Location Prediction in Social Networks

Rong Liu[1], Guanglin Cong[1], Bolong Zheng[2,3], Kai Zheng[1], and Han Su[1(✉)]

[1] Big Data Reaserch Center,
University of Electronic Science and Technology of China, Chengdu, China
{kaizheng,hansu}@uestc.edu.cn, lrong0913@gmail.com, nofloat@163.com
[2] School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
zblchris@gmail.com
[3] Department of Computer Science, Aalborg University, Aalborg, Denmark

Abstract. User locations in social networks are needed in many applications which utilize location information to recommend local news and places of interest to users, as well as detect and alert emergencies around users. However, considering individual privacy, only a small portion users share their location on social networks. Thus, to predict the fine-grained locations of user tweets, we present a joint model containing three sub models: content-based model, social relationship based model and behavior habit based model. In the content-based model, we filter out those location-independent tweets and use deep learning algorithm to mine the relationship between semantics and locations. User trajectory similarity measure is used to build a social graph for users, and historical check-ins is used to provide users' daily activity habits. We conduct experiments using tweets collected from Shanghai during one year. The result shows that our joint model perform well, especially the content-based model. We find that our approach improves accuracy compared to the state-of-the-art location prediction algorithm.

1 Introduction

Since the on-line social media grows, Twitter, Facebook and Sina Weibo have accumulated a large number of users up to now. In China, Sina Weibo, a form of unstructured short texts, has become one of the most popular social networking tools. In Sina Weibo, people post tweets about their daily routines, emergencies they meet, and comments to news. They also attach to their tweets with current locations, a.k.a. check-ins. Check-ins play an important role in location based recommendation and emergency detection/alert, which are utilized by a large number of business organizations. For example, when a user comes to a place and posts her location, she can get recommendation about local news and places of interest around, and also get alerts of unexpected events nearby. But recently, due to concerns about data privacy, weibo users have been increasingly avoiding sharing location information while posting tweets. According to a recently statistical analysis in [9] over 1 billion tweets spanning three months, only 0.58% tweets have location tags. It is becoming harder and harder for business organizations to extract user locations, hindering recommendation and detection.

Y. Cai et al. (Eds.): APWeb-WAIM 2018, LNCS 10988, pp. 151–165, 2018.
https://doi.org/10.1007/978-3-319-96893-3_12

In this paper, we present a novel approach which combines three features including textual content, social relationships and user behavior habits to predict user's current locations for tweets without any location tags. Recent works only consider the first two features, while ignore users' behavior habits. However, based on our study in our work, users' behavior habits play an important role in tweets location prediction. Actually, a Weibo user with a regular everyday life, will have similar daily activities. That is to say, his or her trajectories are similar. In content based model, we leverage Convolutional Neural Network (CNN) to mine location information in tweets. And we also mine another social relationship called user similarity in social relationship based model which cluster users with similar trajectories. Based on behavior similarities, we build a similarity graph which cluster similar users together to help find users with similar daily behavior for a specific user. Through a probabilistic model, we predict a user's location from his or her similar users.

The remainder of this paper is organized as follows. Section 2 overviews related works. The location prediction model is introduced in Sect. 3. Section 4 describes the experiments conducted to verify the accuracy of our model. Finally, Sect. 5 draws the conclusion of the paper.

2 Related Work

With the wide use of social networks, mining user location information from them and apply this to many occasions is significant, such as location based recommendation, emergency detection and alert. Thus, many related works utilized different features and approaches to roughly predict where the users were when they post tweets in their personal devices. The features used in previous works can be categorized into two types: content based and social relationship based.

Content Based. User's tweets content often contains many features, such as textual content, photos, videos and user URLs. [15] generates probabilistic language model based on the photo tags posted by users, and then estimate the location of each photo rely on the language model and Bayesian inference. Comparing with photos, textual content often contains more location clues, since users may mention location names or location related words when posting tweets.

Location prediction approaches based on text are classified to two basic types as well. One is identifying the related geographic terms from textual content, the other is building a probabilistic language model to predict tweets locations. For the reason that a small number users post exact geographic terms in their tweets, most recent works prefer to construct probabilistic models for location prediction based on the statistical linguistic features in textual content. In [1], author uses a variation of probabilistic framework in [2] which adds the feature of relationship between tweets and related reply-tweets, in order to enhance accuracy by estimating the geographic location of the user. [16] proposes a probabilistic model leveraging the Maximum Likelihood Estimation to infer users resident locations, which mines the relationship between locations and words.

Some related works are dissatisfied with such coarse-grained location prediction. In [9], Lee et al. utilize external location sharing services platforms, and the user 'Check-In' information to study the mobility characteristics of the users. This work builds language model for each PoI (Place of Interest) which is the basis of location prediction. However, the cost of building language models is huge, and they just predict site located in a part of city. Our work keeps the idea of mining the relationship between the semantics of tweet content and the locations, but builds language model for all locations through a novel approach.

Graph Based. Social relationship is one of the most essential part of on-line networks. Friendship as a kind of social relationship, always provides pivotal clues for predicting user locations. The way of building user friendship network is usually utilizing users profile, response and dialogues. Backstrom et al. propose a probabilistic model representing the likelihood of relationship between any two users in [3]. Based on this model, user locations can be inferred when given the geographic distribution of the locations. [8] presents several extensions to the model shown in [3], which adding weighting strategies that user friends have different influence on user. In [14], Sadilek et al. add up the time overlaps two users spend at their respective locations and scale each overlap by distance between the locations. Thus, distance value can be used for detecting friendship between users and representing the tightness of this relationship. Friends may stay in the same city in most situations, however, they may not always stay at the same site or regions in the city all times. So we define another social relationship called user similarity to solve this problem. Thus, user tweets can be regarded as a trajectory with timestamps and coordinates. And we leverage trajectory similarity algorithm shown in [4,5,13] to calculate the user similarity.

3 Problem Formulation

User location information in social media plays an important role in many applications, however, only a small portion users share their location for protecting privacy. Our goal is to estimate the location based on features that are minded from tweets which are lack of check-ins. Sina Weibo, one of social applications with a large number of users, provides users a lot of choices that they can post tweets containing textual data, photos and videos, interact with other users or record their lives. Tweets content, especially textual content, often contains location names or location related words, which can extract location related information directly. Trajectory similar users can be clustered through user similarity graph, they probably share the same location at most circumstances. Besides, when users record their daily lives, we can extract their behavior habits through these records. So except check-in data, we can mine user location information through other methods as well. In this work, users tweets content, social relationships and user behavior habits are used to get location information.

Location Estimation Problem: Given a set of tweets $T_{tweets}(u)$ posted by a Weibo user u, estimate a user's probability of being located at a site, such

that the location with maximum probability $l_{cur}(u)$ is the user's actual location $l_{act}(u)$.

With the definition of the problem solved in this paper, we list the Notations in Table 1 used throughout the remainder of this work.

Table 1. Notations used in the paper

Notation	Explanation	Notation	Explanation
U	User set of u_i	$e(u_i, u_j)$	Relationship between u_i and u_j
E_i	Relationship set of u_i	A	Tweet matrix constructed with word vector
$T_{tweets}(u)$	Tweet set of u	X_i	The i-th word vector
$S_{words}(u)$	Words set of u's tweets	$\theta_{similar}$	Trajectory similarity threshold
T	A trajectory	$s(T_i, T_j)$	Similarity between two trajectories T_i and T_j
p	A trajectory point	l_{pre}	Location of previous tweet
V	Similar user set	l_{cur}	Predictive location of tweet
Vct	Region vector	l_{act}	Actual location of tweet
N	Region number of city	M	User location transition matrix
L	Candidate location set	P	Location transition probability matrix

As we noted that location estimation is a difficult and challenging problem. The check-in data in users' tweets is always sparse, and the frequency of user posting tweets is not high. So we divide the map of the city into square grids of different degrees to overcome the sparsity of locations, which are described in Sect. 4.2.

3.1 System Architecture

In this work, we propose a joint probabilistic model which contains the three sub models. Figure 1 provides a sketch of our system architecture for predicting the city area which the tweet belongs to. In this architecture, we define three channels to mine location information:

(1) **Textual Content**: Since users may post their locations or location related words in their tweets, the textual part of tweets becomes the most important clue of location prediction. We extract the textual part of tweets in dataset, filter out tweets without any location clues and train CNN model to predict tweet location based on these textual data.
(2) **Social Relationship**: On-line social relationship has different definitions in this work, we define it as user similarity for the reason that it has stronger connection on location than friendship mentioned in related works. User historical data can be regarded as a trajectory and used to calculate the similarity between users.

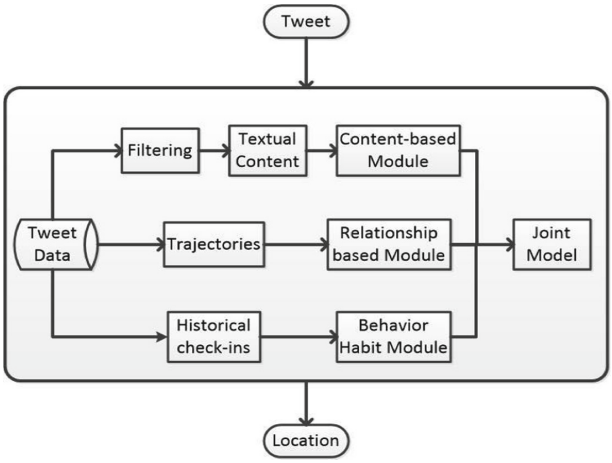


Fig. 1. Framework architecture

(3) **Behavior Habit**: Behavior habit of users provides clues of tweet location for the fact that users prefer to take their own familiar routes. User behavior habits are extracted from historical data as well. From these habits, we know how a user is moving between his or her resident locations.

Based on these features which we considered as location dependent information, we also present corresponding models to predict users' current locations. Specifically, CNN model is used to mine location information from $n-gram$ words of a tweet, user similarity cluster model finds similar users of a specific user and then follows them to where the user is, Markov Chain and Transition Probability Matrix build a customary trajectory for each user based their historical check-in data.

3.2 Content-Based Model

Textual content is the most frequently-used feature, since users may mention location names or location related words while posting tweets. However most words are distributed consistently with the population across different locations, meaning that most words provide little power at distinguishing the location of a user. For example, any user may post tweets like "I'm eating dinner", so tweets like this are called location-independent tweets. Without filter, many noise tweets in dataset increase the difficulty of extracting and distinguishing the location feature for our model. So we utilize $tfidf$ Value to evaluate whether a word is related to a location and filter out location-independent tweets without these location related words firstly. Besides, we use grid-based neighborhood smoothing approach which clusters locations into grids according to their coordinates, to overcome the sparsity of location across tweets in dataset. Thus we divide the entire city into equal-sized grid cells, which applies in the whole work.

Under this circumstance, a novel approach, Convolutional Neural Network (CNN), is used to mine the relationship between textual data and locations avoiding complex feature extractions and data reconstruction process comparing prior works. To get better training effect, traditional content-based models always need multiple parameter adjustments, while we just need pre-training word vectors in CNN model whose parameters are adjusted through backpropagation algorithm. In addition, CNN can extract information from different n-gram words sequence at the same time and is more suitable for large-scale data processing. Thus, we present a CNN architecture for tweets location prediction based on the model in [6] with a slight variance.

Firstly, when given a text portion with several sentences of a tweet, we segment these sentences into tokens which then are converted to a $TweetMatirx$ $A \in \mathbb{R}^{n \times d}$. Suppose that X_i is the $d-$dimensional word vector for the $i-$th word in n words tweet and the text portion of any user's tweet can be also descried as a matrix $X_{1:n}$. Then we define $X_{i:i+j}$ as the cascade of words $X_i, X_{i+1}, \cdots, X_{i+j}$ in tweets. To extract feature in tweet, a $filter$ $W \in \mathbb{R}^{h \times d}$ applied to a window of h words is used in convolution operation. For example, given a tweet with n words, a fixed $filter$ W and filter width h, a new feature c_i generated from sub-matrix $X_{i:i+h-1}$ by Eq. 1. Here $b \in \mathbb{R}$ is the bias term and f is an activation function such as the hyperbolic tangent (tanh) or Rectified Linear Units (ReLu).

$$c_i = f(W \cdot X_{i:i+h-1} + b) \tag{1}$$

Then, a feature map $\boldsymbol{c} = [c_1, c_2, \cdots, c_{n-h+1}]$, with $\boldsymbol{c} \in \mathbb{R}^{n-h+1}$, are produced by each possible sub-matrix of the given tweet $\{X_{1:h}, X_{2:h+1}, \cdots, X_{n-h+1:n}\}$. A maximum value $\hat{c} = max\{\boldsymbol{c}\}$ is taken as the feature corresponding to this particular filter after applying this max-pooling operation. This representation is then fed through a softmax function to generate the final classification. During the training process, the purpose is minimizing categorical cross-entropy loss, and optimizing the parameters including weight vectors for filtering and biases in activation function.

3.3 Social Relationship Model

Although location information can be extracted from textual data of tweets in most cases, users may not always post tweets containing hints about locations. For example, the location-independent tweets like "I'm eating dinner" are not suitable for the CNN model. By mining the social relationship of users, we use the location of the neighbors in users similarity graph to predict their current locations. Related works define user social relationship as friendship through user interactions in social applications. However, not all users may share the closely similar trajectory with their on-line friends even they are off-line friends at all times. We can just estimate which city a user locate in through friendship, but not her real-time locations in the city. Thus, we define another on-line social relationship as similar users who have similar trajectories. For instance, some users work in a same area such as same office buildings may have similar daily

activities but not friends whether in real life or on the Internet. Thus, we can predict a user's current location according to her similar users' locations instead of her friends' locations. The core components include the similar user clustering model and location prediction model that are described in the next part.

Similar User Clustering Model. In our work, we present an novel approach to cluster similar users based on the fact that they share nearly the same trajectories. Next, we define the notations used in this part and the operations on them.

Definition 1 (Trajectory Point). *A trajectory point is a pair:* (p, t)*, where* p *is a location in d-dimensional space, and* t *is the timestamp at which* p *is observed.*

In this model, users move in a two-dimensional space, that is, p is a 2-dimensional vector, and the time attribute is discrete.

Definition 2 (Trajectory). *Trajectory* T *is a sequence of trajectory points, extracted from user's check-ins and ordered by timestamps* t*. Trajectory* T *is represented as a sequence of trajectory sample points. Therefore,* $T = [(p_1, t_1), (p_2, t_2), ...,(p_n, t_n)]$*, where* $(t_1 < t_2 < ... < t_n)$*.*

With the definition of user trajectory, we define some operations on the point p and trajectory T.

1. $s(T_1, T_2)$**:** $s(T_1, T_2)$ represents the similarity rate of two users' trajectories T_1 and T_2.
2. $Head(T)$**:** For trajectory $T = [p_1, p_2, \cdots, p_n]$, $Head(T)$ is to get the first point of the trajectory, that is $Head(T) = p_1$.
3. $Time(p_1, p_2)$**:** $Time(p_1, p_2)$ represents the time difference of points p_1 and p_2.
4. $Rest(T)$**:** For trajectory $T = [p_1, p_2, \cdots, p_n]$, $Rest(T)$ is to get the tail points of the trajectory except the first point. Therefore, $Rest(T) = p_2, p_3, \cdots, p_n$.

In order to calculate the similarity of user trajectories, we use Spatial-Temporal Longest Common Subsequence Similarity (STLCSS) measure. This measure involved two constants:

1. δ: a real number which controls how far in time we can go in order to match a given point from one trajectory to a point in another trajectory.
2. ε: a real number that is the matching threshold. Only when the distance between two points is less than ε, can these two points be regarded as the same point.

$$s_{\delta,\varepsilon}(T_1, T_2) = \begin{cases} 0 & \text{if } T_1 \text{ or } T_2 \text{ is empty} \\ 1 + s_{\delta,\varepsilon}(Rest(T_1), Rest(T_2)) & \text{if } |Head(T_1) - Head(T_2)| < \varepsilon \text{ and} \\ & |Time(Head(T_1), Head(T_2))| \leq \delta \\ \max\left(s_{\delta,\varepsilon}(Rest(T_1), T_2), s_{\delta,\varepsilon}(T_1, Rest(T_2))\right) & \text{otherwise} \end{cases} \tag{2}$$

Given δ and ε, we define the similarity measure $s(T_1, T_2)$ between two trajectories T_1 and T_2, shown as follows:

$$s(T_1, T_2) = \frac{s_{\delta,\varepsilon}(T_1, T_2)}{\min(n, m)} \tag{3}$$

It is observable that the larger the value of $s(T_1, T_2)$ is, the more similar two trajectories are according to the Eq. 3. So if the similarity of two user trajectories is high, they are deemed to be similar users. Based on this conclusion, we can infer each other's positions when no location clues in user's tweets, which is shown in following part. Here we define a parameter $\theta_{similar}$ as a threshold to judge whether the two trajectories are similar. That is to say, if the similarity $s(T_i, T_j)$ between trajectories of u_i and u_j exceeds $\theta_{similar}$, they are similar users to each other. So the influence $e(u_i, u_j)$ between two users is described by their trajectory similarity $s(T_i, T_j)$.

Location Prediction Model. The goal of Location prediction Model is to infer the most likely location of user u while posting a tweet. Based on Similar User Clustering Model, we get any user u_i's personal similarity graph E_i which contains the influences on her similar users. For example, the influence of user u_i on u_j as well as u_j on u_i is described as $e(u_i, u_j)$ that can also be regarded as the weights of user similarity. Since the higher the similarity of user trajectories is, the greater the influence they have on each other.

We firstly define similar user list of u as $V_u = \{v_1, v_2, \cdots, v_m\}$, the location set of user u's similar users as $L = \{l_1, l_2, \cdots, l_m\}$ and the influence on u as $E_u = \{e(v_1, u), e(v_2, u), \cdots, e(v_m, u)\}$. Then we define $d(l_i, l_j)$ as the Euclidean distance between location l_i and l_j, and $t(v_i, u)$ as the time difference between tweets posted by u and v_i . In this model, user u's current location l_{cur} can be estimated through u's previous location l_{pre} and location list L of his similar users during this period, and the weights list E_u of u.

$$p(l_i|u) = \begin{cases} [1 - \frac{d(l_i, l_{pre})}{\sum_{j=1}^{m} d(l_j, l_{pre})}] \times e(v_i, u) & \text{if } d(l_i, l_{pre}) \leqslant \varepsilon \text{ and } t(v_i, u) \leqslant \delta \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Equation 4 shows the probability of user u appearing at i-th similar user's location l_i. Since user's moving distance during a set time period is limited, the closer the distance between u and his similar user v_i, the higher the probability that u is at l_i. There are two kinds of measures to finally estimate user's actual location: one is obtaining location with the top one probability, another is gaining the locations whose probability rank in the top $k(k<m)$.

$$l_{cur} = \underset{l_i \in L}{argmax}\, p(l_i|u) \tag{5}$$

Here we choose the first measure which deems the site l_{act} with the highest probability among the candidate locations collected from similar users as user's current location.

3.4 Behavior Habit Model

Although most users' current location can be predicted through tweets content and their similar users, there are still part of users who have few similar users and dislike to post textual content. For example, a user may have relatively stationary trajectory. Therefore, we use user behavior habits based on the supposition that users have their own daily behavior habits to predict their locations. Thus, a location point can be regarded as a state of user. When given a list of states, we can use Markov chain introduced in [12] to predict the next state on the basis of Markov property that the current state is only related to the previous state but not to the earlier states. Then we construct two matrices extended from [10] to depict user behavior habits. For overcoming the sparsity of tweets across location, the city is divided into equal-sized grid cells which represent the regions of city in this subsection.

Definition 3 (Location Transition Matrix). *The location transition matrix* $M_i \in \mathbb{R}^{N \times N}$ *of user* v_i*, where* N *refers to the number of regions divided from the city. Any element of this matrix* $M_i(r, c)$ *represents the frequency that* u_i *transferred from region* r *to region* c*.*

Definition 4 (Region Vector). *The region vector* Vct_i *of user* v_i *is a* N*-dimension vector. Element* $Vct_i(r)$ *refers to the number of trajectories that* v_i *transfers from region* r *to other regions.*

Definition 5 (Location Transition Probability Matrix). $P_i(r, c)$ *in this matrix* $P_i \in \mathbb{R}^{N \times N}$ *is the probability of transferring to region* c *when the current position of user* v_i *is region* r*.*

According to the Location Transition Matrix M_i and Region Vector Vct_i, we can get the Location Transition Probability Matrix rely on the Eq. 6.

$$P_i(r, c) = M_i(r, c)/Vct_i(r) \tag{6}$$

Then when v_i's previous location belongs to region c, her current location is calculated as:

$$l_{cur} = \underset{j \in N}{argmax}\, P_i(r_j, c) \tag{7}$$

3.5 Joint Model

The framework architecture and sub models are introduced specifically in the previous subsections. These models leverage textual content, user social relationship and individual behavior habit to mine location information. For the purpose of building a fruitful content-based location prediction model, we use $tfidf$ Value to measure the influence of each word on the locations and define location related words whose $tfidf$ values exceed the threshold θ_{tfidf}. Depending on whether the textual content contains location related words, we determine if the location of a given tweet can be predicted using content-based model, since

almost no location features can be extracted from location-independent tweets. Then the locations of tweets which were filtered out can be predicted by a linear combination model that combines social relationship model and behavior habit model.

4 Experiments

In this section, we evaluate the location prediction framework presented in above section through a set of experiments. Our framework is built by three sub models: content-based model, relationship-based model and habit-based model. We report the dataset used in our work and define the general setup of models. Then we design a set of experiments to illustrate the prediction accuracy of different models. In addition, there are several thresholds in these models, such as ε and $\theta_{similar}$. So we also test how these parameters influence the result.

4.1 Datasets

It is not a difficult task to predict users resident cities for the reason that lots of city information can be extracted according to related works. So we can naturally suppose that we know which city the user belongs to. In this paper we gathered data from shanghai, China, since shanghai is one of the most densely populated cities of world whose population is more than 10 million and coverage area is nearly 6340 km^2. We collected about 90 million tweets from nearly 60 thousand users, but only 9 million tweets as well as 10% of the initial data are tagged. While some of them post few tweets or just post links or advertisements of other applications, which do few favor of predicting location. After removing these users and tweets, there are only 1036386 tweets from 10 thousand users left in the dataset.

4.2 Experiment Setup

To predict the fine-grained location of a tweet, we divide the entire city into equal-sized grid cells and each cell is labeled by its diagonal latitude/longitude coordinates. And those locations whose coordinates are falling into the same cell cluster into one category. In this part, we use a turning parameter $cellsize$ to control the granularity of city area division which also is regarded as the prediction error distance. And then we vary the parameter $cellsize$ form 1 to 15 km with the step of 5 km.

In order to evaluate the capability of our model, we calculate the accuracy (ACC) of prediction by:

$$ACC = \frac{|\{l_{cur}|l_{cur} = l_{act}\}|}{|l_{cur}|} \tag{8}$$

Here l_{cur} represents the location of a tweet predicted by our model, and l_{act} is the actual location of this tweet. Throughout our work, we set threshold θ_{tfidf} to

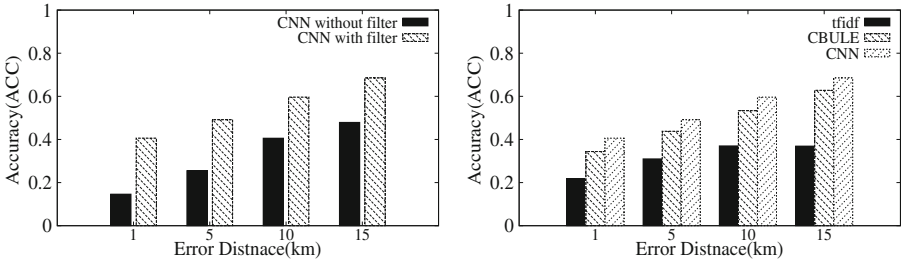
0.1 and filter out tweets without words whose $tfidf$ values exceed θ_{tfidf}. Thus, on the basis of whether mining location information from contents, we can divide the dataset into two parts: one is location correlation dataset and the other is location-independent dataset.

4.3 Capability of Content-Based Model

Data filter using $tfidf$ Value in our work is significant, because the more noise or location-independent tweets, the worse the effect of CNN model will be. So we filter out location dependent tweets utilizing the threshold θ_{tfidf}. After the data filter, we obtain the correlation dataset whose tweets directly or indirectly contain location information and use about 80% as the training set, the rest as the testing set. To build CNN model for location prediction, we firstly turn the check-ins of tweets to the corresponding grid cell, a classification label. And then we segment the textual part of tweets into tokens, remove stop words and punctuations in these tokens at the same time. After these operations, we get a set of words of each tweet and turn these words into word vectors based on the $word2vector$ model trained by Wikipedia Chinese corpus and incrementally trained by weibo corpus extracted from dataset. Thus, each tweet turns to an matrix whose rows represent word vectors with 60 dimension and the matrix composed by word vectors can be used as a word embedding in this model. What's more, to solve the problem of different length of tweet, we specify a maximum input tweet length and fill the part whose length is not enough with zero.

To illustrate the significance of data screening, we design experiments to compare the accuracy of this model using raw dataset and filtered dataset respectively. Meanwhile, to better illustrate the validity of our model, we compare other two approaches in related works [2,7] with our method. In [2], a probability model, Content-Based User Location Estimation (CBULE), based on maximum likelihood estimation, builds the probability distribution over regions in the city for each word in the dataset. [7] uses external location-specific data source Foursquare to train language model for each region and then uses $tfidf$ Value approach to predict user's current locations. Here we use the training set to build language model for regions. The prediction results on testing set are shown as Fig. 2.

Fig. 2(a) shows the prediction accuracy of our content-based model with and without data filter. The accuracy of model using filtered dataset is much higher than using row dataset, for data filter greatly reduces the ratio of location-independent tweets in dataset. The results of different content-based approaches are shown in Fig. 2(b). Using CNN model has better performance than using $tfidf$ Value and CBULE model. The prediction accuracy of CNN model is 40.63% within 1 km. What's more, when the error distance is increased by 5 km, the accuracy is raised by nearly 10%. Content-Based User Location Estimation also perform well within different error distances, but $tfidf$ Value measure can just reach 36.92% at maximum error distance. The probable reason of this result is that location related words in training set used to build language model (LM)



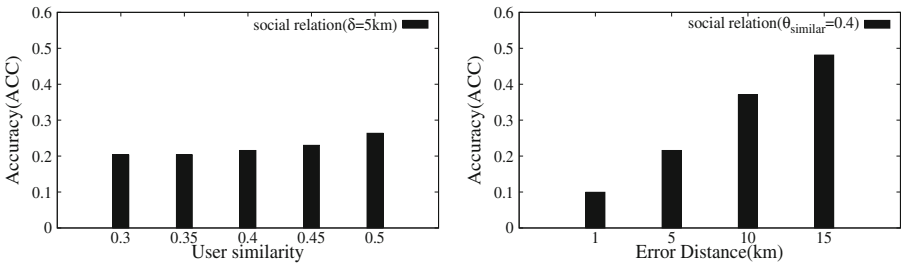
(a) Effect of data filter (b) Effect of different approaches

Fig. 2. The capability of content based model

are sparse across locations. Overall, CNN model can handle the data with sparse distribution of location related words better than other two models.

4.4 Capability of Relation Based Model

In this section, we predict a user's current location based on the location of his or her similar users' locations. For each user u, the first step is finding users who have similar trajectories with u. Here, we use Spatial-Temporal Longest Common Subsequence Similarity (STLCSS) to calculate the trajectory similarity between users according to Eq. 3 shown in Sect. 3.3. After this operation, the similarity graph among all users is built. For any user u in the dataset, his or her similarity graph can be consisted of $V_u = \{v_1, v_2, \cdots, v_m\}$ and the corresponding relationships are described as $E_u = \{e(v_1, u), e(v_2, u), \cdots, e(v_m, u)\}$, where $e(v_i, u)$ is equivalent to $s(T_i, T_u)$ that exceeds $\theta_{similar}$. To explain the influence of the similarity of users on the accuracy of location prediction, we define different threshold values: $\theta_{similar}$ which is set from 0.3 to 0.5 at the interval of 0.05. As with content-based model, we also test the influence of different error distance on prediction accuracy. The result is shown in Fig. 3.



(a) Effect of different $\theta_{similar}$ (b) Effect of user simlarity

Fig. 3. The capability of relation based model

Figure 3(a) shows that the accuracy of prediction is increasing gently with the interval of 0.05 of $\theta_{similar}$ when the error distance is 5 km. Although the accuracy rate shows the trend of overall rise, the increase is small. The reason is that there is a trade-off between the accuracy and the threshold $\theta_{similar}$. Namely, when $\theta_{similar}$ increases, a smaller number of similar users are selected for the prediction. So we fix the threshold $\theta_{similar}$ on 0.4 when testing the effect of error distance based on the result shown in Fig. 3(a). From Fig. 3(b), the prediction accuracy leveraging user similarity continuously increases with the raise of error distance. Because of the low frequency of user posting tweets and small scale of similar user sets in current dataset, the accuracy of this model is low at a very fine granularity.

4.5 Capability of Behavior Habit Based Model

User behavior habit is another important feature for predicting user location, since users always lead regular everyday lives that they have similar daily activities. So we split the whole dataset into two parts, the prior part as the training set and the rest part as testing set. In the training set, we obtain all users' historical data which contains time stamps and latitude/longitude coordinates. For the reason that users are more likely to move within certain regions, we cluster the user locations into regions according to the approach mentioned in Sect. 4.2. Then we will show the effect of different error distance leveraging behavior habit model.

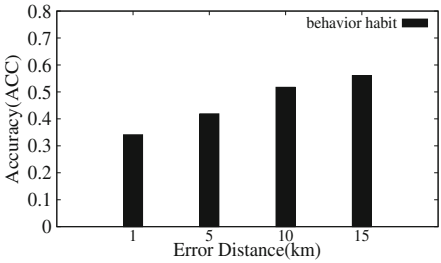
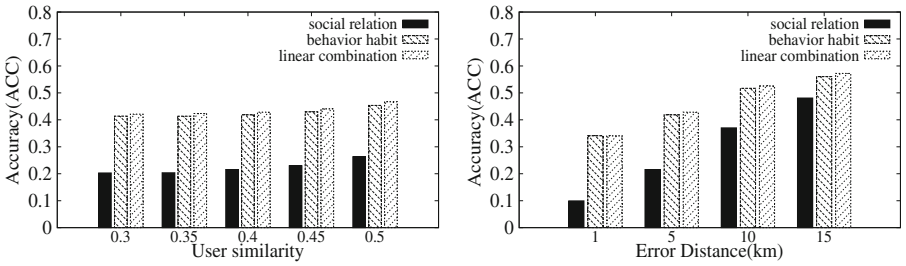


Fig. 4. The capability of behavior habit model

Figure 4 shows that the accuracy of this location prediction model increases with the raise of error distance and locates about 41% of predicted tweets within 5 km from their actual locations. Because users daily activity habits are always repeat nearly everyday. In addition, the larger error distance which also represents the grid cell size, the more locations clustered into a region and the higher accuracy of prediction is.

4.6 Capability of Linear Combination Model

In order to evaluate the effects of last two models, we build a linear combination model to combine these two features and predict tweets locations. Next, we test the effect of different user similarity threshold $\theta_{similar}$ and error distance on behavior habit based model and the combination model of this and social relation based model.



(a) Effect of different $\theta_{similar}$ (b) Effect of user simlarity

Fig. 5. The capability of behavior habit based model

Figure 5(a) shows that user behavior habit performs better than user social relationship in location prediction, and combining these two models can nearly double the accuracy of prediction leveraging the social relationship based model. Because almost all users' daily behavior habits are nearly settled, we mine a lot of location information from users' historical data. As with the reason mentioned in Sect. 4.4, the accuracy of combination model is a little higher than behavior habit based model separately used. In Fig. 5(b), the accuracy of behavior habit model increases with the raising of error distance. We also find that comparing the social relationship based model, behavior habit based model has obvious advantages in predicting fine-grained locations of tweets, while the advantages are gradually weakened in larger error distance. The probable reason is that the number of similar users increases as well as candidate locations in coarse-grained prediction.

5 Conclusion

We present a joint model for tweets location prediction which contains three sub models based on different features mining from tweets data. These models utilize textual content, social relationship and user behavior habit to extract location information, and obtain high prediction precision. From the experimental results, we conclude that content-based model is more suitable for tweets containing location related words, while other tweets can use the combination model to predict current locations.

Moreover, this work can be extended in user social relationships that takes into account user interaction information for building more sophisticated user social graphs. We would like to further reduce the prediction error to get a more granular predictive location of a given tweet as well.

References

1. Chandra, S., Khan, L., Muhaya, F.B.: Estimating Twitter user location using social interactions-a content based approach. In: IEEE Third International Conference on Privacy, Security, Risk and Trust, pp. 838–843 (2012)
2. Cheng, Z., Caverlee, J., Lee, K.: You are where you tweet: a content-based approach to geo-locating Twitter users, vol. 19, no. 4, pp. 759–768 (2010)
3. Crandall, D.J., Backstrom, L., Huttenlocher, D., Kleinberg, J.: Mapping the world's photos. In: International Conference on World Wide Web, pp. 761–770 (2009)
4. Ichiye, T., Karplus, M.: Collective motions in proteins: a covariance analysis of atomic fluctuations in molecular dynamics and normal mode simulations. Proteins Struct. Funct. Bioinf. **11**(3), 205 (1991)
5. Kearney, J.K., Hansen, S.: Stream editing for animation (1990)
6. Kim,Y.: Convolutional neural networks for sentence classification. CoRR, abs/1408.5882 (2014)
7. Kim, Y., Chiu, Y., Hanaki, K., Hegde, D., Petrov, S.: Temporal analysis of language through neural language models. CoRR, abs/1405.3515 (2014)
8. Kong, L., Liu, Z., Huang, Y.: SPOT: locating social media users based on social network context. VLDB Endow. **7**, 1681–1684 (2014)
9. Lee, K., Ganti, R.K., Srivatsa, M., Liu, L.: When Twitter meets foursquare: tweet location prediction using foursquare. In: International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, pp. 198–207 (2014)
10. Lin, S.K., Sheng-Zhi, L.I., Qiao, J.Z., Yang, D.: Markov location prediction based on user mobile behavior similarity clustering. J. Northeast. Univ. (2016)
11. Backstrom, L., Sun, E., Marlow, C.: Find me if you can: improving geographical prediction with social and spatial proximity. In: International Conference on World Wide Web, pp. 61–70 (2010)
12. Gasparini, M.: Markov chain Monte Carlo in practice. Technometrics **39**(3), 338 (1997)
13. Robinson, M.: The temporal development of collision cascades in the binary collision approximation. Nucl. Inst. Methods Phys. Res. B **48**(1–4), 408–413 (1990)
14. Sadilek, A., Kautz, H., Bigham, J. P.: Finding your friends and following them to where you are. In: ACM International Conference on Web Search and Data Mining, pp. 723–732 (2012)
15. Serdyukov, P., Murdock, V., Zwol, R.V.: Placing flickr photos on a map, pp. 484–491 (2009)
16. Xu, D., Yang, S.: Location prediction in social media based on contents and graphs. In: International Conference on Communication Systems Network Technologies, pp. 1177–1181 (2014)