Towards Social User Profiling: Unified and Discriminative Influence Model for Inferring Home Locations*

Rui Li†, Shengjie Wang†, Hongbo Deng†, Rui Wang†, Kevin Chen-Chuan Chang†,*
{ruili1, wang260, hbdeng, ruiwang3, kcchang}@illinois.edu
† Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA
* Advanced Digital Sciences Center, Illinois at Singapore, Singapore

ABSTRACT

Users' locations are important to many applications such as targeted advertisement and news recommendation. In this paper, we focus on the problem of profiling users' home locations in the context of social network (Twitter). The problem is nontrivial, because signals, which may help to identify a user's location, are *scarce* and *noisy*. We propose a unified discriminative influence model, named as $\mathcal{UDI}$, to solve the problem. To overcome the challenge of scarce signals, $\mathcal{UDI}$ integrates signals observed from both social network (friends) and user-centric data (tweets) in a unified probabilistic framework. To overcome the challenge of noisy signals, $\mathcal{UDI}$ captures how likely a user connects to a signal with respect to 1) the distance between the user and the signal, and 2) the influence scope of the signal. Based on the model, we develop *local* and *global* location prediction methods. The experiments on a large scale data set show that our methods improve the state-of-the-art methods by 13%, and achieve the best performance.

Categories and Subject Descriptors

H.2.8 [**Data Management**]: Database Applications - Data Mining; H.4.0 [**Information Systems**]: General

General Terms

Algorithms

Keywords

Social Network, Influence Model, Location Profiling

1. INTRODUCTION

User profiling, which infers a user's essential attributes, such as gender, location and interests, has been a holy grail in enabling effective information services. For example, profiling a user's location (which we will focus) or topic interests enables search engines to provide personalized search results, news sites to recommend localized news, and advertisers to serve targeted ads. To profile a user, traditional approaches leverage limited user-centric data (*e.g.*, search log or purchase history).

The emergence of social network services raises both challenges and opportunities for effective user profiling. Recently, online social network services such as Facebook and Twitter become important platforms for users to connect with friends as well as share information. For example, Twitter, a social network for users to follow each other and publish tweets, now has 140 million active users and generates 340 million tweets daily. On one hand, those services need to "understand" their users better, because old tasks (*e.g.*, targeted ads) now become even more challenging (*e.g.*, serving ads without queries), and new tasks (*e.g.*, recommending "friends") arise in the context of social network. On the other hand, those services generate additional information to leverage, because not only user-centric data (*e.g.*, tweets) is available, but also information from others can be propagated through users' social connections.

In this paper, we are particularly interested in profiling "home locations" for Twitter users with both social network (the following network) and user-centric data (tweets). We define a user's *home location* as the place where most of his activities happen. First, a home location is a static geo *scope* (*e.g.*, Chicago) instead of a real-time geo *point* (*e.g.*, the Starbucks on 5th Ave.). Second, it is a user's "permanent" location instead of other locations that are "temporally" related to him (*e.g.*, the places where he is traveling). A user's home location, even when he is "out of town", captures his *major* and *static* geographic scope of interests, which is therefore a useful target for many location-based services as just mentioned.

On Twitter, a user's home location can not be obtained readily. Only a few people (16%) register city level locations (*e.g.*, Chicago, IL) in their profiles. Most of users leave general (*e.g.*, "IL"), nonsensical (*e.g.*, "my home") or even blank information. Although Twitter supports users to add GPS tags in their tweets, even fewer people (0.5%) use this feature due to privacy concerns. Thus, we aim to profile users' locations in the absence of GPS tags.

Intuitively, a user's following network and tweets provide valuable signals for profiling his home location, as he is likely to 1) follow users, who live close, and 2) tweet nearby locations. However, we face two challenges when utilizing the two types of signals.

- **Scarce Signals:** Based on our crawled data, we find that 1) a user has 126 social connections on average, but only 16% of them provide locations, and 2) there are only about 6 location related terms in every 100 messages. Each type of signals alone is not sufficient to profile all users' locations. It is possible that a user has few social connections and none of them provides a location. It is also possible that a user does not tweet but only consumes information from others.

*This material is based upon work partially supported by NSF Grant IIS 1018723 and the Advanced Digital Science Center of the University of Illinois at Urbana-Champaign. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the funding agencies.

KDD'12, August 12–16, 2012, Beijing, China.

- **Noisy Signals:** A user follows friends from or publishes tweets about different locations other than his home location. Some of them are far away. For example, a user in Chicago may follow Lady Gaga in New York or President Obama in Washington, and tweet about Houston Rocket's game or his vacation in Honolulu.

In this paper, we propose a unified discriminative "influence" model, named as $\mathcal{UDI}$, to tackle the above challenges.

Unified Signals With respect to the scarce signal challenge, $\mathcal{UDI}$ integrates the two types of signals in a unified probabilistic framework. To the best of our knowledge, it is the first that integrates social network and user-centric data for the location profiling task.

To integrate different types of signals (*e.g.*, locations of friends and from tweets), we first abstract them with a unified view. Specifically, we view them as a heterogenous graph, where a user connects to the two types of signals via "following" and "tweeting" edges. Then, we take a probabilistic generative approach to model them jointly. We assume every edge (*e.g.*, a tweeting edge) is "generated" according to the two end nodes' locations (*e.g.*, a user and a tweeted venue), and model the *joint conditional probability* of generating all the edges given the nodes' locations. We estimate the unknown locations as latent variables in the probability.

Discriminative Influence With respect to the noisy signal challenge, $\mathcal{UDI}$ models how likely an edge is "generated" from a head node (*e.g.*, a user) to a tail node (*e.g.*, a tweeted venue) with respect to 1) the distance between them, and 2) the influence scope of the tail node. It successfully captures "closeness" and "credibility" of each signal, and therefore is robust to noisy signals.

- **Influence at different distances:** $\mathcal{UDI}$ captures that 1) a node (*e.g.*, a user) has *influence probabilities* at different locations to attract a user there to build an edge (*e.g.*, a following edge), and 2) a node's influence probability at a location decreases as its distance to the node increases. Thus, $\mathcal{UDI}$ not only exploits our intuition that a user is likely to follow users from or tweet about nearby locations, but also tolerates noisy signals that he may follow friends from and tweet about locations far away. When predicting his location, our model can successfully identify that his location is close to the most dominating region among those of his friends and tweeted venues. *E.g.*, a user has three friends from New York, Chicago, and Champaign (a small town in Illinois) respectively, our model is able to find that he is in Illinois.
- **Influence Scope of each node:** $\mathcal{UDI}$ captures that each node has its own *influence scope*. Intuitively, an influential node (*e.g.*, Lady Gaga) with a "broad" influence scope is more likely than a regular node (*e.g.*, a real friend) to be followed or tweeted by a user far away, and therefore its location is more likely to be a noisy signal for predicting the user's location. Thus, our model overcomes noisy signals by discriminating the locations of influential nodes from the locations of regular nodes. When predicting a user's location, our model can automatically weigh a node (*e.g.*, a real friend) with a narrow influence scope more than a node (*e.g.*, Lady Gaga) with a broad scope.

To mathematically model all users' influence models, we choose a set of discriminative Gaussian distributions. For each node, a gaussian distribution has its center L and variance σ representing the node's location and its influence scope, respectively. A node's influence probability at a location L' is measured as the probability at the corresponding distance of L' from L in the distribution. The simplicity of a gaussian distribution enables us to learn its parameters for each node with scarce signals, and thus results in "rich" modeling– every node has its own unique influence model.

Based on $\mathcal{UDI}$, we develop two location prediction methods with the maximum likelihood (MLE) principle. Our *local prediction method* predicts a user's location by maximizing the probability of generating edges to his "local" signals, *i.e.*, locations of his friends and tweets. We further extend the local scheme to a *global prediction method*. Intuitively, a user's unlabeled friends are useful since their own labeled friends or tweets may indicate their locations explicitly, so as to enhance the prediction of the given user. Thus, we maximize the probability of generating edges to all the signals on the entire graph, and derive an iterative algorithm to make more accurate predictions. We also prove the convergence of the algorithm. In addition, we enhance our prediction methods by using human knowledge (*e.g.*, users only live in cities but not arbitrary geo points) as *constraints*. Those constraints help us to learn a more accurate model with scarce signals.

As a byproduct, $\mathcal{UDI}$ also identifies the influence scope of each node, which is new and different from the "influence score" studied by earlier work [5]. The influence scope measures the broadness in terms of physical distance of a node's influence over the geo space, while the influence score measures how good a node is in spreading information over a social network. A node (*e.g.*, the New York weather channel) can have a large influence score but a small influence scope. In this paper, we use the influence scope to discriminate the credibility of each node in predicting locations, but we see many interesting applications beyond this setting, such as differentiating global authorities (*e.g.*, Lady Gaga) and local authorities (*e.g.*, Texas Representative).

Finally, we conduct extensive experiments to evaluate our prediction methods and compare with the state-of-the-art methods [4, 7] based on a large-scale Twitter data set containing about 160K users and 50 million tweets. The experimental results show that our prediction methods significantly improve the best baseline method by 13%, and achieve accurate results. Particularly, our global method can place 66% users within 100 miles, and the average error distance for its top 60% predictions is less than 5 miles.

2. RELATED WORK

In this section, we discuss some related work, including user profiling and location prediction.

User Profiling Due to the importance of user profiling, many interesting studies have been done on this problem. Most of them focus on profiling users' "topic interests" to serve personalized search [13, 17], targeted advertisement [1, 12], and news recommendation [16]. They mainly explore user-centric data, including query logs [13], browsing behaviors [16] and other types of user generated data [12, 17]. Our work is different in two aspects. First, we aim to profile locations. Second, we explore not only user-centric data (*i.e.*, tweets) but also social network data.

As the rise of social network services, some seminal studies [18, 11] explore social network for user profiling. Yang et al. [18] propose a model to propagate interests of an item among users via their friendships. However, users' locations are different from their interests of an item, and can not be propagated directly. Mislove et al. [11] use friendships to infer Facebook users' attributes. They apply a clustering algorithm to find communities in the network and assign an identical attribute value to users in the same community. Although this method is supposed to work for different types of attributes, it fails in predicting locations, as users follow others living far away and communities are not directly formed based on users' locations. It does not leverage user-centric data as well.

Location Prediction As we focus on profiling users' locations, our work is related to identifying geographical scopes for various kinds of online entities, such as pages [2], queries [3], tags [14], and photos [8]. However, they predict locations for different types of entities with different resources. For example, Amitay et al. [2] explore a web page's content to predict its geo scope based on heuristic rules. Their method extracts location signals (*e.g.*, city names) from a page and uses a gazetteer to find the geo region that covers most of the signals as the region of the page. Our work is different, as we take a probabilistic approach to profile users' locations. Backstrom et al. [3] propose a probabilistic model to assign a geographic center and a rate of diffusion to a query based on the usage of the query. Our method is different from it, as we focus on utilizing social network and tweets in a unified and discriminative approach. Furthermore, our prediction methods are able to utilize additional human knowledge.

Our work is most related to [7, 4], as they also focus on the same user location prediction problem. Cheng et al. [7] estimate a user's location based on the content of his tweets. Specifically, they identify a set of location related words (*e.g.*, "chicago") and use them as features to classify the user to locations. Recently, Chandra et. al. [6] improve this model slightly by associating a user's original tweets to him, and his retweets to the initial user. However, they both treat local words and locations as discrete labels and overlook their explicit relations (*e.g.*, distances between them). Backstrom et al. [4] estimate a user's location based on his friends on Facebook. They first learn a function, which assigns the probability of being friends given the distance of two users, and then estimate a user's location based on MLE. However, their model assumes the probability of being friends given the same distance is the same for different users. This assumption usually does not hold, especially on Twitter. *E.g.*, a famous user is more likely to have a follower far away than a regular user does. Therefore, their model can not differentiate signals with different credibilities. $\mathcal{UDI}$ not only overcomes the disadvantages of the above methods, but also has the following advantages: 1) it models both content and social network, 2) it utilizes relationships from both labeled and unlabeled users, and 3) it supports integrating additional human knowledge.

3. PROBLEM FORMULATION

In this section, we first abstract different types of signals as a heterogeneous graph, and then formalize our problem from there.

Twitter is a social network, where users follow others and publish messages. Given a user, we identify two important types of signals: 1) *following relationships* between the user and other users, and 2) *tweets or messages* tweeted by the user. We note that following relationships are "directional", which means if a user u_i follows a user u_j, u_j does not necessarily follow back. Thus, we further divide a user's following relationships into *followers* who follow the user and *friends* who are followed by the user.

Both types of signals are useful for inferring a user's location. As Sec. 1 mentioned, a user is likely to 1) follow and be followed by users, who live close to him, and 2) mention some "venues" (*e.g.*, Chicago), which may indicate his location. We refer a *venue* as a signal for a place, which could be a city (*e.g.*, Chicago), a place (*e.g.*, Time Square), or an entity with a specific geo position (*e.g.*, Stanford University). If some of a user's followers or friends provide locations in their profiles, we can propagate their locations to him. If a user mentions some venues in his messages, we can use them to infer his location as well.

As shown in Fig. 1, we abstract different types of signals as a directed heterogeneous graph $G = (N, E)$, where N is a set of nodes n_i and E is a set of edges $e\langle n_i, n_j\rangle$ from a tail node n_i to a head node n_j. N contains two types of nodes, *user nodes* U representing all the users and *venue nodes* V representing all the venues tweeted by users. $N = U \cup V$. E contains two types of edges, each of which designates a specific type of relationships between nodes: 1) *following edges* F between user nodes, and 2) *tweeting edges* T between user nodes and venue nodes. $E = F \cup T$. A *following edge* $f\langle u_i, u_j\rangle$ is formed from a user u_i to another user u_j when u_i follows u_j, where u_i is a *follower* of u_j, and u_j is a *friend* of u_i. A *tweeting edge* $t\langle u_i, v_j\rangle$ is formed from a user u_i to a venue v_j, when u_i tweets v_j. As u_i can tweet v_j many times, we use w_{ij} to denote the frequency.

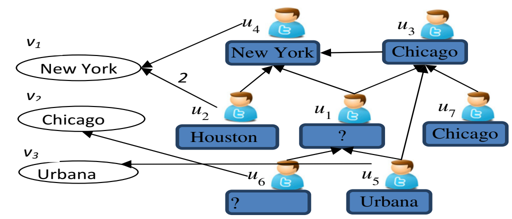


Figure 1: An Example of Twitter Graph.

Generally, every node n_i in the graph is associated with a location, denoted as $\mathcal{L}_{n_i}$. We view $\mathcal{L}_{n_i}$ as a point (X, Y) on the geo space, where X denotes the latitude and Y denotes the longitude. Some user nodes' locations are missing. Our goal is to profile them. We call the users with known locations as *labeled users*, denoted as U^*, and the remaining users as *unlabeled users*, denoted as U^N. $U = U^* \cup U^N$. Formally, our problem can be stated as:

Location Profiling Problem Given a Twitter graph $G(U \cup V, T \cup F)$, $\mathcal{L}_{u_j}$ for $u_j \in U^*$, and $\mathcal{L}_{v_j}$ for $v_j \in V$, estimate a location $\hat{\mathcal{L}}_{u_i}$ for each user $u_i \in U^N$ so as to make $\hat{\mathcal{L}}_{u_i}$ close to u_i's true location $\mathcal{L}_{u_i}$.

As we motivated in Sec. 1, a user is related to inconsistent and noisy locations on the graph, so the problem is non-trivial. We propose a unified discriminative influence based probabilistic framework ($\mathcal{UDI}$) to solve it. Specifically, in Sec. 4, we describe our probabilistic model, which measures how likely an edge is generated between two nodes with respect to their locations. In Sec. 5, we present our prediction methods, which estimate a user's location by maximizing the probability of generating the observed edges.

Notation Before our discussion, we introduce some notations. Generally, we use $\mathcal{I}_e(n)$ and $\mathcal{O}_e(n)$ to denote incoming neighbor nodes of a node n through type e edges, and outgoing neighbors of n through type e edges respectively. Specifically,

- $\mathcal{I}_f(u_i) = \{u_j \in U | f\langle u_j, u_i\rangle \in F\}$ denotes the followers of u_i, and $\mathcal{I}_f^*(u_i) = \mathcal{I}_f(u_i) \cap U^*$ denotes the *labeled followers* of u_i
- $\mathcal{O}_f(u_i) = \{u_j \in U | f\langle u_i, u_j\rangle \in F\}$ denotes the friends of u_i, and $\mathcal{O}_f^*(u_i) = \mathcal{O}_f(u_i) \cap U^*$ denotes the *labeled friends* of u_i.
- $\mathcal{O}_t(u_i) = \{v_j \in V | t\langle u_i, v_j\rangle \in T\}$ denotes venues tweeted by u_i.
- $\mathcal{I}_t(v_i) = \{u_j \in U | t\langle u_j, v_i\rangle \in T\}$ denotes the users who tweet v_i. $\mathcal{I}_t^*(v_i) = \mathcal{I}_t(v_i) \cap U^*$ denotes the labeled users who tweet v_i.

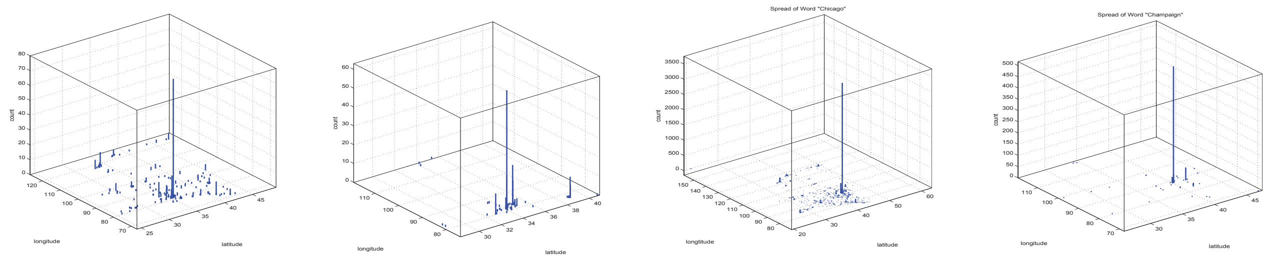
4. INFLUENCE MODEL

In this section, we introduce a probabilistic model named as *influence model* to measure how likely a tail node n_j (*e.g.*, a user u_j) at a location $\mathcal{L}_{n_j}$ builds an edge $e\langle n_j, n_i\rangle$ (*e.g.*, a following edge) to a head node n_i (*e.g.*, a user u_i) at a location $\mathcal{L}_{n_i}$.

4.1 Motivation

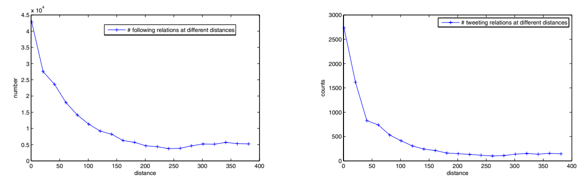
To motivate our model, we investigate about 139,180 randomly crawled Twitter users and observe two key characteristics of the probability that there is $e\langle n_j, n_i\rangle$ from n_j to n_i.

First, the probability decreases as the distance from n_j to n_i increases. Specifically, Fig. 2(a) and 2(b) show the average numbers of followers of a user and the average numbers of users who tweet



(a) Anonymous1's followers (b) Anonymous2's followers (c) Chicago's Users (d) Champaign's Users

Figure 3: Numbers of Relations over the Geo Space



(a) Following Edges (b) Tweeting Edges

Figure 2: Numbers of Edges versus Distances

a venue at different distances. Fig. 2(a) illustrates that generally users have more followers living close than far away, which means that a user, as head node, is more likely to attract users living close to follow. The reason might be that a user's followers tend to know him in real life and are likely to live close to him. This property has also been observed from Facebook network [4] and other social networks [10]. Here we validate it on Twitter network. Similarly, Fig. 2(b) shows that a venue, as a head node, is more likely to attract users living close to tweet about it, because users are more likely to be interested in things happening around.

Second, at the same distance, different head nodes have different probabilities to attract tail nodes. Fig. 3(a) and 3(b) show the numbers of followers of two specific users on Twitter, Anonymous1 and Anonymous2, over the geographic space. Comparing Fig. 3(a) and 3(b), we can tell that Anonymous1, as an influential user, is more likely to attract users who live far away to build following edges than a regular user Anonymous2, because Anonymous1 has a broader influence scope than a regular user in real life. Fig. 3(c) and 3(d) show the numbers of users, who tweet two specific locations, Chicago and Champaign, at different locations. Similarly, we find that Chicago, as an influential city, is much more likely to be tweeted by users who live far away than Champaign, as cities such as Chicago or New York, are more influential than regular cities.

4.2 Model Formulation

Our influence model aims to capture the above characteristics. Conceptually, the *influence model* of a node n_i, denoted as θ_{n_i}, is a probability distribution over the geographic plane, which assigns an "influence probability" to any geo point in the plane. n_i's *influence probability* at a point L represents the probability that n_i influences another node n_j at L to build an edge $e\langle n_j, n_i\rangle$ to it. The higher n_i's influencing probability is, the more likely n_j is to build $e\langle n_j, n_i\rangle$ to n_i. Different nodes have different *influence scopes*. A node with a "broad" influence scope has a larger influence probability at a point far away than a node with a "narrow" influence scope does.

The influence model enables us to measure the probability of observing $e\langle n_j, n_i\rangle$ from n_j to n_i in a generative way. Specifically, we can assume $e\langle n_j, n_i\rangle$ is "generated" according to n_i's influence probability at $\mathcal{L}_{n_j}$, $P(e\langle n_j, n_i\rangle|\theta_{n_i}, \mathcal{L}_{n_j}) = P(\mathcal{L}_{n_j}|\theta_{n_i})$.

Probability Model Mathematically, we need a probability distribution to represent a node's influence model. We reason that an "ideal" distribution should satisfy the following requirements.

- *Expressiveness:* It should capture: 1) probabilities decrease as distances increase, and 2) each node has its own influence scope.
- *Simplicity:* Its parameters should be simple to estimate, as we only have a few observations for each node.

In this paper, we choose a gaussian distribution to capture a node's influence model. In terms of expressiveness, either the heavy tailed distribution [4, 3] or the gaussian distribution [15, 19], which has been widely used for modeling probabilities over the geo space, can be used in our case. In terms of simplicity, a heavy tailed distribution uses several parameters (*e.g.*, α and β in the form of $(\alpha + d)^\beta$ in [4]), while a simple gaussian uses only one parameter (*e.g.*, σ in the form of $N(0, \sigma)$). Thus, a heavy tailed distribution requires more observations than a gaussian for estimating parameters. *E.g.*, in [4], they use observations from all the users to estimate one heavy tailed distribution, and use it to model all the users. In our case, as we aim to estimate a unique gaussian distribution for each node with scarce observations related to the node, we choose a simple gaussian distribution for each node.

We must emphasize that our choice of the gaussian distribution neither conflicts with the heavy tailed distribution observed in [4], nor limits our model's prediction power. First, the heavy tailed distribution is observed based on the aggregation of all users, but we use a gaussian to model each individual. Second, our model uses millions of gaussian distributions, each of which is tailored to a user. It fits each individual better and is more flexible in general than one heavy tailed distribution. As our experiment in Sec. 6 will show, it profiles users' locations more accurately than the method [4] based on the heavy tailed distribution with the same amount of observations.

Specifically, we model a node n_i's influence model θ_{n_i} as a bivariate gaussian distribution, $N(\mathcal{L}_{n_i}, \Sigma_{u_i})$, centered at n_i's location $\mathcal{L}_{n_i} = (X_{n_i}, Y_{n_i})$ and with the covariance matrix Σ_{u_i} as its influence scope. While our model can generally take different variances along the X and Y dimensions as well as their covariances, we assume the influence scope of a node on the X and Y dimensions is the same, as it is easy to estimate with few observations and there isn't clear evidence for "non-symmetric" distributions on X and Y. Therefore, $\Sigma_{u_i} = \begin{pmatrix} \sigma_{n_i} & 0 \\ 0 & \sigma_{n_i} \end{pmatrix}$, and n_i's influence probability at a location L is measured as follows.

$$P(L|\theta_{n_i}) = \frac{1}{2\pi\sigma_{n_i}^2} e^{\frac{(X_{n_i} - X_L)^2}{-2\sigma_{n_i}^2} + \frac{(Y_{u_i} - Y_L)^2}{-2\sigma_{n_i}^2}} \quad (1)$$

To measure probabilities of generating following and tweeting edges, we instantiate two types of influence models.

User Influence Model is to measure $P(f\langle u_j, u_i\rangle|\theta_{u_i}, \mathcal{L}_{u_j})$, the conditional probability that a user u_i influences a user u_j at a location $\mathcal{L}_{u_j}$ to build a following edge $f\langle u_j, u_i\rangle$ to him given u_i's influence model θ_{u_i} and $\mathcal{L}_{u_j}$. We interpret it as follows.

$$P(f\langle u_j, u_i\rangle|\theta_{u_i}, \mathcal{L}_{u_j}) = \frac{1}{2\pi\sigma_{u_i}^2} e^{\frac{(X_{u_i}-X_{u_j})^2+(Y_{u_i}-Y_{u_j})^2}{-2\sigma_{u_i}^2}} \quad (2)$$

Venue Influence Model is to measure $P(t\langle u_j, v_i\rangle|\theta_{v_i}, \mathcal{L}_{u_j})$. Similarly, we interpret it as follows.

$$P(t\langle u_j, v_i\rangle|\theta_{v_i}, \mathcal{L}_{u_j}) = \frac{1}{2\pi\sigma_{v_i}^2} e^{\frac{(X_{v_i}-X_{u_j})^2+(Y_{v_i}-Y_{u_j})^2}{-2\sigma_{v_i}^2}} \quad (3)$$

Conditional Independence Assumption When modeling the probability of generating an edge, we take a *conditional independence assumption*. Specifically, we assume that *each edge (e.g., a tweeting edge) from a tail node (e.g., a user) to a head node (e.g., a venue) is conditionally independent given the head node's influence model and the tail node's location*. In other words, if the head node's influence model and the tail node's location are given, any additional observation (*e.g.*, other nodes or edges) will not affect the probability of generating the edge.

We are aware that, in reality, various factors affect the probability of generating an edge between two nodes. For example, if two nodes share common neighbors, the probability that there is an edge will increase. However, capturing any additional dependency requires additional parameters. The scarce observations and the complexity of estimation prevent us from modeling those comprehensive dependencies. To focus on the location factor only, we simplify our model with the above assumption. This assumption is widely applied in generative models (*e.g.*, Naive Bayes and topic modeling), which our model belongs to, for simplifying models and focusing on key factors. As our experiments will show, like other generative models, our model achieves promising results with the assumption. We further note that this assumption has also been used in other location prediction tasks [3, 4].

5. LOCATION PROFILING METHODS

In this section, we develop our location profiling methods based on the *Maximum Likelihood Estimation* (MLE) principle under the $\mathcal{UDI}$ framework. Specifically, we profile a user's location as the location that maximizes the joint probability of generating following and tweeting edges from and to his followers, friends and tweeted venues. We derive two prediction methods, a local one and a global one, which aim to balance efficiency and effectiveness.

5.1 Local Prediction Method

We first develop a *local prediction method*, which infers a user u_i's location via using locations observed from his "local" edges directly. A user's local edges are the edges which directly connect to him. However, some of them connect to nodes without locations (*e.g.*, an unlabeled friend), and they do not provide any location signal directly. In this setting, to simplify the problem and derive an efficient algorithm, we assume *we only observe the edges between the user and the label nodes*. Specifically, they are: 1) the following edges from his labeled followers, denoted as $f\langle U^*, u_i\rangle = \{f\langle u_j, u_i\rangle \in F | u_j \in U^*\}$, 2) the following edges to his labeled friends, denoted as $f\langle u_i, U^*\rangle = \{f\langle u_i, u_j\rangle \in F | u_j \in U^*\}$, and 3) the tweeting edges to the venues tweeted by him, denoted as $t\langle u_i, V\rangle = \{t\langle u_i, v_j\rangle \in T | v_j \in V\}$.

Based on our influence model, the probability of observing those edges depends on the following factors: 1) the probability of observing $f\langle U^*, u_i\rangle$ from u_i's labeled followers $\mathcal{I}_f^*(u_i)$ to u_i depends on u_i's influence model θ_{u_i} and the locations of $\mathcal{I}_f^*(u_i)$, denoted as $\mathcal{L}_{\mathcal{I}_f^*(u_i)}$, 2) the probability of observing $f\langle u_i, U^*\rangle$ from u_i to his labeled friends $\mathcal{O}_f^*(u_i)$ depends on u_i's location $\mathcal{L}_{u_i}$ and the influence models of $\mathcal{O}_f^*(u_i)$, denoted as $\theta_{\mathcal{O}_f^*(u_i)}$, and 3) the probability of observing $t\langle u_i, V\rangle$ from u_i to his tweeted venues $\mathcal{O}_t(u_i)$ depends on u_i's location $\mathcal{L}_{u_i}$ and the influence models of $\mathcal{O}_t(u_i)$, denoted as $\theta_{\mathcal{O}_t(u_i)}$.

Likelihood Function Given parameters $\theta_{u_i}, \mathcal{L}_{u_i}, \mathcal{L}_{\mathcal{I}_f^*(u_i)}, \theta_{\mathcal{O}_f^*(u_i)}$ and $\theta_{\mathcal{O}_t(u_i)}$, we write the joint conditional probability (the likelihood function) of observing $f\langle U^*, u_i\rangle$, $f\langle u_i, U^*\rangle$ and $t\langle u_i, V\rangle$ as Eq. (4). At step 1, we express the joint conditional probability as the product of $P(e\langle n_j, n_i\rangle|\theta_{n_i}, \mathcal{L}_{n_j})$ based on the conditional independence assumption. $t\langle u_i, v_j\rangle$ is multiplied w_{ij} times, as each $t\langle u_i, v_j\rangle$ appears w_{ij} times in $t\langle u_i, V\rangle$. At step 2, we represent $P(e\langle n_j, n_i\rangle|\theta_{n_i}, \mathcal{L}_{n_j})$ as n_i's influence probability at $\mathcal{L}_{n_j}$ based on our influence model.

$$\begin{aligned}
&P(f\langle U^*, u_i\rangle, f\langle u_i, U^*\rangle, t\langle u_i, V\rangle|\mathcal{L}_{u_i}, \theta_{u_i}, \mathcal{L}_{\mathcal{I}_f^*(u_i)}, \theta_{\mathcal{O}_f^*(u_i)}, \theta_{\mathcal{O}_t(u_i)}) \\
&=^1 \prod_{u_j \in \mathcal{I}_f^*(u_i)} P(f\langle u_j, u_i\rangle|\theta_{u_i}, \mathcal{L}_{u_j}) \times \prod_{u_j \in \mathcal{O}_f^*(u_i)} P(f\langle u_i, u_j\rangle|\theta_{u_j}, \mathcal{L}_{u_i}) \\
&\times \prod_{v_j \in \mathcal{O}_t(u_i)} P(t\langle u_i, v_j\rangle|\mathcal{L}_{u_i}, \theta_{v_j})^{w_{ij}} \\
&=^2 \prod_{u_j \in \mathcal{I}_f^*(u_i)} \frac{1}{2\pi\sigma_{u_i}^2} e^{\frac{(X_{u_i}-X_{u_j})^2+(Y_{u_i}-Y_{u_j})^2}{-2\sigma_{u_i}^2}} \\
&\times \prod_{u_j \in \mathcal{O}_f^*(u_i)} \frac{1}{2\pi\sigma_{u_j}^2} e^{\frac{(X_{u_i}-X_{u_j}+(Y_{u_i}-Y_{u_j})^2)^2}{-2\sigma_{u_j}^2}} \\
&\times \prod_{v_j \in \mathcal{O}_t(u_i)} \left(\frac{1}{2\pi\sigma_{v_j}^2} e^{\frac{(X_{u_i}-X_{v_j})^2+(Y_{u_i}-Y_{v_j})^2}{-2\sigma_{v_i}^2}}\right)^{w_{ij}}
\end{aligned} \quad (4)$$

Based on MLE, we find parameters, u_i's location $\mathcal{L}_{u_i}$ and u_i's influence scope σ_{u_i}, by maximizing the above equation, and use the estimated $\mathcal{L}_{u_i}$ as u_i's location.

However, in Eq. (4), besides $\mathcal{L}_{u_i}$ and σ_{u_i}, which we aim to estimate, there are other unknown parameters. Particularly, for each labeled friend $u_j \in \mathcal{O}_f^*(u_i)$ and each tweeted venue $v_j \in \mathcal{O}_t(u_i)$, their influence scopes σ_{u_j} and σ_{v_j} are unknown, as we only observe their locations. In our local prediction setting, we assume each labeled node's influence scope can be accurately estimated with its labeled neighbors as well. Thus, we estimate them before predicting the user's location, and view them as the known parameters. Next, we discuss how to estimate them.

Influence Scope of a Friend To estimate σ_{u_j} in a labeled friend u_j's influence model θ_{u_j}, we can use u_j's following relationships from his labeled followers. Among u_j's edges, only u_j's following edges $f\langle U, u_j\rangle$ from his followers depend on θ_{u_j}. As those edges also depend on his followers' locations, we use u_j's following edges $f\langle U^*, u_j\rangle$ from his labeled followers $\mathcal{I}_f^*(u_j)$ as observations, and estimate θ_{u_j} by maximizing the joint conditional probability of observing $f\langle U^*, u_j\rangle$ given θ_{u_j} and $\mathcal{L}_{\mathcal{I}_f(u_j)}$. We write the probability as Eq. (5).

$$\begin{aligned}
P(f\langle U^*, u_j\rangle|\theta_{u_j}, \mathcal{L}_{\mathcal{I}_f^*(u_j)}) &= \prod_{u_k \in \mathcal{I}_f^*(u_j)} P(f\langle u_k, u_j\rangle|\theta_{u_j}, \mathcal{L}_{u_k}) \\
&= \prod_{u_k \in \mathcal{I}_f^*(u_j)} \frac{1}{2\pi\sigma_{u_j}^2} e^{\frac{(X_{u_j}-X_{u_k})^2+(Y_{u_j}-Y_{u_k})^2}{-2\sigma_{u_j}^2}}
\end{aligned} \quad (5)$$

In Eq. (5), σ_{u_j} is the only unknown variable, as u_j is a labeled user and u_k is his labeled follower. We directly estimate σ_{u_j} by maximizing Eq. (5). Technically, we get its closed-form solution by differentiating Eq. (5) with respect to σ_{u_j} and setting the result to zero. Eq. (6) shows the solution.

$$\sigma^2_{u_j} = \sum_{u_k \in \mathcal{I}^*_f(u_j)} \frac{(X_{u_j} - X_{u_k})^2 + (Y_{u_j} - Y_{u_k})^2}{2|\mathcal{I}^*_f(u_j)|} \tag{6}$$

Influence Scope of a Venue Similarly, to estimate a venue v_j's influence scope σ_{v_j}, we use the tweeting edges from v_j's labeled twitters, denoted as $t\langle U^*, v_j\rangle = \{t\langle u_i, v_j\rangle \in T | u_i \in U^*\}$. We derive σ_{v_j} by maximizing the conditional probability of generating $t\langle U^*, v_j\rangle$ given v_j's influence model θ_{v_j} and labeled twitter's locations $\mathcal{L}_{\mathcal{I}^*_t(v_j)}$. We write the condition probability as Eq. (7), and derive σ_{v_j} in Eq. (8).

$$P(t\langle U^*, v_j\rangle|\theta_{v_j}, \mathcal{L}_{\mathcal{I}^*_t(v_j)}) = \prod_{u_i \in \mathcal{I}^*_t(v_j)} P(t\langle u_i, v_j\rangle|\theta_{v_j}, \mathcal{L}_{u_i})^{w_{ij}} \tag{7}$$

$$\sigma^2_{v_j} = \sum_{u_i \in \mathcal{I}^*_t(v_j)} \frac{w_{ij}((X_{u_i} - X_{v_j})^2 + (Y_{u_i} - Y_{v_j})^2)}{2\sum_{u_i \in \mathcal{I}^*_t(v_j)} w_{ij}}. \tag{8}$$

Solution Now each tweeted venue v_j's σ_{v_j} and $\mathcal{L}_{v_j}$, each labeled friend u_j's $\mathcal{L}_{u_j}$ and σ_{u_j}, and each labeled follower u_j's $\mathcal{L}_{u_j}$ are known. $\mathcal{L}_{u_i}$ and σ_{u_i} are the unknown variables left. We estimate them by maximizing Eq. (4). We first differentiate Eq. (4) with regard to $\mathcal{L}_{u_i}$ and σ_{u_i}, and obtain Eq. (9) and Eq. (10), which show $\mathcal{L}_{u_i}$ and σ_{u_i} depend on each other. We substitute Eq. (10) for σ_{u_i} in Eq. (9), and obtain a polynomial function of $\mathcal{L}_{u_i}$. We apply the Newton–Raphson method to find its solution, and derive σ_{u_i} accordingly. We note that because X_{u_i} and Y_{u_i} are symmetric in Eq. (4), the solutions for X_{u_i} and Y_{u_i} are in the same form. Due to the space limit, we only give the solution for X_{u_i}.

$$X_{u_i} = \frac{\sum_{u_j \in \mathcal{I}^*_f(u_i)} \frac{X_{u_j}}{\sigma^2_{u_i}} + \sum_{u_j \in \mathcal{O}^*_f(u_i)} \frac{X_{u_j}}{\sigma^2_{u_j}} + \sum_{v_j \in \mathcal{O}_t(u_i)} \frac{w_{ij} X_{v_j}}{\sigma^2_{v_j}}}{\sum_{u_j \in \mathcal{I}^*_f(u_i)} \frac{1}{\sigma^2_{u_i}} + \sum_{u_j \in \mathcal{O}^*_f(u_i)} \frac{1}{\sigma^2_{u_j}} + \sum_{v_j \in \mathcal{O}_t(u_i)} \frac{w_{ij}}{\sigma^2_{v_j}}} \tag{9}$$

$$\sigma^2_{u_i} = \sum_{u_j \in \mathcal{I}^*_f(u_i)} \frac{(X_{u_j} - X_{u_i})^2 + (Y_{u_j} - Y_{u_i})^2}{2|\mathcal{I}^*_f(u_i)|} \tag{10}$$

The above solution also works for the cases that only a subset of resources (*e.g.*, tweets) is used, as we can simply view the unused resource as an empty set in our solution.

Interpretation The above solution can be interpreted meaningfully. As Eq. (10) shows, the influence scope of u_i will be large if u_i's followers are far away from him. Celebrities (*e.g.*, Lady Gaga) will get large influence scopes as their followers are distributed broadly. As Eq. (9) shows, when we estimate a user's location, each node contributes differently, where the weight of a node is inversely proportional to its influence scope. *E.g.*, if we profile a user's location using two friends of him, *e.g.*, Lady Gaga and a regular user, the prediction is close to the regular user, as Lady Gaga has a broad influence scope, and her location is likely to be a noisy signal.

Computation Complexity The algorithm computes a user's location in $O(K^2)$, where K is the average number of edges associated with a user and is less than a hundred. Specifically, it first computes influence scopes for K neighbors of the user, and each of them requires $O(K)$. Then, it uses $O(tK)$ to estimate the location with K edges, where t is the number of iterations in the Newton method. Theoretically, t is $O(d\log^2(d))$ for d digits precision, which is a small constant and can be ignored. In practice, we can precompute the influence scope for each labeled node, and the complexity is reduced to $O(K)$. The algorithm can be viewed as an online algorithm, which efficiently infers a user's location at real-time.

5.2 Global Prediction Method

We further develop a *global prediction method*, which infers a user's location via using all the edges in the graph, and profile users' locations more accurately than the local one.

To motivate our method, we argue that unlabeled users are valuable as we can propagate locations of their tweets, followers and friends to them. Let us revisit the example in Fig. 1. Although u_6 is unlabeled, we can tell u_6 is close to Chicago as he tweets Chicago. As a result, u_6 becomes an additional observation, which suggests that u_1 should be close to Chicago. However, unlabeled users can not be directly used, because we can not tell which unlabeled user we should predict first, say, u_1 or u_6, and how to propagate a user's predicted location to others.

We develop our global prediction method to model all the edges in the graph and utilize all the observed locations. Specifically, it models the joint conditional probability of observing all the following edges F and tweeting edges T in the graph given all the nodes' locations and influence models, and estimates all unlabeled users' locations together via maximizing the probability.

We write the probability as Eq. (11). Step 1 is based on the independence assumption, and step 2 is based on our influence model.

$$\begin{aligned} & P(F, T|\theta_U, \mathcal{L}_U, \theta_V, \mathcal{L}_V) \\ =^1 & \prod_{f\langle u_i, u_j\rangle \in F} P(f\langle u_i, u_j\rangle|\theta_{u_j}, \mathcal{L}_{u_i}) \prod_{t\langle u_i, v_j\rangle \in T} p(t\langle u_i, v_j\rangle|\theta_{v_j}, \mathcal{L}_{u_i})^{w_{ij}} \\ =^2 & \prod_{f\langle u_i, u_j\rangle \in F} \frac{1}{2\pi\sigma^2_{u_i}} e^{\frac{(X_{u_i}-X_{u_j})^2+(Y_{u_i}-Y_{u_j})^2}{-2\sigma^2_{u_i}}} \\ \times & \prod_{t\langle u_i, v_j\rangle \in T} (\frac{1}{2\pi\sigma^2_{v_j}} e^{\frac{(X_{u_i}-X_{v_j})^2+(Y_{u_i}-Y_{v_j})^2}{-2\sigma^2_{v_j}}+})^{w_{ij}} \end{aligned} \tag{11}$$

In the above equation, for $u_i \in U^N$, both $\mathcal{L}_{u_i}$ and σ_{u_i} are unknown; for $u_i \in U^*$ and $v_j \in V$, σ_{u_i} and σ_{v_j} are unknown. We estimate their values by maximizing the probability. To derive them, we first differentiate Eq. (11) with regard to every unknown variable, and obtain the following equations.

$$X_{u_i} = \frac{\sum_{u_j \in \mathcal{I}_f(u_i)} \frac{X_{u_j}}{\sigma^2_{u_i}} + \sum_{u_j \in \mathcal{O}_f(u_i)} \frac{X_{u_j}}{\sigma^2_{u_j}} + \sum_{v_j \in \mathcal{O}_t(u_i)} \frac{w_{ij} X_{v_j}}{\sigma^2_{v_j}}}{\sum_{u_j \in \mathcal{I}_f(u_i)} 1/\sigma^2_{u_i} + \sum_{u_j \in \mathcal{O}_f(u_i)} 1/\sigma^2_{u_j} + \sum_{v_j \in \mathcal{O}_t(u_i)} w_{ij}/\sigma^2_{v_j}} \tag{12}$$

$$\sigma^2_{u_i} = \sum_{u_j \in \mathcal{I}_f(u_i)} \frac{(X_{u_j} - X_{u_i})^2 + (Y_{u_j} - Y_{u_i})^2}{2|\mathcal{I}_f(u_i)|} \tag{13}$$

$$\sigma^2_{v_j} = \sum_{u_i \in \mathcal{I}_t(v_j)} \frac{w_{ij}((X_{u_i} - X_{v_j})^2 + (Y_{u_i} - Y_{v_j})^2)}{2\sum_{u_i \in \mathcal{I}_t(v_j)} w_{ij}}. \tag{14}$$

In these equations, the unknown variables are dependent on each other. Their closed-form solutions are not easy to get. However, if we assume σ_{u_i} and σ_{v_j} for each $u_i \in U$ and each $v_j \in V$ are known, X_{u_i} only depends on $X_{u_j} \in U$ and $X_{v_j} \in V$. In this case, Eq. (12) tries to find X_{u_i} for each $u_i \in U^N$ such that $\sum_{f\langle u_i, u_j\rangle \in F} 1/\sigma^2_{u_j}(X_{u_i} - X_{u_j})^2 + \sum_{t\langle u_i, v_j\rangle \in T} w_{ij}/\sigma^2_{v_j}(X_{u_i} - X_{v_j})^2$ is minimized. An iterative algorithm, which updates each X_{u_i} based on other X_{u_j} iteratively, has been proposed to find X_{u_i} for this problem [20]. When X_{u_i} and Y_{u_i} are derived, σ_{u_i} and σ_{v_j} can be derived directly based on Eq. (13) and (14).

Therefore, we develop a two stage iterative algorithm based on the above intuition. The algorithm is shown in Algorithm 1. At step 1-2, it initializes all $u_i \in U^N$. At step 3-14, the algorithm does the iterative computation. There are two iterations. The outer iteration updates σ_{u_i} and σ_{v_j} according to $\mathcal{L}_{u_i}$ based on Eq. (13) and (14), while the inner iteration (from step 8 to 11) takes a set of fixed σ_{u_i} and σ_{v_j} as inputs and iteratively computes $\mathcal{L}_{u_i}$ based on Eq. (12). The newly obtained $\mathcal{L}_{u_i}$ is then used to update σ_{u_i} and σ_{v_j} again. The algorithm stops until the likelihood converges.

Algorithm 1: Global Prediction Algorithm
Input: G, $\mathcal{L}_{u_i} \forall u_i \in U^*$
Output: $\mathcal{L}_{u_i}, \forall u_i \in U^N$
// Initialization
1 **foreach** $u_i \in U^N$
2 $\quad X_{u_i} = Random$ and $Y_{u_i} = Random$
3 **repeat** //Outer Iteration
4 $\quad$ **foreach** $u_i \in U$
5 $\quad\quad$ update $\sigma^2_{u_i}$ based on Eq. (13)
6 $\quad$ **foreach** $v_j \in V$
7 $\quad\quad$ update $\sigma^2_{v_j}$ based on Eq. (14)
8 $\quad$ **repeat** // Inner Iteration
9 $\quad\quad$ **for** $u_i \in U^N$
10 $\quad\quad\quad$ update $X^{n+1}_{u_i}$ and $Y^{n+1}_{u_i}$ based on Eq. (12)
11 $\quad$ **until** converge
12 $\quad$ **foreach** $u_i \in U^N$
13 $\quad\quad X_{u_i} = X^{n+1}_{u_i}, Y_{u_i} = Y^{n+1}_{u_i}$
14 **until** converge

We can formally prove the convergence of the algorithm based on the following theorem.

Theorem *The global prediction algorithm converges.*

The proof of the theorem is derived based on the intuition of the algorithm stated above. In the inner iteration, the method can converge and yield $\mathcal{L}_{u_i}$ that maximizes the probability with fixed σ_{u_i} and σ_{v_j}, as shown in [20]. Second, the outer iteration directly computes σ_{u_i} and σ_{v_j} that maximize the probability given fixed $\mathcal{L}_{u_i}$ computed in the previous iteration, because Eq. (13) and (14) are the closed-form solutions for maximizing the probability when a set of $\mathcal{L}_{u_i}$ is given. In summary, each iterative step monotonically increases the probability and the probability has a maximum value, so the algorithm must converge.

The above algorithm, like many of other iterative algorithms (*e.g.*, EM), may converge to a local maximum. To avoid that, we can initialize the unknown variables with the values obtained from the local prediction method. The above iterative algorithm will always generate a better solution than the local one as each iteration improves the likelihood monotonically.

Complexity Analysis As each inner iteration requires $O(|E|)$ to update every user's location, the algorithm runs in $O(t|E|)$, where t is the number of iterations and $|E|$ is the number of edges of the graph. In our experiment, it converges after three outer iterations. As our algorithm uses all the edges in the graph, it can be viewed as an offline algorithm, which effectively profiles all users' locations.

5.3 Incorporating Constraints

To further improve our methods, we utilize human knowledge as constraints in our prediction methods. To motivate, let us revisit the example in Fig. 1. Most of u_1's followers and friends are in or close to Chicago (*e.g.*, u_5, u_3) except one (u_4) in New York. Our algorithms will estimate u_1's location to be near but not exactly Chicago. If we ask a human to predict u_1's location, he will definitely pick a city instead of an arbitrary geo point, and he is likely to choose one from Chicago, Urbana and New York, because he knows a user usually has some friends living in the same city.

We model such human knowledge as *constraints* in our prediction methods. A *constraint* specifies the set of candidate locations when we maximize a likelihood function. There are different choices of constraints, such as a candidate must be a city or within 30 miles of a city. Particularly, we apply the following assumption as the constraint in our implementation. We assume that a user's location must be the same as one of his friends, followers or tweeted venues. The assumption is generally valid. In our data, an incomplete crawl of Twitter, there are about 92% of users whose locations appear in their followers, friends or tweets. We note that this constraint may not be the best one. We use it to illustrate how our methods can incorporate constraints.

The constraint version of the local prediction method becomes maximizing Eq. (4) subject to $\{\mathcal{L}_{u_i} \in \mathcal{L}_{\mathcal{I}^*_f(u_i)} \cup \mathcal{L}_{\mathcal{O}^*_f(u_i)} \cup \mathcal{L}_{\mathcal{O}_t(u_i)}\}$. To solve it, we can rank each candidate location $\mathcal{L}_{u_i}$ according to Eq. (4), and use the top one as the prediction.

The constraint version of the global prediction method becomes maximizing Eq. (11) subject to $\{\mathcal{L}_{u_i} \in \mathcal{L}_{\mathcal{I}^*_f(u_i)} \cup \mathcal{L}_{\mathcal{O}^*_f(u_i)} \cup \mathcal{L}_{\mathcal{O}_t(u_i)}\}$ for any $u_i \in U^N$. If we rank all candidate solutions, which consist of all the combinations $\mathcal{L}_{u_i}$ for all $u_i \in U^N$, the complexity of the algorithm is $O(K^N)$, where K is the average number of candidate locations per user (it is usually larger than 2), and N is the number of unlabeled users (about millions). Instead, we propose an approximation algorithm based on the relax and round paradigm, which is widely used by approximation algorithms for optimization with constraints [9]. We first use the global algorithm to find $\mathcal{L}'_{u_i}$ for each u_i without any constraint, then find the closest location $\mathcal{L}_{u_i}$ that satisfies the constraint.

6. EXPERIMENTAL RESULTS

In this section, we conduct experiments on a large-scale data set and show the effectiveness of our methods from different aspects.

6.1 Experiment Setup

Data Set We constructed our data set by crawling Twitter. We randomly selected 100,000 users as seeds to crawl in May 2011. For each user, we crawled his profile, followers and friends. We obtained 3,980,061 users' profiles and their social network. Then, we extracted their registered locations from their profiles based on the rules described in [7]. Specifically, we extracted locations with city-level labels in the form of "cityName, stateName" and "cityName, stateAbbreviation," where we considered all cities listed in the Census 2000 U.S. Gazetteer. We found 630,187 users, who provided city level locations, and treated them as labeled users. Among them, we found 158,220 users, who had at least one labeled friend or follower. We further crawled their tweets and extracted venues from those tweets based on the same gazetteer. We crawled at most 600 tweets for each user. As we could not get some users' tweets due to their privacy settings or lack of tweets, only 139,180 users' tweets were crawled.

We used the 139,180 users with their following relationships and tweets, as our data set. There are 14.8 friends, 14.9 followers, and 29.0 venues per user. We took their registered locations as their home locations, and applied five fold validation, which means that we used 80% of users as labeled users and 20% of users as unlabeled users and reported our results based on the average of 5 runs.

We note that we directly take users' registered locations as their home locations and predict locations for only U.S. users, because we want to set up our experiments in the same way as the existing methods [7, 4]. We are aware that some registered locations are incorrect, but we believe they are rare, as leaving profiles empty is always an easy option. Thus, our results are reliable. Our method can predict locations for international users in the same way.

Methods To fully evaluate our methods, we not only compare them with two state-of-the-art methods in [4] and [7], but also evaluate our prediction methods with different settings. Specifically, our experiments evaluate the following methods.

- $Base_U$ is the method developed in [4], which predicts a user's location based on his social network. Twitter is a directional network, so we treat both followers and friends of a user as his undirected connections ("friends") in this method.
- $Base_C$ is the method developed in [7]. It assigns a location to a user based on a set of local words identified from his tweets.
- UDI$_U$ is our local prediction method, but only uses a user's friends and followers as signals.
- UDI$_C$ is our local prediction method, but only uses venues identified from a user's tweets as signals.
- UDI$_I$ is our local prediction method discussed in Sec. 5.1, which integrates different types of resources.
- UDI$_G$ is our global prediction method discussed in Sec. 5.2.

Measurement We use *average error distance in miles* (*AED*) and *accuracy within 100 miles* (*ACC*) proposed in [7] as measures. Specifically, let $Err(u_i)$ be the error distance between a user's home location and an estimated location. For a set of users U, $AED(U)$ is $\frac{\sum_{u_i \in U} Err(u_i)}{|U|}$, and $ACC(U)$ is $\frac{|\{u_i | u_i \in U \wedge Err(u_i) \leq 100\}|}{|U|}$.

However, as *AED* is easily affected by outliers in results, we report *AED* at different percentiles (60%, 80% and 100%) of users ranked by their error distances. *E.g.*, *AED*@60% is the average error distance of the top 60% of users ranked by their error distances.

We use T-test to conduct *significance tests* between our methods and baseline methods. If a method passes the significant test, we make it **boldface** in result tables.

6.2 Experiment Results

User-based Prediction We first compare *UDI$_U$* with $Base_U$. Both of them profile a user's location based on his social network.

Tab. 1 shows the performance of each method. The results demonstrate that generally our method performs better than $Base_U$. When using the same amount of information, *UDI$_U$* improves $Base_U$ by 4% in terms of *ACC*. Such an improvement soundly proves our assumption that different users have different influence scopes and we should model them discriminatively.

AED@60% tells that the average error distance of the top 60% of predictions of *UDI$_U$* is 20 miles, which is fairly accurate. However, when comparing *AED@80%* and *AED@100%*, we find that *AED* dramatically increases from 159 to 525, because *AED* is easily affected by a small set of users, who are not accurately predicted. Therefore, we should not only focus on *AED@100%*.

To illustrate our results in detail, we plot an *accumulative accuracy at distances*(*AAD*) curve for each method in Fig. 4(a). A point (X, Y) in the curve means that Y percentages of users are accurate within X miles. From the figure, we can tell that *UDI$_U$* has higher accuracy than $Base_U$ within different distances. *E.g.*, *UDI$_U$* places about 47% of users within 25 miles, while $Base_U$ only places 44% of users within that range.

Content-based Prediction In this experiment, we compare *UDI$_C$* with $Base_C$. Both of them profile a user's location with his tweets.

We show results and *AAD* curves of two methods in Tab. 1 and Fig. 4(b) respectively. From them, we can see that 1) *UDI$_C$* significantly improves $Base_C$ by 10% in terms of *ACC*, 2) the improvement is consistent at any distance level, and 3) *UDI$_C$* achieves very good results by making good use of content. The average error distance for the top 60% of its prediction is less than 10 miles. From

Table 1: Prediction Results

Model	$Base_U$	$Base_C$	UDI_U	UDI_C	UDI_I	UDI_G
ACC	52.4%	49.7%	**56.0%**	**60.0%**	**64.4%**	**65.9%**
AED@60%	33.7	21.8	**20.6**	**9.5**	**6.6**	**4.4**
AED@80%	200.0	161.5	**159.6**	**123.6**	**97.0**	**75.0**
AED@100%	616.9	542.5	**524.5**	**483.6**	**440.4**	**421.3**

Table 2: Discriminative vs. Non-discriminative

Model	ACC	AED@60	AED@80	AED@100
$Base_I$	58.5%	11.9	138.4	504.4
UDI$_I$	**64.4%**	**6.6**	**97.0**	**440.4**

the results, we can safely conclude that our method is much better than $Base_C$ as our model captures the relation between a user' location and locations from his tweets in a meaningful way.

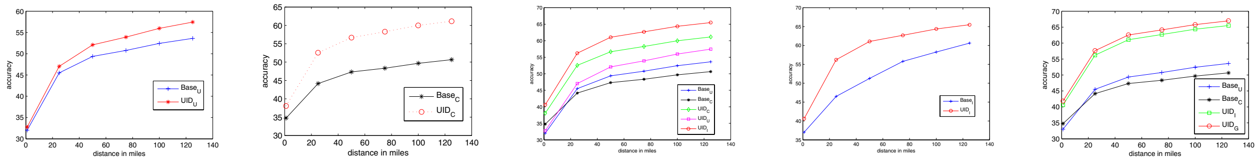
We clarify that $Base_C$ requires human labeling to train a model to select local words, which are the features for the classification model, and $Base_C$'s performance highly depends on the selected words. As labeling is a subjective task, by no means could we get the same set of local words in the original paper. We test performances of $Base_C$ with various local word sets, we get *ACC* ranging from 35.98% to 49.67%. We choose the highest one to report. Our method advances $Base_C$ in this aspect, as we do not require any labeling work, and only use location names in a gazetteer.

Integrated vs. Non-Integrated In this experiment, we evaluate whether our framework can take advantage of integrating more resources. Specifically, we compare *UDI$_I$* with $Base_U$, $Base_C$, *UDI$_C$* and *UDI$_U$*. Tab. 1 shows the performance of each method. As expected, *UDI$_I$* gives a significant improvement (12%) over the best baseline method, and advances *UDI$_C$* and *UDI$_U$* by 4.4% and 8.4%. Fig. 4(c) shows that those improvements are consistent at any distance level. We can safely conclude that integrating different types of resources is useful for profiling locations. Meanwhile, we can find that *UDI$_I$* is very accurate. It correctly places 57% of users within 25 miles. Its *AED* is only 6 miles for the top 60% of its predictions, and less than 100 miles for the top 80%.

Discriminative vs. Non-discriminative In this experiment, we demonstrate the power of discriminative modeling by comparing our methods based on a discriminative model with the methods based on a non-discriminative one. As the user-based prediction experiment has already shown that, when only using social network signals, a discriminative method (*UDI$_U$*) is better than a non-discriminative one ($Base_U$), we now compare the methods that use all the types of resources. We develop a new baseline method $Base_I$, which integrates different resources in a non-discriminative way. Specifically, we first learn one probabilistic distribution for following edges and one for tweeting edges based on [4], and then we apply the prediction method in [4]. Tab. 2 and Fig. 4(d) show the results. We can find that, although $Base_I$ uses the same amount of information as *UDI$_I$*, it is 6% lower than *UDI$_I$* in accuracy, which suggests that we should model observations discriminatively.

Global vs. Local To investigate the usefulness of our global prediction method, we compare *UDI$_G$* with *UDI$_I$* and the two baselines.

We first evaluate the methods on the data set used in the previous experiments, which includes 20% unlabeled users and 80% labeled ones. The last column in Tab. 1 gives the results of *UDI$_G$*. We can see that, although *UDI$_G$* improves *UDI$_I$* slightly (1.5%) in terms of *ACC*, it reduces *AED@80%* a lot, and Fig. 4(e) shows that the improvement is consistent at any level. We believe that the improvement here is limited because there is already enough information from the labeled users and the iterative based method can not add much help. We expect that *UDI$_G$* improves *UDI$_I$* significantly in a more realistic scenario, where less users are labeled.



(a) User Results (b) Content Results (c) Integrated Results (d) Discriminative Results (e) Global Results

Figure 4: Accumulative Accuracy at Various Distance

Table 3: Local vs. Global with 80% Test Users

Model	$Base_U$	$Base_C$	UDI_I	UDI_G
ACC	34.0%	42.4%	**57.0%**	**66.0%**
AED@60	116.9	60.9	**11.7**	**4.3**
AED@80	347.7	259.3	**133.9**	**71.6**
AED@100	897.4	679.9	**514.1**	**415.3**

To test this conjecture, we evaluate those methods in another data set, where only 20% of users are labeled and 80% users are unlabeled. This scenario is more close to the real-world case, where only about 16% users have registered locations. Tab. 3 shows the results. We find that 1) UDI_G significantly outperforms the other three methods, as it can utilize information from even unlabeled users, 2) compared to the preceding experiment, UDI_G achieves nearly comparable results, but the other three methods perform much worse, as they make predictions with limited amount of information. We can conclude that UDI_G utilizes both labeled and unlabeled information and achieves better profiling.

We evaluate UDI_G for its convergence, and find it takes 3 outer iterations to converge. Due to space limit, the figure is omitted.

Table 4: Case Studies

Users	Follower No.	σ	Cities	σ
MythBusters Official	860688	1.127	Honolulu	0.970
Lady Gaga	18428360	0.633	San Francisco	0.582
National Geographic	162870	0.655	New York	0.551
NY Knicks	178297	0.172	Austin	0.11
Philadelphia 76ers	62210	0.161	Houston	0.12
timpawlenty	63896	0.239	Dallas	0.14

Case Studies for Influence Scope We give some concrete examples of influence scopes derived by our methods to illustrate their correctness and usefulness. Tab. 4 shows influence scopes of some Twitter users and venues. For easy understanding, we only choose verified users (celebrities). In Tab. 4, we can clearly distinguish local authorities (*e.g.*, "timpawlenty", a former governor of Minnesota), and national celebrities (*e.g.*, "Lady Gaga"). We note that we can not easily tell the difference between "national graphic" and "NY Knicks" just by the numbers of their followers. Similarly, our methods identify that Honolulu, a famous vacation destination, has a broad influence scope and is likely to be a noisy signal.

7. CONCLUSION

Profiling users' locations is an important problem. In this paper, we have made the following contributions to this problem. 1) We explore both social network and user-centric data for profiling users' locations. 2) We introduce a unified discriminative influence model ($\mathcal{UDI}$), which captures how likely a user follows a user or tweets a venue. 3) We develop two effective location prediction methods. The local method integrates locations observed from his friends, followers and tweets in a discriminative way and profiles users' locations efficiently. The global method extends the local one by using additional unlabeled users, and profiles users' locations more accurately. 4) We extend the two methods by modeling additional human knowledge as constraints. 5) We conduct comprehensive experiments on a large scale data set and demonstrate the effectiveness of our methods.

8. REFERENCES

[1] A. Ahmed, Y. Low, M. Aly, V. Josifovski, and A. J. Smola. Scalable distributed inference of dynamic user interests for behavioral targeting. In *KDD '11*, pages 114–122, 2011.

[2] E. Amitay, N. Har'El, R. Sivan, and A. Soffer. Web-a-where: geotagging web content. In *SIGIR '04*, pages 273–280, 2004.

[3] L. Backstrom, J. Kleinberg, R. Kumar, and J. Novak. Spatial variation in search engine queries. In *WWW '08*, pages 357–366, 2008.

[4] L. Backstrom, E. Sun, and C. Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *WWW '10*, pages 61–70, 2010.

[5] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: quantifying influence on twitter. In *WSDM '11*, pages 65–74, 2011.

[6] S. Chandra, L. Khan, and F. Muhaya. Estimating twitter user location using social interactions–a content based approach. In *2011 ieee third international conference on social computing*, pages 838 –843, 2011.

[7] Z. Cheng, J. Caverlee, and K. Lee. You are where you tweet: a content-based approach to geo-locating twitter users. In *CIKM '10*, pages 759–768, 2010.

[8] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *WWW '09*, pages 761–770, 2009.

[9] T. F. Gonzalez. *Handbook of Approximation Algorithms and Metaheuristics (Chapman & Hall/Crc Computer & Information Science Series)*. Chapman & Hall/CRC, 2007.

[10] D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins. *Proceedings of the National Academy of Sciences of the United States of America*, (33), Aug 2005.

[11] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel. You are who you know: inferring user profiles in online social networks. In *WSDM '10*, pages 251–260, 2010.

[12] F. Provost, B. Dalessandro, R. Hook, X. Zhang, and A. Murray. Audience selection for on-line brand advertising: privacy-friendly social network targeting. In *KDD '09*, pages 707–716, 2009.

[13] F. Qiu and J. Cho. Automatic identification of user interest for personalized search. In *WWW '06*, pages 727–736, 2006.

[14] T. Rattenbury, N. Good, and M. Naaman. Towards automatic extraction of event and place semantics from flickr tags. In *SIGIR '07*, pages 103–110, 2007.

[15] S. Sizov. Geofolk: latent spatial semantics in web 2.0 social media. In *WSDM '10*, pages 281–290, 2010.

[16] R. W. White, P. Bailey, and L. Chen. Predicting user interests from contextual information. In *SIGIR '09*, pages 363–370, 2009.

[17] S. Xu, S. Bao, B. Fei, Z. Su, and Y. Yu. Exploring folksonomy for personalized search. In *SIGIR '08*, pages 155–162, 2008.

[18] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. In *WWW '11*, pages 537–546, 2011.

[19] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Geographical topic discovery and comparison. In *WWW '11*, pages 247–256, 2011.

[20] Z. Yin, R. Li, Q. Mei, and J. Han. Exploring social tagging graph for web object classification. In *KDD '09*, pages 957–966, 2009.