Journal of Artificial Intelligence Research 49 (2014) 451-500 Submitted 10/13; published 03/14

Text-Based Twitter User Geolocation Prediction

Bo Han HANB@STUDENT.UNIMELB.EDU.AU
The University of Melbourne, VIC 3010, Australia
NICTA Victoria Research Laboratory

Paul Cook PAULCOOK@UNIMELB.EDU.AU
The University of Melbourne, VIC 3010, Australia

Timothy Baldwin TB@LDWIN.NET
The University of Melbourne, VIC 3010, Australia
NICTA Victoria Research Laboratory

Abstract

Geographical location is vital to geospatial applications like local search and event detection. In this paper, we investigate and improve on the task of text-based geolocation prediction of Twitter users. Previous studies on this topic have typically assumed that geographical references (e.g., gazetteer terms, dialectal words) in a text are indicative of its author's location. However, these references are often buried in informal, ungrammatical, and multilingual data, and are therefore non-trivial to identify and exploit. We present an integrated geolocation prediction framework and investigate what factors impact on prediction accuracy. First, we evaluate a range of feature selection methods to obtain "location indicative words". We then evaluate the impact of non-geotagged tweets, language, and user-declared metadata on geolocation prediction. In addition, we evaluate the impact of temporal variance on model generalisation, and discuss how users differ in terms of their geolocatability.

We achieve state-of-the-art results for the text-based Twitter user geolocation task, and also provide the most extensive exploration of the task to date. Our findings provide valuable insights into the design of robust, practical text-based geolocation prediction systems.

1. Introduction

The growing volume of user-generated text posted to social media services such as Twitter, Facebook, and Tumblr can be leveraged for many purposes ranging from natural disaster response to targeted advertising (Tuten, 2008; Núnez-Redó, Díaz, Gil, González, & Huerta, 2011; Yin, Lampert, Cameron, Robinson, & Power, 2012). In many circumstances it is important to know a user's location in order to accomplish these tasks effectively. For example, disaster response managers must know where to direct resources in order to effectively coordinate aid, and advertisers could benefit from tailoring advertisements to a user's location. Similarly, search results localisation hinges on knowledge of a user's location. Although many social media services allow a user to declare their location, such metadata is known to be unstructured and ad hoc (Hecht, Hong, Suh, & Chi, 2011) (e.g., *melbo* denoting *Melbourne, AU*[1]), as well as oftentimes non-geographical (e.g., *in my own*

1. Throughout the paper, we present city names with ISO 3166-1 alpha-2 country-level designators such as *AU* = Australia and *CA* = Canada. Where US-based city names are mentioned in the context of the North American regional dataset used in experimentation (NA), we use an ISO 3166-2:US designator such as *US-CA* = California or *US-PA* = Pennsylvania.

little bubble). Text-based geolocation — automatically predicting a user's location based on the content of their messages — is therefore becoming of increasing interest (e.g., Cheng, Caverlee, & Lee, 2010, and others). In this paper we investigate and improve text-based geolocation prediction for Twitter users. Specifically, we exploit the tweets and profile information of a given user to infer their primary city-level location, which we claim is sufficiently fine-grained to support the sorts of applications mentioned above.

As is well established in previous work (e.g., Wing & Baldridge, 2011, and others), it is reasonable to assume that user posts in social media reflect their geospatial locum, because lexical priors differ from region to region. For example, a user in London is much more likely to talk about *Piccadilly* and *tube* than a user in New York or Beijing. That is not to say that those words are uniquely associated with London, of course: *tube* could certainly be mentioned by a user outside of the UK. However, the use of a range of such words with high relative frequency is strongly indicative of the fact that a user is located in London. Most work in this area utilises geotagged data as ground truth for evaluation (e.g., Eisenstein, O'Connor, Smith, & Xing, 2010, and others). The geotagged data contains GPS coordinates inserted with the user's consent by a GPS-enabled device such as a smartphone, and offers accurate information about a user's position at the time of tweeting. Although approaches to text-based geolocation are offering increasingly promising results, the studies to date on this topic have been limited in a number of important ways. We raise some key issues in Section 3 and investigate them in turn, focusing on the following issues:

1.1 Location Indicative Words

Text-based geolocation prediction models for social media are predominantly based on the full text data of tweets, including common words with no geospatial dimension (e.g., *today*), potentially hampering prediction, and because of the large number of words observed in tweets, leading to slower, more memory-intensive models. We tackle this by automatically finding location indicative words (LIWs) via feature selection, and demonstrating that the reduced feature set boosts geolocation accuracy. In Section 5, we carry out extensive evaluation over a wide range of feature selection methods proposed in the literature, and show that an information gain ratio-based approach outperforms benchmark geolocation prediction methods by 10.6 percentage points in terms of accuracy, and reduces the median prediction error distance by 209km on a publicly-available regional (North America) dataset. We similarly demonstrate the effectiveness of LIW selection on a global dataset in Section 6.

1.2 Non-geotagged Tweets

In addition to experimenting with geotagged data, we further extend our analysis to incorporate non-geotagged tweets. Some recent work (e.g., Roller, Speriosu, Rallapalli, Wing, & Baldridge, 2012) has incorporated non-geotagged training data, although little work has analysed the contribution of non-geotagged data, i.e., the extent to which incorporating non-geotagged data improves geolocation accuracy. Furthermore, the evaluation of previous models has been restricted to geotagged data (in order to have access to a ground truth) although the goal of this line of research is to be able to infer locations for users whose locations are not known. However, it is unclear how well models evaluated only on geotagged data will generalise to non-geotagged data. For example, because geotagged tweets are sent from GPS-enabled devices such as smartphones, while non-geotagged tweets are sent from a range of devices (including desktop computers), these two types of data could have different characteristics (Gouws, Metzler, Cai, & Hovy, 2011).

In Section 7, we address these issues by training and testing on geotagged tweets, non-geotagged tweets, and the combination of the two. We show that by exploiting a user's non-geotagged tweets, the city-level accuracy is improved from 12.6% to 28.0% on a benchmark dataset, underlining the potential contribution of non-geotagged data. Furthermore, the numbers also suggest that a model trained on geotagged data indeed generalises to non-geotagged data, although sub-domain differences between geotagged data and non-geotagged data are observed.

1.3 Language Influence

With some exceptions (e.g., Kinsella, Murdock, & O'Hare, 2011), most text-based geolocation studies have been carried out in an English-only setting, or a primarily English setting. Because high-accuracy language identification tools (Lui & Baldwin, 2012; Nakatani, 2010) are now readily available, this is not a problem: messages in the target language can be identified, and text-based geolocation methods can be applied to only those messages. However, it remains to be seen whether text-based geolocation approaches that have been shown to work well for English perform as well on other languages, or perform well in a multilingual setting. English is tweeted throughout the world, whereas languages such as Indonesian are primarily tweeted in localised areas. As such, the performance of methods developed and tested over English data could be very different when applied to other languages. We investigate the language influence on a multilingual dataset in Section 8. The results suggest that our model indeed generalises from a monolingual English to a multilingual setting. Furthermore, the experiments reveal that geolocation prediction is much easier for languages with more geographically-restricted use (e.g., Indonesian) than languages that are more diverse in usage (e.g., English). We then go on to show that a composite model consisting of a number of monolingual geolocation models based on language identification outperforms a model trained on multilingual data.

1.4 Metadata and Ensemble Learning

Although tweet-based geolocation is worthy of study in its own right, tweets are accompanied by rich metadata in public user profiles. This metadata is included in the payload of JSON objects containing tweets, and offers complementary information that may be exploited to improve accuracy, e.g., timezone data and the user-declared location. While there has been some work on utilising timezone (Mahmud, Nichols, & Drews, 2012) and user-declared location (Hecht et al., 2011) information for user geolocation, the metadata remains largely untouched in the literature. In Section 9, we investigate the performance of metadata-based geolocation models and compare them with benchmark methods. We show that by incorporating information from metadata and the tweet message in a stacking-based approach, a city-level accuracy of 49.1%, and a median prediction error distance of just 9km, can be achieved over our global dataset, which is a substantial improvement over any of the base classifiers.

1.5 Temporal Influence

Because Twitter is a growing and evolving medium, the data in Twitter streams tends to be locally temporal to the time of posting. In addition to evaluating the geolocation model on "old" time-homogeneous data (sampled from the same time period as the training data), in Section 10 we evaluate the trained model on a "new" time-heterogeneous dataset, which was collected approximately one year after the training and test data used in our earlier experiments. The observed moderate decline in results indicates that the stacked geolocation model is indeed influenced by temporal changes. Error analysis reveals that this is primarily caused by the unreliability of the base model trained on user-declared locations. In contrast, we find that models trained on tweet text and timezone information are relatively insensitive to temporal changes. This finding on the one hand justifies the efforts to-date in pursuing better text-based geolocation prediction, and on the other hand suggests that if user-declared location data is to be used, the model has to be periodically updated to remain current to temporal changes.

1.6 User Geolocatability and Prediction Confidence

We further discuss the geolocatability of users with regard to tweeting behaviour in Section 11. For instance, does mentioning many local place names have a strong influence on the prediction accuracy? Experiments suggest the number of LIWs (in particular, gazetted location names) and user-declared metadata are key to geolocating a user. Because of different tweeting behaviours among users, not all users are equally geolocatable, with only predictions for a proportion of them being reliable. We further conduct a pilot study on approximating the prediction confidence through a range of variables in Section 12.

This paper advances the state-of-the-art of text-based geolocation prediction in a number of directions, and provides practical guidelines for the design of a text-based geolocation application. This paper builds off our own previously-published work (Han, Cook, & Baldwin, 2012b, 2013) with much more extensive evaluation, and new work in the following areas:

- A large-scale comparative evaluation of twelve feature selection methods for user geolocation — nine of which were not considered in our earlier work — in Sections 4–6.
- The analysis of the impact of training on non-geotagged data in Section 7.
- A new set of experiments, and subsequent analysis, examining the influence of language in Section 8.
- Further analysis of the utility of user-supplied metadata and ensemble learning in Section 9.
- More-detailed analysis of model generalisation on temporal change in Section 10 including city-level meta-analysis.
- A new pilot study on user geolocatablility and privacy in Section 11.

The proposed text-based method primarily uses words for geolocation prediction, and intentionally excludes Twitter specific entities, such as hashtags and user mentions. The prediction accuracy therefore largely depends on whether the text contains sufficient geospatial information for geolocation prediction. Therefore, although this paper focuses exclusively on Twitter, the proposed method could equally be applied to other forms of social media text, such as Facebook status updates or user-submitted comments (to services such as YouTube).

2. Related Work

While acknowledging potential privacy concerns (Mao, Shuai, & Kapadia, 2011; Pontes, Vasconcelos, Almeida, Kumaraguru, & Almeida, 2012), accurate geolocation prediction is a key driver for location-specific services such as localised search, and has been the target of research across different disciplines. For example, the tagging of both user queries (Wang, Wang, Xie, Forman, Lu, Ma, & Li, 2005; Backstrom, Kleinberg, Kumar, & Novak, 2008; Yi, Raghavan, & Leggetter, 2009) and web pages (Ding, Gravano, & Shivakumar, 2000; Amitay, Har'El, Sivan, & Soffer, 2004; Zong, Wu, Sun, Lim, & Goh, 2005; Silva, Martins, Chaves, Afonso, & Cardoso, 2006; Bennett, Radlinski, White, & Yilmaz, 2011) has been considered in information retrieval. In geographical information science, the primary focus has been on recognising location mentions in text (Leidner & Lieberman, 2011), with named entity recognition tools typically employed to detect and extract such mentions (Quercini, Samet, Sankaranarayanan, & Lieberman, 2010; Gelernter & Mushegian, 2011). Within the social media realm, geolocation methods have been applied to images on Flickr (Crandall, Backstrom, Huttenlocher, & Kleinberg, 2009; Serdyukov, Murdock, & van Zwol, 2009; Hauff & Houben, 2012; O'Hare & Murdock, 2013; Laere, Schockaert, & Dhoedt, 2013), Wikipedia articles (Lieberman & Lin, 2009), individual tweets (Kinsella et al., 2011), Twitter users (Eisenstein et al., 2010; Cheng et al., 2010; Kinsella et al., 2011; Wing & Baldridge, 2011; Roller et al., 2012; Han et al., 2012b), and for identifying words and topics on Twitter that are salient in particular regions (Eisenstein et al., 2010; Yin, Cao, Han, Zhai, & Huang, 2011; Hong, Ahmed, Gurumurthy, Smola, & Tsioutsiouliklis, 2012; Dalvi, Kumar, & Pang, 2012).

Identifying Twitter users' locations is non-trivial, mainly due to the unavailability of reliable geographic information. Although Twitter allows users to declare their location in their user profile, the location descriptions are unstructured and ad hoc (Cheng et al., 2010; Hecht et al., 2011), e.g., people use vernacular expressions such as *philly*, or non-standard spellings such as *Filladephia*, to refer to *Philadelphia*; non-geographical descriptions like *in your heart* are also commonly found. Without appropriate processing, the value of these location fields is greatly limited. Hecht et al. (2011) demonstrate that trivially using these location fields in off-the-shelf geolocation tools is ineffective. Alternatively, some tweets sent from mobile devices are geotagged with accurate GPS coordinates, however, the proportion of geotagged tweets is estimated to be a mere 1% (Cheng et al., 2010), and the location of the vast majority of users are not geotagged. Methods based on IP addresses (Buyukokkten, Cho, Garcia-Molina, Gravano, & Shivakumar, 1999) can be applied to the task, and in general web contexts have been shown to achieve around 90% accuracy at mapping Internet hosts to their locations (Padmanabhan & Subramanian, 2001). Such methods are not applicable to Twitter and many other social media services, however, as the IP address of the device the message was sent from cannot be accessed via any of the public APIs. Doubtless Twitter itself has access to this information and can use it for user geolocation, although even here, geographical divisions of IP addresses are not always credible. For instance, departments in an international corporation might use the same IP address range, but their true locations could be spread across the world. VPNs are also a complication for such approaches. Any third-party service provider making use of Twitter data, however, has to look to other sources of geospatially-identifying information, including the text content of the user's posts and metadata information, as targeted in this research.

In the spatial data mining community, geographical references (e.g., gazetteer terms) in text have also been exploited to infer geolocation. Intuitively, if a place is frequently mentioned by a user in their tweets, they are likely tweeting from that region. Methods building on this intuition range from naive gazetteer matching and rule-based approaches (Bilhaut, Charnois, Enjalbert, & Mathet, 2003), to machine learning-based methods (primarily based on named entity recognition: Quercini et al., 2010; Gelernter & Mushegian, 2011). Despite the encouraging results of this approach on longer and more homogeneous documents sets (Quercini et al., 2010), its performance is impeded by the nature of tweets: they are short and informal, and the chances of a user not mentioning gazetted places in their tweets is high. Moreover, the handling of vernacular place names, e.g., *melbo* for *Melbourne*, in this approach is limited. The reliance on named entity recognition is thwarted by the unedited nature of social media data, where spelling and capitalisation are much more ad hoc than in edited document collections (Ritter, Clark, Mausam, & Etzioni, 2011; Han, Cook, & Baldwin, 2012a).

Moving beyond off-the-shelf solutions, recently, many robust machine learning methods have been applied to geolocation, with the primary approach being to estimate locations based on the textual content of tweets. For instance, Cheng et al. (2010) exploit words known to be primarily used in particular regions, along with smoothing techniques, to improve a simple generative geolocation model when applied to data from the continental United States. Wing and Baldridge (2011) divide the world's surface into a uniform-size grid, and compare the distribution of words in a given user's tweets to those in each grid cell using Kullback-Leibler (KL) divergence to identify that user's most likely location. One limitation of this approach is that grid cells in rural areas tend to contain very few tweets, while there are many tweets from more urban grid cells. Roller et al. (2012) therefore extend this method to use an adaptive grid representation in which cells contain approximately the same amount of data, based on a k-d tree (Bentley, 1975). Kinsella et al. (2011) examine geolocation prediction at different granularities (e.g., zip codes, city, state and country). Chang, Lee, M., and Lee (2012) prune noisy data based on geometrically-local words (i.e., words that occur geographically close to each other, and are only found in a limited number of cities) and non-stop words that are dis-similar to stop words, and they experiment with the reduced feature set using both a Gaussian mixture model and Maximum Likelihood Estimation for location prediction. Beyond purely text-based methods (language model-based methods), other sources of information have also been integrated. Li, Serdyukov, de Vries, Eickhoff, and Larson (2011) investigate geolocation prediction based on a linear rank combination of text and temporal factors. Mahmud et al. (2012) combine timezone information and content-based classifiers in a hierarchical model for geolocation. In particular, nouns, hashtags, and place names are considered as content in the method. Schulz, Hadjakos, Paulheim, Nachtwey, and Mühlhäuser (2013) combine scores from various geographical sources (e.g., tweet text, user profile data). The sum of scores for a location is represented by the "aggregated height" on a polygon-partitioned map, and the highest polygon is the predicted location.

Topics discussed on Twitter vary across geographical regions. Intuitively, for instance, Americans are more likely to talk about *NBA* and *baseball* than Australians (who probably mention *AFL* and *rugby* more often). To capture these regional topic variations in Twitter, topic modelling-based approaches have also been used to incorporate geographical regions in the generative process. For instance, Eisenstein et al. (2010) introduce a geographical variable (r); instead of generating an observed word w from a per-word topic distribution ϕ_z as in the standard Latent Dirichlet Allocation (LDA) model (Blei, Ng, & Jordan, 2003), their proposed approach refines this step by additionally modeling the topic distributions across different geographical regions, i.e., w is generated from a per-word region-topic distribution ϕ_{rz}. Therefore, the observed user locations are generated from geographical regions and the region variable in topic modeling is linked with user geographical locations. Generally, a user's location is predicted at the regional level by adopting the location centroid for geotagged tweets from that region. Hong et al. (2012) further improve the approach by considering more fine-grained factors in an additive generative model. In addition to introducing per-region topic variance, they incorporate per-user topic variance, a regional language model, and global background topics. To compensate for the computational complexity associated with these extra hidden variables, they adopt sparse modeling in inference. On top of these geolocation prediction tasks, many other research problems also involve the modelling of geographical locations. Dalvi et al. (2012) exploit the impact of geographical locations on users' discussions of pre-defined objects (e.g., restaurants) in tweets. Yin et al. (2011) propose ways to discover and compare topics for geographical regions by jointly modelling locations and text. Despite the benefits of incorporating per-region topic variance in these models, a few concerns prevent us from using topic modeling approaches in this study. First, the temporal currency of geographical topics can be limited, e.g., *Olympics* or *playoffs*. These temporally-specific topics are less indicative of location for future inference, e.g., geolocating users after the model has been trained. Furthermore, topic modelling is generally computationally expensive, and suffers efficiency problems when applied to large volumes of data, such as that available through social media. Therefore we experiment with language model-based methods that are better suited to large-scale data.

Social network information, including both explicit friendship relations (Backstrom, Sun, & Marlow, 2010; Sadilek, Kautz, & Bigham, 2012; Rout, Bontcheva, Preotiuc-Pietro, & Cohn, 2013) and implicit social interactions (Chandra, Khan, & Muhaya, 2011; Jurgens, 2013), has been shown to be effective in predicting locations. City-level prediction results range from approximately 50–80% (Rout et al., 2013) depending on a wide range of factors including the user density in the social network and the precise scope of the geolocation prediction task. However, social networks are dynamic, and this information is often more difficult to obtain than text data on a large scale. For instance, obtaining social network information requires multiple requests to the rate-limited Twitter API to reconstruct the full social graph. We therefore only focus on approaches based on text, and metadata that accompanies each individual tweet, and leave the possibility of integrating social network information to future work.

3. Key Questions and Geolocation Prediction Framework

Though various geolocation prediction approaches have been proposed and adapted for social media data, some fundamental questions remain. In the rest of the paper, we address each of the these questions in turn.

- Given that text-based methods rely on salient words local to particular regions to disambiguate geolocations, do "location indicative words" improve the accuracy over using the full word set?

- Does a model trained on geotagged data generalise to non-geotagged data? What is the impact of adding non-geotagged texts to the training and test data? Is there an inherent sub-domain difference between geotagged and non-geotagged tweets given that geotagged tweets are primarily sent from mobile devices?

- Does geolocation prediction accuracy vary by language? For example, is a user who primarily tweets in Japanese more geolocatable than a user who tweets mostly in English? If language does influence accuracy, how can we exploit this to improve multilingual geolocation prediction?

- Does the user-declared text metadata provide geographical information complementary to that in the tweets themselves? How can we make use of these multiple sources of textual data to produce a more accurate geolocation predictor?

- As Twitter is rapidly growing and evolving, how do temporal factors influence the model generalisation? Will a model trained on "old" data perform comparably on "new" test data?

- From the perspective of privacy protection, how does a user's tweeting behaviour affect their geolocatability, i.e., the ability of the model to predict their location? Are there steps a user can take to reduce the risk of inadvertently leaking geographical information while sharing tweets with the public?

- Can measures of prediction confidence be formulated to estimate the accuracy of the geolocation prediction?

In this paper, we focus on predicting Twitter users' primary (referred to as their "home") geolocation, and following Cheng et al. (2010) and others, assume that a given user will be based in a single city-based location throughout the time period of study. We approach geolocation prediction as a text classification task. Tweets from each city are taken to represent a class. All tweets from a given user are aggregated and assigned to that user's primary location. We characterise geolocation prediction by four key components, which we discuss in turn below: (1) the representation of different geolocations, (2) the model, (3) the feature set, and (4) the data.

3.1 Representation: Earth Grid vs. City

Geolocations can be captured as points, or clustered based on a grid (Wing & Baldridge, 2011; Roller et al., 2012), city centres (Cheng et al., 2010; Kinsella et al., 2011) or topic regions (Eisenstein et al., 2010; Hong et al., 2012). A point-based representation presents computational challenges, and is too fine-grained for standard classification methods. As for dynamic location partitioning, the granularity of regions is hard to control and will potentially vary across time, and the number of regions is a variable which will depend on the dataset and potentially also vary across time. Fixed grid-based representations are hindered because there is considerable variability in the shape and size of geographical regions: a coarse-grained grid cell is perhaps appropriate in central Siberia, but for densely-populated and linguistically/culturally diverse regions such as Luxembourg, doesn't lead to a natural representation of the administrative, population-based or language boundaries in the region. We therefore opt for a city-based representation, which is able to capture these boundaries more intuitively. The downside to this representation is that it is inappropriate for classifying users in rural areas. As we will see in Figure 1, however, the bulk of Twitter users are, unsurprisingly, based in cities.

Following Han et al. (2012b), we use the publicly-available `Geonames` dataset as the basis for our city-level classes.[2] This dataset contains city-level metadata, including the full city name, population, latitude and longitude. Each city is associated with hierarchical regional information, such as the state and country it is based in, so that London, GB, e.g., is distinguished from London, CA.

2. `http://www.geonames.org`, accessed on October 25th, 2012.

We hence use a city-region-country format to represent each city (e.g., Toronto, CA is represented as toronto-08-ca, where 08 signifies the province of Ontario and ca signifies Canada).[3] Because region coding schemes vary across countries, we only employ the first- and second-level region fields in `Geonames` as the region. Furthermore, if the second-level field is too specific (i.e., longer than 4 letters in our setting), we only incorporate the first-level region field (e.g., instead of using melbourne-07-24600-au, we use melbourne-07-au). Moreover, because cities are sometimes complex in structure (e.g., Boston, US colloquially refers to the metropolitan area rather than the city, which is made up of cities including Boston, Revere and Chelsea), we collapse together cities which are adjacent to one another within a single administrative region, as follows:

1. Identify all cities which share the same region code (i.e., are located in the same state, province, county, etc.) in the `Geonames` dataset.
2. For each region, find the city c with the highest population.
3. Collapse all cities within 50km of c into c.[4]
4. Select the next-largest city c, and repeat.
5. Remove all cities with a population of less than 100K. The remaining cities form our city-based representation of geolocations.

As a result of this methodology, Boston, US ends up as a single city (incorporating Revere and Chelsea), but neighbouring Manchester, US is a discrete city (incorporating Bedford) because it is in New Hampshire. This algorithm identifies a total of 3,709 collapsed cities throughout the world.

3.2 Geolocation Prediction Models

Various machine learning algorithms can be applied to the task of multi-class text categorisation. However, many state-of-the-art learning algorithms are not appropriate for this particular task for reasons of scalability. For example, support vector machines (Vapnik, 1995) are not well suited to massively multi-class problems (i.e., 3,709 cities in our case). Finally, we would ideally like to have a learning algorithm which can be easily retrained, e.g., to incorporate new training data from the Twitter data stream. As such, we primarily experiment with simple learning algorithms and ensemble learning for geolocation prediction.

3.2.1 GENERATIVE VS. DISCRIMINATIVE MODELS

Generative models (e.g., naive Bayes) are based on estimation of joint probability of observing a word vector and a class (i.e., $P(w_1, w_2, \ldots, w_n, c_i)$, where $w_1, w_2, \ldots$ are words and $c_i \in C$ is a city from a combined set of cities C). In contrast, discriminative models are based on estimation of a class given a word vector (i.e., $P(c|w_1, w_2, \ldots, w_n)$). The objective of both models is to find a

3. Country code information can be found in `http://download.geonames.org/export/dump/countryInfo.txt`
4. We use the great-circle distance (Vincenty, 1975) for all distance calculations in our experiments, as opposed to Euclidean distance, to properly capture the three-dimensional surface of the earth. The proximity of cities varies across the world, e.g., cities on the east coast of the United States are much closer to each other than major cities in Australia. There is therefore scope to explore the impact of this 50km setting on the city label set, which we leave to future work.

city $c_{max} \in C$ such that the relevant probability is maximised. In our experiments, we experiment with both models. For instance, we choose a state-of-the-art discriminative geolocation model based on KL divergence over k-d tree partitioned unigrams (KL) (Roller et al., 2012). We also adopt a generative multinomial naive Bayes (NB) model (Hecht et al., 2011) as our default benchmark, for two reasons: (1) it incorporates a class prior, allowing it to classify an instance in the absence of any features shared with the training data; and (2) generative models outperform discriminative models when training data is relatively scarce (Ng & Jordan, 2002).[5]

3.2.2 SINGLE VS. ENSEMBLE MODELS

In addition to single model comparisons (e.g., discriminative KL vs. generative NB in Sections 5 and 6), we further combine multiple base classifiers — e.g., heterogeneous NB models trained on each of Twitter text and user metadata — to improve the accuracy. First, we investigate the accuracies of base classifiers and correlations between them. Then, we apply different ensemble learning strategies in Section 9.

3.3 Feature Set

Predominantly, geolocations are inferred based on geographical references in the text, e.g., place names, local topics or dialectal words. However, these references are often buried in noisy tweet text, in which lexical variants (e.g., *tmrw* for "tomorrow") and common words without any geospatial dimension (e.g., *weather*, *twitter*) are prevalent. These noisy words have the potential to mislead the model and also slow down the processing speed. To tackle this issue, we perform feature selection to identify "location indicative words". Rather than engineering new features or attempting to capture named entities (e.g., *the White House*) or higher-order n-grams, we focus on feature selection over simple word unigrams (see Section 4). This is partly a pragmatic consideration, in that unigram tokenisation is simpler.[6] Partly, however, it is for comparability with past work, in determining whether a strategically-selected subset of words can lead to significant gains in prediction accuracy (see Sections 5 and 6).

In addition to feature selection, the feature set can be further refined and extended in various ways. For instance, feature selection can be enhanced by incorporating non-geotagged tweet data. Furthermore, languages can be used to shape the feature set, as words from different languages carry varying amounts of geospatial information, e.g., because Dutch is primarily used only in the Netherlands, Dutch words are usually more location indicative than English words. Moreover, user-provided metadata (e.g., location and timezone) is readily accessible in the tweet JSON objects. This metadata can be appended as extra text features, in addition to features derived from tweet text. We investigate the impact of these factors in later sections.

5. There is certainly an abundance of Twitter data to train models over, but the number of Twitter users with sufficient amounts of geotagged tweets to be able to perform geolocation prediction is small, relative to the number of parameters in the model (the product of the number of features and classes).
6. Also, preliminary results with both named entities and higher order n-grams were disappointing.

Filtering criterion	Proportion of tweets (relative to preceding step)
Geotagged	0.008
Near a city	0.921
Non-duplicate and non-Foursquare	0.888
English	0.513

Table 1: Proportion of tweets remaining after filtering the data based on a series of cascaded criteria. These numbers are based on a Twitter corpus collected over two months.

3.4 Data

Geolocation prediction models have primarily been trained and tested on geotagged data.[7] We use both regional datasets (i.e., geotagged tweets collected from the continental US: Eisenstein et al., 2010; Mahmud et al., 2012) and global datasets (Kinsella et al., 2011; Han et al., 2012b) in this research. Because of accessibility issues (e.g., many tweets in older datasets have been deleted and are thus not accessible now) and data sparseness (e.g., there were only 10K users in the study of Eisenstein et al., 2010), we are only able to experiment over a small number of public datasets. In this paper, we employ three geotagged datasets:

1. A regional North American geolocation dataset from Roller et al. (2012) (NA hereafter), for benchmarking purposes. NA contains 500K users (38M tweets) from a total of 378 of our pre-defined cities. NA is used as-is to ensure comparability with previous work in Section 5.

2. A dataset with global coverage constructed by us in earlier work (Han et al., 2012b) (WORLD hereafter), collected via the Twitter public Streaming API[8] from 21 Sep, 2011 to 29 Feb, 2012. The tweet collection is further shaped for different evaluation tasks, e.g., geotagged English data WORLD in Section 6, incorporating non-geotagged English data WORLD+NG in Section 7, multilingual geotagged data WORLD+ML in Section 8 and with rich metadata WORLD+META in Section 9.

3. A second dataset with global coverage novel to this research (LIVE), which contains tweets collected more than 1 year after WORLD (from 3 Mar, 2013 to 3 May, 2013), to analyse the influence of temporal recency on geolocation prediction. Unlike the other two datasets, LIVE is used only as a test dataset, in Section 10.

WORLD was restricted to English tweets in order to create a dataset similar to NA (in which English is the predominant language), but covering the entire world. It was pre-processed by filtering the data as follows. First, all non-geotagged tweets were removed. Next, we eliminated all tweets that aren't close to a city by dividing the earth into $0.5^\circ \times 0.5^\circ$ grid cells, and discarding any tweet for which no city in our `Geonames` class set is found in any of the 8 neighbouring grid cells. We then assign each user to the single city in which the majority of their tweets occur. We

7. One exception to this is Cheng et al. (2010), who train on users whose user-declared metadata location fields correspond to canonical locations (e.g., Boston, MA), and test on users whose locations are indicated with GPS coordinates in their metadata.

8. `https://dev.twitter.com/docs/streaming-apis`

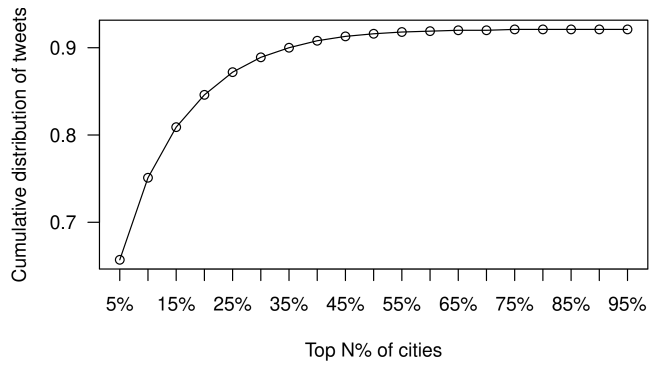

Figure 1: Cumulative coverage of tweets for increasing numbers of cities based on 26 million geo-tagged tweets.

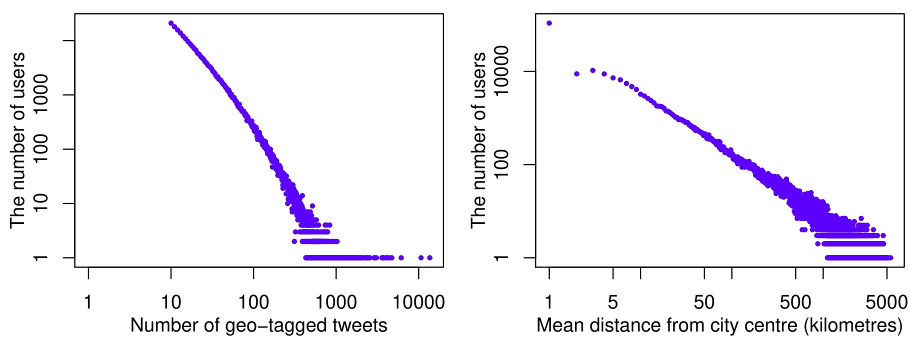

Figure 2: The number of users with different numbers of tweets, and different mean distances from the city center, for WORLD.

further remove cities with fewer than 50 feature types (i.e., word types) to reduce data sparsity. This results in 3135 cities in WORLD (as opposed to 3709 cities in the full `Geonames` class set). We eliminated exact duplicate tweets and Foursquare check-ins (which encode the user location in the form of *I'm at* ...). After that, non-English tweets were further removed using `langid.py`, an open-source language identification tool (Lui & Baldwin, 2012). This filtering is summarised in Table 1 which also shows the proportion of tweets remaining after each step. The total number of users and tweets in WORLD is 1.4M and 12M, respectively. Similar to NA, the development and test datasets both contain 10K users, and the remainder of the users are used in training. The development and test data was sampled such that each user has at least 10 geotagged tweets to alleviate data sparsity.[9] We tokenised the tweets with a Twitter-specific tokeniser (adapted from O'Connor, Krieger, & Ahn, 2010).

Although there are certainly instances of social media users with high mobility (Li, Wang, & Chang, 2012), recent studies have shown that most users tend to tweet from within a limited region (Cho, Myers, & Leskovec, 2011; Hecht et al., 2011). We also analyse the spread of WORLD in

9. This restriction was not applied to the training data.

Figure 2, in terms of: (1) the number of users with at least 10 geotagged tweets; and (2) the number of users with differing levels of geographical spread in their tweets, measured as the average distance between each of a user's tweets and the centre of the city to which that user is allocated.[10] This preliminary analysis shows that most users have a relatively small number of geotagged tweets, and most users stay near a single city (e.g., 83% users have a geographical spread of 50 kilometres or less). The high proportion of users with an average distance of 1km to the city centre is an artefact of their geotagged tweets being mapped to a city centre before performing this analysis. In order to investigate the coverage of the proposed city-based partition, we examine the recall in our original sample of 26 million geotagged tweets (prior to filtering, as described above). The analysis reveals that 92.1% of tweets are "close" to (in a neighbouring $0.5^\circ \times 0.5^\circ$ grid cell) to one of our pre-defined cities, and that the top 40% of cities contain 90% of the geotagged tweets after filtering, as shown in Figure 1. This supports our assumption that most (geotagged) Twitter users are based in cities.

3.5 Evaluation Measures

Having formulated the geolocation prediction task into a discrete class space through the use of our city class set, it is possible to use simple classification accuracy to evaluate our models. However, given that all of our class labels have a location (in the form of latitude–longitude coordinates), we can also sensitise the evaluation to distance-based predictive error. For instance, if the correct location for a user is Seattle, US, a prediction of Portland, US is arguably better than a prediction of Los Angeles, US, on the basis of geospatial proximity. We use a number of evaluation measures which capture spatial proximity, in line with previous work (Wing & Baldridge, 2011; Roller et al., 2012):[11]

1. **Acc**: city-level accuracy, i.e., the proportion of predictions that correspond to the correct city;

2. **Acc@161**: the proportion of predictions that are within a distance of 161 kilometres (100 miles) from the correct city-level location. This empirical measure (Cheng et al., 2010) is a relaxed version of Acc, capturing near-miss predictions.

3. **Acc@C**: country-level accuracy, i.e., the proportion of predicted locations that are in the same country as their corresponding true locations. This measure is useful for applications relying on country-specific Twitter data, e.g., sentiment analysis in specific countries.

4. **Median**: median prediction error, measured in kilometres between the predicted city centres and the true geolocations. We prefer to use the median, as opposed to mean, distance because the median is less sensitive to wildly incorrect predictions — e.g., a user from London, GB classified as being based in Sydney, AU. In contrast, the mean distance can increase substantially due to a small number of extreme misclassifications, although this effect is limited for inherently-bounded regional datasets such as NA.

10. The geographical spread is calculated over a random sub-sample of 10 tweets for a given user, for efficiency reasons.
11. In very recent work, Priedhorsky, Culotta, and Valle (2014) additionally proposed a set of probabilistic metrics to evaluate tweet-based geolocation prediction, including using the expected distance between a tweet's true point location to a random point location drawn from the probability distribution of the geolocation model. While we strongly support this new direction for geolocation modelling and evaluation, depending on the application context, we argue that point- or region-based representations and related discrete evaluation measures are equally important in user geolocation research.

4. Finding Location Indicative Words

Precise user locations for individual messages are embedded in geotagged tweets in the form of latitude–longitude coordinates. By mapping these coordinates to cities and representing each tweet as a bag of words, we are able to make connections between words (i.e., features) and cities (i.e., classes). In this section, we present a range of methods for ranking these words by their location indicativeness, i.e., the degree to which a word is associated with particular cities. Words that either explicitly (e.g., place names) or implicitly (e.g., dialectal words, slang or local references) encode geographical information are collectively referred to as "location indicative words" (LIWs); it is these words that we aim to automatically identify. Examples of LIWs are:

1. local words (1-local) that are used primarily in a single city, namely *yinz* (used in Pittsburgh to refer to the second-person plural pronoun), *dippy* (used in Pittsburgh to refer to a style of fried egg, or something that can be dipped in coffee) and *hoagie* (used primarily in Philadelphia, to refer to a kind of sandwich);[12]

2. semi-local words (n-local) that refer to some feature of a relatively limited subset of cities, namely *ferry* (found, e.g., in Seattle, New York and Sydney), *Chinatown* (common in many of the largest cities in the US, Canada and Australia, but much less common in European and Asian cities), and *tram* (found, e.g., in Vienna, Melbourne and Prague)

In addition to LIWs there are common words (common) which aren't expected to have substantial regional frequency variation, namely *twitter*, *iphone* and *today*.

In the remainder of this section, we present various feature selection methods for identifying LIWs, drawn from the work of Han et al. (2012b), Chang et al. (2012) and Laere et al. (2013). The feature selection methods can be broadly categorised into three types: (1) statistical; (2) information-theoretic; and (3) heuristic. To reduce low-utility words and noise, for all feature selection methods, we remove all words which include non-alphabetic letters, are less than 3 letters long, or have a word frequency < 10.

4.1 Statistical-Based Methods

Statistical hypothesis testing is often used to determine whether an event occurs by chance (i.e., the null hypothesis) or not (i.e., the alternative hypothesis) at a particular confidence level (e.g., 95% $\equiv p < 0.05$). In our case, an event is defined to be a co-occurrence between a word and a city, and the null hypothesis assumes the co-occurrence is by chance, i.e., the word and city are independent. The goal of feature selection is then to find word–city pairs where the null hypothesis is rejected.

4.1.1 χ^2 AND LOG-LIKELIHOOD

The χ^2 statistic is commonly used to examine the degree of independence between random variables. A contingency table representing the observations of the variables is formed, as in Table 2. The general form of the statistic is:

$$\sum_{i}^{n} \frac{(O_i - E_i)^2}{E_i}$$

12. These words were identified with the aid of datasets of regional words such as DARE: `http://dare.wisc.edu/`.

	in c	not in c
w	$O_{w,c}$	$O_{w,\bar{c}}$
non-w word	$O_{\bar{w},c}$	$O_{\bar{w},\bar{c}}$

Table 2: Contingency table for word and city co-occurrence

where O_i represents an observation (i.e., co-occurrence of a city (c) and word (w)), and n is the number of cells in the table. $O_{w,c}$ and $O_{\bar{w},\bar{c}}$ denote the occurrence of word *w* in city *c* and non-*w* words in cities other than *c*, respectively. $E_{w,c}$ denotes the expected frequency of *w* in *c*, calculated from the marginal probabilities and total counts N:

$$\begin{aligned} E_{w,c} &= P(w) \times P(c) \times N = \frac{O_{w,c} + O_{w,\bar{c}}}{N} \times \frac{O_{w,c} + O_{\bar{w},c}}{N} \times N \\ N &= O_{w,c} + O_{\bar{w},c} + O_{w,\bar{c}} + O_{\bar{w},\bar{c}} \end{aligned}$$

If the χ^2 statistic is larger than the number in the χ^2 distribution, with respect to the degrees of freedom (in this case, 1), then the null hypothesis that city *c* and word *w* are independent is rejected. As with many statistical tests, χ^2 can be ineffective when counts are low. We address this through our word frequency thresholding and use of massive amounts of training data.

Conventionally, χ^2 is used to identify the set of features which satisfies a pre-defined confidence level (e.g., $p < 0.05$). However, in the case of LIW selection, we instead use the χ^2 statistic to rank all word–city pairs. The selection of LIWs is deferred to the parameter tuning state, in which the boundary between LIWs and common words is optimised using development data.

At this point, a different ranking of LIWs is produced per city, where what we desire is a global ranking of LIWs capturing their ability to discriminate between cities in the combined label set. There are various ways to do this aggregation. As suggested by Laere et al. (2013), one approach to selecting n features based on χ^2 is to iteratively aggregate the top-m features from each city until n features are obtained. Alternatively, they can be ranked based on the highest-scoring occurrence of a given word for any city, by first sorting all city–word χ^2 test pairs, then selecting the first occurrence of a word type for the aggregated ranking. These two aggregation approaches produce different feature selection rankings, and are distinguished using Chi and $MaxChi$, respectively.[13]

Similar to the χ^2 test, the log-likelihood ratio ("$Loglike$": Dunning, 1993) has also been applied to LIW selection (Laere et al., 2013). The $Loglike$ test determines whether h_0 (the null hypothesis, i.e., the word is independent of the city) is more likely than h_1 (the alternative hypothesis, i.e., the word is dependent on the city). Following Dunning, the likelihood of a hypothesis, $L(\cdot)$, is estimated using binomial distributions.

$$L(h_1) = p_1^{k_1}(1-p_1)^{n_1-k_1}\binom{n_1}{k_1} p_2^{k_2}(1-p_2)^{n_2-k_2}\binom{n_2}{k_2}$$

$$p_1 = P(w|c) = \frac{k_1}{n_1} = \frac{O_{w,c}}{O_{w,c} + O_{\bar{w},c}}$$

13. One possible alternative to computing χ^2 for each word and city, and then aggregating these values into a final ranking of words, would be to compute a single χ^2 value for each word from a contingency table with 2 rows as in Table 2, but with one column per city. Nevertheless, this is not the standard use of χ^2 in feature selection, and we leave this possibility to future work.

$$p_2 = P(w|\bar{c}) = \frac{k_2}{n_2} = \frac{O_{w,\bar{c}}}{O_{w,\bar{c}} + O_{\bar{w},\bar{c}}}$$

k_1 (k_2) represents the occurrences of word w in city c (not in city c), and n_1 (n_2) represents all word occurrences in city c (not in city c). $L(h_0)$ is a special case of $L(h_1)$ for which p_1 and p_2 are equal, as below:

$$p_1 = p_2 = p = \frac{O_{w,c} + O_{w,\bar{c}}}{N}$$

The $Loglike$ test statistic is then expanded using observations:

$$\begin{aligned} Loglike(w) \;=\; & 2[O_{w,c}\log O_{w,c} + O_{\bar{w},c}\log O_{\bar{w},c} + O_{w,\bar{c}}\log O_{w,\bar{c}} + O_{\bar{w},\bar{c}}\log O_{\bar{w},\bar{c}} + N\log N \\ & - (O_{w,c} + O_{\bar{w},c})\log(O_{w,c} + O_{\bar{w},c}) - (O_{w,\bar{c}} + O_{\bar{w},\bar{c}})\log(O_{w,\bar{c}} + O_{\bar{w},\bar{c}}) \\ & - (O_{\bar{w},c} + O_{\bar{w},\bar{c}})\log(O_{\bar{w},c} + O_{\bar{w},\bar{c}}) - (O_{w,c} + O_{w,\bar{c}})\log(O_{w,c} + O_{w,\bar{c}})] \end{aligned}$$

Having calculated the $Loglike$ for each word–city pair, we then aggregate across cities similarly to Chi (by selecting the top-m features per city until n features are obtained), following Laere et al. (2013).[14]

4.1.2 RIPLEY'S K STATISTIC

Spatial information can also be incorporated into the hypothesis testing. For example, the Ripley K function ($Ripley$: O'Sullivan & Unwin, 2010) measures whether a given set of points is generated from a homogeneous Poisson distribution. The test statistic calculates the number of point pairs within a given distance λ over the square of the total number of points. With regards to LIW selection, the set of points (Q_w) is the subset of geotagged users using a particular word w. The test statistic is formulated as follows (Laere, Quinn, Schockaert, & Dhoedt, 2014):

$$K(\lambda) \;=\; A \times \frac{|\{p, q \in Q_w : \text{distance}(p, q) \leq \lambda\}|}{|Q_w|^2}$$

where A represents the total area under consideration (e.g., the whole of North America, or the whole globe); this is dropped when generating a ranking.

A larger value of $K(\lambda)$ indicates greater geographical compactness of the set Q_w (i.e., p and q are spatially close). However, $|Q_w|$ (i.e., the number of users who use word w) varies considerably across words, and can dominate the overall statistic. A number of variations have been proposed to alleviate this effect, including replacing the denominator with a factor based on L1, and taking the logarithm of the overall value (Laere et al., 2014). The quadratic computational complexity of $Ripley$ becomes an issue when $|Q_w|$ is large (i.e., for common words). Randomised methods are usually adopted to tackle this issue, e.g., subsampling points from training data for $Ripley$ calculation relative to different distances λ. For our experiments, we adopt the optimised implementation of Laere et al. using $\lambda = 100$km with 5K samples.

4.2 Information Theory-Based Methods

In addition to statistical methods, we also experiment with information-theoretic feature selection methods based on measures which have been shown to be effective in text classification tasks, e.g., Information Gain (IG) (Yang & Pedersen, 1997).

14. Note also that, as we will see later in our experiments, there is almost no empirical difference between the two aggregation methods for χ^2, so the choice of aggregation method here is largely arbitrary.

4.2.1 INFORMATION GAIN AND GAIN RATIO

Information Gain (IG) measures the decrease in class entropy a word brings about, where higher values indicate greater predictability on the basis of that feature. Given a set of words $\mathbf{w}$, the IG of a word $w \in \mathbf{w}$ across all cities ($\mathbf{c}$) is calculated as follows:

$$\begin{aligned} IG(w) &= H(\mathbf{c}) - H(\mathbf{c}|w) \\ &\propto -H(\mathbf{c}|w) \\ &\propto P(w)\sum_{c\in\mathbf{c}} P(c|w)logP(c|w) + P(\bar{w})\sum_{c\in\mathbf{c}} P(c|\bar{w})logP(c|\bar{w}) \end{aligned}$$

where $P(w)$ and $P(\bar{w})$ represent the probabilities of the presence and absence of word w, respectively. Because $H(\mathbf{c})$ is the same for all words, only $H(\mathbf{c}|w)$ — the conditional entropy given w — needs to be calculated to rank the features.

Words carry varying amounts of "intrinsic entropy", which is defined as:

$$IV(w) = -P(w)logP(w) - P(\bar{w})logP(\bar{w})$$

Local words occurring in a small number of cities often have a low intrinsic entropy, where non-local common words have a high intrinsic entropy (akin to inverse city frequency; see Section 4.3.1). For words with comparable IG values, words with smaller intrinsic entropies should be preferred. Therefore, following Quinlan (1993) we further normalise $IG(w)$ using the intrinsic entropy of word w, $IV(w)$, culminating in information gain ratio (IGR):

$$IGR(w) = \frac{IG(w)}{IV(w)}$$

4.2.2 LOGISTIC REGRESSION-BASED FEATURE WEIGHTS

The previous two information-theoretic feature selection methods (IG and IGR) optimise across all classes simultaneously. Given that some LIWs may be strongly associated with certain locations, but are less tied to other locations, we also conduct per-class feature selection based on logistic regression (LR) modelling.[15] We consider this method to be information theoretic because of its maximisation of entropy in cases where there is uncertainty in the training data.

Given a collection of cities $\mathbf{c}$, the LR model calculates the probability of a user (e.g., represented by word sequence: $w_1, w_2, \ldots, w_n$) assigned to a city $c \in \mathbf{c}$ by linearly combining eligible LR feature weights:

$$P(c|w_1, w_2, \ldots, w_n) = \frac{1}{Z}exp(\sum_{k=1}^{m} \lambda_k f_k)$$

where Z is the normalisation factor, m is the total number of features, and f_k and λ_k are the features and feature weights, respectively. As with other discriminative models, it is possible to incorporate arbitrary features into LR, however, a feature (function) in our task is canonically defined as a word w_i and a city c: when w occurs in the set of messages for users in class c, a feature $f_k(w_i, c)$ is

15. For the logistic regression modeller, we use the toolkit of Zhang Le (`https://github.com/lzhang10/maxent`), with 30 iterations of L-BFGS (Nocedal, 1980) over the training data.

denoted as $[\text{class} = c \wedge w_i \in c]$. Each f_k maps to a feature weight denoted as $\lambda_k \in \mathcal{R}$. The method results in a per-city word ranking with words ranked in decreasing order of λ_k, from which we derive a combined feature ranking in the same manner as $MaxChi$, following Han et al. (2012b).[16]

Notably, incorporating a regularisation factor balances model fitness and complexity, and could potentially achieve better results. We don't explicitly perform regularisation in the modelling stage. Instead, we first obtain the feature list ranked by LR as other feature selection methods and then evaluate the subset of top-n ranked features on the development data. This is in fact equivalent to "filter-based" regularisation (cf. filter-based feature selection: Guyon & Elisseeff, 2003), and we leave experimentation with regularisation integrated into the models to future work.

4.2.3 DISTRIBUTION DIFFERENCE

LIW selection can be likened to finding words that are maximally dissimilar to stop words (Chang et al., 2012). Stop words like *the* and *today* are widely used across many cities, and thus exhibit a relatively flat distribution. In contrast, LIWs are predominantly used in particular areas, and are more skewed in distribution. To capture this intuition, LIW selection is then based on the "distribution difference" across cities between stop words and potential LIW candidates (i.e., all non-stop words). Given a pre-defined set of stop words S, the distribution difference is calculated as:

$$DistDiff(w_{ns}) = \sum_{w_s \in S} Diff(w_{ns}, w_s) \frac{\text{Count}(w_s)}{\text{Count}(S)}$$

where $\text{Count}(w_s)$ and $\text{Count}(S)$ denote the number of occurrences of a stop word w_s and the total number of occurrences of all stop words, respectively. The difference (i.e., $Diff(w_{ns}, w_s)$) between a stop word w_s and non-stop word w_{ns} can be evaluated in various ways, e.g., symmetric KL-divergence ($DistDiff_{skl}$), or the total variance ($DistDiff_{tv}$) of absolute probability difference across all cities $\mathbf{c}$ (Chang et al., 2012):

$$\begin{aligned} Diff_{skl}(w_{ns}, w_s) &= \sum_{c \in \mathbf{c}} P(c|w_{ns}) \log \frac{P(c|w_{ns})}{P(c|w_s)} + P(c|w_s) \log \frac{P(c|w_s)}{P(c|w_{ns})} \\ Diff_{tv}(w_{ns}, w_s) &= \sum_{c \in \mathbf{c}} |P(c|w_{ns}) - P(c|w_s)| \end{aligned}$$

where $P(c|w_{ns})$ and $P(c|w_s)$ denote the probability of a word occurring in a city in the per-word city distribution for w_{ns} and w_s, respectively. The non-stop words are then sorted by distribution difference in decreasing order. In our experiments, we use the implementation of Chang et al..

4.3 Heuristic-Based Methods

Other than commonly-used feature selection methods, a number of heuristics can be used to select LIWs.

4.3.1 DECOUPLING CITY FREQUENCY AND WORD FREQUENCY

High-utility LIWs should have both of the following properties:

16. As with $LogLike$, the choice of aggregation method here is largely arbitrary, based on our empirical results for χ^2.

1. High Term Frequency (TF): there should be a reasonable expectation of observing it from the users' tweets in a city.

2. High Inverse City Frequency (ICF): the word should occur in tweets associated with a relatively small number of cities.

We calculate the ICF of a word w simply as:

$$icf_w = \frac{|\mathbf{c}|}{cf_w}$$

where $\mathbf{c}$ is the set of cities and cf_w is the number of cities with users who use w in the training data. Combining the two together, we are seeking words with high $TF\text{-}ICF$, analogous to seeking words with high $TF\text{-}IDF$ values in information retrieval. In standard $TF\text{-}IDF$ formulations, we multiply TF and IDF. A simple product of TF and ICF tends to be dominated by the TF component, however: for example, *twitter* scores as highly as *Jakarta*, because *twitter* has a very high TF. We resolve this by decoupling the two factors and applying a radix sort ranking: we first rank features by ICF then by TF, in decreasing order. As this approach is largely based on the inverse city frequency, we denote it as ICF below.

4.3.2 GEOGRAPHICAL SPREAD AND DENSITY

LIWs have "peaky" geographical distributions (Cheng et al., 2010). In this section, we discuss two heuristic measures for LIW selection which are based on the geographical distribution of the word.

Geographical spread ($GeoSpread$: Laere et al., 2013) estimates the flatness of a word's distribution over cities. First, the earth is divided into 1° latitude by 1° longitude cells. For each word w, the cells in which w occurs are stored. Then, all neighbouring cells containing w are merged by multi-pass scanning until no more cells can be merged. The number of cells containing w after merging is further stored. Finally, the $GeoSpread$ score for the word w is calculated as follows:

$$GeoSpread(w) \quad = \quad \frac{\#\text{ of cells containing } w \text{ after merging}}{Max(w)}$$

where $Max(w)$ represents the maximum frequency of w in any of the original unmerged cells. Smaller values indicate greater location indicativeness. This measure was originally used to rank Flickr tags by locality, e.g., *London* is more location-indicative than *beautiful*. It ignores the influence of stop words, as they are not common in Flickr tags. However, stop words like *the* are frequent in Twitter, and occur in many locations, making the numerator small and denominator large. Furthermore, stop word frequencies in cells are usually high. Consequently, *the* has a similarly small $GeoSpread$ to *London*, which is undesirable. In other words, $GeoSpread$ is flawed in not being able to distinguish stop words from local words, although it can be effective at ranking less common words (e.g., *London* vs. *beautiful*).

Geographical density ($GeoDen$: Chang et al., 2012) strategically selects peaky words occurring in dense areas. Given a subset of cities $\mathbf{c}' \subseteq \mathbf{c}$ where word $w \in \mathbf{w}$ is used, the $GeoDen$ is calculated as:

$$GeoDen(w) = \frac{\sum_{c \in \mathbf{c}'} P(c|w)}{|\mathbf{c}'|^2 \frac{\sum_{c_j, c_k \in \mathbf{c}' j \neq k} \text{dist}(c_j, c_k)}{|\mathbf{c}'|(|\mathbf{c}'|-1)}} = \frac{\sum_{c \in \mathbf{c}'} P(c|w)}{|\mathbf{c}'| \frac{\sum_{c_j, c_k \in \mathbf{c}' j \neq k} \text{dist}(c_j, c_k)}{|\mathbf{c}'|-1}}$$

where $\text{dist}(c_j, c_k)$ is the great-circle distance between cities c_j and c_k. Similarly, $P(c|w)$ denotes the distribution of word w across each city $c \in \mathbf{c}'$. The denominator is made up of the square of the number of cities $|\mathbf{c}'|$ that w occurs in (which has a similar effect to ICF above), and the average distance between all cities where w is used. LIWs generally have a skewed geographical distribution in a small number of locations, meaning that the denominator is small and the numerator is large. The issue with this measure is the computational complexity for common words that occur in many cities. Furthermore, cities containing a small number of occurrences of w should not be incorporated, to avoid systematic noise, e.g., from travellers posting during a trip. One approach to counter these issues is to set a minimum $P(c|w)$ threshold for cities, and further perform randomised sampling from $\mathbf{c}'$. In this paper, we follow Chang et al. in constructing the final $\mathbf{c}'$: first, all cities containing w are ranked by $P(c|w)$ in decreasing order, then $\mathbf{c}'$ is formed by adding cities according to rank, stopping when the sum of $P(c|w)$ exceeds a pre-defined threshold r. We choose $r = 0.1$ in our experiments, based on the findings of Chang et al..

5. Benchmarking Experiments on NA

In this section, we compare and discuss the proposed feature selection methods. In particular, we investigate whether using only LIWs for geolocation prediction is better than using the full set of features, under various configurations of models and location partitions in Section 5.2. The subsequent experiments in this section are exclusively based on the public NA dataset. We adopt the same user partitions for training, dev and test as was used in the original paper (Roller et al., 2012). We primarily use the city-based class representation in our experiments over NA, but additionally present results using the original k-d tree partitions learned by Roller et al. in Section 5.2, for direct comparability with their published results. For the distance-based evaluation measures (Acc@161 and Median), we calculate the user's location based on the centroid of their tweets, and, depending on the class representation used, represent the predicted location as either: (a) a city centre; or (b) the user-centroid for a given k-d tree cell. In the case of Acc for the city-based class representation, we map the centroid for each user to the nearest city centre ≤ 50km away, and use this as the basis of the Acc calculation. In the case that there is no city centre that satisfies this constraint,[17] we map the user to the NULL class, and will always misclassify the user.[18]

5.1 Comparison of Feature Selection Methods

First, we compare the effectiveness of the various feature selection methods on NA using the city-based class representation. In total, 214K features were extracted from the training section of NA.

17. This occurs for 1139 ($\approx 11.4\%$) of test users.
18. As such, the upper bound Acc for the city-based representation is 0.886. Note also that the Acc for the k-d tree vs. city-based representation is not comparable, because of the different class structure and granularity.

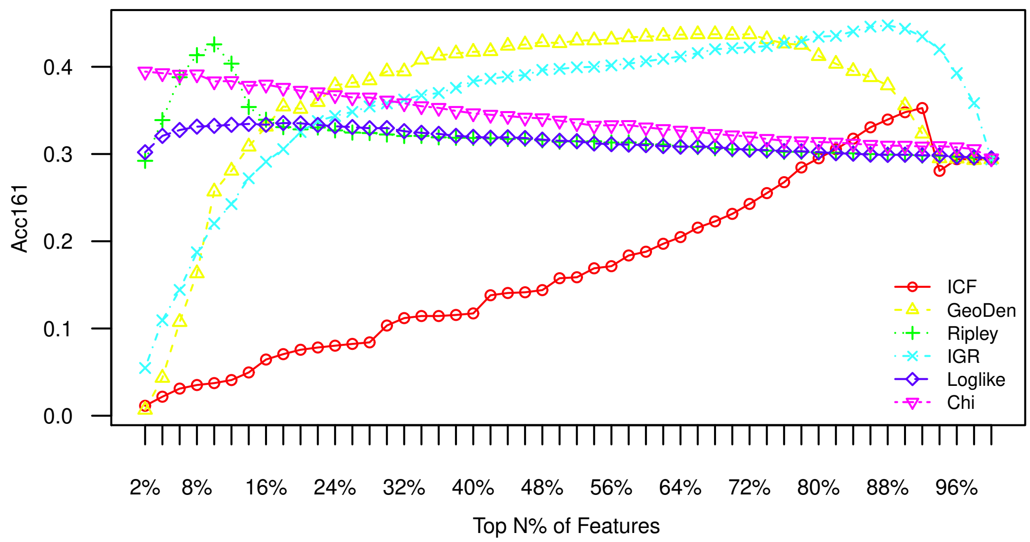

Figure 3: Acc@161 for varying levels of feature selection on the NA dataset, based on the city-based class representation.

We select the top-$n\%$ of features, with a step size of 2%, and then use the selected features within a multinomial naive Bayes learner (we return to explore the choice of learner in Section 5.2). The tuning of n for all methods is based on Acc@161 over the 10K held-out users in the development data. We present results for a sample of feature selection methods in Figure 3, omitting methods which are largely identical in behaviour to other methods presented in the graph, namely:

- $\{DistDiff_{tv}, DistDiff_{skl}\} \equiv ICF$
- $MaxChi \equiv Chi$
- $\{LR, IG, GeoSpread\} \equiv LogLike$

For all methods, the best result is achieved with a proper subset of features based on feature selection, although the proportion of the features that gives the best results for a given method varies greatly (e.g., the optima for $Ripley$, IGR and $GeoDen$ are 10%, 88% and 66%, respectively). This observation agrees with the expectations that: (1) when only a small number of features is used, the trained model generally underfits the data; and (2) if the model is trained using the full feature set, noisy words (e.g., *the*) cause overfitting. For instance, when using just the top 2% of features in IGR, the most likely class for users with features — noting that users with no feature representation will default to the majority class, namely Los Angeles, US-CA — is Monterrey, MX, because Spanish words are highly location-indicative of the small number of Mexican cities in the NA dataset. The features which are selected last are generally high-frequency function words (e.g., *the*) and common words (e.g., *facebook*), which give little indication as to geolocation, and lead to prediction errors.

Two patterns can be observed in the results: (1) Chi, $MaxChi$, IG, $LogLike$, $GeoSpread$, LR and $Ripley$ (i.e., "local" methods, which initially select features for each class, with the exception of IG and $Ripley$) achieve their highest Acc@161 at an early stage, then the numbers drop gradually; and (2) ICF, IGR, $DistDiff_{skl}$, $DistDiff_{tv}$ and $GeoDen$ (i.e., the "collective" group, which select features for all classes at once) gradually increase in accuracy as more features are added, reach a peak when the majority of features are selected, then drop off in accuracy sharply. This difference in behaviour can be attributed to the types of word that are preferred by the methods. The "local" methods tend to prefer 1-local words — taking LR, for example, city names (e.g., *philadelphia*) and names of upper-level administrative regions (e.g., *georgia*) frequently occur in the upper reaches of the ranking. In addition to these gazetted words, many local/regional words are also found in the upper reaches of the feature ranking, including informal place names (e.g., *philly*, an informal name for Philadelphia, US-PA), local transport references (e.g., *skytrain*, a public transport system in Vancouver, CA) and local greetings (e.g., *aloha* in Honolulu, US-HI). However, it is reasonable to believe that 1-local words — words that are predominantly used in one city and are rarely mentioned in other cities — are not common. As a result, the accuracy is bounded by the limited number of true 1-local words. This could be the reason for the early, yet remarkably high, peak in accuracy, and subsequent sharp decline, for $Ripley$; because of its reliance on pairwise distances between users using a given word, $Ripley$ tends to rank 1-local words highly. In contrast, the "collective" methods assume words carry varying amounts of geospatial information. By leveraging combinations of LIWs, the true location of a user can be collectively inferred. For instance, *brunswick* is a common suburb/street name in many cities, e.g., Melbourne, AU and London, GB. This word alone is insufficient to make reliable predictions. However, if other LIWs (e.g., *tram* and *Flinders*, which are again not uniquely disambiguating in themselves) are also observed, then the chance of the location being Melbourne, AU becomes high, since it is unlikely that users from cities other than Melbourne, AU would use that combination of words. This strategy can also be explained in information-theoretic terms: by knowing more words, extra information is obtained, and consequently the entropy is continuously reduced and the prediction of geolocation becomes more certain.

Among all the feature selection methods, IGR, $GeoDen$ and $Ripley$ are the stand-out methods in terms of Acc@161. We further compare the accuracy of classifiers trained using the optimised set of LIWs (based on the development data) to that of the full model. The performance is measured on the 10K held-out test users, using the city-based class representation. The results are displayed in Table 3 (for the same subset of feature selection methods as were displayed in Figure 3), and show that using LIWs offers an improvement over the full feature set for all evaluation measures and all feature selection methods, except for slight dips in Acc@C for IGR and $GeoDen$. Nevertheless, these numbers clearly demonstrate that feature selection can improve text-based geolocation prediction accuracy. IGR performs best in terms of accuracy, achieving 8.9% and 14.2% absolute improvements in Acc and Acc@161, respectively, over the full feature set.

5.2 Comparison with Benchmarks

We further compare the best-performing method from Section 5.1 with a number of benchmarks and baselines. We experiment with two class representations: (1) the city-based class representation based on `Geonames`; and (2) the k-d tree based partitioning of Roller et al. (2012), which creates grid cells containing roughly even amounts of data of differing geographical sizes, such that higher-population areas are represented with finer-grained grids.[19] For both class representations,

19. Recent work (Schulz et al., 2013) also considers irregular-sized polygons, based on administrative regions like cities.

Dataset	Features	Acc	Acc@161	Acc@C	Median
NA	Full	0.171	0.308	0.831	571
	ICF	0.209	0.359	0.840	533
	Chi	0.233	0.402	**0.850**	385
	IGR	**0.260**	**0.450**	0.811	**260**
	LogLike	0.191	0.343	0.836	489
	GeoDen	0.258	0.445	0.791	282
	Ripley	0.236	0.432	0.849	306

Table 3: Results on the full feature set compared to that for each of a representative sample of feature selection methodologies on NA with the city-based class representation. The best numbers are shown in boldface.

we compare learners with and without feature selection. As observed previously, Acc is not comparable across the two class representations. Results based on the distance-based measures (Acc@161 and Median), on the other hand, are directly comparable. Acc@C results are not presented for the k-d tree based class representation because the k-d tree cells do not map cleanly onto national borders; although we could certainly take the country in which the centroid of a given k-d tree cell lies as the country label for the entire cell, such an approach would ignore known geo-political boundaries.

We consider the following methods:

Baseline: Because the geographical distribution of tweets is skewed towards higher-population areas (as indicated in Figure 1), we consider a most-frequent class baseline. We assign all users to the coordinates of the most-common city centre (or k-d tree grid centroid) in the training data.

Placemaker: Following Kinsella et al. (2011), we obtain results from Yahoo! Placemaker,[20] a publicly-available geolocation service. The first 50K bytes (the maximum query length allowed by Placemaker) from the tweets for each user are passed to Placemaker as queries. The returned city centre predictions are mapped to our collapsed city representations. For queries without results, or with a predicted location outside North America, we back off to the most-frequent class baseline.[21]

Multinomial naive Bayes: This is the same model as was used in Section 5.1.

KL divergence: The previous best results over NA were achieved using KL divergence and a k-d tree grid (Roller et al., 2012). Using a k-d tree, the earth's surface is partitioned into near-rectangular polygons which vary in size, but contain approximately the same number of users. Locations are represented as cells in this grid. KL divergence is then utilised to measure the similarity between the distribution of words in a user's aggregated tweets and that in each grid cell, with the predicted location being the centroid of the most-similar grid cell.[22]

20. `http://developer.yahoo.com/geo/placemaker/`, accessed in August 2012.
21. An alternative would be to query Placemaker with each tweet, and then aggregate these predictions (e.g., by selecting the majority location) to get a final user-level prediction. However, Kinsella et al. (2011) found the accuracy of such an approach to be largely similar to that of the approach we use.
22. We use the same settings as Roller et al. (2012): a median-based k-d tree partition, with each partition containing approximately 1050 users.

Partition	Method	Acc	Acc@161	Acc@C	Median
City	Baseline	0.003	0.062	**0.947**	3089
	Placemaker	0.049	0.150	0.525	1857
	NB	0.171	0.308	0.831	571
	NB+IGR	**0.260**	**0.450**	0.811	**260**
	LR	0.129	0.232	0.756	878
	LR+IGR	0.229	0.406	0.842	369

Table 4: Geolocation performance using city-based partition on NA. Results using the optimised feature set (+IGR) are also shown. The best-performing method for each evaluation measure and class representation is shown in boldface.

Partition	Method	Acc	Acc@161	Acc@C	Median
k-d tree	Baseline	0.003	0.118	–	1189
	NB	0.122	0.367	–	404
	NB+IGR	0.153	0.432	–	280
	KL	0.117	0.344	–	469
	KL+IGR	**0.161**	**0.437**	–	**273**

Table 5: Geolocation performance using k-d tree-based partition on NA. Results using the optimised feature set (+IGR) are also shown. The best-performing method for each evaluation measure and class representation is shown in boldface.

Logistic regression: We also apply logistic regression from Section 4.2.2 as a learner. Instead of modelling all the data, we use only the IGR-selected features from Section 5.1. While regularisation is commonly employed in logistic regression learners, we made a conscious choice not to use it in our experiments as the implementation of the regulariser would differ across learners and complicate the direct comparison of feature selection methods (i.e. it would be difficult to tease apart the impact of the specific regulariser from the feature selection). Having said that, if the objective were to maximise the raw classifier accuracy — as distinct from exploring the impact of different features and feature selection methods on classification accuracy — we would advocate the incorporation of a regulariser.

Instead of evaluating every possible combination of model, partition and feature set, we choose representative combinations to test the extent to which LIWs improve accuracy. The results on the city-based partition are shown in Table 4. We begin by considering the baseline results. The most-frequent class for the city-based representation is Los Angeles, US-CA.[23] Both the majority class baseline and Placemaker perform well below multinomial naive Bayes (NB) and logistic regression (LR), and have very high Median distances. Furthermore, when using the features selected in Section 5.1 (i.e., NB+IGR and LR+IGR), the performance is further improved by a large margin for both models, demonstrating that identification of LIWs can improve text-based geolocation prediction. Finally, although LR performs poorly compared to NB, LR+IGR still improves substantially

23. New York is further divided into suburbs, such as manhattan-ny061-us, brooklyn-ny047-us, in `Geonames`. As an artefact of this, these suburbs are not merged into a single city.

over LR. We plan to further explore the reasons for LR's poor performance in future work. Overall, NB+IGR performs best for the city-based representation in terms of Acc, Acc@161, and Median distance.

Turning to the k-d tree-based partition in Table 5, we again observe the low performance of the most-frequent class baseline (i.e., a grid cell near New York state). NB and KL — representative generative and discriminative models, respectively — are evaluated using software provided by Roller et al. (2012).[24] Both approaches clearly outperform the baseline over the k-d tree class representation. Furthermore, performance increases again when using the resultant feature set of LIWs,[25] demonstrating that for a variety of approaches, identification of LIWs can improve text-based geolocation.

Overall, compared to the previously-published results for the k-d tree based representation (KL), IGR-based feature selection on the city-based partition achieves a 10.6% absolute improvement in terms of Acc@161, and reduces the Median prediction error by 209km.

From the results on the k-d tree based representation, it is not clear which of KL or NB is better for our task: in terms of Acc@161, NB outperforms KL, but KL+IGR outperforms NB+IGR. All differences are small, however, suggesting that the two methods are largely indistinguishable for the user geolocation task. As to the question of which class representation should be used for user geolocation, empirically, there seems to be little to separate the two, although further experimentation may shed more light on this issue. The city-based approach is intuitive, and enables a convenient country-level mapping for coarser-grained geolocation tasks. Furthermore, our observation from Figure 1 suggests most Twitter users are from cities. We therefore use the city-based partition for the remainder of this paper for consistency and ease of interpretation.

A spin-off benefit of feature selection is that it leads to more compact models, which are more efficient in terms of computational processing and memory. Comparing the model based on LIWs selected using IGR with the full model, we find that the prediction time is faster by a factor of roughly five.

6. Experiments on WORLD

In addition to establishing comparisons on NA, we further evaluate the feature selection methods on WORLD. This extends the evaluation from regional benchmarks to global geolocation performance. Similar to NA, for WORLD we reserve 10K random users for each of dev and test, and the remainder of the users are used for training (preprocessed as described in Section 3.4). Here and in all experiments over WORLD and related datasets, we base our evaluation on the city label set.

We apply the same tuning procedure as was used over NA to obtain the optimal feature set for each feature selection method. We present results for a representative sample of the best-performing methods in Figure 4. Once again, we omit methods that are largely identical in behaviour to other methods, namely:

- $\{DistDiff_{tv}, DistDiff_{skl}\} \equiv ICF$
- $\{MaxChi, Chi, LogLike, IG, GeoSpread\} \equiv LR$

24. https://github.com/utcompling/textgrounder/wiki/RollerEtAl_EMNLP2012
25. Note that after LIWs are selected, a small proportion of users end up with no features. These users are not geolocatable in the case of KL, a discriminative model. We turn off feature selection for such users, and backoff to the full feature set, so that the number of test instances is consistent in all rows.

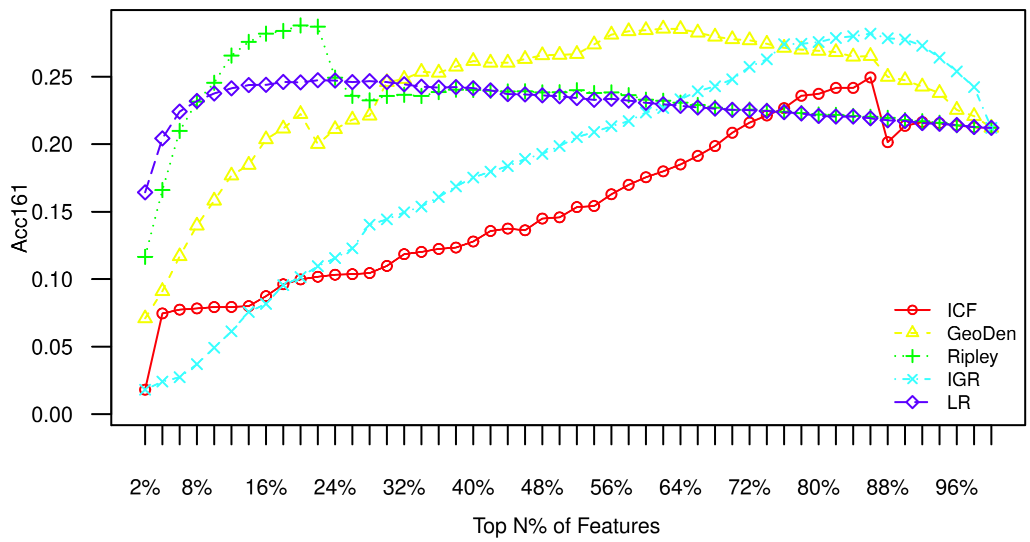

Figure 4: Acc@161 for varying percentages of features selected using representative feature selection methods on the WORLD dataset.

The biggest differences over Figure 3 are: (1) the χ^2-based methods converge in behaviour with LR, $LogLike$ and related methods; and (2) LR performs marginally better than $LogLike$, and is thus the method we present in the graph.

Despite the difference in scope and data size, the overall trend over WORLD mirrors that for NA. In particular, $GeoDen$, IGR and $Ripley$ achieve the best Acc@161 numbers on the dev data, although the numbers are lower than those achieved for NA in Figure 3. This is because WORLD has fewer tweets per user than NA (as we only utilise geo-tagged data), and disambiguation at the global level also makes it a more challenging task.

The results for multinomial naive Bayes with the chosen feature selection methods on WORLD are shown in Table 6. Again $GeoDen$ (62%), IGR (86%) and $Ripley$ (20%) achieve the best accuracy, although there is no clear winner: IGR achieves the best Acc and $Ripley$ achieves the best Acc@161. Nevertheless, the improved city-based Acc and Acc@161 numbers confirm the general effectiveness of feature selection. On the basis of these similar results and the earlier NA results (in which IGR delivers better results), we adopt IGR as our default LIW feature selection method for the remainder of the paper.

In summary, the findings on the utility of feature selection in Table 3 (NA) and Table 6 (WORLD) tell a similar story, namely that feature selection improves user geolocation accuracy. The impact of feature selection on NA is much greater than WORLD, because WORLD has a larger number of classes and smaller average number of tweets per user and also per class, making it a more challenging dataset.

Dataset	Features	Acc	Acc@161	Acc@C	Median
WORLD	Full	0.081	0.200	**0.807**	886
	ICF	0.110	0.241	0.788	837
	IGR	**0.126**	0.262	0.684	913
	LR	0.104	0.233	0.792	**640**
	GeoDen	0.123	0.266	0.691	842
	Ripley	0.121	**0.268**	0.582	1128

Table 6: Results on the full feature set compared to that of each of a representative sample of feature selection methodologies on WORLD using NB. The best numbers are shown in boldface.

Train	Test	Acc	Acc@161	Acc@C	Median
G	G	0.126	0.262	0.684	913
G+NG	G	0.170	0.323	0.733	615
G	G+NG	0.187	0.366	0.835	398
G+NG	G+NG	0.280	0.492	0.878	170
G	NG	0.161	0.331	0.790	516
G+NG	NG	0.241	0.440	0.826	272
G	G-small	0.121	0.258	0.675	960
G	NG-small	0.114	0.248	0.666	1057

Table 7: The results of geolocation models trained and tested on geotagged (G) and non-geotagged (NG) tweets, and their combination.

7. Exploiting Non-geotagged Tweets

Most Twitter-based geolocation research carried out to date (Eisenstein et al., 2010; Wing & Baldridge, 2011) has been trained only on geotagged tweets, that is tweets with known geographical coordinates. Some work (Roller et al., 2012) has also incorporated non-geotagged tweets from users whose location can be inferred from geotagged tweets. Clearly, if it is possible to effectively utilise non-geotagged tweets, data sparsity can be ameliorated (as we aren't restricting ourselves to training on only the approximately 1% of tweets with known location), but there is a clear tradeoff in the confidence we can place in the labels associated with those tweets/users. In this section, we investigate the utility of non-geotagged tweets in geolocation prediction.

For experiments in this section, and the rest of the paper, we use WORLD+NG to denote the dataset which incorporates both the geotagged and non-geotagged tweets from the users in WORLD. We refer to the subparts of this dataset consisting of geotagged and non-geotagged tweets as G and NG, respectively. Of the 194M tweets in WORLD+NG, 12M are geotagged and the remaining 182M are non-geotagged. We use the same partitioning of users into training, development, and testing sets for WORLD+NG as for WORLD. We compare the relative impact of NG in which we train and test the geolocation method on G, NG, or their combination. Results are presented in Table 7.

The first row of Table 7 shows the results using only geotagged data (our best result from Table 6). In rows two and three, we show results when the data for each user in the training and test datasets, respectively, is expanded to incorporate non-geotagged data (without changing the set of users or the label for any user in either case). In both cases, for all evaluation measures, the performance is substantially better than the benchmark (i.e., the first row). This finding is in line with Cheng et al.'s (2010) results that data spareness is a big issue for text-based geolocation. It also validates our hypothesis that non-geotagged tweets are indicative of location. The best results are achieved when non-geotagged tweets are incorporated in both the training and testing data (shown in row four). In this case we achieve an accuracy of 28.0%, a 15.4 percentage point increase over the benchmark using only geotagged tweets to represent a given user. Moreover, our prediction is within 161km of the correct location for almost one in every two users, and the country-level accuracy reaches almost 88%.[26]

Although research on text-based geolocation has used geotagged data for evaluation, the ultimate goal of this line of research is to be able to reliably predict the locations of users for whom the location is not known, i.e., where there is only non-geotagged data. Because geotagged tweets are typically sent via GPS-enabled devices such as smartphones, while non-geotagged tweets are sent from a wider range of devices, there could be systematic differences in the content of geotagged and non-geotagged tweets. We examine this issue in rows five and six of Table 7, where we test our model on only non-geotagged data. In this case we know a test user's gold-standard location based on their geotagged tweets. However these geotagged tweets are not used to represent the user in the test instance; instead, the user is represented only by their non-geotagged tweets. The results here are actually better than for experiments with the same training data but tested on geotagged tweets (i.e., rows one and two of the table).[27] This confirms that a model trained on G or G+NG indeed generalises to NG data. However, it is not clear whether this finding is due to there being much more non-geotagged than geotagged data for a given user, or whether some property of the non-geotagged data makes it easier to classify. To explore this question, we carry out the following additional experiment. First, we construct a new dataset NG-small by down-sampling NG to contain the same number of features per user as G (in terms of the feature token count). To make the comparison fairer, we constructed a second new dataset — G-small — in which we exclude test users with more G tweets than NG tweets. This guarantees that users in NG-small will contain the same number of LIWs as in G-small. We average over five iterations of random subsampling, and list the result in the final row of Table 7.[28] Here we see that the results for NG-small are not as good as G-small (i.e., row seven), suggesting that there might be minor sub-domain differences between geotagged and non-geotagged tweets, though a strong conclusion cannot be drawn without further in-depth analysis. One possible explanation is that there could be differences (e.g., demographic variations) between users who only have non-geotagged tweets and users who have both non-geotagged tweets and geotagged tweets; however, comparing these two sources is beyond the scope of this paper. Nonetheless, the results suggest the difference between NG and G is largely due to the abundant data in NG. This explanation is also supported by the recent work of Priedhorsky et al. (2014).

26. Note that this evaluation is over exactly the same set of users in all four cases; all that changes is whether we incorporate extra *tweets* for the pre-existing set of users, in the training or test data.
27. We remove users who only have geotagged tweets in the test data, reducing the number of users marginally from 10,000 to 9,767.
28. Note that we calculated the variance over the five iterations of random subsampling, and found it to be negligible for all evaluation measures.

In summary, we have quantitatively demonstrated the impact of non-geotagged tweets on geolocation prediction, and verified that models trained on geotagged data are indeed applicable to non-geotagged data, even though minor sub-domain differences appear to exist. We also established that representing a user by the combination of their geotagged and non-geotagged tweets produces the best results.

8. Language Influence on Geolocation Prediction

Previous research on text-based geolocation has primarily focused on English data. Most studies have either explicitly excluded non-English data, or have been based on datasets consisting of primarily English messages, e.g., through selection of tweets from predominantly English-speaking regions (Eisenstein et al., 2010; Cheng et al., 2010; Wing & Baldridge, 2011; Roller et al., 2012). However, Twitter is a multilingual medium and some languages might be powerful indicators of location: for example, if a user posts mostly Japanese tweets, this could be a strong indication that the user is based in Japan, which could be used to bias the class priors for the user. In this section, we explore the influence of language on geolocation prediction. The predominant language in a given tweet was identified using `langid.py`,[29] which has been trained to recognise 97 languages (Lui & Baldwin, 2012).

To create a dataset consisting of multilingual geotagged tweets, we extract all geotagged data — regardless of language — from the same Twitter crawl that WORLD was based on. This multilingual dataset consists of 23M tweets from 2.1M users. 12M tweets are in English as in WORLD, while the remaining 11M tweets are in other languages. Figure 5 shows the proportion of tweets in the fifteen most common languages in the dataset.[30] An immediate observation is the large difference in language distribution we observe for geo-tagged tweets as compared to what has been observed over all tweets (irrespective of geotag: Hong, Convertino, & Chi, 2011; Baldwin, Cook, Lui, MacKinlay, & Wang, 2013): among the higher-density languages on Twitter, there appears to be a weak positive bias towards English users geotagging their tweets, and a strong negative bias against Japanese, Korean and German users geotagging their tweets. We can only speculate that the negative bias is caused by stronger concerns/awareness of privacy issues in countries such as Japan, South Korea, Germany and Austria. We explored the question of whether this bias was influenced by the choice of Twitter client by looking at the distribution of Twitter clients used to post messages in each of English, German, Japanese and Korean: (a) overall (irrespective of whether the message is geotagged or not), based on a 1M sample of tweets from 28 Sep, 2011; and (b) for geotagged tweets, based on WORLD. Overall, we found there to be huge variety in the choice of client used within a given language (with the top-10 clients accounting for only 65–78% of posts, depending on the language), and significant differences in popular clients between languages (e.g. "Keitai Web" is the most popular client for Japanese, "web" for English and German, and "Twitter for Android" for Korean). For geotagged tweets, on the other hand, there is much greater consistency, with the three most popular clients for all languages being "Twitter for iOS", "Twitter for Android" and "foursquare", accounting for a relatively constant two-thirds of posts for each language. This is suggestive of the fact that the choice of client is one factor in biasing the relative proportion of

29. Based on the simplifying assumptions that: (a) every tweet contains linguistic content; and (b) all tweets are monolingual, or at least are predominantly in a single language.
30. We represent languages in Figure 5 using two-letter ISO 639-1 codes.

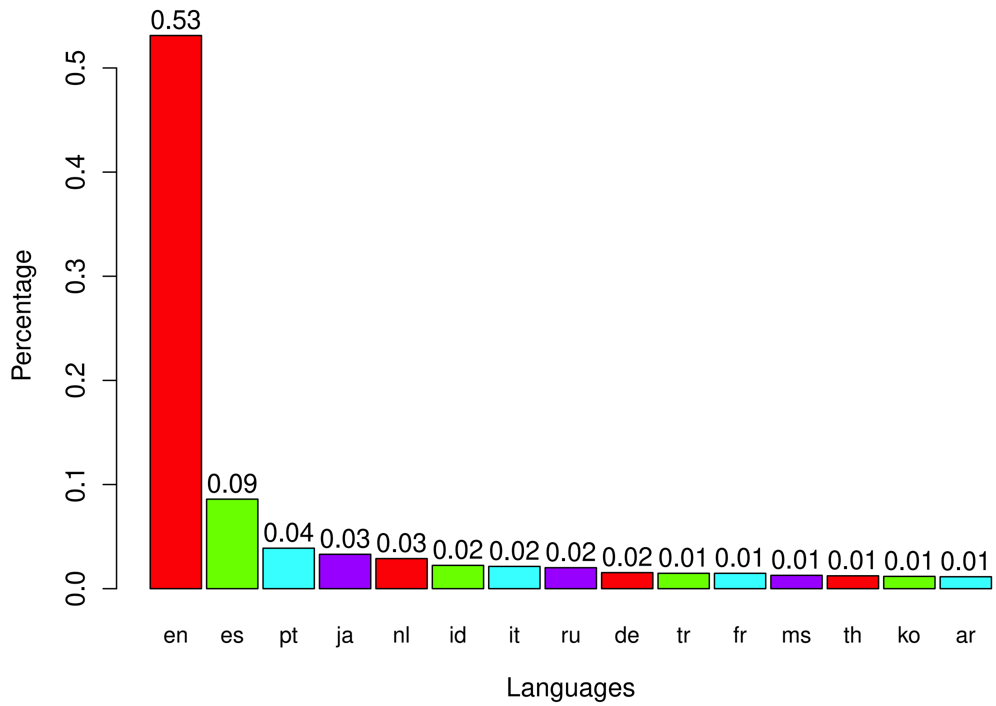

Figure 5: The percentage of tweets in WORLD+ML written in each of the fifteen most frequent languages in the collected Twitter data. These fifteen languages account for 88% of the tweets in the full dataset.

geotagged tweets in the different languages, although more research is required to fully understand this effect.

The training, development and test data is re-partitioned for the multilingual setting to stratify on language, and the resultant dataset is referred to as WORLD+ML. Again, the development and testing sets consist of 10K users each, with the remaining users in the training set as in WORLD. Although in Section 7 we showed that adding non-geotagged data improves geolocation accuracy, the experiments in this section are based only on geotagged data, because of the prohibitive computational cost of experimenting with a much larger dataset. Note that this doesn't limit the generalisability of our results, it simply means that we have to be careful to compare them to the monolingual results from Table 7 based on only geotagged tweets (the first row).

We first compare geolocation performance in a multilingual setting with that in an English-only setting, a comparison that past work on geolocation has not considered. The data in WORLD+ML is further partitioned into two subsets — E and NE — according to whether the majority of a given user's tweets are in English or non-English, respectively. Of the 10K test users in WORLD+ML, 5,916 are English and 4,084 are non-English. One challenge with the multilingual setting of these experiments is tokenisation. Although rudimentary tokenisation of many languages such as English and French can be accomplished using whitespace and punctuation, tokenisation is much more challenging for languages such as Japanese and Chinese which do not represent word boundaries

Train	Test	Acc	Acc@161	Acc@C	Median
E+NE	E+NE	0.196	0.343	0.772	466
E+NE	E	0.134	0.256	0.715	1067
E+NE	NE	0.287	0.468	0.855	200
E	E	0.169	0.317	0.746	632

Table 8: Results for multilingual geolocation, training and testing on English (E) and non-English (NE) users, and their combination.

with whitespace. However, amongst the most-common languages on Twitter (as shown in Figure 5), Japanese is the only language which accounts for a substantial portion of the data ($> 1\%$) and requires a specialised tokenisation strategy (compared to English). For Japanese tweets we apply the Japanese morphological segmenter `MeCab` (with the IPA dictionary),[31] and post-correct tokenisation errors relating to Twitter-specific tokens such as mentions, hashtags, and URLs (e.g., in instances where `MeCab` over-segments a mention into multiple morphemes). For non-Japanese tweets, we apply the same tokeniser based on regular expressions used in our previous English-only experiments.

After resolving the tokenisation issue, we apply the same IGR method from Section 4.2.1 to select the optimised feature selection cut-off, based on Acc over the development data. We observe that a much larger proportion of tokens are selected in the multilingual setting compared to the English-only experiments. For example, of the 400K token types in the multilingual experiment, 384K (the top 96%) are selected as location-indicative, while for the English-only case 83K (the top 86%) location-indicative words are selected from the total of 96K token types.

The experimental results are shown in Table 8.[32] The first row gives results for training and testing on the full dataset of both English and non-English tweets. The next two rows show the results when testing on English (E) and non-English (NE) subsets of the data. The much lower accuracy for E compared to NE indicates that English tweets are much more difficult to geolocate than non-English tweets. One reason for this is that for many non-English languages, there is a strong bias towards a small number of cities. We verify this by calculating the class entropy with respective to a language on the training data. The class probabilities are smoothed using a simple add-α method, with $\alpha = 1/3709$ (where 3709 is the size of the class set). As shown in Table 9, the class entropy on English (en) data is the largest, indicating that English is prevalent across a large number of locations. In contrast, Thai (th) and Turkish (tr) have much smaller entropies, suggesting the location distributions are heavily skewed, and user geolocation over these languages will be easier than for English.

To explore the extent to which the geolocatability of a user varies with respect to the predominant language of their tweets, we further break down the results by language in Table 10, which shows results for the top-10 most frequent languages (by number of tweets) with at least 100 users in our test data. This cut-off on users ensures we do not consider under-represented languages.

31. `http://sourceforge.net/projects/mecab/`
32. The English-only results reported here are not the same as for the comparable experiment in Table 7 using only geotagged data, because the test sets consist of different users in these two cases.

Language	Entropy	Language	Entropy	Language	Entropy
en	6.279	id	3.868	fr	5.538
es	5.069	it	5.244	ms	3.970
pt	4.144	ru	3.772	th	2.697
ja	3.523	de	6.207	ko	2.781
nl	3.820	tr	2.888	ar	3.281

Table 9: Geolocation class entropy for top-15 languages

		Per-language Majority Class				Unified Multilingual				Monolingual Partitioning			
Lang.	No.	Acc	Acc@161	Acc@C	Med.	Acc	Acc@161	Acc@C	Med.	Acc	Acc@161	Acc@C	Med.
en	5916	0.019	0.039	0.655	3671	0.134	0.256	0.715	1067	0.169	0.317	0.746	632
es	945	0.116	0.159	0.324	4267	0.267	0.346	0.734	391	0.362	0.478	0.802	185
pt	673	0.223	0.296	0.952	490	0.232	0.305	0.952	490	0.400	0.489	0.961	200
id	398	0.264	0.472	0.899	197	0.324	0.565	0.965	115	0.440	0.736	0.960	16
nl	342	0.175	0.789	0.889	87	0.173	0.789	0.889	87	0.298	0.871	0.845	58
ja	298	0.326	0.530	0.960	96	0.336	0.544	0.956	95	0.463	0.695	0.950	27
ru	217	0.336	0.378	0.857	633	0.346	0.387	0.862	633	0.341	0.378	0.862	633
tr	186	0.538	0.656	0.930	0	0.538	0.656	0.930	0	0.522	0.645	0.930	0
ar	164	0.335	0.470	0.463	379	0.354	0.488	0.500	301	0.457	0.591	0.750	21
th	154	0.325	0.766	0.981	20	0.279	0.623	0.792	41	0.325	0.766	0.974	30
All	10000	0.107	0.189	0.693	2805	0.196	0.343	0.772	466	0.255	0.425	0.802	302

Table 10: Geolocation performance and comparison for the top-10 most frequent languages in the multilingual test data, using (1) language prior (i.e., the city where a language is mostly used); (2) a unified multilingual model (i.e., training and testing on multilingual data regardless of languages); and (3) language-partitioned monolingual models (i.e., first identify the primary language of users, train one model per language, and classify test users with the model corresponding to the language of their tweets)

We observe that the results vary remarkably by language in the multilingual section of Table 10. The results are overall lowest for English (en), although the lowest country-level accuracy is for Arabic (ar); we speculate that this is caused by the large number of countries that Arabic is spoken in, and the relatively small number of Arabic speakers in our training data. Furthermore, the city-level accuracy is better than 30% for Indonesian (id), Japanese (ja), Russian (ru), Turkish (tr) and Arabic (ar); the regions in which these languages are commonly-spoken are more geographically-restricted than for English, suggesting that geolocation accuracy on languages with smaller geographic footprints will tend to be higher than for languages which are widely-used throughout a larger geographical area. This finding agrees with the recent work of Priedhorsky et al. (2014), and further underlines the power of language information in predicting locations. The best city-level accuracy of 53.8% is observed for Turkish (one of the languages with the lowest city-level entropy). Manually inspecting the outputs, we find that this is because our model predicts the city Istanbul for all Turkish users, and a large proportion of Turkish tweets come from this city.

Based on this finding, we further consider a language-based benchmark which predicts the most frequent city given the predominant language of a user's tweets (denoted as Per-language Majority Class). We also observe the performance gap between the multilingual model on English (the second row of Table 8) and an English-only model (the bottom row in Table 8). These results show that if the target data is known to be written in a single language then a monolingual model outperforms a multilingual one. It also suggests an alternative approach for multilingual geolocation prediction: rather than training and predicting on multilingual data (E+NE), we can train and evaluate models on language-specific data. Motivated by this observation, we also apply a monolingual partitioned model for users of a particular language based on `langid.py` (i.e., language partitions), e.g., selecting all Japanese users in the training data, and only applying the Japanese-specific model to Japanese users in the test data. This is denoted as Monolingual Partitioning in Table 10, and is contrasted with the simple approach of a combined model for all languages and users ("Unified Multilingual").

By comparing the Per-language Majority Class with the Unified Multilingual model, we find that the unified model performs better overall, with the exception of Thai (th) and Dutch (nl), both of which are associated with a very small number of cities, and one city which is much larger than the others (Bangkok, TH and Amsterdam, NL, respectively). Because of the relatively poor results for this benchmark method on languages such as English (en) and Spanish (es) which are frequent on Twitter, and its relatively poor overall performance, the Per-language Majority Class is not an appropriate method for this task. Nevertheless, when using a Monolingual Partitioning model, the results are far superior, and the partitioning effect of language can be seen. This suggests that modelling each language independently can improve geolocation performance.

In summary, this series of experiments has shown the influence of language on geolocation prediction. Among the top-10 languages found on Twitter, English is the most difficult to perform user geolocation over, as English is the most global language. Despite language variance, multilingual geolocation prediction is certainly feasible, although the best way to leverage language for geolocation prediction is by training language-partitioned monolingual models and geolocating users based on their primary language.

9. Incorporating User Meta Data

The metadata accompanying tweets is a valuable source of geographical information beyond that available in tweets. In this section, we explore incorporating metadata information into our text-based geolocation system. We begin by selecting four metadata fields that could potentially provide insights into the location of a user, and first evaluate models trained on each of these sources of information. We then consider a number of ways to incorporate information from this metadata with our best text-based method developed in Section 7. As discussed in Section 8, language has a strong influence on geolocation prediction, and English-posting users are the hardest to geolocate. As such, we experiment only on English data (i.e., WORLD+NG) for the remainder of this paper.

Data	LOC	TZ	DESC
Training	0.813	0.752	0.760
Test	0.813	0.753	0.761

Table 11: The proportion of users with non-empty metadata fields in WORLD+NG

9.1 Unlock the Potential of User-Declared Metadata

We choose the following four user-supplied metadata fields for our study: location (LOC), timezone (TZ), description (DESC), and the user's real name (RNAME).[33] In contrast to rich social network information which is much more expensive to extract, these metadata fields are included in the JSON object that is provided by the Twitter Streaming API, i.e., we can extract this metadata at no extra crawling cost. This information, however, is dynamic, i.e., users can change their profiles, including the metadata of interest to us. By aggregating the extracted tweet-level metadata for each user, we can calculate the ratio of users that change each metadata field. 18% of users changed their DESC field during the approximately five months over which our dataset was collected. During this same time period, for each of the other fields considered, less than 8% of users updated their data. Given the relatively small number of profile updates, we ignore the influence of these changes, and use the most frequent value for each metadata field for each user in our experiments.

All of this user-supplied metadata can be imprecise or inaccurate, because the user is free to enter whatever textual information they choose. For example, some LOC fields are not accurate descriptions of geographical locations (e.g., *The best place in the universe*). Moreover, although some LOC fields are canonical renderings of a user's true location (e.g., *Boston, MA, USA*), a large number of abbreviations and non-standard forms are also observed (e.g., *MEL* for Melbourne, AU). Cheng et al. (2010) find that only a small proportion of location fields in their US-based dataset are canonical locations (i.e., of the form *city, state*). Nevertheless, these non-standard and inaccurate location fields might still carry information about location (Kinsella et al., 2011), similarly to how the text of tweets can indicate location without explicitly mentioning place names.

These metadata fields also differ with respect to the explicitness of the location information they encode. For instance, while LOC and TZ can give direct location information, DESC might contain references to location, e.g., *A geek and a Lisp developer in Bangalore*. Although RNAME does not directly encode location there are regional preferences for names (Bergsma, Dredze, Van Durme, Wilson, & Yarowsky, 2013), e.g., *Petrov* might be more common in Russia, and the name *Hasegawa* might be more common in Japan. Finally, for all of the tweets that we consider, the text field (i.e., the content of the tweet itself) and RNAME are always present, but LOC, TZ, and DESC can be missing if a user has chosen to not supply this information. The proportion of non-empty metadata fields for LOC, TZ and DESC for users in WORLD+NG are listed in Table 11.

9.2 Results of Metadata-Based Classifiers

Because of the variable reliability and explicitness of the selected metadata, we incorporate these fields into our statistical geolocation model in a similar manner to the message text. In prelimi-

33. The user-supplied real name could be any name — i.e., it is not necessarily the user's actual name — but is a different field from the user's screen name.

Classifier	Acc	Acc@161	Acc@C	Median
LOC	0.405	0.525	0.834	92
TZ	0.064	0.171	0.565	1330
DESC	0.048	0.117	0.526	2907
RNAME	0.045	0.109	0.550	2611
BASELINE	0.008	0.019	0.600	3719
TEXT	0.280	0.492	0.878	170

Table 12: The performance of NB classifiers based on individual metadata fields, as well as a baseline, and the text-only classifier with IGR feature selection.

nary experiments, we considered bag-of-words features for the metadata fields, as well as bag-of-character n-gram features for $n \in \{1, ..., 4\}$.[34] We found character 4-grams to perform best, and report results using these features here. (A bag-of-character 4-grams represents the frequency of each four-character sequence including a start and end symbol.) The geolocation performance of a classifier trained on features from each metadata field in isolation, as well as the performance of a most frequent city baseline (BASELINE) and our best purely text-based classifier (TEXT, replicated from Table 7), is shown in Table 12.

The classifier based on each metadata field outperforms the baseline in terms of Acc, Acc@161, and Median error distance. This suggests these metadata fields do indeed encode geographically-identifying information, though some classifiers are less competitive than TEXT. Notably, despite the potential for noise in the user-supplied location fields, this classifier (LOC) achieves even better performance than the purely text-based method, reaching a city-level accuracy of over 40%, predicting a location within 161km of the true location for over half of the users. This suggests LOC contains valuable information, even though LOC fields are noisy (Cheng et al., 2010), and are not easily captured by off-the-shelf geolocation tools (Hecht et al., 2011). Manual analysis suggests many vernacular place names are captured in the statistical modelling, such as *Kiladelphia* and *Philly* used to represent *Philadelphia*. The utility of metadata fields is also confirmed by the recent work of Priedhorsky et al. (2014).

9.3 Ensemble Learning on Text-Based Classifiers

To further analyse the behaviour of the four metadata classifiers, we consider the pairwise city-level prediction agreement between them. Cohen's Kappa (Carletta, 1996) is a conventional metric to evaluate inter-annotator agreement for categorical items (such as the predicted cities in our case); larger Kappa values indicate higher pairwise agreement. The double fault measure (Giacinto & Roli, 2001) incorporates gold-standard information, and is equal to the proportion of test cases for which both classifiers make a false prediction. This measure offers the empirical lowest error bound for the pairwise ensemble classifier performance.

34. Although we could certainly also consider character n-grams for the text-based classifier, we opted for a bag-of-words representation because it explicitly captures the LIWs that we believe are especially important for geolocation. There could also be location-indicative character n-grams, the exploration of which we leave for future work.

TEXT	0.461	0.689	0.702	0.704
0.181	LOC	0.577	0.578	0.581
0.066	0.063	TZ	0.903	0.907
0.067	0.041	0.085	DESC	0.923
0.065	0.049	0.080	0.088	RNAME

Table 13: Pairwise correlation of the base classifiers using Cohen's Kappa (bottom left, in light grey; higher numbers indicate greater prediction similarity) and the double fault measure (top right, in white; lower numbers indicate greater prediction similarity).

Pairwise scores for Cohen's Kappa and the double fault measure are shown in Table 13. The Kappa scores (bottom-left of Table 13) are very low, indicating that there is little agreement between the classifiers. Because the classifiers achieve better than baseline performance, but also give quite different outputs, it might be possible to combine the classifiers to achieve better performance. The double fault results (top-right) further suggest that improved accuracy could be obtained by combining classifiers.

We combine the individual classifiers using meta-classification. We first adopt a feature concatenation strategy that incrementally combines the feature vectors of TEXT, LOC, TZ, DESC and RNAME. We also consider *stacked generalisation* (Wolpert, 1992), referred to simply as *stacking*, in which the outputs from the base classifiers, and the true city-level locations, are used to train a second classifier which produces the final output. The base classifiers, and the second classifier, are referred to as the *L0* and *L1* classifiers, respectively. In conventional applications of stacking, homogeneous training data is used to train heterogeneous *L0* classifiers; in our case, however, we train homogeneous *L0* multinomial Bayes models on heterogeneous data (i.e., different types of data such as TEXT, LOC, and TZ). We consider logistic regression (Fan, Chang, Hsieh, Wang, & Lin, 2008) and multinomial Bayes as the *L1* classifier.

We carry out 10-fold cross validation on the training users to obtain the *L1* (final) classifier results, a standard procedure for stacking experiments. We use stratified sampling when partitioning the data because the number of users in different cities varies remarkably, and a simple random sample could have a bias towards bigger cities. The ensemble learning results are tabulated in Table 14.

The combination of TEXT and LOC is an improvement over LOC (i.e., our best results so far). However, using feature concatenation and multinomial naive Bayes stacking, accuracy generally drops as metadata feature sets that perform relatively poorly in isolation (i.e., TZ, DESC, RNAME) are incorporated. On the other hand, using logistic regression stacking, we see small increases in accuracy as features that perform less well in isolation are incorporated. Though DESC and RNAME are moderately useful (as shown in Table 12), these fields contribute little to the strong ensembles (i.e., TEXT, LOC and TZ). The best model (using logistic regression stacking and all features) assigns users to the correct city in almost 50% of the test cases, and has a Median error of just 9km. Moreover, with this approach the country-level accuracy reaches almost 92%, indicating the effectiveness of our method for this coarse-grained geolocation task.

Feature concatenation

	Features	Acc	Acc@161	Acc@C	Median
1.	TEXT +LOC	0.444	0.646	0.923	27
2.	1. + TZ	0.429	0.639	0.929	32
3.	2. + DESC	0.319	0.529	0.912	127
4.	3. + RNAME	0.294	0.503	0.912	156

Multinomial Bayes stacking

	Features	Acc	Acc@161	Acc@C	Median
1.	TEXT +LOC	0.470	0.660	0.933	19
2.	1. + TZ	0.460	0.653	0.930	23
3.	2. + DESC	0.451	0.645	0.931	27
4.	3. + RNAME	0.451	0.645	0.931	27

Logistic regression stacking

	Features	Acc	Acc@161	Acc@C	Median
1.	TEXT +LOC	0.483	0.653	0.903	14
2.	1. + TZ	0.490	0.665	0.917	9
3.	2. + DESC	0.490	0.666	0.919	9
4.	3. + RNAME	0.491	0.667	0.919	9

Table 14: The performance of classifiers combining information from text and metadata using feature concatenation (top), multinomial Bayes stacking (middle), and logistic regression stacking (bottom). Features such as “1. + TZ” refer to the features used in row “1.” in combination with TZ.

It is interesting to observe that, while we found NB to outperform LR as a standalone classifier in Section 5.2, as an L1 classifier, LR clearly outperforms NB. The reason for this is almost certainly the fact that we use a much smaller feature set relative to the number of training instances in our stacking experiments, under which circumstances, discriminative models tend to outperform generative models (Ng & Jordan, 2002).

10. Temporal Influence

In addition to the held-out English test data in WORLD+NG, we also developed a new geotagged test dataset to measure the impact of time on model generalisation. The training and test data in WORLD+NG are time-homogeneous as they are randomly partitioned based on data collected in the same period. In contrast, the new test dataset (LIVE) is much newer, collected more than 1 year later than WORLD+NG. Given that Twitter users and topics change rapidly, a key question is whether the statistical model learned from the “old” training data is still effective over the “new” test data? This question has implications for the maintenance and retraining of geolocation models over time. In the experiments in this section we train on WORLD+NG and test on our new dataset.

The LIVE data was collected over 48 hours from 3 Mar, 2013 to 5 Mar, 2013, based on geotagged tweets from users whose declared language was English. Recent status updates (up to 200) were crawled for each user, and `langid.py` was applied to the data to remove any remnant non-English messages. In addition to filtering users with less than 10 geotagged tweets for the test data as in WORLD+NG, we further exclude users with less than 50% of geotagged tweets from one

WORLD+NG

Features	Acc	Acc@161	Acc@C	Median
1. TEXT	0.280	0.492	0.878	170
2. LOC	0.405	0.525	0.834	92
3. TZ	0.064	0.171	0.565	1330
1. + 2. + 3.	0.490	0.665	0.917	9

LIVE

Features	Acc	Acc@161	Acc@C	Median
1. TEXT	0.268	0.510	0.901	151
2. LOC	0.326	0.465	0.813	306
3. TZ	0.065	0.160	0.525	1529
1. + 2. + 3.	0.406	0.614	0.901	40

Table 15: Generalisation comparison between the time-homogeneous WORLD+NG and time-heterogeneous LIVE (1. + 2. + 3. denotes stacking over TEXT, LOC and TZ).

city. This is because if a user's geotagged tweets are spread across different locations, it is less credible to adopt the user's most frequent location as their true primary location in evaluation. A post-check on the WORLD+NG test data shows that 9,977 out of 10K users satisfy this requirement on geographical coherence, and that we aren't unnecessarily biasing the data in LIVE in applying this criterion. Finally, all status updates are aggregated at the user-level, as in WORLD+NG. After filtering, 32K users were obtained, forming the final LIVE dataset.

We use only TEXT, LOC and TZ in this section, as they require less computation and achieve accuracy comparable to our best results, as shown in Table 14. The temporal factor impact on geolocation prediction model generalisation is revealed in the accuracy for WORLD+NG and LIVE shown in Table 15. Acc and Acc@161 numbers in the stacked model (1. + 2. + 3.) drop by approximately 8 and 5 percentage points, respectively, on LIVE as compared to WORLD+NG. The Median prediction error distance also increases moderately from 9km to 40km. By decomposing the stacked models and evaluating against the base classifiers, we find the accuracy declines are primarily caused by accuracy drops in the LOC classifier on the new LIVE data, of approximately 9% in Acc and 6% in Acc@161. This could be viewed as a type of over-fitting, in that the stacked classifier is relying too heavily on the predictions from the LOC base classifier. The TZ classifier performs relatively constantly in terms of accuracy, although the Median error increases slightly. The TEXT classifier is remarkably robust, with all numbers except for Acc improving marginally.

We further investigate the poor LOC classifier generalisation on LIVE. First, we down-sample LIVE to 10K users, the same size as WORLD+NG, and then compare the per-city prediction numbers on the two datasets using only the LOC classifier. We find two factors jointly cause the accuracy decrease on LIVE: (1) the composition of test users, and (2) the decline in per-city recall. For instance, 80 test users are from London, GB in WORLD+NG. This number sharply increases to 155 in LIVE, meaning that the influence of London, GB test users on the overall accuracy in LIVE is almost doubled. Furthermore, the recall — the proportion of users from a given location who are correctly predicted as being from that location — for London, GB drops from 0.676 in WORLD+NG to 0.568 in LIVE. We observe that the proportion of empty LOC fields among London, GB test users jumps from 13% (WORLD+NG) to 26% (LIVE). This reduces the utility of the LOC data in LIVE

Rank	cities in LIVE	LIVE users	LIVE recall	WORLD+NG users	WORLD+NG recall
1	Los Angeles, US	201	0.766	81	0.691
2	Kuala Lumpur, MY	168	0.482	50	0.560
3	London, GB	155	0.568	80	0.675
4	Jakarta, ID	129	0.550	86	0.686
5	Anaheim, US	85	0.447	26	0.346
6	Singapore, SG	76	0.474	160	0.556
7	Fort Worth, US	76	0.289	35	0.371
8	Chicago, US	72	0.569	123	0.577
9	Pittsburgh, US	72	0.431	39	0.487
10	San Antonio, US	66	0.455	82	0.585

Table 16: The number of test users, and recall using LOC, by city, for the top-10 largest cities in LIVE, compared with WORLD+NG.

and explains why the per-city recall drops: all test users with an empty LOC field are assigned to the city with highest class prior in the model (i.e., Los Angeles, US). Overall, the ratios of empty LOC fields in WORLD+NG test data and LIVE are 0.176 and 0.305, respectively, suggesting that user-declared locations in LIVE carry much less geospatial information than in WORLD+NG. We show other comparisons for the top-10 cities in terms of test users in Table 16,[35] as the accuracy of more highly-represented cities has a greater impact on overall results than that of smaller cities. Like London, GB, most cities shown in Table 16 experience lower recall scores for LIVE, and many of them have more test users in LIVE than in WORLD+NG. Nevertheless, some cities have higher recall and more test users in LIVE, e.g., Los Angeles, US and Anaheim, US in Table 16. The overall numbers are, of course, determined by aggregated performance over all cities. To provide some insight, 35.6% of cities in WORLD+NG have more than 40% in recall, but the number is only 28.5% in LIVE.

While an important base classifier in the stacked model, the LOC accuracy numbers are most influenced by temporal changes, whether it is because of an increased reluctance to supply a user-declared location (although admittedly for users who geotag their tweets), or primarily due to variance in user proportions from different cities in the sampled stream. Either way, a periodically retrained LOC classifier would, no doubt, go some way towards remedying the temporal gap. Overall, the numbers suggest that time-homogeneous data (WORLD+NG) is easier to classify than time-heterogeneous data (LIVE). However, training on "old" data and testing on "new" data has been shown to be empirically viable for the TEXT and TZ base classifiers in particular. This result also validates efforts to optimise text-based user geolocation classification accuracy. Recently, similar results on tweet-level geolocation prediction were observed by Priedhorsky et al. (2014), supporting the claim that the accuracy of geolocation prediction suffers from diachronic mismatches between the training and test data.

35. We observe that the city proportions changed drastically between WORLD+NG and LIVE. The reasons for this are unclear, and we can only speculate that it is due to significant shifts in microblogging usage in different locations around the world.

11. User Tweeting Behaviour

Having improved and extended text-based geolocation prediction, we now shift our focus to user geolocatability. If a user wishes to keep their geolocation private, they can simply disable public access of their tweets and metadata. However, if users choose to share their (non-geotagged) tweets, are there different tweeting behaviours which will make them more susceptible to geolocation privacy attacks? To investigate this question, in this section, we discuss the impact of user behaviour on geolocation accuracy relative to predictions over LIVE based on the stacking model from Section 10.[36]

As an obvious first rule of thumb, geotagged tweets should be avoided, because they provide immediate access to a user's geographical footprint, e.g., favourite bars, or their office address. Second, as an immediate implication of our finding that location metadata is a strong predictor of geolocation (Section 9.2), if a user wants to avoid privacy attacks, they should avoid presenting location metadata, in effect disabling the LOC base classifier in our stacked classifier. Third, the text of a user's posts can be used to geolocate the user (at approximately 27% Acc, from Table 15). To investigate the impact of the volume of tweets on user "geolocatability", we perform a breakdown of results over LIVE across two dimensions: (1) the number of LIWs, to investigate whether the sheer volume of tweets from a user makes them more geolocatable; and (2) the source of geospatial information which we exploit in the geolocation model. We evaluate these questions in Figure 6 in four feature combination settings, relative to the: (1) tweet text-based classifier; (2) tweet text-based classifier with gazetteer names removed;[37] (3) metadata stacking using LOC and TZ (invariant to tweet number changes); and (4) the stacking of TEXT, LOC and TZ for all users. In each case, we partition the data into 20 partitions of 5% of users each, ranked by the total number of LIWs contained in the combined posts from that user. In addition to the Acc for each user partition, we also indicate the average number of LIWs per user in each partition (as shown in the second y-axis, on the right side of the graph).

Overall, the more LIWs are contained in a user's tweets, the higher the Acc for text-based methods. When gazetted terms are removed from the tweets, Acc drops by a large margin. This suggests gazetted terms play a crucial role in user geolocation. Metadata also contributes substantially to accuracy, improving the text-based accuracy consistently. Moreover, if a user tweets a lot, the Acc of the tweet text-based approach is comparable to our best model, even without access to the metadata (as shown in the top right corner of the graph). As an overall recommendation, users who wish to obfuscate their location should leave the metadata fields blank and avoid mentioning LIWs (e.g., gazetted terms and dialectal words) in their tweets. This will make it very difficult for our best geolocation models to infer their location correctly (as demonstrated to the bottom left of the graph). A similar conclusion on user geolocatability was recently obtained by Priedhorsky et al. (2014). To help privacy-conscious Twitter users to avoid being geolocated by their tweets, we have made the list of LIWs publicly available.[38]

36. Our analysis is limited to behaviours that could easily be adopted by many users. Given that our system predicts the most likely city from a fixed set for a given user, one simple way to avoid being geolocated is to move far away from any of these cities. However, it seems unlikely that this strategy would be widely adopted.
37. Our gazetteer is based on the ASCII city names in the Geonames data.
38. http://www.csse.unimelb.edu.au/~tim/etc/liw-jair.tgz

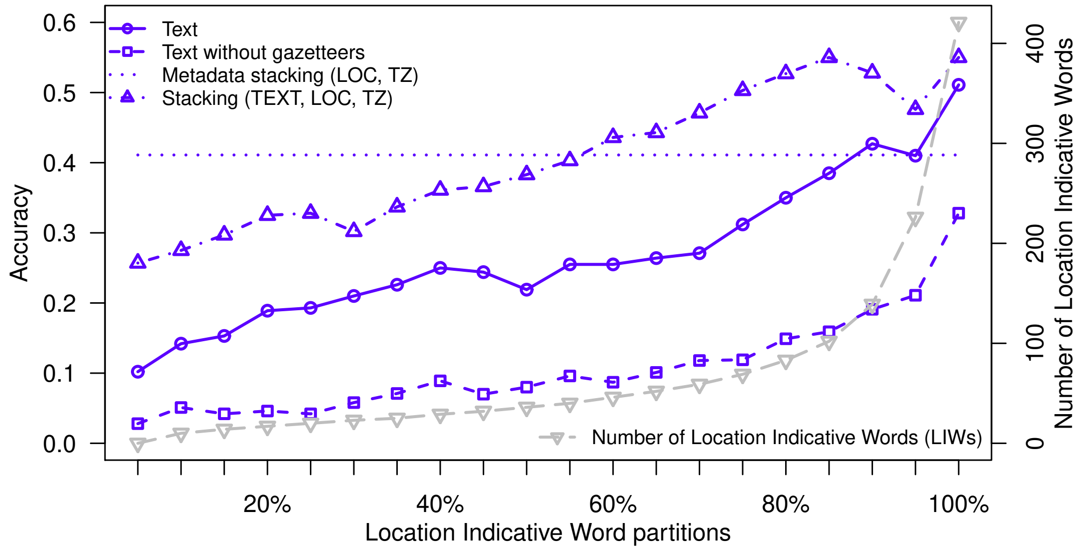

Figure 6: The impact of the use of LIWs on geolocation accuracy. Users are sorted by the number of LIWs in their tweets, and are partitioned into 20 bins. Metadata includes LOC and TZ.

12. Prediction Confidence

In the task setup to date, we have forced our models to geolocate all users. In practice, however, many users don't explicitly mention any geolocating words in their posts, making the task nigh on impossible even for a human oracle. An alternative approach would be to predict a user geolocation only when the model is confident of its prediction. Here, we consider a range of variables that potentially indicate the prediction confidence.

Absolute probability (AP): Only consider predictions with probability above a specified threshold.

Prediction coherence (PC): We hypothesise that for reliable predictions, the top-ranked locations will tend to be geographically close. In this preliminary exploration of coherence, we formulate PC as the sum of the reciprocal ranks of the predictions corresponding to the second-level administrative region in our class representation (i.e., state or province) of the top-ranking prediction, calculated over the top-10 predictions.[39] For example, suppose the top-10 second-level predictions were in the following states in the US: US-TX, US-FL, US-TX, US-TX, US-CA, US-TX, US-TX, US-FL, US-CA, US-NY. The top-ranking state-level prediction is therefore US-TX, which also occurs at ranks 3, 4, 6 and 7 (for different cities in Texas). In this case, PC would be $\frac{1}{1} + \frac{1}{3} + \frac{1}{4} + \frac{1}{6} + \frac{1}{7}$.

Probability ratio (PR): If the model is confident in its prediction, the first prediction will tend to be much more probable than other predictions. We formulate this intuition as PR, the ratio of the probability of the first and second most-probable predictions.

39. It could be measured by the average distance between top predictions as well.

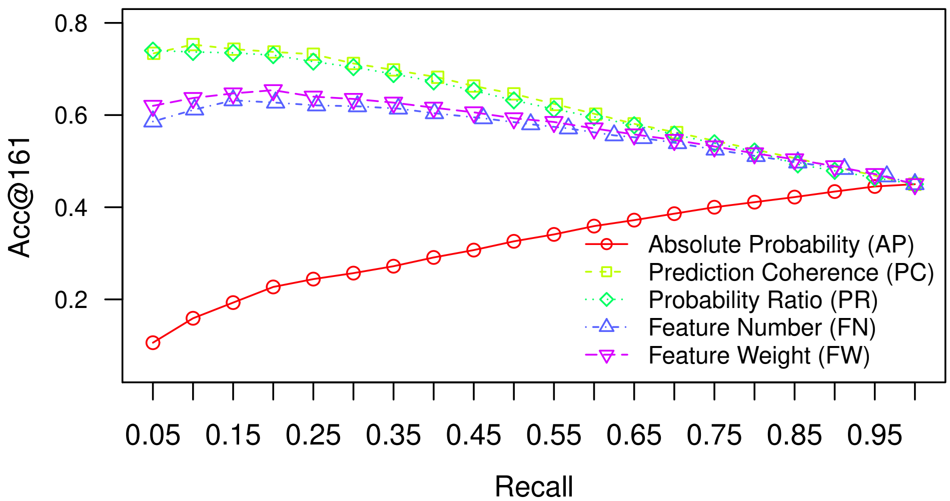

Figure 7: Acc@161 for classification of the top-$n\%$ most-confident predictions for each measure of text-based prediction confidence on NA

Feature number (FN): We take the number of features found in a user's posts as the prediction accuracy. The intuition here is that a geolocation prediction based on more features is more reliable than a prediction based on fewer features.

Feature weight (FW): Similar to FN, but in this case we use the sum of IGR of all features, rather than just the number of features.

We investigate these variables on both NA and LIVE results. In particular, we only evaluate them using the text-based model, as we experiment only with text-based user geolocation in this section. Nevertheless, exploration of other metadata classifiers is also possible. We sort the predictions by confidence (independently for each measure of prediction confidence) and measure Acc@161 among the top-$n\%$ of predictions for the following values of n: $\{0.0, 0.05, ..., 1.0\}$, akin to a precision–recall curve, as shown in Figures 7 and 8. Results on Acc show a very similar trend, and are omitted from the paper.

The naive AP method is least reliable with, surprisingly, accuracy increasing as AP decreases in both figures. It appears that the raw probabilities are not an accurate reflection of prediction confidence. We find this is because a larger AP usually indicates a user has few LIW features, and the model often geolocates the user to the city with the highest class prior. In comparison, PR — which focuses on relative, as opposed to raw, probabilities — performs much better, with higher PR generally corresponding to higher accuracy. In addition, PC shows different trends on the two figures. It achieves comparable performance with PR on NA, however it is incapable of estimating the global prediction confidence. This is largely because world-level PC numbers are often very small and less discriminating than the regional PC numbers, reducing the utility of the geographic proximity of the top predictions. Furthermore, FN and FW display similar overall trends to PR, but don't outperform PR.

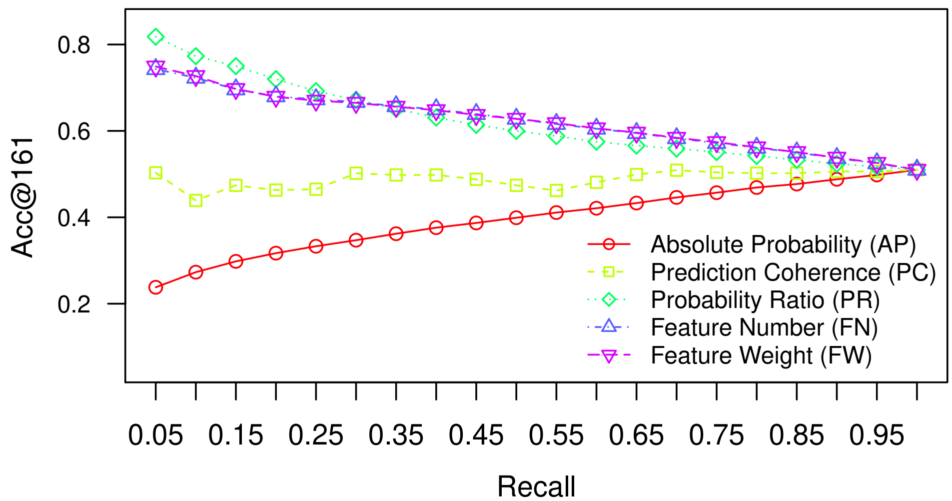

Figure 8: Acc@161 for classification of the top-n% most-confident predictions for each measure of text-based prediction confidence on LIVE

These experiments suggest that there is indeed a trade-off between coverage and accuracy, which could be further exploited to obtain higher-accuracy predictions in applications that do not require all the data to be classified. PR, as well as FN and FW, are fairly effective indicators of predictive accuracy. A further extension on this line of research would be to investigate the prediction confidence per city, e.g., are users from New York, US more predictable than users from Boston, US?

13. Future Work

This research could be expanded in a number of directions. First, hierarchical classification models (Mahmud et al., 2012; Ahmed, Hong, & Smola, 2013) are becoming increasingly popular, and could be combined with our stacked model. Although explicit social network data (e.g., followers) can be non-trivial to retrieve, user interactions can be reconstructed from the content of tweets (e.g., replies, retweets and user mentions: Jurgens, 2013). This implicit network information could be combined with our current text-based geolocation methods to further improve geolocation accuracy. Additionally, we hypothesise that text-based geolocation prediction is a challenging task for humans, and that our method is achieving or surpassing the accuracy levels of a human. It would be interesting to test this hypothesis, e.g., using crowdsourcing methods.

Recently, Priedhorsky et al. (2014) proposed evaluating message-level geolocation. They use Gaussian mixture models to characterise n-gram probability distributions and evaluate the geolocation prediction accuracy using probabilistic metrics. Their conclusions strongly agree with our findings, although our task setting is at the user-level and the evaluation metrics are different. In the future, we plan to adapt our methods to tweet-level geolocation and carry out a systematic evaluation with their probabilistic analysis of geolocation.

14. Summary

In this paper, we have investigated a series of key issues relating to text-based geolocation prediction for Twitter users. We applied a number of feature selection methods to identify location indicative words (LIWs), and demonstrated the effectiveness of feature selection on both regional (NA) and global (WORLD) datasets. We then extended our study to analyse the impact of non-geotagged data, the influence of language and the complementary geographical information in the user metadata. We further evaluated our model on a time-heterogeneous dataset to assess the model's sensitivity to temporal change. Moreover, we discussed how users' tweeting behaviour affects geolocation prediction, and drew conclusions on how users make themselves less easily geolocatable. Finally, we explored various indicators to estimate prediction confidence, in terms of the balance between prediction coverage and accuracy.

A number of conclusions can be drawn from this study, corresponding to the different sections of the paper. We believe these findings contribute to a deeper understanding of text-based geolocation prediction, and further shape the design of practical solutions to the problem:

- We demonstrate that explicit selection of location indicative words improves geolocation prediction accuracy, as compared to using the full feature set.

- Non-geotagged tweets (from users whose location is known) boost the prediction accuracy substantially in both training and testing. We also demonstrate that modeling on geotagged data and inferencing on non-geotagged data is indeed feasible. This is largely because of the similarity between geotagged data and non-geotagged data, although minor differences are observed between geotagged and non-geotagged tweets.

- Modelling and inference on multilingual data is viable and easier than on monolingual English data. This is because tweet language strongly affects the prediction accuracy. Due to the uneven geographical distribution of languages in tweets, users of geographically-diverse languages (e.g., English and Spanish) are much harder to geolocate than users of geographically-focused languages (e.g., Japanese or Dutch). Although trivially determining locations based on the language in tweets is fine for geographically-focused languages, it is insufficient for the majority of users who post tweets using geographically-diverse languages. By integrating language information in different ways, we found training a range of monolingual models based on language identification, and predicting location using a model based on the user's primary language, achieves better results than a monolithic multilingual model.

- User-declared metadata, though noisy and unstructured, offers complementary location-indicative information to what is contained in tweets. By combining tweet and metadata information through stacking, the best global geolocation results are attained: over 49% of English users can be correctly predicted at the city level, with a Median error distance of just 9km.

- Results on time-heterogeneous evaluation suggest applying a model trained on "old" data to predict "new" data is generally feasible. Although the user-declared location field (LOC) is sensitive to temporal change, classifiers based on the tweet content (TEXT) and user timezone (TZ) generalise reasonably well across time.

- Our pilot study on user geolocatability led to the following recommendations to preserve geolocation privacy: (1) reduce the usage of location indicative words, particularly gazetted terms; and (2) delete location-sensitive metadata (e.g., user-declared location and timezone metadata).

- Probability ratio, which measures the ratio of the probability of the top prediction with that of the second prediction, can be used to estimate prediction confidence, and select only users where the system prediction is more accurate, e.g., for downstream applications that require more-reliable geolocation predictions and where exhaustive user geolocation is not required.

Acknowledgments

The authors wish to thank Stephen Roller and Jason Baldridge making their data and tools available to replicate their NA experiments.

NICTA is funded by the Australian government as represented by Department of Broadband, Communication and Digital Economy, and the Australian Research Council through the ICT Centre of Excellence programme.

References

Ahmed, A., Hong, L., & Smola, A. J. (2013). Hierarchical geographical modeling of user locations from social media posts. In *Proceedings of the 22nd international conference on World Wide Web*, WWW '13, pp. 25–36, Rio de Janeiro, Brazil.

Amitay, E., Har'El, N., Sivan, R., & Soffer, A. (2004). Web-a-where: geotagging web content. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004)*, pp. 273–280, Sheffield, UK.

Backstrom, L., Kleinberg, J., Kumar, R., & Novak, J. (2008). Spatial variation in search engine queries. In *Proceeding of the 17th international conference on World Wide Web*, WWW '08, pp. 357–366, Beijing, China.

Backstrom, L., Sun, E., & Marlow, C. (2010). Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 61–70, Raleigh, USA.

Baldwin, T., Cook, P., Lui, M., MacKinlay, A., & Wang, L. (2013). How noisy social media text, how diffrnt social media sources?. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*, pp. 356–364, Nagoya, Japan.

Bennett, P. N., Radlinski, F., White, R. W., & Yilmaz, E. (2011). Inferring and using location metadata to personalize web search. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pp. 135–144, Beijing, China.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communication of the ACM*, *18*(9), 509–517.

Bergsma, S., Dredze, M., Van Durme, B., Wilson, T., & Yarowsky, D. (2013). Broadly improving user classification via communication-based name and location clustering on Twitter. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies (NAACL-HLT 2013), pp. 1010–1019, Atlanta, USA.

Bilhaut, F., Charnois, T., Enjalbert, P., & Mathet, Y. (2003). Geographic reference analysis for geographic document querying. In *Proceedings of the HLT-NAACL 2003 workshop on Analysis of geographic references - Volume 1*, pp. 55–62, Edmonton, Canada.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Buyukokkten, O., Cho, J., Garcia-Molina, H., Gravano, L., & Shivakumar, N. (1999). Exploiting geographical location information of web pages. In *ACM SIGMOD Workshop on The Web and Databases (WebDB'99)*, pp. 91–96, Philadelphia, USA.

Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*, *22*(2), 249–254.

Chandra, S., Khan, L., & Muhaya, F. (2011). Estimating Twitter user location using social interactions–a content based approach. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing (SocialCom)*, pp. 838–843, Boston, USA.

Chang, H.-w., Lee, D., M., E., & Lee, J. (2012). @Phillies tweeting from Philly? predicting Twitter user locations with spatial word usage. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 111–118, Istanbul, Turkey.

Cheng, Z., Caverlee, J., & Lee, K. (2010). 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*, pp. 759–768, Toronto, Canada.

Cho, E., Myers, S. A., & Leskovec, J. (2011). Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1082–1090, San Diego, USA.

Crandall, D. J., Backstrom, L., Huttenlocher, D., & Kleinberg, J. (2009). Mapping the world's photos. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pp. 761–770, Madrid, Spain.

Dalvi, N., Kumar, R., & Pang, B. (2012). Object matching in tweets with spatial models. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM 2012)*, pp. 43–52, Seattle, USA.

Ding, J., Gravano, L., & Shivakumar, N. (2000). Computing geographical scopes of web resources. In *Proceedings of the 26th International Conference on Very Large Data Bases*, VLDB '00, pp. 545–556, Cairo, Egypt.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, *19*(1), 61–74.

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)*, pp. 1277–1287, Cambridge, USA.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., & Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874.

Gelernter, J., & Mushegian, N. (2011). Geo-parsing messages from microtext. *Transactions in GIS*, *15*(6), 753–773.

Giacinto, G., & Roli, F. (2001). Design of effective neural network ensembles for image classification purposes. *Image and Vision Computing*, *19*(9–10), 699–707.

Gouws, S., Metzler, D., Cai, C., & Hovy, E. (2011). Contextual bearing on linguistic variation in social media. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pp. 20–29, Portland, USA.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Han, B., Cook, P., & Baldwin, T. (2012a). Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning 2012 (EMNLP-CoNLL 2012)*, pp. 421–432, Jeju, Korea.

Han, B., Cook, P., & Baldwin, T. (2012b). Geolocation prediction in social media data by finding location indicative words. In *Proceedings of the 24th International Conference on Computational Linguistics*, pp. 1045–1062, Mumbai, India.

Han, B., Cook, P., & Baldwin, T. (2013). A stacking-based approach to Twitter user geolocation prediction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 7–12, Sofia, Bulgaria.

Hauff, C., & Houben, G.-J. (2012). Geo-location estimation of Flickr images: social web based enrichment. In *Proceedings of the 34th European Conference on Advances in Information Retrieval*, pp. 85–96, Barcelona, Spain.

Hecht, B., Hong, L., Suh, B., & Chi, E. H. (2011). 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*, pp. 237–246, Vancouver, Canada.

Hong, L., Ahmed, A., Gurumurthy, S., Smola, A. J., & Tsioutsiouliklis, K. (2012). Discovering geographical topics in the Twitter stream. In *Proceedings of the 21st International Conference on World Wide Web (WWW 2012)*, pp. 769–778, Lyon, France.

Hong, L., Convertino, G., & Chi, E. H. (2011). Language matters in Twitter: A large scale study. In *Proceedings of the 5th International Conference on Weblogs and Social Media (ICWSM 2011)*, pp. 518–521, Barcelona, Spain.

Jurgens, D. (2013). That's what friends are for: Inferring location in online social media platforms based on social relationships. In *Proceedings of the 7th International Conference on Weblogs and Social Media (ICWSM 2013)*, pp. 273–282, Boston, USA.

Kinsella, S., Murdock, V., & O'Hare, N. (2011). "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*, pp. 61–68, Glasgow, UK.

Laere, O. V., Quinn, J., Schockaert, S., & Dhoedt, B. (2014). Spatially-aware term selection for geotagging. *IEEE Transactions on Knowledge and Data Engineering*, *26*(1), 221–234.

Laere, O. V., Schockaert, S., & Dhoedt, B. (2013). Georeferencing Flickr resources based on textual meta-data. *Information Sciences*, *238*, 52–74.

Leidner, J. L., & Lieberman, M. D. (2011). Detecting geographical references in the form of place names and associated spatial natural language. *SIGSPATIAL Special*, *3*(2), 5–11.

Li, R., Wang, S., & Chang, K. C.-C. (2012). Multiple location profiling for users and relationships from social network and content. *VLDB*, *5*(11), 1603–1614.

Li, W., Serdyukov, P., de Vries, A. P., Eickhoff, C., & Larson, M. (2011). The where in the tweet. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM 2011)*, pp. 2473–2476, Glasgow, UK.

Lieberman, M. D., & Lin, J. (2009). You are where you edit: Locating Wikipedia contributors through edit histories. In *Proceedings of the 3rd International Conference on Weblogs and Social Media (ICWSM 2009)*, pp. 106–113, San Jose, USA.

Lui, M., & Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012) Demo Session*, pp. 25–30, Jeju, Korea.

Mahmud, J., Nichols, J., & Drews, C. (2012). Where is this tweet from? Inferring home locations of Twitter users. In *Proceedings of the 6th International Conference on Weblogs and Social Media (ICWSM 2012)*, pp. 511–514, Dublin, Ireland.

Mao, H., Shuai, X., & Kapadia, A. (2011). Loose tweets: an analysis of privacy leaks on Twitter. In *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society*, pp. 1–12, Chicago, USA.

Nakatani, S. (2010). Language detection library for Java. `http://code.google.com/p/language-detection/`.

Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In *Advances in Neural Information Processing Systems 14 (NIPS-02)*, pp. 841–848, Whistler, Canada.

Nocedal, J. (1980). Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, *35*(151), 773–782.

Núnez-Redó, M., Díaz, L., Gil, J., González, D., & Huerta, J. (2011). Discovery and integration of Web 2.0 content into geospatial information structures: a use case in wild fire monitoring. In *Proceedings of the 6th International Conference on Availability, Reliability and Security*, pp. 50–68, Vienna, Austria.

O'Connor, B., Krieger, M., & Ahn, D. (2010). TweetMotif: Exploratory search and topic summarization for Twitter. In *Proceedings of Fourth International AAAI Conference on Weblogs and Social Media*, pp. 384–385, Washington, D.C., USA.

O'Hare, N., & Murdock, V. (2013). Modeling locations with social media. *Information Retrieval*, *16*(1), 30–62.

O'Sullivan, D., & Unwin, D. J. (2010). *Point Pattern Analysis*, pp. 121–155. John Wiley & Sons, Inc.

Padmanabhan, V. N., & Subramanian, L. (2001). An investigation of geographic mapping techniques for internet hosts. In *Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '01, pp. 173–185, San Diego, USA.

Pontes, T., Vasconcelos, M., Almeida, J., Kumaraguru, P., & Almeida, V. (2012). We know where you live: Privacy characterization of Foursquare behavior. In *4th International Workshop on Location-Based Social Networks (LBSN 2012)*, Pittsburgh, USA.

Priedhorsky, R., Culotta, A., & Valle, S. Y. D. (2014). Inferring the origin locations of tweets with quantitative confidence. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work and Social Computing*, Baltimore, USA. To appear.

Quercini, G., Samet, H., Sankaranarayanan, J., & Lieberman, M. D. (2010). Determining the spatial reader scopes of news sources using local lexicons. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, pp. 43–52, San Jose, USA.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, USA.

Ritter, A., Clark, S., Mausam, & Etzioni, O. (2011). Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 1524–1534, Edinburgh, UK.

Roller, S., Speriosu, M., Rallapalli, S., Wing, B., & Baldridge, J. (2012). Supervised text-based geolocation using language models on an adaptive grid. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 1500–1510, Jeju Island, Korea.

Rout, D. P., Bontcheva, K., Preotiuc-Pietro, D., & Cohn, T. (2013). Where's @wally?: A classification approach to geolocating users based on their social ties. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*, pp. 11–20, Paris, France.

Sadilek, A., Kautz, H., & Bigham, J. P. (2012). Finding your friends and following them to where you are. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, pp. 723–732, Seattle, USA.

Schulz, A., Hadjakos, A., Paulheim, H., Nachtwey, J., & Mühlhäuser, M. (2013). A multi-indicator approach for geolocalization of tweets. In *Proceedings of the 7th International Conference on Weblogs and Social Media (ICWSM 2013)*, pp. 573–582, Boston, USA.

Serdyukov, P., Murdock, V., & van Zwol, R. (2009). Placing Flickr photos on a map. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2009)*, pp. 484–491, Boston, USA.

Silva, M. J., Martins, B., Chaves, M. S., Afonso, A. P., & Cardoso, N. (2006). Adding geographic scopes to web resources. *Computers, Environment and Urban Systems*, *30*, 378–399.

Tuten, T. L. (2008). *Advertising 2.0: Social media marketing in a Web 2.0 world*. Praeger Publishers, Westport, USA.

Vapnik, V. N. (1995). *The nature of Statistical Learning Theory*. Springer-Verlag, New York, USA.

Vincenty, T. (1975). Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, *22*(176), 88–93.

Wang, L., Wang, C., Xie, X., Forman, J., Lu, Y., Ma, W.-Y., & Li, Y. (2005). Detecting dominant locations from search queries. In *Proceedings of the 28th Annual International ACM SIGIR*

Conference on Research and Development in Information Retrieval (SIGIR 2005), pp. 424–431, Salvador, Brazil.

Wing, B. P., & Baldridge, J. (2011). Simple supervised document geolocation with geodesic grids. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 955–964, Portland, USA.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*(2), 241–259.

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pp. 412–420, San Francisco, USA.

Yi, X., Raghavan, H., & Leggetter, C. (2009). Discovering users' specific geo intention in web search. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pp. 481–490, Madrid, Spain.

Yin, J., Lampert, A., Cameron, M., Robinson, B., & Power, R. (2012). Using social media to enhance emergency situation awareness. *Intelligent Systems*, *27*(6), 52–59.

Yin, Z., Cao, L., Han, J., Zhai, C., & Huang, T. (2011). Geographical topic discovery and comparison. In *Proceedings of the 20th International Conference on World Wide Web*, pp. 247–256, Hyderabad, India.

Zong, W., Wu, D., Sun, A., Lim, E.-P., & Goh, D. H.-L. (2005). On assigning place names to geography related web pages. In *ACM/IEEE Joint Conference on Digital Libraries*, pp. 354–362, Denver, USA.