Twitter user geolocation using web country noun searches

Paola Zola[a,*], Paulo Cortez[b], Maurizio Carpita[a]

[a] *Department of Economy and Management, University of Brescia, Brescia, Italy.*
[b] *ALGORITMI Centre, Department of Information Systems, University of Minho, 4804-533 Guimarães, Portugal*

Abstract

Several Web and social media analytics require user geolocation data. Although Twitter is a powerful source for social media analytics, its user geolocation is a nontrivial task. This paper presents an purely word distribution method for Twitter user country geolocation. In particular, we focus on the frequencies of tweet nouns and their statistical matches with Google Trends world country distributions (GTN method). Several experiments were conducted, using a recently created dataset of 744,830 tweets produced by 3,298 users from 54 countries and written in 48 languages. Overall, the proposed GTN approach is competitive when compared with a state-of-the-art world distribution geolocation method. To reduce the number of Google Trends queries, we also tested a machine learning variant (GTN2) that is capable of matching the GTN responses with an 80% accuracy while being much faster than GTN.

Keywords: Country Geolocation; Google Trends; Machine Learning; Natural Language Processing; Twitter.

1. Introduction

Due of the expansion of the Internet, Web and social media analytics are becoming a key element of many decision support systems. Modern Web plat-

* Corresponding author at: Department of Economy and Management, University of Brescia, C.da S. Chiara, 50 - 25121 Brescia, Italy.
Email addresses: paola.zola@unibs.it (Paola Zola), pcortez@dsi.uminho.pt (Paulo Cortez), maurizio.carpita@unibs.it (Maurizio Carpita)

March 18, 2019

forms, such as Twitter and Google Trends (GT), provide valuable big data that are easy to collect. Twitter is an important microblogging service with approximately 330 million active users that generate opinionated texts[1]. Twitter sentiment analysis has been used to predict stock markets [1], political elections [2], movie sales [3], and English Premier League soccer wins [4]. GT is another relevant Web source, providing Google statistics of search terms across different world regions. GT data-based analytics were used to predict flu trends [5], unemployment rates [6], consumer behavior [7], and the status of trending topics [8].

Several Web and social media analytics systems require user geographic location data. Examples include disaster early warning systems [9], property crime detection [10], event detection, epidemic dispersion, and news recommendations [11]. However, estimating the current location of a user is a nontrivial task for several microblogging services. For example, Twitter allows users to add profile locations and geographically tag their tweets, but the percentage of geotagged tweets is low [12, 13] and Twitter user profile location data is often unreliable [14].

In this paper, we present a novel statistical approach for country-level location detection of Twitter users. This geolocation is potentially valuable in several decision support system applications, allowing them to easily filter users from a specific country. For instance, it can be used in Twitter sentiment analysis related to country commodity prices, such as steel, silver, or cotton prices.

Our approach assumes that people tend to write about news, events, and so on, from the country to which they are more related. It follows that, even if a user lives in country A, she/he might be more interested in news or information linked to another country B, so the potential information held in the user's tweet is likely to refer to country B. Consider the following examples related to two tweets about steel production:

1. "chinese steel rebar production reach the maximum over a year"; and

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

2. "downhill price for steel beams".

Although it is clear for the first tweet example that the country of interest is China, for the second one it is not possible to link the information to a specific country. In contrast with the stock market domain, where easy identifiable cashtags[2] are common (for example, $AAPL for Apple stocks) [1], commodity country-specific tweets tend to be similar to the second tweet example: unstructured and without an obvious geographic term, hashtag, or cashtag. Moreover, these tweets are often written in English, so they could be related to any country's market. It follows that our approach aims to associate a tweet with a highly probable country context when such a geographic context is not explicitly known to assist in country-level Twitter analytics.

To identify the unknown country, we analyze the word distribution of past user tweets. In contrast with previous studies that use specific geographical dictionaries, based on named-entity recognition (NER) modules [15], we consider generic nouns. As shown in Section 4.2, these nouns can incorporate geographic terms (like NER) but also non-geographic terms that are specific to a country. Examples of such nouns include "Brexit" (related to the United Kingdom), "Trump" (United States of America) and "cricket" (popular in Pakistan). In addition, because of cultural differences, there are nouns that are used in distinct countries with different frequencies (for example, "thanks" in Table 9) and such information can potentially aid in country discrimination. Moreover, non-English users can tweet in their native languages, and so non-English nouns (for example, "sono" and "stato" for Italy) can help in determining the country. To take advantage of this implicit information, we perform matching between frequent country-level GT and user tweet nouns (GTN). To the best of our knowledge, this is the first time that GT data has been used to detect geographical user information.

As a case study, we consider the steel production domain and recent Twitter

[2] https://techcrunch.com/2012/07/30/twitter-clickable-ticker-symbols/

data, which includes 744,830 tweets from 3,298 users. Following an empirical design science research approach [16], we show that our GTN model is competitive when compared with a state-of-the-art NER [15] (Section 4.1). To reduce the GT querying time, we also propose a GTN variant that uses machine learning (for example, deep multilayer perceptron and random forest) to learn the GT responses (Section 4.3). Finally, we demonstrate the applicability of GTN to non-steel commodity domains using more recent Twitter data and a different but smaller sample of users (Section 4.4).

The contributions of the proposed approach include:

1. We perform a Twitter estimation of the most probable user country of interest when such explicit context is not known.
2. The estimation is based on generic nouns, retrieved from the user's historical tweets, which can include geographic words and other country-specific terms (including news, sports, religion, events, people, and native language nouns).
3. The proposed Google Trends nouns (GTN) method uses GT to solve a spatial detection task rather than a temporal task (as proposed in previous GT studies).
4. To reduce the GT query time, we proposed a second approach, termed GTN2, that uses machine learning.
5. We created a recent dataset related to the steel domain, which includes a conservative country estimate for 3,298 users, to empirically compare GTN with a state-of-the-art NER.

The paper is organized as follows. Section 2 discusses the literature review related to social network location estimation methods. Section 3 details the country-level Twitter estimation methods. Section 4 presents and analyzes the experimental results. Finally, Section 5 summarizes the work, highlighting its main advantages, limitations, and future directions.

2. Related work

Several studies have investigated Web and social network user location estimation. Before the rise of social networks, the Internet protocol (IP) address was the main element used for Web geotagging [17]. However, microblogs typically do not provide IP addresses. Moreover, the increasing use of virtual private networks (VPNs) reduces the reliability of IP address location.

Focusing on Twitter, user geographic estimation is a nontrivial task. Twitter location data can be directly retrieved by accessing geotagged tweets or user location field profiles. However, only a small fraction of tweets are geotagged. For example, the literature mentions low percentage values, varying from 0.42% [12] to 3.17% [13]. While mobile devices are increasingly used, users often switch off global positioning system (GPS), for privacy reasons or to save battery consumption. Moreover, although Twitter users can add a geographic reference to their profiles, the field is free text and often unreliable locations are used (for example, "in your heart" or "everywhere"). Hecht et al. [14] estimate that approximately 34% of Twitter users add nonrealistic text locations.

Table 1 summarizes the state-of-the-art research work on social network user location estimation, using chronological order and emphasizing the Twitter data source (**data source** column). There are three main types of social network user location estimation methods (**type** column):

1. Image recognition (IR): digital photos posted on social networks provide a vast amount of information, including location. For instance, Aulov et al. [18] studied the Deepwater horizon oil spill disaster in the Gulf of Mexico using Flickr photos and locating them to the desired area.
2. Friendship network (FN): the assumption is that the user's location can be inferred by the locations of her/his friendship network. Examples of work that followed this assumption are [19, 20, 21].
3. Word distribution (WD): related to our approach, it includes methods that are based on text analysis and word extraction. Some studies use existing NER modules, location indicative words (LIW), and gazetteers (ge-

ographic dictionaries) to extract locations from tweets (e.g., [15]). Other studies are based on tweet word frequencies, proposing methods to filter local words [12, 22, 23].

Some studies complement the previous methods with the use of additional features (AF), such as the location field from the user profile metadata [24, 25] or the tweeted time zone [11]. Other studies combine the different types, such as: IR and WD [26]; WD and FN [27, 28, 29, 30, 31]; and WD, FN, and AF [25].

The related work can also be characterized by the text **language**, location **target**, discrimination **level**, search **area** of interest, computational **algorithm**, evaluation method (**val.**), and **metric**. The type of language is often associated with the search area. In most cases, the messages are written in English. Regarding the target, while some studies focus on where the tweet was written (e.g., [32, 24, 25]), the majority try to detect the user's home location (e.g., [12, 27, 34, 29, 31]). As for the discrimination level, there are two main approaches: detecting larger regions (e.g., countries or states) or smaller regions (e.g., cities, landmarks, geographic coordinates, or postal codes). Some fine-grained level detection methods (e.g., geographic coordinates) are often associated with a specific geographic area and events, such as natural disasters or emergency responses [18, 36, 24, 39]. The location level often affects the type of evaluation **metric** used. Large region discrimination methods tend to perform multiclass tasks, so common classification metrics [41] are often adopted (e.g., accuracy, precision, or recall). More diverse measures are used by the small region discrimination methods, including standard classification metrics (e.g., accuracy and precision), classification accuracy within a tolerance radius (Acc@R), or even regression metrics (e.g., root mean squared error). A wide variety of algorithms were adopted, including: approaches based on data frequency and statistics (e.g., information gain), generic machine learning models (e.g., neural network, support vector machine, or random forest), and specific geographic/Twitter-dependent methods (e.g., geocontext locator, geoparsing,

Table 1: Summary of the related work.

Study	Type[a]	Lang.[b]	Data Source[c]	Tar.[d]	Level[e]	Data Period[f]	User size[f]	Data size[f]	Val.[g]	Area[h]	Algorithm[i]	Metric[j]
Crandall et al. [26]	IR,WD	EN	Flickr	F	SP	ND	307K	33M	ND	W	BC,SVM	Acc
Backstrom et al. [19]	FN	EN	TW	U	SP	ND	2.9M	ND	ND	USA	MLE	Acc@25mi
Cheng et al. [12]	WD	EN	TW	U	CI	2009-10	1M	3M	*10*CV	USA	MLE	Acc@100ml
Davis et al. [20]	FN	PT	TW	T	CI	ND	25K	ND	*10*CV	BR	DFS	P
Kinsella et al. [32]	WD		TW	T	CO,SP	2010	7M	ND	*5*CV,HO	W	PM,KL,QL	Acc
Aulov et al. [18]	IR	EN	Flickr	F	SP	2010	ND	190	ND	MXG	GNOME	RMSE
Dalvi et al. [22]	WD	EN	TW	T	CI	2009-11	14M	200M	ND	USA	DM,LM	P,R
Li [27]	WD,FN	EN	TW	U	CI	2011	4.0M	ND	*5*CV	USA	UDI	Acc
Chang et al.[33]	WD	EN	TW	U	CI	2009-10	136K	9M	HO	USA	GMM,LM, MLE	Acc
Compton et al. [34]	FN	EN	TW	U	CI	2012-14	110M	ND	*5*CV,HO		TVM	ME (km)
Han et al. [35]	WD	EN Mixed	TW	U	SP	2011-12	500K 1.4M	38M 12M	*10*CV	NA W	DFS	Acc@161km
Mahmud et al.[11]	WD,AF	EN	TW	U	SP	2011	10K	1M	*10*CV	USA	HE	Acc@100mi
Middleton et al.[36]	WD	EN,TR, IT,PT	TW	T	SP	2011-13	ND	1.5M	ND	USA	G	F1
Ryoo et al. [23]	WD	KR	TW	U	SP	2010-11	3.3M	615M	*5*CV	KR	PGM	Acc@10km
Minot et al. [28]	FN,WD		TW	U	CI	2014	29K	7.0M	ND	AFR	SVM, CBF	Acc@10km
Lee et al. [15]	WD	EN	TW	T	ST	2013-14	ND	113K	*10*CV	USA	SVM, BC,RF	R
Rahimi et al. [21]	FN	EN	TW TW	U	SP	2011-12	9.5K 450K 1.4M	380K 39M 12M	HO	USA USA W	LP	Acc@161km
Rahimi et al. [29]	FN,WD	EN	TW TW TW	U	SP	2011-12	9.5K 450K 1.4M	380K 39M 12M	HO	USA USA W	LP	Acc@161km
Rodrigues et al. [30]	FN,WD	PT	TW	U	CI	2010	12K	2M	*10*CV	BR	MM,BC, MRW	Acc
Kotzias et al.[37]	FN	EN	TW	U	CI	2013	43K 40K 55K	1.9M 1.3M 1.5M	*10*CV	IR UK USA	LDA	P
Laylavi et al. [24]	WD,AF	EN	TW	T	SP	2015	2K	ND	ND	AUS	MELI	Acc@12.2km
Singh et al. [38]	WD	EN HI	TW	T	SP	2015-16	ND	32K	ND	IN	MM	Acc
Williams et al. [25]	WD, FN,AF	EN	TW	T	SP	2016	15K	ND	ND	W	GCL	Acc@5km Acc@160km
Quian et al. [31]	FN,WD	EN, ZH	TW TW Weibo FB	U	CO,CI	2011	1.5M 329K 1.0M 1K	ND	HO	W USA CH ND	NN	Acc
Avvenuti et al. [39]	WD	EN, IT	TW	T	SP	2011-15	ND	9K 2K	ND	W IT	G	Acc
Rahimi et al. [40]	FN,WD	EN	TW TW TW	U	SP	2011-12	9.5K 450K 1.4M	380K 39M 12M	HO	USA USA W	NN DCCA	Acc@161km
This work	WD	Mixed	TW	U	CO	2017	49K	21M	*10*CV	W	GTN,GTN2	Acc, WF1

[a] image recognition (IR), friendship network (FN), word distribution (WD), additional features (AF).

[b] **Language**: Chinese (ZH), English (EN), Hindi (HI), Italian (IT), Korean (KR), Portuguese (PT), Turkish (TR); mixed: combination of multiple languages.

[c] Facebook (FB), Twitter (TW).

[d] **Target**: Flickr picture location (F), tweet location (T), user's home location (U).

[e] city (CI), country (CO), one of 50 states (ST), specific place (SP) from a region (e.g., coordinates, landmark or ZIP code).

[f] nondisclosed (ND), thousand (K), million (M); user and data size represent the initial collected values, before filtering.

[g] **Validation**: n-fold cross validation (nCV), hold out (HO), nondisclosed (ND).

[h] Africa (AFR), Australia (AUS), Brazil (BR), China (CH), India (IN), Ireland (IR), Italy (IT), Korea (KR), Mexican Gulf (MXG), nondisclosed (ND), North America (NA), United Kingdom (UK), United States of America (USA), World (W).

[i] Bayesian classifier (BC), consensus-based fusion (CBF), data frequency or statistic (DFS)-based, deep canonical correlation analysis (DCCA), distance model (DM), geoparsing-based (G), geocontext locator (GCL), Gaussian mixture model (GMM), general NOAA oil modeling environment (GNOME), Google Trends nouns (GTN), Google Trends nouns and machine learning (GTN2), hierarchical ensemble (HE), Kullback-Leibler (KL) divergence, label propagation (LP), language model (LM), latent Dirichlet allocation (LDA), Markov model (MM), maximum likelihood (MLE)-based, multi rank walk (MRW), multi-elemental location inference (MELI), neural network (NN), placemaker (PM) using tweet content, probabilistic generative model (PGM), query likelihood (QL), random forest (RF), support vector machine (SVM), total variation minimization (TVM), unified discriminative influence (UDI) model.

[j] accuracy (Acc), accuracy using a radius of R (Acc@R, R in miles (mi) or kilometers (km)), F1-score (F1), mean error (ME), precision (P), recall (R), root mean square error (RMSE), weight averaging F1-score (WF1).

or placemaker using tweet content). These algorithms were validated using either the simpler holdout (train and test split) or the more robust k-fold cross-validation.

The last row of Table 1 positions our work, which assumes a pure WD approach, a country-level detection, and multilingual tweets (mixed). The main novelty is the usage of generic nouns and GT source (the GTN method), as detailed in Section 3 and compared with a state-of-the-art WD method [15] in Section 4.

3. Data and methods

3.1. Data

Using automatic computational code (written in Python and R) and tools, we created a dataset with recent Twitter data to test the country geolocation methods. As an example in the decision support system application domain, we have targeted steel alloy. For the initial selection of users, we selected all tweets that included one of the keywords {"steel price", "steel industry", "steel production"}, from March to November 2017. These queries resulted in 138,484 tweets, related to 49,203 users. Only a tiny fraction of the tweets (192) were geotagged. In addition, only 33,886 users had a filled location profile field. We note that, in this work, retweets are treated in the same manner as common tweets, because retweets might be helpful in identifying the user's country of interest (e.g., retweets of a politician).

To set the ground truth, we designed a conservative procedure that discards a large number of users but is more reliable for comparing geolocation methods. The procedure is based on a strong double-source verification that considers both metadata (user profile location field) and LIW from historical user tweets. We considered the set of 33,886 users with some location profile data and retrieved up to a maximum of 3,200 past tweets for each user. We then used `OpenNLP` [42] and the `ggmap` R package [43] tools to extract LIW from the historical tweets (`OpenNLP`) and obtain the Google Maps country for each LIW (`ggmap`).The most

frequent country, computed over the full set of LIW for a given user, was then compared with the metadata information. After removing country mismatches, including metadata with slang and nonrealistic locations, the final ground truth dataset contains 3,298 users and 744,830 tweets, representing an average of 226 tweets per user.

While all selected users have written at least one English term, from the set {"steel price", "steel industry", "steel production"}, the collected historical tweets were written by users from both native English speaking (e.g., Australia) and non-native English speaking (e.g., Spain) countries. Table 2 presents the percentage of tweets written in a specific language (**tweets** column) and the percentage of users per country (**users** column). Figure 1 plots these last values visually on a world map (the higher the percentage, the darker is the country color). The language values were obtained by using the `textcat` R package [44]. The majority of the tweets were written in English (66.2%), followed by the German (18.8%) and Catalan (4.4%) languages. As for the countries, most

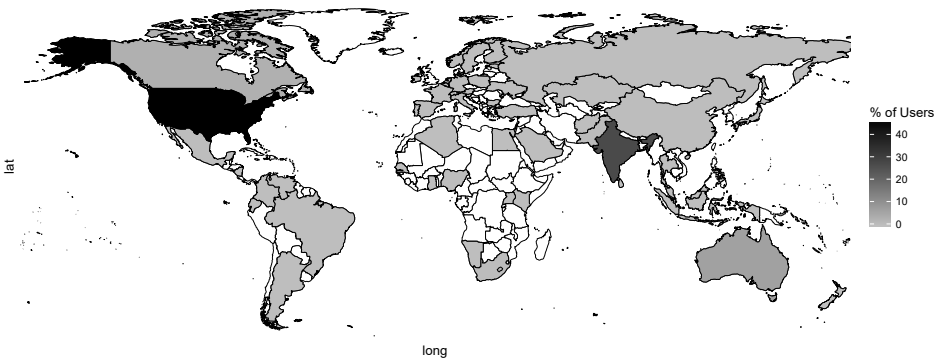

Figure 1: Percentage of users per country plotted on a world map.

users come from anglophone countries, such as United States of America (USA) (45.7%), United Kingdom (UK) (12.3%), and Australia (6.4%). As for the non-anglophone countries, most users are from India (27.1%), while other countries

Table 2: Dataset tweet languages and users per country.

Language	**Tweets**	**Country**	**Users**
English	66.2%	United States of America (USA)	45.7%
German	18.8%	India	27.1%
Catalan	4.4%	United Kingdom (UK)	12.3%
Danish	1.9%	Australia	6.4%
Nepali	1.3%	Canada	3.1%
Indonesian	1.1%	Germany	0.5%
Latin	0.9%	Pakistan	0.5%
Rumantsch	0.8%	South Africa	0.4%
Slovak	0.9%	China	0.3%
French	0.4%	France	0.3%
Esperanto	0.3%	Nigeria	0.3%
Swahili	0.3%	Spain	0.3%
Sanskrit	0.3%	Kenya	0.2%
Spanish	0.2%	Italy	0.2%
Romanian	0.2%	Mexico	0.2%
Swedish	0.2%	Finland	0.1%
Czech	0.2%	Ireland	0.1%
Malay	0.1%	Japan	0.1%
Hungarian	0.1%	Argentina	0.1%
Afrikaans	0.1%	Belgium	0.1%
Slovenian	0.1%	Brazil	0.1%
Dutch	0.1%	Colombia	0.1%
Tagalog	0.1%	Indonesia	0.1%
Basque	0.1%	Malaysia	0.1%
Others	0.6%	Others	1.2%

are much less prevalent (e.g., Germany with 0.5%). In total, the dataset contains tweets written in 48 languages and users from 54 countries.

Only one state-of-the-art study performed a mixed-language tweet geolocation [35], as shown in Table 1. Our work does not separately consider datasets

of tweets written in a specific language, because it is more trivial to identify the country when the language is distinctive of a nation (e.g., Japanese).

Following the work of [35], we adopted a mixed language approach, which is more natural for the geolocation of countries, because Twitter is a multilingual platform. Nevertheless, the values in Table 2 reflect the steel domain scenario. Therefore, most of the tweets are written in English, which is a geographically widespread language that is more difficult to geolocate [35], making this dataset challenging and interesting for comparing purely WD methods.

3.2. Google Trends nouns

As explained above, the proposed GTN WD approach uses only tweet nouns, because we assume they are the most representative part of speech able to identify different countries.

For user u, the GTN approach works by first identifying the sequence of the most frequent nouns $\mathbf{n}_u = \langle n_1, n_2, ..., n_{l_u} \rangle$, in descending order and with a length of l_u elements. To obtain $\mathbf{n}_u$, the tweets are first preprocessed by transforming the text to lowercase and removing English stopwords. The `TextBlob` Python module is then used to extract noun phrases and then the nouns. We note that the `TextBlob` module is faster than other tools [45].

For each noun $n_i \in \mathbf{n}_u$, a GT query is executed by using the `Pytrends` Python module. To limit the number of queries, a fixed pruning threshold (p) is used, such that $l_u \leq p$ for all u users. The GT query result for noun n_i is a sequence with integer confidence scores for an alphabetic list of countries C with a length of $l_c = 250$. The scores range from 0 (lowest confidence) to 100 (highest confidence). Let $\mathbf{G}_u$ denote the GT confidence score matrix for user u with a size of $l_c \times l_u$, where each score is represented as $g_{c,i}$ for country $c \in C$ and the i-th most frequent noun. We test three strategies to weight the GT scores, resulting in the weighted confidence score matrix $\mathbf{S}_u$ ($l_c \times l_u$) with the elements $s_{c,i}$ (country, noun):

- equal weights (EQ): no weights are used, and so $s_{c,i} = g_{c,i}$.

- Internet usage (IU): weighted according to the fraction of Internet users for a specific country c (w_c) according to the World Bank statistics[3]:

$$\forall c \in C, \forall i \in \{1, ..., l_u\} : s_{c,i} = w_c g_{c,i} \tag{1}$$

- nouns frequency (NF): weighted according to the order of the nouns (more frequent nouns have stronger weights):

$$\forall c \in C, \forall i \in \{1, ..., l_u\} : s_{c,i} = w_i g_{c,i} \tag{2}$$

  where $w_i = (l_u - i + 1)/l_u$.

Once the confidence score is computed, we explore two statistical approaches to estimate the most probable country c_u for user u:

- join frequency (JF) – based on the highest score country when summing all noun scores:

$$c_u = \underset{c}{\text{argmax}}(\sum_{i=1}^{l_u} s_{c,i}) \tag{3}$$

- absolute frequency (AF) – selects the most common country (mode) when considering the highest score countries for all nouns:

$$c_u = Mode(argmax_c(s_{c,i}) \forall i \in \{1, ..., l_u\}) \tag{4}$$

  where $Mode$ denotes the mode of a set.

3.3. Machine learning

In this paper, we use machine learning for three different goals: to obtain the benchmark geolocation method outputs (for comparison purposes with GTN); to access the quality of the proposed GTN; and to mimic the GTN responses. For all three goals, the input features consist of the classical bag-of-words (BoW) [46], in a total of 24,269 unique nouns for the 3,298 users considered. The classifier output is the geolocation country but the target values depend on the

[3] `https://data.worldbank.org/indicator/IT.NET.USER.ZS`

machine learning goal. The first goal is detailed in Section 3.4. The second goal is applied during the error analysis procedure [47], to verify whether the GTN errors are solvable by machine learning. The third goal, termed the GTN2 method here, is used to reduce the number of GT queries. Similarly to other Web query geolocation methods (for example, based on Google Maps), GTN requires a substantial computational effort because of the large number of GT requests. To solve this problem, we use GTN as an oracle, providing the target classification responses for the machine learning methods.

We explore four classification algorithms with powerful learning capabilities [48, 49]: bagging (BG), random forest (RF), support vector machine (SVM), and a deep learning multilayer perceptron (MLP).

Breiman's bagging or bootstrap aggregation algorithm (BG) trains t independent classifiers on a given training set by sampling, with replacement, instances from the training set. The essential idea is to average noise and avoid overfitting by using unbiased models that reduce the variance [48]. Bagging is normally applied using decision trees as the individual weak learners, which corresponds to the BG model used in this work.

RF is a successful model that was proposed in 2001: it combines t decision trees based on bagging and random selection of input features [50]. RF tends to obtain good classification results even when using its default parameters and when no feature selection method is adopted [48]. In a recent large comparison study, the RF classifier was ranked as the best classifier among 17 of the main machine learning types of algorithms [51].

SVM are widely used in text classification [52]. The model is based on a maximized margin criterion [53]. For binary classification, the SVM algorithm can compute the best separating hyperplane in a feature space, which is defined by a kernel transformation. In this work, we adopt the linear kernel, because it is very fast and works well with high-dimensional input features, which is the case with our nouns dataset. The model contains one hyperparameter (C) that controls the tradeoff between fitting the errors and obtaining a smooth decision boundary. Because we have 54 class labels, we used the one-vs-rest multiclass

classification, which involves training a single classifier per class [54].

Moreover, recent remarkable developments were proposed in the field of deep learning, leading to neural network architectures that obtained the best results in diverse competitions (for example, computer vision and natural language processing) [55]. Such success revived the popularity of the MLP neural model. In this work, we assume a modern MLP representation, also known as deep feedforward neural network [49], with three hidden layers (with h_1, h_2, and h_3 hidden nodes) that uses [55]: the ReLU activation function on all hidden units, the Softmax function on the output layer, a dropout regularization, and early stopping (to reduce overfitting).

All classifiers are evaluated by using an external 10-fold cross-validation scheme, as explained in Section 3.5. For each of the 10 cross-validation iterations, the available data is divided into training data (90% of the instances) and test data (10%). The test data is used to measure the classification performance of the selected models. The training data is used to fit the machine learning models and to perform the hyperparameter selection. To reduce the bias towards a particular model [56], we apply the same hyperparameter selection procedure for BG, RF, SVM, and MLP. Using standard practice [48, 47], the training data is further split into training and validation sets (internal holdout validation). The training set, with 80% of the training instances (0.8×0.9 =0.72% of all available data), is used to fit the classifier. The validation set, with the other 20% of the training data examples (0.18% of all data), is used to monitor the best generalization capability, in terms of global classification accuracy, associated with a hyperparameter or set of hyperparameter values. After selecting the hyperparameters, the machine learning model is retrained with all training data. To provide a fair comparison, we applied a grid search with 10 different hyperparameter combinations for each machine learning algorithm. For BG and RF, the number of trees ranged through $t \in \{50, 100, 150, 200, 250, 300, 500, 1000, 1500, 3000\}$. For SVM the C parameter was searched using $C \in \{0.01, 0.05, 0.1, 0.2, 0.5, 1, 5, 10, 50, 100\}$. For MLP, we tested ten different MLP models, which correspond to different combinations of numbers of hidden nodes

and dropout values, as detailed in Table 3. The number of MLP inputs is large, because it includes all unique dataset nouns. Therefore, to reduce computational effort, and following what is suggested in [57], the MLP combinations assume a decreasing hidden layer size structure, where $h_1 > h_2 > h_3$. The other parameters were set to their default values, as implemented using the `keras` and `sklearn` Python modules.

Table 3: Different MLP models tested during the hyperparameter selection stage.

Model Number	Hidden layer size 1 (h_1)	Hidden layer size 2 (h_2)	Hidden layer size 3 (h_3)	Dropout
1	200	100	70	0.4
2	200	100	70	0.3
3	300	150	50	0.4
4	300	100	50	0.4
5	500	200	100	0.4
6	500	200	50	0.4
7	200	150	50	0.4
8	200	150	50	0.3
9	500	150	70	0.4
10	500	100	50	0.4

Because the country classes are unbalanced (for example, 45.7% of users are from the USA, while only 0.1% are from Brazil; see Table 2), we applied an oversampling procedure [58] to all training sets of the machine learning algorithms. The goal is to improve classifier performance for the minority classes by performing random sampling, with repetition, such that the training set becomes balanced. We note that we did not consider undersampling because some classes are very rare, and so undersampling would lead to very small training sets. In addition, the test sets retain the original unbalanced class distribution.

3.4. Benchmark methods

For comparison purposes, we selected a recent WD geolocation benchmark method (BM) [15] that can be simulated using similar procedures and tools already used in this research. The BM method first uses an NER tool (`Stanford CoreNLP`[4]) to extract geolocation terms. The terms are fed to Google Maps to

[4] `https://stanfordnlp.github.io/CoreNLP/`

obtain the geographic coordinates. When Google Maps does not return a single country, this is considered an ambiguous case, which is then estimated by using a machine learning algorithm: naive Bayes, SVM, or RF. Using only training data (the BoW approach), the algorithm is fitted to the subset of unambiguous cases and then used to predict all ambiguous cases, including those from the test data. Because RF achieved the best results in [15], we adopt this learning classifier for BM. We also test a hybrid benchmark method (BM2), which works similarly to BM except that the ambiguous cases are estimated using GTN instead of the learning classifier (RF).

3.5. Evaluation

The created Twitter dataset is described in Section 3.1; it includes 3,298 users (instances) related to 54 countries. The input features consist of 24,269 unique nouns. The countries were identified by the ground truth procedure that is based on a conservative double-source verification, which considers both metadata (user profile location field) and LIW, given all historical tweets (744,830 messages). The Twitter user country geolocation is modeled as a multiclass task (with 54 output labels), and so common classification performance metrics are adopted. The confusion matrix maps predicted values to actual values. From this matrix, several multiclass performance measures can be computed. For a particular class c, we use [41]: accuracy$_c$ (Acc$_c$), precision$_c$, recall$_c$, and F1-score$_c$.

To obtain a single performance measure from the multiclass results, we adopt global accuracy (Acc), which is widely used in classification tasks. The F1-score is a more reliable measure when the data are unbalanced, which is true in our case (as shown in Table 2). Therefore, we also compute a single global F1-score by performing a weight averaging operation (WF1), in which each F1-score is weighted proportionally to the class frequency in the data. The evaluation metrics were computed using the `sklearn` module.

GTN is a statistical approach that does not require training data. Nevertheless, for comparison with the machine learning approaches (Table 12), we adopt

the popular 10-fold cross-validation scheme (Section 2) in all comparison tests. The data are randomly split into ten equal-sized folds; then, using a rotation scheme, one fold is selected for testing and all of the others are used for training (if needed by the method). This results in 10 sets of predictions and desired values for each method. To aggregate the results, we average the $k = 10$ distinct classification performance results, and the statistical significance is obtained by applying the nonparametric Mann-Whitney test [59].

4. Results

4.1. Google Trends nouns results

We conducted preliminary experiments with GTN, to tune the method. The preliminary experiments considered a random subset of our data related to 267 users (8%). Adopting the EQ and JF methods, we first tested distinct pruning threshold values, which were based on some noun distribution statistics (median, sixth percentile, third quartile, mean): $p \in \{112, 156, 298, 770\}$. The best results (with an accuracy of 76.0%) were achieved for $p = 298$, which was fixed. Using the same preliminary sample, we then compared different weighting methods for the country confidence scores and country classification, in a total of six GTN models (Table 4). The best classification results were achieved by the first model, which uses EQ and JF, becoming the selected configuration for the GTN method.

The average 10-fold country geolocation results for GTN and benchmark methods are presented in Table 5. When analyzing both classification metrics, global accuracy (Acc) and weight-averaging F1-score (WF1), the comparison clearly favors GTN with respect to the state-of-the-art WD method (BM), showing a substantial difference (15.7 percentage points for Acc and 8.5 percentage points for WF1) that has statistical significance. The hybrid NER GTN method (BM2) provides better performance than BM, indicating that GTN handles the ambiguous cases better than RF. Nevertheless, GTN achieves the best overall

Table 4: Comparison of different GTN weighting and country classification strategies (**bold** denotes best value).

Model	Score Weighting	Classification Strategy	Acc
1	**EQ**	**JF**	**76.0**
2	EQ	AF	73.0
3	IU	JF	56.6
4	IU	AF	40.1
5	NF	JF	75.3
6	NF	AF	45.3

results, with an improvement of 2.3 percentage points for Acc and 1.8 for WF1, although these are not statistically significant.

Table 5: Country geolocation results (in %, best dataset values in **bold**).

Metric	BM	BM2	GTN
Acc	64.9	78.3	**80.6**$^{\diamond}$
WF1	72.8	79.5	**81.3**$^{\diamond}$

$\diamond$ – Statistically significant under a pairwise comparison when compared with BM (p-value < 0.05).

4.2. Error analysis

To better understand the errors produced by GTN, we performed an error analysis [47], in which we manually inspected a total of 638 Twitter user accounts related to GTN country misclassification examples. Table 6 details the errors in terms of four main categories (**error type** column). There are 76 cases (11.9%) for which GTN provided the correct classification (error type A) when the conservative ground truth method (Section 3.1) was wrong. These cases are mostly related to user metadata with ambiguous geolocation terms that can refer to more than one anglophone country (for example, "Newport" city can refer to USA or UK; see Table 7). We have recomputed the classification performance for GTN, BM, and BM2 by using the manually adjusted

76 "true" cases. The results obtained are presented in Table 8, which confirms that the "true" classification performance for GTN is actually higher than the results shown in Table 5. In fact, in Table 8 the GTN achieves an Acc of 83.0% and a WF1 of 83.4%. We particularly note that GTN statistically outperforms both benchmark methods (BM and BM2) when adjusted to the "true" values. A common GTN error (type B) is an anglophone country mismatch (32.0%, e.g., UK or Canada instead of USA). There are also some errors (type C, 3.1%) related to proximate countries when considering the location (e.g., Belgium and Netherlands) or language (e.g., Portugal and Brazil). Most GTN mismatches (type D, 53.0%) are related to other mismatches not included in the previous error types. Table 7 reports some examples of the A, B, C, and D error types. In the Table 7, the user name is omitted for privacy reasons.

Table 6: Error analysis for GTN.

Error type	**Number**	**Percentage**
Correct classification (A)	76	11.9
Anglophone mismatch (B)	204	32.0
Close country by language or location (C)	20	3.1
Other mismatches (D)	338	53.0
Total	638	100.0

To better exemplify how the nouns can be associated with countries, we present the distribution of the ten most frequent nouns used by the GTN method to identify the country. Table 9 is related to a sample of four anglophone countries (Australia, Canada, UK, and USA), while Table 10 shows the most frequent nouns for four examples of non-anglophone countries (Finland, Italy, Pakistan, and Singapore). To create the tables, we considered all nouns from all users that were correctly classified by the adjusted GTN model of Table 8. The respective classification accuracy (Acc) values for the selected country examples are: Australia – 80%, Canada – 32%, UK – 81%, USA – 94%, Finland – 75%, Italy – 100%, Pakistan – 74%, and Singapore – 100%.

Table 7: Examples of misclassified locations.

Error type	**Lang.**[a]	**Metadata location**	**Ground truth**	**GTN**	**Manual assessment**
A	EN	Newport	USA	UK	UK
A	EN	North East	USA	UK	UK
B	EN	Scotland	UK	USA	UK
C	NL	Mechelen	Belgium	Netherlands	Belgium
C	ES	Barcelona	Spain	Guatemala	Spain
C	EN	Suri	India	Bangladesh	India
C	PT	Portugal	Portugal	Brazil	Portugal
D	ES	Philadelphia	USA	Colombia	USA

Language: English (EN), Dutch (NL), Portuguese (PT), Spanish (ES).

Table 8: Country geolocation results for the adjusted ground truth (in %, best dataset values in **bold**).

Metric	**BM**	**BM2**	**GTN**
Acc	63.6	79.1	83.0$^{\diamond}$
WF1	71.6	80.1	83.4$^{\diamond}$

$\diamond$ – Statistically significant under a pairwise comparison when compared with BM and BM2 (p-value < 0.05).

Tables 9 and 10 show specific geographic terms that can be used to identify the country, working similarly to an NER tool. These include geographic nouns such as: "australia", "sydney", "canada", "scotland" (Table 9); and "finland", "oulu", "pakistan" (Table 10). GTN also benefits from language differences, as shown by the Italian examples of Table 10. However, even when considering the English language, there are also non-geographic terms (not used by NER) that do seem country specific and so can contribute added discrimination capability to GTN. For instance, "brexit" is associated with the UK, while "trump" is related to the USA. For Pakistan there are several other examples of country-specific terms, such as "maryamnsharif" (popular Pakistani politician), "cricket" (highly popular in the country), and "allah" (religion). A

different interesting example is provided by the term "thanks", which is used in three anglophone countries (Canada, UK, USA) but with different frequencies (e.g., 0.46% in Canada vs 0.22% in USA). This might be because of cultural differences between countries. In contrast, there are other nouns that are often used with similar frequencies, such as "time" (0.39% for Canada and USA) and "year" (0.29% for Canada and 0.33% for USA). These generic nouns limit the GTN capability to discriminate between countries that use the same language, as shown by the anglophone errors of Table 6.

Table 9: Most frequent nouns for four examples of anglophone countries.

Australia		**Canada**		**UK**		**USA**	
Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency
year	0.35%	canada	0.51%	time	0.46%	time	0.39%
time	0.31%	thanks	0.41%	people	0.42%	people	0.34%
people	0.30%	time	0.39%	news	0.34%	year	0.33%
australia	0.28%	year	0.29%	thanks	0.33%	news	0.26%
world	0.28%	business	0.29%	year	0.32%	trump	0.25%
news	0.26%	project	0.27%	work	0.29%	work	0.24%
work	0.24%	industry	0.27%	brexit	0.28%	world	0.23%
business	0.24%	news	0.24%	christmas	0.27%	life	0.22%
industry	0.22%	work	0.24%	scotland	0.26%	years	0.22%
sydney	0.21%	check	0.24%	government	0.24%	thanks	0.22%

Table 10: Most frequent nouns for four examples of non-anglophone countries.

Finland		**Italy**		**Pakistan**		**Singapore**	
Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency
congratulations	0.34%	sono	0.50%	pakistan	1.16%	china	0.62%
camp	0.22%	perch	0.40%	maryamnsharif	0.73%	steel	0.62%
finland	0.22%	anche	0.40%	people	0.58%	price	0.47%
business	0.22%	stato	0.30%	allah	0.58%	prices	0.47%
thesis	0.22%	grande	0.30%	world	0.44%	time	0.47%
time	0.22%	posso	0.30%	cricket	0.44%	year	0.47%
seminar	0.22%	prima	0.30%	pakistani	0.44%	data	0.47%
technology	0.22%	bella	0.30%	morning	0.44%	report	0.47%
oulun	0.22%	bello	0.30%	imran	0.44%	conference	0.47%
oulu	0.22%	alla	0.30%	army	0.44%	trade	0.47%

Following Table 6, we performed another error analysis step in which machine learning was used. We considered two machine learning error analysis setups:

- **I** – The 204 misclassified user examples who live in anglophone coun-

tries (Table 6) are removed from the dataset and are always used as the same test set in the 10 iterations of the 10-fold procedure. The remaining dataset examples pass through a 10-fold validation, to generate 10 training sets and learning models that are tested on the same 204 test set cases.

- **II** – Similar to the previous setup, except that the fixed test set is composed of all $638 - 76 = 562$ "true" misclassified users (Table 6).

The machine learning models require a substantial computational effort because the nouns dataset is high-dimensional, with 24,269 features and 3,298 instances. To reduce the computational effort, the hyperparameter selection is first applied to the dataset, from Section 3.1. The best hyperparameters for each classifier are then fixed and used in the 10-fold evaluation of all machine learning comparisons (setups I and II and experiments of Section 4.3). The hyperparameter selection procedure uses a 10-fold validation. During each 10-fold iteration, the training data is split using an internal holdout (80%/20%). For each learning algorithm, ten different models (described in Section 3.3) are trained. The best hyperparameter values are selected as the best 10-fold mean global accuracy (Acc) and this resulted in: BG – $t = 300$ trees, RF – $t = 150$ trees, SVM – $C = 0.01$, and MLP – model 7 of Table 3 ($h_1 = 200$, $h_2 = 150$, $h_3 = 50$, dropout=0.4).

The machine learning error analysis results are presented in Table 11. The obtained classification measure values (WF1 and Acc) range from 21% (setup I, Acc, and RF) to 50.8% (setup I, WF1, and SVM). The best results were obtained by BG (setup I) and SVM (setup II). Globally, low performances were achieved, in particular, if compared with the machine learning results of Section 4.3. The machine learning difficulties in classifying both the anglophone misclassified users (setup I) and the GTN uncorrected responses (setup II) reinforce the competitiveness of the GTN approach.

Table 11: Machine learning error analysis results (in %, best values in **bold**).

	Classification metric							
	Acc				**WF1**			
Setup	BG	RF	SVM	MLP	BG	RF	SVM	MLP
I	**41.7**$^{\diamond}$	21.3	38.8	23.7	**50.8**$^{\diamond}$	29.7	45.5	28.8
II	40.2	31.2	**43.1**$^{\diamond}$	34.3	42.2	32.4	**44.4**$^{\diamond}$	30.0

$\diamond$ - Statistically significant under a pairwise comparison when compared with other models (p-value < 0.05).

4.3. Machine learning classification results

While the proposed GTN approach provides competitive country geolocation results (Table 5), it requires a substantial computational effort in terms of GT requests. During the experiments performed in this work, a total of 24,269 GT queries were executed: one for each distinct noun, requiring an average of 1.4 s for each GT query. Because there are 3,298 users, the average user GTN response time is 10.3 s.

To reduce the GTN request effort, we tested whether the GTN classification responses could be directly modeled as targets by the machine learning methods (the GTN2 method). The 10-fold average test results for GTN2 are shown in Table 12. The best values were achieved by the deep learning method (MLP), which outperforms other machine learning models for both classification metrics, presenting a statistical significance when compared with BG, RF, and SVM (for Acc), and BG and RF (for WF1). MLP obtained a high-quality predictive performance (Acc of 80% and WF1 of 77%). Using an Intel Xeon E5 2.30-GHz computational server, the whole MLP training (for one 10-fold iteration) required approximately 1,200 s and the MLP testing time is much faster, requiring approximately 3 ms per user. These results confirm that GTN2 is a valuable and computationally fast alternative to GTN. For future multiclass machine learning comparisons, the data used in this section has been made publicly available

at https://github.com/paolazola/Twitter-country-geolocation.

Table 12: Country geolocation results for GTN2 (in %, best values in **bold**).

Metrics	BG	RF	SVM	MLP
Acc	61.3	69.6	73.8	**80.3**$^{\diamond}$
WF1	64.2	66.2	76.2	**77.4***

$\diamond$ – Statistically significant under a pairwise comparison when compared with RF, BG, and SVM (p-value < 0.05).

$*$ – Statistically significant under a pairwise comparison when compared with RF and BG (p-value < 0.05).

4.4. Demonstration application

To further demonstrate the applicability of GTN, we assume a decision scenario in which an analyst wants to distinguish the country of interest of Twitter users that tweet about commodity prices. New data was fetched during the first week of January 2019: this comprised the last 10 days of public tweets of users that typed at least one of the keywords {"copper commodity", "sugar commodity", "cotton commodity", and "silver commodity" }. The original user sample was composed of 100 unique accounts. The Twitter profiles of these users were manually inspected, analyzing both the metadata and historical tweets, to detect the country of interest. This resulted in a set of 71 users with a clear country label. Although the sample is small, we note that a larger sample (concerning 3,298 steel production-related users) and more robust validation (10-fold) was already tested in Section 4.1. Therefore, the goal of this demonstration is just to show, as a proof of concept, the potential applicability of GTN to other non-steel commodity domains (with other users and more recent Twitter data).

The GTN method was then applied (as detailed in Section 4.1) to estimate the country for the set of 71 users. Because the number of users is relatively small, the results are shown in terms of a three-class task that includes the

two top countries of Table 2: "USA", "India", and "other". The prediction results are shown in Table 13, in terms of the confusion matrix and individual class measures (the last three rows show Acc_c, prediction_c, and recall_c). The obtained results show a very good classification performance for India (17 users, $\text{Acc}_{\text{India}}$=90.1%, $\text{precision}_{\text{India}}$=100.0%, $\text{recall}_{\text{India}}$=70.8%) and a reasonable classification for USA (39 users, Acc_{USA}=67.6%, $\text{precision}_{\text{USA}}$=53.8%, $\text{recall}_{\text{USA}}$=80.8%).

Table 13: Confusion matrix and classification measures for the GTN demonstration example.

		Target country			
		USA	India	other	Total
GTN	USA	**21**	0	5	26
predictions	India	6	**17**	1	24
	other	12	0	**9**	21
	Total	39	17	15	71
	Acc_c=	67.6%	90.1%	74.6%	
	precision_c=	53.8%	100.0%	60.0%	
	recall_c=	80.8%	70.8%	42.9%	

5. Discussion and conclusions

With the expansion of the Internet, Web and social media analytics are a key tool of diverse decision support systems. Several of these social media analytic systems require user geographic location data. In this work, we propose a novel GTN approach to detect the most probable Twitter user country of interest when such context is not explicitly known. GTN is a purely word distribution method that does not require training data. It is based on the frequency of users' tweet nouns and GT country word distribution data. The main advantage of the GTN method, with respect to existing geographic dictionary models, is its ability to obtain information from generic and adaptable nouns, dynamically provided by GT, such as "Brexit", "Trump", or "cricket". Moreover, using GT

as source, the GTN method is able to benefit from country term frequency or language differences. Conversely, the GTN has some limitations. For example, as shown in Table 7, there are popular generic nouns (e.g., "time" and "year") that show a similar frequency of use in different countries. In addition, GTN assumes just one implicit country of interest, whereas some users might travel or tweet implicitly about more than one country.

Following a design science research methodology [16], we validated GTN empirically. Using a conservative procedure, we created a recent dataset with 3,298 Twitter users from 54 countries with 744,830 tweets written in 48 languages. The obtained GTN results are of high quality (83% accuracy and weighted F1-score) and competitive when compared with a state-of-the-art word distribution method [15]. An error analysis was also performed on the GTN misclassifications, revealing different types of errors, such as mismatches between different anglophone countries (32% of the errors) and between countries that are similar or share a language or location (3%). Several experiments were conducted, using four machine learning classifiers: bagging (BG), random forest (RF), support vector machines (SVM), and a deep learning inspired multilayer perceptron (MLP). The experiments have shown that the GTN errors are difficult to outperform, confirming the value of the GTN responses. One limitation of GTN is its dependency on GT and the required GT request time. As an alternative, we tested the GTN2 approach, in which a machine learning method models the GTN responses. The best results were achieved by the GTN2 MLP model (80% accuracy and 78% weighted F1-score when modeling GTN), which is a much faster method than GTN. Finally, we have demonstrated the applicability of GTN to non-steel commodities (such as cotton), using more recent Twitter data and a different but smaller sample of users.

Because the percentage of geotagged tweets is small and Twitter user profile location data is frequently unreliable [12, 14], as also shown in this study, the proposed GTN and GTN2 approaches can be valuable to support Web and social media analytic systems. In future work, we intend to apply GTN in real-world applications, such as for filtering country tweets related to a particular

commodity price (for example, gold or wheat prices from Germany). In addition, we wish to complement GTN with extra geolocation features, such as friendship networks or user profile metadata, and investigate more fine-grained location levels. Finally, we plan to research whether feature selection filtering methods, such as pointwise mutual information [1], can be used to discard the GTN generic nouns that are used equally by different countries, thereby potentially improving the GTN performance. However, we note that such a filtering approach would require a GTN adaptation that involves a training set.

Acknowledgments

Research carried out with the support of resources of Big and Open Data Innovation Laboratory (BODaI-Lab), University of Brescia, granted by Fondazione Cariplo and Regione Lombardia. The work of P. Cortez was supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2019. We would also like to thank the anonymous reviewers for their helpful suggestions.

References

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

[2] A. Tumasjan, T. O. Sprenger, P. G. Sandner, I. M. Welpe, Predicting elections with twitter: What 140 characters reveal about political sentiment., Icwsm 10 (1) (2010) 178–185.

[3] H. Rui, Y. Liu, A. Whinston, Whose and what chatter matters? the effect of tweets on movie sales, Decision Support Systems 55 (4) (2013) 863–870.

[4] R. P. Schumaker, A. T. Jarmoszko, C. S. Labedz, Predicting wins and spread in the premier league using a sentiment analysis of twitter, Decision

Support Systems 88 (2016) 76–84. doi:10.1016/j.dss.2016.05.010.
URL https://doi.org/10.1016/j.dss.2016.05.010

[5] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, L. Brilliant, Detecting influenza epidemics using search engine query data, Nature 457 (7232) (2009) 1012.

[6] H. Choi, H. Varian, Predicting initial claims for unemployment benefits, Google Inc (2009) 1–5.

[7] H. Choi, H. Varian, Predicting the present with google trends, Economic Record 88 (s1) (2012) 2–9.

[8] Z. Fang, C. C. Chen, A novel trend surveillance system using the information from web search engines, Decision Support Systems 88 (2016) 85–97. doi:10.1016/j.dss.2016.06.001.
URL https://doi.org/10.1016/j.dss.2016.06.001

[9] D. Wu, Y. Cui, Disaster early warning and damage assessment analysis using social media data and geo-location information, Decision Support Systems 111 (2018) 48–59. doi:10.1016/j.dss.2018.04.005.
URL https://doi.org/10.1016/j.dss.2018.04.005

[10] L. Vomfell, W. K. Härdle, S. Lessmann, Improving crime count forecasts using twitter and taxi data, Decision Support Systems 113 (2018) 73–85.

[11] J. Mahmud, J. Nichols, C. Drews, Home location identification of twitter users, ACM Transactions on Intelligent Systems and Technology (TIST) 5 (3) (2014) 47.

[12] Z. Cheng, J. Caverlee, K. Lee, You are where you tweet: a content-based approach to geo-locating twitter users, in: Proceedings of the 19th ACM international conference on Information and knowledge management, ACM, 2010, pp. 759–768.

[13] F. Morstatter, J. Pfeffer, H. Liu, K. M. Carley, Is the sample good enough? comparing data from twitter's streaming API with twitter's firehose, 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, 2013.
URL http://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/6071

[14] B. Hecht, L. Hong, B. Suh, E. H. Chi, Tweets from justin bieber's heart: the dynamics of the location field in user profiles, in: Proceedings of the SIGCHI conference on human factors in computing systems, ACM, 2011, pp. 237–246.

[15] S. Lee, M. Farag, T. Kanan, E. A. Fox, Read between the lines: A machine learning approach for disambiguating the geo-location of tweets, in: Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries, ACM, 2015, pp. 273–274.

[16] David Arnott, G. Pervan, A critical analysis of decision support systems research revisited: the rise of design science, JIT 29 (4) (2014) 269–293. doi:10.1057/jit.2014.16.
URL https://doi.org/10.1057/jit.2014.16

[17] O. Buyukkokten, J. Cho, H. Garcia-Molina, L. Gravano, N. Shivakumar, Exploiting geographical location information of web pages, in: S. Cluet, T. Milo (Eds.), ACM SIGMOD Workshop on The Web and Databases, WebDB 1999, Philadelphia, Pennsylvania, USA, June 3-4, 1999. Informal Proceedings, INRIA, 1999, pp. 91–96.
URL http://www-rocq.inria.fr/%7Ecluet/WEBDB/gravano.ps

[18] O. Aulov, M. Halem, Human sensor networks for improved modeling of natural disasters, Proceedings of the IEEE 100 (10) (2012) 2812–2823.

[19] L. Backstrom, E. Sun, C. Marlow, Find me if you can: improving geographical prediction with social and spatial proximity, in: Proceedings of the 19th international conference on World wide web, ACM, 2010, pp. 61–70.

[20] C. A. Davis Jr, G. L. Pappa, D. R. R. de Oliveira, F. de L Arcanjo, Inferring the location of twitter messages based on user relationships, Transactions in GIS 15 (6) (2011) 735–751.

[21] A. Rahimi, T. Cohn, T. Baldwin, Twitter user geolocation using a unified text and network prediction model, arXiv preprint arXiv:1506.08259.

[22] N. Dalvi, R. Kumar, B. Pang, Object matching in tweets with spatial models, in: Proceedings of the fifth ACM international conference on Web search and data mining, ACM, 2012, pp. 43–52.

[23] K. Ryoo, S. Moon, Inferring twitter user locations with 10 km accuracy, in: Proceedings of the 23rd International Conference on World Wide Web, ACM, 2014, pp. 643–648.

[24] F. Laylavi, A. Rajabifard, M. Kalantari, A multi-element approach to location inference of twitter: A case for emergency response, ISPRS International Journal of Geo-Information 5 (5) (2016) 56.

[25] E. Williams, J. Gray, B. Dixon, Improving geolocation of social media posts, Pervasive and Mobile Computing 36 (2017) 68–79.

[26] D. J. Crandall, L. Backstrom, D. Huttenlocher, J. Kleinberg, Mapping the world's photos, in: Proceedings of the 18th international conference on World wide web, ACM, 2009, pp. 761–770.

[27] R. Li, S. Wang, H. Deng, R. Wang, K. C.-C. Chang, Towards social user profiling: unified and discriminative influence model for inferring home locations, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2012, pp. 1023–1031.

[28] A. S. Minot, A. Heier, D. King, O. Simek, N. Stanisha, Searching for twitter posts by location, in: Proceedings of the 2015 international conference on the theory of information retrieval, ACM, 2015, pp. 357–360.

[29] A. Rahimi, D. Vu, T. Cohn, T. Baldwin, Exploiting text and network context for geolocation of social media users, arXiv preprint arXiv:1506.04803.

[30] E. Rodrigues, R. Assunção, G. L. Pappa, D. Renno, W. Meira Jr, Exploring multiple evidence to infer users location in twitter, Neurocomputing 171 (2016) 30–38.

[31] Y. Qian, J. Tang, Z. Yang, B. Huang, W. Wei, K. M. Carley, A probabilistic framework for location inference from social media, arXiv preprint arXiv:1702.07281.

[32] S. Kinsella, V. Murdock, N. O'Hare, I'm eating a sandwich in glasgow: modeling locations with tweets, in: Proceedings of the 3rd international workshop on Search and mining user-generated contents, ACM, 2011, pp. 61–68.

[33] H.-w. Chang, D. Lee, M. Eltaher, J. Lee, @ phillies tweeting from philly? predicting twitter user locations with spatial word usage, in: Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012), IEEE Computer Society, 2012, pp. 111–118.

[34] R. Compton, D. Jurgens, D. Allen, Geotagging one hundred million twitter accounts with total variation minimization, in: Big Data (Big Data), 2014 IEEE International Conference on, IEEE, 2014, pp. 393–401.

[35] B. Han, P. Cook, T. Baldwin, Text-based twitter user geolocation prediction, Journal of Artificial Intelligence Research 49 (2014) 451–500.

[36] S. E. Middleton, L. Middleton, S. Modafferi, Real-time crisis mapping of natural disasters using social media, IEEE Intelligent Systems 29 (2) (2014) 9–17.

[37] D. Kotzias, T. Lappas, D. Gunopulos, Home is where your friends are: Utilizing the social graph to locate twitter users in a city, Information Systems 57 (2016) 77–87.

[38] J. P. Singh, Y. K. Dwivedi, N. P. Rana, A. Kumar, K. K. Kapoor, Event classification and location prediction from tweets during disasters, Annals of Operations Research (2017) 1–21.

[39] M. Avvenuti, S. Cresci, L. Nizzoli, M. Tesconi, Gsp (geo-semantic-parsing): Geoparsing and geotagging with machine learning on top of linked data, in: European Semantic Web Conference, Springer, 2018, pp. 17–32.

[40] A. Rahimi, T. Cohn, T. Baldwin, Semi-supervised user geolocation via graph convolutional networks, arXiv preprint arXiv:1804.08049.

[41] I. Witten, E. Frank, M. Hall, C. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4th Edition, Morgan Kaufmann, San Fransicco, USA, San Francisco, CA, 2017.

[42] J. Baldridge, The opennlp project, URL: http://opennlp. apache. org/index. html, (accessed 2 February 2012).

[43] D. Kahle, H. Wickham, ggmap: Spatial visualization with ggplot2., R Journal 5 (1).

[44] I. Feinerer, C. Buchta, W. Geiger, J. Rauch, P. Mair, K. Hornik, The textcat package for n-gram based text categorization in r, Journal of statistical software 52 (6) (2013) 1–17.

[45] S. Loria, P. Keen, M. Honnibal, R. Yankovsky, D. Karesh, E. Dempsey, et al., Textblob: simplified text processing, Secondary TextBlob: Simplified Text Processing.

[46] Y. Goldberg, Neural network methods for natural language processing, Synthesis Lectures on Human Language Technologies 10 (1) (2017) 1–309.

[47] A. Ng, Machine Learning Yearning, deeplearning.ai, 2018.

[48] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition, Springer-Verlag, NY, USA, 2008.

[49] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.

[50] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.

[51] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems?, The Journal of Machine Learning Research 15 (1) (2014) 3133–3181.

[52] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in: European conference on machine learning, Springer, 1998, pp. 137–142.

[53] Z. Wang, X. Xue, Multi-class support vector machine, in: Support Vector Machines Applications, Springer, 2014, pp. 23–48.

[54] Christopher M. Bishop, Pattern recognition and machine learning, 5th Edition, Information science and statistics, Springer, 2007.
URL http://www.worldcat.org/oclc/71008143

[55] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep learning, Vol. 1, MIT press Cambridge, 2016.

[56] D. Hand, Classifier technology and the illusion of progress, Statistical Science 21 (1) (2006) 1–15.

[57] Steven Walczak, N. Cerpa, Heuristic principles for the design of artificial neural networks, Information & Software Technology 41 (2) (1999) 107–117. doi:10.1016/S0950-5849(98)00116-5.
URL https://doi.org/10.1016/S0950-5849(98)00116-5

[58] G. E. Batista, R. C. Prati, M. C. Monard, A study of the behavior of several methods for balancing machine learning training data, ACM SIGKDD explorations newsletter 6 (1) (2004) 20–29.

[59] M. Hollander, D. A. Wolfe, Nonparametric statistical methods, Wiley-Interscience, 1999.