

感情に基づいた観光情報の推薦

篠田 広人^{†1} 柴田 有基^{†1}
難波 英嗣^{†2} 石野 亜耶^{†3} 竹澤 寿幸^{†1}

概要: 本研究では、旅行者の感情に基づいた観光推薦システムの構築を目指す。このシステムは、例えば驚きや喜びの観光名所など、感情という観点から観光情報を旅行者に推薦する。このような推薦を実現するため、本研究では、旅行ブログに着目する。旅行ブログ中の画像から、旅行者本人の驚きや幸せの表情を読み取れることがある一方で、ブログ本文中に感情に関する言語表現が使われることもある。そこで本研究では、画像とその対応文を対応付けし、それぞれに対して、感情を推定することで、感情に基づいた観光情報の推薦システムを構築する。提案手法の有効性を確認するため、実験を行った。画像と本文の対応付け実験においては、精度 0.889、再現率 0.013、F 値 0.025 が得られ、感情推定実験においては、CNN を用いることで、精度 0.923、再現率 0.974、F 値 0.948 が得られた。

キーワード: 旅行ブログ, 感情, 観光情報, 推薦

Emotion-based Recommendation of Tourist Spots

HIROTO SHINODA^{†1} NAOKI SHIBATA^{†1}
HIDETSUGU NANBA^{†2} AYA ISHINO^{†3} TOSHIYUKI TAKEZAWA^{†1}

Abstract: We propose a new tourist recommendation system based on travelers' emotions. This system recommends tourist spots to travelers in terms of emotions, such as surprising and joyful tourist spots. To create this emotion-based system, we use travel blogs. Travelers' emotions can be recognized from texts as well as images in the blogs. We constructed a tourist spot recommendation system based on emotions by aligning images and their corresponding sentences, and estimating emotions from them. In order to confirm the effectiveness of our method, we conducted some experiments. In the experiment on aligning images with sentences, we obtained precision of 0.889, recall of 0.013, and F-measure of 0.025. In the experiment on emotion estimation, precision of 0.923, recall of 0.974, and F-measure of 0.948 were obtained.

Keywords: Travel blog, Emotion, Tourist information, Recommendation

1. はじめに

近年、訪日外国人旅行者は増加の一途をたどっており、来たる 2020 年の東京オリンピックに向けて、少なくとも今後数年はこの傾向が続くものと考えられる。訪日外国人旅行者は、旅先の経験を、ツイッターやブログなどのソーシャルメディア上で写真と共に紹介することも少なくない。こうした情報はリアルタイム性が高く、他の旅行者が旅行計画を立てる上で有用なことが多い。そこで本研究では、ソーシャルメディア、特に旅行ブログを情報源とした観光推薦システムの構築を目指す。

従来は、例えば、ショッピング、宿泊といった旅行者の目的別に観光スポットの推薦を行う手法があった。そうした手法に対して、感情に基づいた観光情報推薦は、他の研究にはない新たな観点からの観光情報推薦であると考えられる。感情に基づいた観光情報推薦とは、例えば驚きの観光名所や名所に関する説明など、感情を 1 つの軸として旅行者に観光情報を提示するものである。感情という観点から実際の旅行ブログを眺めると、旅行ブログ中の画像から、

旅行者本人の驚きや幸せの表情を読み取れることがある一方で、ブログ本文中に感情に関する言語表現が使われることもあることが分かる。そこで本研究では、旅行中の画像とその対応文を抽出し、それぞれに対して、感情を推定することで、感情に基づいた観光情報の推薦システムを構築する。こうした推薦ができるようになると、感情が原因となる出来事を抽出できるようになり、よりその地域に関する情報も深く知ることができるようになると考えられる。

本論文の構成は以下の通りである。2 節で関連研究について述べ、3 節では、本研究のシステムの構成と提案手法を説明し、4 節では提案手法の有効性を調べるために行った実験について述べる。5 節では実験結果についての考察を行い、6 節で本稿のまとめを述べる。

2. 関連研究

本節では、関連研究について述べる。2.1 節では観光情報推薦について、2.2 節では、画像・テキストからの感情推定に関する研究について紹介する。

^{†1} 広島市立大学

^{†2} 中央大学

^{†3} 広島経済大学

2.1 観光情報推薦

観光に関する研究は、ツイッターやブログなどのソーシャルメディアを利用して観光に関する情報の抽出や可視化など、様々な研究が存在する。酒井ら[1]は、ユーザーが観光地域を指定したとき、ヒトの嗜好の多様性を考慮して、そのユーザーが過去に撮影した写真の地域情報も組み込んだ協調フィルタリングにより、その観光スポットを推薦するとともに、次に訪れる観光スポットを順次推薦するシステムの構築を目指している。

荒木ら[2]は、ツイッターを利用して、サービスの向上にはどのような情報が有効かの分析・検証を行うための訪日外国人観光行動システムの開発を行っている。上記で示した研究はいずれもツイッターや写真などを用いて観光スポットの抽出や可視化をしているが、本研究では、旅行ブログ本文や旅行ブログ中の画像を対応づけ、それぞれに対して感情を推定することで、感情に基づいた観光情報推薦を提案する。

2.2 画像・テキストからの感情推定

感情に関する研究は、昔から数多く存在するが、この節では画像・テキストを情報源とした感情推定に関する研究について紹介する。まずは、画像の感情推定に関する研究を説明する。Dandilら[3]は、画像から顔を検出し、そこから6種類の感情(“anger”, “fear”, “happy”, “surprise”, “sad”, “neutral”)を推定する研究を行っている。感情推定手法として、CNNを用い、画像やデータセットを増やしていくことで、平均72%の正解率が得られた。また、Minaeeら[4]は、画像中の人物の顔から、attentionを含んだCNNを用いて7種類の感情(“angry”, “disgust”, “fear”, “happy”, “neutral”, “sad”, “surprise”)を推定している。画像からの感情推定実験を行った結果、90%近い正解率が得られた。上記の研究では、CNNを始めとした深層学習手法を用いているが、本研究では、Google Cloud Vision APIの顔検出における感情推定機能を用い、画像からの感情推定を行う。

ここからは、テキストからの感情推定に関する研究を紹介する。Strapparavaら[5]はニューヨークタイムスやBBCといった新聞記事の見出しから6感情(“anger”, “disgust”, “fear”, “joy”, “sadness”, “surprise”)に分類するタスクをチーム別で競わせている。また、Mohammadら[6]は、与えられたツイートから11感情(“anger”, “anticipation”, “disgust”, “fear”, “joy”, “love”, “optimism”, “pessimism”, “sadness”, “surprise”, “trust”)に分類するタスクを示し、チーム別に競わせている。感情分類において使ったデータセットはMohammadら[7]が作ったデータセットであり、クラウドソーシングサービスの一つであるCrowdFlowerを使い、303人の被験者を集め、合計10983件の感情のラベル付けを行っている。上記の研究は、新聞記事の見出しやツイートをを用いて、感情を推定するような研究を紹介したが、本研究ではGhaziら[8]が提供している感情データセットを用いて

感情推定のためのモデルを構築し、実際に旅行ブログに対してモデルを適用することで感情推定を行う。

3. 感情に基づいた観光推薦システムの構築

本研究で開発するシステムは、旅行ブログ中の画像とその画像の対応文を抽出するモジュール、対応文から感情を推定するモジュールから構成される。本章では、3.1節でシステム概要、3.2節で画像と本文の対応付け、3.3節で画像と対応文に対する感情推定について説明する。

3.1 システム概要

システム全体の処理手順として、まず、旅行ブログ中の画像から対応文を抽出した後に、感情推定を行う手法が考えられる。このようなシステムにすると、例えば、ユーザーが驚きの体験がしたいとなると“surprise”に関する観光情報を画像と対応文の形で表示できるようになり、他の地域での“surprise”に関する情報も画像と対応文の形で推薦できるようになると考えられる。

図1にシステムの概要を示す。旅行ブログ中の画像とその画像の対応文を抽出するモジュールについては、3.2節、画像と対応文から感情を推定するモジュールについては、3.3節でそれぞれ説明する。

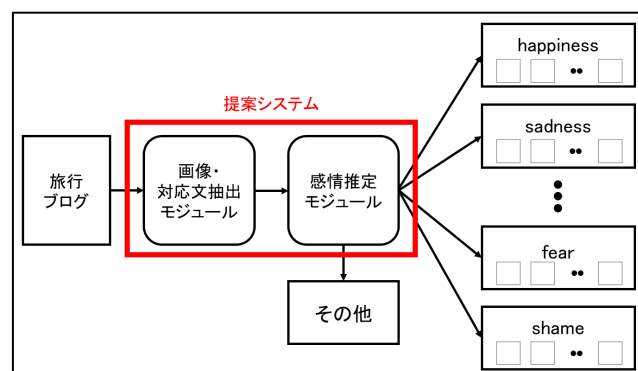


図1: システム概要

3.2 画像と本文の対応付け

本研究では、旅行ブログ中の画像に対して、その画像の対応文を自動抽出する対応文抽出モジュールを構築する。まず、画像に対する対応文を判定するために、人手で画像と対応文の対応付けを行った。画像と本文の対応付け例を図2に示す。図2には、ブログ著者が大阪旅行に行った様子が書かれており、図2の右側にブログ本文の一部を示している。図2の左側の写真から、カスタードやチョコチップが見えてとれ、右側の本文中の赤字で示した文は、“hot custard”や“chocolate chips”という記述が書かれているため、対応文だと考えられる。図2で示した画像と本文の対応付け例を基に以下のような流れで画像と本文の対応付けを行う。

① 旅行ブログから、ブログ本文とブログ画像をそれぞれGoogle Cloud Natural Language API, Google Cloud Vision APIを用いて、ブログ本文抽出とブログ画像解析を行う。

- ② ブログ本文抽出結果とブログ画像解析結果それぞれに対して SWEM[9]を用いてベクトル化を行う。
- ③ ②で作成した文ベクトルと画像解析ベクトルに対して、コサイン類似度で類似度を図る。
- ④ ③で求めたコサイン類似度と自分で設定したペナルティをかけた値のうち、上位1件を対応文とする。



図 2：画像と本文の対応付け例

(<https://www.travelblog.org/Asia/Japan/Osaka/Osaka/Shinsaibashi/blog-771187.html>)

ここからは、上記の①から④にある SWEM, コサイン類似度, ペナルティについて説明する。

SWEM

SWEM は、word2vec[10]などを始めとした単語埋め込みを利用して、ベクトルを作成する手法である。SWEM には、大きく分けて 3 つの手法があり、SWEM-average は、各次元の平均値を取ることで、ベクトルが生成できるようになっている。一方で、SWEM-max は、各次元の最大値を取ることでベクトルが生成されるようになっている。最後に、SWEM-concat の説明を行う。図 3 に SWEM-concat の概要を示す。

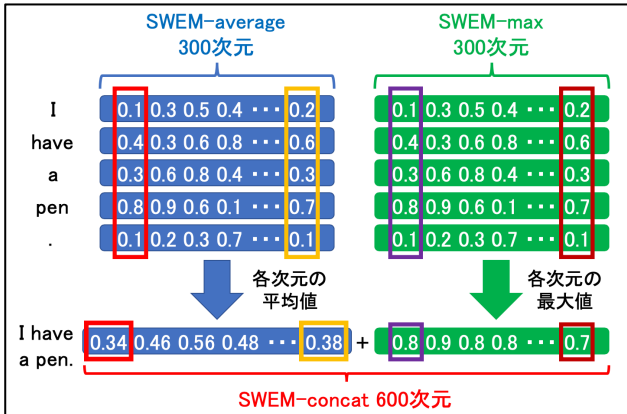


図 3：SWEM-concat 概要

SWEM-concat は、上記で説明した SWEM-average と SWEM-max を結合させてベクトル化した手法である。本研究では旅行ブログ中の文と画像の解析結果をそれぞれ SWEM-concat でベクトル化している。

コサイン類似度

コサイン類似度とは、2 つのベクトル間の類似度を測る方法である。1 に近ければ近いほど、2 つのベクトル間の類似度が高いということになる。2 つのベクトルには、SWEM-concat で作成した文ベクトルと画像解析ベクトルを用いる。

文ベクトルを \vec{p} 、画像解析ベクトルを \vec{q} とすると下記の式(1)で表される。

$$\cos(\vec{p}, \vec{q}) = \frac{\vec{p} \cdot \vec{q}}{|\vec{p}| |\vec{q}|} = \frac{\sum_{i=1}^{|\vec{V}|} p_i q_i}{\sqrt{\sum_{i=1}^{|\vec{V}|} p_i^2} \cdot \sqrt{\sum_{i=1}^{|\vec{V}|} q_i^2}} \quad (1)$$

ペナルティ

本研究では、より正確な対応文抽出のために、ペナルティを設定した。人手で対応付けた 256 件の画像と対応文の正解データから、基準文と対応文の分布をグラフにまとめた。そのグラフを図 4 に示す。

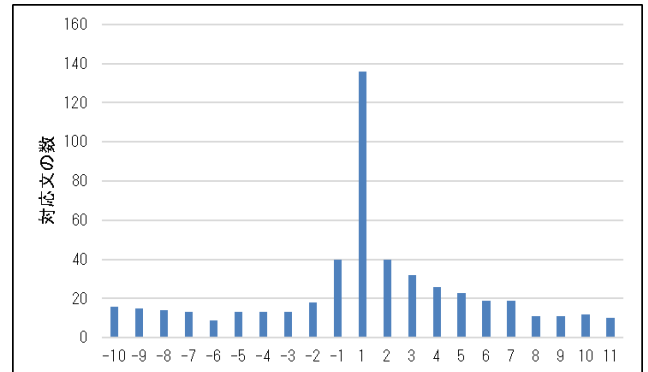


図 4：基準文と対応文の分布

図 4 の横軸の“1”が基準文を表しており、この“1”は、画像の直後 1 文を表している。そこから正の方向に例えば、“2”は、画像の直後 2 文というようになり、一方で負の方向では例えば、“-1”は画像の直前 1 文を表している。このことから図 4 を見ると、画像の直後 1 文の場合の対応文の数が特に多いことがわかる。この分布を基にしたペナルティの設定の例を図 5 に示す。

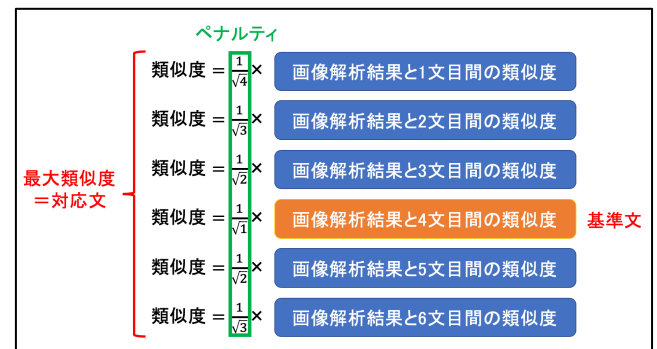


図 5：ペナルティの設定の例

図 5 は、4 文目を基準文とした場合を示しており、そこから上下に分母のルートに“+1”し、画像解析結果と文の類似度をかけることで、画像と文の距離を考慮した類似度を求めることができる。そして求めた類似度の中で最大の類似度を持った文を対応文とする。

3.3 旅行ブログにおける感情推定

本研究では、3.2 節で説明した画像と対応文を対象に、感情を推定する感情推定モジュールについて説明する。本研究では、旅行ブログ中の画像と本文の両方から感情推定を行う。3.3.1 節に画像からの感情推定、3.3.2 節で対応文から

の感情推定について説明する。

3.3.1 画像からの感情推定

画像から感情を推定する方法として、Google Cloud Vision API の顔検出の機能を使用する。顔検出では、画像中の顔の位置を検出することができるほか、顔の要素、例えば目の位置なども検出することができる。本研究では、顔検出機能の中で画像中の人物の顔から感情を推定する機能を使用し、画像からの感情推定を行う。図 6 に写真の感情推定結果を示す。

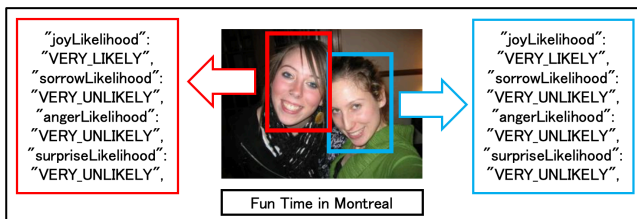


図 6：画像の感情推定結果

(<https://www.travelblog.org/North-America/Canada/Alberta/Calgary/blog-76853.html>)

図 6 のように、写真に複数の人物が写っている場合は、人数分の感情推定結果が返される。そして、推定できる感情の種類としては、“joy”、“sorrow”、“anger”、“surprise” の 4 種類で表され、各感情の下に感情の度合い “Likelihood” を示している。感情の度合いの種類は表 1 のようになっている。図 6 の赤枠、水色で囲った人物両方の感情推定結果は、“joy” の可能性が非常に高いことがわかり、また、その他の感情は、可能性として非常に低いことがわかる。本研究では、各感情の度合いが “LIKELY” あるいは、“VERY_LIKELY” であるならば、その感情が生起しているとみなす。よって、図 6 の場合だと、“joy” が生起していると考えられる。

表 1：感情の度合いの種類とその意味

感情の度合い	意味
UNKNOWN	不明
VERY_UNLIKELY	非常に低い
UNLIKELY	低い
POSSIBLE	おそらくそう
LIKELY	高い
VERY_LIKELY	非常に高い

3.3.2 対応文からの感情推定

対応文から感情推定する上で、まずは、本研究で使った感情データセットの説明をする。このデータセットはフレーム意味論に基づくオンライン英単語情報プロジェクト FrameNet の要素のうち感情の生起原因に着目したデータセットである。このデータセットの感情の種類は、“happiness”、“anger”、“sadness”、“fear”、“surprise”、“shame”、“disgust” の 7 種類である。一例として、図 7 に “happiness” と判定した文例、図 8 に “surprise” と判定した文例をそれ

ぞれ示す。

George Grindal was pleased that he could turn to the bedside cabinet and take up the feeding glass containing orange juice.

図 7：“happiness” と判定した文例

The boy bowed again, enjoying her astonishment in the same way he had enjoyed the applause of the T'ang earlier that day when he had played Tsu Tiao.

図 8：“surprise” と判定した文例

本研究では、感情推定する手法として、深層学習に基づいた手法である CNN[11] を用いて上記で示した 7 種類の感情を推定する。図 9 に CNN のモデルを示す。まず、文を単語に区切り、それらを単語埋め込みの列に変換する。次にそのデータを文の長さ $n \times$ 単語埋め込みの次元 k でチャンネルを作成し、特徴マップを生成する。この時、ウィンドウサイズをいくつか指定することで、複数の特徴マップを生成している。その後、畳み込みを行い、Max-pooling によって特徴マップの情報を圧縮して、特徴量を得る。最後に Full Connection 層において、前の層の情報を全て結合して softmax によって正規化することで、最終的に分類先を決定する。本研究では、多クラス分類を行う上で、各感情において生起したかどうかの 2 値分類を拡張することで実現する。

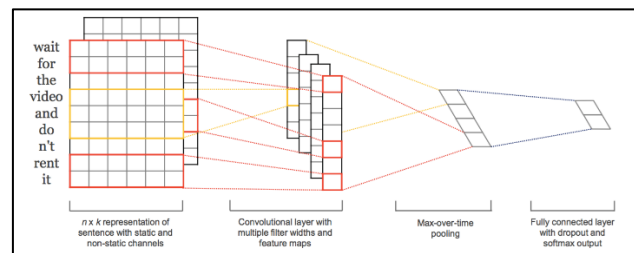


図 9：CNN[11] のモデル

4. 実験

本節では、本研究で行った評価実験について説明する。

4.1 節で画像と本文の対応付けに関する実験、4.2 節で感情データセットを用いた感情推定に関する実験についてそれぞれ説明する。

4.1 画像と本文の対応付け

実験データの収集方法について説明する。最大の旅行ブログサイトの 1 つである TravelBlog のデータを用いて、任意に選んだ旅行ブログの中の 256 画像を対象に、該当する対応文を手手で付与した。表 2 に大陸別の旅行ブログ件数、対象画像数、対応文数を示す。実験データに対して、3.2 節で説明した SWEM を旅行ブログ中の文と画像の解析結果に対して適応し、文と画像解析結果間の類似度の上位 1 件を対応文として評価を行った。本実験では、単語埋め込みに適用する学習済みモデルとして、Google News のデータセットで学習されたモデルを使用した。このモデルは、300

万語の 300 次元ベクトルが含まれたモデルになっている。

表 2: 4.1 節の実験で使用する旅行ブログデータ

大陸	ブログ件数	対象画像数	対応文数
North America	95	170	353
Europe	43	69	208
Asia	11	17	79
合計	149	256	640

比較手法には、画像直後 1 文を対応文とする手法を用いた。評価尺度には、精度、再現率、F 値を用いた。計算式は以下の通りである。

$$\text{精度} = \frac{\text{システムにより抽出した正解数}}{\text{システムにより抽出した数}} \quad (2)$$

$$\text{再現率} = \frac{\text{システムにより抽出した正解数}}{\text{人手により判定した正解数}} \quad (3)$$

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

4.2 感情データセットを用いた感情推定

実験データに関しては、3.3.2 節で述べた 7 種類の感情データセットを用いて実験を行った。表 3 に各感情のデータ件数を示す。本研究では、提案手法として、3.3.2 節で説明した CNN を使用した。CNN のパラメータとして、batch size を 7、単語埋め込みの次元を 250、dropout を 0.049、epoch 数を 9 とし、最適化アルゴリズムには、rmsprop を用いた。

表 3: 4.2 節の実験で使用する感情データ

感情	合計
happy	479
sadness	575
surprise	213
disgust	95
anger	483
fear	423
shame	145
合計	2413

また、比較手法として、SVM[12]、fastText[13]を適用した。SVM のパラメータは、コストを表すパラメータ C を 100、カーネルは、RBF カーネルを選択した。fastText のパラメータは、epoch 数を 100、単語埋め込みの次元を 300 とした。評価方法は、5 分割交差検定を行い、評価尺度は精度、再現率、F 値を用いた。

5. 実験結果と考察

4 章での実験について結果を示し、それぞれについて考察を行う。5.1 節で画像と本文の対応付けに関する実験について、5.2 節で感情データセットを用いた感情推定に関する実験について考察を行う。

5.1 画像と本文の対応付け

4.1 節で説明した実験の結果を表 4 に示す。本研究では、より正確な対応文抽出を行うため、類似度に対して閾値を設定し、精度向上を図った。表 4 における SWEM-concat は、閾値 0.89 の場合の結果を示している。表 4 から、SWEM-concat は、比較手法と比べて、再現率と F 値が低いものの、精度においては、比較手法よりも良い性能を示した。これにより、精度の観点から SWEM-concat の有効性が確認できた。

表 4: 画像と本文の対応付けの実験結果

	精度	再現率	F 値
SWEM-concat	0.889	0.013	0.025
画像直後 1 文	0.531	0.213	0.304

ここからは、手法 SWEM-concat の主な誤分類例を示す。実際に対応文と判定されなかった例を図 10 に示す。

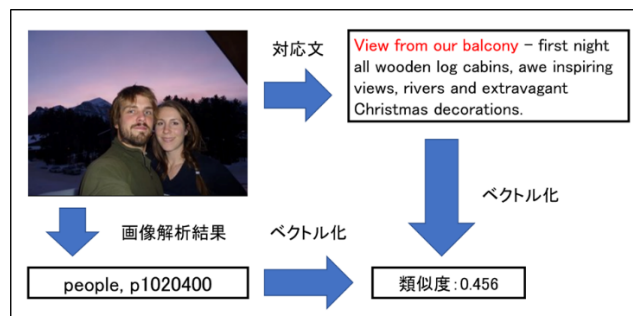


図 10: 対応文と判定されなかった例

([https://www.travelblog.org/North-](https://www.travelblog.org/North-America/Canada/Alberta/Lake-Louise/blog-461117.html)

[America/Canada/Alberta/Lake-Louise/blog-461117.html](https://www.travelblog.org/North-America/Canada/Alberta/Lake-Louise/blog-461117.html))

図 10 は、左下に画像解析結果を表しており、右上に画像に対する対応文を表している。画像と対応文の中身を見てみると、画像は、人物 2 人と外の風景を写した写真であり、対応文からは、赤字で“View from our balcony”と景色を表す表現があることから画像に対する対応文であることがわかる。では、なぜ対応文と判定されなかったのかの理由として、画像解析結果で得られた単語の少なさが挙げられる。画像解析結果と文間の類似度は、一般的に、2 つの関係が近ければ近いほど、高い値が出るようになっているが、今回の画像解析結果は、図 10 にあるように、“people”と“p1020400”の 2 単語しかないため、類似度をとっても 0.456 と高い値が出ないことがわかった。よって、画像解析結果があまり取れなかった場合だと、対応文が抽出されにくいことが考えられる。このように画像解析結果が 2 単語以下で、対応文と判定されなかった例は他にも 17 件見られた。このように画像の解析結果が少ない場合は、イメージキャプションといった方法で、画像の説明文を生成することで、より正確に対応文を推定できると考えられる。

5.2 感情データセットを用いた感情推定

4.2 節で説明した実験結果を表 5 に示す。本研究では、もれなく感情を推定することが望ましいと考え、再現率を

重視する。表 5 から、精度は fastText が、再現率は CNN がそれぞれ良いことがわかる。F 値に関しては、SVM は 0.747 ポイント、fastText は 0.542 ポイント、CNN は 0.948 ポイントであることがわかり、SVM と CNN を比較すると、CNN が 0.201 ポイント、fastText と CNN を比較すると、CNN が 0.406 ポイント上回ったことがわかる。これより、再現率、F 値の観点から、感情データセットにおいて CNN の有効性が確認できた。

表 5：各手法における感情推定結果

	精度	再現率	F 値
CNN	0.923	0.974	0.948
SVM	0.685	0.821	0.747
fastText	0.974	0.376	0.542

ここからは、CNN における主なエラー例を示す。CNN の感情別に精度、再現率、F 値を求めたところ、特に“disgust”の精度が 0.647 という低い結果になったことについて考察を行う。図 11 に誤って“disgust”と判定された例を示す。“disgust”と誤って判定した原因として図 11 中の太字で示してあるように、“distrustful”という単語が挙げられる。“distrustful”は、もともと疑い深いという意味であり、嫌悪や嫌気を表す“disgust”におおよそ近い意味であると分かる。しかし、下線で表される“resentful”や“angry”は怒りの意味を表す単語であることがわかる。正解データを見ると図 11 は、“anger”と付与された文であるが、“distrustful”という単語があるため、“disgust”と推定されてもよい文だと考えられる。このように、正解データにある感情のほかにも図 11 のように“disgust”にも推定されてもよい例がほかにも 46 件見られた。

I have grown up to be **distrustful** of men, doubtful of my own goodness, extremely resentful and explosively angry.

図 11：誤って“disgust”と判定された例

6. おわりに

本研究では、旅行ブログを対象に、画像とその対応文を抽出し、抽出された画像と対応文それぞれに対して、感情推定を行うことで、感情に基づいた観光情報推薦システムを構築した。画像と本文の対応付け実験において、提案手法は再現率 0.013 ポイントと低いものの、精度が 0.889 ポイント得ることができ、その点で比較手法よりも上回ったことで有効性を確認することができた。また、感情データセットを用いた感情推定実験に関しては、提案手法である CNN が精度 0.923 ポイント、再現率 0.974 ポイント、F 値 0.948 ポイントと全体的に高い値をとることができ、再現率、F 値においては、fastText を上回ることができ、有効性を確認できた。

参考文献

- [1] 酒井 勇人, 熊野 雅仁, 木村 昌弘. Flickr データに基づいたインタラクティブ観光スポット推薦システム. 人工知能学会第 14 回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. 2016, p. 24-29.
- [2] 荒木 徹也, 加藤 大受, 吉田 千鶴, 廣田 雅春, 遠藤 雅樹, 石川 博. ソーシャルビッグデータを活用した観光行動分析の実証実験. 第 10 回データ工学と情報マネジメントに関するフォーラム(DEIM 2018). 2018.
- [3] Emre Dandil, Rıdvan Özdemir. Real-time Facial Emotion Classification Using Deep Learning. DATA SCIENCE AND APPLICATIONS. 2019, p. 13-17.
- [4] Shervin Minaee, Amirali Abdolrashidi. Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network. arXiv:1902.01019. 2019.
- [5] Strapparava, Carlo and Mihalcea, Rada. Semeval-2007 task 14: Affective text. Proceedings of the 4th International Workshop on Semantic Evaluations. 2007, p. 70-74.
- [6] Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. SemEval-2018 Task 1: Affect in Tweets. Proceedings of International Workshop on Semantic Evaluation (SemEval-2018). 2018, p. 1-17.
- [7] Saif M. Mohammad and Svetlana Kiritchenko. Understanding Emotions: A Dataset of Tweets to Study Interactions between Affect Categories. Proceedings of the Eleventh International Conference on Language Resources and Evaluation, 2018.
- [8] Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. Detecting Emotion Stimuli in Emotion-Bearing Sentences. Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics. 2015, p. 152-165.
- [9] Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, Lawrence Carin. Baseline Needs More Love: On Simple Word-Embedding-Based Models and Associated Pooling Mechanisms. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 2018, p. 440-450.
- [10] Quoc Le, Tomas Mikolov. Distributed representations of sentences and documents. Proceedings of the 31st International Conference on International Conference on Machine Learning. 2014, p. 1188-1196.
- [11] Yoon Kim. Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014, p. 1746-1751.
- [12] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. Machine Learning. 1995, vol. 20, no. 3, p. 273-297.
- [13] Joulin, Armand and Grave, Edouard and Bojanowski, Piotr and Mikolov, Tomas. Bag of Tricks for Efficient Text Classification. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. 2017, p. 427-431.