Tweet Geolocation: Leveraging Location, User and Peer Signals

Wen-Haw Chong
Singapore Management University
80 Stamford Road, Singapore 178902
whchong.2013@phdis.smu.edu.sg

Ee-Peng Lim
Singapore Management University
80 Stamford Road, Singapore 178902
eplim@smu.edu.sg

ABSTRACT

Which venue is a tweet posted from? We referred this as *fine-grained geolocation*. To solve this problem effectively, we develop novel techniques to exploit each posting user's content history. This is motivated by our finding that most users do not share their visitation history, but have ample content history from tweet posts.

We formulate fine-grained geolocation as a ranking problem whereby given a test tweet, we rank candidate venues. We propose several models that leverage on three types of signals from locations, users and peers. Firstly, the location signals are words that are indicative of venues. We propose a location-indicative weighting scheme to capture this. Next we exploit user signals from each user's content history to enrich the very limited content of their tweets which have been targeted for geolocation. The intuition is that the user's other tweets may have been from the test venue or related venues, thus providing informative words. In this regard, we propose query expansion as the enrichment approach. Finally, we exploit the signals from peer users who have similar content history and thus potentially similar visitation behavior as the users of the test tweets. This suggests collaborative filtering where visitation information is propagated via content similarities. We proposed several models incorporating different combinations of the three signals. Our experiments show that the best model incorporates all three signals. It performs 6% to 40% better than the baselines depending on the metric and dataset.

CCS CONCEPTS

•**Information systems** →**Data mining;** *Geographic information systems;*

KEYWORDS

Tweet geolocation; query expansion; collaborative filtering

1 INTRODUCTION

In Twitter, users post tweets from their current locations with the option of associating their tweets with location coordinates. Such geocoded tweets can be mined for insights on visitation behavior or to support various applications such as venue recommendation or location-based advertising. However studies [2, 13] have shown that as much as 98% of tweets are not geocoded. This motivates the need for location inference with tweet geolocation techniques [2, 4, 19, 21, 26]. As tweets are short and colloquial, tweet geolocation is highly challenging.

In this work, we conduct *fine-grained tweet geolocation* [15, 19, 21, 23], which links tweets to the specific venues from which they were posted, e.g. a specific restaurant. This recovers the venue context which is useful for applications. Basically, a tweet is associated with different venue contexts when it is posted from different venues. This is true even if the candidate venues are adjacent to each other with effectively the same location coordinates. Such a fine-grained geolocation task is thus very different from most of the earlier works on coarse-grained geolocation [2, 13, 14, 17]. These works link tweets to their originating cities or to location coordinates. Clearly, applications such as location-based advertising can be better tailored if the venue context is known.

We formulate fine-grained geolocation as a ranking problem. Given a tweet, we rank venues such that highly ranked venues are more likely to be its posting venue. We refer to tweets to be geolocated as test tweets and the users posting them as test users. We assume that *test users have no observed visitation history*. This is the most common scenario since most tweets are not geocoded [2, 13] and most users do not share visitation history. The scenario also makes fine-grained geolocation more challenging due to the absence of information on the user's activity regions or hangout places. To mitigate this, we design our models to leverage three signals from *locations, users and peers*.

Firstly, we exploit location signals from words that are indicative of locations, e.g. the word 'airport' in the test tweet suggests that the posting venue could likely be an airport. To better leverage this signal, we propose a *location-indicative weighting* scheme and incorporate it into the naive Bayes geolocation model [17, 19]. Next, we consider signals from the content history of test users. We enrich the limited content in the test tweets by a novel query expansion method using the relevant users' historical content. The intuition is that the test users may have tweeted previously from the same venue, or from venues related via functionality or spatial proximity. Our proposed query expansion method treats the test tweet as a query and expands it by adding selected words from the test user's content history. We then geolocate the expanded test tweet. Lastly, we introduce peer signals from other users to estimate a test user's visitation behavior, so as to better geolocate his tweet. Peer signals are derived from users with known visitation history and who share similar content history as the test user. As users with similar content are also similar to some extent in their visitation behavior, we introduce *collaborative filtering* to harness the peer signals.

In short, our contribution is a model that exploits location, user and peer signals for better geolocation. We list each model aspect below, together with the intuitions (italicized):

CIKM'17, November 6–10, 2017, Singapore, Singapore

DOI: 10.1145/3132847.3132906

- We use **location-indicative weighting** to assign more weights to location-indicative words. *Such words are more important for inferring venues than other words.*
- We expand test tweets via **query expansion** and geolocate the expanded tweet. *Users have habits or constraints, often making repeated visits to the same or related venues.*
- We propose **collaborative filtering** to propagate visitation information across users connected via content similarities. *Users with more similar tweet content history may be more similar in their venue visitation history.*

The intuitions are elaborated with each model aspect in Section 4. For user and peer signals, we also justify the associated intuitions with empirical analysis in Sections 3.2 and 3.3.[1] We obtain best results from fusing all three signals, achieving 6% to 40% improvement over the baseline, depending on the dataset and metric.

2 TWEETS WITH POSTING VENUES

For model building and testing, we need to associate tweets with their posting venues. We obtain two types of tweets with such associations. The first type is tweets pushed to Twitter from location-apps, e.g. Foursquare. Such tweets have to be processed appropriately for non-trivial experiments. The second type is the user-authored tweets in Twitter, which we linked properly to venues for experiments. Next we describe the processing steps for the two tweet types.

2.1 Shouts

We use content pushed from Foursquare, a highly popular location app. In Foursquare, users can write comments and broadcast them to Twitter while they check-in to a venue. Following Foursquare terminology, we refer to such tweets as *shouts*. We process shouts appropriately and treat them as normal tweets with known ground truth venues. Such a setup is a convenient source of ground truth and has been used in prior work [4, 23].

A shout contains the user-authored comment plus an app-generated portion indicating the check-in venue. We discard the latter portion and use only the comments for geolocation. For example, consider the sample shout "**Passport photo look retarded** (@ Immigration & Checkpoints Authority w/ 5 others)". Only the user-authored comments (bolded) are used for geolocation and empirical analysis.

2.2 Pure Tweets

We refer to tweets that are authored by users and non-retweets as *pure tweets*. We find the Twitter account (if any) corresponding to each Foursquare user and extract their non-geocoded pure tweets. We then attempt to link these non-geocoded pure tweets to the posting venues of check-ins that occur around the same time. A pure tweet is assumed to be posted from a check-in venue when the tweet and check-in are performed within 5 minutes of each other.

Our linking approach requires the pure tweets' users to have visited the linked venues around the time they post their pure tweets. This is more stringent than just using geocoded pure tweets with location coordinates and assigning them to the nearest venues. The latter is unsatisfactory in an urban setting since many venues may share the same location coordinates, e.g. in a high rise building.

[1] We omit empirical analysis for location signals since the intuition is well known.

Table 1: Statistics for 50,000 sampled users from Singapore and Jakarta. $\mathbb{U}_c$: set of users with only content history.

	Singapore	Jakarta
Total Tweets	136,548,216	20,466,019
Geocoded Tweets	4,394,378 (3.22%)	946,432 (4.62%)
$\lvert\mathbb{U}_c\rvert$	34,831 (69.66%)	29,018 (58.04%)
Ave. tweets / $\mathbb{U}_c$ user	1820.02	558.80

Terminology. For the rest of this paper, 'tweets' refer to both pure tweets and shouts. Where differentiation is required, we use each term explicitly, i.e. pure tweets or shouts. Geocoded tweets may or may not be associated explicitly with venues. Minimally, they are associated with location coordinates.

2.3 Datasets

We collect data for users from Singapore (SG) and Jakarta (JKT). For Singapore, we collected 1,190,522 Foursquare check-ins from the year 2014, of which 30% involve shouts. The check-ins are posted by 29,301 users over 65,701 venues. We refer to this dataset as **SG-SHT**. Based on the linking process from Section 2.2, we also collected 90,250 pure tweets from 6424 users over 12,616 venues, which we designate as **SG-TWT**. For Jakarta, the **JKT-SHT** dataset comprises 177,570 check-ins for the period 2015 to mid-2016, of which 49% are shouts. The check-ins are from 12,119 users over 45,213 venues. Linking these check-ins to pure tweets, we obtain only 1335 pure tweets (**JKT-TWT**) posted by 592 users from 886 venues. This small number is possibly due to platform API changes which affected crawling. We use JKT-TWT only for testing in one experiment.

The datasets in this section are used for empirical analysis in Sections 3.2, 3.3 and for experiments in Section 5. For the empirical analysis in Section 3.1, we conducted a one time sampling of users as will be discussed next.

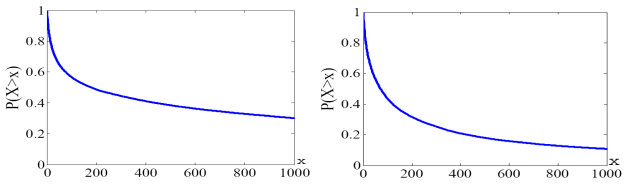
3 EMPIRICAL ANALYSIS

3.1 Scenario Study

We show that a major proportion of users have no visitation history but substantial content history. This motivates our discussed scenario. We randomly sample 50,000 Twitter users from Singapore for 2014 and from Jakarta for June to Dec 2016, with the condition that each user has posted at least one tweet during the study period. Table 1 shows the statistics compiled. The count values are higher for Singapore due to the longer study period considered, however the conclusion is the same for both cities.

Table 1 shows that the proportion of geocoded tweets is small at only 3.22% for Singapore and 4.62% for Jakarta. There is a substantial proportion of users with no geocoded tweets, which we denote as the set $\mathbb{U}_c$. Users in $\mathbb{U}_c$ have no visitation history and only content history from their non-geocoded tweets. $\mathbb{U}_c$ constitutes 69.66% of the sampled users for Singapore and 58.04% for Jakarta. These users have substantial numbers of non-geocoded tweets, e.g. each user in $\mathbb{U}_c$ has an average of 1820.02 tweets over one year for Singapore.

The tweet distributions further illustrate that $\mathbb{U}_c$ users have rich content history. Figure 1 plots the Complementary Cumulative Distribution Function (CCDF) of average tweet count for $\mathbb{U}_c$ users.



(a) Singapore (b) Jakarta

Figure 1: CCDF of average tweet count for $\mathbb{U}_c$ users.

Table 2: Repeat Visit Analysis

	Singapore	Jakarta
No. of tuples	603,198	108,428
tuples with freq=1	465,256 (77.13%)	88,219 (81.36%)
tuples with freq>1	137,942 (22.87%)	20,209 (18.64%)

For Singapore, around 55% of users have more than 100 tweets over a one year period, while for Jakarta, the proportion is around 45% over a half year period. Thus even though users in $\mathbb{U}_c$ have no visitation history, there is substantial content history. We seek to exploit such content history for better geolocation.

3.2 User Signals

To tap on user signals, we expand a test tweet with additional words from the same user and then geolocate the expanded tweet. This assumes the user has other tweets with words which are indicative of the posting venue. The presence of such words can be explained by several user behaviour aspects:

- **Repeat visits**: The user may have tweeted from the test venue before and used more informative words.
- **Nearby visits**: The same user tweeting from venues near each other may mention local geographical features. For example, assume we are geolocating a user's first tweet from a quayside restaurant. If he had previously visited neighboring restaurants and mention about the quay, then this will be indicative of the test venue to some extent.
- **Functionally related visits**: The test venue may belong to a functional group of venues that the user frequents, e.g. nightclubs. Functionally related words, e.g. 'clubbing' will indicate a clubbing test venue with some probability even if the test venue is being visited for the first time.

In this section, we only study repeat visits, while deferring empirical analysis of the other aspects to future work. Repeat visits is most straightforward to quantify and already motivates exploiting user signals. We examine shouts and tabulate the frequencies of repeated visits to venues, on a per user basis. Given user u and venue v, denote the user-venue tuple as (u, v). We only need to count tuples for users visiting a venue at least once. We iterate through all shouts and tabulate the frequencies of each tuple. Repeat visits are then user-venue tuples that occur more than once.

Table 2 shows that the proportion of repeat visits is substantial at 22.87% for Singapore (SG-SHT) and 18.64% for Jakarta (JKT-SHT). Thus, repeat visits is an established user behavior. This and the earlier discussed behavior aspects imply the presence of more informative words beyond the test tweet and justify query expansion.

Table 3: Query Expansion example.

Query/Test Tweet	"2nd day of orientation"
Query words	{day, orientation}
Sample tweets linked by common word 'day'	"Graduation day" "Last day of exam then holiday!!"
Query and Added words with weights	{ (day, 1.0), (orientation, 1.0), (exam,0.113), (graduation, 0.063), (school, 0.048), (holiday,0.045),...}

To illustrate how query expansion may help, Table 3 displays an actual query/test tweet which was sent from a school. The user had previously tweeted from the same venue. This is shown in the third row containing sample tweets linked by the common word 'day'. If informative words, e.g. 'exam' from these other tweets can be added to the test tweet, then the test tweet can be better geolocated. We discuss further details and revisit Table 3 in Section 4.2.

3.3 Peer Signals

Table 1 shows that each city has a smaller fraction of users with both content and visitation history, e.g. 30.34% for Singapore. Such users provide linkages between content and visitation behavior, which may help to geolocate tweets of $\mathbb{U}_c$ users. This motivates our empirical analysis to investigate the following question: *Are users who are more similar in content history also more similar in their visitation history?* If this is true, then one can devise collaborative filtering approaches, based on content similarities. In our analysis, we use the shout datasets: SG-SHT and JKT-SHT.

For each user u, we conduct the following:

(1) Represent each user u by a TFIDF vector $\mathbf{t}_u$ constructed from u's tweet content. Find the k nearest neighbors (denote as $nb(u)$) of u based on content similarity between $\mathbf{t}_u$ and the other users. The content similarity of two vectors is defined by cosine similarity. Also sample k dissimilar users (i.e. cosine similarity of 0) as non-nearest neighbors. Denote as the set $nnb(u)$.

(2) Compute average cosine similarity between u and his content view neighbors: $p_{nb}(u) = \frac{1}{|nb(u)|} \sum_{u_i \in nb(u)} sim(\mathbf{l}_u, \mathbf{l}_{u_i})$ where $\mathbf{l}_u$ and $\mathbf{l}_{u_i}$ are the TFIDF vectors constructed for u and u_i respectively from their venue visits. Repeat a similar computation with k sampled non-nearest neighbors $nnb(u)$ to obtain $p_{nnb}(u)$.

Note that in step (2) above, we have computed similarities in the venue view. For a set of users, we then compute the mean venue view similarities for the nearest neighbor and non-nearest neighbor sets: $\overline{p_{nb}}$ and $\overline{p_{nnb}}$. We also count the fraction of cases where $p_{nb}(u) > p_{nnb}(u)$. We studied 4271 users from Singapore (SG-SHT) and 911 users from Jakarta (JKT-SHT), who have at least 20 shouts. We experiment with $k = 10$ and 500, respectively representing small and large nearest neighbor sets. Table 4 displays the statistics.

Table 4 shows that the content and venue views are correlated. Consistently, across cities and different k values, content-based nearest neighbors give higher mean similarities $\overline{p_{nb}}$ than non-nearest neighbors $\overline{p_{nnb}}$. The last column also indicates that a major

Table 4: Profile analysis for Singapore and Jakarta users.

	$\overline{p_{nb}}$	$\overline{p_{nnb}}$	$p_{nb}(u) > p_{nnb}(u)$
SG-SHT, k=10	1.67E-2	4.76E-3	63.52 %
SG-SHT, k=500	1.19E-2	4.77E-3	77.08%
JKT-SHT, k=10	1.60E-2	2.76E-3	71.79%
JKT-SHT, k=500	9.81E-3	3.06E-3	64.65%

fraction of users see their content-based neighbors having more similar visitation history than sampled non-neighbors.

The empirical results indicate that collaborative filtering is feasible for our geolocation scenario. To better geolocate the tweet of users with only content history, we shall propagate information from users with both content and venue history.

4 MODELS

We present several proposed models covering three aspects, namely: location-indicative weighting, query expansion and collaborative filtering. Respectively, these incorporate location, user and peer signals. We also describe the fused model combining all aspects.

Our proposed models are based on a probabilistic framework using naive Bayes model as the base [17, 19]. Let $\mathbf{w}$ be the set of words in a test tweet. For notation simplicity, also assume each unique word w in the test tweet only occurs once. In [17, 19], $p(v)$, the probability of posting from venue v is assumed to be constant. We adopt the same assumption, which is reasonable for datasets that are not overly skewed towards popular venues. Given a test tweet $\mathbf{w}$, the probability of venue v is $p(v|\mathbf{w}) \propto \prod_{w \in \mathbf{w}} p(w|v)$. Venues are then ranked by $p(v|\mathbf{w})$. The naive Bayes model is efficient and has been shown to work well.

4.1 Location-Indicative Weighting Model (LW)

The next model, location-indicative weighting (LW) model, incorporates location signal with the intuition that *Location-indicative words are more important for inferring venues than other words.* Such words indicate one or more venues with high probabilities, i.e. high $p(v|w)$. This concept differs from the venue probabilities over words $p(w|v)$ which is prescribed by the naive Bayes model. For example, a dining venue v may have high probability for the word 'dinner', i.e. high $p(\text{'dinner'}|v)$. However if there are many dining venues, 'dinner' may not necessarily indicate the venue with high probability i.e. low $p(v|\text{'dinner'})$. If a tweet mentions dinner and venue-specific dishes or characteristics, then the latter words are more location-indicative and should be assigned more importance when computing $p(v|\mathbf{w})$.

To capture the discussed intuition, we propose a location-indicative weighting scheme which assigns weights on a continuous scale, and is easily incorporated into the naive Bayes model. Interestingly, combining naive Bayes with weighting schemes had been previously explored [10, 31]. It was to address classification tasks which are very different from the tweet geolocation task. Let $\beta(w)$ be the *location-indicative* weight for word w. The LW model is therefore:

$$\ln p(v|\mathbf{w}) \propto \sum_{w \in \mathbf{w}} \beta(w) \ln p(w|v) \quad (1)$$

where we have used the logarithmic form to avoid underflow errors.

Location-Indicative Weights. In our context, locations are discrete venues and akin to documents. Hence we can apply the vector space model. Words that are location-indicative will have large inverse-document frequencies, i.e. they occur in fewer venues. Formally given word w, we set its weight as:

$$\beta(w) = \log(1 + V/df(w)) \quad (2)$$

where V is the number of distinct venues and $df(w)$ counts venues where w occurs at least once. $\beta(w)$ is computed for all words that meet a minimum support frequency. We exclude rare noisy words..

4.2 Query Expansion Model (QE)

This model incorporates user signal with the intuition that *users have habits or constraints, often making repeated visits to the same or related venues.* This is intuitive e.g. work or school are usually carried out repeatedly at the same venue, or a user may have favourite hangouts. In Section 3.2, we have shown repeated visits to be an established user behavior. We also discussed that users visiting venues that are near or similar in function to the test tweet's venue justifies query expansion as well.

Query expansion has been largely used for document retrieval [9, 27, 29]. Adapting it for tweet geolocation is a novel idea. In our context, the query refers to the test tweet. Geolocating a test tweet on its own is difficult due to its brevity and information sparsity. With query expansion, we seek to retrieve words from related tweets to fill in the missing information. Given a test tweet, we iterate through its words and add co-occurring words from the user's other tweets. The added words are also scored appropriately.

Given query/test tweet $\mathbf{w}$ from user u, we score candidate words w' which appears in u's other tweets and where $w' \notin \mathbf{w}$. The scoring aims to assess w''s suitability for adding to the query and are designed to reflect the relationship strength to the original query words $w \in \mathbf{w}$. Many scoring schemes exist and we adopt a cosine similarity scheme [9]. For a candidate word w', we compute its average relatedness $\alpha_u(w', \mathbf{w})$ to the original query words:

$$\alpha_u(w', \mathbf{w}) = \frac{1}{|\mathbf{w}|} \sum_{w \in \mathbf{w}} \frac{d_u(w', w)}{\sqrt{d_u(w) d_u(w')}} \quad (3)$$

where $0 \leq \alpha_u(w', \mathbf{w}) \leq 1$, $d_u(w', w)$ is the count of u's tweets with both w' and w; and $d_u(w)$ is the count of u's tweets with w. Intuitively, words that co-occur more are more related. However relatedness is dampened if one or both words are overly common.

We add all words w' with $\alpha_u(w', \mathbf{w}) > 0$ to the query. Note that original query words $w \in \mathbf{w}$ have a weight of 1, hence added words are weighted less or at most equal to the original query words. After query expansion, we derive a weighted naive Bayes model to combine two different word sets:

$$\ln p(v|\{\mathbf{w}, \mathbf{w'}\}, u) \propto \sum_{w \in \mathbf{w}} \ln p(w|v) + \sum_{w' \in \mathbf{w'}} \alpha_u(w', \mathbf{w}) \ln p(w'|v) \quad (4)$$

where $\mathbf{w'}$ is the set of added words for tweet $\mathbf{w}$ from user u. Given that $0 \leq \alpha_u(w', \mathbf{w}) \leq 1$, Equation 4 illustrates that the original query words $w \in \mathbf{w}$ have greatest importance in the naive Bayes model while newly added words $w' \in \mathbf{w'}$ have varying degrees of importance based on how related they are to the query. Table 3 illustrates query expansion for a sample tweet. The original query

words are 'day' and 'orientation' (after stop word and rare word exclusion). The last row of the table shows the query words and added words along with their weights after query expansion. Since the added words **w'** are from the user's other tweets, we have in fact personalized the model to the user.

4.3 Fused Model (LWQE)

We now propose a weighted naive Bayes model that combines the LW and QE models. Intuitively, a word is important only when it is both location-indicative and highly related to the test tweet. Consider the cases where either requirement is not satisfied. If a word is not location-indicative, then it is less useful for geolocation even if it is in the original query or is a highly related word. Conversely, a location-indicative, but unrelated word to the query will introduce noise and hurt geolocation acccuracy.

We capture the above intuitions by multiplying weights from location-indicative weighting and query expansion. We formulate the combined model LWQE, as follows:

$$\ln p(v|\{\mathbf{w}, \mathbf{w'}\}, u) \propto \sum_{w \in \mathbf{w}} \beta(w) \ln p(w|v) + \sum_{w' \in \mathbf{w'}} \beta(w') \alpha_u(w', \mathbf{w}) \ln p(w'|v) \quad (5)$$

4.4 Collaborative Filtering Model (LWQE-CF)

Lastly we incorporate peer signals with collaborative filtering, based on the intuition:*Users with more similar tweet content history may be more similar in their venue visitation history.* This is also supported by our empirical analysis in Section 3.3.

Given tweets associated with venues (either through location-apps or our linking process in Section 2.2), we use collaborative filtering to estimate the user visitation distribution to venue $p(v|u)$. We then use $p(v|u)$ to personalize our earlier models. For example, the naive Bayes model can be extended as $p(v|\mathbf{w}, u) \propto p(v|u) \prod_{w \in \mathbf{w}} p(w|v)$. However $p(v|u)$ is not directly computable for users without visitation history, i.e. $u \in \mathbb{U}_c$ (See Table 1). To overcome this, we use collaborative filtering to propagate visitation information from users not in $\mathbb{U}_c$ to those within. Propagation is via content similarities since content history exist for all users.

Let $\mathbf{t}_u$ represent u in the content view. Many forms of representations are possible. For simplicity, we represent each user as a TFIDF vector where vector dimensions correspond to words in u's content. To compute $p(v|u)$ for user $u \in \mathbb{U}_c$, we first estimate u's visit frequencies to venues from similar users in the content view. Let $nb(u)$ contain k users with visitation history and who are most similar to u in terms of content cosine similarity. Also denote $\hat{c}(u, v)$ as the estimated frequency from user u to venue v. We compute:

$$\hat{c}(u, v) = S^{-1} \sum_{u' \in nb(u)} sim(\mathbf{t}_u, \mathbf{t}_{u'}).c(u', v) \quad (6)$$

where $c(u', v)$ is the observed frequency from user u' to venue v, $sim(,)$ is cosine similarity and S sums the similarities for normalization. We then compute $p(v|u)$ as:

$$p(v|u) = \frac{\hat{c}(u, v) + 1}{\sum_{v'} \hat{c}(u, v') + V} \quad (7)$$

Lastly, we extend Equation 5 with the probability $p(v|u)$:

$$\ln p(v|\{\mathbf{w}, \mathbf{w'}\}, u) \propto \ln p(v|u) + \sum_{w \in \mathbf{w}} \beta(w) \ln p(w|v) + \sum_{w' \in \mathbf{w'}} \beta(w') \alpha_u(w', \mathbf{w}) \ln p(w'|v) \quad (8)$$

Equation 8 encapsulates all our main model aspects: location-indicative weighting, query expansion and collaborative filtering.

4.4.1 Weighted Similarities. For collaborative filtering, we can modify the content similarity measure such that visitation and content similarities are more correlated. The intuition is that *users are more similar in their visitation history if they are more similar in their usage of location-indicative words.* For example, two users who mentioned a restaurant-specific dish will be more likely to have visited the same restaurant, compared to users who only mentioned 'dinner'. This implies that location-indicative words should be given greater importance in the similarity function between content history. This is easily done by including weights in the similarity function. Given two users u and u', we compute the weighted cosine similarity as:

$$wsim(\mathbf{t}_u, \mathbf{t}_{u'}) = \frac{\sum_w \beta(w)^2 \mathbf{t}_u(w).\mathbf{t}_{u'}(w)}{\| \mathbf{t}_u \|_2 \| \mathbf{t}_{u'} \|_2} \quad (9)$$

where $\beta(w)$ was discussed in Equation 2 and $\mathbf{t}_u(w)$ is the w-th dimension of vector $\mathbf{t}_u$. Replacing $sim(,)$ with $wsim(,)$ in Equation 6, we obtain the model LWQE-LW-CF.

5 EXPERIMENTS

We conduct fine-grained geolocation experiments in order to:

(1) Compare our models among each other and with other state-of-the-art baseline models.
(2) Assess the importance of different signals.

For each dataset (see Section 2.3), we conduct 10 runs where each run differs from the others by the tweets that are sampled for training/testing. In each run for SG-SHT, we randomly sample 5000 tweets for testing. For the smaller datasets, SG-TWT and JKT-SHT, we sample 2000 tweets for testing. Tweets not sampled for testing are used for model training. The JKT-TWT dataset (1335 tweets) is too small for training. It is used only in a single run as a test set for the model trained on JKT-SHT. We geolocate test tweets regardless of whether they contain any location-indicative words or not.

Recap that we are focusing on the scenario where users of test tweets have no visitation history. To emulate this scenario, we process the training tweets as follows: If a user has one or more tweets sampled for testing, we iterate through his training tweets and hide their venues (if any). This is repeated for all users of test tweets. Thus, the training set consists of a mixture of tweets with hidden venue associations due to the owners having some tweets selected for testing, and other tweets whose venue associations are retained. We consider a venue as candidate for ranking only if it is associated with at least 3 training tweets. We also exclude stop words and rare words with frequency < 3. Due to such filtering, the number of test cases per run is less than the number of sampled test tweets. The average number of test cases and venues to rank are reported in Section 5.3 which discusses the results of each dataset.

We compare the unweighted naive Bayes model (Nb) [17, 19] with our models incorporating different signal combinations (in brackets):

- LW (Location): Location-indicative weighting as indicated in Equation 1
- QE (User): Query expansion as indicated in Equation 4
- LWQE (Location+User): Fusion of query expansion and location-indicative weighting as shown in Equation 5.
- LWQE-P (Location+User): LWQE with a Laplace-smoothed global venue popularity model. This is Equation 8 with $p(v|u)$ replaced by $p(v)$
- LWQE-CF (Location+User+Peer): This is LWQE combined with a personalized venue distribution $p(v|u)$ from collaborative filtering. See Equation 8.
- LWQE-LW-CF (Location+User+Peer): This is similar to LWQE-CF except that we use location-indicative weighting when computing cosine similarities between users.

We also compare with the following baseline models:

- KL: This model [23] assigns scores to venues based on time information and the Kullback-Leibler divergences between the language models of tweets and venues. The scores are then used to rank venues.
- GMM: This model [5] represents each word as a Gaussian mixture over 2-d space, and a test tweet as the product of Gaussian mixtures. Venues are ranked by the probability of the product of Gaussian mixtures generating their coordinates. As in [5], we set the number of clusters to 3.
- TM: [8] proposes topic models to generate Foursquare check-ins and tips. Among them, we use the Udoc model to learn topics for both training tweets associated with and not associated with venues.[2] For each tweet, Udoc generates a user-dependent topic which generates the tweet words. If the tweet is associated with a venue, the venue is generated conditional on the topic. In our experiments, we used 40 topics, which exhibits optimal ranking performance.

In our experiments, we geolocate tweets regardless of whether they contain any location-indicative words or not. We note that the work in [19] includes a filtering step, such that only tweets with location-indicative words are geolocated. For better comparison with [19], we conduct a stratified experiment in Section 5.4.

5.1 Metrics

Each tweet is posted from one venue, which is desired to be ranked high. Since there is only one relevant venue, we use the Mean Reciprocal Rank (MRR) as the evaluation metric. Given tweet $\mathbf{w}_i$, let the rank of its posting venue be $r(\mathbf{w}_i)$, where $r(\mathbf{w}_i) = 0$ for the top rank. Over N test cases, MRR is defined as:

$$\text{MRR} = N^{-1} \sum_{i=1}^{N} (r(\mathbf{w}_i) + 1)^{-1} = N^{-1} \sum_{i=1}^{N} \text{RR}(\mathbf{w}_i) \quad (10)$$

where RR(.) is Reciprocal Rank.

MRR considers micro-averages. For randomly sampled test cases, popular venues will contribute a larger proportion of tweets, and be more important in determining MRR. In practical applications e.g. geolocating a stream of tweets, this is realistic and there is no reason to avoid this. However for further analysis, we consider the case where all venues are treated as equally important, regardless of their popularities. Thus we introduce a second evaluation metric, denote as VMRR. This is simply the macro-averaged version of MRR. For all test cases from the same posting venue, we average their MRR such that each test venue contributes only one value. We then do a second averaging over distinct test venues. Formally:

$$\text{VMRR} = V^{-1} \sum_{i=1}^{V} \text{MRR}(v_i) \quad (11)$$

where $\text{MRR}(v_i)$ is MRR values averaged over all test cases from venue v_i and V is the number of distinct test venues.

5.2 Result Summary

We summarize the MRR and VMRR results for every dataset in Tables 5 and 6. We conduct significance testing except for JKT-TWT which has just a single run. For other datasets, the "model 1 < model 2" notation means that model 2 significantly outperforms model 1 by the Wilcoxon signed rank test. In each row, models are arranged from left to right in ascending order of performance. Models that are not significantly different at p-value of 0.05 are grouped in brackets. For example, Table 5 shows that for SG-SHT, QE performs better than Nb for MRR and the results are statistically significant. For the same metric and data set, LWQE-CF and LWQE-LW-CF perform the best, but they are not statistically different from each other. In rare cases, we list a model twice if it is statistically insignificant against two closest models (in terms of performance), but the two models are significant against each other.

While there are permutations in model ordering, some general trend holds. Comparing against Nb, GMM and KL performs poorer while QE and LW performs better. TM's performance is mixed and tends to be poorer for VMRR. For the MRR metric in Table 5, QE is better than Nb in all datasets while LW outperforms Nb in SG-SHT, JKT-SHT and is on-par in SG-TWT. For the VMRR metric in Table 6, QE performs better than Nb, except for SG-SHT. In the same table, LW outperforms Nb in all datasets.

While QE and LW perform relatively well against Nb, we achieve more consistent improvement by fusing both approaches. This is illustrated by the model LWQE. In both Tables 5 and 6, LWQE always outperform Nb. It is also typically better than QE or LW alone, lying to the right of both models for most cases. Finally, we achieve the best results with LWQE-CF and LWQE-CF-LW-CF, which combines location-indicative weighting, query expansion and collaborative filtering. Except for one case (VMRR on SG-SHT), these two models are consistently the best performers.

5.3 Detailed Results

Tables 7, 8 and 9 display the average MRR and VMRR values over 10 runs for SG-SHT, SG-TWT and JKT-SHT respectively. Table 10 displays the results where models are trained on JKT-SHT and tested on JKT-TWT in a single run.

As shown in Tables 7 to 10, both KL and GMM perform poorly, underperforming even the Nb model. As tweets are very short, modeling each with a smoothed language model, as done by KL is inadequate. This in turn affects the computing of KL divergences

[2]We found the Vdoc model to perform worst as it can only model tweets associated with venues. Results omitted.

Table 5: Result Summary for MRR

SG-SHT: {GMM} < {KL} < {TM} < {Nb} < {QE} < {LW} < {LWQE} < {LWQE-P} < {LWQE-CF, LWQE-LW-CF}
SG-TWT: {GMM} < {KL} < {Nb, LW} < {QE, LWQE} < {LWQE-P,TM} < {TM, LWQE-CF, LWQE-LW-CF}
JKT-SHT: {KL} < {GMM} < {TM, Nb} < {LW} < {QE} < {LWQE} < {LWQE-P} < {LWQE-LW-CF, LWQE-CF}
JKT-TWT: KL < GMM < LW < Nb < LWQE < LWQE-P < QE < LWQE-CF < LWQE-LW-CF < TM

Table 6: Result Summary for VMRR

SG-SHT: {GMM, TM} < {KL} < {QE} < {Nb} < {LWQE-P} < {LWQE < LWQE-CF} < {LWQE-LW-CF, LW}
SG-TWT: {GMM} < {KL} < {TM} < {Nb} < {QE} < {LW} < {LWQE-P, LWQE} < {LWQE-CF} < {LWQE-LW-CF}
JKT-SHT: {GMM, TM, KL} < {Nb} < {QE} < {LW} < {LWQE, LWQE-P}< {LWQE-CF, LWQE-LW-CF}
JKT-TWT: GMM < TM < KL < Nb < QE < LW < LWQE-P < LWQE < LWQE-CF < LWQE-LW-CF

Table 7: SG-SHT results. Bracketed numbers are percentage improvement over Nb. Best results are bolded. On average, there are N=3248.5 test cases and V=9209.1 venues to rank per run.

Models	MRR	VMRR
KL	0.0447 (-54.98%)	0.0254 (-27.22%)
GMM	0.0317 (-68.08%)	0.0119 (-65.90%)
TM	0.0665 (-33.03%)	0.0125 (-64.18%)
Nb	0.0993	0.0349
LW	0.1049 (5.69%)	**0.0406 (16.55%)**
QE	0.1008 (1.53%)	0.0339 (-2.72%)
LWQE	0.1066 (7.38%)	0.0402 (15.29%)
LWQE-P	0.1074 (8.20%)	0.0399 (14.46%)
LWQE-CF	0.1088 (9.61%)	0.0402 (15.46%)
LWQE-LW-CF	**0.1090 (9.84%)**	0.0405 (16.12%)

Table 8: SG-TWT results. Notations as in Table 7. On average per run, N=1049.9, V=2672.5.

Models	MRR	VMRR
KL	0.0275 (-53.15%)	0.0136 (-28.42%)
GMM	0.0170 (-71.04%)	0.0119 (-37.37%)
TM	0.0666 (13.46%)	0.0151 (-20.53%)
Nb	0.0587	0.0190
LW	0.0596 (1.5%)	0.0221 (16.09%)
QE	0.0638 (8.57%)	0.0196 (2.91%)
LWQE	0.0646 (9.96%)	0.0230 (20.71%)
LWQE-P	0.0650 (10.70)	0.0230 (20.60%)
LWQE-CF	0.0674 (14.79%)	0.0236 (23.81%)
LWQE-LW-CF	**0.0675 (14.89%)**	**0.0238 (25.06%)**

between the word distributions of tweets and venues. Even with the inclusion of time information, performance is not promising. For GMM, performance is poor as we have to geolocate even tweets where words do not have peaky Gaussian distributions. The topic model TM has mixed performance. It achieves good MRR values for SG-TWT in Table 8 and JKT-TWT in Table 10, but performs poorly for MRR for other datasets, as well as for the VMRR metric. The poor VMRR performance implies that TM is biased to a larger extent towards more popular venues and works poorly when all venues are treated as equally important.

For shouts and pure tweets of both Singapore and Jakarta, LW improves over Nb much more substantially for VMRR than MRR. This can be seen by comparing the rows 'LW' and 'Nb' in Tables 7 to 10. For example in Table 7, LW improves over Nb by 5.69% for MRR. For VMRR, the corresponding improvement is much larger at 16.55%. This trend means that test tweets posted from less popular venues experience relatively larger improvement from location-indicative weighting. As less popular venues are associated with fewer tweets, information is sparse for modeling and it is harder to geolocate their tweets. For such venues, location-indicative words becomes relatively more important for a geolocation model.

We now compare QE to Nb. The result for QE is mixed for SG-SHT (Table 7). it achieves a small improvement of 1.53% in MRR, but results in a slight dip of 2.72% for VMRR. For SG-TWT (Table 8), JKT-SHT (Table 9) and JKT-TWT (Table 10), QE is more consistent in improving over Nb for both metrics. Generally the results indicate room for improvement. We note that query expansion may be noisy and expand a test tweet with words that are less relevant to the test venue. This also depends on the word relatedness function. We have currently used a relatively simple cosine similarity based function. While more complicated selection mechanisms [11] can be explored, the current query expansion technique is already shown to be useful over the combination of datasets and metrics.

LWQE combines the intuitions of LW and QE, i.e. words are more important if they are *both* location-indicative and highly related to the query. As can be seen, LWQE mostly outperforms LW or QE. In 6 out of 8 dataset-metric combinations, LWQE outperforms both LW and QE. For example in Table 8 for SG-TWT, LWQE's VMRR is 0.023, better than QE (0.0196) or LW (0.0221) alone.

For non-collaborative filtering models, LWQE and LWQE-P are best performers. Comparing both models in Tables 7 to 10, LWQE-P is always better than LWQE for the MRR metric, but not for VMRR. This is expected since LWQE-P utilizes a globally estimated venue distribution $p(v)$ which is related to venue popularity. Venue popularity is however controlled for in VMRR.

We now compare the collaborative filtering model LWQE-CF, against LWQE and LWQE-P, which are best performing models without collaborative filtering. Across Tables 7 to 10, LWQE-CF always improve on MRR and VMRR against both LWQE and LWQE-P. Hence information propagated from users with visitation history is useful for geolocating the tweets of users with only content history. Since propagation is across content similarities, it also affirms our stated intuition that users that are more similar in content history are more similar in their visitation behavior.

Lastly we note that LWQE-LW-CF is either comparable (Table 9) or provides very small improvement (Tables 7, 8 and 10) over LWQE. This is probably due to the fact that other aspects of the model, e.g. collaborative filtering, location-indicative weighting already captures much existing information that are useful for geolocation.

Table 9: JKT-SHT results. Notations as in Table 7. On average per run, N=626, V=2492.8.

Models	MRR	VMRR
KL	0.0759 (-54.52%)	0.0259 (-27.04%)
GMM	0.1296 (-22.35%)	0.0232 (-34.65%)
TM	0.1657 (-0.72%)	0.0250 (-29.58%)
Nb	0.1669	0.0355
LW	0.1691 (1.32%)	0.0403 (13.69%)
QE	0.1716 (2.82%)	0.0372 (4.82%)
LWQE	0.1737 (4.08%)	0.0424 (19.42%)
LWQE-P	0.1760 (5.42%)	0.0425 (19.77%)
LWQE-CF	**0.1778 (6.51%)**	0.0435 (22.58%)
LWQE-LW-CF	0.1777 (6.48%)	**0.0437 (23.06%)**

Table 10: JKT-TWT results. Notations as in Table 7. There is 1 run: N=475, V=4299.

Models	MRR	VMRR
KL	0.0521 (-43.68%)	0.0205 (-8.89%)
GMM	0.0678 (-26.70%)	0.0148 (-34.22%)
TM	0.1130 (22.16%)	0.0157 (-30.22%)
Nb	0.0925	0.0225
LW	0.0924 (-0.11%)	0.0284 (26.07%)
QE	0.0959 (3.66%)	0.0266 (17.84%)
LWQE	0.0933 (0.90%)	0.0305 (35.11%)
LWQE-P	0.0956 (3.43%)	0.0301 (33.58%)
LWQE-CF	0.0975 (5.46%)	0.0313 (38.73%)
LWQE-LW-CF	**0.0982 (6.19%)**	**0.0322 (42.86%)**

5.4 Stratified Experiment

We compare the geolocation performance for tweets with and without Location-Indicative (LI) words. We also examine if we can obtain meaningful geolocation accuracy for the latter. Such tweets were considered not appropriate for fine-grained geolocation and excluded in an earlier work [19].

For determining LI words, we implement the approach in [19]: LI words have high occurrence probability in at least one venue and occur at relatively few venues. Following [19], we score words as $\max_v\{p(w|v)\log(\frac{V}{df(w)})\}$. Instead of specifying dataset dependent thresholds, we designate the top 5% scoring words as LI words. Our experiment setup is similar as before, except that test tweets are now stratified into tweets with/without LI words. We compute MRR and VMRR for each group of test tweets. Due to space constraints, we only display the results for selected models on the Singapore datasets. The observations for Jakarta datasets are similar.

In both Tables 11 and 12, tweets with LI words are easier to geolocate. All three models obtain higher MRR and VMRR values for such tweets. Consistently over both tweet types, LWQE and LWQE-LW-CF outperform Nb with statistical significance. LWQE-LW-CF is the best performer, while LWQE also improves over Nb.

Assuming that tweets without LI words are indeed noise, then geolocating such tweets will result in candidate venues being randomly ranked. Based on random ranking, we can derive the expected reciprocal rank per tweet as $E_r(RR) = (1/V)\sum_{i=1}^{V}(1/i)$. This leads to an expected random ranking MRR value of 0.0011 for Table 11 and 0.0032 for Table 12. Interestingly even for tweets with no LI words, all models obtain much higher ranking accuracy than the MRR values expected from random ranking. Thus, one may not wish to regard such tweets as purely noise, as advocated in [19].

Table 11: Results on SG-SHT. MRR^{LI}=MRR for test tweets with LI words. $MRR^{\neg LI}$=MRR for test tweets without LI words. Notation is similar for VMRR. On average per run, there are 2194.8 test tweets with LI words and 1053.7 without. V=9209.1

Models	MRR^{LI}	$VMRR^{LI}$	$MRR^{\neg LI}$	$VMRR^{\neg LI}$
Nb	0.1222	0.0420	0.0514	0.0340
LWQE	0.1296	0.0488	0.0586	0.0381
LWQE-LW-CF	0.1303	0.0486	0.0648	0.0403

Table 12: Results on SG-TWT. Notations as in Table 11. On average for each run, there are 281.5 test tweets with LI words and 768.4 without. V=2672.5

Models	MRR^{LI}	$VMRR^{LI}$	$MRR^{\neg LI}$	$VMRR^{\neg LI}$
Nb	0.1093	0.0475	0.0401	0.0159
LWQE	0.1148	0.0530	0.0462	0.0197
LWQE-LW-CF	0.1164	0.0537	0.0496	0.0209

5.5 Case Studies

We now present some example cases to show the effects of using LW and QE. Table 13 displays sample test tweets from SG-SHT, where geolocation is improved by location-indicative weighting, i.e. the model LW. Also displayed is the change in reciprocal rank ΔRR_{LW}, which is computed as $\Delta RR_{LW} = \frac{1}{(r_{LW}+1)} - \frac{1}{(r_{Nb}+1)}$, where r_X denotes the ranked position of posting venue under the model X. Note that the best possible ranked position is 0.

Within each test tweet, modeled words are italicized and sized proportionately to their assigned location-indicative weights. For example in tweet S1, LW assigns largest weights to the words 'Tallest' and 'Balloon'. These are words that appear in relatively fewer venues and are more location-indicative. Compared to not weighting the words, reciprocal rank improves by 0.163, due to the ranked position of the posting venue being elevated from 26 to 4.

Similarly, tweet S2 is better geolocated due to the emphasis on location-indicative words. S2 is posted from a parade preparation venue. 'Chingay' refers to an annual parade event held in the city area of Singapore, thus the word is highly indicative of venues associated with the parade. In S3, emphasizing 'Karaoke' increases the probabilities for venues providing such entertainment activity. Since karaoke venues are relatively few in number, the actual karaoke venue of the test tweet is elevated in rank.

Table 14 displays sample test tweets from SG-SHT, where geolocation is improved by query expansion, i.e. the model QE. The user posting tweet S4 had also visited the posting venue multiple times. On its own, S4 is not informative since there are many dining venues where one can have breakfast. However on another visit to the same venue, the user mentioned having "Nasi Lemak" [3] for

[3]Malay name for a rice dish cooked with coconut milk

Table 13: Sample test tweets from SG-SHT to illustrate location-indicative weighting. Modeled words are italicized and sized proportionately to their assigned weights. r_X denotes the ranked position of posting venue under the model X. ΔRR_X=change in reciprocal rank incurred by model X over the Nb model.

		ΔRR_{LW}	r_{Nb}	r_{LW}
S1	"*Singapore*'s *Tallest Balloon* Sculpture."	0.163	26	4
S2	"*Chingay work* last *day*"	0.321	83	2
S3	"*Morning Karaoke*?"	0.389	8	1

Table 14: Sample test tweets from SG-SHT. Below each tweet, we list up to 5 added words that are most related to the query, along with their relatedness score. Notations as in Table 13.

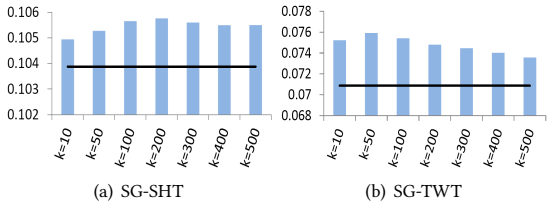
		ΔRR_{QE}	r_{Nb}	r_{QE}
S4	"Breakfast!" (teddy,0.120), (buying,0.104), (lemak,0.085) (nasi,0.085), (prata,0.070), ...	0.046	75	16
S5	"2nd time spiderman2" (captain,0.25),(america,0.25)	0.008	24	20

breakfast. This is a dish which the test venue is popular for, resulting in the ranking improvement. The last tweet S5 is associated with functionally related visits (Section 3.2), instead of repeated visits. The user visited the posting venue (a movie theatre) once to catch the movie "Spiderman", and another theatre to catch "Captain America". Due to query expansion, the latter's title words are added to S5. In this case, the test venue screens "Captain America" as well. Thus the added words are relevant although they arise from a different venue. This improves geolocation since the expanded tweet now describes venue characteristics more effectively.

5.6 Parameter Sensitivity Studies

In the collaborative filtering models, we propagate visitation information from the test user's k nearest neighbors, where similarity is based on content history. In our experiments, we used $k = 500$. In this section, we show that ranking accuracy is insensitive to k.

We conduct sensitivity studies on both Singapore and Jakarta datasets. Due to space constraints, we illustrate only the results for Singapore. As VMRR exhibits similar robustness to MRR, we plot only the results for the latter in Figure 2. In the figure, the bars are MRR values for the model LWQE-LW-CF with different k values. For comparison, we also plot the MRR value of LWQE as a horizontal line. Recall that LWQE is a model that performs well, but does not include any collaborative filtering. Both figures show that MRR is not sensitive to the value of k, remaining in a narrow band as we vary k from 10 to 500. For all k values, LWQE-LW-CF also consistently gives higher MRR than LWQE which is reassuring. Thus collaborative filtering easily improves ranking accuracies without much tuning of k.



(a) SG-SHT (b) SG-TWT

Figure 2: MRR (Y-axis) with different k values (X-axis) for LWQE-LW-CF. Horizontal line = MRR for LWQE.

6 RELATED WORK

Tweet geolocation. There are fine-grained and coarse-grained tweet geolocation. The latter is further differentiated by the task of geolocating individual tweets or inferring the home city/region of users from their tweets. For the latter task, [7] proposed a novel spatial model while [5] used Gaussian Mixture Models (GMM). Both works exploited spatially focused words in a user's tweets to infer his home city/region. Other spatial models are extensively compared in [12]. There are also ensemble classification [25] and label propagation [16] approaches. The latter propagates location labels over relationship graphs. This assumes that a user is likely to be near his friends' home locations. [22] infers users' home cities using the idea that a user's home location affects his follower/followee relationships and what venues he explicitly mention in his tweets. They proposed spatial influence models for users and venues based on Gaussian distributions. For geolocating individual tweets, [2, 13] infer region-indicative geographical topics. Each tweet is then geolocated based on its inferred topic. GMMs have also been used [26] to detect location-indicative n-grams to geolocate tweets. We also note works by [14, 17] which fits language models to both tweets and locations. For a test tweet, divergences are then computed between the language models of a test tweet and various locations, to be used for ranking candidate venues.

In contrast to coarse-grained geolocation, we work on fine-grained geolocation which links tweets to specific venues. For this, language models are also applicable [19, 23]. Our baseline includes [19] which models each venue with a naive Bayes model. In [23], external content information from web pages and tweet posting time are included for modeling the ten most popular venues per city are geolocated. Our experiments are much more comprehensive as we model tweets from thousands of venues. We also note the work by [4], which does feature engineering with content, visitation history and relationships. They used Foursquare-specific features, e.g. venue categories, user mayorships etc. Our work seeks to develop a more general approach that relies less on platform specific features. [30] proposed a topic model which utilizes tweet content, visitation history and posting time information. Since our test users do not have visitation history, modeling their personal regions as required by the model is not possible. We also expect that our approaches can be further improved with the additional aspect of posting time information and intend to make further comparisons. Finally works by [15, 21] geolocate venue mentions in tweets. Although colloquial mentions are handled, relying on mentions is a bottleneck, e.g. a tweet 'safely landed' is indicative of the airport although it has not mentions. In our work, we geolocate all tweets even if no mentions exist.

Query Expansion. Query expansion is typically used for document retrieval. The initial query is expanded by adding potentially relevant words using term weighting schemes. To enhance query expansion, [9] used genetic programming to learn weighting schemes. [29] compared local and global query expansion techniques. Global techniques [27] used corpora-wide word occurrences and relationships to expand queries while local techniques considers the top ranked documents retrieved for the original query. An example of a global technique [27] uses a similarity thesaurus to add words that are most similar to the query concept. Applying query expansion to tweets, [3] retrieved relevant tweets given a user query. Given initial key words, they retrieved web pages and used their titles as a source of expansion words to retrieve more tweets. [11] worked on a different task of retrieving more relevant keywords related to natural hazard events, e.g. considering candidate words from tweets close in space and time to an event-related tweet.

Instead of document retrieval, we use query expansion for the different purpose of improving tweet geolocation. Given the local/global categorization, our query expansion here can also be viewed as a partially global technique, in that we consider user-specific portions of the tweet corpora.

Collaborative Filtering. Collaborative filtering is widely used in recommender systems [1], with techniques ranging from nearest neighbor techniques, matrix factorization [18] to topic models [28]. The venues in location based social network are analogous to items, leading to collaborative filtering approaches for venue recommendation. [6, 24] designed matrix factorization approaches while [20] proposed a co-clustering approach which clusters users, locations and activities, i.e. GPS trajectory segments. Note that these are recommendation models requiring users' visitation history. In contrast, our collaborative filtering approach is tailored for fine-grained geolocation, not venue recommendation. We also focus on the scenario where tweets targeted for geolocation are posted by users with no visitation history.

7 CONCLUSION

We have proposed a novel model to geolocate tweets at fine granularities to specific venues. Our model is widely applicable in Twitter where many users post frequently but neglect to geocode any of their tweets. Such users have rich content history, but no visitation history. In particular, we achieve better geolocation of these user's tweets by exploiting three types of signals from locations, users and peers. Our model aspects capture the signals based on intuitive ideas. Firstly through location-indicative weighting, we place more importance on words that are indicative of venues. Secondly through query expansion, we add potentially informative words to test tweets before geolocation. Lastly, we use collaborative filtering to propagate visitation information from users with visitation history to those without. Our best model combines all three aspects.

8 ACKNOWLEDGMENTS

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative, and DSO National Laboratories.

REFERENCES

[1] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE TKDE* (2005).
[2] Amr Ahmed, Liangjie Hong, and Alexander J. Smola. 2013. Hierarchical Geographical Modeling of User locations from Social Media Posts. *WWW* (2013).
[3] Ayan Bandyopadhyay, Mandar Mitra, and Prasenjit Majumder. 2011. Query Expansion for Microblog Retrieval. *TREC* (2011).
[4] Bokai Cao, Francine Chen, Dhiraj Joshi, and Philip S. Yu. 2015. Inferring crowdsourced venues for tweets. *Big Data* (2015).
[5] Hau-Wen Chang, Dongwon Lee, Mohammed Eltaher, and Jeongkyu Lee. 2012. @Phillies Tweeting from Philly? Predicting Twitter User Locations with Spatial Word Usage. *ASONAM* (2012).
[6] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. 2012. Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. *AAAI* (2012).
[7] Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: a content-based approach to geo-locating twitter users. *CIKM* (2010).
[8] Wen-Haw Chong, Bing Tian Dai, and Ee-Peng Lim. 2015. Prediction of Venues in Foursquare Using Flipped Topic Models. *ECIR* (2015).
[9] Ronan Cummins. 2008. The Evolution and Analysis of Term-Weighting Schemes in Information Retrieval (Doctoral dissertation). (2008).
[10] J.T.A.S. Ferreira, D.G.T. Denison, and D.J. Hand. 2001. *Weighted naive Bayes modelling for data mining.* Technical Report. Department of Mathematics, Imperial College.
[11] Víctor Fresno, Arkaitz Zubiaga, Heng Ji, and Raquel Martínez-Unanue. 2015. Exploiting Geolocation, User and Temporal Information for Natural Hazards Monitoring in Twitter. *Procesamiento del Lenguaje Natural* 54 (2015).
[12] Bo Han, Paul Cook, and Timothy Baldwin. 2014. Text-based Twitter user geolocation prediction. *JAIR* (2014).
[13] Liangjie Hong, Amr Ahmed, Siva Gurumurthy, Alexander Smola, and Kostas Tsioutsiouliklis. 2012. Discovering geographical topics in the Twitter stream. *WWW* (2012).
[14] Yohei Ikawa, Miki Enoki, and Michiaki Tatsubori. 2012. Location inference using microblog messages. *WWW (Companion Volume)* (2012).
[15] Zongcheng Ji, Aixing Sun, Gao Cong, and Jialong Han. 2016. Joint Recognition and Linking of Fine-Grained Locations from Tweets. *WWW* (2016).
[16] David Jurgens. 2013. That's What Friends Are For. Inferring Location in Online Social Media Platforms Based on Social Relationships. *ICWSM* (2013).
[17] Sheila Kinsella, Vanessa Murdock, and Neil O'Hare. 2011. "I'm eating a sandwich in Glasgow": modeling locations with tweets. *SMUC* (2011).
[18] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *IEEE Computer* (2009).
[19] Kisung Lee, Raghu K. Ganti, Mudhakar Srivatsa, and Ling Liu. 2014. When Twitter meets Foursquare: tweet location prediction using Foursquare. *MobiQuitous* (2014).
[20] Kenneth Wai-Ting Leung, Dik Lun Lee, and Wang-Chien Lee. 2011. CLR: a collaborative location recommendation framework based on co-clustering. *SIGIR* (2011).
[21] Chenliang Li and Aixing Sun. 2014. Fine-grained location extraction from tweets with temporal awareness. *SIGIR* (2014).
[22] Rui Li, Shengjie Wang, Hongbo Deng, Rui Wang, and Kevin Chen-Chuan Chang. 2012. Towards social user profiling: unified and discriminative influence model for inferring home locations. *KDD* (2012).
[23] Wen Li, Pavel Serdyukov, Arjen P. de Vries, Carsten Eickhoff, and M. Larson. 2011. The where in the tweet. *CIKM* (2011).
[24] Xutao Li, Gao Cong, Xiaoli Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. 2015. Rank-GeoFM: A Ranking based Geographical Factorization Method for Point of Interest Recommendation. *SIGIR* (2015).
[25] Jalal Mahmud, Jeffrey Nichols, and Clemens Drews. 2012. Where Is This Tweet From? Inferring Home Locations of Twitter Users. *ICWSM* (2012).
[26] Reid Priedhorsky, Aron Culotta, and Sara Y. Del Valle. 2014. Inferring the Origin Locations of Tweets with Quantitative Confidence. *CSCW* (2014).
[27] Yonggang Qiu and Hans-Peter Frei. 1993. Concept based query expansion. *SIGIR* (1993).
[28] Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. *KDD* (2011).
[29] Jinxi Xu and W. Bruce Croft. 1996. Query expansion using local and global document analysis. *SIGIR* (1996).
[30] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. 2013. Who, where, when and what: discover spatio-temporal topics for twitter users. *KDD* (2013).
[31] Nayyar A. Zaidi, Jesús Cerquides, Mark James Carman, and Geoffrey I. Webb. 2013. Alleviating naive Bayes attribute independence assumption by attribute weighting. *JMLR* (2013).