Geolocation for Twitter: Timing Matters

Mark Dredze[1,2], **Miles Osborne**[1], **Prabhanjan Kambadur**[1]

[1] Bloomberg L.P.
731 Lexington Ave, New York, NY 10022

[2] Human Language Technology Center of Excellence
Johns Hopkins University, Baltimore, MD 21211
mdredze@cs.jhu.edu mosborne29,pkambadur@bloomberg.net

Abstract

Automated geolocation of social media messages can benefit a variety of downstream applications. However, these geolocation systems are typically evaluated without attention to how changes in time impact geolocation. Since different people, in different locations write messages at different times, these factors can significantly vary the performance of a geolocation system over time. We demonstrate cyclical temporal effects on geolocation accuracy in Twitter, as well as rapid drops as test data moves beyond the time period of training data. We show that temporal drift can effectively be countered with even modest online model updates.

1 Introduction

Geolocation – the task of identifying a social media message's location – can support a variety of downstream applications, such as advertising, personalization, event discovery, trend analysis and disease tracking (Watanabe et al., 2011; Hong et al., 2012; Kulshrestha et al., 2012; Broniatowski et al., 2013). Geolocation work has mostly focused on Twitter, since tweets are readily accessible and true location available from user geocoded tweets (*inter alia* (Eisenstein et al., 2010; Han et al., 2014; Rout et al., 2013; Compton et al., 2014; Cha et al., 2015; Jurgens et al., 2015; Osborne et al., 2014; Dredze et al., 2013)).

Most previous work consider the task of author geolocation, the identification of a author's primary (home) location (Eisenstein et al., 2010; Han et al., 2014). Author geolocation systems rely on multiple tweets from each author to identify the location. In this work, we consider the task of tweet geolocation, where a system identifies the location where a single tweet was written (Osborne et al., 2014; Dredze et al., 2013). This approach is necessary when geolocation decisions must be made quickly, with limited resources, or when the location of a specific tweet is required.

When focusing on a single tweet, time becomes relevant. Intuitively, tweets written in the morning might be in different locations (at home) than say tweets written during the day (at work). This information is often ignored but can provide important clues as to a tweet's location. Likewise, models built using historical data never adapt as time evolves. These factors may have a significant impact on geolocation accuracy, and downstream system's should be sensitive to these variations.

For the first time, we consider the impact of time on Twitter geolocation and predict where a post was made (rather than the more usual, and easier task of author location). We take a supervised learning approach, training a multi-class classifier to identify the city of a tweet. We train a system on 250 million tweets sampled from a 45 month period, perhaps the largest evaluation to date. We find that:

- Geolocation accuracy is cyclical, varying significantly with time.
- While access to massive training data improves accuracy, these effects are largely lost when models are deployed on new tweets, in large part due to new users and duplicate tweets.
- Periodically updating geolocation models, even with data available from the free Twitter API,

Proceedings of NAACL-HLT 2016, pages 1064–1069,
San Diego, California, June 12-17, 2016.

can largely supplant massive training datasets.

Our study is similar to that of Pavalanathan and Eisenstein (2015), who called into question the accuracy of geolocation models due to mismatches between the behavior of users in available training data as compared to users encountered in live data. While our work provides a cautionary tale, it provides a guide for how these models can be used in practice.

2 Dataset

We start with *every geocoded tweet* (based on the "location" field) from January 1, 2012 to September 30, 2015: 8,530,693,792 tweets.[1] These tweets are associated with a specific location by Twitter (the "location" field is populated.)

We took several steps to remove tweets that were not relevant to the task. We removed tweets posted by location sharing services (FourSquare and jSwarm) since these are not written by users. We removed retweets for the same reason. We also remove tweets that do not have a specific latitude/longitude (geo) while nevertheless containing a location. Twitter allows user's to tag a tweet with a location (populating the location field) even when the user's device does not provide a latitude and longitude (geo field). To ensure we know the precise location of the user we only consider tweets with the geo field.

We matched each tweet to a city using the procedure of Han et al. (2014), with 3,709 cities derived from the geonames database[2]. Only 2983 locations contained a tweet; locations without tweets were mostly in Africa and China, which has low Twitter usage. Following Han et al. (2014) we focus on English tweets only, removing non-English tweets based on the metadata language code. We also identified the tweet's country for a country prediction task (161 labels). We divided this dataset into two time periods. We use tweets from January 1, 2012 to March 30, 2015 for a standard train/dev/test evaluation, selecting $\frac{2}{10{,}000}$ of the data for development and test sets. Data from March 31, 2015 to September 30, 2015 forms an "out of time" sample.

The most common cities were Los Angeles, London, Jakarta, Chicago, Kuala Lumpur and Dallas. The city clustering procedure of Han et al. (2014) greatly influences this list. For example, Los Angeles ends up as one large city, whereas the New York City area is divided into several smaller cities.

[1] Data is available from third party resellers, such as *Gnip*.

[2] `http://www.geonames.org/`

3 Geolocation Model

We treat geolocation as a multi-class task, with each city (or country) a label (Jurgens et al., 2015).

Features All of our features are extracted from a single tweet (text or metadata) without requiring additional queries to the Twitter API. [3] These include: **Text**: We extracted unigrams and bigrams from the text of each tweet after tokenizing with Twokenizer (O'Connor et al., 2010). We removed all punctuation, and replaced unique usernames and urls with placeholder tokens. Numbers were replaced with a NUM token. **Profile location**: Unigrams and bigrams extracted from the user supplied profile location field, as well as a feature for the entire location string. These fields often provide clues as to the user's location, e.g. "New York Living". **Timezone**: Each tweet has a timezone that reflects a specific location, e.g. "Pacific Time (US & Canada)", "Atlantic Time (Canada)", "Casablanca". We also include the UTC offset of the timezone. **Time**: We use a feature indicating the hour of the day (in UTC time) at which the tweet was posted.

Learning We used vowpal wabbit (version 8.1.1) (Agarwal et al., 2011), a linear classifier trained using stochastic gradient descent with adaptive, individual learning rates (Duchi et al., 2011) that minimizes the hinge loss. We used feature hashing with a 31-bit feature space. We selected the best model and parameters based on initial tests using *development data*. All other parameters used default settings.

[3] Our reliance on text features created a very large feature space, but only a small fraction of these occur with any regularity. Previous work has shown feature selection helpful for geolocation (Han et al., 2014). We tried L1 regularization for feature selection without a significant change to our results. It may be that our larger volume of training data removes the need for feature selection. Alternatively, we use feature hashing (to a 31-bit feature space) which can be a form of regularization as feature collisions mitigate overfitting (Ganchev and Dredze, 2008; Weinberger et al., 2009).

4 Evaluation

We report the four evaluation metrics of Han et al. (2014): city accuracy (AccCi), country accuracy (AccCo), accuracy within 161 km (100 miles) (Acc@161), and the median error in km (Median).

Baselines We include two baselines: (1) the majority predictor: always predicts the most popular label. (2) alias matching: we create a list of aliases for each of the 2983 cities from the genomes dataset, which includes the smaller cities clustered together by Han et al. (2014). We search each tweet and the user's profile location for these aliases, assigning a tweet with a matched alias to the corresponding city; unmatched tweets are assigned the majority label. When multiple cities match a tweet, we selected the correct one (if present) using oracle knowledge. About 90% of matches were in the profile. This strategy is similar to that of Dredze et al. (2013).

Duplicates A tweet may be duplicated in our dataset, appearing in both training and held out data, or appearing multiple times in held out data. We define duplicates as tweets with identical feature representations. We removed duplicates from dev and test splits, to ensure evaluation examples are unseen in training, yielding 22,966 dev and 23,240 test tweets.

5 Baseline Results

We begin by establishing the models' performance with a large training set, as measured on held out evaluation data drawn from the same time period. Here we use a standard setting, where there is no online adaptation. We include results for city and country models trained with the tweet text features alone (content). These evaluations train with a sample of 25,822,353 tweets, similar to previous large scale training for geolocation (Han et al., 2014).

Table 2 shows our model beating both baselines, with the additional features generally improving over content features alone. Interestingly, improvements from adding features appears to be additive: the final model's accuracy is nearly the sum of the individual improvements from each feature set. On the non-deduped test dataset (25,941 tweets), the accuracy was higher (city: 0.2920, country: 0.8777) but the trends of adding features remain unchanged. Our time feature, which captures a temporal prior over locations, does not seem to help, providing only a small boost.

We consider the impact of training data size in Figure 1, including a model trained on 258,222,490 tweets, an order of magnitude larger than Han et al. (2014), which improves accuracy by roughly 3%. This figure provides guidance on how much data is necessary to do well on this task.

To summarize: our approach yields tweet level geolocation accuracy similar to, or better than, state of the art user level geolocation.[4] We note that for small datasets (tens of millions of training examples, which can be obtained from the Twitter streaming API), one can obtain a reasonable model.

6 Temporal Factors in Geolocation

We now consider factors that influence geolocation temporal accuracy using our largest city model (258M training tweets), which has an accuracy of 0.3302 on test data (0.3062 excluding duplicates).

6.1 Question 1: How do daily and weekly patterns impact geolocation accuracy?

Twitter traffic varies over the course of a day and a week. User behavior may change at different times, and different locations are active at different times.

Figure 3 shows the number of tweets and test geolocation accuracy by the hour of the day (b) and day of the week (c). The day of the week has a minor impact on geolocation accuracy; the standard deviation of the 7 days is 2.7% of the total mean. Tweet volume has a negative correlation with accuracy (-0.435), i.e. more tweets may be indicative of more people from different locations tweeting, which makes the task harder. Notably, Monday is significantly harder, with an accuracy of 1.5 standard deviations below the mean. However, the hour of the day has much more significant impact on accuracy; some times of the day are significantly easier and harder than the average. The standard deviation is 6.8% of the mean, and tweet volume is strongly negatively correlated with accuracy (-0.647). Geolocation is easier during times when there are fewer locations actively tweeting. This is most apparent during

[4] Direct comparisons are not possible because of different datasets and tasks. However, our results are on par with the user-level geolocation system of Han et al. (2014).

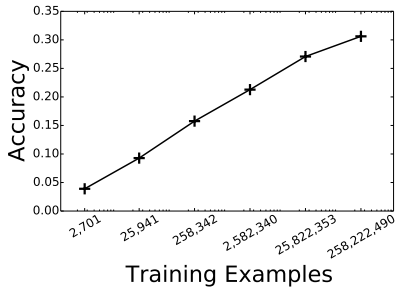
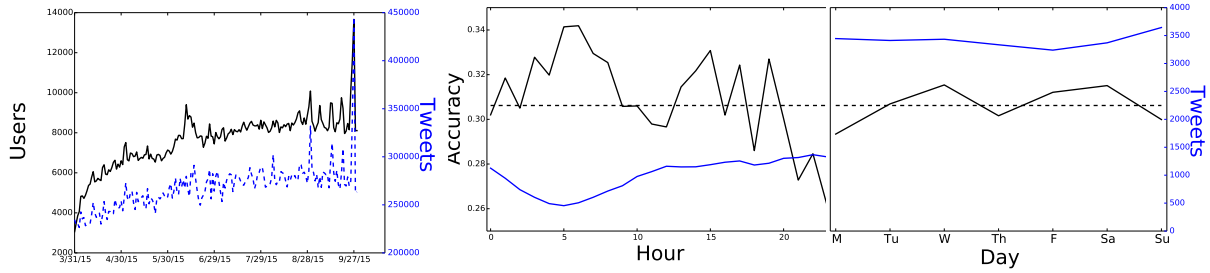


Figure 1: Varying training data size.

Model		AccCi	AccCo	Acc@161	Median	AccCo
		City				Country
Baselines:	Majority	0.0209	0.6410	0.0402	3582	0.6363
	Alias Match	0.1923	0.7317	0.2096	3169	0.7253
Features:	Content	0.0259	0.4093	0.0602	3216	0.4285
	+ Profile	0.2120	0.5609	0.2917	1659	0.7537
	+ Timezone	0.0415	0.5682	0.0974	1690	0.5273
	+ Time	0.0279	0.4282	0.0598	3074	0.4142
	All features	0.2708	0.5861	0.3612	1008	0.8734

Figure 2: Results for different features sets on test data from the same time period as training data for both city and country prediction tasks.



(a) (b) (c)

Figure 3: (a) New users and tweets each day. Accuracy and number of tweets by hour (US eastern) (b) and day (c).

the nighttime in the US, where there are much fewer tweets overall and many fewer active locations. In short, the accuracy of a geolocation system depends on when it is running.

6.2 Question 2: How do changes over time impact a fixed geolocation model?

We now turn to our data sample taken after the training data: a 10% sample of 49,307,720 tweets from 2015/3/31 - 2015/9/30.[5] These tweets will demonstrate the accuracy of a trained model deployed on new data over time.

Evaluating on these tweets (duplicates included), our model yields an accuracy of 0.2661, down from 0.3302, a 19% relative drop. Surprisingly, this isn't a gradual change over time; the drop is quite rapid. The week immediately following the training period has an accuracy of 0.2884. Figure 4 shows the decline in accuracy over time.[6]

What factors contribute to this rapid drop? We consider two: new users and reposted tweets.

New Users One factor affecting geolocation performance might be new users joining, posting a few tweets and then no longer posting. In a sense, users have a temporal lifespan, after which information originating from them is of less predictive value. One measure of this is the number of users encountered in the evaluation data, which have never been previously encountered, either in training or earlier in the evaluation data. Over the six month evaluation period, the number of new tweets from geocoded users per day *increases*, even as a percentage of all tweets (Figure 3(a)).

We remove all tweets in the evaluation period from users that we have previously encountered, either in training or earlier in evaluation data. Accuracy drops to 0.1859, a 30% relative decrease from 0.2661, suggesting that the training data learns features specific to the users it observes. By comparison, the alias match baseline has an accuracy of 0.2113 on this data.

While trained models remain effective on users

[5]While training data is taken from the first 39 months, it is biased towards more recent months due to Twitter growth: the last 12 months (30% of the time) account for 37% of tweets. We evaluated with a 10% sample for efficiency.

[6]While accuracy continues to degrade over time, it begins to rise in August 2015. It may be that there are seasonal effects in geolocation accuracy, or recent changes by Twitter are making geolocation easier. However, we were unable to determine the source of this change.

present in training, it has difficulty generalizing to new users. Far from a small percentage of the total, new users make up a significant number of tweets, at a rate that does not appear to be slowing.

Reposted Tweets Users often repost content, which can include repeating simple message (e.g. "feeling good!") or tweeting the same content to multiple users. Users are more likely to repost content shortly after it was first created, making the number of reposts go down over time. For example, while 8% of test tweets from the same time period as training data are duplicates (they appear in the training data), only 3.8% of tweets in the six month evaluation period are duplicates.

How much of an impact do these reposts have on accuracy? For the test data from the same time period, we saw model performance drop from 0.3302 to 0.3062, a fairly large difference. By comparison, removing reposts in the the six month evaluation period drops accuracy from 0.2661 to 0.2541, a more modest change. Reposts help to inflate geolocation accuracy, and their decrease as time progresses from training removes this accuracy inflation.

7 Question 3: Can periodic model updates maintain a trained geolocation system?

Our results so far are sobering: shortly after a static model is deployed performance degrades to a model using *two orders of magnitude less training data* (compare the drop in §6.2 with Figure 1). Increasing the amount of training data might be an option, but given our previous results on new users, etc., this is unlikely to be sufficient.

A simple method for addressing model degradation over time is to continuously update the model over time using online learning on new data as it becomes available. For example, we can continuously download a stream of (at least) 1% of geocoded tweets from the Twitter API to use as training for updating a deployed system. What is the impact on a system's accuracy when it is updated on these geocoded tweets with SGD updates (§3)?

Figure 4 shows the performance of our system in an online setting (dashed black line). This model updates on every 100th example (1% of all geocoded tweets) encountered in the six-month evaluation period. When we update this previously trained static model, we see a quick recovery to accuracy levels that meet or exceed those on the test set from the same time period as training (horizontal line.)

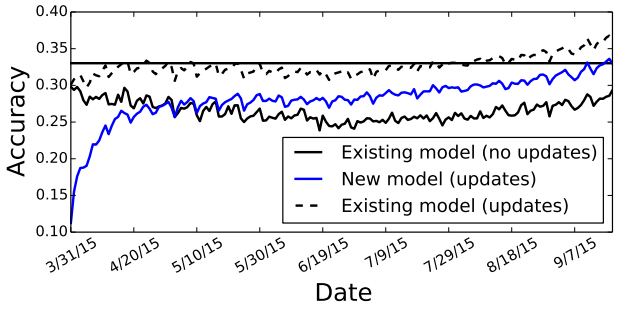


Figure 4: Accuracy over the six months following training. The horizontal line reflects the existing model's performance on test from the same time period as training.

Finally, we consider the case where a practitioner starts from scratch with no training data, but updates using just 1% of geocoded tweets. Can someone with access to no prior training data build an effective model? Encouragingly, within 20 days the new model (solid blue line) catches the previously trained static model (solid black line, "Existing model: no updates"). This is an extremely promising result as it suggests that most practitioners **who do not have access to all geolocated data** can produce geolocation prediction models that approximate models trained using hundred of millions of examples.

8 Conclusion

We have presented a tweet geolocation system that considers an order of magnitude more data than any prior work. Despite hundreds of millions of training examples, the resulting system is sensitive to the time the tweet was authored. Additionally, accuracy suffers when deployed on data beyond the training period. We show that online updates can mitigate problems caused by concept drift. In short, sheer volume of data is not enough: geolocation models should adapt to new data. Encouragingly, starting from no training data and updating on just 1% of geocoded tweets, within 20 days we can recover a model that catches a static model previously trained on hundreds of millions of tweets.

Acknowledgments We thank Bo Han and Tim Baldwin for their help in reproducing their city labels.

References

Alekh Agarwal, Olivier Chapelle, Miroslav Dudík, and John Langford. 2011. A reliable effective terascale linear learning system. *CoRR*, abs/1110.4198.

David Broniatowski, Michael J. Paul, and Mark Dredze. 2013. National and local influenza surveillance through twitter: An analysis of the 2012-2013 influenza epidemic. *PLOS ONE*, December 9.

Miriam Cha, Youngjune Gwon, and HT Kung. 2015. Twitter geolocation and regional classification via sparse coding. In *Ninth International AAAI Conference on Web and Social Media*.

Ryan Compton, David Jurgens, and David Allen. 2014. Geotagging one hundred million twitter accounts with total variation minimization. In *Big Data (Big Data), 2014 IEEE International Conference on*, pages 393–401. IEEE.

Mark Dredze, Michael J Paul, Shane Bergsma, and Hieu Tran. 2013. Carmen: A twitter geolocation system with applications to public health. In *AAAI Workshop on Expanding the Boundaries of Health Informatics Using AI (HIAI)*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*.

Jacob Eisenstein, Brendan O'Connor, Noah A Smith, and Eric P Xing. 2010. A latent variable model for geographic lexical variation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kuzman Ganchev and Mark Dredze. 2008. Small statistical models by random feature mixing. In *Proceedings of the ACL08 HLT Workshop on Mobile Language Processing*, pages 19–20.

Bo Han, Paul Cook, and Timothy Baldwin. 2014. Text-based twitter user geolocation prediction. *Journal of Artificial Intelligence Research*, pages 451–500.

Liangjie Hong, Amr Ahmed, Siva Gurumurthy, Alexander J Smola, and Kostas Tsioutsiouliklis. 2012. Discovering geographical topics in the twitter stream. In *Proceedings of the 21st international conference on World Wide Web*, pages 769–778. ACM.

David Jurgens, Tyler Finethy, James McCorriston, Yi Tian Xu, and Derek Ruths. 2015. Geolocation prediction in twitter using social networks: A critical analysis and review of current practice. In *Proceedings of the 9th International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Juhi Kulshrestha, Farshad Kooti, Ashkan Nikravesh, and P Krishna Gummadi. 2012. Geographic dissection of the twitter network. In *ICWSM*.

Brendan O'Connor, Michel Krieger, and David Ahn. 2010. Tweetmotif: Exploratory search and topic summarization for twitter. In *International Conference on Weblogs and Social Media (ICWSM)*.

Miles Osborne, Sean Moran, Richard McCreadie, Alexander Von Lunen, Martin D Sykora, Elizabeth Cano, Neil Ireson, Craig Macdonald, Iadh Ounis, Yulan He, et al. 2014. Real-time detection, tracking, and monitoring of automatically discovered events in social media. In *Association for Computational Linguistics (ACL)*.

Umashanthi Pavalanathan and Jacob Eisenstein. 2015. Confounds and consequences in geotagged twitter data. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2138–2148, Lisbon, Portugal, September. Association for Computational Linguistics.

Dominic Rout, Kalina Bontcheva, Daniel Preoţiuc-Pietro, and Trevor Cohn. 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*, pages 11–20. ACM.

Kazufumi Watanabe, Masanao Ochi, Makoto Okabe, and Rikio Onai. 2011. Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2541–2544. ACM.

Kilian Weinberger, Anirban Dasgupta, John Langford, Alex Smola, and Josh Attenberg. 2009. Feature hashing for large scale multitask learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1113–1120. ACM.