ツイート発言の座標またはグリッドの予測基盤の開発

大西駿太朗 [1]　矢田竣太郎 [1]　若宮翔子 [1]　荒牧英治 [1]
[1] 奈良先端科学技術大学院大学
{shuntaro.onishi.or3,s-yada,wakamiya,aramaki}@is.naist.jp

概要

Twitter をはじめとするソーシャルネットワークサービスの分析は．都市デザイン，災害・疾病把握や社会調査などに多く用いられてきた．しかし，位置情報の付与されたツイートは全体の 1%未満と報告されている．そこで．位置情報を推定する研究が行われている．これらの研究は入力の形式と出力の形式で分けることができる．本稿では入力をツイート発言，出力を緯度経度および地域メッシュとし，位置推定モデルを作成した．作成したモデルのボトルネックを考察するとともに，今後開発されるであろう位置情報を用いたさまざまなサービスの基盤となる位置推定 Web API を構築したので報告する．

1　はじめに

Twitter はソーシャルセンシングによく用いられている．ソーシャルセンシングとは既存のセンサでは捉えることができない情報を捉えるため，ソーシャルネットワークサービスのユーザをセンサとみなし，実世界の現象を観測することである [1]．ソーシャルセンサは，これまで，都市デザイン，災害・疾病把握や社会調査などに多く用いられてきた．都市デザインにおいては，土地利用やランドマークの特定など，街並みの特徴付けの可能性が検討されてきた [2, 3, 4]．災害・疾病予測では，対災害 SNS 情報分析システム [1)], 台風軌道の予測 [5]，インフルエンザの流行把握 [6] や COVID-19 のクラスタ検出 [7] が行われてきた．社会調査における利用も多く，選挙，株価，政治的思考の違い [8] などの研究や，人々の気分を推定し，時間変化や地域差を可視化する研究 [9, 10] も行われている．

このように，ソーシャルセンシング研究の多くにおいて位置情報は必須である．しかし，Twitter で得られる位置情報は十分でないことが多い．例えば，ツイートの投稿者の位置情報はユーザが任意に付与するものであり，位置情報の付与されているツイートは全体の 0.4%ほどであると報告されている [11]．このため，多くのソーシャルセンシングサービスは，メタ情報やテキスト中のランドマーク表現など，何らかの方法を用いて位置情報を推定することでカバレージを上げている．位置情報を補完するためのツイートの位置予測研究も，この 10 年ほど活発に行われてきた．中でも，W-NUT2016 の Twitter の位置情報予測シェアードタスク [2)] [12] のデータは，現在でもベンチマークとしてしばしば利用されている．

位置推定タスクは，入力の形式と出力の形式の 2 つの軸で整理が可能である．

入力の形式: 位置推定の入力としては，発言者・ユーザの一連の文書か，単一文書（発言メッセージ，ツイート）かの 2 つのレベルがある．本稿では，前者の発言者の居住地を予測することをユーザレベル，後者の発言メッセージが投稿された位置を予測することをメッセージレベルと呼ぶ．

出力の形式: 位置推定の出力は，緯度経度，県や都市など様々な形式がある．等間隔のグリッドに分割することや [13]，投稿の多い地域では細かいグリッド，投稿の少ない地域では大きいグリッドを生成する手法が提案されている [14]．またメッセージレベルにおいて，その推定の不確実性を表現するために，出力位置の確率分布を出力する場合もある [15]．代表的な等間隔のグリッドとして，地域メッシュがある．地域メッシュは緯度経度から算出され，行政区画の変更にも左右されないという利点から，統計調査によく用いられる．

このように考えると，例えば，前述のシェアードタスクの距離ベースの評価によるベストシステム [16] は，メッセージレベルを入力とし，緯度経度を出力すると捉えることができる．

位置情報を利用した多くのサービスが提案されているが，位置情報を推定するために，それぞれ個別

1） https://disaana.jp/rtime/search4pc.jsp

2） https://noisy-text.github.io/2016/geo-shared-task.html

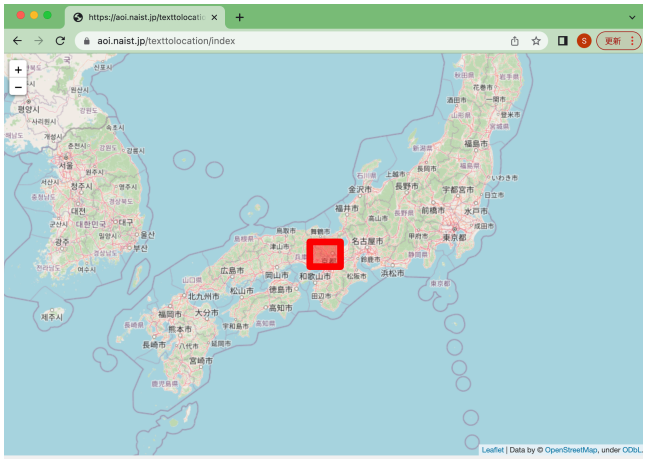


図 1: メッセージレベルの位置予測 Web API.「琵琶湖なう」に対して予測した地域メッシュ（1 次メッシュ）を出力した例.

の実装を行っていたり，ブラックボックス化している場合も多い．そこで，我々は，標準的な位置予測基盤があれば，多くのサービス開発を簡素化できるだけでなく，精度比較や再現性検証においても重要な役割を果たすと考えている．ユーザレベルにおいては，デモグラフィック解析，またはソーシャルリスニングツールとして商用サービスがあるものの，メッセージレベルの推定については，手法の透明性が高いスタンダードなサービスが存在しない．本稿では，今後開発されるであろうサービスの基盤となる座標と等間隔グリッド（地域メッシュ）の両方に対応したメッセージレベルの位置推定 Web API[3] を構築したので報告する．図 1 に Web API の例を示す．「琵琶湖なう」というメッセージレベルの入力に対し，地域メッシュを（1 次メッシュ）を予測して地図上に出力した結果である．

2 データ

ツイート投稿者の日本国内における位置を予測するモデルを構築するために，日本における位置情報付きツイートを利用する．Twitter API v2 の Academic Research アクセスで “place_country:jp” のクエリを指定し，2022 年 7 月 1 日から同月 31 日まで，1 時間あたりおよそ 1500 件，合計 1,111,576 件の位置情報 (GeoTag) 付きツイートを取得した．

GeoTag には，特定の緯度・経度を示す “Point” と特定のエリアを表す “Place” があり，それぞれ一意に定まる place_id と関連付けられている．この place_id に基づき，対応する緯度・経度を各ツイートに付与する．なお，GeoTag が特定のエリアを表す “Place” の場合は，そのエリアの南端と西端および北端と東端の緯度・経度を付与する．

次に，各ツイートの緯度・経度に基づき，地域メッシュコードを付与する．地域メッシュは，都道府県よりも粒度が小さく，行政区画の変更にも左右されないという利点から，統計調査によく用いられる地域分割である．1 次メッシュは地域メッシュのなかで最も粒度の大きい分割で，1 辺が約 80km である．2 次メッシュは 1 次メッシュを 8 × 8 マスに分割したもので 1 辺が約 10km である．今回は，各ツイートの緯度・経度に基づき，1 次メッシュコードと 2 次メッシュコードを付与する．

本文に含まれる URL，メンションおよび半角・全角スペースは除去した．

3 提案手法

本研究では，緯度・経度を回帰問題として推定するモデル（**緯度経度推定モデル**）と，地域メッシュを分類問題として推定するモデル（**メッシュ推定モデル**）を構築する．メッシュ推定モデルはさらに，1 次メッシュ推定モデルと 2 次メッシュ推定モデルとで別個に作成する．入力テキストをエンコードするモデルとして Bidirectional Encoder Representations from Transformers (BERT) [17] を日本語コーパスで事前学習したモデル[4] を採用した．緯度経度推定モデルおよび 1 次メッシュ推定モデルでは日本全体を予測範囲とする．一方，2 次メッシュ推定モデルでは予測範囲を東京（1 次メッシュコード: 5339）に限定し，計算負荷が高くなり過ぎない程度の予測メッシュ数に抑えた．1 次メッシュ推定モデルで 140，2 次メッシュ推定モデルで 61 のメッシュを予測する．

緯度経度推定モデルの目的関数と評価関数は，ともに緯度経度の平均二乗誤差とする．なお，地球が完全な球でないことによって起きる実際の距離のずれは補正しない．また．メッシュ推定モデルの目的関数と評価関数には cross entropy を用いた．

4 結果

位置情報付きツイートデータを 8（訓練）: 1（検証）: 1（テスト）の割合で分割し，構築した緯度経

3） https://aoi.naist.jp/texttolocation/

4） https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking

表 1: 各モデルに対する test loss と test accuracy

推定モデル	Test loss	Test accuracy
緯度経度	24	–
1 次メッシュ	3.3	0.21
2 次メッシュ	2.9	0.27

度推定モデルとメッシュ推定モデルの精度を評価した．なお，ツイートの多くは東京近辺および人口の多い都市で投稿されるため，データの不均衡が生じる．そのため，1 次メッシュ推定モデルにおいては同一メッシュ内のデータを 300,000 件以内に制限し，1,111,576 件の位置情報付きツイートを用いた．また 2 次メッシュ推定モデルについては東京都の大部分を含む地域メッシュ (1 次メッシュコード: 5339) を対象としたため，323,234 件の位置情報付きツイートを用いた．学習条件として最適化手法は Adam，学習率は 1.0×10^{-5}，エポック数は 5，バッチサイズは 2 とした．表 1 に緯度経度およびメッシュ推定のテストデータ推定結果を示す．

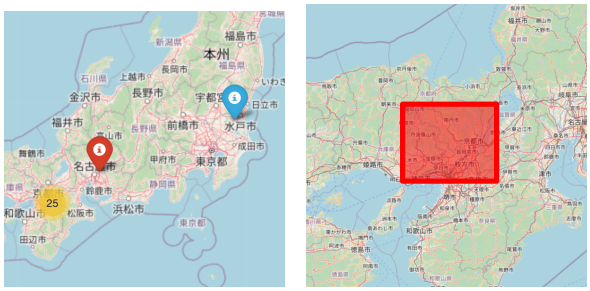
5 考察

5.1 緯度経度推定モデル

日本周辺の緯度，経度 1° あたりの距離は約 90km であることから，$\sqrt{\text{test loss}} = \sqrt{24} \fallingdotseq 4.9$ は非常に大きなずれと言える．この原因を分析するために，緯度経度推定モデルによる「奈良公園なう」の予測位置と，学習データにおいて “奈良公園” を含むツイートを地図上に可視化した（図 2a）．赤色のピンがモデルの出力であり，黄色の円で示された範囲に 25 件の学習データがあり，青色のピンに 1 件学習データが存在した．実際には奈良公園は黄色の円内に存在する．青色のピンのツイート本文は「奈良公園の鹿みたいに白鳥がいます」であり．この投稿においては，投稿者が実際にいる地点と言及された地点が一致していない．この地点のずれは A/B 問題 [18] と呼ばれている．この影響により予測位置が東側に大きく引っ張られている．

5.2 メッシュ推定モデル

1 次メッシュ推定モデルにおける「奈良公園なう」の出力結果を図 2b に示す．緯度経度推定モデルと異なり，奈良公園を含む地域メッシュを正しく出力している．これは緯度経度推定モデルが回帰問題を



(a) 緯度経度推定モデル　　**(b)** 1 次メッシュモデル

図 2: 「奈良公園なう」の予測結果，(a) 赤色のピンは緯度経度推定モデルによる「奈良公園なう」の予測位置，青色のピンと黄色の円は「奈良公園」を含む train データの位置を示す

扱っているのに対し，1 次メッシュ推定モデルはクラス分類問題を扱ったためであると考えられる．

続いて 1 次メッシュ推定モデルにおける本文に含まれる地名の数と推定精度の関係を分析した．具体的には，Spacy の “ja_core_news_md”[5] の GPE ラベルに基づき，ツイート中の地名を抽出した（表 3）．テストデータのうち，地名を含まないものが大半（約 85%）を占め，地名を 1 つ含むものが約 11%，地名を複数含むものが約 4%であった．

同様に 2 次メッシュ推定モデルにおいても地名の有無と予測精度の関係を分析した．表 2 の (1)(4)(7)(10) は地名を含まない例，(2)(5)(8)(11) は地名を 1 つ含む例，(3)(6)(9)(12) は地名を複数含む例である．(1) のように地名を含まないツイートの予測精度は低い．このようなツイートの位置を予測することは人間にも難しいので妥当な結果と言える．地名を複数含むときは，地名を 1 つしか含まないときと比べ予測精度が高い．その要因として，(3) の旭川市と北海道のように包含関係にある地名が出現する，または同一の地名が繰り返し出現することが挙げられる．また同一の地名が繰り返し出現するツイートは，ハッシュタグにより地名が追加されていることが多かった．一方で，(12) のように位置的に全く異なる地点を挙げている場合は予測が難しい．

また，1 次メッシュ推定モデルにおける地名を含まない正解データは 7,305 件であった．地名以外の地点を特定できる要因を調べるために該当データを分析したが，有益な分析結果は得られなかった．

このように地名を含まないツイートが多いことが，予測精度が向上しない一因であると考える．加えて，隣接する地域メッシュにおいて，ツイート内容に大きな差が見られないことが，メッシュ推定モ

5) https://spacy.io/models/ja

表 2: メッシュ推定モデルにおける正解例と誤り例．(1)-(6) は 1 次メッシュ推定モデルによる結果，(7)-(12) は 2 次メッシュ推定モデルによる結果．太字は固有表現抽出により抽出された地名を表す．

ID	正解（おおよその位置）	出力（おおよその位置）	本文
(1)	5235（大阪府）	5235（大阪府）	あんたらはなーんもせんやんけ。
(2)	4931（大分県）	4931（大分県）	**大分**に来たら待ち合わせは大分駅のあの場所に#夜のイチスタ
(3)	6542（北海道旭川市）	6542（北海道旭川市）	**旭川**であれ**北海道**に来たからには寿司を食わねばなりません。
(4)	5338（山梨県）	5436（石川県）	サントリー白州蒸溜所ツアー樽貯蔵庫に足を踏み入れた瞬間の香り…
(5)	5135（奈良県）	5238（静岡県）	当券があるみたいなのでそろそろ**難波**寄ってから現場に向かいますか
(6)	5133（瀬戸内海）	5135（京都府）	**九州**から「さくら」に乗って新**大阪**から**大和西大寺**へ ww
(7)	533965（埼玉県さいたま市）	533965（埼玉県さいたま市）	雨の中ですが、着々と準備が進んでいます！
(8)	533946（東京都台東区）	533946（東京都台東区）	**上野**着弾！！！ [URL]
(9)	533945（東京都新宿区）	533945（東京都新宿区）	新宿 1352 発東京メトロ丸ノ内線**池袋**行き**東京**まで#ノア乗車録
(10)	533931（神奈川県相模原市）	533945（東京都新宿区）	セミリタイアしてしまったんですかね
(11)	533904（神奈川県横浜市）	533913（神奈川県綾瀬市）	**神奈川**の結果…
(12)	533944（東京都武蔵野市）	533945（東京都新宿区）	**札幌**が重要現場被りで行けないから次の**大阪**を見たら終了か

表 3: 地名の有無と推定精度の分析．地名を含むツイートの方が Accuracy が高いことがわかる．

	ツイート数		Accuracy	
	1 次	2 次	1 次	2 次
すべて	65,504	32,323	0.21	0.27
地名を含まない	55,933	27,319	0.13	0.21
地名を含む（１つ）	7,479	3,833	**0.54**	**0.57**
地名を含む（複数）	2,092	1,171	**0.63**	**0.70**

デルの正解率が 20%から 30%にとどまった原因であると考えられる．これは特に 2 次メッシュ推定モデルにおいて顕著である．例えば，観光地において 10km 先に有名なスポットがある場合，もしくは，80km 離れた地点であれば方言など，ツイートに言語的な差が現れる可能性がある．しかし，街中や住宅街において言語的な差が現れる可能性は低く，依然として難しい問題であると言える．

6 おわりに

本研究では Twitter API から取得した位置情報付きツイートを用いて位置推定モデルを作成した．同時に Web API として公開した．緯度経度推定モデルでは A/B 問題により予測が大きくずれること，また地域メッシュモデルでは地名を含まないツイートが多いために位置推定が困難であることを示した．

本研究で提案した手法の精度は，緯度経度モデルで平均約 441km ($\sqrt{testloss} * 90 = \sqrt{24} * 90 \fallingdotseq 441$)，メッシュ推定モデルでは Accuracy が 20%から 30%であり，位置情報をもとにしたサービスや研究への応用のためには，さらなる精度の向上は不可欠である．ツイート本文には地名が含まれていないものが多く，人間にも推定が難しいと考えられる．ユーザのプロフィール情報や投稿時間など他の情報を用いることにより精度が改善する可能性がある．

また出力の提示方法についても検討の余地がある．例えば「朝起きるのが辛かった」というツイートがあったとき，人間であれば予測不可能な問題として扱うことになる．しかし，提案モデルは必ず特定の位置を出力するため，位置推定における曖昧性を適切に表すことができていない．特定の地域メッシュではなく，すべての地域メッシュに対する確率を出力する方法も検討したい．また，そもそも人間が正確に位置予測できるツイートの割合や典型的な誤りの粒度（地方レベル，都道府県レベル，ランドマークレベルなど）はあまり知られていない．人間にも予測可能なツイートのみを用いてモデルを評価し，課題を検討したい．

謝辞

本研究は，JSPS 科研費 JP22H03648, JP22K12041, JST SICORP JPMJSC2107，Yahoo 株式会社共同研究費の支援を受けたものである．

参考文献

[1] Kenta Sasaki, Shinichi Nagano, Koji Ueno, and Kenta Cho. Feasibility Study on Detection of Transportation Information Exploiting Twitter as a Sensor. In **Proceedings of the International AAAI Conference on Web and Social Media**, 2012.

[2] Vanessa Frias-Martinez, Victor Soto, Heath Hohwald, and Enrique Frias-Martinez. Characterizing Urban Landscapes Using Geolocated Tweets. In **2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing**, pp. 239–248, 2012.

[3] Ryong Lee, Shoko Wakamiya, and Kazutoshi Sumiya. Urban area Characterization Based on Crowd Behavioral Lifelogs over Twitter. **Personal and Ubiquitous Computing**, Vol. 17, p. 605–620, 2013.

[4] Panote Siriaraya, Yuanyuan Wang, Yihong Zhang, Shoko Wakamiya, Péter Jeszenszky, Yukiko Kawai, and Adam Jatowt. Beyond the Shortest Route: A Survey on Quality-Aware Route Navigation for Pedestrians. **IEEE Access**, Vol. 8, pp. 135569–135590, 2020.

[5] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake Shakes Twitter Users: Real-Time Event Detection by Social Sensors. In **Proceedings of the 19th International Conference on World Wide Web**, WWW '10, p. 851–860, 2010.

[6] Eiji Aramaki, Sachiko Maskawa, and Mizuki Morita. Twitter Catches The Flu: Detecting Influenza Epidemics using Twitter. In **Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing**, pp. 1568–1576, 2011.

[7] Shohei Hisada, Taichi Murayama, Kota Tsubouchi, Sumio Fujita, Shuntaro Yada, Shoko Wakamiya, and Eiji Aramaki. Surveillance of Early Stage COVID-19 Clusters Using Search Query Logs and Mobile Device-Based Location Information. **Scientific Reports**, Vol. 10, p. 18680, 2020.

[8] Adam Badawy, Emilio Ferrara, and Kristina Lerman. Analyzing the Digital Traces of Political Manipulation: The 2016 Russian Interference Twitter Campaign. In **2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)**, pp. 258–265, 2018.

[9] Alan Mislove, Sune Lehmann, Yong-Yeol Ahn, Jukka-Pekka Onnela, and J. Niels Rosenquist. Pulse of the Nation: U.S. Mood Throughout the Day inferred from Twitter, 2010. https://www.ccs.neu.edu/home/amislove/twittermood/.

[10] Mark E. Larsen, Tjeerd W. Boonstra, Philip J. Batterham, Bridianne O'Dea, Cecile Paris, and Helen Christensen. We Feel: Mapping Emotion on Twitter. **IEEE Journal of Biomedical and Health Informatics**, Vol. 19, No. 4, pp. 1246–1252, 2015.

[11] ツイッターの空間分析. 古今書院, 2019.

[12] Bo Han, Afshin Rahimi, Leon Derczynski, and Timothy Baldwin. Twitter Geolocation Prediction Shared Task of the 2016 Workshop on Noisy User-generated Text. In **Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)**, 2016.

[13] Benjamin Wing and Jason Baldridge. Simple Supervised Document Geolocation with Geodesic Grids. In **Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies**, 2011.

[14] Stephen Roller, Michael Speriosu, Sarat Rallapalli, Benjamin Wing, and Jason Baldridge. Supervised Text-Based Geolocation Using Language Models on an Adaptive Grid. In **Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning**, EMNLP-CoNLL '12, p. 1500–1510, 2012.

[15] Hayate Iso, Shoko Wakamiya, and Eiji Aramaki. Density Estimation for Geolocation via Convolutional Mixture Density Network. **arXiv preprint arXiv:1705.02750**, 2017.

[16] Yasuhide Miura, Motoki Taniguchi, Tomoki Taniguchi, and Tomoko Ohkuma. A Simple Scalable Neural Networks based Model for Geolocation Prediction in Twitter. In **Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)**, pp. 235–239, 2016.

[17] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, author =.

[18] Rabindra Lamsal, Aaron Harwood, and Maria Rodriguez Read. Where id you Tweet from? Inferring the Origin Locations of Tweets Based on Contextual Information. **arXiv preprint arXiv:2211.16506**, 2022.