり得る
- ・ アスペクト比を固定し、本来の文字のバランスを学習させる



縦横のバランスが崩れ混乱の元？

※ 物体検出タスクとしては、(a)もよく採用される



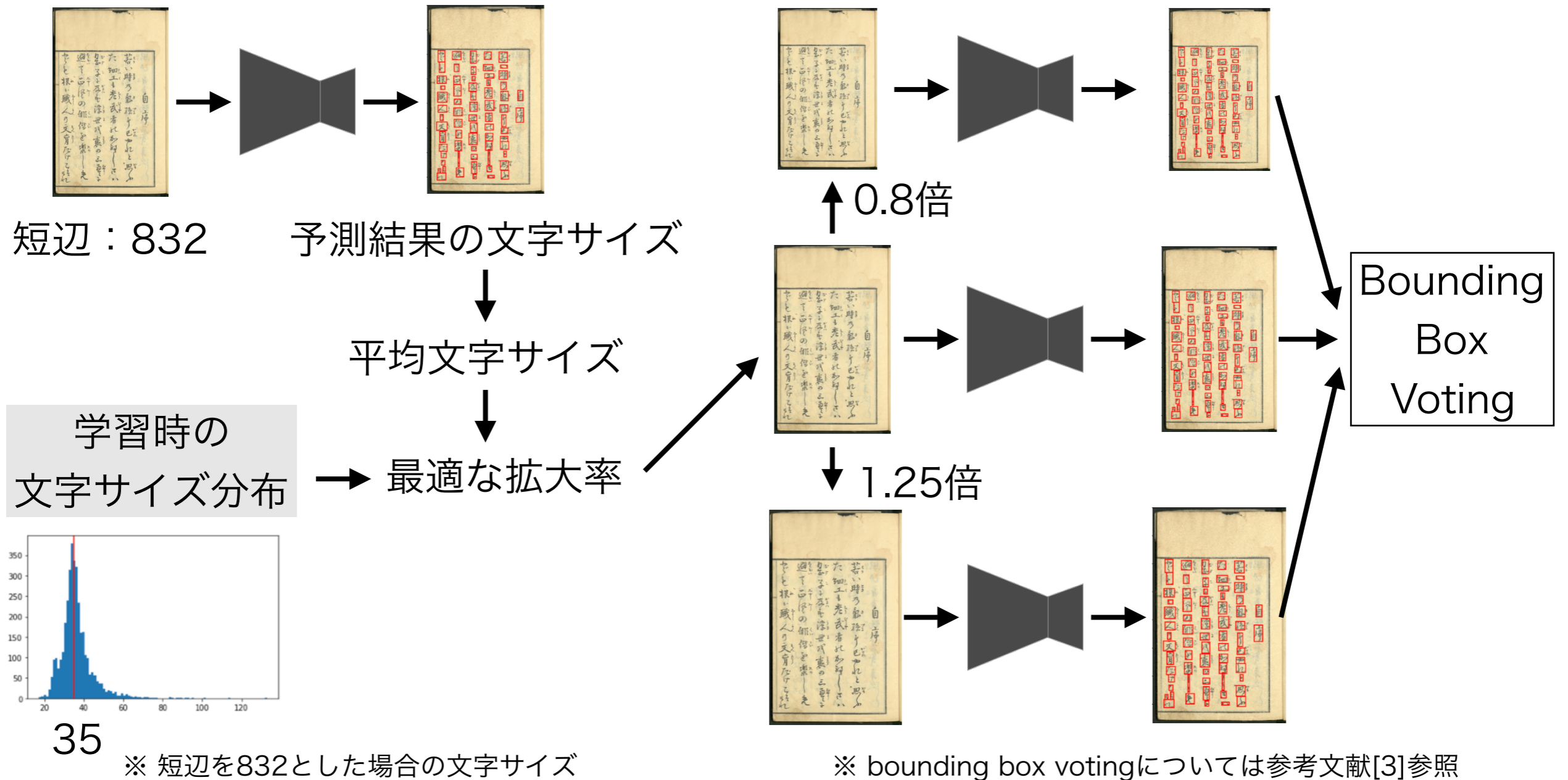
本来の文字のバランスを学習

他の恩恵 ...

- ・ ランダムクロップ → バリエーションの確保
- ・ 小さい学習画像 → 学習時の計算資源節約

# 文字検出の工夫2 – スケールの扱い –

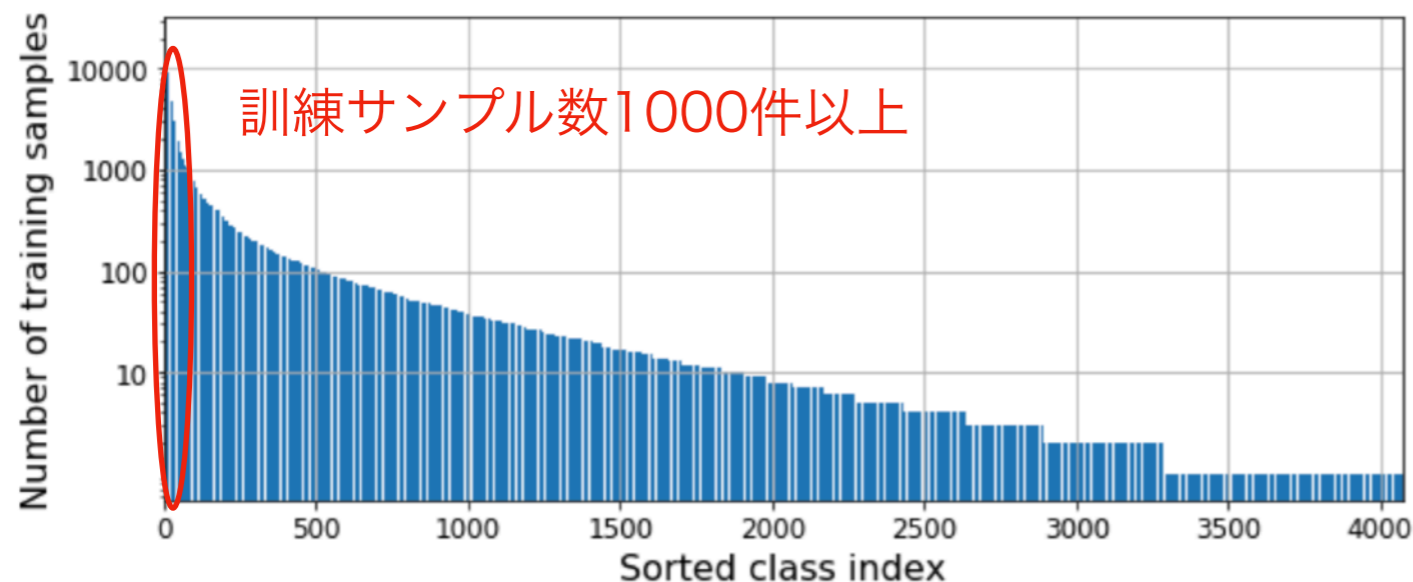
- 学習時の文字サイズと乖離すると、漢字の一部などを誤検知
- 学習したスケールへの調整 + 複数スケールの統合で見逃しを低減



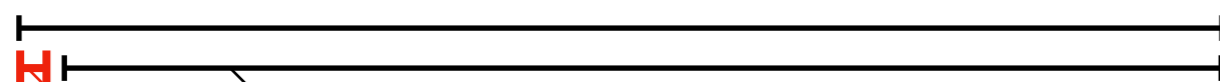
# 文字分類の工夫1 – 誤認識原因の分析 –

- 誤認識の原因がどこにあるか？ 検証セットの傾向から分析
  - よく登場する文字も精度が十分でなく、影響が大きい
  - 同じ文字でも書籍によって見た目が大きく変化

訓練セット：



検証セット： 文字分類精度: 88.8% (9/18時点)



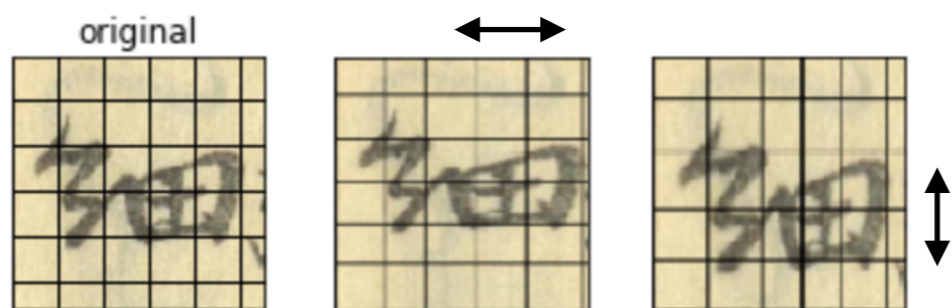
訓練サンプル数1000以下:  
精度79%, 誤差への寄与6.3%

訓練サンプル数1000以上 (81クラス):  
分類精度93%, 誤差への寄与4.8%



# 文字分類の工夫2 – くずし字のデータ拡張 –

- 意味が変わらない範囲で文字の見た目のバリエーションを増やす
- 部分を隠すことで、文字の全体の特徴を使うように訓練する



(a) Grid distortion



(b) Elastic transform



(c) Random erasing



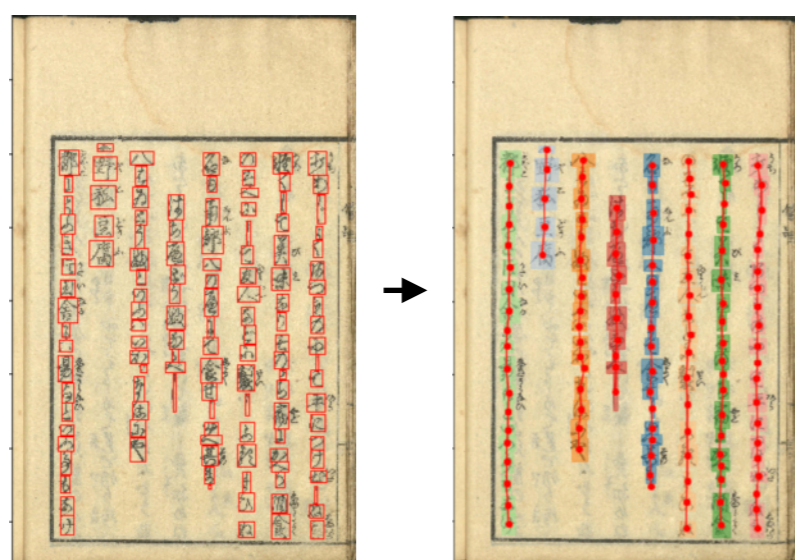
実際の学習サンプルの例

(色、明度、コントラストの変化やノイズなども付与)

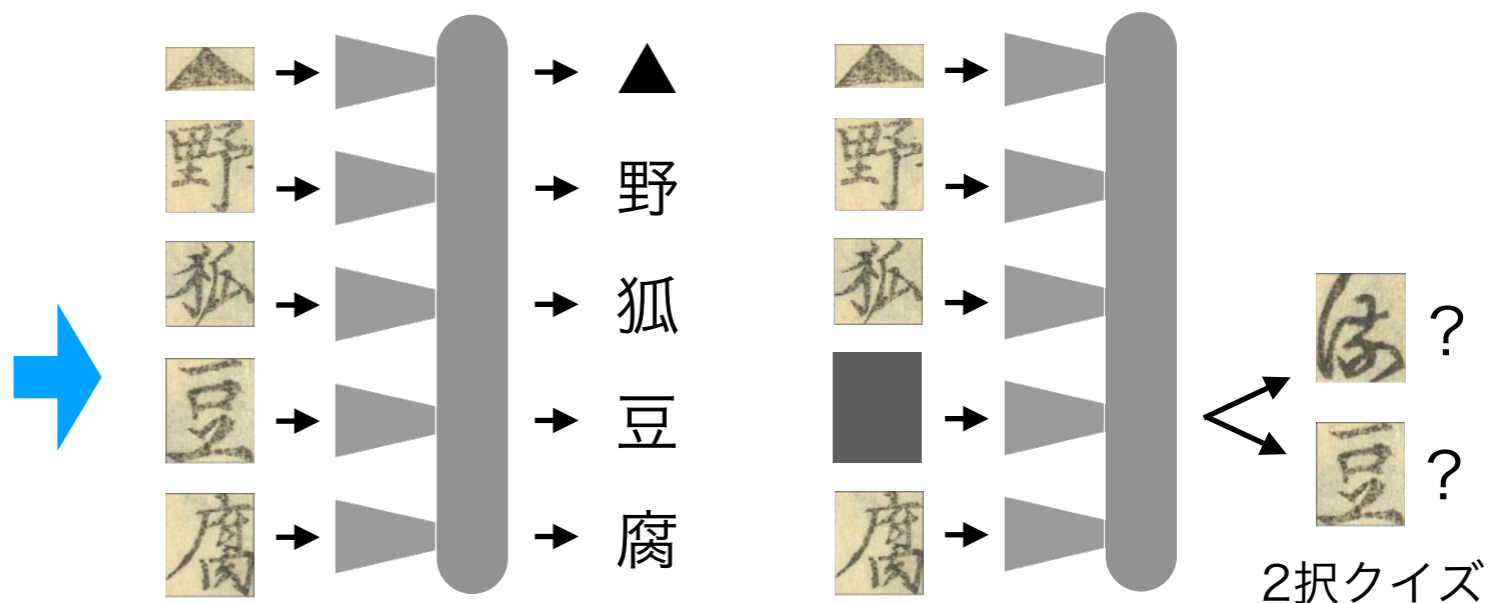


# 上手くいかなかったこと

- 文脈を使った文字認識
  - 前後の文字を考慮しないと判断の難しい文字があり、各々工夫
  - 自分はラベルなし画像からも文脈を学べる方法を模索（不採用）



行のグルーピング



a) 教師あり学習

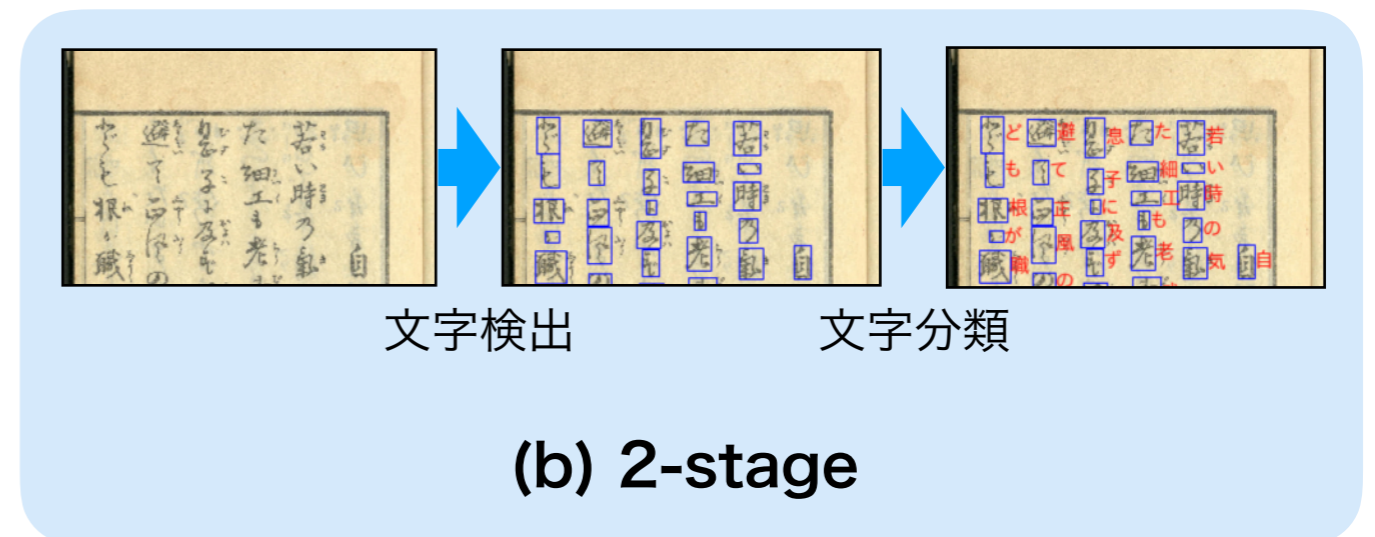
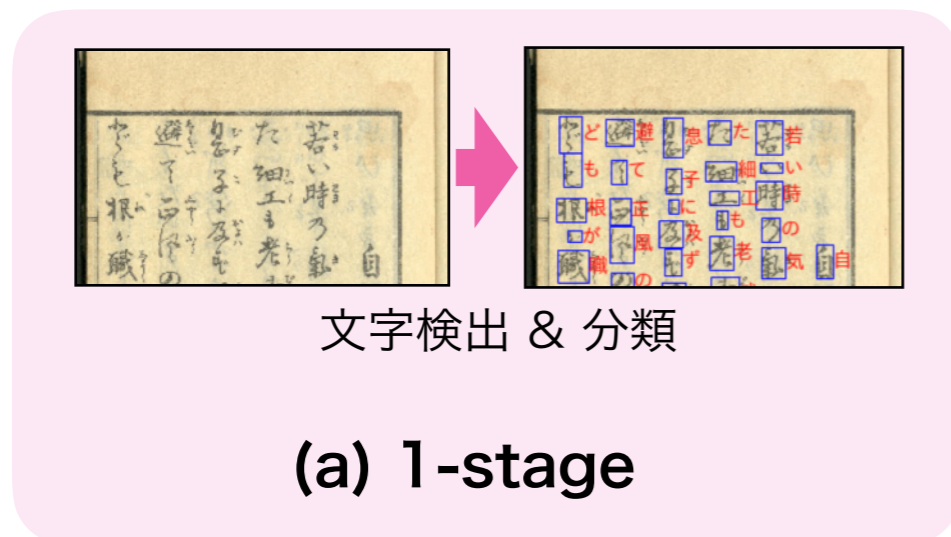
b) 教師なし学習

今回のコンペの  
学習データ  
(3881枚)

大量の  
画像のみのデータ  
(~60万枚)

# 学んだこと・気づき

- 2-stageの大枠を決めてしまうのが早かった
  - 文脈を扱うには1-stageの手法がシンプルだった
    - 4000クラス規模で1-stageは厳しいのではと試さなかった



- Kaggle参加の恩恵
  - モデリングに集中できる（指標やデータが与えられる）
  - スコアや順位が出るのでやる気が出る（負けたくない）
  - 競技期間中や終了後の議論やコード共有が勉強になる

# まとめ

- 手法：2段階の認識パイプライン
  - 文字検出：アスペクト比やスケールの扱い
  - 文字分類：くずし字のデータ拡張の工夫

手法：<https://www.kaggle.com/c/kuzushiji-recognition/discussion/113419>

コード：<https://github.com/t-hanya/kuzushiji-recognition>

- コンペ参加を終えて
  - 初参加で反省点も多いが、学びの多い3ヶ月（+ $\alpha$ ）だった
  - 膨大なラベルなしデータの活用は今後面白そう
- 最後に
  - 意義深く興味深いコンペの開催にあらためて感謝。
  - コンペ参加者の皆さんからも多くを学びました。ありがとう！

# 参考文献

1) Objects as Points

<https://arxiv.org/abs/1904.07850>

2) Searching for MobileNetV3

<https://arxiv.org/abs/1905.02244>

3) Object detection via a multi-region & semantic segmentation-aware CNN model

<https://arxiv.org/abs/1505.01749>

4) Random Erasing Data Augmentation

<https://arxiv.org/abs/1708.04896>