The VLDB Journal (2019) 28:623–648
https://doi.org/10.1007/s00778-019-00553-0

REGULAR PAPER

Location prediction in large-scale social networks: an in-depth benchmarking study

Nur Al Hasan Haldar[1] · Jianxin Li[1] · Mark Reynolds[1] · Timos Sellis[2] · Jeffrey Xu Yu[3]

Received: 25 September 2018 / Revised: 3 April 2019 / Accepted: 27 June 2019 / Published online: 9 July 2019

Abstract
Location details of social users are important in diverse applications ranging from news recommendation systems to disaster management. However, user location is not easy to obtain from social networks because many users do not bother to provide this information or decline to do so due to privacy concerns. Thus, it is useful to estimate user locations from implicit information in the network. For this purpose, many location prediction models have been proposed that exploit different network features. Unfortunately, these models have not been benchmarked on common datasets using standard metrics. We fill this gap and provide an in-depth empirical comparison of eight representative prediction models using five metrics on four real-world large-scale datasets, namely Twitter, Gowalla, Brightkite, and Foursquare. We formulate a generalized procedure-oriented location prediction framework which allows us to evaluate and compare the prediction models systematically and thoroughly under extensive experimental settings. Based on our results, we perform a detailed analysis of the merits and limitations of the models providing significant insights into the location prediction problem.

Keywords Location prediction · Experimental evaluation · Large social network

1 Introduction

User location information contributes to in-depth social network data analytics. Discovering physical locations of users from social media helps us to bridge the online and offline worlds. This also supports many real-life applications like emergency reporting [2,46], disaster management [30,53], location-based recommendation [24], location-based advertisement [55], region-specific topic summarization [41], and disease outbreak monitoring [36]. However, location information is not always available in social networks because most users do not want to disclose locations in their profiles for reasons such as users' privacy, users' attitude, or even lack of interest to disclose [19]. For instance, only 16% of users in Twitter register location information in their profiles [28]. In another study, Cheng et al. [10] report that 21% of Twitter users from USA provide their location as city name and 5% provide their geo-coordinates. This calls for the development of location prediction methods that can exploit various implicit information inside the network to estimate users' locations.

Social networks provide elementary means for declaring spatial information through (1) self-reported context and (2) GPS-enabled geo-tagging of posts and check-ins [40]. There is significant interest in predicting user locations through public posts, metadata, and network information. Many researchers have focused on predictive algorithms to infer the locations of social users [5,9,19,32,44,57]. Some of these leverage the user-generated content (UGC) from the social stream [9,19,57] to predict users' locations using *location indicative words* (or "local" words) available within the users' GPS-tagged posts. The prediction performance of these models depends on the availability of local words in post contents. However, the location information in user-

✉ Jianxin Li
jianxin.li@uwa.edu.au

Nur Al Hasan Haldar
nur.haldar@research.uwa.edu.au

Mark Reynolds
mark.reynolds@uwa.edu.au

Timos Sellis
tsellis@swin.edu.au

Jeffrey Xu Yu
yu@se.cuhk.edu.hk

1 The University of Western Australia, Perth, Australia

2 Swinburne University of Technology, Melbourne, Australia

3 The Chinese University of Hong Kong, Hong Kong, China

generated posts is too limited. Ryoo et al. [44] report that only 0.4% of tweets (collected from the Korea region) have some GPS-tagged location information. In another study, Hetch et al. [19] report that only 0.77% of tweets among global users have some location information. Therefore, content-based location prediction approaches may not perform well due to the sparseness of location indicative words in users' posts. Hence, instead of using social contents, some prediction techniques [3,20,43] rely on the graph structure of a social network. These techniques exploit the network features while inferring users' locations using their social connections. For example, Backstrom et al. [3] assume that an unlabeled user is co-located with one of their friends in the network and a location is estimated by maximizing likelihood of their friends' locations. McGee et al. [33] integrate various social factors (e.g., number of followers) for the location prediction task.

Hybrid prediction models [27,28,38,40], on the other hand, exploit both the user-generated contents as well as the network information. If some neighbors (i.e., followers, friends) provide locations in their profile, or they mention some places in their posts, the hybrid prediction models can use such information to predict locations of unlabeled users. However, these models have the flexibility to use either one or both information types. Neural network-based geolocation prediction models are reported in [34,39]. Recently, some probabilistic frameworks [14,37] are proposed, which consider features learned through deep learning from social contexts. Apart from "user location" prediction, some studies focus on predicting other types of locations such as *post* (e.g., tweet) location, *mention* location, and *work* location. Meanwhile, the majority of the available works on predicting location of "posts" [8,9,29,42,45] rely on the social contents. The "mention" location prediction models [16,25,26,31] extract textual fragments in posts that observe some location names. There exists some work [8,11,58] which aim at predicting location types such as work place, or supermarket. Cho et al. [11] consider the temporal and social information to distinguish home locations and work places. Pang et al. [35] propose a feature learning framework based on deep learning, and it can predict user demographics and location category. Other notable work in predicting the next place visit of social users is available in [48,61]. However, in this study we are mainly focusing on the tasks of stable "user location" prediction using the network information. Additionally, we do not consider machine learning-based location prediction models in our benchmark study which are heavily dependent on the quality of training datasets.

With so many different models available for location prediction, it becomes important to compare their performance on standard benchmark datasets using similar metrics. The majority of the existing models are based on different internal configurations that best suit their targeted applications, and hence it is difficult to analyze, compare, and evaluate their suitability in a common base. It is also not clear how these models will perform in different scenarios such as different social network, different types of users, and location sparsity. Since the list of location prediction models is extensive, it is important to choose the representative approaches from each prediction category and develop a generalized benchmark to compare their relative performances.

In this paper, we compare models for *stable* "user location" predictions in social media. A *stable* location is defined as the long-term residential address (e.g., city level) or location where the majority of the activities are performed. Our main aim is to compare location prediction models that take the *network* features as input and predict users' *stable* locations. We also test whether the existing models can explain the observed data adaptation in Twitter microblog as well as in other location-based social network (LBSN) (i.e., check-in) datasets including Gowalla, Brightkite, and Foursquare. We assume that the majority of the activities of a user occur near to her stable location [4,11,24]. Meanwhile, locations may require different granularity given the specific application needs. For the sake of standard evaluation, we choose a uniform granularity level, i.e., city-level user locations. From here onward, we simply use "user location" instead of "*stable* user location" for brevity.

We divide existing models into four major categories (details to follow in Sect. 3.2) based on the prediction approach they use, and from each category, we choose representative models in a unified framework (see Fig. 1) to perform comparative analysis. Our aim is to gain insights into the general approaches (of the four categories) as well as the specific algorithms selected for comparison w.r.t. multiple aspects. Specially, we perform experiments on four social media datasets with different levels of location sparsity and compare the performances of the models with various user-centric and model-specific configurations. These evaluations give us novel insights into the relative merits of the specific location prediction models.

1.1 Challenges

The process of benchmarking location prediction models poses three major challenges:

- Due to the large diversity in existing models, it is difficult to abstract a unified benchmarking framework. It is critical to understand and diagnose the existing models from a common viewpoint.
- For a fair comparison and in-depth analysis of different location prediction model types, it is essential to apply these models on the exact same datasets. This requires the software implementation of these models which are not publicly available.
  Re-implementing the representative models on a com-

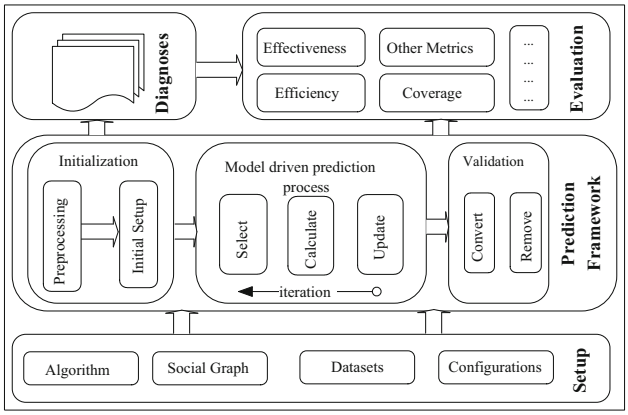


Fig. 1 Proposed benchmarking framework

mon coding platform and setting their parameters is a daunting task.
– Previous researchers have tested their approaches using a small number of metrics with limited scopes leading to the possibility of incomplete views of the model's performance. It is important to identify a suite of metrics that can evaluate multiple aspects of the location prediction outcomes of all existing models. Defining such a suite of metrics is a challenging task.

1.2 Generalized procedural framework

We propose a generalized benchmarking framework for the location prediction problem. We implement eight different representative models and evaluate them on the prediction task: "given a social network and information about geography, infer the locations of the 'unlabeled' users." The "unlabeled users" is defined in Definition 1 in Sect. 3.1. Our benchmarking framework consists of four core components as shown in Fig. 1, such as: (1) **Setup** includes a set of location prediction models, real-world datasets, parameter configuration, and the social graph generated from the datasets; (2) **Prediction Framework** presents a generalized location prediction module, with deep cogitation of the common workflow in the location prediction framework (see Sect. 4 for the framework details, and Sect. 5 for the mapping procedure); (3) **Diagnoses** discusses the key factors that affect the prediction performance of these models; (4) **Evaluation** provides a comprehensive evaluation module to verify and compare the models using both the dataset settings and the model-wise parameters. The structural components of our proposed framework are inspired from the benchmarking framework designed for community detection [54]. However, the internal functions of the components such as model-driven procedure, evaluation strategies, initial setup configurations, and the diagnoses approaches of our framework are very different. Moreover, in our study, we have implemented all the selected models in a common code base in a similar software environment allowing for a fair comparative analysis.

1.3 Contributions

We have conducted a comprehensive benchmarking study that performs in-depth analyses and comparisons of the different location prediction models. Specifically, we make the following major contributions:

– We review existing location prediction techniques and re-implemented eight representative models in a common code base.
– We perform an in-depth evaluation of the models using four real-world large-scale social media datasets with five different data settings on user location sparsity.
– We evaluate eight representative prediction models using five evaluation metrics under different user-centric parameters and data settings to demonstrate their strengths and limitations in a transparent comparison framework.
– We have adapted the network-based location prediction techniques for predicting user location in check-in datasets.
– We provide significant insights into the location prediction problem within large-scale social media and draw some interesting take-away conclusions about the compared models.

The remainder of this paper is organized as follows. In Sect. 2, we compare our work to existing survey papers on the location prediction problem. We generalize the problem of location prediction and sketch out the existing works in Sect. 3. Next, we propose a universal framework on location prediction of social users in Sect. 4. In Sect. 5, we map each individual model to the framework without any loss in their accuracy. We conduct extensive experiments to compare these models under similar configurations using various metrics and different data settings in Sect. 6. Finally, we conclude our study in Sect. 8 by giving some interesting insights into the existing location prediction models in Sect. 7.

2 Related work

There are two survey papers in the existing literature that compare the available models for location prediction in social media. Ajao et al. [1] studied the basic concepts in location inference techniques on Twitter social network and reported the accuracy of ten existing models. However, their comparisons are limited to the results presented in those ten works.

From their survey [1], it is not possible to derive a fair comparison of the models because the evaluations in the original papers were not performed on the same datasets and standard configurations. Another survey on location prediction on Twitter is reported recently by Zheng et al. [59]. This survey focused on comparing the models on three types of location (i.e., home location, tweet location, and mention location) prediction tasks. However, their comparisons are based on the summaries of the prediction models and lack comparative analysis. The survey [59] also does not provide the technical backgrounds of the prediction models.

Jurgens et al. [21] conducted a comparative review and analysis of nine network-based geolocation inference techniques using a bidirectional Twitter mention network dataset. They investigated the performance of the models on the task of predicting "tweet location" of an arbitrary user's post. However, they did not investigate the models' effectiveness under different parameter settings. Moreover, they did not provide any insights into the models' designs. The effectiveness and efficiency of the existing location prediction models may vary due to model-centric parameter settings as well as dataset properties. A comprehensive comparison of the models requires testing the models under different data-centric parameters and on different types of social networks. Hence, the comparisons reported in [21] are insufficient as they use only one dataset under limited model-centric settings. Moreover, the prediction performance of network-based models is significantly affected by variations in location sparsity. However, the analysis of Jurgens et al. [21] does not consider variations in location sparsity at all. More precisely, they consider the data setting with a majority of the users (i.e., 80%) with location annotated to predict the locations of the remaining 20% users only. This setting is far from real-world scenarios.

The prediction of "post" location (e.g., Tweet location) and that of "user" location are two different tasks [6,9,42] and hence require different approaches and evaluation metrics. Another limitation of the analysis in [21] is the choice of evaluation metrics. They used area under curve (AUC), Median–Max, and user coverage (instead of post coverage) which are not designed for similar types of prediction tasks. For example, AUC is used to evaluate the predicted locations of the posts, whereas the Median–Max and user coverage measure the user-level performances. In this case, the highest error of a user's predicted posts' locations is identified and then the median of these errors across all users is reported as the Median–Max of the location prediction. There is a high chance of getting a misleading conclusion when the majority of the user's posts have lower error distance and few posts are predicted very far from the original post locations. In this case, the Median–Max distance errors of each user may give a higher value, but the performance of the corresponding models may yet generate a better accuracy. In such a case, while comparing different models, the Median–Max metric fails to produce coherent results leading to misleading conclusions.

Also, it is difficult to decide from [21], which models perform better on accuracy and prediction coverage. For example, if a model predicts locations of a few posts of each user, the user coverage of the model will be high, but it will fail to justify the post coverage and accuracy of the model. Hence, the metrics used by Jurgens et al. [21] are insufficient to produce conclusive comparisons. As an example, if we consider the results of *Backstrom* [3] and *SLP* [20] models as reported in [21], the AUC of these two models is similar, whereas Median–Max error is 30.2 km lower in *Backstrom*, and the user coverage is 43.4% higher in *SLP* model. Hence, it is hard to conclude which model has overall better performance among these two models. In addition, the analysis in [21] lacks the functionality-wise comparison of the models in a common frame.

In our study, we address the above-mentioned issues and conduct a systematic in-depth benchmarking study by comparing eight location prediction models on four real-world datasets with different essential settings. We present several comparisons using different location sparsity levels, geographic region-specific predictions, agreements between model pairs, and the impact of user-centric information in location prediction tasks. Our comparisons give significant insights into the models' performances under various user- and data-centric settings.

3 Preliminaries and background

3.1 Preliminary

Since the majority of the models were originally tested on the Twitter data, they used Twitter-related terminologies in their discussions. However, in this paper we use different types of social media datasets, and hence, generic terminologies, i.e., "message" or "post" instead of "Tweet," will be used in this paper.

Definition 1 (*Social Networks*) A *social network* is a mathematical structure consisting of a set of entities (i.e., social users and locations) and their relationships. We define it as $G(V, E, L, T)$ where,

- V is the set of social users. It includes the labeled (V^*) and the unlabeled users (V^N), i.e., $V = V^* \cup V^N$. The "*labeled*" users ($u_i^* \in V^*$) are location-annotated users who have disclosed their locations in profiles. In some check-in datasets (e.g., Gowalla, Brightkite) if no profile locations are available, we can choose a "single" representative location among the multiple check-ins (discussed in Sect. 6.2) of the users where a majority of

Table 1 Features and time complexity of the models

Models	f_1	f_2	f_3	f_4	f_5	f_6	Complexity
UDI [28]	✓	✓	✓	✓	✓		$O(m\|E\|)$
MLP [27]	✓	✓	✓	✓			$O(m\|E\|)$
Backstrom [3]	✓		✓				$O(\|V\|k^2)$
SLP [20]	✓		✓				$O(\|V\|k^2)$
TFIDF [23]			✓	✓	✓		$O(\|V\|\|L\|)$
Friendly [33]	✓		✓			✓	$O(\|V\|ck^2 + k\|V\|\log\|V\|)$
SPOT [22]	✓		✓			✓	$O(\|V\|k^2)$
LMM [56]	✓		✓			✓	$O(\|V\|k^2)$

the activities occur. These locations are used to annotate the "labeled" users in check-in datasets. The remaining users, i.e., $(V - V^*)$, are considered as unlabeled users (V^N).

- L is the set of locations available in social network which contains the users' profile location, check-in locations, and locations available in users' posts (e.g., Tweets).
- E is the set of directed edges $e\langle v_i, v_j\rangle$ from v_i to v_j, which consists of *"following relationships"* $E_F : \{V \times V\}$ and *"messaging relationships"* $E_T : \{V \times L\}$. Edge $e\langle v_i, v_j\rangle$ is written as $f\langle i, j\rangle \in E_F$ where user $u_i \in V$ follows another user $u_j \in V$, or as $t\langle i, j\rangle \in E_T$ where user $u_i \in V$ mentions a location $l_j \in L$ in her posts. If some datasets do not have any user posts, the *"messaging relationships"* edges will be absent in G. In Twitter terminology, the *"messaging relationships"* is similar to *"twitting relationships."*
- T represents the set of posts (or messages) posted by V in G. The messages can be user posts, replies, or even forwarded messages (e.g., re-tweets). If a dataset do not have any message contents, the corresponding tuple of graph $G(V, E, L, \phi)$ remains null.

Different geolocation models consider different types of inputs e.g., content, network, and contextual information. Following relationships (f_1) and user location (f_3) are the main input features of network-based location prediction models. The other features such as messaging relationship (f_2), message or post contents (f_4), mentioned location frequency (f_5), social tie and closeness (f_6) have been used in different prediction models.

Table 1 lists the features that are used by the prediction models. The last column presents the time complexity of each model, "m" being the number of iterations, "k" the average number of labeled neighbors, and "c" the number of partitions for the tree regressions used in [33]. The tree regressor divides the dataset into smaller partitions to sort out the best contacts for the location prediction.

Definition 2 (*Location Type*) In social media, there are three types of locations, i.e., location of "post," "mentioned" location, and "user" location. The "post" location is generally available in the geo-tags of a post (e.g., tweet). "Mentioned" location refers to the locations available in post contents, whereas "user" location is available in self-reported profile and other check-in activities. Such locations may be home location, work location, or favorite location. We consider "stable" user location (e.g., home location) at city level where the majority of user activities occur.

Definition 3 (*Location Prediction Problem*) Given a social network $G(V = V^* \cup V^N, E, L, T)$, location prediction problem is to label a set of users $\hat{V}^N \in V^N$ with the locations selected from set L using a specified prediction model M_x, such that the predicted location $\hat{l}_{u_i}$ of $u_i \in \hat{V}^N$ is close to the actual location l_{u_i}.

3.2 Prediction models and algorithms

We focus on the fundamental problem of location prediction that aims to identify the locations of unlabeled users as precisely as possible. In most studies, the stable user locations are predicted at city level, state level or sometimes at the country level. Existing models have used three main input types, namely contents, network, and context. The models are broadly categorized into either content- or network-based approaches. The hybrid models, on the other hand, use both the content and network information simultaneously. However, we categorize the existing models based on their key approaches rather than the input type. For each category as shown in Fig. 2, we re-implement their representative models.

Probabilistic approaches The models in this category examine the probability distribution of different characteristics in the social network. *Language Model* [5,6,9,19,23,32,57], a sub-category of probabilistic approaches, analyzes the text-based contents of labeled users and build a "language model" (LM) using location indicative words (e.g., local words) available in users' posts. The local words have strong correlation with a specific location. These models calculate the

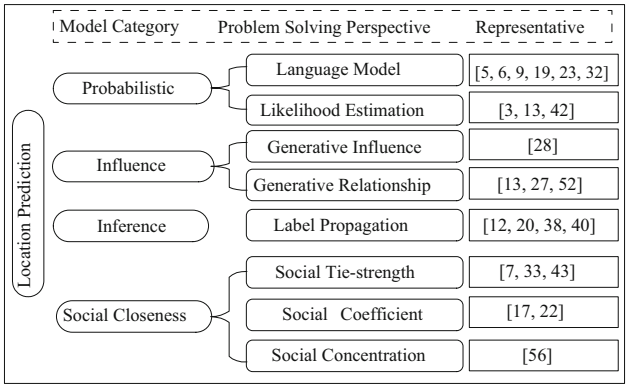


Fig. 2 Categorization of location prediction approaches

word distribution extracted from the labeled users' posts (e.g., tweets), and scores of locations are measured using probability distribution. Hecht et al. [19] and Yamaguchi et al. [57] calculate CALGARI and KL-divergence scores of words, respectively, to identify the local words in social contents. Mahmud et al. [32] apply heuristic rules to identify such words, whereas Cheng et al. [9] propose a model-driven approach based on the observed geographic distribution of the words. We choose *TFIDF* [23] model as the representative language model where a similar approach is considered to build a language model based on word distribution. A word is considered as a "local" word to a location if the corresponding *tf–idf* score is higher than a threshold parameter.

The *likelihood estimation* approach [3,13,42] estimates location of an unlabeled user by constructing a probabilistic model and measures the likelihood of users' friendship within a distance. Ren et al. [42] assume that if the majority of the user's friends live at a particular area, there is a higher chance for the user to co-locate with their friends. The *Backstrom* [3] model is the base model in this category, and we consider it as the representative model.

Influence-based approaches This type of model captures influence scope of nodes (i.e., user, location) by considering the relationship factors of social users and locations. The *generative influence*-based approach [28] integrates the social network and user-centric data in a generative framework to model the "*following*" and "*messaging*" relationships jointly. It predicts user's location by calculating the influence probability of how likely they follow other users or mention a location in their posts. *Generative relationship*-based approaches [13,27,52], in this category, measure the probability of various relationships using structural and spatial properties. Davis Jr. et al. [13] consider the structural relationship where the popular locations among the friends are considered as the location of a user. Li et al. [27] propose a generative approach that models the probability of generating "*following*" and "*messaging*" (i.e., twitting) relationship based on users' location. The most likely locations are assigned to an unlabeled user using the relationship probability of followers and mentioned locations.

We choose *UDI* [28] and *MLP* [27] as representative models of *generative influence* and *generative relationship* sub-categories, respectively. These two models use similar features (*following* and *messaging* relationships). However, the approaches of observing such relationships have significant differences as below:

(1) A *Generative influence*-based approach (e.g., *UDI*) calculates influence scope of nodes (i.e., user, venue) using Gaussian distribution and models them in a generative way. On the other hand, *Generative relationship*-based approaches directly observe *following* and *messaging* relationships and model them using power law and multinomial distributions, respectively (e.g., *MLP*).
(2) A *Generative influence*-based model (e.g., *UDI*) iteratively computes location of a user and updates the influence scope of her neighbors and mentioned places. This process continues until the likelihood converges. On the other hand, *Generative relationship*-based model (e.g., *MLP*) calculates the joint probability of the observed (e.g., *following* and *messaging*) relationships, and the maximum likelihood locations are assigned as the inferred locations.

Inference-based approaches These approaches are based on semi-supervised iterative algorithms which consider the spatial distribution of locations and infer a suitable location based on the social relationships. The factor graph model [37] uses location inference techniques to propagate the labeled locations by incorporating the deep features learned from the social context. The *Label Propagation* method [12,20,38,40] infers location by spatially propagating locations using the neighborhood information. Rahimi et al. [38,40] propose a hybrid approach which combines logistic regression with network-based label propagation to improve the location predictions. Both the models [38,40] use label propagation technique as the main prediction approach, but the initial estimation of user location is made by text-based geolocation techniques. For example, the model [40] considers logistic regression model prior for test users and the similar label propagation approach as *SLP* [20] is used to infer users' locations using updated median of neighbors' locations. Among the existing label propagation models [12,20,38,40], the *SLP* [20] is the basic one and has extended the label propagation concept of [60]. We choose *SLP* [20] as the representative model of this category.

Social closeness-based approaches The models in this category consider different network properties such as friendships, interactions, and social trust to estimate the locations of users. These models are based on the concept that social closeness of two users is better indicator of home proximity.

Algorithm 1: Generalized Location Prediction Procedure

Input: Social graph $G = (V^* \cup V^N, E, T, L)$, Model $(M_x, \mathcal{P})$
Output: $\hat{V}^N$

1 Initialize: $\mathcal{P} \leftarrow \Phi$;
2 **for** *each user* u_i *in* V^N **do**
3 $Seq \leftarrow (u_i, \text{FEATUREINIT}(u_i))$
4 PRECOMPUTE($G, \mathcal{P}, M_x$);
5 **while** ITERATION ! = $\text{ITERATION}_{\text{MAX}}$ **do**
6 **for** *each user* u_i *in Seq* **do**
7 SELECT(G, M_x, Seq);
8 $l_{u_i}^{tmp} \leftarrow$ CALCULATE($u_i, G, \mathcal{P}, M_x$);
9 $\hat{V}^N \leftarrow$ UPDATE($l_{u_i}^{t}, l_{u_i}^{tmp}, \hat{V}^N$);
10 ITERATION++;
11 VALIDATE(G, l_{u_i});
12 **return** $\hat{V}^N$

The *social tie-strength*-based approach [7,15,33,43] predicts the location of an unlabeled user considering the tie strength of the user and their labeled neighbors. Various social relationships like friendship, user mention, and node degree are used to measure the tie strength. We select *FriendlyLocation* [33] as the representative model of this sub-category. The *social coefficient*-based models [17,22] in this category measure the closeness of two users based on their quantitative neighbor information. Gu et al. [17] proposed the concept of social trust to measure the closeness in the social structure using the number of common friends. Kong et al. [22] propose *SPOT* model that calculates social closeness using cosine similarity between a user pair. We consider *SPOT* [22] as the representative of *social coefficient*-based model subcategory. The *social concentration*-based model (e.g., *LMM* [56]) infers locations from neighbors who have the higher spatial concentration with their social connections. The representative models of each category are discussed in Sect. 5.

4 The generalized procedure

We propose our benchmarking framework and use a generalized procedure to map the functionalities of eight location prediction models. To re-implement the existing models from a common view point, we formulate an adaptive procedure so that the models can be adjusted easily in different types of social networks. Our framework comprises three main phases, namely *Initialization*, *Model-driven location prediction process*, and *Validation* in the "Prediction Framework" as illustrated in Fig. 1. These phases are the key steps to characterize the generic procedure of the location prediction models. Algorithm 1 shows the generalized procedure of the proposed framework, and the details of the phases are discussed below.

4.1 Initialization phase

The *Initialization* phase initializes the primary configuration parameters of the models. We extract the model-specific features using FEATUREINIT() method (Line 3 of Algorithm 1) and create user prediction sequence Seq. We initialize the maximum number of iterations, $\text{ITERATION}_{\text{MAX}}$, as suggested by the original authors and set the parameter to 1 for the models which do not have multiple passes (e.g., *TFIDF*, *LMM*). Some models need to pre-calculate some parameters in PRECOMPUTE() (Line 4), such as power law distribution parameters in *MLP* [27].

4.2 Model-driven prediction

Initialization is followed by the prediction process. We abstract the three common key steps including SELECT, CALCULATE, and UPDATE at Lines 5–10. In SELECT() method, we pick the unlabeled users one by one, and the corresponding features of each user are loaded. The CALCULATE() method infers a location to the unlabeled user $u_i \in V^N$. Finally, the method UPDATE() assigns a new location by replacing any previously predicted geo-points. The three methods, i.e., SELECT, CALCULATE, and UPDATE, iterate until the termination criteria of the respective model are met.

4.3 Validation phase

In the final phase, we transform the geo-points into our predictable location type using the nearest city name. Some existing models assign geo-points to the predicted users, while the other return city names. This step ensures that the locations are validated consistently. Some model may return a "null" value corresponding to the users locations. Such invalid locations are removed in this step and the corresponding users remain unlabeled.

5 Within framework implementation

We recapitulate eight representative models with necessary adaptations to the proposed generic framework.

5.1 Probabilistic language model

Language model-based location prediction models characterize word distributions in users' texts (i.e., posts) and follow a *probabilistic* approach to infer users' locations. The models in this category construct a language model (LM) using "local words" available in labeled users' posts. Local words are tightly coupled with semantic locations. Though the language model-based prediction approaches use the social contents to predict users' locations, we consider such models to compare

with the other network-based models. Significant amount of research (e.g., [5,9,23]) in this category have been carried out on identifying "local words." For example, Bo et al. [5] proposed a model based on inverse location frequency (ILF) and inverse city frequency (ICF) to measure the probability of words in a location. A representative probabilistic model proposed by Cheng et al. [9] uses the distribution of user's home location l with the post (tweet) contents. Given a set of words w extracted from user u's posts $T(u)$, the probability of the user u being located at location l is calculated as

$$p(l|T(u)) = \sum_{w \in T(u)} p(l|w) * p(w). \quad (1)$$

Here, $p(w)$ is the probability of word w in the dataset and is calculated using the occurrence of the word w in the local word dataset. We consider *TFIDF* [23] as the representative model of this category. The mapping of this representative model in our framework is described as follows:

In PRECOMPUTE method, a language model (LM) is built using the location information of the labeled users and their corresponding post contents. The purpose of creating an LM is to compute the probability distribution of each location indicative words. The probability of a word w is calculated using term frequency and inverse document frequency (TFIDF) in a location l as

$$p(l|w) = \frac{c(w,l)}{\sum_{i=1}^{n} c(w_i,l)}, \quad c(w,l) = \sum_{s \in \text{post}(l)} tf(w,s). \quad (2)$$

$c(w,l)$ calculates the total number of occurrences (term frequency) of word w in the posts of "labeled" users who have location l. In SELECT, each unlabeled user is chosen one by one, and in CALCULATE, the probability of a user u located at location l is calculated using Eqs. 1 and 2. A location with the maximum likelihood probability is considered as the predicted location. However, the content-based probabilistic model may not perform well, as the availability of location indicative textual information is very rare in ordinary users' posts[19,44].

5.2 Generative influence-based model

The "Generative Influence Model" is based on modeling the influence scope of nodes in generative way. It follows a probabilistic approach to model the influences. The unified and discriminative influence model (*UDI* [28]) considers both the influence of neighbors and the locations mentioned in their posts. This approach models user's influence as a bivariate Gaussian distribution, and the variance of the distribution is interpreted as influence scope. Further, an iterative process is followed to update an unlabeled user's location using neighbor information. The newly predicted locations are subsequently used to estimate other users in the network.

The influence probability of a "node" n_i at a location l is modeled using a Gaussian distribution (refer Eq. 3). It considers a "node" as both a user and a location.

$$P(l|\theta_{n_i}) = \frac{1}{2\pi\sigma_{n_i}^2} e^{-\left[\frac{(X_{n_i}-X_l)^2}{2\sigma_{n_i}^2} + \frac{(Y_{n_i}-Y_l)^2}{2\sigma_{n_i}^2}\right]}. \quad (3)$$

Here, θ_{n_i} is the node n_i's influence model and σ_{n_i} is the influence scope of n_i. Two types of influence models are generated to measure the probabilities of generating *following* and *messaging* relationships using Eq. 3. A location that maximizes the joint probability of generating such relationship edges with the labeled neighbors and mentioned locations is inferred as the corresponding user's location. The generative influence model has two types of prediction methods. The *Local* prediction method observes the direct edges to infer the location of a user. The *Global* prediction method utilizes all relationships available in the entire graph, and it allows to iterate multiple times until it converges. Initially, this model assigns unlabeled users with random locations and then iteratively updates those locations using their neighbors' locations and mentioned locations.

Remark The *UDI* [28] model assigns a random location to the unlabeled users first. It follows multiple inner iterations to converge the assigned locations by updating influence scope of friends' and their mentioned locations. We notice that location prediction process in *UDI* using "random" location initialization takes time to converge. To optimize the process, we initialize the unlabeled users with the centroid of (at most) ten labeled neighbors' locations (rather using random locations). Such an approach has reduced 18% of the total inner iterations without affecting the overall accuracy. However, if a user has less than ten labeled neighbors, we consider all her labeled neighbors (to calculate centroid) to initialize the user. Meanwhile, we assign a random location to the unlabeled users if they do not have any "labeled" neighbor.

5.3 Generative relationship-based model

MLP [27] model, a representative of generative relationship-based model category, uses a supervised extension of latent Dirichlet allocation (LDA) to model the relationship between users and locations. The *MLP* model considers the effect of noisy relationships generated due to influences of famous personalities (e.g., "Lady Gaga") and popular venues ("Hollywood"). However, the approaches of considering the influences in a generative model are different from the influence-based model (discussed in Sect. 5.2).

MLP calculates the likelihood of a user following spatially close friends and mentioning nearby places. Locations

are observed from labeled users, and explicit correlations between locations and *following* relationships are measured. The following probability at distance "d" can be expressed as $P(d|\alpha, \beta) = \beta.d^{\alpha}$, and the values of the parameters α and β are learned using the labeled user information. Additionally, the location-based messaging model captures the messaging relationships using the mentioned location information. The *messaging* probabilities are modeled as multinomial distribution. However, this component (e.g., location-based messaging model) is not effective when there is no content-based information available in dataset. However, some relationships may not be generated based on the location distances. The *MLP* model captures noisy and location-based relationships using random generative models that measure the probability of randomly following a user or tweeting a venue.

This model combines the discrete (power law) and continuous (multinomial) distributions in a non-trivial manner, and Gibbs sampling-based algorithm is used to estimate the location assignments. After obtaining the location assignments for relationships of each user, their corresponding location distribution θ_i is measured with the maximal likelihood estimation, $p(l|\theta_i) = \frac{\varphi_{i,l}+\gamma_{i,l}}{\varphi_i+\sum_{l=1}^{L}\gamma_{i,l}}$, where φ is the user location assignments and γ is the prior distribution parameter of θ. The locations with largest probabilities in θ_i are estimated as the "stable" multiple locations of user u_i.

Remark *MLP* model can discover a user's multiple locations. We make a small addition to this approach to identify single "stable" location among the estimated multiple locations. We select the closest location to the centroid of the "multiple" predicted locations as the "stable" location of the user.

5.4 Probabilistic likelihood estimation-based model

The location prediction models (e.g., Backstrom et al. [3], Davis et al. [13], Ren et al. [42]) in this category study the interplay between geographic distance and social relationships. We choose *Backstrom* model ("Back" in short), one of the primitive models as the representative of this category. Location inference begins by building a probabilistic model representing the likelihood of observing a relationship between the users when a geographic distance is given. Based on the location distribution of labeled neighbors, a user is assigned a location which has the maximum likelihood. This model assumes that the location distribution of a typical user does not have many friends at long distances. Although the original paper [3] mentioning *Backstrom* model is conducted on Facebook, we adapt this model to other social media like Twitter and Foursquare.

In a large social network, the probability of friendship is roughly inversely proportional to the physical distance between the social friends [3]. Given a distance "d" between two users, the probability of having an edge (i.e., following relationship) between them is measured as $p(d) = a(b + d)^{-c}$. As mentioned in the original paper [3], the value of the constants $a = 0.0019, b = 0.196, c = 1.05$ is empirically determined using Facebook data. However, these values may vary in different datasets with different population distributions. For a given location l_u of user $u \in U^*$, if $L_v \in L(ngbr(u))$ are the locations of the labeled friends of u the edge probability for each neighbor location is computed as $p(|l_u - l_v|) = a(b + |l_u - l_v|)^{-c}$ s.t. $l_v \in L_v$. A location l_u co-located with one of u's friends is considered as the location of user u if the value of $\gamma(l_u)$ (refer Eq. 4) returns maximum value than considering the other neighbors' locations.

$$\gamma(l_u) = \prod_{e(u,v_j)\in E_F} \frac{p(|l_u - l_v|)}{1 - p(|l_u - l_v|)}, \qquad l_u \neq l_v. \tag{4}$$

Computing γ for each location is itself an expensive operation. As suggested in [3], the value of γ likelihood of each location can be pre-computed and we compute γ using PRECOMPUTE() method in our generalized framework.

Remark In the original paper [3], the authors mention that the model performs better for the users with 16 or more located friends. Hence, we exclude inferring some users who have "a few" (e.g., one or two) neighbors and it helps to improve the efficiency of this model.

5.5 Social tie-strength-based model

The tie-strength-based models [7,33,43] investigate social relationships that have stronger social tie and incorporate them in predicting users' locations. *FriendlyLocation* [33] (abbreviated as *Friendly*) is the representative model of this category. It leverages the relationship between tie strength of users pairs and their mutual distances. The basic assumption of this model is that users with strong ties are more likely to live near each other. Several social factors, e.g., following relationships, number of friends, conversations between social users, etc., are considered to measure the user proximity.

The *Friendly* model is semi-supervised model where the aforementioned social factors are used to train a decision tree classifier to distinguish between users' pairs who are likely to live nearby, and those who are distant. This model divides the predicted distance returned by the regression tree into "m" number of quantiles. Let $\{q_0, q_1, \ldots, q_m\}$ be the boundaries. Each predicted distance d_i^p of ith edge (s.t. $e_i\langle u, v\rangle \in E$) is assigned with a quantile number:

$$qntl(d_i^p) = \max_{j\in\{0,\ldots,m\}} \{j : d_i^p < q_j\}.$$

The number of socially connected edges ($actEdges$) in each quantile with distance "d" is measured as

$$actEdges(k,d) = |f_i\langle u,v\rangle \in E_F : d = d_i^a \wedge k = qntl(d_i^p)|.$$

Similarly, possible number of edges ($stgrEdges$) that could have existed at a distance d is calculated as

$$stgrEdges(d) = |e\langle u,v\rangle : u \in V \wedge v \in V \wedge d = dist(l_u, l_v)|.$$

Finally, the probability of a neighbor in a quantile j lives within d distance is measured as

$$p^*(k,d) = \frac{actEdge(k,d)}{stgrEdges(d)}.$$

Using training data, $p^*(k,d)$ function can be fit into curve for each quantile:

$$p^*(k,d) = a_k(b_k + d)^{-c_k}.$$

Now, the likelihood of a location $l \in L(ngbr(u))$ is maximized and the best location is inferred to the user:

$$F(l,L) = \prod_{l(ngbr_i(u)), d_i^p \in D^p} \frac{p^*(qntl(d_i^p), |l, l(ngbr_i(u)|)}{(1 - p(|l, l(ngbr_i(u)|))}.$$

5.6 Social coefficient-based model

Social coefficient-based model is based on the hypothesis that social distance can identify the closest friends in location estimation. In this model category, Kong et al. [22] propose *SPOT* model that calculates the energy of a user $u_i \in U^N$ locating at location l and having the social closeness score s_{ij} with neighbors. The "social closeness" is calculated using the cosine similarity between a pair of users u_i and u_j:

$$s_{ij} = |ngbr(u_i) \cap ngbr(u_j)| / \sqrt{|ngbr(u_i)||ngbr(u_j)|}.$$

Given the information of the labeled users in network, the probability $p(d_{ij}, s_{ij})$ of users $u_i \in V^*$ and $u_j \in V^*$ located at distance (d_{ij}) with their social closeness s_{ij} is measured. The maximum likelihood of a neighbor location w.r.t. social closeness score is predicted as the user location.

SPOT [22] model improves the location estimation errors due to highly uneven neighbor distribution and location sparsity problem. In these scenarios, to enhance the performance of "social closeness"-based models, the energy and local social coefficient-based approach is introduced to measure total energy of a user u_i locating at a location l_i:

$$Q(u_i, l_i) = -\sum_{j=1}^{|ngbr(u_i)|} s_{ij}.g(u_i, u_j).$$

Here, $g(u_i, u_j) = -e^{-|l_i,l_j|/d(s_{ij})}$, $u_j \in ngbr(u_i) \wedge U^*$ and $d(s_{ij})$ is the average distance of user u_i and neighbors u_j when the social similarity score is s_{ij}.

Local social coefficient of each user is calculated as

$$C(u_i) = \frac{3Q_\triangle}{3Q_\triangle + Q_\wedge}.$$

Here, $Q_\triangle$ is the number of closed triplet and $Q_\wedge$ is the number of open triplet connected by u_i with her neighbors. The energy value, $Q(u_i, l_i)$, and the social coefficient, $C(u_i)$, of each friend location are ranked to fit a logistic response function. The location with the highest probability is predicted as the user location.

5.7 Label propagation-based model

The label propagation-based location prediction approaches (e.g., *SLP* [20]) predict a user's location by propagating the location labels among their neighbors. It follows a multi-pass iterative process. *SLP* [20] model, a representative of this category, assigns a location of a user with the geometric median of neighbors' locations. The inferred locations can be used further to predict the location of the adjacent users while making new inferences.

In *SLP*, a location among the neighbors is selected by analyzing the spatial arrangement of the neighbors' locations. The geometric median "m":

$$m = \arg \min_{l \in L(ngbr(u))} \sum_{l_v \in L(ngbr(u)} |l, l_v|$$

is estimated as the location of the user $u \in U^N$ in the first pass. In each iteration, the newly predicted user location is further used to infer unlabeled neighbors' locations. This process continues until it satisfies convergence criteria. In *SLP*, the concept of the iteratively propagating the newly predicted location generates a flatter population distribution, which contradicts the concept that the majority of users live in dense area.

This model iterates multiple times until the stopping criterion is satisfied. In each iteration, the estimated locations are further used to predict the neighbors. In this way, some users with no labeled friends at the beginning may be predicted after a certain number of iterations. However, two problems may arise: (a) The incorrect estimation of a friend may lead to decreased accuracy when such predicted location is further used to infer the user. (b) Extensive iterations required for convergence may shift the correctly predicted location

Table 2 Summary of the datasets used

Dataset name	# Users	# Edges	Average degree	Average neighbor distance (km)	Average node locality
Twitter	138,012	2,274,416	32.95	1402	0.82
Gowalla	107,092	456,830	8.53	1722	0.52
Brightkite	51,406	197,167	7.67	1819	0.69
Foursquare	2,127,093	8,640,352	8.12	2629	0.79

away, if a large number of incorrect neighbor locations are included in each iteration.

5.8 Social concentration-based model

The social concentration-based model assumes that a higher proportion of a user's friends live in a dense location region. The representative *Landmark Mixture Model* (LMM) [56] model first identifies the set of *landmark* users who resides in a close proximity to others. After that, a location with the maximum likelihood among the landmark users' locations is inferred. Each landmark user is connected with a number of labeled neighbors, and the centroid of the neighbors is used to calculate dominance distribution using Gaussian mixture model (GMM). The users with lower variance in dominance distribution and having sufficient neighbors (e.g., landmark user) are used to infer the unlabeled neighbors. The maximum likelihood location of landmark users (among labeled neighbors) is selected as the predicted location.

The landmark users must have a large number of immediate neighbors, and the probability density of neighbor location distribution at the mode point should be high. However, the selection process of landmark users in *LMM* [56] is trivial. It lacks theoretical proof in identifying such users, and the optimal parameter ranges are not mentioned. Moreover, this model may fail to predict a suitable location if the neighbors are distributed sparsely.

6 Benchmarking evaluation

We perform extensive evaluation using our benchmark to measure the effectiveness, efficiency, memory consumption, prediction coverage, and combined performance of the eight representative models. The framework is implemented using Python in a Windows environment with Intel i7 CPU and 40 GB memory.

6.1 Datasets

We use four real-world datasets including Twitter microblog, Gowalla, Brightkite, and Foursquare LBSN. All these four datasets have graph structure where each user is considered a vertex (i.e., node), and the relationship between two users (if exists) is represented with an edge. The spatial distribution of users' network can be used to infer individuals' locations. We have adapted the functionalities of the selected location prediction models in LBSN (i.e., check-in) datasets. It is noted that in real world, many LBSN users might have social connections but no check-ins (or profile locations) [50]. Hence, it is useful to infer locations of such users from the network properties of LBSN. Table 2 gives a summary of the four datasets.

Twitter The Twitter (TW) dataset [28] was originally collected in May 2011, and the users are distributed in different cities of USA. We select 138,012 active users from this dataset who have both network information and tweet contents. This dataset is location-annotated (see [28] for details), and we transform the location name to geo-points using Google Geolocation API.[1]

Gowalla The Gowalla (GW) LBSN dataset is collected from SNAP repository (http://snap.stanford.edu) and contains 6,442,892 check-ins on 1,280,969 places worldwide over a period spanning from February 2009 to October 2010. In this dataset, 107,092 users have multiple check-in locations and form an explicit social network.

Brightkite Brightkite (BK) is another publicly available LBSN dataset in SNAP repository. The original data were collected over the period April 2008–October 2010. This dataset has multiple check-in locations, and a social graph is constructed using 50,686 users who have both check-ins and network information.

Foursquare The Foursquare (FS) LBSN dataset was collected using public API [24]. A total of 2.12 million users have self-reported location profile. We create a social graph, G, using the users who have self-reported locations and social connections.

6.2 Ground-truth information of datasets

Self-reported profile locations are used as the ground truth in Twitter [28] and Foursquare [24]. Since no profile locations are explicitly available in Brightkite and Gowalla datasets, we select the ground-truth location using approaches similar to Cho et al. [11]. We discretize the spherical earth surface

[1] https://developers.google.com/maps/documentation/.

Table 3 Summary of the Twitter dataset

Dataset ID	# Unlabeled users	# Labeled users	# Average labeled neighbors
Dataset TW-I	27,602	110,410	28.86
Dataset TW-II	55,205	82,807	23.20
Dataset TW-III	82,807	55,205	15.44
Dataset TW-IV	110,410	27,602	7.68
Dataset TW-V	124,211	13,801	5.72

Table 4 Summary of the Gowalla dataset

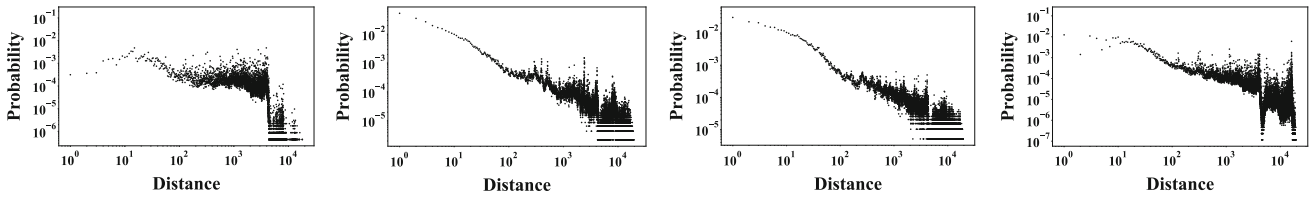
Dataset ID	# Unlabeled users	# Labeled users	# Average labeled neighbors
Dataset GW-I	21,418	85,674	11.08
Dataset GW-II	42,830	64,262	8.85
Dataset GW-III	64,262	42,830	7.09
Dataset GW-IV	85,674	21,418	5.43
Dataset GW-V	96,383	10,709	4.15

Table 5 Summary of the Brightkite dataset

Dataset ID	# Unlabeled users	# Labeled users	# Average labeled neighbors
Dataset BK-I	10,137	40,549	6.54
Dataset BK-II	20,274	30,412	4.93
Dataset BK-III	30,412	20,274	3.43
Dataset BK-IV	40,549	10,137	2.02
Dataset BK-V	45,617	5069	0.80

Table 6 Summary of the Foursquare dataset

Dataset ID	# Unlabeled users	# Labeled users	# Average labeled neighbors
Dataset FS-I	425,418	1,701,674	7.73
Dataset FS-II	850,837	1,276,2557	5.76
Dataset FS-III	1,276,255	850,837	3.83
Dataset FS-IV	1,701,674	425,418	2.08
Dataset FS-V	1,914,382	212,710	0.61



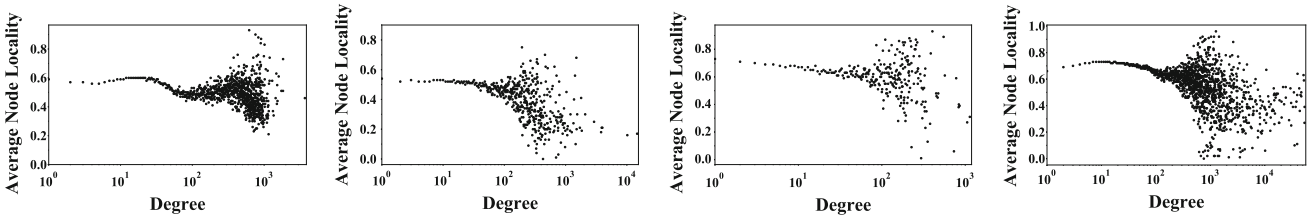
(a) Twitter **(b)** Gowalla **(c)** Brightkite **(d)** Foursquare

Fig. 3 Probabilities of following as function of distance

into 0.2 degree by 0.2 degree cells, which is approximately equal to 22 by 22 km w.r.t. equatorial region. For a given user, we find the cell with the "most number of check-ins" [49] and within this cell, we select the average check-in position as the ground truth for Gowalla and Brightkite datasets (Tables 3, 4, 5, and 6).

6.3 Friendship, distance, and check-in characteristics

In Fig. 3, we plot the following probability with distance between a pair of users u_i and u_j, s.t. $e(u_i, u_j) \in E$. The figure shows that (1) the *following* probability decreases when the distance between the user pairs increases, (2) in Twitter, the distribution is much flatter than the other three datasets. Gowalla and Brightkite have similar distributions. All the



(a) Twitter **(b)** Gowalla **(c)** Brightkite **(d)** Foursquare

Fig. 4 Average node locality as a function of node degree

patterns successfully capture the fact that a user is likely to follow others who live close. The following probabilities in each dataset can be fitted into a power law distribution curve if we ignore the larger distance pairs in each dataset. The user check-in activities also follow a heavy-tailed distribution [50], and the majority of check-in venues are near to users' stable locations [11].

6.4 Node locality

Node Locality is useful to quantify the geographic closeness of the neighbors to a certain user. For a user u_i with $|ngbr(u_i)|$ 1-hop neighbors, the *node locality* is calculated as [47]:

$$NL(u_i) = \frac{1}{|ngbr(u_i)|} \times \sum_{v_j \in ngbr(u_i)} e^{-\frac{d(u_i,v_j)}{\beta}},$$

where β is a scaling factor and it is calculated as follows:

$$\beta = \frac{1}{|E|} \times \sum_{u,v \in V, e(u,v) \in E} d(u, v).$$

Table 2 provides the average neighbor distance and the average node locality of each dataset. A dataset with higher average node locality should have large number of social connections within a close geographic region [47]. The users in Twitter dataset are distributed within USA, and this dataset has a higher average node locality score (0.82). Meanwhile, Gowalla has a lower node locality score (0.52) among the four datasets. This provides evidence that users in Gowalla are engaged with a geographically spread set of individuals rather than only with users at closer distances.

The correlation between node degree and node locality is useful to understand the socio-spatial properties of users. Figure 4 shows the average node locality as a function of node degree. It shows a fairly constant trend in each dataset when the average node degree is less than 100. These set of users may have similar properties with a similar proportion of neighbors living distant. In Gowalla, the average node locality drops significantly with the increase in node degree.

Table 7 Parameter settings in different models

Model	Parameters and value
UDI [28]	Outer iteration = 3, convergence error = 0.1
MLP [27]	Location-based following probability distribution parameters, $\alpha = -0.55$, $\beta = 0.0045$ (values of α, β depend on data type)
Back [3]	Friendship distance coefficient: $a = 0.0019$, $b = 0.196$, $c = -1.05$ (values of a, b, c depend on data type)
SLP [20]	No. of iterations = 4
TFIDF [23]	*tfidf* threshold = 0.1
Friendly [33]	LCR min dist = 40 km, quantile number = 10, max sample leaf = 1000
SPOT [22]	No. of iterations = 4

6.5 Parameter settings

In our evaluation, the essential parameters of the models are configured as recommended in the original papers. Table 7 shows the parameter settings of the models. One of the important parameters is the number of times a model is allowed to iterate. We set the default value as "two" for those models that follow multiple iterations but do not clearly mention in the original papers (e.g., *Backstrom* [3]). Meanwhile, the friendship distance coefficient [3] and location-based following probability parameters (α, β) [27] are sensitive to the types of data. *MLP* [27] model reports the value of $\alpha = -0.55$ and $\beta = 0.0045$ in Twitter; however, it is calculated as $\alpha = -0.42$ and $\beta = 0.0030$ in our selected Twitter social graph. Similarly, in our experiment, the values of (α, β) in Gowalla, Brightkite, and Foursquare are measured as (-1.52, 0.612), (-1.14, 0.20), and (-0.65, 0.016), respectively.

6.6 Metrics for evaluation

Table 8 lists the metrics used in the existing location prediction models. In this section, we discuss the suite of metrics used in our study to compare the models in a transparent comparison frame. We use Haversine [51] formula to measure the distance between two geo-points in kilometer unit.

Table 8 Metrics used in different models

Metrics	Model/work reference
AED@d, MeanED	UDI [28], Friendly [33], LMM [56], SLP [20], Cheng et al. [9]
AED@k%	UDI [28]
MedianEd	Friendly [33], LMM [56]
Acc, Acc@d	UDI [28], MLP [27], Friendly [33], Rout et al. [43], Cheng et al. [9]
Acc@K	Cheng et al. [9], [10], MLP [27], Bo et al. [18]
Precision, DP@K	OLIM [57], MLP [27], Davis Jr. et al. [13]
Recall, DR@K	OLIM [57], MLP [27], Davis Jr. et al. [13], Compton et al. [12], LMM [56]
CDF	Back [3], SLP [20]

Haversine strikes a good balance between correctness and computational efficiency that works over spherical earth surface.

Metric I Average Error Distance (AED) calculates the average distance between the actual location (l_{u_i}) and the predicted location ($\hat{l}_{u_i}$) of users:

$$AED(\hat{V}^N, M_x) = \frac{\sum_{u_i \in \hat{V}^N} Err(u_i, M_x)}{|\hat{V}^N|},$$

where $Err(u_i, M_x) = d(l_{u_i}, \hat{l}_{u_i})$ is the *Error Distance* (ED) between the actual and the predicted location of a user u_i in model M_x.

We evaluate the models using both "distance"- and "percentage"-based *AED*. The distance-based *AED@d* metric measures the average of the error distances of those users whose locations are predicted within "*d*" km:

$$AED@d = \frac{1}{|u_i|} \sum_{u_i \in \hat{V}^N} Err(u_i | u_i \in \hat{V}^N \wedge Err(u_i, M_x) \leq d).$$

The percentage-based *AED@k%* calculates the average of the error distance of top "*k%*" predicted users.

Metric II Precision (Prec) measures the quality of a prediction model. This metric calculates the percentage of the users predicted with error distance less than *"d"* kilometers. In a set of predicted users $\hat{V}^N$, the *precision* of a model M_x is calculated as:

$$Prec@d = \frac{|\{u_i | u_i \in \hat{V}^N \wedge Err(u_i, M_x) \leq d\}|}{|\hat{V}^N|}.$$

Metric III Accuracy (Acc@d) (or *Recall* [27,56]) measures the proportion of the correctly predicted users (with error distance less than *"d"*) among the test users V^N in a location prediction model M_x,

$$Acc@d = \frac{|\{u_i | u_i \in V^N \wedge Err(u_i, M_x) \leq d\}|}{|V^N|}.$$

Metric IV Prediction coverage measures the percentage of the unlabeled users (V^N) who have been assigned a location by a model regardless of the prediction accuracy. Let, a model M_x predicts $\hat{V}^N$ users among V^N unlabeled users in a dataset. The coverage of the model is calculated as ($\frac{|\hat{V}^N|}{|V^N|} \times 100$).

Metrics V Mutual Prediction Ratio (MPR) measures the percentage of similar predictions of two different models M_x and M_y within a given error distance d (e.g., 20 km):

$$MPR(M_x, M_y) = \frac{|\bigcap_{M_x, M_y} \{u_i | u_i \in \hat{V}^N \wedge Err(u_i) \leq d\}|}{|\bigcup_{M_x, M_y} \{u_i | u_i \in \hat{V}^N \wedge Err(u_i) \leq d\}|}.$$

This metric measures the mutual agreement of two models on location prediction.

6.7 Performance evaluation configuration

6.7.1 Evaluation on data types and location sparsity

To evaluate the models using various "data-centric" configurations, we perform experiments on the Twitter microblog and three LBSN datasets (e.g., Gowalla, Brightkite, and Foursquare). Initially, these datasets are location-annotated. We investigate the effect of the location sparseness and randomly choose 20%, 40%, 60%, 80%, and 90% users from each dataset to mask their locations (labeled as I, II, III, IV, V). Data setting I is the "less" sparse with 20% unlabeled and 80% labeled users, whereas data setting IV is the "highly" sparse containing 80% unlabeled and 20% labeled users. Data setting V has "extreme" sparsity with 90% unlabeled users. The location masked users are termed as "unlabeled" (V^N) and not used in the location estimation process. The statistics of the datasets with five sparsity levels are given in Tables 3, 4, 5, and 6.

6.7.2 Evaluation on different types of users

We design some experimental settings using node degree and node locality to analyze the effects of "user-centric" properties in the network.

Different node degree Different number of neighbors may affect the accuracy of the location prediction models. We divide the users into four groups w.r.t. node degree as: "5–10," "10–20," "20–30," and "> 30."

Different node locality Moreover, to explore the effect of neighbors' geographic distances in the prediction accuracy, we also group the users as "0.0–0.2," "0.2–0.4," "0.4–0.6," "0.6–0.8," "0.8–1.0" on node locality.

6.7.3 Region-specific model performance

Different social media captures different kinds of users distributed in various spatial regions. The network properties of social media in particular regions may have distinct set of characteristics. Hence, for some model, it may be much easier to predict a large number of users in a specific spatial region than the other models.

6.7.4 Scalability evaluation

Scalability of the prediction models is an important dimension for practical point of view in various applications (e.g., emergency reporting system). We compare the time cost and average memory consumption of the models in different data settings.

6.8 Effectiveness on different types of social media datasets with different parameter settings

We analyze the eight models on the metrics defined in Sect. 6.6. Note that *TFIDF* model can only be tested on Twitter data as the three LBSN datasets do not have content information.

6.8.1 AED@d

In Fig. 5, we report *AED@d* within 20 km, 50 km, 100 km, and 160 km of error distances using data settings I (less sparse), IV (highly sparse), and V (extreme sparse). The models in the Twitter dataset have lower *AED* value than three LBSN data. For example, in Twitter with data sparsity I and IV, the *UDI* model has 2.4 km and 2.7 km *AED@20*, respectively. However, it is observed 5 km higher in the LBSN datasets with similar data settings. The *AED* in extreme sparsity (i.e., level V) level is little higher than sparsity level IV. We found *AED@d* is always higher in *TFIDF* model. Below, we discuss models' relative performance using *AED@d*.

SLP model The *SLP* model has the lowest *AED@d* values in three LBSN datasets with "less" sparse data setting (e.g., 20% unlabeled), and it generates *AED@160* as 19.3 km, 22.6 km, and 22.6 km in Gowalla, Brightkite, and Foursquare, respectively. However, the *AED@160* increases by 3.0, 3.6, and 2.6 km in these three datasets, respectively, when data sparsity level changes from I to IV (e.g., from "less" to "highly" sparse). In Twitter, the value of *AED@160* increases by 2.4 km only.

Backstrom, SPOT, and friendly models The relative *AED@d* of *Backstrom*, *SPOT*, and *Friendly* remains similar in each data types. However, in Foursquare with high data sparsity (e.g., FS-IV), the *AED@160* of *Backstrom* has higher value than the other two models.

LMM model The *LMM* model has higher *AED@d* in majority of the data settings in LBSN. A significant increase in *AED@160* is noticed while location sparsity changes in Twitter dataset. However, the *AED* of *LMM* does not change much in LBSN. For example, in Twitter, the *AED@160* is 10 km and in Brightkite 1.1 km higher when sparsity level changes from I to IV.

UDI and MLP model In comparison with Twitter, the *UDI* and *MLP* models have always higher *AED* in LBSN datasets. This is because the content information available in Twitter helps to predict more precise locations than three LBSN datasets.

6.8.2 AED@k%

A distance-based $AED@d$ can be easily affected by the outliers in the results. In addition, different amount of predictions within a certain error distance may not make a transparent comparison of the models. Hence, we use percentile-based *AED*s that calculate average error distance using top $k\%$ (i.e., 60%, 80%, and 100%) of the predicted users ranked by their error distances. Figure 6 shows the $AED@k\%$ of the models.

UDI and MLP models The *UDI* model has the lowest $AED@k$ in Gowalla and Brightkite. In Twitter, *MLP* and *UDI* have similar $AED@k$ in both I and IV settings. However, we have noticed that *MLP* model has predicted more precise locations in FS-IV setting which generates a better in $AED@100\%$. Meanwhile, the number of predicted users is lower in *MLP*. We have discussed the effect of prediction coverage in Sect. 6.11.

Backstrom, friendly, and LMM models The relative $AED@k\%$ of *Backstrom* and *Friendly* models is similar in each dataset. The prediction approaches of these two models maximize the probability of locations based on the curve fit using the edge probability with distance. However, the edge probabilities are different in these two models, but in similar setting these two models can predict users with similar error distances. The *LMM* model has relatively the highest $AED@k$ in each of the data settings.

6.8.3 Precision

Figures 7, 8, 9, and 10 show the precision of each model in four datasets with five sparsity levels. We only discuss the significant observations of the models' performance w.r.t. precision at 160 km (i.e., *Prec@160*).

UDI model The *UDI* model has higher precision in majority of the datasets in different sparsity levels. In three LBSN datasets, the precision in *UDI* does not change much when location sparsity changes. For example, in Gowalla, the precision of *UDI* model drops only 3.12%, while no significant changes are observed in Brightkite and Foursquare datasets. However, in Twitter, the precision changes notably w.r.t. loca-

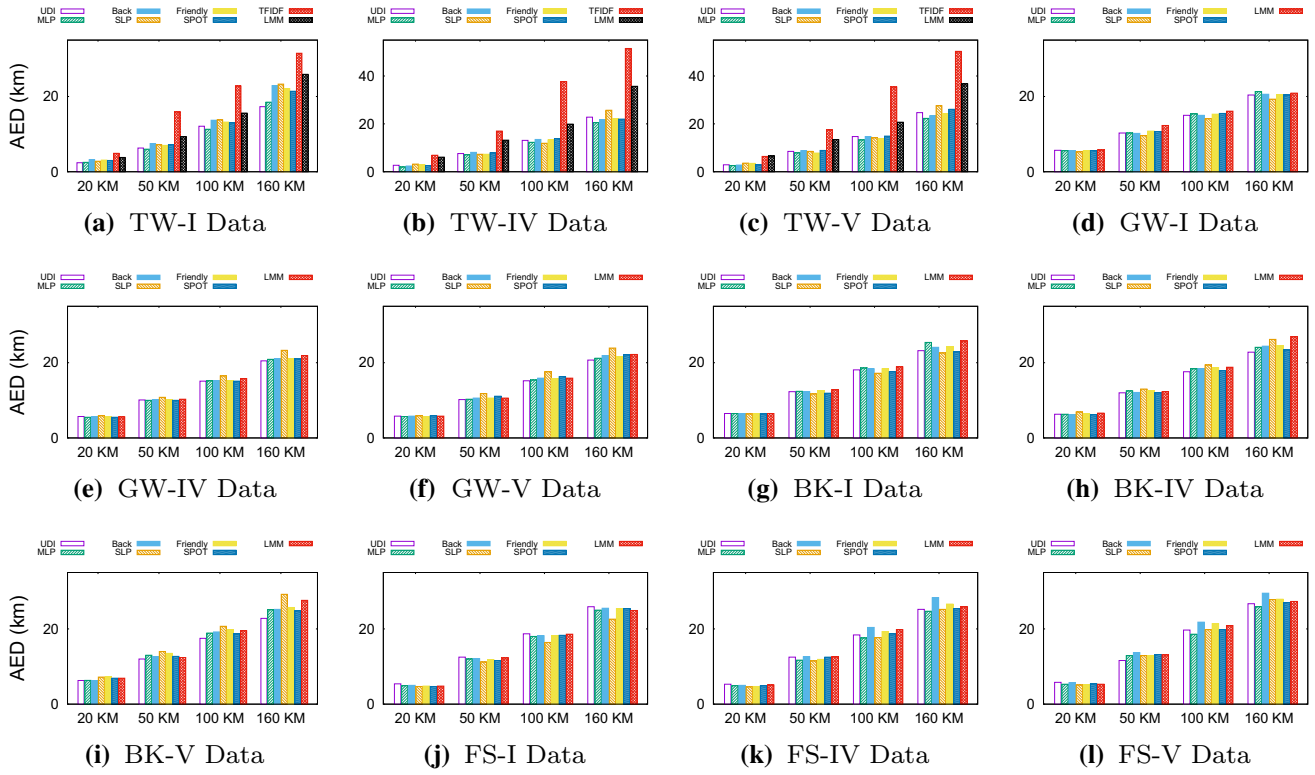


Fig. 5 *AED@d* using different data settings

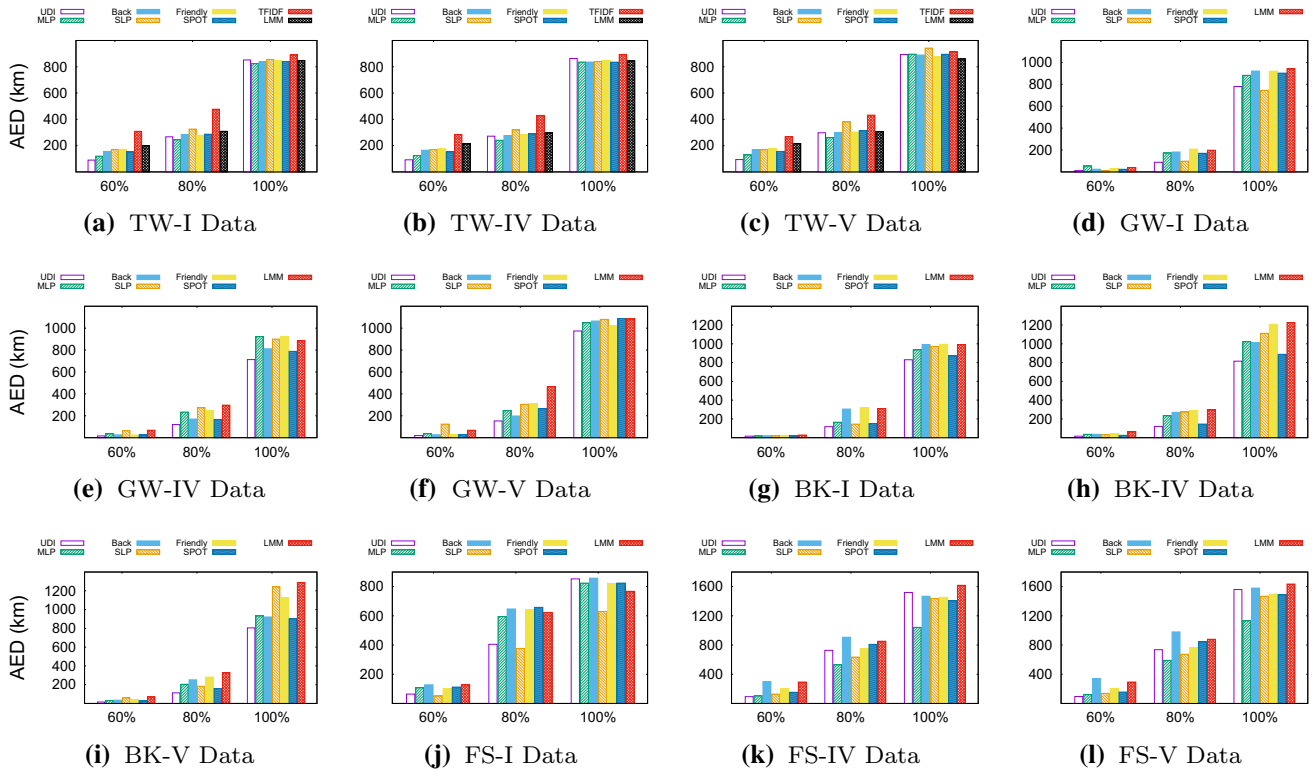


Fig. 6 *AED@k%* using different data settings

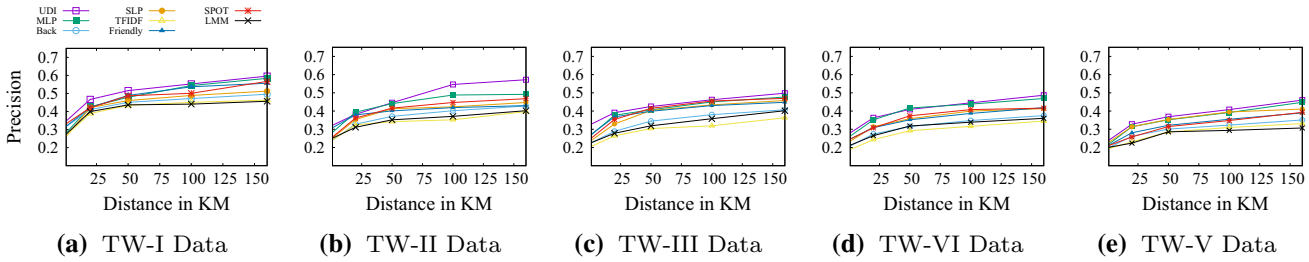


Fig. 7 Precision of the location prediction models using Twitter dataset

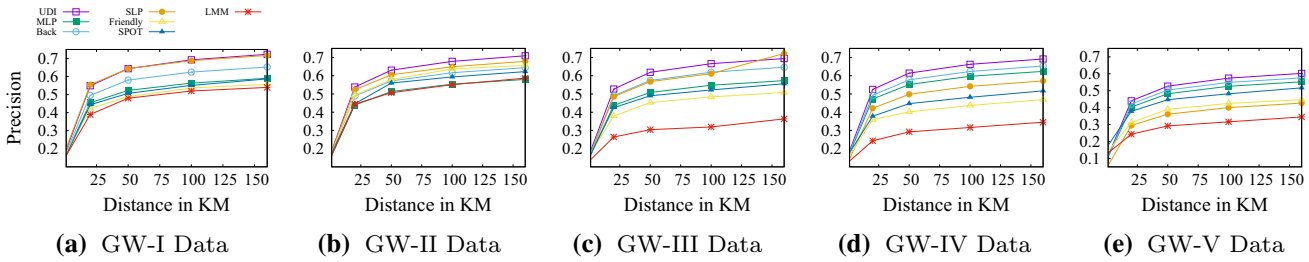


Fig. 8 Precision of the location prediction models using Gowalla dataset

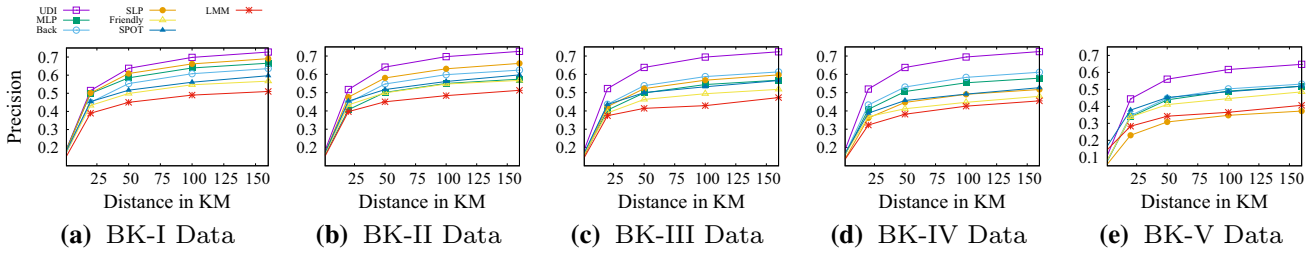


Fig. 9 Precision of the location prediction models using Brightkite dataset

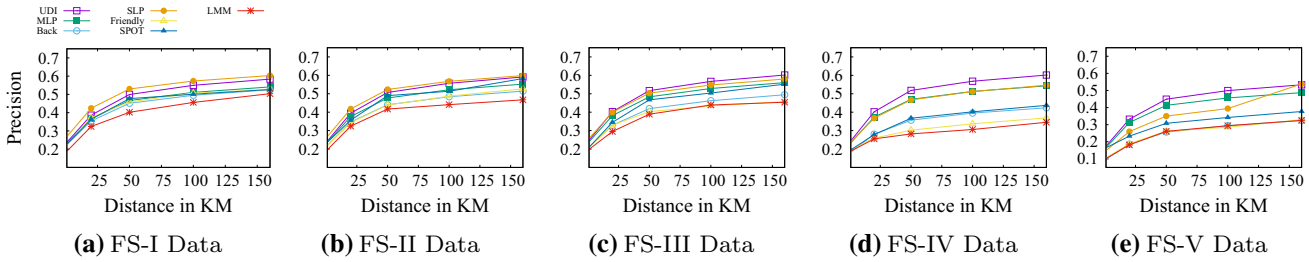


Fig. 10 Precision of the location prediction models using Foursquare dataset

tion sparsity. For example, it drops 11% when sparsity level increases from I from IV. Similarly, in extreme sparseness (e.g., TW-V), the precision drops 14% in Twitter.

SLP model In Gowalla and Brightkite datasets, the *SLP* model has the second best *Prec@160* in sparsity levels I and II. Similarly, in Foursquare dataset, *SLP* has the highest precision on FS-I (68.33%) and FS-II (67.87%) data settings. However, we notice a significant decrease in precision of *SLP* when the location sparsity increases to higher and extreme higher levels.

Backstrom, SPOT, friendly models The relative precision of these three models in Gowalla and Brightkite datasets are similar, where *SPOT* has higher *Prec@160* than *Friendly* and lower than *Backstrom*. However, in Twitter dataset, *SPOT* has better *Prec@160* than other two models. This may be because the local social coefficient factor is effective in Twitter dataset and the average neighbor distance is lower in Twitter.

MLP and SLP models The *MLP* model has better precision than *SLP* in Twitter dataset. However, *Prec@160* in *SLP* is higher than *MLP* model in the first three data settings (i.e., I, II, and III) in LBSN datasets. Meanwhile, *MLP* has a better

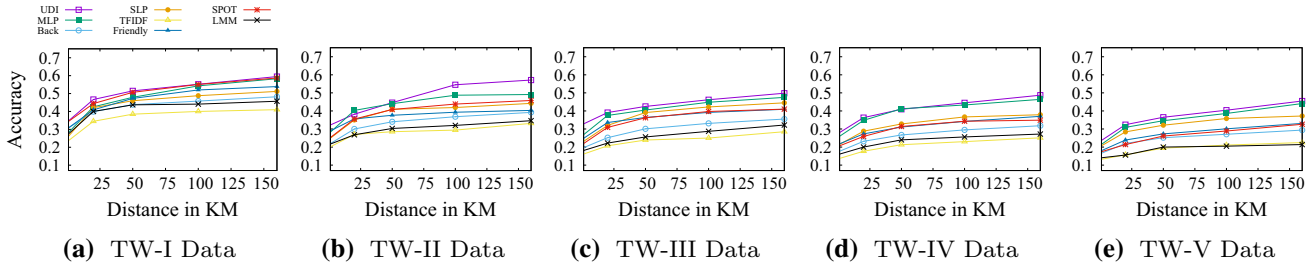


Fig. 11 Accuracy of the location prediction models using Twitter dataset

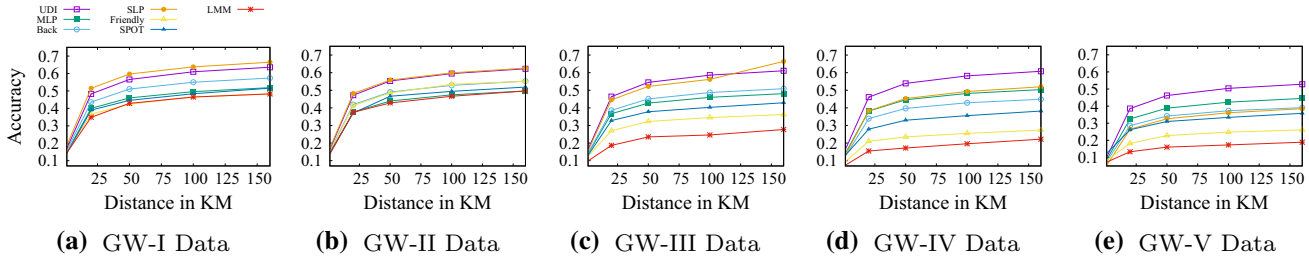


Fig. 12 Accuracy of the location prediction models using Gowalla dataset

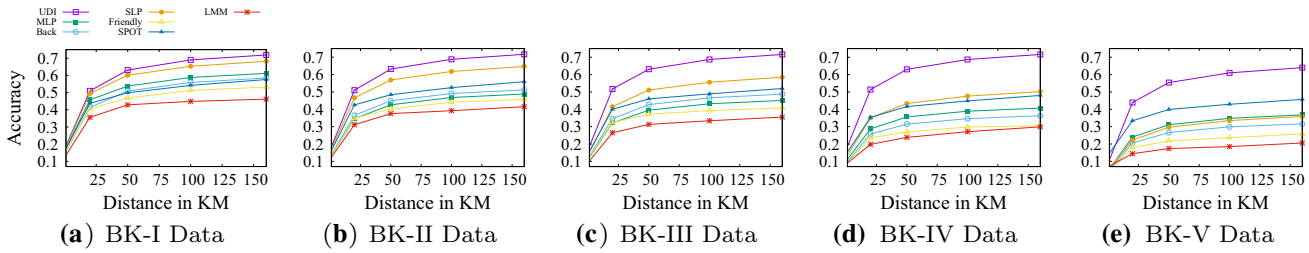


Fig. 13 Accuracy of the location prediction models using Brightkite dataset

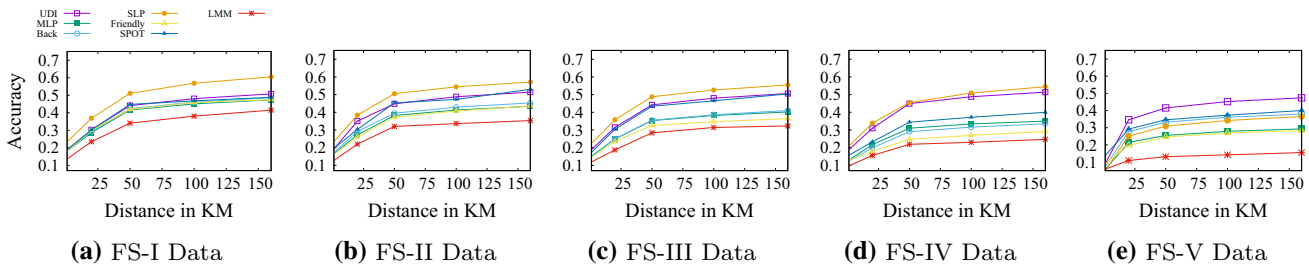


Fig. 14 Accuracy of the location prediction models using Foursquare dataset

precision in sparser data settings. For example, in Gowalla GW-I setting, the value of *Prec@160* is 12% higher in *SLP*; however, *MLP* generates 5% and 12% better *Prec@160* in GW-IV and GW-V, respectively.

LMM model The precision in *LMM* decreases drastically in three LBSN datasets with the increase in data sparsity. For example, *Prec@160* of *LMM* changes in Gowalla from 54% to 35% with sparsity level changes from I to IV. This is because *LMM* does not iterate multiple times to precise the predicted users' locations.

6.8.4 Accuracy

Figures 11, 12, 13, and 14 show the accuracy of the models in the four datasets under five location sparsity settings and different error distances ranging from 20 to 160 km. We discuss accuracy of the models with 160 km (i.e., *Acc@160*) error distance in the following discussions.

UDI model UDI has the highest accuracy in Twitter dataset, but it is lower than *SLP* model in less sparse Gowalla and Foursquare data. However, with "high" and "extreme" sparsity level in these two datasets, the *UDI* model outperforms

the second best model by 2–12% in *Acc@160*. This is due to multiple inner iterations performed by *UDI* model, where a better location is assigned until it converges. Hence, in sparse datasets, the *UDI* model can predict more users with precise locations. In Brightkite, the accuracy of *UDI* is the highest in each of the five different data settings.

SLP model *SLP* has better accuracy in less sparse data settings, and it decreases heavily when location sparsity increases. The performance of *SLP* model is always better when a large number of users have sufficient neighbor information. It reports the highest *Acc@160* in Gowalla and Foursquare with data setting I and outperforms the second best by 7% and 3%, respectively. However, in Twitter TW-I the *Acc@160* is 51% only.

MLP model The accuracy of *MLP* model decreases smoothly when the number of unlabeled user increases. In Twitter dataset, the *Acc@160* drops 14% when sparsity level changes from I to V. However, in LBSN datasets the accuracy of *MLP* model is always lower than Twitter and it generates similar accuracy trend as *Backstrom* model.

Backstrom, SPOT, and friendly These three models generate different pattern in different types of social network. For example, *SPOT* and *Friendly* have higher *Acc@160* than *Backstrom* model in Twitter dataset. However, in LBSN datasets *Backstrom* outperforms the other two models. This is because in Twitter various factors related to social tie improve the prediction results in *SPOT* and *Friendly*, whereas the LBSN datasets lack such social factor parameters.

6.8.5 Mutual prediction ratio

The *Mutual Prediction Ratio* (MPR) between a "pair" of prediction models is shown in Tables 9, 10, 11, and 12. Note that we use data settings I and IV of Twitter microblog and Foursquare LBSN to evaluate similar predictions of model pairs within 20 km of error distance. The *SLP* model returns higher MPR when pair with *UDI* and *MLP* models in Twitter dataset. This is because these models consider user relationships and their neighbor distance as important factors in their prediction task, and similar set of users with higher node locality are predicted within a lower error distance. The *TFIDF* model produces lower MPR score with others models. This is because the prediction approach and features used in *TFIDF* model are totally different from the remaining network-based models. On the other hand, *Backstrom* model produces a higher MPR score with *Friendly* and *SPOT* models. This is because these models consider similar social factors such as friendship and social closeness with the neighbor distance. The MPR scores between model pairs decrease in highly sparse datasets. For example, in FS-I, the majority of the model pairs have MPR score larger than 0.50, whereas in FS-IV the majority of the MPR scores are below 0.40.

Table 9 Mutual prediction ratio in TW-IV data

	MLP	Back	SLP	TFIDF	Friendly	SPOT	LMM
UDI	0.39	0.32	0.51	0.22	0.37	0.33	0.24
MLP	–	0.35	0.53	0.29	0.39	0.35	0.26
Back	–	–	0.37	0.31	0.42	0.59	0.32
SLP	–	–	–	0.25	0.35	0.43	0.34
TFIDF	–	–	–	–	0.30	0.24	0.20
Friendly	–	–	–	–	–	0.36	0.45
SPOT	–	–	–	–	–	–	0.29

Table 10 Mutual prediction ratio in TW-IV data

	MLP	Back	SLP	TFIDF	Friendly	SPOT	LMM
UDI	0.25	0.12	0.25	0.26	0.15	0.35	0.23
MLP	–	0.18	0.63	0.20	0.19	0.27	0.17
Back	–	–	0.20	0.11	0.25	0.45	0.18
SLP	–	–	–	0.15	0.22	0.36	0.20
TFIDF	–	–	–	–	0.11	0.15	0.16
Friendly	–	–	–	–	–	0.24	0.21
SPOT	–	–	–	–	–	–	0.18

Table 11 Mutual prediction ratio in FS-I data

	MLP	Back	SLP	Friendly	SPOT	LMM
UDI	0.67	0.64	0.63	0.60	0.58	0.51
MLP	–	0.60	0.58	0.63	0.52	0.48
Back	–	–	0.62	0.68	0.65	0.55
SLP	–	–	–	0.60	0.56	0.52
Friendly	–	–	–	–	0.52	0.45
SPOT	–	–	–	–	–	0.53

Table 12 Mutual prediction ratio in FS-IV data

	MLP	Back	SLP	Friendly	SPOT	LMM
UDI	0.30	0.33	0.43	0.30	0.32	0.30
MLP	–	0.51	0.34	0.41	0.34	0.29
Back	–	–	0.37	0.44	0.35	0.25
SLP	–	–	–	0.34	0.36	0.31
Friendly	–	–	–	–	0.32	0.25
SPOT	–	–	–	–	–	0.23

6.9 Effectiveness on local vs. global inference

6.9.1 Local inference technique

Local Inference (Local prediction) technique uses one- or two-hop friendship information to infer users' locations. We compare the model performances using Twitter microblog (see Fig. 15) and Foursquare (see Fig. 16), the largest among

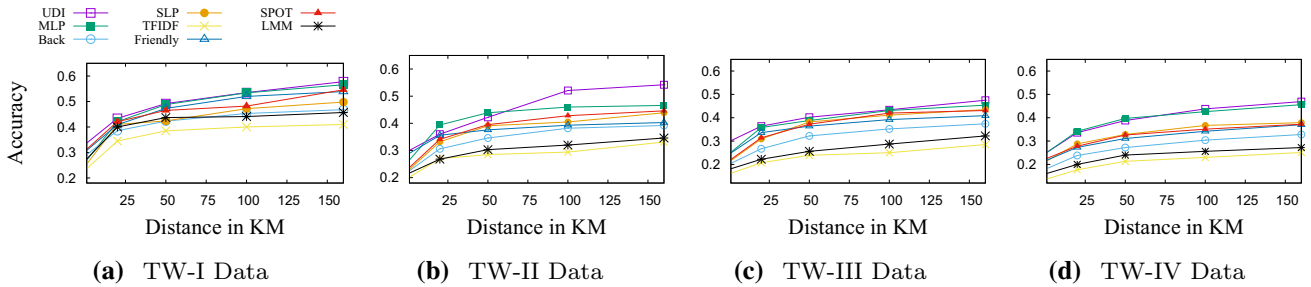


Fig. 15 Local prediction accuracy of the location prediction models using Twitter dataset

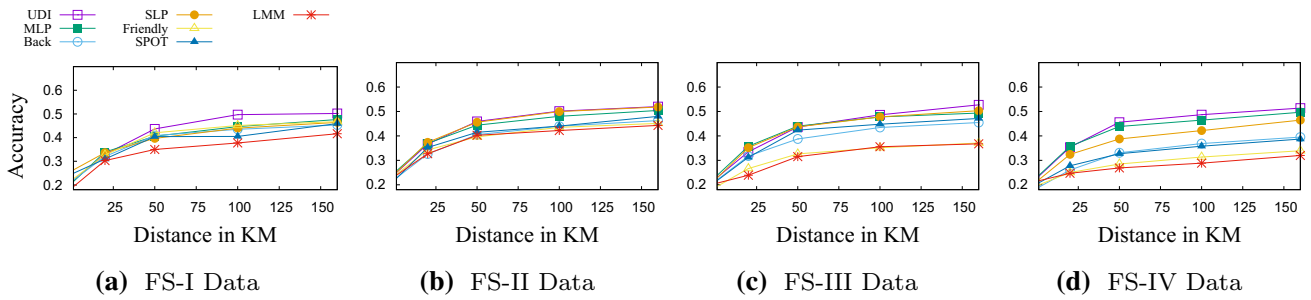


Fig. 16 Local prediction accuracy of the location prediction models using Foursquare dataset

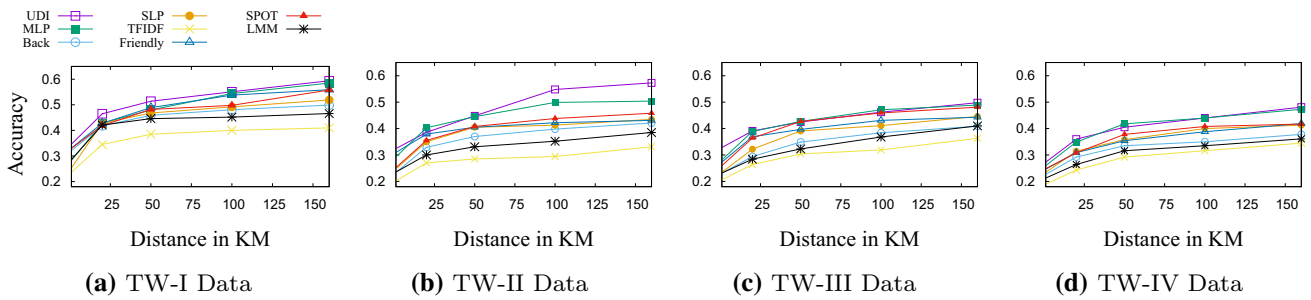


Fig. 17 Global prediction accuracy of the location prediction models using Twitter dataset

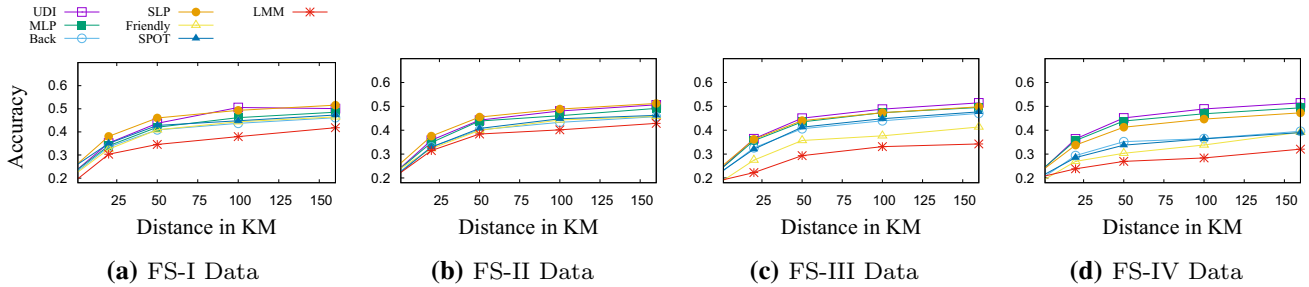
the three LBSNs datasets for the comparison of the model performance using first four data settings (e.g., I–IV).

Considering the one-hop neighbor information, the *UDI* and *MLP* models produce higher accuracy than the other models in Twitter and Foursquare datasets. The accuracy of *SPOT* has declined by 4% (in TW-I dataset) compared to its default configuration with four iterations. In Twitter, there is no major differences in local inference accuracy of *Friendly*, *LMM* models with their default configuration. The local prediction accuracy is stable in *TFIDF* model, as the number of iteration is ineffective to the performance of this model. In both datasets, the accuracy of *Backstrom*, *Friendly*, and *SPOT* declines faster with the increases in location sparsity.

6.9.2 Global inference technique

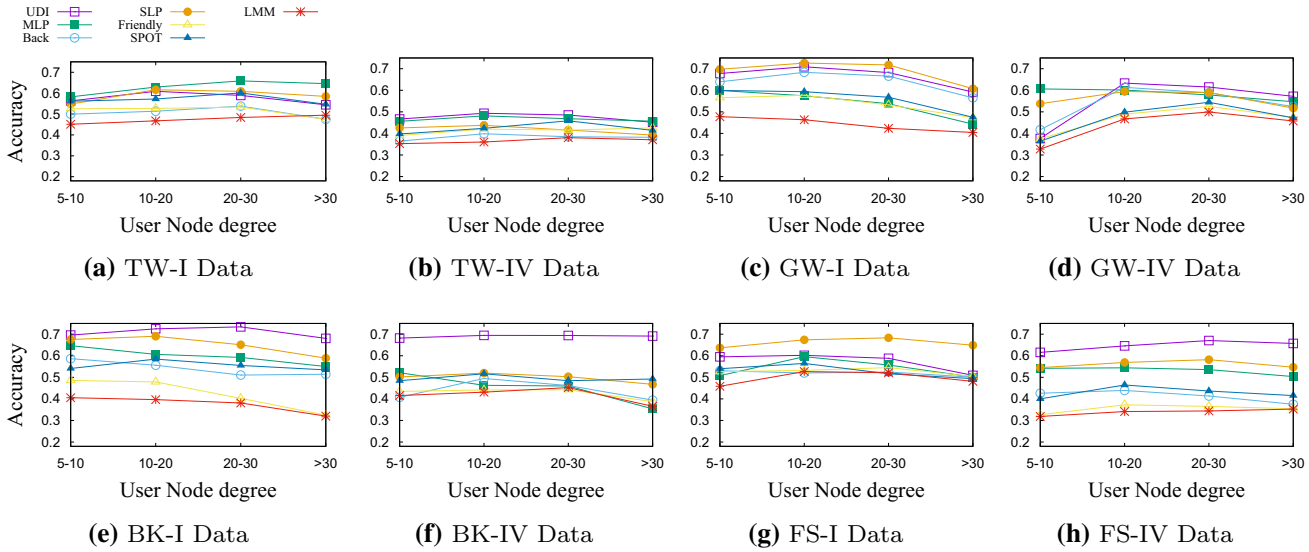
Global Inference (Global prediction) technique is used to overcome the location sparsity problem where a newly predicted location can be used further and updated iteratively to predict the locations of other users in the network. We set the number of iteration as 4 and show the accuracy of the models in Figs. 17 and 18 using data settings I–IV in Twitter and Foursquare data, respectively.

The *Backstrom*, *Friendly*, and *LMM* models have significant improvements in accuracy compared to the local inference. For example, in TW-IV data settings the accuracy *Acc@160* increases in these three models by 5%, 5%, and 9%, respectively. In Foursquare FS-IV, the accuracy of these three models improves between 4 and 11%. The relative performance of the *UDI* model is quite stable in both of the datasets w.r.t the default settings. This is because the difference in number of iterations in global inference and default setting is only one in *UDI*, and hence, no significant new predictions occur. A large number of iterations may not always improve the performance of the models. We have identified *SLP* as the most sensitive model to "number of iterations." This model performs best in Foursquare with four and in Twitter with three iterations.



(a) FS-I Data **(b)** FS-II Data **(c)** FS-III Data **(d)** FS-IV Data

Fig. 18 Global prediction accuracy of the location prediction models using Foursquare dataset



(a) TW-I Data **(b)** TW-IV Data **(c)** GW-I Data **(d)** GW-IV Data

(e) BK-I Data **(f)** BK-IV Data **(g)** FS-I Data **(h)** FS-IV Data

Fig. 19 Performance of models with different node degrees

6.10 Effectiveness on different types of users

6.10.1 Users with different node degrees

Figure 19 shows the performance of the location prediction models on users with different numbers of neighbors (i.e., node degree). Here, we are reporting the result using "less" sparse (i.e., data setting I with 80% labeled users) and "high" sparse (i.e., data setting IV with 20% labeled users) data. The location inference technique of *TFIDF* model is independent of the network information; hence, we exclude this model from the discussions. We make the following observations:

In Twitter, the relative accuracy of the models in different node degrees is quite similar. In TW-IV data setting, the average accuracy decreases linearly in each model when node degree increases beyond 20. Similar pattern occurs in majority of the models in the remaining three datasets. In Gowalla, the users with node degree between 10 and 20 have higher accuracy in *UDI*, *MLP*, and *SLP*. In Brightkite BK-IV, the accuracy of *UDI* is constant w.r.t. different node degree ranges. In Foursquare dataset with FS-I setting, the relative accuracy of users in different ranges of node degree is similar with Gowalla GW-I data settings. The majority of the models obtain higher accuracy for the users who have node degree between 10 and 30. Users with a very large node degree fail to infer better locations.

6.10.2 Users with different node locality

Figure 20 shows the proportion of the users predicted by each model within 160 km from the actual users' location. Each model has predicted a very small proportion of the users who have smaller node locality (i.e., less than 0.2), while a large amount of users with node locality more than 0.8 have been predicted precisely within 160 km. In Twitter, there is a steady increase in proportion of predicted users with the increase in node locality score. We notice that there is a sudden growth in the predicted user proportion in Brightkite and Foursquare when node locality scores of the users are more than 0.8. Among the four datasets, more than 80% of users in Foursquare with node locality scores 0.8–1.0 have been predicted precisely within 160 km of error distance in *UDI*, *MLP*, and *SLP* models. In general, large proportion of the users with higher node locality are predicted precisely by

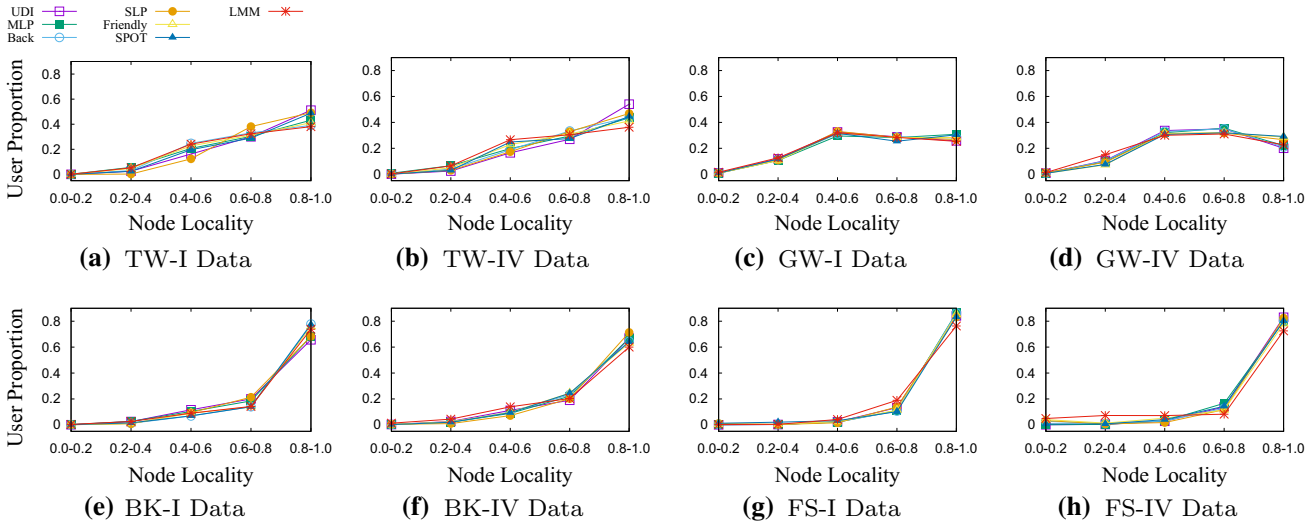


Fig. 20 Predicted users proportion (with error distance less than 160 km) with different node locality

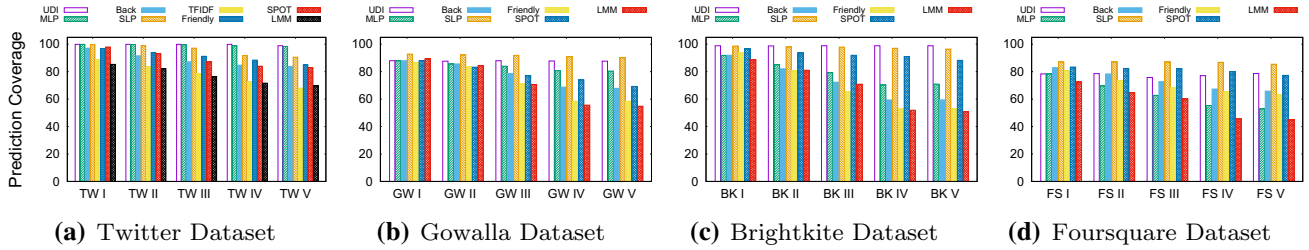


Fig. 21 Prediction coverage of models in different datasets with default configuration

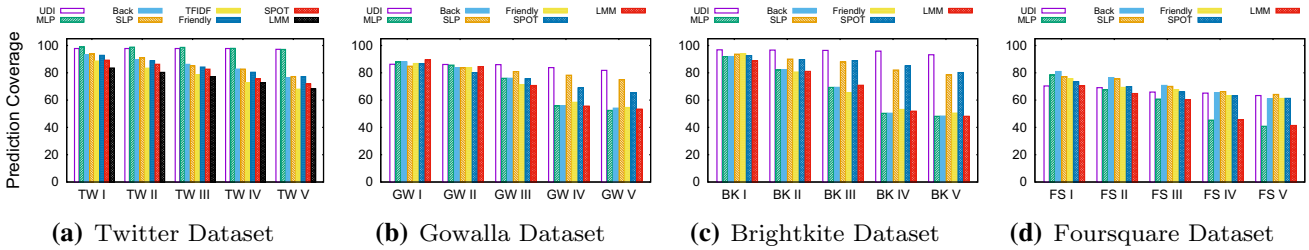
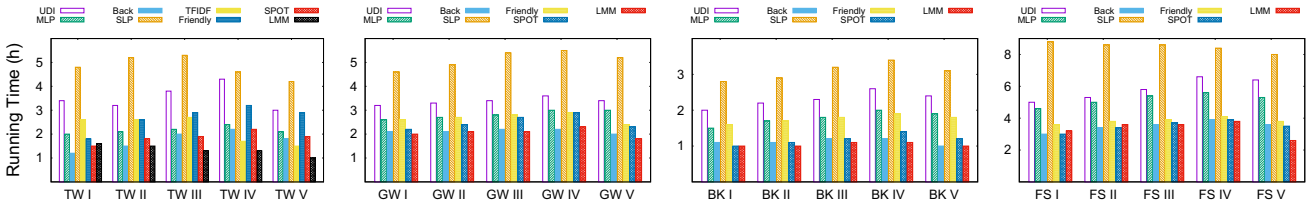


Fig. 22 Prediction coverage of models when local prediction is considered

the location prediction models in each dataset. This means the predicted users are more concentrated in some locations closer to the actual locations co-shared by the neighbors.

6.11 User prediction coverage

Figure 21 shows the user prediction coverage of the models using default number of iterations. The relative coverage of the models in Gowalla and Brightkite is quite similar. *UDI*, *MLP*, and *SLP* models have higher prediction coverage (Fig. 21a) in Twitter, and the remaining models decline significantly with location sparsity. For example, in *Friendly* and *LMM*, the user coverage declines from 97% to 88% and 89% to 75%, respectively, when the sparsity level changes from I to IV. This is because the default settings of these models do not consider multiple iterations. The *TFIDF* model has the lowest prediction coverage among the eight representative models. In Fig. 22, we show the prediction coverage of the models when they are allowed to iterate only once. All the network-based models have lower prediction coverage than the default configuration. However, the prediction coverage in *SLP* model drops significantly from "less" to "extreme" sparsity level, e.g., it drops 21% in Foursquare. The majority of the models in default configuration execute multiple iteration and have similar coverage with the global prediction. Hence, we do not include the global prediction coverage here.



(a) Twitter Dataset **(b)** Gowalla Dataset **(c)** Brightkite Dataset **(d)** Foursquare Dataset

Fig. 23 Running time of the different models

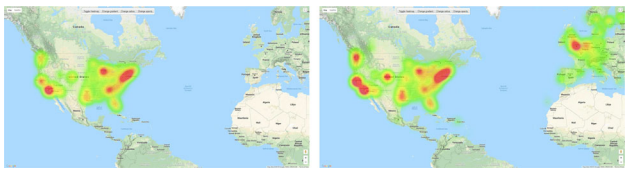
6.12 Running time and memory consumption

The time costs (in hour) of the model-driven prediction process are shown in Fig. 23. The *Backstrom* model is the most time-efficient among the other models, whereas *SLP* consumes the maximum time to process the four large-scale datasets. Meanwhile, the memory consumption does not vary much in different data settings; it depends on the size of the dataset. We report the memory consumption in Twitter as reference to the other datasets. The *Backstrom* model consumes a lower memory of 810 MB, as it only stores the neighbor information while processing each dataset. The memory consumption of *MLP* is higher (e.g., 1725 MB), because it integrates various generative modules and each of the modules stores the *following* and *messaging* information throughout the program. The remaining models have memory costs between 850 MB and 1380 MB.

6.13 Region-specific comparison of overall prediction performance of the models

Different social networks have different region-specific characteristics. Some models may have better performance in predicting users' locations from a specific region. Here, we compare and visualize the proportion of the users predicted within 160 km of error distance with actual locations using Google Maps. We choose *UDI*, *MLP*, *Backstrom*, and *SLP* models to compare the region-specific predictions in Twitter and Brightkite datasets. In Twitter, the majority of the users are distributed in New York and Los Angeles region, whereas the users in Brightkite are spanning over New York, Los Angeles, San Francisco, and London. In Fig. 24, the dark red regions show relatively higher population density in the original Twitter and Brightkite datasets.

In Twitter, the *UDI* model has relatively higher prediction in Chicago and Atlanta region, whereas a lower prediction in Los Angeles area (Fig. 25a). The prediction proportion of the other three models is identical with the original datasets. In Brightkite, the relative prediction of *UDI* and *MLP* in New York, Los Angeles, and San Francisco area is similar with the ground-truth location distribution. However, the predicted



(a) Twitter **(b)** Brightkite

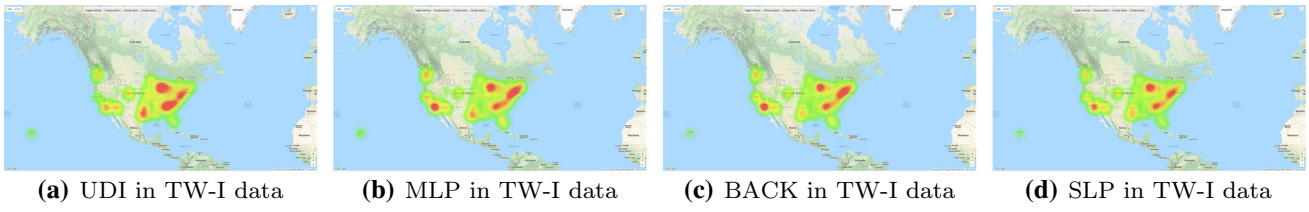
Fig. 24 User location distribution in original datasets

location density using *Backstrom* and *SLP* models is slightly sparse in these three regions (refer Fig. 25).

7 Summary of the findings

Our comprehensive evaluations have brought up many interesting insights that are useful for better understanding of the location prediction models. These insights are helpful in designing and optimizing models for different scenarios such as location sparsity, data type and neighbor types. We summarize our findings below:

- In a dataset that has both network information and social content, the *UDI* model achieves better accuracy (Fig. 11).
- *SLP* model is highly sensitive to location sparsity. The prediction performance of this model drops significantly when the location sparsity increases. On the other hand, *UDI* model is less sensitive to sparsity and can predict precise users' locations in sparse data also. This is because the inner iterations of *UDI* model execute to find the best location by updating the influence scope of each user from minimal location information (Figs. 11–14).
- The performance of *MLP*, *Backstrom*, and *Friendly* models heavily relies on the type of data. Different social networks capture different kinds of users, and hence, they have different probabilities of friendships w.r.t. distances. Therefore, the friendship coefficient parameters must be calculated each time for a new social network dataset.



(a) UDI in TW-I data **(b)** MLP in TW-I data **(c)** BACK in TW-I data **(d)** SLP in TW-I data

(e) UDI in BK-I data **(f)** MLP in BK-I data **(g)** BACK in BK-I data **(h)** SLP in BK-I data

Fig. 25 Heatmap of user prediction in different models using Twitter and Brightkite data setting I

- With respect to execution speed, *UDI* model performs the best on all datasets as it uses local inference (i.e., one iteration). *SLP* model performance (accuracy and prediction coverage) drops significantly with the number of iterations (Figs. 11–14, 15–16, 21–22)
- In terms of training scalability, *SPOT* and *Backstrom* are the most scalable models as their preprocessing time is constant w.r.t. the number of labeled users. However, the preprocessing time of *Friendly* model increases linearly with the increase in number of labeled users.
- *Backstrom* and *LMM* are the most cost-effective models, while *SLP* is the least efficient (Fig. 23).
- Users with moderate node degree (i.e., 10–30) have higher probabilities to be predicted precisely. A large number of neighbors may not substantiate better accuracy to users with high node degree (Fig. 19).
- In datasets with high declination in following probability with distance as in the case of Gowalla and Brightkite datasets, the *Backstrom* and *MLP* models perform better (Figs. 3, 12–13).
- *SLP* model has higher accuracy in datasets with properties similar to Foursquare. However, it suffers from high execution time. If execution time is not an issue, *SLP* model can be the best option to choose. Otherwise, *UDI* is a better option as it strikes a good balance between efficiency and effectiveness (Figs. 14, 23).

8 Conclusion

In this paper, we performed a comprehensive evaluation of eight representative location prediction models on four large-scale real-world datasets. This benchmarking study can also advance research on social computing problems such as uncovering meaningful spatial communities, visiting location recommendation, and location-based event planning. We compared the prediction accuracy of the models using network properties such as friendships and interactions, neighbor proximity, location sparsity, node locality, and degree. We have summarized our key findings of the models with different parameter settings. Our analysis shows that the effectiveness of location prediction is heavily dependent on the richness of neighbor information. In sparse networks, global inference techniques are more effective in prediction tasks. The key findings of this study strongly suggest that service providers can greatly improve the quality of their services by selecting suitable location prediction models based on their application need.

Acknowledgements This research was mainly supported by the ARC Discovery Projects under Grant No. DP160102114 and CSIRO Data61 Scholarship Program. This research is also partially supported by the ARC Linkage Projects under Grant No. LP180100750, Research Grants Council of Hong Kong SAR, China, under Grant No. 14203618 and 14221716. The authors would like to thank Dr. Quanxi Shao and Dr. Cecile Paris (Data61, CSIRO) for their helpful advice, and Prof. Ajmal Mian (UWA) for proofreading the paper.

References

1. Ajao, O., Hong, J., Liu, W.: A survey of location inference techniques on twitter. J. Inf. Sci. **41**(6), 855–864 (2015)
2. Ao, J., Zhang, P., Cao, Y.: Estimating the locations of emergency events from twitter streams. Proc. Comput. Sci. **31**, 731–739 (2014)
3. Backstrom, L., Sun, E., Marlow, C.: Find me if you can: improving geographical prediction with social and spatial proximity. In: Proceedings of the 19th International Conference on World Wide Web, pp. 61–70. ACM (2010)
4. Bao, J., Zheng, Y., Wilkie, D., Mokbel, M.: Recommendations in location-based social networks: a survey. GeoInformatica **19**(3), 525–565 (2015)
5. Bo, H., Cook, P., Baldwin, T.: Geolocation prediction in social media data by finding location indicative words. In: Proceedings of COLING, pp. 1045–1062 (2012)
6. Chang, H.W., Lee, D., Eltaher, M., Lee, J.: @ Phillies tweeting from philly? Predicting twitter user locations with spatial word usage. In:

Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining, pp. 111–118. IEEE (2012)
7. Chen, J., Liu, Y., Zou, M.: From tie strength to function: Home location estimation in social network. In: Computing, Communications and IT Applications Conference (ComComAp), pp. 67–71. IEEE (2014)
8. Chen, Y., Zhao, J., Hu, X., Zhang, X., Li, Z., Chua, T.S.: From interest to function: location estimation in social media. In: Proceedings of the 27th AAAI Conference on Artificial Intelligence, pp. 180–186. AAAI Press (2013)
9. Cheng, Z., Caverlee, J., Lee, K.: You are where you tweet: a content-based approach to geo-locating twitter users. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 759–768. ACM (2010)
10. Cheng, Z., Caverlee, J., Lee, K.: A content-driven framework for geolocating microblog users. ACM Trans. Intell. Syst. Technol. (TIST) **4**(1), 2 (2013)
11. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1082–1090. ACM (2011)
12. Compton, R., Jurgens, D., Allen, D.: Geotagging one hundred million twitter accounts with total variation minimization. In: 2014 IEEE International Conference on Big Data (Big Data), pp. 393–401. IEEE (2014)
13. Davis Jr., C.A., Pappa, G.L., de Oliveira, D.R.R., de L Arcanjo, F.: Inferring the location of twitter messages based on user relationships. Trans. GIS **15**(6), 735–751 (2011)
14. Do, T.H., Nguyen, D.M., Tsiligianni, E., Cornelis, B., Deligiannis, N.: Multiview deep learning for predicting twitter users' location (2017). arXiv:1712.08091
15. Gao, H., Tang, J., Liu, H.: Exploring social-historical ties on location-based social networks. In: International AAAI Conference on Weblogs and Social Media (2012)
16. Gelernter, J., Balaji, S.: An algorithm for local geoparsing of microtext. GeoInformatica **17**(4), 635–667 (2013)
17. Gu, Y., Song, J., Liu, W., Zou, L.: HLGPS: a home location global positioning system in location-based social networks. In: 2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 901–906. IEEE (2016)
18. Han, B., Cook, P., Baldwin, T.: A stacking-based approach to twitter user geolocation prediction. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pp. 7–12 (2013)
19. Hecht, B., Hong, L., Suh, B., Chi, E.H.: Tweets from Justin Bieber's heart: the dynamics of the location field in user profiles. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 237–246. ACM (2011)
20. Jurgens, D.: That's what friends are for: inferring location in online social media platforms based on social relationships. ICWSM **13**, 273–282 (2013)
21. Jurgens, D., Finethy, T., McCorriston, J., Xu, Y.T., Ruths, D.: Geolocation prediction in twitter using social networks: a critical analysis and review of current practice. In: Ninth International AAAI Conference on Web and Social Media, vol. 15, pp. 188–197 (2015)
22. Kong, L., Liu, Z., Huang, Y.: Spot: locating social media users based on social network context. Proc. VLDB Endow. **7**(13), 1681–1684 (2014)
23. Lee, K., Ganti, R.K., Srivatsa, M., Liu, L.: When twitter meets foursquare: tweet location prediction using foursquare. In: Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, pp. 198–207. ICST (2014)
24. Levandoski, J.J., Sarwat, M., Eldawy, A., Mokbel, M.F.: Lars: a location-aware recommender system. In: 2012 IEEE 28th International Conference on Data Engineering (ICDE), pp. 450–461. IEEE (2012)
25. Li, C., Sun, A.: Fine-grained location extraction from tweets with temporal awareness. In: Proceedings of the SIGIR Conference on Research & Development in Information Retrieval, pp. 43–52. ACM (2014)
26. Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A., Lee, B.S.: Twiner: named entity recognition in targeted twitter stream. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 721–730. ACM (2012)
27. Li, R., Wang, S., Chang, K.C.C.: Multiple location profiling for users and relationships from social network and content. Proc. VLDB Endow. **5**(11), 1603–1614 (2012)
28. Li, R., Wang, S., Deng, H., Wang, R., Chang, K.C.C.: Towards social user profiling: unified and discriminative influence model for inferring home locations. In: Proceedings of the 18th ACM SIGKDD, pp. 1023–1031. ACM (2012)
29. Li, W., Serdyukov, P., de Vries, A.P., Eickhoff, C., Larson, M.: The where in the tweet. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 2473–2476. ACM (2011)
30. Lingad, J., Karimi, S., Yin, J.: Location extraction from disaster-related microblogs. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 1017–1020. ACM (2013)
31. Liu, X., Zhang, S., Wei, F., Zhou, M.: Recognizing named entities in tweets. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, pp. 359–367. ACL (2011)
32. Mahmud, J., Nichols, J., Drews, C.: Where is this tweet from? Inferring home locations of twitter users. ICWSM **12**, 511–514 (2012)
33. McGee, J., Caverlee, J., Cheng, Z.: Location prediction in social media based on tie strength. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, pp. 459–468. ACM (2013)
34. Miura, Y., Taniguchi, M., Taniguchi, T., Ohkuma, T.: Unifying text, metadata, and user network representations with a neural network for geolocation prediction. In: Proceedings of the 55th Annual Meeting of the ACL, vol. 1, pp. 1260–1272 (2017)
35. Pang, J., Zhang, Y.: Deepcity: a feature learning framework for mining location check-ins. In: Eleventh AAAI Conference on Web and Social Media (2017)
36. Paul, M.J., Dredze, M.: You are what you tweet: analyzing twitter for public health. ICWSM **20**, 265–272 (2011)
37. Qian, Y., Tang, J., Yang, Z., Huang, B., Wei, W., Carley, K.M.: A probabilistic framework for location inference from social media (2017). arXiv:1702.07281
38. Rahimi, A., Cohn, T., Baldwin, T.: Twitter user geolocation using a unified text and network prediction model (2015). arXiv:1506.08259
39. Rahimi, A., Cohn, T., Baldwin, T.: A neural model for user geolocation and lexical dialectology. In: Proceedings of the 55th Annual Meeting of the ACL, ACL 2017, vol. 2, pp. 209–216 (2017)
40. Rahimi, A., Vu, D., Cohn, T., Baldwin, T.: Exploiting text and network context for geolocation of social media users (2015). arXiv:1506.04803
41. Rakesh, V., Reddy, C.K., Singh, D.: Location-specific tweet detection and topic summarization in twitter. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 1441–1444. ACM (2013)
42. Ren, K., Zhang, S., Lin, H.: Where are you settling down: geolocating twitter users based on tweets and social networks. In: Asia Information Retrieval Symposium, pp. 150–161. Springer (2012)
43. Rout, D., Bontcheva, K., Preoiuc-Pietro, D., Cohn, T.: Where's@ wally? A classification approach to geolocating users based on

their social ties. In: Proceedings of the 24th ACM Conference on Hypertext and Social Media, pp. 11–20. ACM (2013)
44. Ryoo, K., Moon, S.: Inferring twitter user locations with 10 km accuracy. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 643–648. ACM (2014)
45. Sadilek, A., Kautz, H., Bigham, J.P.: Finding your friends and following them to where you are. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, pp. 723–732. ACM (2012)
46. Sakaki, T., Okazaki, M., Matsuo, Y.: Tweet analysis for real-time event detection and earthquake reporting system development. IEEE Trans. Knowl. Data Eng. **25**(4), 919–931 (2013)
47. Scellato, S., Mascolo, C., Musolesi, M., Latora, V.: Distance matters: geo-social metrics for online social networks. In: The Proceedings of 3rd Workshop on Online Social Networks. USENIX Association (2010)
48. Scellato, S., Musolesi, M., Mascolo, C., Latora, V., Campbell, A.T.: Nextplace: a spatio-temporal prediction framework for pervasive systems. In: International Conference on Pervasive Computing, pp. 152–169. Springer (2011)
49. Scellato, S., Noulas, A., Lambiotte, R., Mascolo, C.: Socio-spatial properties of online location-based social networks. In: Fifth International AAAI Conference on Weblogs and Social Media (2011)
50. Scellato, S., Noulas, A., Mascolo, C.: Exploiting place features in link prediction on location-based social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1046–1054. ACM (2011)
51. Sinnott, R.W.: Virtues of the haversine. Sky Telesc. **68**, 159 (1984)
52. Tigunova, A., Lee, J., Nobari, S.: Location prediction via social contents and behaviors: location-aware behavioral LDA. In: International Conference on Data Mining Workshop (ICDMW), pp. 1131–1135. IEEE (2015)
53. Vieweg, S., Hughes, A.L., Starbird, K., Palen, L.: Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1079–1088. ACM (2010)
54. Wang, M., Wang, C., Yu, J.X., Zhang, J.: Community detection in social networks: an in-depth benchmarking study with a procedure-oriented framework. Proc. VLDB Endow. **8**(10), 998–1009 (2015)
55. Xu, W., Chow, C.Y., Zhang, J.D.: CALBA: capacity-aware location-based advertising in temporary social networks. In: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 364–373. ACM (2013)
56. Yamaguchi, Y., Amagasa, T., Kitagawa, H.: Landmark-based user location inference in social media. In: Proceedings of the first ACM Conference on Online Social Networks, pp. 223–234. ACM (2013)
57. Yamaguchi, Y., Amagasa, T., Kitagawa, H., Ikawa, Y.: Online user location inference exploiting spatiotemporal correlations in social streams. In: Proceedings of International Conference on Conference on Information and Knowledge Management, pp. 1139–1148. ACM (2014)
58. Yuan, Q., Cong, G., Ma, Z., Sun, A., Thalmann, N.M.: Who, where, when and what: discover spatio-temporal topics for twitter users. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 605–613. ACM (2013)
59. Zheng, X., Han, J., Sun, A.: A survey of location prediction on twitter. IEEE Trans. Knowl. Data Eng. **30**(9), 1652–1671 (2018)
60. Zhu, X., Ghahramani, Z.: Learning from labeled and unlabeled data with label propagation. Technical report, Citeseer (2002)
61. Zhuang, Y., Fong, S., Yuan, M., Sung, Y., Cho, K., Wong, R.K.: Location-based big data analytics for guessing the next foursquare check-ins. J. Supercomput. **73**(7), 3112–3127 (2017)