人工知能学会 インタラクティブ
情報アクセスと可視化マイニング研究会(第24回)
SIG-AM-24-02

看板画像を用いた検索システムにおける学習モデル構築手法の検討

Construction Method of Learning Model in Search System with Signboard Images

三溝 俊介 *　　松下 光範
Shunsuke Mitsumizo　Mitsunori Matsushita

関西大学総合情報学部
Faculty of Informatics, Kansai University

Abstract: The aim of the research is to develop a search system for foreigners who explore shops when they visit Japan. The system provides detailed information from shops with signboards written in Japanese characters that they do not understand. To develop the system efficiently, we constructed a learning model for signboard classification with dataset constructed from data that exist on the web. Our proposed system enables users to obtain detailed information by snap signboards with his/her own smartphone. The field experiment that we conducted revealed that the proposed method can determine shops appropriately.

1　はじめに

人々が都市空間において情報を取得する手段のひとつとして看板が挙げられる．看板の目的はその看板が掲示されている場所に存在する店舗や会社の名称を提示してその存在を知らせたり，企業や製品，音楽や映画などを告知したりすることが主であり，その存在が人の目に入ることで，そこに描かれているコンテンツやデザイン（文字等を含む）に何らかの影響を受けて具体的な行動に移すことが企図されている [1].

一方，近年の携帯端末の普及に伴い，都市空間における情報取得手段が変化してきている．携帯端末では，テキストを用いた検索や GPS 情報を用いた位置検索など多様な検索手法が提供されており，人は得たい情報に応じて様々な検索手法を使い分けることで，任意の場所で必要な情報を取得できるようになってきている．

こうした情報取得手段の多様化は，街中での店舗検索行動を変えつつある．例えば，人々が外出先で飲食店を選択する際には，提供される料理のメニューやその店舗の口コミなどの情報を検索したり，WEB 配信されている割引クーポンの有無を検索したりすることが一般的になってきている．看板は，これまでの役割に加え，こうした店舗検索行動におけるトリガーにもなっている．

*連絡先：関西大学総合情報学部
〒 569-1095 大阪府高槻市霊仙寺町 2-1-1
E-mail: mat@res.kutc.kansai-u.ac.jp

このような店舗検索を行う際，ユーザが看板の文字を理解し，携帯端末に店名などのテキストをキーワードとして入力することが前提となっているが，看板に書かれている文字が利用者の母国語でない場合や，崩した文字で書かれているような場合（蕎麦屋や割烹など）は，検索が容易でない．

北村らはこの問題に対し，携帯端末のカメラで撮影した看板を機械学習を用いて分類し，AR を用いて店舗の詳細情報を画面上に重畳表示することで解決を図った [2]．しかし，北村が提案した手法では，看板を分類する機械学習モデルを構築するために手動で看板画像の収集とアノテーションを行っているため，データセットの構築に多大なコストがかかるという問題があり，検索可能な店舗を増やすことは容易ではなかった．そこで，本研究ではこのデータセット構築を自動化することで対象店舗の拡大に要するコストの削減を図る．その端緒として本稿では対象を飲食店に限定し，その看板画像をグルメサイトから収集して，機械学習モデルを構築する手法を提案する．

2　関連研究

本章では本研究で用いるオブジェクトの検出・分類手法について述べる．

オブジェクト検出手法の 1 つである You Only Look Once（YOLO）や YOLO を改良した YOLOv2 では入

力画像を複数のグリッドに分割し，セルごとにどのクラスかを予測し，グリッドとは別に複数のバウンディングボックスとボックス内にオブジェクトが存在する確率を求め，オブジェクトの領域とクラスを出力する [3]．YOLOv2 ではリアルタイムなオブジェクト検出が可能だが，セルごとにクラス分類を行うという性質上，1 つのセルに複数のオブジェクトが存在する場合に正確な分類が困難である．

オブジェクト分類手法の 1 つに VGG16 モデルがある [4]．VGG16 は ImageNet[5] の ILSVRC-2012 データセットに含まれる 1000 クラスに分類された 130 万枚の画像を学習させた CNN モデルの 1 つであり，13 層の畳み込み層と 3 層の全結合層の計 16 層で構成されている．VGG16 では入力された画像を複数回の畳み込みとプーリングを行い，分類したクラスとその確率を出力する．

3　看板を用いた検索手法

本章では提案手法の基礎となる看板を用いた検索手法について述べる．

北村らは，看板画像から店舗情報にアクセスする手法を提案している [2]．人手で看板画像の収集とアノテーションを行ったデータセットを用いて，看板画像の認識･分類を行うモデルを構築し，送信された看板画像から店舗情報を返却する WebAPI サーバの構築を行っている．この手法では，一定時間ごとに携帯端末で撮影された画像を API サーバに送信し，API サーバ上で分類した店舗に紐付いた OpenStreetMap（OSM）[6] に登録されている情報を Overpass API[1]で取得している．このシステムでは Tensorflow1.0[7] で実装した YOLOv2 を用いて看板領域の検出を行っており，検出した領域を VGG16 で分類している．分類した看板が存在する座標にその店舗の情報を吹き出しによって携帯端末上に重畳表示することで，リアルタイムな認識と情報提示を実現している．

この手法は看板の分類に文字列ではなく看板画像を用いているため，文字の並び方が特殊な場合や崩して書かれた文字であっても分類が可能である．しかし，北村らの提案手法では看板画像の収集及びアノテーションを手動で行う必要があり，対象地域の拡大を行う際に多大なコストがかかる．

4　デザイン指針

本章では，提案手法の実装にあたってのデザイン指針を定める．

3 章で述べたように，文字認識を行わず，看板画像を分類することで文字の並び方が特殊な場合や崩した文字であっても分類が可能である．そのため，本研究では北村らの手法を参考に看板画像を分類することで検索を行う．しかし，北村らの提案手法は手動でデータ収集とアノテーションを行っており，対象地域の拡大に要するコストが莫大であった．そのため，本研究ではこれらのコストを削減するために，ウェブ上に存在するデータを活用し，自動でデータの収集とアノテーションを行う．

本研究で実装する検索システムは，外出時にユーザが目の前にある飲食店に関する詳細な情報を得たいが，文字が読めず検索ができない状況を想定している．そのため，看板から離れた位置で行う検索や，看板と遮蔽物が重なった状態での検索は対象としない．

上述したデザイン指針とシステムが使用される状況から本システムが満たすべき要件を以下に定める．

1. データ収集が自動で行われること
2. アノテーションが自動で行われること
3. 複雑な文字であっても検索が可能であること
4. 実世界において飲食店の検索が可能であること

これら 4 つの要件を満たすシステムの実現を目指し，実装したシステムでフィールド実験を行うことで，実世界において検索が可能か評価を行う．

5　実装

本研究では，4 章で述べた要件を達成できるシステムの実装を行う．要件 1 を達成するために，ウェブ上のデータからスクレイピングで看板画像の収集を行う．要件 2 を達成するために，画像の収集は整理されたウェブサイトから行う．要件 3 を達成するために，文字認識を行わず看板画像の分類を行うことで複雑な文字であるかどうかに関わらず認識を可能にする．要件 4 が達成されていることを確認するために，実装したシステムを用いてフィールド実験を行う．

本研究では対象とする店舗をグルメサイト食べログ[2]に掲載されている飲食店とし，BeautifulSoup4[3]を用いて食べログの外観画像ページに存在する店舗の外観画像をスクレイピングによって取得する．スクレイピングは以下の手順で 2019 年 9 月 26 日から 9 月 27 日にかけて行った．まず，食べログのランキングページの 1 ページ目を取得し，取得したページから html の a タグで class が「list-rst__rst-name-target」の要素に各店

[1] http://overpass-api.de/（2020/1/31 存在確認）

[2] https://tabelog.com/ （2020/1/31 存在確認）

[3] https://pypi.org/project/beautifulsoup4/（2020/1/31 存在確認）

舗情報のページの URL があるため，そこから各店舗情報ページの URL を取得し，リストに格納する．次にリストに格納された各店舗情報ページの URL の末尾に「dtlphotolst/dtlphotolst/4/smp2/D-normal/1/」を加えることで各店舗の外観画像ページの 1 ページ目にアクセスできる．外観画像ページには 1 ページあたりに 40 件の投稿された画像があり，a タグで class が「js-imagebox-trigger js-analytics rstdtl-thumb-list__target」の要素に各画像の URL があるため，その URL から各画像を保存する．ページ内の画像を全て取得した場合は末尾の数字を 1 つ増やすことで次のページにアクセスする．同様の処理を最後の外観画像ページまで行い，取得できる画像がなくなるとリストに保存されている次の店舗で同様の処理を繰り返す．リストの中に保存されている店舗の画像を全て取得した場合はランキングページの次のページで同様の処理を行う．以上の処理を指定回数行うことで外観画像を収集する．

スクレイピングによって得られた画像を次節で述べる YOLOv2 を用いて看板を検出した．検出された信頼度が 80%以上の看板の領域を切り出し，切り出された画像が 20 枚以上存在する店舗を対象に，画像の 60%をトレーニングデータ，20%をバリデーションデータ，20%をテストデータとして店舗ごとに保存した．看板の検出には YOLOv2 を用いる [3]．YOLOv2 によるオブジェクト検出は 2 章で述べたように，1 つのセルの中に多数のオブジェクトが存在する場合の正確な分類が困難である．しかし，4 章で述べたように，本研究で対象とする状況は目の前に存在する店舗の検索を行う状況であるため，多数の看板が 1 つのセルの中に密集することは稀である．そのため看板の検出には YOLOv2 を採用した．本研究では，北村が学習させたモデルを用いて看板の検出を行う．このモデルは特定の店舗の看板を検出するのではなく様々な店舗の看板を検出することが可能な汎用的なモデルであるため，YOLOv2 で学習させていないスクレイピングで収集した店舗の看板を検出することが可能である．また，北村の行った実験で実世界での看板検出を十分に行えることが確認されているため，看板の検出アルゴリズムとしてその精度が担保されていると考えられる．

看板の分類には VGG16[4] を用いる．ニューラルネットワークのライブラリである Keras で実装された VGG16 を用いて，北村の手法を参考に，全結合層を取り除き，新たに全結合層を追加する転移学習を用いた[4]．そのため，学習に用いる画像が少ない店舗であっても短時間で比較的高精度なモデルの構築が可能である．

上述したモデルを用いて，携帯端末のカメラで撮影された看板の店舗を検索するシステムを実装した．看板認識の処理は高負荷であり，ハイスペックな GPU を要する．そのため，これらの処理は携帯端末上では行わず，ハイスペックな GPU を搭載したマシン上で看板の認識を行うウェブ API サーバを構築し，携帯端末から 500 ミリ秒ごとにこの API を呼び出すことで分類の処理を行う．看板認識の処理を行うマシンのスペックを表 1 に示す．API の実装には Python3.7.4 を用いて WebAPI フレームワークの Falcon[5]を用いて構築する．この API は携帯端末から送信された画像内の看板領域を Tensorflow1.13.1[7] で実装された YOLOv2 を用いて検出し，検出された看板を VGG16 で分類する．検出した看板領域の左上の座標と右下の座標及び判別した店舗の食べログページの URL を JSON 形式で返却する．

表 1: 看板画像の認識を行うマシンのスペック

要素	スペック
CPU	Intel® Core™ i7-8700K @ 3.70GHz
RAM	16GB
GPU	NVIDIA GeForce GTX1080ti
VRAM	12GB
OS	Ubuntu 18.04

携帯端末で検索する際のインタフェースは HTML5 と JavaScript を用いて実装した．JSON 形式で返却された看板領域の左上と右下の座標からボックスを生成し，その領域以外の明度を下げることで看板領域を図 1 のようにハイライトする．店舗情報の表示には AR を用いず，ハイライトされた領域をユーザがタップした際に，返却された URL に遷移する．また，API に送信された画像内に複数の看板が検出された場合は，ユーザがタップした店舗に対応する URL に遷移する．

6 評価

対象地域拡大によって分類するクラス数が増加するため，クラス数の増加による分類精度への影響を確認する必要がある．そのため，収集した画像から 150 店舗の分類を行うモデルと 30 店舗の分類を行うモデルの 2 種類を構築した．150 店舗の分類を行うモデルは画像サイズを 224×224，バッチサイズを 16，学習率を 10^{-4}，慣性を 0.9，エポック数を 300 で学習させ，30 店舗の分類を行うモデルはエポック数を 100 で残りの条件は同じ状態で学習させる．最適化アルゴリズムには Stochastic Gradient Descent（SGD），損失関数には交差エントロピーを用いる．また，学習時に Keras の ImageDataGenerator を用いて学習画像の拡張を行う．

あるクラスについて陽性と予測したクラスが実際に陽性であるものを True Positive（TP），陽性と予測し

[4]https://keras.io/(2020/3/2 存在確認)

[5]https://falconframework.org/（2020/1/31 存在確認）

図 1: ハイライトされた看板

たクラスが実際には陰性であったものを False Positive（FP），陰性と予測したクラスが実際には陽性であったものを False Negative（FN），陰性と予測したクラスが実際に陰性であったものを True Negative（TN）とした場合に陽性と予測したものの中でそれが実際に正しかった割合である適合率，実際には陽性であるクラスの中で陽性と予測したものの割合である再現率，適合率と再現率の調和平均である F 値はそれぞれ以下の式に表される．

$$\text{正解率} = \frac{\text{予測と正解の一致数}}{\text{テストデータ数}} \quad (1)$$

$$\text{適合率} = \frac{TP}{TP+FP} \quad (2)$$

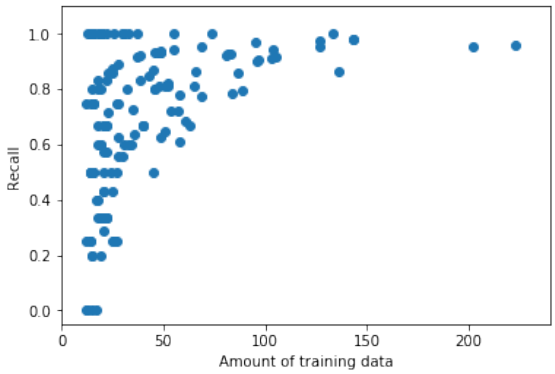
$$\text{再現率} = \frac{TP}{TP+FN} \quad (3)$$

$$F\text{ 値} = \frac{2 \times \text{適合率} \times \text{再現率}}{\text{適合率} + \text{再現率}} \quad (4)$$

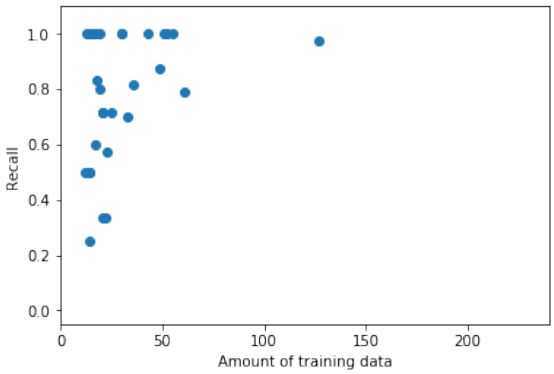
テストデータを用いて 2 つのモデルについて全体の正解率及び適合率，再現率，F 値に関して全クラスの結果の平均であるマクロ平均を算出した．150 クラスの分類を行うモデルの全体の正解率は 0.808，適合率は 0.801，再現率は 0.710，F 値は 0.726 であった．また，30 クラスの分類を行うモデルの全体の正解率は 0.848，適合率は 0.873，再現率は 0.784，F 値は 0.795 であった．150 クラスのモデルと 30 クラスのモデルを比較した場合，全てにおいて 30 クラスのモデルの方が高いことが確認できた．そのため，より数の多いクラスの分類を行う際に分類の精度が低下することが示唆された．また，150 店舗のモデルにおけるトレーニングデータの数に対する F 値の変化を図 2（a）に示し，30 店舗のモデルにおいてのトレーニングデータの数に対する F 値の変化を図 2（b）に示す．図に示されるように双方のモデルにおいてトレーニングデータの数が少ない店舗はデータの数が多い店舗と比較して精度が低いものが多く含まれ，トレーニングデータの数が多くなるほど精度の向上が低減することが確認された．しかし，データ数が最も多い 1 店舗については F 値が大きく低下している．これは，学習に用いたデータ数が店舗によって異なったため，過度にこの 1 店舗が陽性となるように最適化した可能性がある．

前節で述べたように，全体的にデータ数が増えるほど全ての評価指標における精度の向上が確認されたが，データが不均衡であったためか最もデータ数が多い 1 店舗の F 値が大きく低下していた．本節ではトレーニングデータが 20 枚以上であった 101 店舗を対象としてトレーニングデータ数に変更を行わず学習させたモデルと全ての店舗のトレーニングデータ数を 20 枚にアンダーサンプリングして学習させた 2 つのモデルを評価する．2 つのデータセットを 150 店舗の分類を行うモデルと同様のパラメータで学習させ，評価には同じテストデータを用いた．アンダーサンプリングを行わなかったモデルにおける全体の正解率は 0.843，適合率は 0.825，再現率は 0.772，F 値は 0.778 であった．一方でアンダーサンプリングを行ったモデルにおける全体の正解率は 0.826，適合率は 0.807，再現率は 0.784，F 値は 0.778 であり，アンダーサンプリングによって全体の F 値は改善されなかった．

テストデータだけでなく実世界においても，システムが使用可能であることを確認するために，5 章で述べたシステムを 150 店舗の分類を行うモデルを用いて実際の看板で使用した．実装したシステムで 10 店舗の検索を行い，各店舗ごとに 10 回ずつ複数の角度からの検索を行った．1 つの店舗に複数の看板がある場合は 2 つの看板で 5 回ずつ検索を行った．表 2 に検索を行った店舗名，10 回の内正しく分類された回数を示す．2 つの看板で検索したものは括弧内に 5 回ずつ行った結果を示す．表 2 に示されるように，全体的に正しく分



（a）150 店舗のモデル



（b）30 店舗のモデル

f

図 2: トレーニングデータ数と F 値の関係

表 2: 実世界での認識割合

店舗名	分類した割合
板前焼肉 一斗 天下茶屋本店	6（5,1）
釜揚うどん一紀	10
無鉄砲 大阪店	8（3,5）
ラーメン人生 JET	10（5,5）
まき埜	10
烈志笑魚油 麺香房 三く	8（5,3）
ROUTE271 梅田本店	5（0,5）
渡邊カリー 梅田本店	10
鮨処 多田	9
中村商店 高槻本店	8

類される回数が多く，店舗によっては 10 回全て正しく分類するものもあった．2 つの看板で検索したものについてはどちらか一方については 5 回の内 5 回とも正しく分類を行うことができた.「ROUTE271 梅田本店」の一度も正しく分類できなかった看板については, 看板の上に時計がついている特殊なものであり，検出を行う際に時計を含めて検出される場合と看板のみが検出される場合があり，看板領域の検出が不安定であった．そのため，正しい分類が行われなかったと考えられる．また，一部の店舗でのれん画像がトレーニングデータに含まれていた．ユーザが実際にシステムを使用する場合にのれんから検索を行うことが想定できる．そのため，実験を行った店舗のうち，のれんがあった「烈志笑魚油 麺香房 三く」の検索を行う際には通常の看板に加え，のれんの検索も行った．のれんの分類は 5 回のうち 3 回は正しく分類できたが，風などによってのれんの形状が変化した場合には精度がさらに低下すると考えられる．

7　課題と展望

本研究では食ベログのデータを活用することで，看板画像を用いた検索システムにおいて従来は手動で行われていたデータ収集とアノテーションの自動化を図った．これにより，多数の飲食店の看板が分類可能になった．しかし，食ベログから得られるデータは飲食店のみであり，飲食店以外の店舗には拡張できないという問題がある．この問題を解決するには，多種多様なウェブ上のデータを収集し，店舗ごとに整理する必要があり，これを自動化することは容易ではない．

また，実験により，このモデルが実世界での店舗の分類にもある程度用いられることが確認されたが，データ数の少ない店舗や学習に不要な画像が混じっている店舗があった．一般的にトレーニングデータの不足や誤ったデータの混入が精度に影響を与える．このことから開店して間もない店舗や看板と誤認識するオブジェクトが含まれる店舗の分類が困難である．そのため，精度の向上には構築されたデータセットに含まれている不要なデータを取り除く処理が必要であると考えられる．

本研究では，対象とするデータを飲食店の看板としたが，飲食店の看板には矩形の中に文字が存在する典型的な看板から矩形が無い看板や特殊な形をしたものまで様々な看板がある．また，光が看板の状態に影響を与えるため，電飾の有無も考慮する必要がある．これらを厳密に定義し，収集したデータを用いて，モデルを構築することで，不要な画像の検出量を減らすことが可能になると考えられる．

データ収集とアノテーションについては自動化を行い，大幅なコストの削減を図った．しかし，機械学習モデルを構築する際のパラメータをデータの数や種類によって適宜調整しなければならず，本研究においてもこの部分については手動で調整を行っているため，完全な自動化を行うにはこのようなモデル構築時の設定

を自動化する必要がある．

8　終わりに

本研究では，食べログの外観画像ページに存在する画像データをスクレイピングで収集し，看板領域を抽出した画像を用いて構築した機械学習モデルの評価を行った．今後，7 章で述べた課題を解決し，より多種多様な店舗を対象としたシステムの実現を目指す．

参考文献

[1] 小山 雅明, 高橋 由樹, 椎塚 久雄: “そそる看板” デザインの基礎的考察, 日本感性工学会論文誌, Vol. 15, No. 1, pp. 65-73 (2016).

[2] 北村 茂生, 松下 光範: オンサイト検索:携帯端末を用いた看板画像からの店舗情報アクセス手法, *DEIM Forum 2019*, F6-4 (2019).

[3] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection, *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788 (2016).

[4] Liu, S. and Deng, W.: Very deep convolutional neural network based image classification using small training sample size, *2015 3rd IAPR Asian Conference on Pattern Recognition*, pp. 730-734 (2015).

[5] Deng, J. *et al.*: ImageNet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248-255 (2009).

[6] Haklay, M. and Weber, P.: OpenStreetMap: User-Generated Street Maps, *IEEE Pervasive Computing*, Vol. 7, No. 4, pp. 12-18 (2008).

[7] Abadi, M. *et al.*: TensorFlow: A System for Large-Scale Machine Learning, *12th USENIX Symposium on Operating Systems Design and Implementation*, pp. 265-283 (2016).