Twitter 食事画像からの詳細カテゴリ発見

伊藤 祥文[1,a)]　柳井 啓司[1,b)]

1. はじめに

現在, 多くの日本人が Twitter を用いて手軽に書き込みを行うようになっている. また, 昨今のスマートフォンの普及により, 撮影した写真などをテキスト情報とともに Twitter に投稿することが多くなった. これらのデータから, 画像ごとに共通する単語や, 画像と単語の関連性を見ることができる. これにより, 特定の単語と他の単語の関連について新たな発見をする可能性がある.

本研究では, Twitter 上の画像付きツイートからその内容を分析することで, いくつかの単語に対してその関連単語を発見することを目的とする.

2. 目的

本研究では, ノイズが極めて多く含まれる Twitter 画像データから詳細カテゴリを発見し, 詳細カテゴリ画像データセット自動構築することを目的とする. 本研究では, 特に食事画像について扱うこととし, ラーメン画像と, カレー画像に詳細カテゴリ画像データセットの自動構築を行った.

3. 関連研究

Twitter を用いた研究として, Bian らの研究がある [1]. 現在, Twitter は多くの研究において Web からのテキストマイニングの対象として広く研究されており, この研究では投稿されたテキストをひとつのブログとして用いることでイベントの検出や内容の要約を行っている. しかし, この研究では英語や中国語の投稿内容を参考にしているため, 日本語のような単語間の区切りがないような言語に対しては, キーワード発見の方法が異なると考えられる.

また, 画像とテキストを関連づけた研究として, 金子らの研究がある [2]. この研究では, イベントの検出をする際に日本語のツイート内容を対象としているため, 英語などより難解とされる日本語に対しても単語抽出や分析を行うことができることを示している. また, 投稿内容を見ることで, 内容と添付画像の間に関係性が見られるものとそうでないものを分類することができることを示している.

[1] 電気通信大学　〒 182–8585 東京都調布市調布ケ丘 1–5–1
[a)] ito-y@mm.inf.uec.ac.jp
[b)] yanai@mm.inf.uec.ac.jp

また, 近年の画像認識に関する研究の一つとして, 大量の画像を分類する研究が行われている. 例えば, Yang らの研究 [3] では, 車の画像のデータセットを作成し, それを用いて一般の車の画像を分類する手法を提案している.

車以外の画像を分類した例として, 松田らの研究がある [4]. この研究では, 大量の食事画像を用いて UEC-FOOD100 食事画像データセットを作成している. 本研究では, この研究で作成された食事画像データセットを用いて, Twitter 上の大量の画像から食事画像を検出する.

図 1　食事画像データセット

4. 手法

本研究の手法の全体的な流れは, 以下のようになっている.

(1) テキストからの単語抽出
(2) 認識可能性の評価
(3) 画像の選別

画像の選別後, 再度認識可能性を評価する.

4.1 テキストからの単語抽出

2011 年 5 月から 2016 年 4 月までの 5 年分の Twitter の過去ログから, 目的の単語 (以後, 親単語とする) を含む食事画像付きツイートを収集する. 得られたテキストに対して形態素解析を行い, 親単語を含む名詞をタグとして抽出する.

形態素解析には, オープンソースの日本語形態素解析システム MeCab [*1] を用いる. これを用いることにより, 単語と単語が区切られていない日本語においても, 図 1 のように形態素解析を行うことができる. なお, 本研究では「博多ラーメン」などのような, 二つ以上の名詞で構成された名詞句を一つの名詞として扱う.

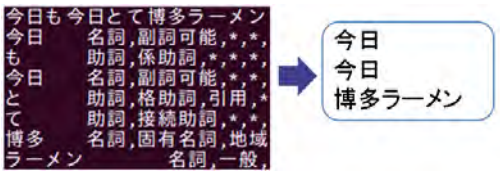

図 2 MeCab の実行例

得られた単語のうち, 多く見られた単語を用いて過去ログから画像を収集する. ここで収集した画像には, このとき検索で用いた単語をタグとして登録する.

4.2 認識可能性の評価

得られた画像を用いてマルチクラス分類を行い, その分類率を見ることで詳細カテゴリの認識可能性を評価する. 分類率とは, 全画像のうち正しく分類された画像の割合を表す. 本研究では, 分類率として上位一位の場合と五位の場合を見, その中に画像の含まれるカテゴリがあるかどうかで分類率を測定する.

本研究では, マルチクラス分類に, Deep Learning のライブラリである Caffe [*2] を用いる. Caffe[5] とは, GPU に対応した高速な Deep Learning フレームワークであり, 画像認識の分野で広く用いられている.

本研究では, Caffe を動かすシステムとして, DIGITS [*3] を用いた. DIGITS とは, NVIDIA 社が提供している Deep Learning 用のトレーニングシステムである. これを用いることにより, データベースの構築や, それを用いた深層学習を簡単に行うことができる. また, 学習経過や結果を視覚的に捉えることができ, 学習状況の確認や中間層の可視化などを行うことができる.

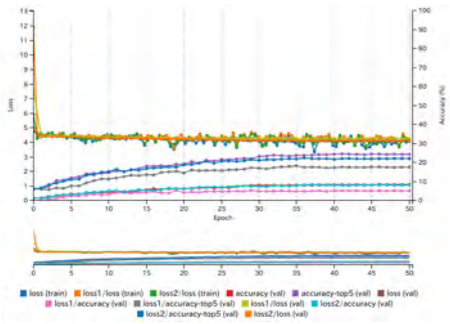

図 3 DIGITS での学習状況を表す画面の例

*1 http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html
*2 http://caffe.berkeleyvision.org/
*3 https://developer.nvidia.com/digits

本研究の場合, CNN 特徴+SVM では詳細分類は困難なため, GoogLeNet[6] をベースモデルとして用いてファインチューンすることとした. 評価の結果は, 図 4 のように確率値で表す.

図 4 認識可能性の評価を行った例

4.3 画像の選別

データセットの信頼性を上げるために, データセット内のノイズ画像を除去する. 画像を選択する方法として, 各画像毎の確率値を用いる. 確率値の低い画像は, カテゴリ特有の特徴を持っていないと判断できるため, 無関係な画像として除去する.

図 5 画像選別の例

5. 実験

本研究では, 2011 年 5 月から 2016 年 4 月までに収集された, 5 年分の食事画像認識済み Twitter 過去ログを用い, 以下の 2 つの実験を行った.

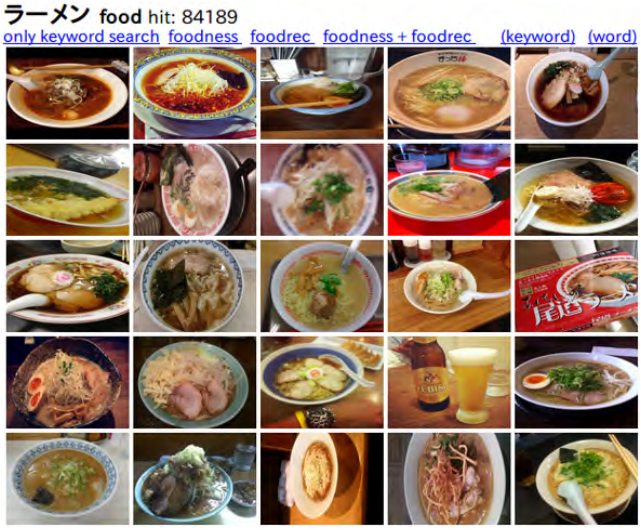

図 6 食事画像認識済み twitter 過去ログの一部

実験では, 親単語をひとつ設定し, その単語を含む単語を用いてカテゴリ設定を行い, カテゴリごとに画像を収集した. まず, 各カテゴリあたり 250 枚の画像を用いて分類率の測定を行った. その後, 各カテゴリにおいて, 確率値が低い順に 50 枚の画像をノイズ画像として除去し, 分類率を再測定した.

5.1 ラーメン画像データセットの作成

親単語を「ラーメン」という単語にし, 「味噌ラーメン」のような, 親単語を含む単語を 50 個抽出して画像を収集し, データセットを作成した.

表 1 抽出されたラーメンカテゴリ一覧

味噌ラーメン	塩ラーメン	醤油ラーメン	台湾ラーメン	家系ラーメン
豚骨ラーメン	とんこつラーメン	徳島ラーメン	博多ラーメン	カレーラーメン
トマトラーメン	尾道ラーメン	カップラーメン	佐野ラーメン	チキンラーメン
喜多方ラーメン	みそラーメン	朝ラーメン	ネギラーメン	和歌山ラーメン
インスタントラーメン	辛味噌ラーメン	熊本ラーメン	野郎ラーメン	長浜ラーメン
ラーメンテロ	しょうゆラーメン	特製ラーメン	ラーメン最高	二郎系ラーメン
ラーメン二郎	北極ラーメン	札幌ラーメン	旭川ラーメン	激辛ラーメン
八王子ラーメン	野菜ラーメン	ラーメンセット	ラーメン大	スタミナラーメン
最近ラーメン	ラーメン画像	牛骨ラーメン	味玉ラーメン	彩華ラーメン
大砲ラーメン	ラーメン人生	味噌カレー牛乳ラーメン	横浜家系ラーメン	8 番ラーメン

作成したデータセットと, データセットからノイズ画像を除去したもののそれぞれにおいて分類率を測定した. その結果, 分類率は表 2 のように変化した.

表 2 ラーメン画像における認識可能性評価の変化

データセット	Top-1 分類率	Top-5 分類率
ノイズ除去前	40.2 %	61.9 %
ノイズ除去後	49.6 %	69.9 %

また, 図 7 のような画像が得られた.

5.2 カレー画像データセットの作成

親単語を「カレー」という単語にし, 「カツカレー」のような, 親単語を含む単語を 50 個抽出して画像を収集し, データセットを作成した.

表 3 抽出されたカレーカテゴリ一覧

カレーうどん	スープカレー	カレーライス	チキンカレー	キーマカレー
グリーンカレー	インドカレー	朝カレー	ドライカレー	ビーフカレー
野菜カレー	カレーパン	バターチキンカレー	夏野菜カレー	タイカレー
ゴーゴーカレー	ポークカレー	トマトカレー	チーズカレー	オムカレー
シーフードカレー	牛すじカレー	カレーランチ	海軍カレー	焼きカレー
カレードリア	レッドカレー	マトンカレー	インデアンカレー	牛タンカレー
カレー鍋	インディアンカレー	金沢カレー	スライムカレー	ほうれん草カレー
牛スジカレー	カレー曜日	金曜カレー	黒カレー	手作りカレー
カレーセット	カレーそば	豆カレー	ダムカレー	ナイスカレー
カレー皿	カレースープ	唐揚げカレー	カツカレー	エビカレー

図 7 得られたラーメン画像の例
(下: カテゴリ名, 左:データセット内の画像, 右:Google 画像検索で上位に現れた画像)

作成したデータセットと, データセットからノイズ画像を除去したもののそれぞれにおいて分類率を測定した. その結果, 分類率は表 4 のように変化した.

表 4 カレー画像における認識可能性評価の変化

データセット	Top-1 分類率	Top-5 分類率
ノイズ除去前	34.3 %	59.3 %
ノイズ除去後	41.4 %	67.2 %

また, 図 8 のような画像が得られた.

6. 考察

Twitter の投稿内容という, 大量のノイズがあるようなデータセットから, ノイズ画像を除去することにより, 分類率を上げることができた. 分類率が上がった原因として, Twitter の投稿内容と異なる画像を除外することができたため, カテゴリごとの特徴を明確にすることができたからであると考えられる. 例えば, これらのラーメン画像は単語検索では同じカテゴリとされたが, 実際には異なっていた画像である. 正しい画像と比較すると, 図のような違いが見られる.

図 8 得られたカレー画像の例
(下: カテゴリ名, 左:データセット内の画像, 右:Google 画像検索で上位に現れた画像)

図 9 ラーメン画像の例
(下: カテゴリ名, 左:正しく判断された画像, 中央:除去された画像, 右: Google 画像検索で上位に現れた画像)

また, 図 10 のカレー画像も同様に, 投稿文にカテゴリ名が含まれていたが, ノイズであるとされ, 除去された画像である. これらの画像も, 正しく分類されている画像と比較すると, カテゴリの特徴を持っていないと考えられる.

また, 今回の 2 つの実験では, カテゴリ数が 50 となっているが, いずれも Top-5 の分類率がおよそ 70 % と, 十分な分類率が得られた. この結果から, Deep Learning を用いることで, 一見似ている画像群に対しても, 詳細画像分類が可能であるということを表していると考えられる.

今後は, 分類率のさらなる向上のために, Bing API などを用いることで学習用画像を集め, その学習結果を用いて Twitter から収集した画像を処理し, 各カテゴリごとの画像を整理することを考えている.

図 10 カレー画像の例
(下: カテゴリ名, 左:正しく判断された画像, 中央:除去された画像, 右: Google 画像検索で上位に現れた画像)

7. 終わりに

我々はより詳細な食事画像データセットの作成を目標としている. 今回の結果から, 分類精度のある程度高いデータセットを作成することが可能になったため, 今後は分類精度をより向上させるとともに, データセットのカテゴリごとに材料やカロリーなどの情報を加えることを考えている.

参考文献

[1] Bian, J. and Yang, Y. and Zhang, H. and Chua, T. S., “Multimedia Summarization for Social Events in Microblog Stream.” IEEE trans. Multimedia, vol. 17, no. 2, pp. 216–228, 2015.

[2] Kaneko, Takamu and Yanai, Keiji, “Event photo mining from Twitter using keyword bursts and image clustering” Neurocomputing,vol. 172, pp. 143–158, 2016.

[3] Yang, L. and Luo, P. and Loy, C. C. and Tang, X., A Large-Scale Car Dataset for Fine-Grained Categorization and Verification, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3973–3981, 2015.

[4] Matsuda, Y. and Hoashi, H. and Yanai, K., Recognition of Multiple-Food Images by Detecting Candidate Regions, Proc. of IEEE International Conference on Multimedia and Expo (ICME), pp. 25–30, 2012.

[5] Jia, Y. and Shelhamer, E. and Donahue, J. and Karayev, S. and Long, J. and Girshick, R. and Guadarrama, S. and Darrell, T., Caffe: Convolutional architecture for fast feature embedding, Proceedings of the ACM International Conference on Multimedia, pp. 675–678, 2014.

[6] Szegedy, C. and Liu, W. and Jia, Y. and Sermanet, P. and Reed, S. and Anguelov, D. and Erhan, D. and Vanhoucke, V. and Rabinovich, A., Going deeper with convolutions, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9, 2015.