"I'm Eating a Sandwich in Hong Kong": Modeling Locations with Tweets

Sheila Kinsella*
Digital Enterprise Research Institute
National University of Ireland
Galway, Ireland
sheila.kinsella@deri.org

Vanessa Murdock
Yahoo! Research
Barcelona, Spain
vmurdock@yahoo-inc.com

Neil O'Hare†
CLARITY: Centre for Sensor Web Technologies
Dublin City University, Ireland
nohare@computing.dcu.ie

ABSTRACT

Social media such as Twitter generate large quantities of data about what a person is thinking and doing in a particular location. We leverage this data to build models of locations to improve our understanding of a user's geographic context. Understanding the user's geographic context in turn allows us to present information, recommend businesses and services, and place advertisements that are relevant at a hyper-local level.

In this paper we create language models of locations using coordinates extracted from geotagged Twitter data. We model locations at varying levels of granularity, from zip code to the country level. We measure the accuracy of these models by the degree to which we can predict the location of an individual tweet, and further by the accuracy with which we can predict the location of a user. We find that we can meet the performance of the industry standard tool for predicting both the tweet and the user, at the country, state and city levels, and far exceed its performance at the hyper-local level, achieving a three- to ten-fold increase in accuracy at the zip code level.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Experimentation

Keywords

geolocation, geotagging, language models, user-generated content, Twitter

*Work done while the author was an intern at Yahoo! Research

†Work done while the author was a visiting scientist at Yahoo! Research

1. INTRODUCTION

Geotagged Twitter[1][2] data affords a window into what people are thinking and doing in a given time and place. Among the microblog updates people post, they discuss events on a local and global level, details of their daily lives, messages intended for specific friends and for the public at large. The commentary spans the spectrum from the most banal "I'm eating a sandwich," to detailed commentary on current political and social trends, all in fewer than 140 characters [3].

The nature of the Twitter data, being a rich and dynamic, albeit succinct, representation of one person's thoughts at a given time and place, make it especially valuable for understanding how people describe the other people, places and events around them. In this paper we build models of places based on the language of tweets. Having a language model of a specific place allows us to understand a user's geographic context, even when the user does not explicitly mention the place name, or allow his geographic coordinates to be known.

Privacy concerns aside, understanding the user's geographic context enables the system to better infer a geographical intent in search queries, place advertisements more appropriately, and to show the user information about events, points of interest, and people around them. As more users interact with the Web via mobile devices, results at the state or country level become less relevant, as users seek information about their immediate vicinity, at the city or neighbourhood levels.

The key to understanding the user's geographic context is to know their current location. Traditionally the user's location is determined by their IP address, in the absence of other information. However IP addresses are unreliable due to VPN networks, which disguise a user's true location, and by dynamic allocation of IP addresses by ISPs such as

[1]http://twitter.com/ visited June 2011

[2]Geotagging refers to associating geographic coordinates to data at the time they are created.

[3]http://www.pearanalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009.pdf visited June 2011

AOL. The degree of unreliability varies, depending on the country, and the desired granularity of the location. By one estimate [9], in the United States, 83% of addresses can be accurately resolved to within 25 miles. For Spain and Ireland, this figure drops to 77% and 61% respectively. On a zip code level, *i.e.* within a mile or two, the accuracy is likely to be much lower. Therefore we require alternate methods to locate a user on a hyper-local level.

Furthermore, the user's current location is only one piece of the puzzle. It does not help to know a person's location if we don't know anything about the location itself. Since geotags are generally automatically assigned by GPS-enabled devices, social media can provide much more fine-grained geographic information, as well as the ability to model the language associated with the location.

In this paper, we create language models of locations using coordinates extracted from geotagged tweets. We model locations at varying levels of granularity, from the zip code to the country level. We measure the accuracy of these models by the degree to which we can predict the location of an individual tweet, and further by the accuracy with which we can predict the location of a user. We find that we can meet the performance of the industry standard tool for predicting the location of both the tweet and the user, at the country, state and city levels, and far exceed its performance at the hyper-local level, achieving a three- to ten-fold increase in accuracy at the zip code level. We begin by reviewing related work in Section 2. In Section 3 we describe how the language of tweets is modeled. In Section 4 we discuss Twitter, and the Yahoo! Geoplanet services. In Section 5 we describe the experimental setup and present the results. We conclude in Section 7.

2. RELATED WORK

Some recent papers have investigated techniques for geolocating Twitter users using models built with tweets originating from known locations. Cheng et al. [3] proposed a probabilistic framework for estimating a Twitter user's city-level location based on tweet content. Their approach does not use geotags from individual tweets, but instead uses place information reported in a user's Location field. The system identifies words that have a local focus, models their geographic distribution, and applies geographic smoothing. Then for each user the city where they are most probably located can be determined, based on their tweet content. For their experiments, city models were built from over 4 million tweets from approximately 131,000 users who have reported United States cities in their Location field. The test set was limited to around 5,000 users who have reported coordinates in their Location field and have 1,000 or more tweets. Their method correctly placed 51% of Twitter users within 100 miles of their correct location. They did not report results on a more local scale and their evaluation considered only cities in the United States.

Eisenstein et al. [5] introduced a method for building models of linguistically consistent regions and thus predicting the location of an author of textual content. The approach involves building a cascading topic model for each topic, and then generating regional variants. Their experiments were limited to authors from the contiguous United States who created 20 or more status updates. The dataset used contains approximately 9,500 users and 380,000 tweets. They defined each user's ground truth location as the location of their first tweet in the sample. The evaluation shows that the system could correctly place users within a mean of 900 kilometres from their correct location, and could identify the correct state of the user in 24% of cases.

Hecht et al. [7] conducted an investigation into usage of the Location field in Twitter, and report on experiments that attempt to predict the home country and state of Twitter users. Their approach uses a Multinomial Naive Bayes model to classify tweets and incorporates an algorithm to bias the models towards terms with a regional focus. Their experiments used a limited dataset of 4 countries, and the state-level experiments were restricted to the United States. They did not make use of geotags but instead obtained location data from the users' Location fields. After filtering out users with less than 10 tweets, a dataset of almost 100,000 users remained. Their approach correctly placed users to their home states with an accuracy of up to 30%. The evaluation found that their models could correctly place the users at a much better accuracy than random, indicating that users implicitly reveal location information in their tweets.

Unlike the previous three works we make use of the geotags of individual tweets to learn language models. The other approaches rely on Location field information or on one geotag per user, which means that the resulting models are less accurate. In addition, we apply our approach to a worldwide dataset and predict locations to as precise a granularity as postal codes. The previous studies were limited to the continental US [3, 5], or only four countries [7]. The work of [3] and [7] reported results at the city level at best. Finally, the previous studies focus on placing users while we place individual tweets as well as users.

Much of the work on predicting locations has been conducted on public geotagged Flickr data[4], which is a rich source of information about locations. Sigurbjörnsson et al. [13] found that over 13% of Flickr tags could be classified as locations using Wordnet. The current work is most related to the work of Serdyukov et al. [12] who use language models to predict the locations of Flickr photos. The models are built from 120,000 photos and the locations are cells from grids of varying size placed on the world map. They are able to correctly place 7% of photos within 1km, and 19% within 10km. In our work, the language modeling approach is similar to the methods described in Serdyukov et al., but the data is substantially different. Furthermore, we take the flexible boundaries defined by zip codes and official neighbourhood and city boundaries, rather than fixed grid cell sizes. This means that our models are based on logical location boundaries rather than arbitrary cells. The work of Serdyukov et al. considered a small-scale data set of only 120,000 geotagged photos with textual metadata. We show results for both small- and large-scale data. Finally we focus on the results at the zip code level, whereas the focus of the work of Serdyukov et al. was on cells of larger sizes, with their best results on cells larger than a typical urban area.

[4]http://www.flickr.com/ visited June 2011

A similar work by Crandall et al. [4] investigates the use of visual, textual and temporal features to classify photos within specific cities. For 100 cities, they identify the top ten landmarks in that city and perform 10-way classification of photos geotagged around these landmarks. Our work differs from these works in that we seek to locate users based on their aggregated tweets, in addition to the media artifacts (in this case tweets) themselves, which are analogous to individual tag sets in the work of both Crandall et al. and Serdyukov et al. While they focus on well-known landmarks in 100 cities, we consider generic locations such as zip codes where no such landmark may be present. We report results of predicting which of ten cities a tweet originated from, but find this is a rather straight-forward task, as the languages used in a given city may be quite distinct from another, even within the same country.

Another attempt to use geotagged images to identify locations is that of Hays and Efros [6] which uses visual features of Flickr photos to predict the location with a nearest-neighbour classification approach. They report correctly placing 16% of photos within 200km, using only the visual characteristics of the photos themselves.

Related work has been done in the area of determining the geographic intent of search queries. Jones et al. [8] provide the first study that relates the user's location to the location mentioned in the query. Yi et al. [14] build language models for U.S. cities from queries with explicit placename mentions. The models are then used to predict the locations associated with queries with no placename mentioned.

3. MODELING THE LANGUAGE OF LOCATIONS

We use the language modeling approach as described in Ponte and Croft [11] to build models of locations. For each location, we estimate a distribution of terms associated with the location. We can then estimate the probability that a tweet was issued from a given location by sampling from the term distribution for that location. We rank the locations by the probability that they "generated" the tweet. More concretely, given a set of locations L, and a tweet T, our goal is to rank the locations by $P(L|T)$. Rather than estimate this directly, we use Bayesian inversion:

$$\begin{aligned} P(L|T) &= \frac{P(T|\theta_L)P(L)}{P(T)} \\ P(T|\theta_L) &= \prod_i P(t_i|\theta_L) \end{aligned} \tag{1}$$

where θ_L is the model of the location, which is a smoothed term distribution of terms associated with the location. In this work we assume the prior probability of the locations, $P(L)$, is distributed uniformly. We ignore $P(T)$ since it is the same for all locations, and thus does not affect the ranking.

The locations can be ranked directly by the probability of having "generated" the tweet, or they can be ranked by comparing the model yielded by the tweet, to the model of the location, using Kullback-Leibler (KL) divergence. In this paper we use both methods for ranking locations.

When ranking by KL divergence, we let θ_T be the language model for the tweet T and θ_L be the language model for the location L. Then the negative divergence from the query language model to the document language model is:

$$KL(\theta_T|\theta_L) = \sum_t p(t|\theta_T) log \frac{p(t|\theta_T)}{p(t|\theta_L)} \tag{2}$$

where t is a term. We smooth the term distribution estimates for the location models using Dirichlet smoothing [15]:

$$p(t|\theta_L) = \frac{c(t,L) + \mu P(t|\theta_C)}{|L| + \mu} \tag{3}$$

where μ is a parameter, set empirically, $c(t,L)$ is the term frequency of a term t for a location L, $|L|$ is the number of terms in the location L, and θ_C is the term distribution over all locations. The term distribution for tweets is smoothed in an analogous way. The KL divergence is smoothed according to:

$$KL(\theta_T|\theta_L) = \sum_t p(t|\theta_T) log \frac{P(t|\theta_T)}{\alpha P(t|\theta_L)} + log(\alpha) \tag{4}$$

where:

$$\alpha = \frac{\mu}{\mu + |L|} \tag{5}$$

In this paper we use the Lemur Toolkit [1] for small-scale experiments, ranking with KL-divergence. To cope with the computational requirements of running large-scale experiments in a time efficient manner, it was necessary to use the Hadoop MapReduce framework, which Lemur does not support. For this reason, we used the Java-based Terrier [10] framework to run experiments on large-scale data. Since Terrier does not support Language Models (it implements an approximation of Dirichlet and Jelinek Mercer language models), we extended the Terrier matching function for retrieval to support Language Models, and implemented a query likelihood Language Model with Dirichlet smoothing.

4. TWITTER AND GEOTAGGING

Twitter. As described in the Introduction, Twitter is a micro-blogging service which allows users to share 140 character messages, also known as statuses and tweets. Users are automatically shown the tweets of other users who they "follow". They can also keep track of conversations by searching for topics or usernames of interest. Status updates can be either publicly available or restricted to a user's connections. Users can make status updates on the Twitter website, or using one of many applications that interface with Twitter.

Twitter has many mobile users, including some who use GPS-enabled devices to geotag their tweets. It is also possi-

ble to allow Twitter to access the browser location information to geotag the tweets. Application developers have two options for attaching geotags to tweets: they can include the latitude and longitude of the tweet, or they use Twitter's reverse geocoding function to include a description of a place, for example at the neighbourhood level. Our analysis makes use of those tweets which are tagged with the user's coordinates.

There is a Twitter-specific syntax which will later be taken into account in building the language models. Tweets can contain mentions of usernames, specified by prefixing a username with an @ symbol as in @exampleuser. Tweets can be tagged with a topic or other annotation, by prefixing a tag with a hash to make a "hashtag" *e.g.* #twitter. Twitter users can also "re-tweet" other user's status updates to relay a message to their own followers, by prefixing a message with "RT @username:", or by clicking a "re-tweet" button.

Table 1 lists the five most commonly occurring sources of geotagged tweets in a 24-hour period. A source is the service such as a website or application from which the user sent the tweet. Some services have the purpose of providing information about the location of the user at the time the tweet was issued. For example, Foursquare (#2) allows users to "check-in" at a venue to win points. A check-in results in the creation of a tweet containing location information such as *"Safe travels! (@ LaGuardia Airport (LGA))"*. Foursquare is the most popular location-oriented Twitter application, and has been used for analysis of user spatio-temporal behaviour [2], but other location-based services provide similar functionality, allowing users to check-in to locations for rewards while simultaneously updating their status.

Service	% of Tweets
UberTwitter	24.2%
Foursquare	18.3%
Twitter for iPhone	12.3%
Twitter for Android	6.3%
Echofon	5.7%

Table 1: Top 5 sources of tweets

Yahoo! Geoplanet. To obtain ground-truth location data, the latitude/longitude coordinates of each geotag are reverse geocoded using Yahoo! Geoplanet[5]. Geoplanet provides identifiers called WOEIDs (Where On Earth IDs) to identify places. Each WOEID corresponds to a unique place, which is described by a centroid and a bounding box, and belongs to of a hierarchy of geographic entities. Therefore for each place it is also possible to retrieve locations which are above that place in the hierarchy. We verify that the bounding box of the region associated with the centroid encompasses the coordinates of the geotag in question. If it does not, we retrieve all neighbours of the region and identify the neighbour which does contain the coordinates. This is necessary when the closest centroid is for a smaller neighbouring location, and the bounding box does not include the coordinates of the tweet, and the bounding boxes do not overlap.

[5]http://developer.yahoo.com/geo/geoplanet/ visited June 2011

Name	Description
Country	A country (in ISO 3166-1 standard)
State	A primary administrative area (state, province, prefecture, or region)
City	A major populated place (city, town, or village)
Neighbourhood	A subdivision within a city (suburb, neighbourhood, or ward)
Zip Code	A zip code or postal code

Table 2: Place types included in our analysis

Yahoo! Placemaker. The Yahoo! Placemaker[6] service identifies and disambiguates places in free text. Like Geoplanet, Placemaker returns results as WOEIDs, and provides access to parent locations in the hierarchy. It is possible to supply a focus WOEID to be used as a search focus when determining the location of the query text. Placemaker determines the geographical scope of a piece of text by extracting mentions of placenames and returning the spatial entity which is most likely to encompass them.

Placemaker is used as a baseline in the experiments of Section 5. Since Placemaker identifies known geographic entities in text, it provides a way to detect explicit geographic information in tweets. We compare our results against Placemaker in order to observe the improvements that can be gained from implicit geographic information. Placemaker has a maximum document of 50,000 characters, so we truncate all queries to this length.

5. EXPERIMENTATION

We conduct analysis on Twitter data from both the publically available, limited stream, and the complete public status stream. We evaluate two tasks. The first predicts which location a single tweet originated from. The second predicts which location a user is in, as evidenced by his tweets aggregated for the entire period. If a user's tweets span several locations, the most frequent location is taken as the ground truth.

5.1 The Data

Data for the experiments was collected from the Twitter status streams. Since the approach is not limited to a particular language, stemming and stopword removal are not performed. Usernames and hashtags are preserved as tokens for building the language models. Any duplicates which occur in the status stream are removed. Re-tweets are removed from the dataset, since they are duplicates or near-duplicates of the original tweet. Finally, all hyperlinks are removed from tweets. Table 3 shows statistics for a sample of the data set sampled in a 24-hour period from the public Twitter stream.

The datasets which we used to evaluate the approach were:

SPRITZER. This dataset was obtained by consuming the generally available public Twitter stream of 5% of all tweets, called the Spritzer. The location filter was used to retrieve all tweets from 10 cities with high Twitter usage. The set of

[6]http://developer.yahoo.com/geo/placemaker/ visited June 2011

Average length (characters)	70.6 ± 40.4
Average length (words)	9.7 ± 6.8
% containing hashtag(s)	11.0%
% containing username(s)	52.0%
% with user location in profile	80.3%
% re-tweets (using official button)	5.4%
% re-tweets (using unofficial syntax)	14.0%
% containing any geotag	0.86%
% containing reverse geocoded place	0.61%
% containing geo coordinates	0.54%

Table 3: Properties of a sample of tweets

cities were chosen to be geographically and linguistically diverse. The cities were, in order of number of tweets: Jakarta (Indonesia), New York (USA), London (UK), Chicago (USA), San Francisco (USA), Houston (USA), Toronto (Canada), Amsterdam (The Netherlands), Sydney (Australia) and Santiago (Chile). This dataset covers the four week time period from May 25th to June 21st, 2010. For each tweet, the corresponding city and neighbourhood are retrieved from Geoplanet.

FIREHOSE. This dataset is from the Twitter Firehose stream, which is the full stream of all public statuses. The original data consists of over 7.3 million tweets posted during summer 2010. Each tweet was reverse geocoded to a country, state, city and zip code, if possible. Table 4 shows the number of unique tweets which could be reverse geocoded to each place type. There are a different number of tweets for each place type since not all tweets which correspond to a country also correspond to a city, for example.

Place Type	Located Tweets	Distinct Places
Country	7,262,002	222
State	7,313,098	2,290
Town	6,295,523	72,617
Zip Code	7,192,172	104,694

Table 4: Number of tweets which can be reverse geocoded to each place type.

5.2 Evaluation measures

For the evaluation, we tune parameters on a held-out set to maximise accuracy, but we additionally report accuracy within an extended geographic range as follows:

Accuracy (Acc). The percentage of correctly predicted locations over all test queries.

Accuracy within N hops (Acc@N). The percentage of predicted locations which lie within N hops of the correct location. For example, for zip code level prediction, Acc@1 measures the percentage of predicted zip codes which are correct, or are direct neighbours of the correct zip code.

5.3 Prediction methods

We report results using the following baseline methods and language model-based methods.

Trivial Classifier (TC). For each tweet, we simply select the most commonly occurring WOEID in the training set.

Placemaker using Location field (PM-L). For each tweet, the user's self-reported location is extracted from their profile and submitted to Placemaker. The most probable candidate location of the appropriate place type is selected. This method allows the detection of explicit geographic references in a user's self-reported location.

Placemaker using Tweet content (PM-T). For each tweet, the content is submitted to Placemaker. The most probable candidate location of the appropriate place type is selected. This method allows the detection of explicit geographic references in the tweet content.

Kullback-Leibler divergence (KL). For each tweet, locations are ranked according to the KL-divergence (Equations 4, 5) between the location model and the tweet model. The location whose model has lowest divergence is selected.

Query Likelihood (QL). For each tweet, locations are ranking according to their query likelihood and the location whose model ranks highest is selected.

5.4 Methods for user location prediction

For user location prediction there may be multiple tweets available. This was not an issue for **TC**, because we simply chose the most commonly occurring location in the training set. For **PM-L**, the Location field generally did not vary, however in the rare cases when it did change, we queried Placemaker for the user's most common location. For **PM-T** and the two language model approaches we experimented with two options for dealing with multiple tweets:

Aggregation (agg.). All tweets are aggregated into one text which is used to determine the user location.

Majority Vote (m.v.). A location is predicted for each tweet and the most frequent one is taken as the user location.

5.5 Results

We present the results for two sets of experiments. The first uses the SPRITZER data set and includes two tasks: prediction of tweet location at the city level, and prediction of tweet location at the neighbourhood level within New York City. The second set of experiments uses the FIREHOSE data set, and attempts to predict the locations of both tweets and users at the zip code, city, state and country levels.

5.5.1 SPRITZER Data

The small-scale SPRITZER experiments were performed using five-fold cross-validation. The data was partitioned at the user level to avoid highly similar tweets by the same user occurring in different partitions.

City-level prediction. We first investigated the performance of our method in classifying tweets on the city level. We built an index with a document corresponding to each of the ten cities in the dataset. We performed a parameter sweep in the range [1, 5, 10, 50, 100, 500, 1,000, 5,000, 10,000] for the Dirichlet prior on a subset of the indexed data and on every round a parameter of 10,000 was found to be optimal.

The results are shown in Table 5. For just over 10% of tweets, a geogaphic reference was detected and disambiguated

by Placemaker (**PM-T**) and the correct parent city was reported. The language model method (**KL**) correctly placed over 65% of tweets. The trivial classifier predicts every tweet originated from Jakarta, which accounts for 40 percent of the tweets.

Method	Acc
TC	0.403 ± 0.013
PM-T	0.108 ± 0.028
KL	0.657 ± 0.011

Table 5: City prediction results for SPRITZER.

Neighbourhood-level prediction. For this experiment we focus on placing tweets within the neighbourhoods of New York City. The SPRITZER dataset contained tweets from 502 New York City neighbourhoods identified by Geoplanet. We performed the same parameter sweep for the Dirichlet prior as in the previous experiment and found 10,000 to be optimal for all splits.

The results are shown in Table 6. Results from Placemaker (**PM-T**) included New York City as a focus location along with the tweet as the query to Placemaker. In 24% of cases, the KL-divergence method (**KL**) returns a neighbourhood within one hop of the correct one, compared to only 4.6% of cases with **PM-T**.

Method	Acc	Acc@1	Acc@2
TC	0.034 ± 0.035	0.075 ± 0.032	0.172 ±0.027
PM-T	0.015 ± 0.001	0.046 ± 0.009	0.081 ± 0.018
KL	0.209 ± 0.018	0.242 ± 0.018	0.290 ± 0.013
QL	0.203 ± 0.017	0.227 ± 0.016	0.268 ± 0.017

Table 6: Neighbourhood prediction results for NYC in the SPRITZER dataset.

In this experiment, we ran both the KL-divergence function provided by Lemur (**KL**) and the Terrier implemented query likelihood method (**QL**) to show the difference in the results. Terrier is used in the large-scale experiments reported in the next section.

5.5.2 *FIREHOSE Data*

For the experiments on large-scale data reported in this section, the data for each placetype was split to have 80% of tweets for building models, 10% for tuning, and 10% for testing. For example, in the country dataset, we reserve approximately 5.8 million tweets for index building, 700k for tuning, and 700k for testing. The data was partitioned by user to avoid highly similar tweets by the same user occurring in different partitions. We tuned the Dirichlet prior for retrieval to a parameter in the range $[1, 5, 10, 50...10^{10}]$, due to the exceptionally large sizes of some pseudo-documents, particularly those representing countries. For the country indexes, the pseudo-document generated for the largest country (USA) was initially too large to index, so a random sample of 80% of all tweets was used to make indexing possible. Also, Geoplanet does not provide information about neighbours of countries, so for country-level experiments only **Acc** is reported. For the other place types, **Acc@1** and **Acc@2** are also reported.

Tweets originating from location-focused services such as Foursquare comprise a large subset of the dataset, and are likely to have a major influence on the results. For the results in Tables 7 and 8 labeled "w/o Location Services", we manually inspected the sources, identified location-focused services which comprise more than 1% of total tweets, and removed these from the test set. This accounts for approximately 25% of tweets. The remaining tweets have fewer explicit references to locations.We retain tweets from location-focused services in our language models.

Tweet location prediction. The results for predicting the location of a tweet are shown in Table 7. The ground truth is the WOEID from which the tweet was posted. The best performing method is always either the language modeling approach (**QL**) or the parsing of the location field using Placemaker (**PM-L**). For zip code prediction, the language modeling approach outperforms all others. At the city level, **QL** and **PM-L** perform similarly when location-based services are included, but **PM-L** performs better when they are omitted. At the state level, **PM-L** gives the highest accuracy and at the country level, both **QL** and **PM-L** perform well. In all cases, the language model method achieves better results than parsing the tweet text for explicit mentions of geographic entities (**PM-T**).

User location prediction. The results for predicting the location of a user are shown in Table 8. The ground truth is the WOEID from which the user most often posts. Again, we show results of both the entire test set, and the test set after tweets from location-focused services have been removed. For **PM-T** and **QL**, we show the results of user location prediction based on both an aggregation of all their tweets (**agg.**), and a majority vote from the individual tweets (**m.v.**).

Comparing the different location prediction methods, we see a similar pattern to the tweet placing experiments. The language model method **QL** outperforms **TC** and **PM-T** in all cases. At a country level, **QL** and **PM-L** give similar results, for states and cities **PM-L** has the highest accuracy, and at a postal code level, **QL** gives the best results. This suggests that using a language model is helpful for placing users at the hyper-local level.

6. DISCUSSION

The small-scale SPRITZER experiments gave promising results for the language modeling approach to tweet location prediction, achieving much better accuracy than either a trivial classifier or prediction based on explicit geographic references found by Placemaker. For the city level results, the success of the language model compared to Placemaker is largely due to the difference in languages spoken between cities. However the results show the advantages of using language information as well as placename knowledge for geographically placing text. The language model approach also gave much better results than the other approaches for neighbourhood prediction within New York City. In this task, there may still be variation in the languages spoken in different regions, but to a lesser extent than for the city prediction task. Thus the good performance of the language modeling approach shows that useful location clues can be found not only from the language used, but also from other regional variations such as mentions of venues or local slang.

	All Tweets			w/o Location Services		
Method	Acc	Acc@1	Acc@2	Acc	Acc@1	Acc@2
Country						
TC	0.469	-	-	0.434	-	-
PM-T	0.222	-	-	0.120	-	-
PM-L	0.528	-	-	**0.518**	-	-
QL	**0.532**	-	-	0.514	-	-
State						
TC	0.063	0.082	0.101	0.060	0.076	0.091
PM-T	0.160	0.170	0.173	0.076	0.081	0.084
PM-L	**0.407**	**0.449**	**0.462**	**0.401**	**0.440**	**0.450**
QL	0.316	0.405	0.458	0.246	0.343	0.400
Town						
TC	0.062	0.062	0.062	0.061	0.061	0.062
PM-T	0.141	0.151	0.153	0.060	0.066	0.067
PM-L	0.269	**0.323**	**0.342**	**0.269**	**0.324**	**0.342**
QL	**0.298**	0.317	0.326	0.217	0.234	0.244
Zip Code						
TC	0.004	0.004	0.004	0.005	0.005	0.005
PM-T	0.025	0.034	0.034	0.018	0.025	0.026
PM-L	0.017	0.025	0.029	0.017	0.025	0.028
QL	**0.139**	**0.166**	**0.188**	**0.052**	**0.073**	**0.094**

Table 7: Results for tweet location prediction on the FIREHOSE dataset.

	All Tweets			w/o Location Services		
Method	Acc	Acc@1	Acc@2	Acc	Acc@1	Acc@2
Country						
TC	0.446	-	-	0.426	-	-
PM-L	0.577	-	-	0.559	-	-
PM-T (agg.)	0.405	-	-	0.295	-	-
PM-T (m.v.)	0.418	-	-	0.295	-	-
QL (agg.)	**0.759**	-	-	**0.710**	-	-
QL (m.v.)	0.501	-	-	0.435	-	-
State						
TC	0.073	0.093	0.118	0.069	0.088	0.110
PM-L	**0.471**	0.507	0.518	**0.447**	**0.482**	**0.493**
PM-T (agg.)	0.276	0.326	0.350	0.185	0.226	0.247
PM-T (m.v.)	0.296	0.334	0.352	0.186	0.219	0.237
QL (agg.)	0.449	**0.541**	**0.589**	0.334	0.436	0.491
QL (m.v.)	0.343	0.412	0.453	0.238	0.313	0.356
Town						
TC	0.034	0.034	0.035	0.031	0.032	0.033
PM-L	0.314	**0.361**	**0.379**	**0.292**	**0.339**	**0.356**
PM-T (agg.)	0.175	0.217	0.236	0.104	0.127	0.139
PM-T (m.v.)	0.218	0.244	0.256	0.114	0.132	0.142
QL (agg.)	**0.319**	0.346	0.362	0.215	0.234	0.249
QL (m.v.)	0.281	0.298	0.308	0.174	0.187	0.196
Postal Code						
TC	0.002	0.002	0.002	0.002	0.002	0.002
PM-L	0.023	0.034	0.038	0.023	0.033	0.037
PM-T (agg.)	0.025	0.045	0.058	0.012	0.020	0.026
PM-T (m.v.)	0.025	0.041	0.051	0.011	0.017	0.022
QL (agg.)	0.135	0.177	**0.213**	0.052	**0.080**	**0.106**
QL (m.v.)	**0.149**	**0.182**	0.207	**0.054**	0.077	0.099

Table 8: Results for user location prediction on the FIREHOSE dataset

The large-scale experiments showed that the benefits of the language modeling approach are most clear at the zip code level. Considering the brevity of tweets and the lack of explicit geographic references in many tweets, the approach achieved promising results. Even though Placemaker could detect geographic entities in only 2.5% of all tweets, our language modeling approach could correctly place 13.9% of them. This indicates that there are substantial benefits to be gained from text features other than explicit geographic references. These benefits are most obvious at the zip code level because users almost never provide their location to such a level of detail, and the tweets themselves are unlikely to mention the types of places of points of interest that would be found in a geographic database. Instead, the tweets may mention local venues, events or terms from local dialects that are not widely-known but can provide valuable information for a language model of that location.

Although the language model approach performed well for zip code prediction, we observe that for the country, state and city level, querying Placemaker for the self-reported location performed best. However this method requires that the user has provided a valid location. Our language model approach could still be useful for the 20% of users who do not report a location (see Table 3), for users whose reported location does not resolve to an identifiable place, and for other applications where a user-provided location is unavailable.

The user placing task showed that accuracy for country, state and city prediction improves on the user level compared to the tweet level, thanks to the additional information provided by multiple status updates. The accuracy for user country prediction after removing tweets from location-focused services is 71% compared to 51% for tweet country prediction. Results for postal code prediction are similar or slightly worse on a user level than on a tweet level. This could be because an individual's tweets are more likely to be dispersed across multiple postal codes than multiple states, for example. It is not clear from these results whether location prediction from a user's tweets should be calculated using aggregation or majority voting. For the language modeling approach, aggregation of tweets into one text gives better results at the city level and above, and when omitting location-based services. At the zip-code level, when location-based services are included, the majority-vote yields slightly better performance.

In addition to user placing, location models from geotagged tweets can provide a rich representation of a place that can be exploited for augmenting location-based services. For example, a system could provide a user with suggestions of the venues that are currently popular in her local area, or could inform a travelling user of the topics that are currently of interest in the city he is visiting. The fact that the language models can be used to place a user with reasonable reliablity indicates that the information contained in the models does provide a useful description of the locations.

7. CONCLUSION

Given the increasing use of mobile devices to access Web-based services, we would like to be able to offer users information about the people, places, events, and services in their direct vicinity. We show in this paper that by leveraging the language of Twitter status updates, we can achieve a higher degree of accuracy in predicting the origin of a tweet at the hyper-local level than with standard tools. Similarly, we find that we can pinpoint the user within two zip codes 20% of the time, compared to 4% of the time with standard tools.

In future work we improve the modeling of locations described in this work with evidence from other sources of user-generated content. We also leave to future work employing these models in applications such as inferring geographic intent, disambiguating location mentions, and point-of-interest discovery.

8. REFERENCES

[1] J. Allan, J. Callan, K. Collins-Thompson, W. B. Croft, F. Feng, D. Fisher, J. Lafferty, L. Larkey, T. N. Truong, P. Ogilvie, L. Si, T. Strohman, H. Turtle, and C. Zhai. The Lemur toolkit for language modeling and information retrieval, 2005. http://www.cs.cmu.edu/ lemur.

[2] C. M. Anastasios Noulas, Salvatore Scellato and M. Pontil. An empirical study of geographic user activity patterns in Foursquare. In *ICWSM*, 2011.

[3] Z. Cheng, J. Caverlee, and K. Lee. You are where you tweet: A content-based approach to geo-locating Twitter users. In *CIKM*, 2010.

[4] D. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *WWW*, 2009.

[5] J. Eisenstein, B. O'Connor, N. A. Smith, and E. Xing. A latent variable model for geographic lexical variation. In *EMNLP*, 2010.

[6] J. Hays and A. Efros. IM2GPS: Estimating geographic information from a single image. In *CVPR*, 2008.

[7] B. Hecht, L. Hong, B. Suh, and E. Chi. Tweets from Justin Bieber's heart: The dynamics of the "location" field in user profiles. In *CHI*, 2011.

[8] R. Jones, W. Zhang, B. Rey, P. Jhala, and E. Stipp. Geographic intention and modification in Web search. *International Journal of Geographical Information Science*, 22(3):229–246, 2008.

[9] I. MaxMind. GeoIP City Accuracy for Selected Countries, May 2010. http://www.maxmind.com/app/city_accuracy.

[10] I. Ounis, C. Lioma, C. Macdonald, and V. Plachouras. Research directions in Terrier: a search engine for advanced retrieval on the Web. *Novatica/UPGRADE Special Issue on Web Information Access*, 2007.

[11] J. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR*, 1998.

[12] P. Serdyukov, V. Murdock, and R. van Zwol. Placing Flickr photos on a map. In *SIGIR*, 2009.

[13] B. Sigurbjörnsson and R. Van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW*, 2008.

[14] X. Yi, H. Raghavan, and C. Leggetter. Discovering users' specific geo intention in Web search. In *WWW*, 2009.

[15] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR*, 2001.