A Google Trends spatial clustering approach for a worldwide Twitter user geolocation

Paola Zola[a,*], Costantino Ragno[b], Paulo Cortez[c]

[a] *Institute for Informatics and Telematics (IIT) of the National Research Council of Italy (CNR), Pisa, Italy*
[b] *ANIMA Sgr S.p.a., Corso Giuseppe Garibaldi 99, 20121 Milan, Italy.*
[c] *ALGORITMI Centre, Department of Information Systems, University of Minho, 4804-533 Guimarães, Portugal*

Abstract

User location data is valuable for diverse social media analytics. In this paper, we address the non-trivial task of estimating a worldwide city-level Twitter user location considering only historical tweets. We propose a purely unsupervised approach (no location data is used) that is based on a synthetic geographic sampling of Google Trends (GT) city-level frequencies of tweet nouns and three clustering algorithms. The approach was validated empirically by using a recently collected dataset, with 3,268 worldwide city-level locations of Twitter users, obtaining competitive results when compared with a state-of-the-art Word Distribution (WD) user location estimation method. The best overall results were achieved by the GT noun (GTN) DBSCAN (GTN-DB) method, which is computationally fast, and correctly predicts the ground truth locations of 15%, 23%, 39% and 58% of the users for tolerance distances of 250 km, 500 km, 1,000 km and 2,000 km.

Keywords: City-level geolocation; Clustering; Google Trends; Natural language processing; Twitter.

* Corresponding author at: IIT-CNR, Via G. Moruzzi 1, 56124 Pisa, Italy.
Email addresses: paola.zola@iit.cnr.it (Paola Zola), costantino.ragno@unicam.it (Costantino Ragno), pcortez@dsi.uminho.pt (Paulo Cortez)

1. Introduction

Spatial data is a core element of several Web and social media analytics. In effect, knowing the country, city or even more fine-grained location of a user can help in: event detection, disaster early warnings, road traffic prediction, public welfare activity information and tourism prediction (Ozdikis et al., 2016; Zahra et al., 2020; Alkouz & Aghbari, 2020; Chen et al., 2020; Khatibi et al., 2019). For instance, Zahra et al. (2020) used geotagging and a text-based Location Indicative Words (LIW) algorithm to extract traffic locations (e.g., Downtown Dubai) from Twitter and Instagram posts. The collected time and geolocation data allowed to build an analytics system that was capable of detecting and predicting road traffic jams.

The automatic inference of social media geolocation data is a non-trivial task. Focusing on Twitter, which is a widely used microblog service, with around 326 million active users[1], the geographic data are available mainly at two different levels: single tweet Global Positioning System (GPS) latitude and longitude coordinates (e.g., $43.4722854, -80.5448576$) and user profile information (e.g., University of Waterloo, Waterloo, Ontario, Canada). However, recent studies demonstrate that only a tiny fraction (1%) of tweets are geotagged with coordinates and only 66% of the user profile data are reliable (Schulz et al., 2013). In effect, users tend to turn off GPS capabilities to save battery power or ensure privacy. Moreover, the Twitter location profile form is free text, thus users can add non-real locations such as "worldwide" or "right here". Given these limitations, diverse studies have proposed geolocation estimation methods that are purely based on tweet Word Distribution (WD) analysis. WD methods are based on a supervised machine learning or geographic dictionaries (LIW). Yet, these WD works have some disadvantages, as detailed in Section 2: the training of supervised learning WD methods requires geotagged tweets, while LIW WD methods use a finite set of locations, often associated with a particular world

[1] `https://blog.hootsuite.com/twitter-statistics/`

region.

In this work, we propose a novel unsupervised learning WD approach based on user tweet noun distributions, Google Trends (GT) statistics and spatial clustering models. GT analyzes the relative popularity of Google search queries across various world regions and languages (e.g., the search term "brexit" is more popular in the United Kingdom). It is a valuable big data source that has been mostly studied under a temporal perspective. For instance, GT was used to explain consumer behavior changes and to detect trending topics (Jun et al., 2018; Kwak et al., 2018). In previous work (Zola et al., 2019), we adopted GT for a spatial analysis, in which tweet nouns (e.g., "scotland", "brexit", "cricket") were matched with country-level GT statistics, allowing to estimate the country of interest of Twitter users. This paper extends this approach to a more informative and challenging task: a worldwide city-level estimation of the implicit user location context. To achieve such goal, we use tweet nouns, city-level GT data, a synthetic sampling of most probable GT noun world location points and spatial clustering models (one for each user). In (Paule et al., 2019), it was shown that tweet content, and not just geographic dictionary terms, is correlated with user location. Based on this knowledge, we particularly focus on tweet nouns, as recently proposed in our previous work (Zola et al., 2019), since it is assumed that users often tweet about specific sites, events, people, organizations, and so on, which are linked with their home context or place of interest. For instance, the term "brigittemacron" is strongly associated in GT with French cities, such as Paris and Lyon (Section 3.1). In contrast with most WD works, the proposed approach only uses historical tweets and freely Web data (GT, Google Maps), thus it does not require geographic labeled Twitter data or specialized LIW geographic dictionaries. Moreover, it assumes the entire world region and can be applied to any language. In addition, the obtained clustering models provide not only the most probable geographic user location but also a measure of its spatial dispersion, which can be valuable for diverse social media analytics (e.g., sentiment analysis, tracking consumer behavior).

1.1. Research objective

The main research objective of our study is to investigate the usefulness of city-level GT data, associated with historical tweet nouns, to estimate a worldwide city-level Twitter user location. As in (Zola et al., 2019), we focus on historical tweet nouns (e.g., sites, events, people), which are expected to have a spatial context that is helpful for user location estimation. In contrast with several state-of-the-art works, we address a challenging but valuable pure unsupervised learning WD setting, which assumes no geolocation target labels (geotagged tweets or user location profiles) and no access to geographic dictionaries. To address the research goal, we propose an approach that includes city-level GT scores for historical tweet nouns, a synthetic sampling of most probable GT noun world location points and spatial clustering models (one for each user).

1.2. Contributions

The main contributions of the paper are:

1. we estimate the implicit city-level geographic coordinate context of a Twitter user given her/his historical tweets, which is valuable when geotagged tweets or Twitter user location profiles are unavailable or unreliable (a common situation in practice);
2. we propose a new unsupervised clustering approach for the user location estimation, based on GT noun city distributions, city polygons, a synthetic sampling and three clustering algorithms: Gaussian Mixture Models (GMM), K-means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN); and
3. we create a recent Twitter user dataset (made publicly available[2]), with ground truth city-level locations for 3,268 worldwide users, to compare

[2] `https://github.com/paolazola/Google-Trends-for-Twitter-users-location-estimation`

the proposed approach with a WD method that uses GMM and labeled tweets (Priedhorsky et al., 2014).

The paper is structured as follows. Section 2 discusses the related work. Section 3.1 presents the data, location estimation methods and evaluation measures. Section 4 describes the experiments and obtained results. Finally, Section 5 concludes the paper.

2. Related work

The issue of geolocating social media users has been widely researched due to its importance for business analytics and other real-world applications (Zheng et al., 2018). The initial studies about Web geolocation were based on the Internet Protocol (IP) address. For example, in Backstrom et al. (2008) a probabilistic model was used to link North American Yahoo! queries with IP address locations. Yet, Virtual Private Networks (VPNs) diminish the accuracy of the IP location estimation. More importantly, social media typically does not disclose IP data, thus two main alternative approaches have been proposed: Friendship Networks (FN) or WD analysis. The FD approach uses a social network analysis, in which the user location is assumed to be close to the locations of her/his friends (Kotzias et al., 2016). This work is focused on WD geolocation, which involves a pure analysis of user texts.

Most WD methods search for location named entities (Liu & Zhou, 2013; Ngoc & Mothe, 2018), which are also known as LIW (Han et al., 2014; Lee et al., 2015; Chi et al., 2016). Another popular WD approach is to associate words to specific geographic areas by using a supervised learning (e.g., geotagged tweets) (Eisenstein et al., 2010; Ozdikis et al., 2019). However, these WD methods have limitations, assuming a finite set of locations (Han et al., 2016) or, if the estimation is fine-grained, a small fraction of the Globe surface (Roller et al., 2012; Laylavi et al., 2016). Moreover, few users use geotagging (e.g., due to privacy issues) (Schulz et al., 2013), which reduces greatly the impact of supervised learning WD methods. Some recent studies adopt hybrid approaches,

combining: WD and FN (e.g, (Rahimi et al., 2015; Bakerman et al., 2018)); WD and features derived from user account Metadata (MD) (Ozdikis et al., 2016; Dredze et al., 2013; Schulz et al., 2013); and WD, FN and MD (Williams et al., 2017).

A relevant dimension of the geolocation analysis is the level of granularity. Most research works that focus on Geographic Coordinates (GC) analyze tweets related with a specific country or area (Eisenstein et al., 2010; Paule et al., 2019). Also based on a fine-grain resolution is the work of Celik & Dokuz (2018), which targets Socially Important Locations (SIL), such as a user's home or office. To the best of our knowledge, only Pontes et al. (2012) and Schulz et al. (2013) estimated the most probable location of users given the entire world, while other WD world location studies were focused on tweet locations (and not users) (Backstrom et al., 2010; Priedhorsky et al., 2014; Williams et al., 2017; Paule et al., 2019). The Twitter user location supervised WD method proposed in Pontes et al. (2012) requires geotagged tweets, which is a limitation (as previously discussed). As for the work of Schulz et al. (2013), it considers an hybrid approach, combining WD with MD and the text time zone.

Table 1 summarizes the state-of-the-art studies on Web and social media location estimation, using a chronological order and focusing on Twitter data (**source** column). The **Type** column identifies the research approach used (WD, FN or mixed), **Lang** column details the language of the texts (e.g., English) and **Period** the data source collection period. The type of estimated geolocation is detailed in columns: **Target**, set in terms of user (U) or tweet (T) location goal; **Level**, the granularity level (e.g., Country, COORD); **LIS**, the location information source (e.g., geotagged tweets); and **Area**, the geographic targeted area. In the table, non disclosed elements are marked with the – symbol.

The last row of Table 1 compares the proposed research with the state-of-the-art works. This work extends our previous study (Zola et al., 2019), which used a simple statistical approach, based on the highest GT country tweet noun scores, to perform a worldwide country-level Twitter user location estimation. In this paper, we address the more challenging city-level location.

In particular, we use city-level GT noun scores, which are combined with city polygons to synthetically generate GT noun geographic points, used to fit spatial clustering models. In contrast with several state-of-the-art works (e.g., Celik & Dokuz, 2018; Do et al., 2018; Huang & Carley, 2019; Paule et al., 2019; Ozdikis et al., 2019), a pure unsupervised WD approach is adopted and thus no geographic labeled data (e.g., tweets or user location profiles) is required, only historical tweet nouns and GT data. Moreover, we do not use LIW, as adopted in (Ozdikis et al., 2016; Williams et al., 2017; Bakerman et al., 2018; Shahraki et al., 2019), since LIW often assumes finite and rather static set of locations, typically associated to small world regions. Instead, we use tweet nouns, which can be dynamically updated and that can refer to geographic words and also other terms with a location context (e.g., events, people or organizations).

3. Data and methods

3.1. Data

The Twitter data used in this study was collected by the authors in previous work (Zola et al., 2019). We adopted this data for several reasons: it is related with a real-world application from the alloy steel domain; it already contains a worldwide list of users; and it is more recent than other geolocation benchmarks (shown Table 1), thus the respective nouns should be more relevant for the GT queries. The data consisted of an initial sample of 49,203 users that tweeted one of the keywords {"steel price", "steel industry", "steel production"}, between March 2016 to November 2017. Since very few tweets were geotagged, a semi-automated double source verification was used in Zola et al. (2019) to set the geolocation ground truth, which was based on metadata (the user profile location field) and a Location Indicative Words (LIW) match from historical tweets. The result was a country-level geolocation dataset[3] with 744,830 tweets written by 3,298 users from 54 countries.

[3] `https://github.com/paolazola/Twitter-country-geolocation`

Table 1: Summary of the related work.

Study	Type[a]	Lang[b]	Source[c]	Target[d]	Level[e]	LIS[f]	Period	Area[g]	Method[h]	Metrics[i]
Cheng et al. (2010)	WD	EN	TW	U	CI	GTW	2009-10	USA	MLE	Acc@x mi
Eisenstein et al. (2010)	WD	EN	TW	T	GC	GTW	2010	USA	CTM	MAE,MdAE
Pontes et al. (2012)	WD,FN, MD	–	FS,GP, TW	U	CI,GC	GTW,MD	2011-12	W	Mode	Acc, Acc@x km
Roller et al. (2012)	WD	EN	WP,TW	T	GC	GTW,GWP	-	USA	SL	Acc, Acc@x km, MAE,MdAE
Dredze et al. (2013)	WD,MD	EN	TW	T	CI,ST, CO	LIW,MD, GTW	2013	USA	LIW	Acc,Acc@x km
Schulz et al. (2013)	WD,MD	-	TW	T,U	GC	LIW,MD, GN	2011-12	W	LIW	MAE,MdAE, MSE
Han et al. (2014)	WD	EN, Mixed	TW	U	CI	GTW,GN	2011-12	NA,W	SL	Acc@x km
Priedhorsky et al. (2014)	WD,MD	Mixed	TW	T	GC	GTW,MD	2012-13	W	GMM	CAE
Ryoo & Moon (2014)	WD	KR	TW	U	GC	GTW	2010-11	KR	SL	Acc@x km
Rahimi et al. (2015)	WD,FN	EN	TW	U	GC	GTW	2011-12	USA,W	SL	Acc@x km
Lee et al. (2015)	WD	EN	TW	T	ST	LIW	2013-14	USA	SL	R
Liu & Inkpen (2015)	WD	EN	TW	U	GC	-	2010, 2012	USA, NA	AE	Acc,MAE, MdAE
Chi et al. (2016)	WD	EN	TW	T,U	CI	GN, LIW	-	-	SL	Acc,MAE, MdAE
Dredze et al. (2016)	WD,MD	EN	TW	T	CI	GTW,MD	2012-15	W	–	Acc,Acc@x km, MdAE
Han et al. (2016)	WD	EN	TW	-	CI	GTW	2014	–	SL	Acc
Kotzias et al. (2016)	FN	EN	TW	U	CI	GTW	2013	IR,UK, USA	TC	P
Miura et al. (2016)	WD,MD	–	TW	T	CI	GTW,MD	-	-	SL	Acc,MdAE
Ozdikis et al. (2016)	WD,MD	TR	TW	T	CI	LIW,MD	2013	TR	DS	MAE
Williams et al. (2017)	WD,FN, MD	EN	TW	T	GC	LIW,MD	2015-16	W	TC	Acc@x km
Zubiaga et al. (2017)	WD,MD	Mixed	TW	T	CO	GTW	2014-15	W	SL	Acc,P,R, F1,MSE
Avvenuti et al. (2018)	WD	EN,IT	TW	T	GC	LD	2011-15	IT,W	SL	Acc
Bakerman et al. (2018)	WD,FN	EN	TW	T	CG	LIW,GTW	2010	USA	GMM	CAE
Celik & Dokuz (2018)	WD	TR	TW	SIL	GC	GTW	2008-15	TR	SL	
Do et al. (2018)	WD, FN	EN	TW	U	GC	–	2010-11	USA	SL	Acc,Acc@161 km
Huang & Carley (2019)	WD,MD	EN	TW	U	CO,GC	MD,LIW	-	USA,W	SL	MdAE, Acc Acc@x km
Paule et al. (2019)	WD	EN	TW	T	GC	GTW	2014-16	Chicago, New York	SL	MAE,Acc@G, Acc@x km
Ozdikis et al. (2019)	WD	Mixed	TW	T	GC	GTW	2015	London, Paris, Berlin	SL	MdAE, Acc Acc@x km
Shahraki et al. (2019)	WD,MD	EN	TW	T	CO,CI,GC	LIW,MD	-	USA	DS	Acc,MAE
Zola et al. (2019)	WD	Mixed	TW	U	CO	GT	2017-19	W	GTN	Acc,F1
This work	WD	Mixed	TW	U	GC	GT	2017-19	W	GMM,KM, DBSCAN	MAE,MdAE, Acc@x km

[a] Friendship Network (FN), Metadata (MD), Word Distribution (WD).

[b] **Language**: English (EN), Italian (IT), Korean (KR), Turkish (TR), combination of multiple languages (Mixed).

[c] Foursquare (FS), Google+ (GP), Twitter (TW), Wikipedia (WP).

[d] Socially Important Locations (SIL), Tweet location (T), User location (U).

[e] City (CI), Country (CO), Geographic Coordinates (GC), one of 50 States (ST).

[f] Geotagged News (GN), Geotagged Tweets (GTW), Geotagged Wikipedia (GWP), Location Indicative Words (LIW), Metadata (MD), Internet Protocol (IP).

[g] Ireland (IR), Italy (IT), North America (NA), Korea (KR), United Kingdom (UK), United States of America (USA), Turkey (TR), World (W).

[h] **Estimation Method**: Autoencoder (AE), Cascading Topic Model (CTM), Dempster-Shafer Theory (DS), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), Gaussian Mixture Model (GMM), Google Trends Nouns (GTN), K-means (KM), most common (Mode), Location Indicative Words (LIW), Topic Clustering (TC), Supervised Learning (SL).

[i] Accuracy (Acc), Accuracy using a radius of x (Acc@x), x in miles (mi) or kilometers (km), Comprehensive Accuracy Error (CAE), F1-score (F1), Mean Absolute Error (MAE), Median Absolute Error (MdAE), Mean Squared Error (MSE), Precision (P), Recall (R), Root Mean Square Error (RMSE).

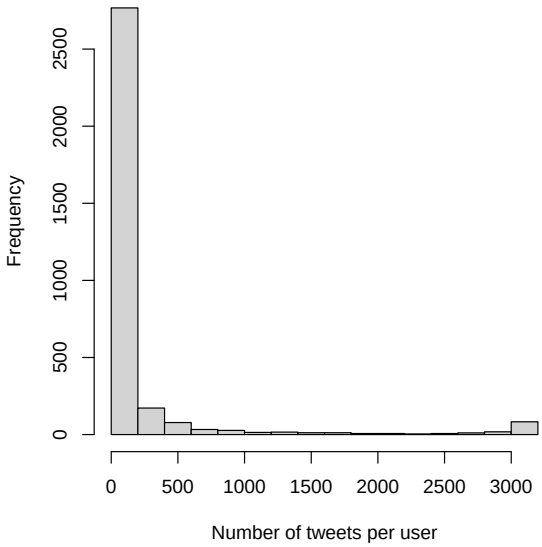


Figure 1: Histogram of the number of total tweets per user.

Given that the country-level Twitter dataset is not fine-grained, additional data processing procedures were implemented in this work, in order to achieve city-level geographic coordinates. First, we have manually checked all the 3,298 users profiles, removing all accounts that were no longer available, suspended or deleted, similarly to what was executed in Gilani et al. (2017), resulting in 3,268 users and 737,090 tweets. The Twitter API allows to retrieve a maximum of 3,200 past tweets for each user. Nevertheless, the majority of users in our dataset have a smaller tweet history (due to a younger account or smaller posting frequency), as shown in Figure 1. Only 2,134 of the tweets were geotagged (around 0.3%). Then, the selected 3,268 users were divided into three groups, according to the lowest level of granularity available at the location metadata, namely: country (**A**), state (**B**) and city (**C**). Examples of these group locations include: **A** – United States of America (USA), Italy and Australia; **B** – California, Scotland and Texas; **C** – Mumbai, New York and Sydney. Table 2 summarizes the groups in terms of number of **Users** and number of **Unique** locations. The last row of Table 2 shows the additional group **D** that represents

the set of all users (**A**, **B** and **C**). We highlight that most users (73%, from **C**) contain a city-level location. Since the experimental comparison (Section 4) uses numeric distance measures for all location granularity levels, we used the Google Maps service to set the geographic coordinates as the center of the country (for the 414 users), state (468 users) or city (2,386 users), as a reasonable proxy for the real coordinates. We note that in this work the ground truth data is only used as external data (as detailed in Section 3.2.5), thus it is not used to fit the clustering models but rather to evaluate their location estimation capabilities.

Table 2: Summary of the Twitter ground truth user datasets.

Location level	**Users** (%)	**Unique locations**
country (**A**)	414 (12.7%)	47
state (**B**)	468 (14.3%)	165
city (**C**)	2,386 (73.0%)	1,191
Total (**D**)	3,268 (100.0%)	1,403

Although we only targeted users that tweeted at least one English term from the alloy steel domain (e.g., "steel price"), the collected historical tweets include a diverse range of topics, with several of the messages being written in other languages. Table 3 presents a summary of the main detected languages in the adopted dataset of 3,268 users, when using the `langdetect` Python module (Shuyo, 2010) for language profile estimation of the 737,090 tweets. While a total of 32 distinct languages were detected, the majority of the tweets were written in English (91.2%), followed by the Croatian (1.7%), Indonesian (1.2%) and German (1.0%) languages. The percentage of extracted nouns per language exhibits a similar pattern, most of the detected nouns are in English (89.9%), followed by the Croatian (2.5%), Indonesian (1.4%) and German (1.1%) languages. Figure 2 shows the distribution of the number of detected languages per user, revealing that most users (34.3%) use just one language, followed by users that write in two (28.8%) and three (15.3%) languages. For a few users, `langdetect` detected a high number of languages (e.g., 2 users were associated

with 13 languages), thus to simply the visualization, we opted to merge all values equal or higher to seven into a single bin (≥ 7). We note that language detection with short tweets, often with acronyms and slang, is a non-trivial task and thus `langdetect` might overestimate the identification of distinct languages. However, we selected `langdetect` because it obtained better results when compared with other language detection tools. For example, the `textcat` from the R environment is computationally faster than `langdetect` but it provides a higher number of errors (e.g., "Aluminum Sheet Property" is detected as written in the classical Latin language).

Table 3: Language distribution of the collected tweets and nouns.

Language	**Tweets (%)**	**Nouns (%)**	**Language**	**Tweets (%)**	**Nouns (%)**
English	91.2	89.9	Tagalog	0.3	0.4
Croatian	1.7	2.5	Catalan	0.3	0.3
Indonesian	1.2	1.4	Somali	0.3	0.5
German	1.0	1.1	Italian	0.3	0.4
French	0.5	0.6	Norwegian	0.2	0.2
Dutch	0.5	0.3	Russian	0.2	0.2
Spanish	0.4	0.3	Estonian	0.2	0.1
Afrikaans	0.3	0.3	Others	1.4	1.5

Figure 3 plots the ground truth locations for all users, which are spread on a worldwide basis, although with different regions of density. In effect, the dataset includes a majority of users from Anglophone countries (e.g., USA, UK, Canada) or where English has a strong presence (e.g., India), which is aligned with the language analysis of **??**. Nevertheless, the dataset also includes users from non-Anglophone countries, such as Germany, Italy, China or Mexico.

3.2. Methodology

The proposed approach is shown in Figure 4 and it includes five steps that are detailed in the next subsections:

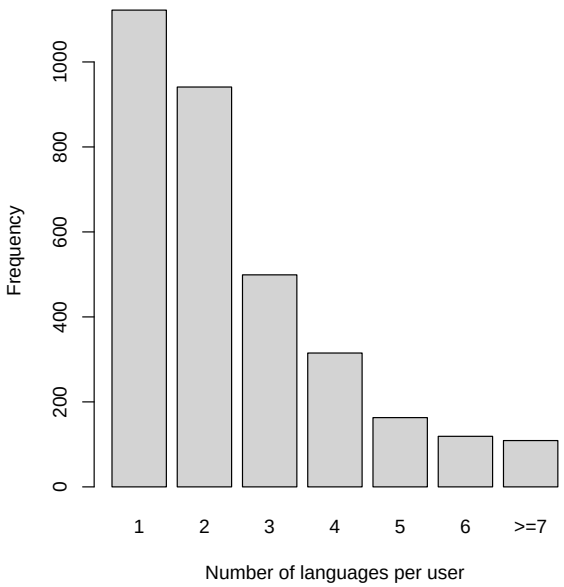


Figure 2: Histogram of the number of detected languages per user.

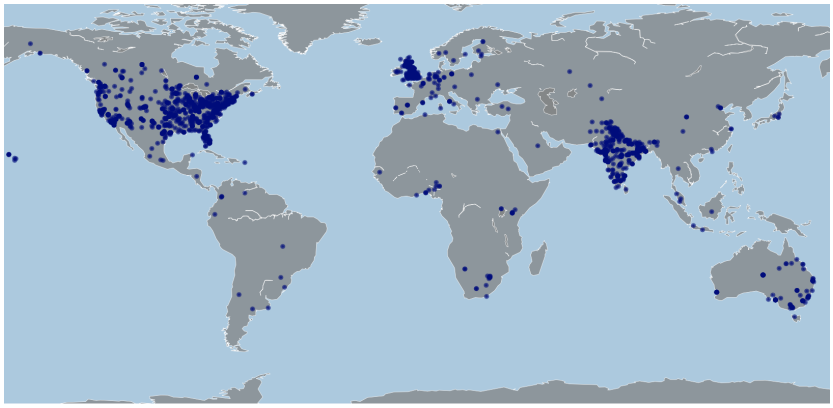


Figure 3: World distribution of the Twitter ground truth user locations (blue points).

1. User noun collection: all user historical tweets are extracted and the respective nouns are filtered (Section 3.2.1).
2. City-level noun collection: the city-level GT scores for all the nouns of a particular user are retrieved (Section 3.2.2).
3. Synthetic spatial sampling: city polygons are built and a synthetic geographic sampling of the GT noun frequencies is created (Section 3.2.3).

4. Spatial clustering and user location estimation: a clustering algorithm is fit to the user synthetic data and then the location is estimated (Section 3.2.4).
5. Performance evaluation: evaluate the quality of the estimated user location with respect to the ground truth (Section 3.2.5).

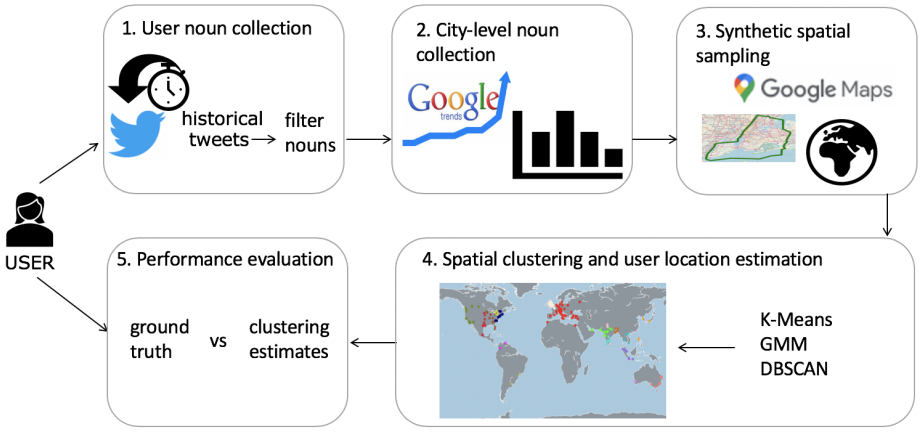


Figure 4: Schematic of the proposed approach.

3.2.1. User noun collection

Let U denote the set of all Twitter 3,268 users. For each user $u \in U$, all historical tweets are extracted (up to a maximum of 3,200 texts). The collected tweets are preprocessed by converting the text to lowercase. Following the procedure adopted in Zola et al. (2019), we process retweets as normal tweets, since retweets can be informative in terms of the user context location (e.g., retweets of a local politician or event). Then, the nouns (common and proper) were extracted using the `TextBlob` Python module, since it is faster when compared with other POS tagger tools (Loria, 2014). The result is a sequence of $\mathbf{n}_u = \langle n_1, n_2, ..., n_{l_u} \rangle$ nouns for each user $u \in U$ (e.g., $\langle$"day", "brigittemacron"$\rangle$), where l_u is the length of the sequence. For the set of 3,268 users, the total number of unique nouns obtained is 368,852.

3.2.2. City-level noun collection

Using the previously collected user nouns ($\mathbf{n}_u$, $u \in U$), the respective city-level GT data was extracted by using the `Pytrends` Python module. The result is the city noun user (CNU) distribution $\mathbf{cn}_n = \langle (c_1, g_1), (c_2, g_2), ..., (c_{l_n}, g_{l_n}) \rangle$ for noun n and that includes a list of l_n world cities (c_k), each assigned with the respective GT score (g_k). The GT scores are already normalized to the number of searches performed within that city and they range from 100 (highest impact given all Google searches) and 0 (lowest impact). Since cities without any noun queries do not appear in the GT results, we opted to round all city 0 scores into 1, thus providing a minimum relevance value.

Table 4 exemplifies two different CNU distributions ($\mathbf{cn}_n$, $n \in \{$"day", "brigittemacron"$\}$). The generic term "day" is highly used by two geographically distant countries (Australia and USA), while the more specific "brigittemacron" noun corresponds to the wife of the President of the France and thus it is mostly used in French cities (e.g., Paris, Lyon). We also note that the generic term "day" exhibits a slower initial high score decay when compared with the specific "brigittemacron" noun. For instance, the tenth score entry for "day" is $g_{10} = 82$, while the value is much lower for "brigittemacron", where $g_{10} = 50$.

Table 4: Example of GT city distributions for "day" and "brigittemacron" nouns.

"day"		"brigittemacron"	
City	**GT score**	**City**	**GT score**
Melbourne	100	Paris	100
Brisbane	95	Lyon	90
Sydney	95	Bordeaux	86
Washington	88	Nice	77
Boston	85	Strasbourg	77
Chicago	85	Clermont-Ferrand	68
New York	85	Toulouse	63
Philadelphia	85	Marseille	54
San Diego	84	Montpellier	54
Seattle	82	Dijon	50
...		...	
Vancouver	0	Warsaw	0
Winnipeg	0	Zurich	0

3.2.3. Synthetic spatial sampling

The previously obtained CNU distributions $\mathbf{cn}_n$ are now aggregated. The overall user u distribution is $\mathbf{c}_u = \langle (c1, s_1), (c2, s_2), ... \rangle$, where s_j denotes the sum of all g_k scores for all $\mathbf{n}_u$ nouns and same city c_j. Table 5 provides examples of the GT city distribution scores ($\mathbf{cn}_n$, five highest scores) for three nouns selected from an Italian user: *appennino* (Italian mountains), *lambrusco* (Italian wine), and *prodi* (politician). The last row of Table 5 (**total**) presents the overall city distribution for the Italian user ($\mathbf{c}_u$).

Table 5: GT values for Italian user example.

Noun	**GT scores**
appennino	⟨(Sora,100),(Sassuolo,42),(Reggio Emilia,27),(Modena,25),(Carpi,21),...⟩
lambrusco	⟨(Modena,100),(Parma,72),(Bologna,49),(Puebla,17),(Milan,15),...⟩
prodi	⟨(Surakarta, 100),(Depok,98),(Kediri,92),(Jaber,76),(Yogyakarta,73),...⟩
total	⟨(Depok,128),(Modena,125),(Sora,100),(Surakarta,100),(Bologna,93),...⟩

Next, the city coordinates (latitude and longitude) are extracted using the `googlegeocoder` Python module. In order to avoid degenerate geolocation data (e.g., clustering of a unique city data point for one user), we follow the suggestion in Schulz et al. (2013) and construct first a city polygon area and then we randomly sample coordinates according to the GT CNU distributions and respective polygon areas. The city polygon P_c for city c is defined by the geographic boundaries found at the `OpenStreetMap.com` (OPS) Web page, which was built by a community of mappers. In particular, the OPS *administrative boundary* were used, which are recognized by governments for administrative purposes. The OPS boundaries were manually inspected for some cities. In a few cases, the OPS boundary points did not have the correct polygon building sequence, such as shown in the left of Figure 5. To solve this issue, we applied the concave hull method (Moreira & Santos, 2007), which is based on a k-nearest neighbors algorithm. The concave hull improved all inspected problematic polygons, such as exemplified in the right of Figure 5. Also, around 10% of the cities

(often related with small remote locations) did not have OPS data. Such cases were handled by using a city circumference that considered the city centroid provided by the Google Maps and the radius $\sqrt{\frac{A}{\pi}}$, where A is the city area obtained using WikiData[4]. The developed Python library for the city polygon definition is freely available at `https://github.com/CostRagno/geopolygon`.

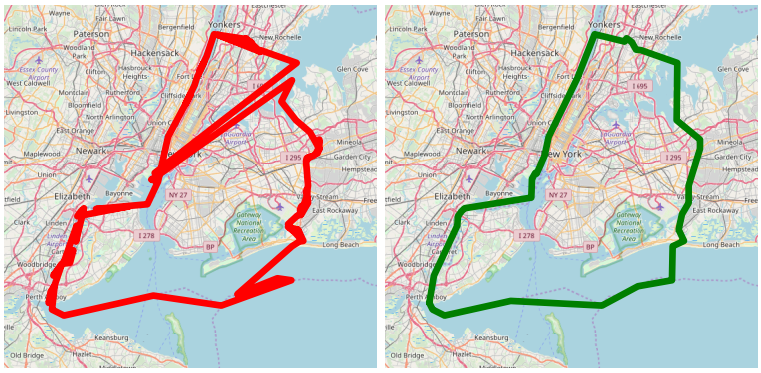


Figure 5: Example of the New York area obtained using OPS data (left, red polygon) and the concave hull approach (right, green polygon).

The two dimensional synthetic dataset (with latitude and longitude values) is used to fit a clustering model for a given user and it considers a random uniform sampling of coordinates given the respective GT noun data and city polygons. Let R denote a parameter that corresponds to the total number of random samples included in the synthetic dataset for user u. Then, the number of sampled points r_c for city c is set proportionally to the GT noun overall scores, namely $r_c = round(R \times \frac{s_c}{S_u})$, where $round$ represents the round to integer function, s_c is the GT city noun overall score and $S_u = \sum_{i=1}^{l_n} s_i$. For a particular city c, the r_c samples are created by performing a uniform coordinate sample (latitude and longitude) within the rectangular region that includes the city polygon P_c. If the coordinate is outside the polygon, the sampling procedure is repeated, until r_c valid samples are created.

The single parameter R controls the quality of the generated GT noun coor-

[4] `https://www.wikidata.org/wiki/Wikidata:Main_Page`

dinates. A high value produces a fine-grained world map but also increases the clustering computational effort. In this paper, we set R experimentally, testing different values in the range $R \in \{500, 1{,}000, 2{,}500, 5{,}000, 10{,}000, 15{,}000\}$.

3.2.4. Spatial clustering and user location estimation

The obtained synthetic spatial data is used to fit a clustering model for each user. Then, the most probable user location is set as the centroid of the largest cluster (with more synthetic data points). All clustering experiments were implemented using the `scikit-learn` Python library. We explore three popular clustering algorithms that employ distinct forms of grouping items (Aggarwal & Reddy, 2014): GMM, to represent the class of probability distribution methods; K-means, which uses a centroid model; and DBSCAN, as a representative of a density based approach. GMM is based on a linear combination of K Gaussian models, called components, that represent the clusters.

Regarding the clustering setup, the GMM internal parameters (linear weights, component means and covariances) are often estimated by using the Expectation Maximization (EM) algorithm, which was adopted in this work. As for K-means, it is a partitional iterative algorithm that starts K initial points as centroids. Then, given a proximity measure (e.g., Euclidean distance), each point is assigned to the closest centroid. Once the K clusters are defined, the centroids are updated. In this paper, the initial centroids were set using the K-means++ algorithm (Aggarwal & Reddy, 2014). Finally, DBSCAN is a density-based method that does not require setting the number of clusters (K) and that is suited for large and sparse datasets, since the clusters are based on high density point areas, where low density regions are considered noise or outliers (Ester et al., 1996). The method contains two hyperparameters: ϵ, the radius of the cluster; and *MinPts*, the minimum number of points needed to create a cluster.

Both GMM and K-means require the *a priori* definition of the number of K clusters. Since a different clustering model is built for each user u, we search for the best number of clusters K_u by using an automatic grid search within the

range $K_u \in \{1, ..., K_{\max}\}$. In this paper, the maximum number of searched clusters is set experimentally, using the set of values $K_{\max} \in \{5, 10, 25, 50, 100\}$. The automatic selection of the optimal K_u value is based on three model selection measures: the Bayesian Information Criterion (BIC) and two Gap criteria. The BIC is a known model selection criterion that is used here under two versions (Schwarz et al., 1978), one for GMM and other for K-means:

$$K_u = \{\min K : q \log(R) - 2L_K\} \qquad \text{(BIC for GMM)} \qquad (1a)$$

$$K_u = \{\min K : RSS_K + 2\log(R)K\} \qquad \text{(BIC for K-means)} \qquad (1b)$$

where q is the number of GMM parameters, L_K is the log-likelihood function for K. The $RSS_K = \sum_{k=1}^{K} \sum_{p \in C_k} |p - m_k|$ corresponds to the residual sum of squares, where C_k denotes a cluster with the centroid m_k and p is a data point. Regarding the Gap statistic, it is based on a comparison of the total intra-cluster variation with a null reference distribution of the data. In this work, we test two criteria for the K_u selection (Tibshirani et al., 2001):

$$K_u = \{\min K : Gap(K) \geq Gap(K+1) - sd_{K+1}\} \qquad (\text{Gap}_1) \qquad (2a)$$

$$K_u = \{K : \max Gap(K)\} \qquad (\text{Gap}_2) \qquad (2b)$$

where sd_K represents a standard deviation for K when using a reference Monte Carlo sample. Regarding DBSCAN, a grid search is used to set the two hyper-parameters: $\epsilon \in \{0.1, 0.25, 0.5, 0.75, 1, 1.5, 2, 5, 10, 25, 50, 100\}$ and $MinPts \in \{5, 10, 25, 50, 100\}$.

To demonstrate the proposed clustering approach, we select the Italian user and three nouns from the example related with Table 5. The overall city distribution $\mathbf{c}_u$ values (last row of Table 5) ranks Depok (Indonesian city) as the most probable user location. This occurs because *prodi* can refer to a famous Italian politician but it is also an Indonesian noun, and thus several Indonesian cities present high GT scores for this term (e.g., Surakarta, Depok). The ground truth is an Italian city (Fidenza, the blue star in Figure 6), which is highly distant from Depok (around 14,261 km). For this example, the clustering approach used the K-means algorithm with 20 clusters. The left of Figure 6 shows the

respective clusters in the world map, where each cluster is colored differently and contains a radius that is proportional to the number of its data points. The world map plot shows two large regions, around Italy (largest one) and Indonesia (second largest one). The right of Figure 6 zooms the map around the largest Italy cluster, showing that the clustering estimated location (red star) is very close (the distance is just 90 km) to the ground truth (blue star).

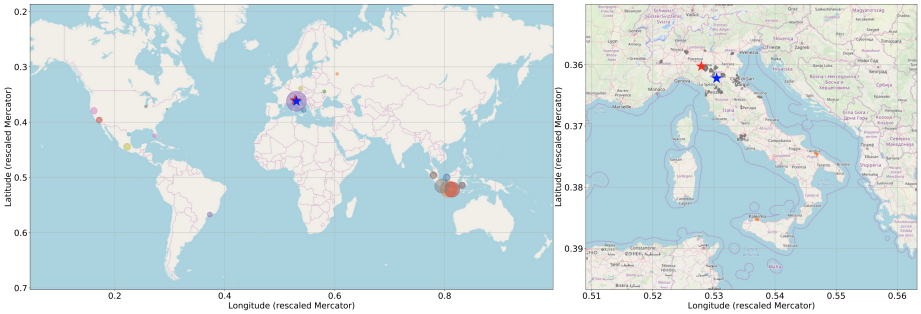


Figure 6: Example of a world clustering map (left) and its zoom around Italy (right).

3.2.5. Performance evaluation

To evaluate the unsupervised clustering approaches, we use external validation measures (Aggarwal & Reddy, 2014), where the predicted coordinates are compared with the ground truth location (not used to create the clustering models). For a given set of users, we compute the absolute errors using the Haversine distance (as adopted in Eisenstein et al., 2010; Shahraki et al., 2019) in kilometers (km), which measures the shortest surface distance between two points on a sphere, given their longitudes and latitudes. The overall distance is obtained by computing the Mean Absolute Error (MAE) and Median Absolute Error (MdAE). The non parametric Wilcoxon (W) signed-rank test (Hollander et al., 2015) (p-value <0.05) is used to confirm if the MdAE values provided by two location methods are statistically significant (paired comparison). We also compute the Regression Error Characteristic (REC) curves (Bi & Bennett, 2003), which allow a visual comparison of the results obtained by distinct location estimation methods, plotting in the y-axis the percentage of correctly

classified examples (Acc@x km) for a given tolerance radius distance (x-axis). For a maximum tolerance value ($x_{\max}$), it is possible to compute the normalized Area under the REC curve (AREC). The higher the AREC, the better are the location estimates, with the ideal method obtaining an AREC of 1.0.

3.3. Model validation

To reduce the computational effort, and similarly what was executed in Zola et al. (2019), preliminary experiments with a small random sample of 100 users (3% of all users) are first used to tune the hyperparameters of the clustering approaches. The preliminary experiment result analysis is performed using the MAE and MdAE distance measures and the clustering algorithm execution time (in minutes). Then, the best clustering approaches are evaluated and compared with the baseline methods. We note that the clustering approaches do not require any labeled training data. Thus, using the standard clustering validation practice (Aggarwal & Reddy, 2014), the user location estimation evaluation is performing using all data points (complete set of 3,268 Twitter user accounts), under an external target validation (since the location data was not used to generate the clustering models). For future comparisons, the data (e.g., ground truth, cumulative GT city noun distributions) are freely available at `https://github.com/paolazola/Google-Trends-for-Twitter-users-location-estimation`.

3.4. Baseline comparisons

The proposed methodology is compared with two baseline methods: a simpler GT noun city selection method and the approach proposed by (Priedhorsky et al., 2014). The simpler First City (FCity) method considers only the CNU distributions $\mathbf{c}_u = \langle (c1, s_1), (c2, s_2), ... \rangle$, selecting the city c_1 related with the highest overall score s_1. Then, the geographic coordinate is the city centroid given by Google Maps (Section 3.2.3). We also compare our methodology with the work of Priedhorsky et al. (2014), since there are some research similarities. As shown in Table 1, both approaches target the world map, consider a pure text-based geolocation (WD) and use the same clustering algorithm (GMM).

However, method proposed in Priedhorsky et al. (2014) differs substantially from our approach since it assumes a supervised learning, in which a GMM is fit to n-grams of Geotagged tweets (GMMG), where each word or n-gram is mapped to a location. To implement the GMMG approach, we used as training set the collection of 2,134 geotagged tweets available in our ground truth dataset. Yet, this GMMG training set represents only a tiny fraction of all data, which exemplifies in our case study the limitation of using a supervised approach based on geotagged tweets.

4. Results

4.1. Hyperparameter selection of clustering methods

As explained in Section 3.3, a small random sample of 100 users was used to set the clustering hyperparameters and select the best clustering approach. For GMM and K-means, we performed a two-dimensional grid search with distinct R and $K_{\max}$ values, using different K_u selection criteria (BIC, Gap_1 and Gap_2). As for DBSCAN, to the reduce the number of computational experiments, we first used the default $\epsilon = 0.5$ and $MinPts = 5$ `scikit-learn` implementation values in order to set R. Then, a two-dimension grid search was executed to select the final ϵ and $MinPts$ values.

The grid search results are presented in Table 6 (for GMM and K-means) and Tables 7 and 8 (for DBSCAN). As expected, the computational effort (Time) tends to enlarge when the number of sampled points (R) or maximum number of searched clusters ($K_{\max}$) increases. Overall, DBSCAN is computationally much faster than the other two clustering approaches methods, which require the extra K_u search. As for the distance errors, the best Table 6 MAE and MdAE results are: GMM – $R = 500$, $K_{\max}$ =25 and usage of BIC; K-means – $R = 1,000$, $K_{\max}$ =5 and usage of Gap_1 statistic. Turning to DBSCAN, the best first grid search sets $R = 15,000$ (Table 7) and then the second grid fine tunes the other hyperparameters into $\epsilon = 0.25$ and $MinPts$=5 (Table 8). When comparing the best performing clustering strategies, the lowest MAE and

MdAE values are provided by DBSCAN, followed by K-means. For example, the best median distance is 1,344.4 km for DBSCAN, which is 134.2 km and 475.2 km lower when compared with the best K-means and GMM values.

Table 6: Grid search results for GMM and K-means (best distance values in **bold**).

R	$K_{\max}$	GMM BIC			GMM Gap_1			GMM Gap_2			K-means *BIC*			K-means Gap_1			K-means Gap_2		
		MAE	MdAE	Time*	MAE	MdAE	Time*	MAE	MdAE	Time*	MAE	MdAE	Time*	MAE	MdAE	Time*	MAE	MdAE	Time*
500	5	5295.6	3178.3	14.9	5301.5	3338.1	11.1	6970.3	6305.9	2.0	5069.1	2400.9	12.8	4803.1	1612.0	12.8	8766.5	8769.8	34.3
	10	5725.1	3605.4	36.2	5283.7	2129.9	35.1	7235.0	6507.2	15.0	5075.3	2400.9	34.6	4720.5	2154.4	34.6	5920.3	5189.0	82.6
	25	4653.6	1819.6	192.3	5710.1	4250.8	181.3	7059.1	6226.5	8.9	5039.0	2400.9	105.8	5051.0	2545.5	105.8	4876.5	2415.5	232.7
	50	4938.3	2365.9	852.4	5324.0	3751.7	540.7	7182.9	6416.1	25.5	5074.5	2400.9	385.1	5341.5	2553.0	385.1	4746.7	1951.2	676.4
1,000	5	5518.8	3879.9	17.7	5125.0	2952.0	25.4	6980.3	6226.7	14.0	4917.9	2473.9	48.5	**3920.6**	**1478.6**	48.5	8799.5	8769.8	32.0
	10	5228.2	3432.4	41.9	4971.3	2726.3	51.4	7164.6	6226.7	31.7	4796.8	2349.9	355.5	4968.1	1815.6	355.5	5039.5	1946.8	75.8
	25	5468.7	3514.7	193.1	5196.3	2954.2	123.9	7157.2	6363.3	51.7	4620.5	2349.9	549.0	5552.8	3709.8	549.0	4797.9	1883.2	275.5
	50	5519.0	3787.9	713.0	5026.9	2360.6	632.6	7087.4	6226.7	79.4	4852.8	2473.9	594.7	4448.5	2534.3	594.7	4932.4	1938.2	563.6
2,500	5	4963.6	2780.5	39.1	5449.0	3680.3	39.6	7018.5	6471.3	47.0	4766.7	2528.0	81.6	4327.6	1506.8	81.6	8400.4	8758.5	44.8
	10	5306.2	3268.1	79.2	5006.2	2780.5	78.5	7172.6	6540.0	74.5	4894.4	2599.3	212.3	4589.8	1485.2	212.3	5257.1	2372.1	138.9
	25	5033.9	2511.1	212.0	4841.5	3171.3	817.6	7053.9	6389.2	526.1	4784.1	2570.1	447.5	5556.5	3671.8	447.5	4843.7	2031.9	432.1
	50	5174.2	2519.6	3698.6	5838.8	4015.5	1250.9	7341.0	6647.2	799.1	4882.5	2622.1	3070.4	5162.6	3287.8	3070.4	5084.4	2098.5	8578.9
5,000	5	6148.1	5021.9	121.0	5538.9	3671.3	117.6	7051.6	6215.5	170.4	5127.6	3029.3	183.6	4507.1	1532.6	183.6	8690.5	8761.2	91.5
	10	4415.6	1939.6	167.5	5287.1	2964.5	133.2	6846.0	6107.2	232.3	5121.4	3029.3	305.5	4803.8	1866.2	305.5	5477.1	3180.9	234.7
	25	4552.1	2088.9	1091.4	5179.5	3114.6	411.0	7256.8	6302.6	1626.1	5133.4	3029.3	2577.6	5331.6	3351.8	2577.6	5178.0	2720.2	1245.3
	50	4938.3	2365.9	852.4	4933.3	2759.8	2540.9	5626.8	3782.3	4915.9	5120.3	3029.3	13062.8	4520.6	1883.5	13062.8	5320.4	2915.4	4079.8
10,000	5	5356.0	3529.4	553.3	5614.3	4294.8	426.1	7187.9	6351.2	848.4	5312.0	3283.8	941.9	4169.0	1539.6	941.9	8633.8	8761.2	380.8
	10	5148.3	2509.0	1306.0	5251.2	3357.5	719.9	7230.6	6418.5	1978.0	5410.6	3457.2	1093.9	4918.1	2106.9	1093.9	5235.8	2733.0	835.5
	25	4709.6	2028.3	2104.0	5191.6	2536.2	1988.1	7212.3	6351.2	6291.6	5410.6	3457.2	4473.5	5410.6	3457.2	4473.5	4994.4	2076.1	4572.1
	50	5939.6	3842.8	7743.0	5428.6	3348.6	6489.2	7126.4	6353.3	8136.0	5411.8	3457.2	16471.9	5079.3	2568.9	16471.9	7013.6	6176.8	9002.8
15,000	5	5548.6	4174.5	883.3	5084.7	2563.8	1721.4	7217.6	6369.5	2396.2	5304.2	3272.7	909.8	4168.9	1522.0	909.8	8825.5	8790.8	738.6
	10	5306.2	3474.7	2237.7	5050.6	3448.7	2769.2	7075.2	6181.9	6582.9	5224.7	3272.7	1406.6	4757.9	1625.1	1406.6	5251.6	2911.2	1033.5
	25	4806.0	2397.0	3868.0	4876.0	2228.3	2956.4	7065.8	6316.6	15731.9	5206.1	3064.6	5451.0	5781.0	3800.9	5451.0	5212.3	2032.0	4981.1
	50	5219.2	3263.9	6135.1	5397.2	2482.4	1675.6	7315.7	6427.7	13268.1	5226.0	3272.7	24415.0	4563.0	2470.2	24415.0	5001.1	1888.4	13139.7

* Time is expressed in minutes.

Table 7: Grid search results for DBSCAN (Time is expressed in minutes, best distance values in **bold**).

R	MAE	MdAE	Time
500	7615.9	6719.0	0.0
1,000	6961.3	6343.8	0.1
2,500	5233.1	4355.5	0.2
5,000	5265.7	2822.2	0.2
10,000	4790.0	1860.8	0.3
15,000	**4119.0**	**1767.8**	0.4

Table 8: Grid search results for DBSCAN when $R = 15,000$ (best distance values in **bold**).

MinPts	Metric	ϵ											
		0.1	0.25	0.5	0.75	1	1.5	2	5	10	25	50	100
5	MAE	4002.8	**3576.0**	4119.0	3944.3	4816.5	5114.1	5254.2	4574.5	4833.5	4603.4	6001.8	7245.6
	MdAE	1645.0	**1344.4**	1767.8	1686.3	2203.4	2560.7	3288.6	1503.1	1760.1	2056.8	5482.9	6369.5
	Time	0.3	0.3	0.4	0.3	0.4	0.4	0.4	0.7	1.6	2.9	3.4	5.1
10	MAE	4926.7	4012.5	4515.1	4196.6	4816.5	5114.1	5252.7	4581.4	4833.5	4603.5	6001.8	7245.6
	MdAE	3909.8	1767.1	1904.4	1884.3	2203.4	2560.7	3288.6	1503.1	1760.1	2056.8	5482.9	6369.5
	Time	0.4	0.4	0.4	0.4	0.4	0.4	0.5	0.7	1.2	2.3	3.4	5.6
25	MAE	7509.5	5312.9	5230.7	5031.2	5582.0	5037.7	5250.5	4575.0	4833.5	4716.7	6001.8	7245.6
	MdAE	6335.2	5312.8	5078.5	4988.1	5248.9	2560.7	3288.6	1455.9	1760.1	2126.4	5482.9	6369.5
	Time	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.7	1.1	2.1	3.1	4.9
50	MAE	8052.1	6997.9	6805.7	6782.2	6639.4	6265.4	5094.9	4442.6	4826.7	4716.4	6001.8	7245.6
	MdAE	6535.1	6201.2	6367.2	6566.4	6544.1	6225.9	3596.6	1455.9	1620.4	2126.4	5482.9	6369.5
	Time	0.3	0.4	0.4	0.4	0.4	0.5	0.5	0.8	1.3	2.4	3.4	5.5
100	MAE	8054.4	7498.9	7276.9	7334.8	7520.3	7294.1	6900.2	4444.1	4819.3	4700.3	5870.9	7245.6
	MdAE	6933.5	6744.2	6631.1	6670.6	6996.2	6896.9	6543.2	1487.1	1620.4	2156.8	5240.4	6369.5
	Time	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.7	1.2	2.2	3.4	5.6

4.2. Geolocation estimation results

In this section, we compare the proposed GT Noun (GTN) clustering approach with the baseline methods: FCity and GMMG (Section 3.4). Following the results of Section 4.1, we selected the two best clustering algorithm setups (DBSCAN and K-means), namely:

- GTN-DB: DBSCAN with $R = 15,000$, $\epsilon = 0.25$ and $MinPts = 5$;
- GTN-KM: K-means using Gap_1 criterion, $R = 1,000$ and $K_{\max} = 5$.

Table 9 summarizes the obtained results, in terms of MAE, MdAE and Wilcoxon test for MdAE significance (column W), for the four geolocation methods and Table 2 user datasets. For all datasets, the proposed clustering methods provide better MAE and MdAE values when compared with the baseline methods. In particular, most of the MdAE comparisons with baseline methods are statistically significant. We note that GMMG provided the worst geolocation

performance, which is a natural result since it is a supervised learning method and very few tweets are geotagged. The other baseline, FCity, is ranked at third place, which confirms the usefulness of the proposed clustering approach. Regarding the comparison of the two clustering approaches, Table 9 does not show a clear winner. An important result is that GTN-DB obtained a significantly better MdAE value for dataset **C** (City), which contains the coordinates with the lowest level of granularity and that corresponds to the majority of the users (73%). Overall (dataset **D**), GTN-KM provides the best MdAE results (but without a significant difference) and GTN-DB obtains the best MAE.

Table 9: Geolocation distance results (best values in **bold**).

	Country (**A**)			State (**B**)			City (**C**)			Total (**D**)		
Method	MAE	MdAE	W*	MAE	MdAE	W*	MAE	MdAE	W*	MAE	MdAE	W*
GTN-DB	**4284.3**	1904.7	^{k,f,g}	3559.9	1545.4	^{f,g}	**4267.0**	**1365.2**	^{k,f,g}	**4167.9**	1548.3	^{f,g}
GTN-KM	4760.2	**1270.4**	^{d,f,g}	**3141.4**	**1451.6**	^{f,g}	4550.6	1433.0	^{d,f,g}	4375.3	**1421.5**	^{f,g}
FCity	5387.3	2356.3	^{g}	4720.0	2567.0	^{g}	5453.5	2834.5	^{g}	5340.0	2607.7	^{g}
GMMG	7773.8	7579.9		5237.0	4776.3		6949.1	6685.7		6807.8	6582.2	

* – statistically significant under a pairwise comparison with: GTN-DB (d), GTN-KM (k), FCity (f) or GMMG (g).

The clustering models provide a spatial dispersion model that is more informative than just the final estimated location. In this work, we indirectly measure this informative value by adopting a cluster tolerance analysis, which considers the minimum distance between the ground truth and any of the L centroids of the largest K_L clusters. Table 10 presents the respective results when K_L is increased from 1 (no cluster tolerance) to 3 (tolerance of 2 clusters). The obtained results show an interesting reduction in terms of MAE and MdAE distance values for both clustering approaches. For example, the GTN-DB MdAE value is reduced by 594.8 km (decrease of 38%) when the best of two ($K_L = 2$) centroid estimates is considered, a value that further diminishes by 207.0 km when a cluster tolerance of 2 ($K_L = 3$) is admitted (decrease of 52% when compared with K_L =1). Thus, the distinct centroid locations (in this

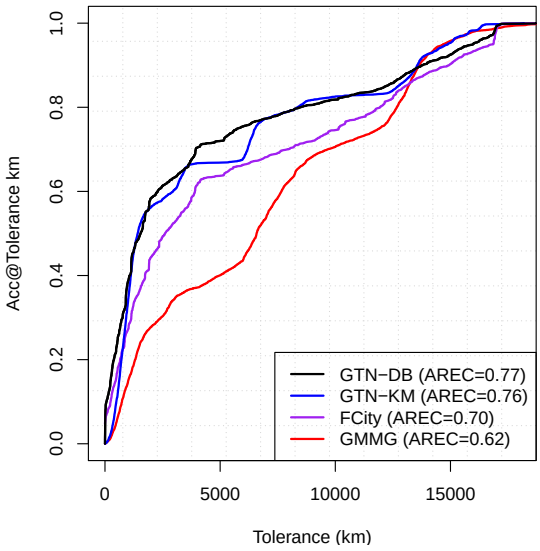
case, related with the second and third largest clusters) contain valuable spatial locations that can be used when several probable locations are needed or when the first location prediction fails.

Table 10: Geolocation distance results for the cluster tolerance analysis.

		K_L		
Method	**Metric**	1	2	3
GTN-DB	MAE	4167.9	2687.1	1952.4
	MdAE	1548.3	953.5	746.5
GTN-KM	MAE	4375.3	2460.3	1592.6
	MdAE	1421.5	1139.0	941.4

When considering the single prediction model, where the location is set as the centroid of the largest cluster ($K_L = 1$), an additional method comparison is provided by the REC analysis, which shows clear differences among the methods. The REC curves are plotted in the left of Figure 7, where the maximum tolerance was set to $x_{\max}$ =18,000 km (the value that sets ACC@x equal to 100%). The best overall curve is provided by GTN-DB, corresponding to the highest normalized area (AREC of 0.77), which is slightly better than GTN-KM (1 percentage point) and substantially higher when compared with FCity (7 percentage points) and GMMG (15 percentage points). We particularly highlight that GTN-DB consistently outperforms other methods for smaller tolerance distances (shown in the right of Figure 7), which are more useful in practice. For example, when a very small tolerance range is set (250 km), GTN-DB correctly classifies 15% of the users, while FCity accurately classifies 10% of the locations and GTN-KM only predicts well 2% of the examples. In effect, GTN-DB provides an interesting range of predictive ACC@x values: 15% for x =250 km, 23% for x =500 km, 39% for x =1,000 km, 58% for x =2,000 km and 70% for x =4,000 km. For demonstration purposes, Figure 8 shows examples of GTN-DB quality predictions, within a 1,000 km tolerance distance (1,268 blue points, 39% of the users), and lower-class estimations, with location distances higher

than 10,000 km (595 red points, 18% of the users). The quality predictions cover regions that are predominant in the dataset, such as USA, India, Europe and Australia. As for the high error estimates, several are related with anglophone mismatches between USA, Australia and UK (e.g., 142 users assigned by GTN-DB to Australia are from USA).



	Tolerance (x-axis)				
ACC@x	250	500	1,000	2,000	4,000
GTN-DB	15%	23%	39%	58%	70%
GTN-KM	2%	10%	34%	56%	67%
FCity	10%	16%	27%	44%	61%
GMMG	1%	5%	15%	28%	37%

Figure 7: REC curves for the geolocation estimation methods (left plot, $x_{\max}$ =18,000) and examples of some ACC@x km values (right table).

5. Conclusions and discussion

Spatial data is a key element for several social media analytics systems. In this paper, we address the challenging task of inferring the most probable implicit city-level context location of worldwide Twitter users based only on a Word Distribution (WD) analysis of historical tweets. To achieve this, we propose a novel unsupervised approach that includes several steps. For each user, it first filters the tweet nouns, retrieving the respective city-level Google Trends (GT) scores. Then, it creates a synthetic spatial dataset based on city polygons and a sampling of the most probable overall GT Noun (GTN) world

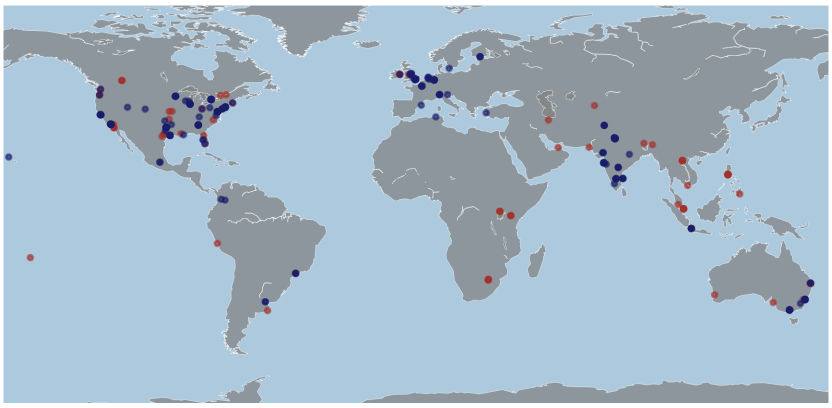


Figure 8: World distribution of the GTN-DB predictions with a ground truth location distance lower than 1,000 km (blue points) and higher than 10,000 km (red points).

location points. Finally, the spatial dataset is used to fit a clustering model and the user location is estimated as the centroid of the largest data point cluster.

To validate the proposed approach, we analyzed a recently collected Twitter dataset with 3,268 ground truth worldwide city-level user locations. First, preliminary experiments using a small random sample of 100 users were used to tune three clustering algorithms, namely Gaussian Mixture Models (GMM), K-means and Density-Based Spatial Clustering of applications with Noise (DBSCAN). Then, the best clustering approaches, GTN DBSCAN (GTN-DB) and GTN K-means (GTN-KM), were further analyzed by considering all 3,268 users and several location performance measures using haversine distances, including Mean Absolute Error (MAE), Median Absolute Error (MdAE), normalized area under the Regression Error Characteristic (AREC) curve and accuracy for a tolerance distance of x (Acc@x). The clustering results were compared with two baselines, the highest GT score city (FCity) and a GMM fit to n-grams of Geotagged tweets (GMMG) (Priedhorsky et al., 2014). Both clustering approaches (GTN-DB and GTN-KM) outperformed the baseline methods when considering the MAE, MdAE and AREC location measures.

5.1. Research implications

In terms of theoretical implications, an innovative aspect of our research is that we study the impact of GT for a social media spatial location rather than the more common temporal evolution analysis (e.g., Jun et al., 2018; Kwak et al., 2018). In particular, this paper presents the first WD attempt to use freely available city-level GT data to infer the worldwide location of Twitter users. When compared with state-of-the-art WD works, the proposed clustering approach presents several advantages. First, it does not require any Twitter geographic labeled data (e.g., geotagged tweets or user metadata), which are needed by supervised learning WD approaches (e.g., Celik & Dokuz, 2018; Ozdikis et al., 2019). Second, it does not use a static geographic dictionary, such as adopted by Location Indicative Words (LIW) methods (e.g., Ozdikis et al., 2016; Shahraki et al., 2019), which is often designed for a specific language and world region. In contrast, the GTN clustering approach automatically assigns nouns from any written language to world regions. Thus, the extracted nouns are more flexible (reflecting not only location sites but also events, people or organizations) and they can be dynamically updated (as they arise in recent tweets). Moreover, while this work specifically addresses only a single user context location (e.g., home, place of interest), the obtained clustering models provide a spatial dispersion model, which can be used in additional analyses (e.g., study of consuming behaviours or traveling patterns).

As for practical implications, we recommend the GTN-DB approach, which requires much less computation when compared with GTN-KM (it is around 160 times faster, as shown in Section 4.1). Also, our experimental results favor GTN-DB for several relevant geolocation analyses, such as: lowest MAE and MdAE distances for city-level users (dataset **C** from Table 2, which is the largest with 2,386 users); lowest MdAE values for a cluster tolerance of 1 and 2; and highest accuracy values for several low tolerance distances (ACC@x). We highlight the last results, since GTN-DB obtained interesting user location accuracy values of 15%, 23%, 39% and 58% for tolerance distances of 250 km, 500 km, 1,000 km and 2,000 km. Thus, when no geotagged tweets or reliable user location metadata

are available (which often occurs in practice), GTN-DB can be used as a valuable tool to infer a spatial context that can be used to support social media spatial analytics (e.g., sentiment analysis, monitoring consumer behavior). Another practical contribution is the creation of a recent Twitter location user dataset, related with the alloy steel domain, which is made publicly available, allowing other researchers to compare different approaches against GTN-DB.

5.2. Limitations and future work

While interesting results were achieved by GTN-DB, the depth and accuracy of the research needs to be improved. For instance, we analyzed the worldwide locations of 3,298 users related with only one case study (alloy steel prices). Also, we considered all historical nouns (maximum of 3,200 tweets per user). A richer analysis could be provided by considering additional user locations from other application domains and different historical time periods (e.g., last 3 months). Moreover, some high error estimates were obtained by GTN-DB (e.g., 18% of the users present GTN-DB location distances higher than 10,000 km). The proposed approach uses all nouns. While some nouns are location specific (e.g., "brigittemacron"), others are more universal (e.g., "day") and thus have less informative value, potentially prejudicing the GTN-DB location performance. To address these limitations, in future work we intend to consider more users from additional case studies, such as related with commodity prices (e.g., gold, coffee) or road traffic events. Furthermore, we wish to extend the proposed approach by studying the temporal effect of tweet nouns and their city-level GT frequencies (e.g., comparing one year of historical data versus one month) and by targeting smaller world regions (e.g., considering just the USA country, in order to estimate the user state). Also, we plan to explore hybrid approaches, in which GTN-DB is combined with other location methods, such as LIW or friendship networks (use of social network analysis). Another interesting research direction is the application of a semi-supervised learning approach, in which a small sample of geotagged tweets or users could be used to enhance the GTN-DB location performance. For instance, by applying statistical measures,

such as Term Frequency-Inverse Document Frequency (TF-IDF) or Information Gain (IG) (Oliveira et al., 2016), on a shorter set of labeled data to filter the more generic and less relevant nouns.

Acknowledgements

The work of P. Cortez was supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020. We would also like to thank the anonymous reviewers for their helpful suggestions.

References

Aggarwal, C. C., & Reddy, C. K. (Eds.) (2014). *Data Clustering: Algorithms and Applications*. CRC Press. URL: `http://www.crcpress.com/product/isbn/9781466558212`.

Alkouz, B., & Aghbari, Z. A. (2020). SNSJam: Road traffic analysis and prediction by fusing data from multiple social networks. *Information Processing & Management*, 57(1). doi:`10.1016/j.ipm.2019.102139`.

Avvenuti, M., Cresci, S., Nizzoli, L., & Tesconi, M. (2018). GSP (geo-semantic-parsing): Geoparsing and geotagging with machine learning on top of linked data. In A. Gangemi, R. Navigli, M. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, & M. Alam (Eds.), *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings* (pp. 17–32). Springer volume 10843 of *Lecture Notes in Computer Science*. doi:`10.1007/978-3-319-93417-4\_2`.

Backstrom, L., Kleinberg, J. M., Kumar, R., & Novak, J. (2008). Spatial variation in search engine queries. In J. Huai, R. Chen, H. Hon, Y. Liu, W. Ma, A. Tomkins, & X. Zhang (Eds.), *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008* (pp. 357–366). ACM. doi:`10.1145/1367497.1367546`.

Backstrom, L., Sun, E., & Marlow, C. (2010). Find me if you can: improving geographical prediction with social and spatial proximity. In M. Rappa, P. Jones, J. Freire, & S. Chakrabarti (Eds.), *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010* (pp. 61–70). ACM. doi:10.1145/1772690.1772698.

Bakerman, J., Pazdernik, K., Wilson, A. G., Fairchild, G., & Bahran, R. (2018). Twitter geolocation: A hybrid approach. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(3), 34:1–34:17. doi:10.1145/3178112.

Bi, J., & Bennett, K. P. (2003). Regression error characteristic curves. In T. Fawcett, & N. Mishra (Eds.), *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA* (pp. 43–50). AAAI Press. URL: http://www.aaai.org/Library/ICML/2003/icml03-009.php.

Celik, M., & Dokuz, A. S. (2018). Discovering socially similar users in social media datasets based on their socially important locations. *Information Processing & Management*, 54(6), 1154–1168. doi:10.1016/j.ipm.2018.08.004.

Chen, L., Lyu, D., Xu, Z., Long, H., & Chen, G. (2020). A content-location-aware public welfare activity information push system based on microblog. *Information Processing & Management*, 57(1). doi:10.1016/j.ipm.2019.102137.

Cheng, Z., Caverlee, J., & Lee, K. (2010). You are where you tweet: a content-based approach to geo-locating twitter users. In J. Huang, N. Koudas, G. J. F. Jones, X. Wu, K. Collins-Thompson, & A. An (Eds.), *Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010* (pp. 759–768). ACM. doi:10.1145/1871437.1871535.

Chi, L., Lim, K. H., Alam, N., & Butler, C. J. (2016). Geolocation prediction in twitter using location indicative words and textual features. In B. Han,

A. Ritter, L. Derczynski, W. Xu, & T. Baldwin (Eds.), *Proceedings of the 2nd Workshop on Noisy User-generated Text, NUT@COLING 2016, Osaka, Japan, December 11, 2016* (pp. 227–234). The COLING 2016 Organizing Committee. URL: https://www.aclweb.org/anthology/W16-3930/.

Do, T. H., Nguyen, D. M., Tsiligianni, E., Cornelis, B., & Deligiannis, N. (2018). Twitter user geolocation using deep multiview learning. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018* (pp. 6304–6308). IEEE. doi:10.1109/ICASSP.2018.8462191.

Dredze, M., Osborne, M., & Kambadur, P. (2016). Geolocation for twitter: Timing matters. In K. Knight, A. Nenkova, & O. Rambow (Eds.), *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016* (pp. 1064–1069). The Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/N16-1122/.

Dredze, M., Paul, M. J., Bergsma, S., & Tran, H. (2013). Carmen: A Twitter Geolocation System with Applications to Public Health. In *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence* (pp. 20–24).

Eisenstein, J., O'Connor, B., Smith, N. A., & Xing, E. P. (2010). A latent variable model for geographic lexical variation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP 2010, 9-11 October 2010, MIT Stata Center, Massachusetts, USA, A meeting of SIGDAT, a Special Interest Group of the ACL* (pp. 1277–1287). ACL. URL: https://www.aclweb.org/anthology/D10-1124/.

Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, & U. M. Fayyad (Eds.), *Proceedings of the Second*

International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA (pp. 226–231). AAAI Press. URL: http://www.aaai.org/Library/KDD/1996/kdd96-037.php.

Gilani, Z., Kochmar, E., & Crowcroft, J. (2017). Classification of twitter accounts into automated agents and human users. In J. Diesner, E. Ferrari, & G. Xu (Eds.), *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, Sydney, Australia, July 31 - August 03, 2017* (pp. 489–496). ACM. doi:10.1145/3110025.3110091.

Han, B., Cook, P., & Baldwin, T. (2014). Text-based twitter user geolocation prediction. *Journal of Artificial Intelligence Research*, 49, 451–500. doi:10.1613/jair.4200.

Han, B., Jimeno-Yepes, A., MacKinlay, A., & Chi, L. (2016). Temporal modelling of geospatial words in twitter. In T. Cohn (Ed.), *Proceedings of the Australasian Language Technology Association Workshop 2016, Melbourne, Australia, December 5 - 7, 2016* (pp. 133–137). ACL. URL: https://www.aclweb.org/anthology/U16-1015/.

Hollander, M., Wolfe, D. A., & Chicken, E. (2015). *Nonparametric Statistical Methods*. (3rd ed.). John Wiley & Sons.

Huang, B., & Carley, K. M. (2019). A Hierarchical Location Prediction Neural Network for Twitter User Geolocation. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019* (pp. 4731–4741). Association for Computational Linguistics. doi:10.18653/v1/D19-1480.

Jun, S.-P., Yoo, H. S., & Choi, S. (2018). Ten years of research change using Google Trends: From the perspective of big data utilizations and applications. *Technological Forecasting and Social Change*, 130, 69–87.

Khatibi, A., Belém, F., da Silva, A. P. C., Almeida, J. M., & Gonçalves, M. A. (2019). Fine-grained tourism prediction: Impact of social and environmental features. *Information Processing & Management*, 57(2). doi:10.1016/j.ipm.2019.102057.

Kotzias, D., Lappas, T., & Gunopulos, D. (2016). Home is where your friends are: Utilizing the social graph to locate twitter users in a city. *Information Systems*, 57, 77–87. doi:10.1016/j.is.2015.10.011.

Kwak, H., An, J., Salminen, J., Jung, S., & Jansen, B. J. (2018). What We Read, What We Search: Media Attention and Public Attention Among 193 Countries. In P. Champin, F. L. Gandon, M. Lalmas, & P. G. Ipeirotis (Eds.), *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018* (pp. 893–902). ACM. doi:10.1145/3178876.3186137.

Laylavi, F., Rajabifard, A., & Kalantari, M. (2016). A multi-element approach to location inference of twitter: A case for emergency response. *ISPRS International Journal of Geo-Information*, 5(5), 56. doi:10.3390/ijgi5050056.

Lee, S., Farag, M. M., Kanan, T., & Fox, E. A. (2015). Read between the lines: A machine learning approach for disambiguating the geo-location of tweets. In P. L. B. II, S. Allard, H. Mercer, M. Beck, S. J. Cunningham, D. H. Goh, & G. Henry (Eds.), *Proceedings of the 15th ACM/IEEE-CE Joint Conference on Digital Libraries, Knoxville, TN, USA, June 21-25, 2015* (pp. 273–274). ACM. doi:10.1145/2756406.2756971.

Liu, J., & Inkpen, D. (2015). Estimating user location in social media with stacked denoising auto-encoders. In P. Blunsom, S. B. Cohen, P. S. Dhillon, & P. Liang (Eds.), *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015, June 5, 2015, Denver, Colorado, USA* (pp. 201–210). The Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/W15-1527/.

Liu, X., & Zhou, M. (2013). Two-stage NER for tweets with clustering. *Information Processing & Management*, 49(1), 264–273. doi:10.1016/j.ipm.2012.05.006.

Loria, S. (2014). Textblob: Simplified Text Processing. URL: http://textblob.readthedocs.org/en/dev/.

Miura, Y., Taniguchi, M., Taniguchi, T., & Ohkuma, T. (2016). A simple scalable neural networks based model for geolocation prediction in twitter. In B. Han, A. Ritter, L. Derczynski, W. Xu, & T. Baldwin (Eds.), *Proceedings of the 2nd Workshop on Noisy User-generated Text, NUT@COLING 2016, Osaka, Japan, December 11, 2016* (pp. 235–239). The COLING 2016 Organizing Committee. URL: https://www.aclweb.org/anthology/W16-3931/.

Moreira, A. J. C., & Santos, M. Y. (2007). Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. In J. Braz, P. Vázquez, & J. M. Pereira (Eds.), *GRAPP 2007, Proceedings of the Second International Conference on Computer Graphics Theory and Applications, Barcelona, Spain, March 8-11, 2007, Volume GM/R* (pp. 61–68). INSTICC - Institute for Systems and Technologies of Information, Control and Communication.

Ngoc, H. T. B., & Mothe, J. (2018). Location extraction from tweets. *Information Processing & Management*, 54(2), 129–144. doi:10.1016/j.ipm.2017.11.001.

Oliveira, N., Cortez, P., & Areal, N. (2016). Stock market sentiment lexicon acquisition using microblogging data and statistical measures. *Decision Support Systems*, 85, 62–73. URL: https://doi.org/10.1016/j.dss.2016.02.013. doi:10.1016/j.dss.2016.02.013.

Ozdikis, O., Oguztüzün, H., & Karagoz, P. (2016). Evidential estimation of event locations in microblogs using the dempster-shafer theory. *Information Processing & Management*, 52(6), 1227–1246. doi:10.1016/j.ipm.2016.06.001.

Ozdikis, O., Ramampiaro, H., & Nørvåg, K. (2019). Locality-adapted kernel densities of term co-occurrences for location prediction of tweets. *Information Processing & Management*, 56(4), 1280–1299. doi:10.1016/j.ipm.2019.02.013.

Paule, J. D. G., Sun, Y., & Moshfeghi, Y. (2019). On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing & Management*, 56(3), 1119–1132. doi:10.1016/j.ipm.2018.03.011.

Pontes, T., Magno, G., Vasconcelos, M. A., Gupta, A., Almeida, J. M., Kumaraguru, P., & Almeida, V. A. F. (2012). Beware of what you share: Inferring home location in social networks. In J. Vreeken, C. Ling, M. J. Zaki, A. Siebes, J. X. Yu, B. Goethals, G. I. Webb, & X. Wu (Eds.), *12th IEEE International Conference on Data Mining Workshops, ICDM Workshops, Brussels, Belgium, December 10, 2012* (pp. 571–578). IEEE Computer Society. doi:10.1109/ICDMW.2012.106.

Priedhorsky, R., Culotta, A., & Valle, S. Y. D. (2014). Inferring the origin locations of tweets with quantitative confidence. In S. R. Fussell, W. G. Lutters, M. R. Morris, & M. Reddy (Eds.), *Computer Supported Cooperative Work, CSCW '14, Baltimore, MD, USA, February 15-19, 2014* (pp. 1523–1536). ACM. doi:10.1145/2531602.2531607.

Rahimi, A., Cohn, T., & Baldwin, T. (2015). Twitter user geolocation using a unified text and network prediction model. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers* (pp. 630–636). The Association for Computer Linguistics. URL: https://www.aclweb.org/anthology/P15-2104/.

Roller, S., Speriosu, M., Rallapalli, S., Wing, B., & Baldridge, J. (2012). Supervised text-based geolocation using language models on an adaptive grid. In J. Tsujii, J. Henderson, & M. Pasca (Eds.), *Proceedings of the*

2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea (pp. 1500–1510). ACL. URL: https://www.aclweb.org/anthology/D12-1137/.

Ryoo, K., & Moon, S. (2014). Inferring twitter user locations with 10 km accuracy. In C. Chung, A. Z. Broder, K. Shim, & T. Suel (Eds.), *23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014, Companion Volume* (pp. 643–648). ACM. doi:10.1145/2567948.2579236.

Schulz, A., Hadjakos, A., Paulheim, H., Nachtwey, J., & Mühlhäuser, M. (2013). A multi-indicator approach for geolocalization of tweets. In E. Kiciman, N. B. Ellison, B. Hogan, P. Resnick, & I. Soboroff (Eds.), *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, July 8-11, 2013*. The AAAI Press. URL: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/6063.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2), 461–464.

Shahraki, Z. K., Fatemi, A., & Malazi, H. T. (2019). Evidential fine-grained event localization using Twitter. *Information Processing & Management*, 56(6). doi:10.1016/j.ipm.2019.05.006.

Shuyo, N. (2010). Language detection library for java. URL: https://github.com/shuyo/language-detection.

Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411–423.

Williams, E., Gray, J., & Dixon, B. (2017). Improving geolocation of social media posts. *Pervasive and Mobile Computing*, 36, 68–79. doi:10.1016/j.pmcj.2016.09.015.

Zahra, K., Imran, M., & Ostermann, F. O. (2020). Automatic identification of eyewitness messages on twitter during disasters. *Information Processing & Management*, 57(1). doi:10.1016/j.ipm.2019.102107.

Zheng, X., Han, J., & Sun, A. (2018). A survey of location prediction on twitter. *IEEE Transactions on Knowledge and Data Engineering*, 30(9), 1652–1671. doi:10.1109/TKDE.2018.2807840.

Zola, P., Cortez, P., & Carpita, M. (2019). Twitter user geolocation using web country noun searches. *Decision Support Systems*, 120, 50–59. doi:10.1016/j.dss.2019.03.006.

Zubiaga, A., Voss, A., Procter, R., Liakata, M., Wang, B., & Tsakalidis, A. (2017). Towards real-time, country-level location classification of worldwide tweets. *IEEE Transactions on Knowledge and Data Engineering*, 29(9), 2053–2066. doi:10.1109/TKDE.2017.2698463.