@Phillies Tweeting from Philly? Predicting Twitter User Locations with Spatial Word Usage

Hau-Wen Chang*, Dongwon Lee*, Mohammed Eltaher† and Jeongkyu Lee†
* The Pennsylvania State University, University Park, PA 16802, USA
Email: {hauwen|dongwon}@psu.edu
† University of Bridgeport, Bridgeport, CT 06604, USA
Email: {meltaher|jelee}@bridgeport.edu

***Abstract*—We study the problem of predicting home locations of Twitter users using *contents* of their tweet messages. Using three probability models for locations, we compare both the Gaussian Mixture Model (GMM) and the Maximum Likelihood Estimation (MLE). In addition, we propose two novel unsupervised methods based on the notions of *Non-Localness* and *Geometric-Localness* to prune noisy data from tweet messages. In the experiments, our unsupervised approach improves the baselines significantly and shows comparable results with the supervised state-of-the-art method. For 5,113 Twitter users in the test set, on average, our approach with only 250 selected local words or less is able to predict their home locations (within 100 miles) with the accuracy of 0.499, or has 509.3 miles of average error distance at best.**

I. Introduction

Knowing users' home locations in social network systems bears an importance in applications such as location-based marketing and personalization. In many social network sites, users can specify their home locations along with other demographics information. However, often, users either do not provide such geographic information (for laziness or privacy concern) or provide them only in inconsistent granularities (e.g., country, state, or city) and reliabilities. Therefore, recently, being able to automatically uncover users' home locations using their social media data becomes an important problem. In general, finding the geographic location of a user from the user-generated contents (that are often a mass of seemingly pointless conversations or utterances) is challenging. In this paper, we focus on the case of Twitter users and try to predict their city locations based on only the *contents* of their tweet messages, without using other information such as user profile metadata or network features. When such additional information is available, we believe one can estimate user locations with a better accuracy and will leave it as a future work. Our problem is formalized as follows:

Problem 1 *For a user u, given a set of his/her tweet messages $T_u = \{t_1, ..., t_{|T_u|}\}$, where t_i is a tweet message up to 140 characters, and a list of candidate cities, C, predict a city c ($\in C$) that is most likely to be the home location of u.* ∎

Intuition behind the problem is that geography plays an important role in our daily lives so that word usage patterns in Twitter may exhibit some geographical clues. For example, users often tweet about a local shopping mall where they plan to hang out, cheer a player in local sports team, or discuss local candidates in elections. Therefore, it is natural to take this observation into consideration for location estimation.

A. Related Work

The location estimation problem which is also known as *geolocating* or *georeferencing* has gained much interests recently. While our focus in this paper is on using "textual" data in Twitter, a similar task using multimedia data such as photos or videos (along with associated tags, description, images features, or audio features) has been explored (e.g., [11], [12], [6], [8]). Hecht et al [5] analyzed the user location filed in user profile and used Multinomial Naive Bayes to estimate user's location in state and country level.

Current state-of-the-art approach, directly related to ours, is by [3] that use a probabilistic framework to estimate city-level location based on the contents of tweets without considering other geospatial clues. Their approach achieved the accuracy on estimating user locations within 100 miles of error margin, (at best) varying from 0.101 (baseline) to 0.498 (with local word filtering). While their result is promising, their approach requires a manual selection of local words for training a classification model, which is neither practical nor reliable. A similar study proposed by [2] took a step further by taking the "reply-tweet" relation into consideration in addition to the text contents. [7] approached the problem with a language model with varying levels of granularities, from zip codes to country levels. [4] studied the problem of matching a tweet to an object from a list of objects of a given domain (e.g., restaurants) whose geolocation is known. Their study assumes that the probability of a user tweeting about an object depends on the distance between the user's and the object's locations. The matching of tweets in turn can help decide the user's location. [9] studied the problem of associating a single tweet to a tag of point of interests, e.g., club, or park, instead of user's home location.

Our contributions in this paper are as follows: (1) We provide an alternative estimation via *Gaussian Mixture Model (GMM)* to address the problems in *Maximum Likelihood Estimation (MLE)*; (2) We propose unsupervised measures to evaluate the usefulness of tweet words for location prediction task; (3) We compared 3 different models experimentally with proposed GMM based estimation and local word selection methods;

and (4) We show that our approach can, using only less than 250 local words (selected by unsupervised methods), achieve a comparable performance to the state-of-the-art that uses 3,183 local words (selected by the supervised classification based on 11,004 hand-labeled ground truth).

II. Modeling Locations of Twitter Messages

Recently, the *generative methods* (e.g., [11], [7], [12]) have been proposed to solve the proposed Problem 1. Assuming that each tweet and each word in a tweet is generated independently, the prediction of home city of user u given his or her tweet messages is made by the conditional probability under Bayes's rule and further approximated by ignoring $P(T_u)$ that does not affect the final ranking as follows:

$$\begin{aligned} P(C|T_u) &= \frac{P(T_u|C)P(C)}{P(T_u)} \\ &\propto P(C)\prod_{t_j \in T_u}\prod_{w_i \in t_j} P(w_i|C) \end{aligned}$$

where w_i is a word is a tweet t_j. If $P(C)$ is estimated with the maximum likelihood, the cities having a high usage of tweets are likely to be favored. Another way is to assume a uniform prior distribution among cities, also known as the *language model* approach in IR, where each city has its own language model estimated from tweet messages. For a user whose location in unknown, then, one calculates the probabilities of the tweeted words generated by each city's language model. The city whose model generates the highest probability of the tweets from the user is finally predicted as the home location. This approach characterizes the language usage variations over cities, assuming that users have similar language usage within a given city. Assuming a uniform $P(C)$, we propose another approach by applying Bayes rule to the $P(w_i|C)$ of above formula and replace the products of probabilities by the sums of log probabilities, as is common in probabilistic applications:

$$\begin{aligned} P(C|T_u) &\propto P(C)\prod_{t_j \in T_u}\prod_{w_i \in t_j} \frac{P(C|w_i)P(w_i)}{P(C)} \\ &\propto \sum_{t_j \in T_u}\sum_{w_i \in t_j} \log(P(C|w_i)P(w_i) \end{aligned}$$

Therefore, given C and T_u, the home location of the user u is the city c ($\in C$) that maximizes the above function as:

$$\text{argmax}_{c \in C} \sum_{t_j \in T_u}\sum_{w_i \in t_j} \log(P(c|w_i)P(w_i)$$

Instead of estimating a language model for a city, this model suggests to estimate the city distribution on the use of each word, $P(C|w_i)$, which we refer to it as **spatial word usage** in this paper, and aggregate all evidences to make the final prediction. Therefore, its capability critically depends on whether or not there is a distinct pattern of word usage among cities. Note that the proposed model is similar to the one used in [3], $P(C|T_u) \propto \sum_{t_j \in T_u}\sum_{w_i \in t_j} P(C|w_i)P(w_i)$, where the design was based on the observation rather than derived theoretically.

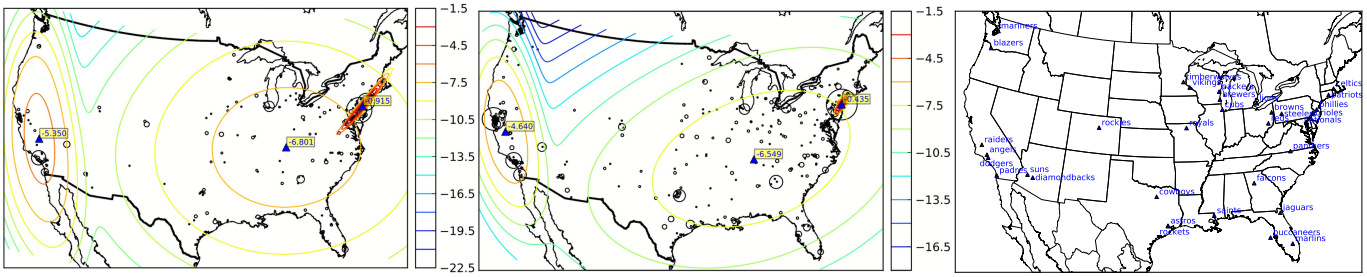
The *Maximum Likelihood Estimation (MLE)* is a common way to estimate $P(w|C)$ and $P(C|w)$. However, it suffers from the data sparseness problem that underestimates the probabilities of words of low or zero frequency. Various *smoothing* techniques such as Dirichlet and Absolute Discount [13] are proposed. In general, they distribute the probabilities of words of nonzero frequency to the words of zero frequency. For estimating $P(C|w)$, the probability of tweeting a word in locations where there are zero or few twitter users are likely to be underestimated as well. In addition to these smoothing techniques, some probability of a location can be distributed to its neighboring locations, assuming that two neighboring locations tend to have similar word usages. While reported effective in other IR applications, however, the improvements from such smoothing methods to estimate user locations have been shown to be limited in the previous studies [11], [3]. One of our goals in this paper is therefore to propose a better estimation for $P(C|w)$ to improve the prediction while addressing the spareness problem.

III. Estimation with Gaussian Mixture Model

The Backstrom model [1] demonstrated that users in a particular location tend to query some search keywords more often than users in other locations, especially, for some topic words such as sport teams, city names, or newspaper. For example, as demonstrated in [1], `redsox` is searched more often in New England area than other places. In their study, the likelihood of a keyword queried in a given place is estimated by $Sd^{-\alpha}$, where S indicates the strength of frequency on the local center of the query, and α indicates the speed of decreasing when the place is d away from the center. Therefore, the larger S and α in the model of a keyword shows a higher local interest, indicating strong local phenomena. While promising results are shown in their analysis with query logs, however, this model is designed to identify the center rather than to estimate the probability of spatial word usage and is difficult to handle the cases where a word exhibits multiple centers (e.g., `giants` for the NFL NY Giants and the MLB SF Giants).

Therefore, to address such issues, we propose to use the bivariate *Gaussian Mixture Model (GMM)* as an alternative to model the spatial word usage and to estimate $P(C|w)$. GMM is a mature and widely used technique for clustering, classification, and density estimation. It is a probability density function of a weighted sum of a number of Gaussian components. Under this GMM model, we assume that each word has a number of centers of interests where users tweet it more extensively than users in other locations, thus having a higher $P(c|w)$, and that the probability of a user in a given location tweeting a word is influenced by the word's multiple centers, the magnitudes of the centers, and user's geographic distances to those centers. Formally, using GMM, the probability of a city c on tweeting a word w is:

$$P(c|w) = \sum_{i=1}^{K} \pi_i N(c|\mu_i, \Sigma_i)$$



(a) `phillies` (b) `giants` (c) US sports teams

Fig. 1. Results of GMM estimation on selected words in Twitter data set.

where each $N(c|\mu_i, \Sigma_i)$ is a bivariate Gaussian distribution with the density as:

$$\frac{1}{2\pi|\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}(c-\mu_i)^T\Sigma_i^{-1}(c-\mu_i)\right\}$$

where K is the number of components and $\sum_{i=1}^{K}\pi_i = 1$. To estimate $P(C|w)$ with GMM, each occurrence of the word w is seen as a data point (lon, lat), the coordinate of the location where the word is tweeted. In other words, if a user has tweeted `phillies` 3 times, there are 3 data points (i.e., (lon, lat)) of the user location in the data set to be estimated by GMM. Upon convergence, we compute the density for each city c in C, and assign it as the $P(c|w)$. In the GMM-estimated $P(C|w)$, the mean of a component is the hot spot (i.e., center) of tweeting the word w, while the covariance determines the magnitude of a center. Similar to the Backstrom model, the chance of tweeting a word w decreases exponentially away from the centers. Unlike the Backstrom model, however, GMM easily generalizes to multiple centers and considers the influences under different centers (i.e., components) altogether. Furthermore, GMM is computationally efficient since the underlying EM algorithm generally converges very quickly. Compared to MLE, GMM may yield a high probability on a location where there are few Twitter user, as long as the the location is close to a hot spot. It may also assign a low probability to locations with high frequency of tweeting a word if that location is far way from all the hot spots. On the other hand, GMM-based estimation can be also viewed as a radical geographic smoothing such that neighboring cities around the centers are favored.

Example 1. In Fig. 1(a), we show the contour lines of log-likelihood of a GMM estimation with 3 components (i.e., $K = 3$) on the word `phillies` which has been tweeted 1,370 times from 229 cities in Twitter data set (see Section V). A black circle in the map indicates a city, where radius is proportional to the frequency of `phillies` being tweeted by users in the city. The corresponding centers are plotted as blue triangles. Note that there is a highly concentrated cluster of density around the center in northeast, close to Philadelphia, which is the home city of `phillies`. The other two centers and their surrounding areas have more low and diluted densities. Note that GMM works well in clustering probabilities around the location of interests with the evidences of tweeting location, even if the number of components (K) is not set to the exact number of centers. Sometimes, there might be more than one distinct cluster in a city distribution for a word. For example, `giants` is a name of a NFL team (i.e., New York Giants) as well a MLB team (i.e., San Francisco Giants). Therefore, it is likely to be often mentioned by Twitter users from both cities. As shown in Fig. 1(b), the two highest peaks are close to both cities. The peak near New York city has a higher likelihood than that near San Francisco, indicating `giants` is a more popular topic for users around New York city area. In Fig. 1(c), finally, we show that GMM can be quite effective in identifying the location of interests by selecting the highest peaks for various sport teams in US. □

As shown in Example 1, in fitting the spatial word usage with GMM, if a word has strong patterns, one or more major clusters are likely be formed and centered around the locations of interests with highly concentrated densities. If two close locations are both far away from the major clusters, their probabilities are likely to be smoothed out to a similar and low level, even if they are distinct in actual tweeted frequencies.

IV. Unsupervised Selection of Local Words

[3] made an insightful finding that in estimating locations of Twitter users, using only a selected set of words that show strong locality (termed as *local words*) instead of using entire corpus can improve the accuracy significantly (e.g., from 0.101 to 0.498). Similarly, we assumed that words have some locations of interests where users tend to tweet extensively. However, *not* all words have a strong pattern. For example, if a user tweets `phillies` and `libertybell` frequently, the probability for Philadelphia to be her home location is likely to be high. On the other hand, even if a user tweets words like `restaurant` or `downtown` often, it is hard to associate her with a specific location. That is because such words are commonly used and their usage will not be restricted locally. Therefore, excluding such globally occurring words would likely to improve overall performance of the task.

In particular, in selecting local words from the corpus, [3] used a supervised classification method. They manually labeled around 19,178 words in a dictionary as either local or non-local and used parameters (e.g., S, α) from the Backstorm's model and the frequency of a word as features to build a supervised classifier. The classifier then determines whether

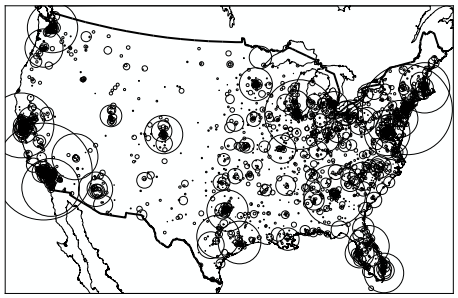


Fig. 2. The occurrences of the stop word `for` in Twitter data set.

other words in the data set are local. Despite the promising results, we believe that such a *supervised selection approach is problematic*–i.e., not only their labeling process to manually create a ground truth is labor intensive and subject to human bias, it is hard to transfer labeled words to new domain or data set. Moreover, the dictionary used in labeling process might not differentiate the evidences on different forms of a word. For example, the word `bears` (i.e., name of an NFL team) is likely to be a local word, while the word `bear` might not be. As a result, we believe that a better approach is to automate the process (i.e., *unsupervision*) such that the decision on the localness of a word is made only by their actual spatial word usage, rather than their semantic meaning being interpreted by human labelers. Toward this challenge, in the following, we propose two *unsupervised* methods to select a set of "local words" from a corpus using the evidences from tweets and their tweeter locations directly.

A. Finding Local Words by Non-Localness: NL

Stop words such as `the`, `you`, or `for` are in general commonly used words that bear little significance and considered as noises in many IR applications such as search engine or text mining. For instance, compare Fig. 2 showing the frequency distribution for the stop word `for` to Fig. 1 showing that for word with strong local usage pattern like `giants`. In Fig. 2, one is hard to pinpoint a few hotspot locations for `for` since it is globally used.In the location prediction task, as such, the spatial word usage of these stop words shows a somewhat *uniform* distributions adjusted to the sampled data set. As an automatic way to filter noisy non-local words out from the given corpus, therefore, we propose to use the stop words as *counter examples.* That is, local words tend to have the farthest distance in spatial word usage pattern to stop words. We first estimate a spatial word usage $p(C|w)$ for each word as well as stop words. The similarity of two words, w_i and w_j, can be measured by the distance between two probability distributions, $p(C|w_i)$ and $p(C|w_j)$. We consider two divergences for measuring the distance: Symmetric Kullback-Leibler divergence (sim_{SKL}) and Total Variation (sim_{TV}):

$$\text{sim}_{SKL}(w_i, w_j) = \sum_{c \in C} P(c|w_i) \ln \frac{P(c|w_i)}{P(c|w_j)} + P(c|w_j) \ln \frac{P(c|w_j)}{P(c|w_i)}$$

$$\text{sim}_{TV}(w_i, w_j) = \sum_{c \in C} |P(c|w_i) - P(c|w_j)|$$

For a given stop word list $S = \{s_1, ..., s_{|S|}\}$, we then define the *Non-Localness*, $NL(w)$, of a word w as the average similarity of w to each stop word s in S, weighted by the number of occurrences of s (i.e., frequency of s, $freq(s)$) :

$$NL(w) = \sum_{s \in S} \text{sim}(w, s) \frac{freq(s)}{\sum_{s' \in S} freq(s')}$$

From the initial tweet message corpus, finally, we can rank each word w_i by its $NL(w_i)$ score in *ascending* order and use top-k words as the final "local" words to be used in the prediction.

B. Finding Local Words by Geometric-Localness: GL

Intuitively, if a word w has: (1) a smaller number of cities with high probability scores (i.e., only a few peaks), and (2) smaller average inter-city geometric distances among those cities with high probability scores (i.e., geometrically clustered), then one can view w as a local word. That is, a local word should have a high probability density clustered within a small area. Therefore, based on these observations, we propose the *Geometric-Localness*, GL, of a word w:

$$GL(w) = \frac{\sum_{c'_i \in C'} P(c'_i|w)}{|C'|^2 \frac{\sum \text{geo-dist}(c_u, c_v)}{|\{(c_u, c_v)\}|}}$$

where geo-dist(c_u, c_v) measures the geometric distance in miles between two cities c_u and c_v. Suppose one sort cities c $(\in C)$ according to $P(c|w)$. Using a user-set threshold parameter, r $(0 < r < 1)$, then, one can find a sub-list of cities $C' = (c'_1, ..., c'_{|C'|})$ s.t. $P(c'_i|w) \geq P(c'_{i+1}|w)$ and $\sum_{c'_i \in C'} P(c'_i|w) \geq r$. In the formula of $GL(w)$, the numerator then favors words with a few "peaky" cities whose aggregated probability scores satisfy the threshold r. The denominator in turn indicates that $GL(w)$ score is inversely proportional to the number of "peaky" cities (i.e., $|C'|^2$) and their average inter-distance (i.e., $\frac{\sum \text{geo-dist}(c_u, c_v)}{|\{(c_u, c_v)\}|}$). From the initial tweet message corpus, finally, we rank each word w_i by its $GL(w_i)$ score in *descending* order and use top-k words as the final "local" words to be used in the prediction.

V. Experimental Validation

A. Set-Up

For validating the proposed ideas, we used the same Twitter data set collected and used by [3]. This data set was originally collected between Sep. 2009 and Jan. 2010 by crawling through Twitter's public timeline API as well as crawling by breadth-first search through social edges to crawl each user's followees/followers. The data set is further split into *training* and *test* sets. The training set consists of users whose location is set in city levels and within the US continental, resulting in 130,689 users with 4,124,960 tweets. The test set consists of 5,119 active users with around 1,000 tweets from each, whose location is recorded as a coordinate (i.e., latitude and longitude) by GPS device, a much more trustworthy data than user-edited location information. In our experiments, we

TABLE I
BASELINE RESULTS USING DIFFERENT MODELS.

Probability Model	ACC	AED
(1) $\sum\sum \log(P(c\|w_i)P(w_i))$	0.1045	1,760.4
(2) $\sum\sum P(c\|w_i)P(w_i)$	0.1022	1,768.73
(3) $\sum\sum \log P(w_i\|c)$	0.1914	1,321.42

TABLE II
RESULTS OF MODEL (1) ON GMM WITH VARYING # OF COMPONENTS K.

K	1	2	3	4	5
ACC	0.0018	0.025	**0.3188**	0.2752	0.2758
AED	958.94	1785.79	**700.28**	828.71	826.1
K	6	7	8	9	10
ACC	0.2741	0.2747	0.2739	0.2876	0.3149
AED	830.62	829.14	830.33	786.34	746.75

considered only 5,913 US cities with more than 5,000 of population in Census 2000 U.S. Gazetteer. Therefore, the problem that we experimented is to correctly predict *one* city out of 5,913 candidates as the home location of each Twitter user. We preprocess the training set by removing non-alphabetic characters (e.g., "@") and stop words, and selects the words of at least 50 occurrences, resulting in 26,998 unique terms at the end in our dictionary. No stemming is performed since singular and plural forms may provide different evidences as discussed in Section IV. Data sets and codes that we used in the experiments are publicly available at: `http://pike.psu.edu/download/asonam12/`.

To measure the effectiveness of estimating user's home location, we used the following two metrics also used in the literature [3], [2], [11]. First, the *accuracy (ACC)* measures the average fraction of successful estimations for the given user set U: $ACC = \frac{|\{u|u\in U \text{ and } dist(Loc_{true}(u),Loc_{est}(u))\leq d\}|}{|U|}$. The successful estimation is defined as when the distance of estimated and ground-truth locations is less than a threshold distance d. Like [3], [2], we use $d = 100$ (miles) as the threshold. Second, for understanding the overall margins of errors, we use the *average error distance (AED)* as: $AED = \frac{\sum_{u\in U} dist(Loc_{true}(u),Loc_{est}(u))}{|U|}$.

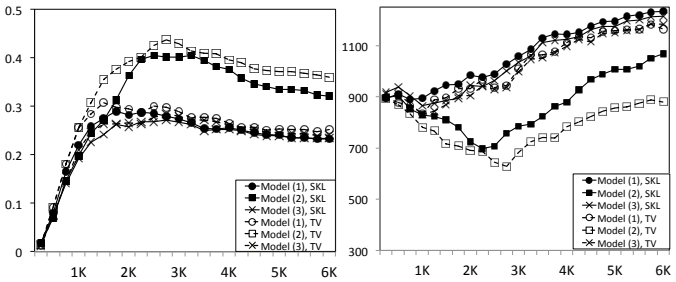
B. Baselines

In Section 2, we compared three different models as discussed in Sec II to understand the impact of selecting the underlying probability frameworks. Table I presents the results of different models for location estimation. All the probabilities are estimated with MLE using all words in our dictionary. The baseline Models (1) and (2) (proposed by [3]) utilize the spatial word usage idea, and have around 0.1 of ACC and around 1,700 miles in AED. The Model (3), a language model approach, shows a much improved result– about two times higher ACC and AED with 400 miles less. These results are considered as baselines in our experiments.

C. Prediction with GMM Based Estimation

Next, we study the impact of the proposed GMM estimation[1] for estimating locations. In general, the results using GMM shows much improvements over baseline results of Table I. Table II shows the results using Model (1) whose probabilities are estimated by GMM with different # of components K, using all the words in the corpus. Except the cases with $K = 1$ and $K = 2$, all GMM based estimations show substantial improvements over MLE based ones, where the best ACC (0.3188) and AED (700.28 miles) are achieved at $K = 3$. Although the actual # of locations of interests varies for each word, in general, we believe that the words that have too many location of interests are unlikely to make contribution to the prediction. That is, as K becomes large, the probabilities are more likely to be distributed, thus making the prediction harder. Therefore, in subsequent experiments, we focus on GMM with a small # of components.

[1]Using EM implementation from scikit-learn, http://scikit-learn.org



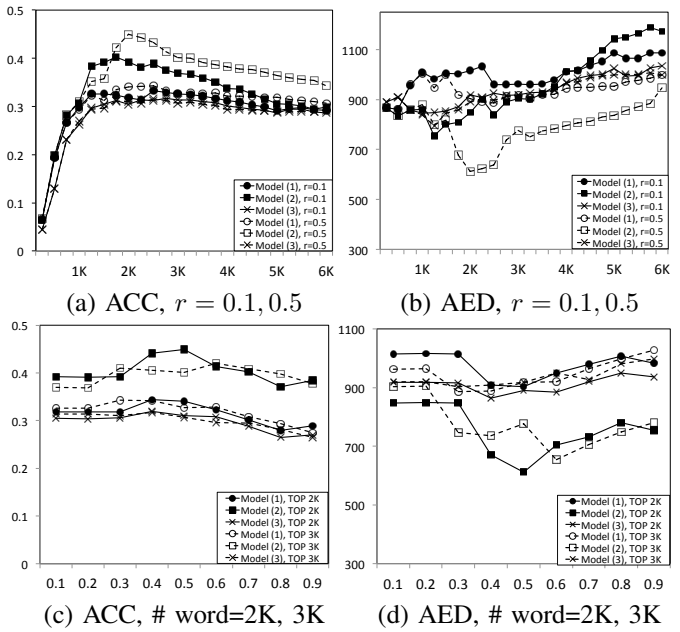
(a) ACC (b) AED

Fig. 3. Results with local words selected by *Non-Localness* (NL) on MLE estimation (X-axis indicates # of top-k local words used).

D. Prediction with Unsupervised Selection of Local Words

We attempt to see if the "local words" idea first proposed in [3] can be validated even when local words are selected in the unsupervised fashion (as opposed to [3]'s supervised approach). In particular, we validate with two unsupervised methods that we proposed on MLE estimation.

1) Non-Localness (NL): In measuring $NL(w)$ score of a word w, we use the English stop word list from SMART system [10]. A total of 493 stop words (out of 574 in the original list), roughly 1.8% of all terms in our dictionary, occurred about 23M times (52%) in the training data. Due to their common uses in the corpus, such stop words are viewed as the least indicative of user locations. Therefore, $NL(w)$ measures the degree of similarity of w to average probability distributions of 493 stop words. Accordingly, if w shows the most dissimilar spatial usage pattern, i.e. $P(C|w)$, from those of stop words, then w is considered to be a candidate local word. The ACC and AED (in miles) results are shown in Fig. 3, as a function of the # of local words used (i.e., chosen as top-k when sorted by $NL(w)$ scores). In summary, Model (2) shows the best result of ACC (0.43) and AED (628 miles) with 3K local words used, a further improvement over the best result by GMM in Section V-C of ACC (0.3188) and AED (700.28 miles). Model (1) has a better ACC but a worse AED than Model (3) has. In particular, local words chosen using sim_{TV} as the similarity measure outperforms sim_{SKL} for all three Models.



(a) ACC, $r = 0.1, 0.5$ (b) AED, $r = 0.1, 0.5$

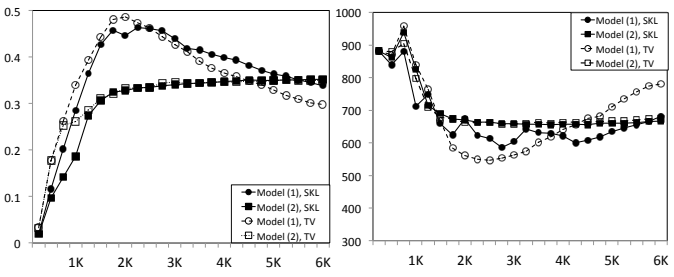
(c) ACC, # word=2K, 3K (d) AED, # word=2K, 3K

Fig. 4. Results with local words selected by *Geometric-Localness* (GL) on MLE estimation (X-axis in (a) and (b) indicates # of top-k local words used and that in (c) and (d) indicates r of $GL(w)$ formula).

2) Geometric-Localness (GL): Our second approach selects a word w as a local word if w yields only a small number of cities with high probability scores (i.e., only a few peaks) and a smaller average inter-city geometric distances. Fig. 4(a) and (b) show the ACC and AED of three probability models using either $r = 0.1$ and $r = 0.5$. The user-set parameter r ($=\sum_{c'_i \in C'} P(c'_i|w)$) of $GL(w)$ formula indicates the sum of probabilities of top candidate cities C'. Overall, all variations show similar behavior, but in general, Model (2) based variations outperform Model (1) or (3) based ones. Model (2) in particular achieves the best performance of ACC (0.44) and AED (600 miles) with $r = 0.5$ and 2K local words. Note that this is a further improvement over the previous case using NL as the automatic method to pick local words–ACC (0.43) and AED (628 miles) with 3K local words. Fig. 4(c) and (d) show the impact of r in $GL(w)$ formula, in X-axis, with the number of local words used fixed at 2K and 3K. In general, GL shows the best results when r is set to the range of 0.4 – 0.6. In particular, Model (2) is more sensitive to the choice of r than Models (1) and (3). In general, we found that GL slightly outperforms NL in both ACC and AED metrics.

E. Prediction with GMM and Local Words

In previous two sub-sections, we show that both GMM based estimation with all words and MLE based estimation with unsupervised local word selection are effective, compared to baselines. Here, further, we attempt to improve the result by combining both approaches to have unsupervised local word selection on the GMM based estimation. We first use the GMM to estimate $P(C|w)$ with $K = 3$, and calculate both $NL(w)$ and $GL(w)$ using $P(C|w)$. Finally, we use the top-k local words and their $P(C|w)$ to predict user's location. Since Model (3) makes a prediction with $P(W|C)$ rather than $P(C|W)$, GMM based estimation cannot be used for Model (3), and thus is not compared. Due to the limitation of space, we report the best case using $NL(w)$ in Fig 5. Model (1) generally outperforms Model (2) and achieves the best result so far for both ACC (0.486) and AED (583.2 miles) with sim_{TV} using 2K local words. While details are omitted, it is worthwhile to note that when used together with GMM, NL in general outperforms GL, unlike when used with MLE.



(a) ACC (b) AED

Fig. 5. Results with local words selected by *Non-Localness* (NL) on GMM estimation (X-axis indicates # of top-k local words used).

TABLE III
EXAMPLES OF CORRECTLY ESTIMATED CITIES AND CORRESPONDING TWEET MESSAGES (LOCAL WORDS ARE IN BOLD FACE).

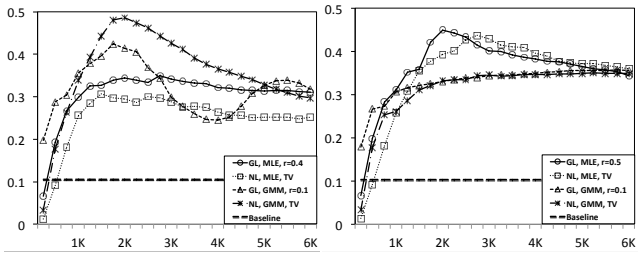
Est. City	Tweet Message
Los Angeles	i should be working on my monologue for my **audition** thursday but the thought of memorizing something right now is crazy
Los Angeles	i knew deep down inside ur powell s biggest fan p **lakers** will win again without **kobe** tonight haha if morisson leaves **lakers** that means elvan will not be rooting for **lakers** anymore
New York	the march **vogue** has caroline trentini in some awesome **givenchy** bangles i found a similar look for less an intern from teen **vogue** fashion dept just e mailed me asking if i needed an assistant aaadorable

Table III illustrates examples where cities are predicted successfully by using NL-selected local words and with GMM-based estimation. Note that words such as `audition` (i.e., the Hollywood area is known for movie industries) and `kobe` (i.e., name of the basketball player based in the area) are a good indicator of the city of the Twitter user.

In summary, overall, Model (1) shows a better performance with GMM while Model (2) with MLE as the estimation model. In addition, Model (1) usually uses less words to reach the best performance than Model (2) does. In terms of selecting local words, NL works better than GL in general, with sim_{TV} in particular. In contrast, the best value of r depends on the model and the estimation method used. The best result for each model is summarized in Table IV while further details on different combinations of those best results for Models (1) and (2) are shown Fig. 6.

TABLE IV
SUMMARY OF BEST RESULTS OF PROBABILITY AND ESTIMATION MODELS.

Model	Estimation	Measure	Factor	#word	ACC	AED
(1)	GMM	NL	sim_{TV}	2K	0.486	583.2
(2)	MLE	GL	$r = 0.5$	2K	0.449	611.6
(3)	MLE	GL	$r = 0.1$	2.75K	0.323	827.8



(a) Model (1) (b) Model (2)

Fig. 6. Settings for two models to achieve the best ACC.

F. Smoothing vs. Feature Selection

The technique to simultaneously increase the probability of unseen terms (that cause the sparseness problem) and decrease that of seen terms is referred to as *smoothing*. While successfully applied in many IR problems, in the context of location prediction problem from Twitter data, it has been reported that smoothing has very little effect in improving the accuracy [11], [3]. On the other hand, as reported in [3], *feature selection* seems to be very effective in solving the location prediction problem. That is, instead of using the entire corpus, [3] proposed to use a selective set of "local words" only. Through the experiments, we validated that the feature selection idea via local words is indeed effective. For instance, Fig. 6 shows that our best results usually occur when around 2,000 local words (identified by either NL or GL methods), instead of 26,998 original terms, are used in predicting locations. Having a reduced feature set is beneficial, especially in terms of speed. For instance, with Model (1) estimated by MLE, using 50, 250, 2,000, and 26,999 local words, it took 27, 32, 50, and 11,531 seconds respectively to finish the prediction task. In general, if one can get comparable results in ACC and AED, solutions with a smaller feature set (i.e., less number of local words) are always preferred. As such, in this section, we report our exploration to reduce the number of local words used in the estimation even further.

Figs. 4–6 all indicate that both ACC and AED (in all settings) improve in proportion to the size of local words up to 2K–3K range, but deviate afterwards. In particular, note that those high-ranked words within top-300 (according to NL or GL measures) may be good local words but somehow have limited impact toward overall ACC and AED. For instance, using GMM as the estimation model, GL yields the following within the top-10 local words: {`windstaerke`, `prazim`, `cnen`}. Upon inspection, however, these words turn out to be Twitter user IDs. These words got high local word scores (i.e., GL) probably because their IDs were used in re-tweets or mentioned by users with a strong spatial pattern. Despite their high local word scores, however, their usage in the entire corpus is relatively low, limiting their overall impact. Similarly, using MLE as the estimation mode, NL found the followings at high ranks: {`je`, `und`, `kt`}. These words are Dutch (thus not filtered in preprocessing) and heavily used in only a few US towns[2] of Dutch descendants, thus exhibiting a strong locality.

[2]Nederland (Texas), Rotterdam (New York), and Holland (Michigan)

TABLE V
PREDICTION WITH REDUCED # OF LOCAL WORDS BY FREQUENCY.

(a) Model (1), GMM, NL

		Number of local words used				
		50	*100*	*150*	*200*	*250*
ACC	Top 2K	0.433	0.447	0.466	0.476	**0.499**
	Top 3K	0.446	0.449	0.444	0.445	0.446
AED	Top 2K	603.2	599.6	582.9	565.7	531.1
	Top 3K	**509.3**	567.7	558.9	539.9	536.5

(b) Model (1), MLE, GL

		Number of local words used				
		50	*100*	*150*	*200*	*250*
ACC	Top 2K	0.354	0.382	0.396	0.419	**0.420**
	Top 3K	0.397	0.400	0.399	0.403	0.416
AED	Top 2K	771.7	761.0	760.3	730.6	**719.8**
	Top 3K	806.2	835.1	857.5	845.9	822.3

(c) Model (3), MLE, GL

		Number of local words used				
		50	*100*	*150*	*200*	*250*
ACC	Top 2K	0.2227	0.276	0.315	0.336	0.343
	Top 3K	0.301	0.366	0.385	0.401	**0.408**
AED	Top 2K	743.9	663.3	618.5	577.7	570.3
	Top 3K	620.7	565.6	535.1	510.8	**503.3**

However, again, their overall impact is very limited due to the rarity outside those towns. From these observations, therefore, we believe that both localness as well as frequency information of words must be considered in ranking local words.

Informally, $score(w) = \lambda \frac{localness(w)}{\Delta_l} + (1-\lambda)\frac{frequency(w)}{\Delta_f}$, where Δ_l and Δ_f are normalization constants for $localness(w)$ and $frequency(w)$ functions, and λ controls the relative importance between localness and frequency of w. The localness of w can be calculated by either NL or GL, while frequency of w can be done using IR methods such as relative frequency or TF-IDF. For simplicity, in this experiments, we implemented the $score()$ function in two-steps: (1) we first select base 2,000 or 3,000 local words by NL or GL method; and (2) next, we re-sort those local words based on their frequencies. Table V shows the results of ACC and AED using only a small number (i.e., 50–250) of top-ranked local words after re-sorted based on both localness and frequency information of words. Note that using only 50–250 local words, we are able to achieve comparable ACC and AED to the best cases of Table IV that use 2,000–3,000 local words. The improvement is the most noticeable for Model (1). The results show the quality of the location prediction task may rely on a small set of frequently-used local words.

Table VI shows top-30 local words with GMM, when re-sorted by frequency, from 3,000 NL-selected words. Note that most of these words are *toponyms*, i.e., names of geographic locations, such as `nyc`, `dallas`, and `fl`. Others include the names of people, organizations or events that show a strong local pattern with frequent usage, such as `obama`, `fashion`, or `bears`. Therefore, it appears that toponyms are important in predicting the locations of Tweeter users. Interestingly, a previous study in [11] showed that toponyms from image tags were helpful, though *not* significantly, in predicting the

TABLE VI
TOP-30 FREQUENCY-RESORTED LOCAL WORDS (GMM, NL).

la	nyc	hiring	dallas	francisco
obama	fashion	atlanta	houston	denver
san	diego	sf	austin	est
chicago	los	seattle	hollywood	yankees
york	boston	washington	angeles	bears
ny	miami	dc	fl	orlando

TABLE VII
PREDICTION WITH ONLY TOPONYMS.

		Number of toponyms used		
		50	*200*	*400*
ACC	Model (1)	0.246	0.203	0.115
	Model (2)	0.306	0.291	0.099
	Model (3)	0.255	**0.347**	0.330
AED	Model (1)	1202.7	1402.7	1719.7
	Model (2)	741.9	953.5	1777.4
	Model (3)	668.2	512.4	**510.1**

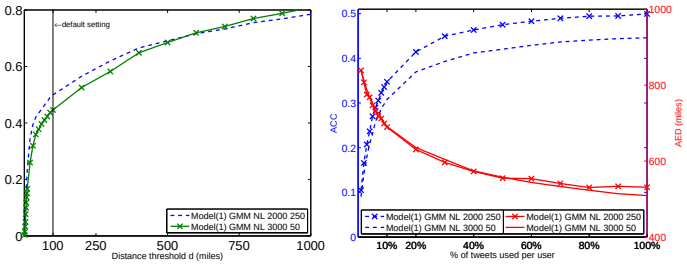
location of the images. Table VII shows the results using city names with the highest population in U.S. gazetteer as the "only" features for predicting locations (without using other local words). Note that performances are all improved with all three models, but are not good as those in Table V. Therefore, we conclude that using toponyms in general improve the prediction of locations, but *not* all toponyms are equally important. Therefore, it is *important to find critical local words or toponyms using our proposed NL or GL selection methods*. It further justifies that such a selection needs to be made from the evidences in tweet contents and user location, rather based on semantic meanings or types of words (as [3] did).

G. Discussion on Parameter Settings

First, same as the setting in literature, we used $d = 100$ (miles) in computing ACC–i.e., if the distance between the estimated city and ground truth city is less than 100 miles, we consider the estimation to be correct. Fig. 7(a) shows that ACC as a function of d using the best configuration (Model (1), GMM, NL) with 50 and 250 local words, respectively. Second, the test set that we used in experiments consists of a set of active users with around 1K tweets, same setting as [3] for comparison. Since not all Twitter users have that many tweets, we also experiment using different portion of tweet messages per user. That is, per each user in the test set, we randomly select from 1% to 90% of tweet messages to predict locations. The average results from 10 runs are shown in Fig. 7(b). While we achieve ACC (shown in left Y-axis) of 0.104 using 1% (10 tweets) per user, it rapidly improves to 0.2 using 3% (30 tweets), and 0.3 using 7% (70 tweets). Asymmetrically, AED (shown in right Y-axis) decreases as tweets increases.

VI. CONCLUSION

In this paper, we aim to improve the quality of predicting Twitter user's home location under probability frameworks. We proposed a novel approach to estimate the spatial word usage probability with Gaussian Mixture Models. We also proposed unsupervised measurements to rank the local words which effectively remove the noises that are harmful to the



(a) Varying d (b) Varying $|T_u|$

Fig. 7. Change of ACC and AED as a function of d and $|T_u|$.

prediction. We show that our approach can, using less than 250 local words selected by proposed methods, achieve a comparable or better performance to the state-of-the-art that uses 3,183 local words (selected by the supervised classification based on 11,004 hand-labeled ground truth).

ACKNOWLEDGMENT

The research was in part supported by NSF DUE-0817376 and DUE-0937891, and Amazon web service grant (2011). We thank Zhiyuan Cheng (infolab at TAMU) for providing their dataset and feedbacks on their implementation.

REFERENCES

[1] L. Backstrom, J. Kleinberg, R. Kumar, and J. Novak, "Spatial variation in search engine queries," in *WWW*, Beijing, China, 2008, pp. 357–366.
[2] S. Chandra, L. Khan, and F. B. Muhaya, "Estimating twitter user location using social interactions-a content based approach," in *IEEE SocialCom*, 2011, pp. 838–843.
[3] Z. Cheng, J. Caverlee, and K. Lee, "You are where you tweet: a content-based approach to geo-locating twitter users," in *ACM CIKM*, Toronto, ON, Canada, 2010, pp. 759–768.
[4] N. Dalvi, R. Kumar, and B. Pang, "Object matching in tweets with spatial models," in *ACM WSDM*, Seattle, Washington, USA, 2012, pp. 43–52.
[5] B. Hecht, L. Hong, B. Suh, and E. H. Chi, "Tweets from justin bieber's heart: the dynamics of the location field in user profiles," in *ACM CHI*, Vancouver, BC, Canada, 2011, pp. 237–246.
[6] P. Kelm, S. Schmiedeke, and T. Sikora, "Multi-modal, multi-resource methods for placing flickr videos on the map," in *ACM Int'l Conf. on Multimedia Retrieval (ICMR)*, Trento, Italy, 2011, pp. 52:1–52:8.
[7] S. Kinsella, V. Murdock, and N. O'Hare, ""i'm eating a sandwich in glasgow": modeling locations with tweets," in *Int'l workshop on Search and Mining User-Generated Contents (SMUC)*, Glasgow, Scotland, UK, 2011, pp. 61–68.
[8] M. Larson, M. Soleymani, P. Serdyukov, S. Rudinac, C. Wartena, V. Murdock, G. Friedland, R. Ordelman, and G. J. F. Jones, "Automatic tagging and geotagging in video collections and communities," in *ACM Int'l Conf. on Multimedia Retrieval (ICMR)*, Trento, Italy, 2011, pp. 51:1–51:8.
[9] W. Li, P. Serdyukov, A. P. de Vries, C. Eickhoff, and M. Larson, "The where in the tweet," in *ACM CIKM*, Glasgow, Scotland, UK, 2011, pp. 2473–2476.
[10] G. Salton, *The SMART Retrieval System - Experiments in Automatic Document Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1971.
[11] P. Serdyukov, V. Murdock, and R. van Zwol, "Placing flickr photos on a map," in *ACM SIGIR*, Boston, MA, USA, 2009, pp. 484–491.
[12] O. Van Laere, S. Schockaert, and B. Dhoedt, "Finding locations of flickr resources using language models and similarity search," in *ACM Int'l Conf. on Multimedia Retrieval (ICMR)*, Trento, Italy, 2011, pp. 48:1–48:8.
[13] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to information retrieval," *ACM Trans. Inf. Syst.*, vol. 22, no. 2, pp. 179–214, Apr. 2004.