Author(s): Eric Malmi, Arno Solin, and Aristides Gionis

Title: The blind leading the blind: Network-based location estimation under uncertainty

Year: 2015

Version: Author accepted / Post print version

Please cite the original version:

Eric Malmi, Arno Solin, and Aristides Gionis. The blind leading the blind: Network-based location estimation under uncertainty. In Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2015), Porto, Portugal, pages 406-421, 2015. DOI: 10.1007/978-3-319-23525-7_25

This publication is included in the electronic version of the article dissertation:
Malmi, Eric. Collective Entity Resolution Methods for Network Inference.
Aalto University publication series DOCTORAL DISSERTATIONS, 73/2018.

The blind leading the blind: Network-based location estimation under uncertainty

Eric Malmi, Arno Solin, and Aristides Gionis

Helsinki Institute for Information Technology, and
Department of Computer Science
Aalto University, Finland
{firstname.lastname}@aalto.fi

Abstract. We propose a probabilistic method for inferring the geographical locations of linked objects, such as users in a social network. Unlike existing methods, our model does not assume that the exact locations of any subset of the linked objects, like neighbors in a social network, are known. The method efficiently leverages prior knowledge on the locations, resulting in high geolocation accuracies even if none of the locations are initially known. Experiments are conducted for three scenarios: geolocating users of a location-based social network, geotagging historical church records, and geotagging Flickr photos. In each experiment, the proposed method outperforms two state-of-the-art network-based methods. Furthermore, the last experiment shows that the method can be employed not only to network-based but also to content-based location estimation.

1 Introduction

Observations recorded as data are typically associated with a location. Similarly, data attributes are often spatially correlated. For example, consider friendship relations in a social network that correlate with the geographical distances between friends, or business types that cluster in different parts of a city, such as restaurants and cafés being more concentrated in touristic areas.

On the other hand, there is a plethora of available datasets that lack explicit location information, even though the data objects they contain are inherently associated with a location (or a distribution of locations). For instance, consider online social networks where only a small fraction of the users provide their location explicitly. As a second example, motivated by the domain of historical research and social sciences, consider historical documents, such as letters or public registry records. Such documents contain many pieces of valuable information, but are often not accurately geolocated, either because their authors assumed that the location is implicit, or because there is a reference to an uncertain location, as the location of an old village can be uncertain.

As can be easily motivated from the previous examples, identifying the location of data objects is an important problem, and has compelling applications.

For example, as pointed out by Backstrom et al. [2], locating the users of an online social network can be used to improve the network security by detecting "phishing" attempts, or to improve the user experience by offering personalized functionalities. Similarly, locating place names or whereabouts of people mentioned in historical documents is an extremely valuable tool for research in history or other social sciences. Yet another domain where geolocation can provide vital insights is the field of forensics where it has been used to pinpoint serial offenders [9].

In this paper, we propose a new method for inferring the geographical location of linked objects. In a nutshell, our method can be described as follows. We consider a set of objects for which certain attributes are known but the location information is missing. We assume that the known attributes can be used to provide two types of additional information:

1. *A prior distribution of each data object over a set of candidate locations.* For instance, in the social-network scenario, known friend locations can be taken as candidate locations [2]. Or if we know the city where the object is located, we can simply define a grid over the city and impose a uniform prior over the grid cells, as done in the Flickr experiment in this paper.
2. *Links between data objects.* In the case of social networks, friendship relations between users are available, indicating that friends are more likely to be located in nearby locations. Similarly, in the photo-location application, two photos taken by the same user within a short time interval are more likely to be located in nearby locations.

Our method follows a probabilistic inference approach and gives predictions for the locations of the objects in the dataset by taking into account the prior distribution over locations and the links between objects.

The most closely-related work to our paper is the study of Backstrom et al. [2], who propose a probabilistic model for inferring the locations of Facebook users, given the location of their friends. However, our method extends and improves this prior work in the following ways:

- Our method does not assume that the exact locations for any of the linked data objects, like neighbors in a social network, are known. Instead we impose a prior distribution over these locations. This generalization makes the method very well suited to cope with the uncertainty that is present in most datasets. The case that the location of some objects is known, can also be naturally incorporated in our model.
- The proposed model offers a general abstraction that can be used to infer the locations of any kind of linked data with spatial dependencies. We demonstrate the generality of the model by applying it to three application scenarios: (*i*) geolocating users in a social network; (*ii*) geotagging historical church records from the 1600s to 1800s; and (*iii*) geotagging Flickr photos.
- The last experiment regarding Flickr photos shows that the proposed method can be adapted to content-based analysis, even though it is primarily designed for network-based geolocation.

Even though many relevant problems naturally fall under this problem setup as is demonstrated in the experiments, to our knowledge there is a lack of methods that would attempt to perform location estimation based on linked items whose locations are not known exactly.

The rest of the paper is organized as follows. In Section 2 we give an overview of previous related work. Our method is presented in Section 3, where we formalize the abstract geolocation problem and present the probabilistic algorithm for solving it. The three experiments discussed above are presented in detail in Section 4. Section 5 contains a final discussion and suggestions for future research directions.

2 Related work

With the abundance of data gathered from all kinds of human activity, and the wide spread of social media applications that support collection of large amounts of user-generated content, problems related to geolocating various types of data have gained importance. As a result, a large number of related papers have appeared in machine learning, data mining, and web science venues. These pieces of work can be roughly categorized under network-based and content-based methods [10].

Network-based geolocation. In network-based methods, only the network structure and the location information about the other nodes are used for geolocation. Examples of this type of approach are the methods proposed by Backstrom et al. [2] and Jurgens [5]. These two methods will be further described in Section 3.4.

Rout et al. [10] approach the user geolocation problem as a classification problem and apply an SVM classifier. The classification is done on a city-level, and for each city a number of features, including the number of friends in the city and its total population, are extracted. The performance on the city prediction problem is better than the performance by the Backstrom et al. [2] method, but it is not clear how the classification approach scales down if we need to predict more fine-grained locations since the number of classes and sparsity of the data would both increase.

McGee et al. [8] build on the work of Backstrom et al., and they study the effect of incorporating information about tie strengths in the geolocation model. This line of thinking is complementary to our method. As we discuss later, our approach supports having different edge types, which can be learned separately. Nevertheless, our focus is in the general network-based geolocation problem, whereas many of the features used for inferring the tie strength in McGee et al. are Twitter specific, like the number followers and mentions in Twitter.

Sadilek et al. [11] propose a probabilistic method for location prediction based on dynamic Bayesian networks. This method is shown to provide high accuracy estimates, but the problem setting is more specific than ours. They assume that a time series of friend locations is provided as an input for the system.

Content-based geolocation. A different approach to the problem of geolocating users in social networks is to perform a more detailed analysis on the content generated by users. Most of the content-based methods from the recent years have focused on geolocating Twitter users [1,6,7,13].

The focus of our work is in network-based geolocation, but in our last experiment, we show that the proposed method can be used for content-based estimation as well. In that experiment, we aim to geotag Flickr photos based on textual annotations (tags). We show that the solution obtained using our framework corresponds to the method proposed by Serdyukov et al. [12]. Additionally, we present the idea of linking consecutive photos of a user in order to estimate their locations jointly and show that it slightly improves the geotagging accuracy. Another approach for geotagging Flickr photos has been proposed by Crandall et al. [4]. Their method also uses tags, but additionally they extract visual features (SIFT descriptors) from the photos. The photos are then geolocated using the resulting distribution over the joint feature space. However, the idea of estimating the locations of consecutive photos jointly is not explored in either of these works.

3 Methods

In this section we describe the general setting for the geolocation problem. As mentioned in the introduction, our model offers a unified framework that can fit various, seemingly very different types of geolocation problems. We present our solution in two steps. First we derive an exact solution for the general problem in the case of geolocating a single object. Then we extend to the multiple object case and show how to obtain an approximate solution. Later we show how the specific application scenarios fit under the general model.

3.1 Problem setting

We consider a set V of items whose locations we want to find out. We assume that relations between items have been observed and represented by a set of edges E. Thus, the data items form a graph $G = (V, E)$. The neighbors of an item u in the graph G are denoted by $N(u) = \{v \mid \{u, v\} \in E\}$.

We also consider a discrete set of locations $\mathcal{L}$ which are the candidate locations to place the items in V. A distance function $d : \mathcal{L} \times \mathcal{L} \to \mathbb{R}$ is defined between locations, such that $d(\ell_1, \ell_2)$ denotes the distance between locations $\ell_1, \ell_2 \in \mathcal{L}$. In this paper, $d(\cdot, \cdot)$ is considered to be a geodetic distance, except for the photo geotagging case study where employ use the Manhattan (city block) distance since we consider the center of New York City. For each item $u \in V$ we write $\ell(u) \in \mathcal{L}$ to denote the location where to u is mapped. The mapping of all the items in V to locations in $\mathcal{L}$ is denoted by (boldface) vector $\boldsymbol{\ell}$, in other words, $\boldsymbol{\ell} = \langle \ell(u) \mid u \in V \rangle$.

We model uncertainty by considering a probability distribution $\Pr[\ell(u)]$ for item u over the space of possible locations. In our problem formulation we assume

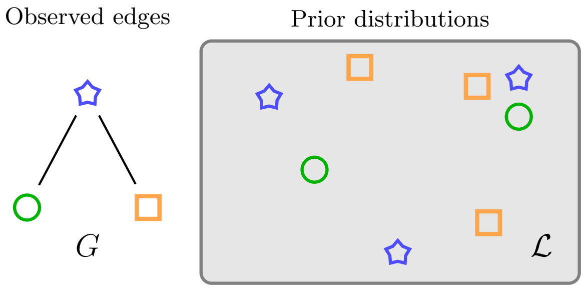


Fig. 1. A simple example of the GeoLocation problem with three items to be located, and a set of discrete candidate locations.

that initially, as part of the input, a prior distribution $\Pr[\ell(u)]$ for each item u is given. This is a fairly natural assumption for many applications as illustrated by our experiments. For item u, we denote by $\mathcal{L}(u)$ the subset of locations in $\mathcal{L}$ for which the prior distribution $\Pr[\ell(u)]$ is non-zero. In other words, $\mathcal{L}(u)$ is the set of candidate locations where to place item u. Depending on the application, $\mathcal{L}(u)$ may be a significantly smaller set than $\mathcal{L}$. In practice, we can further prune the set $\mathcal{L}(u)$ by removing locations that have very small prior probability for u. If the exact location of an item u is known then $\Pr[\ell(u)]$ is a delta distribution. If no information about an item u is known then $\Pr[\ell(u)]$ is the uniform distribution. Note that we also assume that the set of candidate locations $\mathcal{L}$ is discrete. This is the case with the first two of our case studies. In cases where the set of locations is continuous, we can discretize it on a set of grid cells, as done in our Flickr photo geolocation case study.

We consider data for which spatial dependencies are present. The existence of an edge between two items u and v is thus assumed to depend on the location of the items. In our probabilistic model setting, the locations of the items are viewed as model parameters and edges as the observed data. Each edge $\{u, v\} \in E$ is assumed to be produced by a generative process that depends on the location of the two items u and v. Given two candidate locations $\ell(u)$ and $\ell(v)$ for items u and v we write $\Pr[\{u, v\} \in E \mid \ell(u), \ell(v)]$ for the likelihood of an edge between u and v given their candidate locations. To simplify the problem we assume that an edge depends only on the distance between the two candidate locations, and we write $\Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))]$ to denote this likelihood.

In the context of social networks, the generative process would correspond to the process of people forming social ties. Even for online social networks, formed in a virtual world, distance have been shown to play an important role in the process of relationship formation [2]. We also note that we may have different kinds of edges and the likelihood of an edge may depend also on the edge type. For instance, in our third case study, where we estimate the locations of Flickr photos based on their tags and consecutive photos, we have two types of edges: “photo-to-tag” and “photo-to-photo.”

Our problem can now be defined as follows.

*Problem 1. (*GeoLocation*)* Consider a graph $G = (V, E)$ over items V, and a set of candidate locations $\mathcal{L}$. For each item $u \in V$ we are given a prior distribution $\Pr[\ell(u)]$. The goal is to infer a mapping $\boldsymbol{\ell}$ of items to locations in order to maximize the likelihood $\Pr[E \mid \boldsymbol{\ell}]$ of observing the data given the inferred locations.

In some cases, we are interested in estimating the locations for a subset of the items $U \subseteq V$. In this case, we consider that for the rest of the items $V \setminus U$ the prior distributions are kept fixed. This special case can be easily incorporated in our framework. Unless stated otherwise, we assume that $U = V$.

An illustration of a very simple instance of the geolocation problem is shown in Figure 1. In this case there are three items (depicted by three different shapes: ○, ☆, and □) and two edges, as shown in the left side of the figure. The prior distribution of each item to 2 or 3 candidate locations is assumed to be uniform and it is shown in the right. If the edge probability increases as the distance between items decreases, then the maximum-likelihood estimate will give a solution according to which the items lie in the upper-right corner of the figure.

3.2 Estimating a single location

We start by deriving the *maximum a posteriori* (MAP) estimate for the location $\ell(u)$ of a single item $u \in V$ assuming that the location distributions of the other items are kept fixed. The likelihood function of the observed edges is given by

$$\begin{aligned}
\Pr[E \mid \ell(u)] &= \sum_{\boldsymbol{\ell}_{N(u)}} \Pr[E, \boldsymbol{\ell}_{N(u)} \mid \ell(u)] \\
&= \sum_{\boldsymbol{\ell}_{N(u)}} \Pr[\boldsymbol{\ell}_{N(u)} \mid \ell(u)]\ \Pr[E \mid \ell(u), \boldsymbol{\ell}_{N(u)}] \\
&= \sum_{\boldsymbol{\ell}_{N(u)}} \prod_{v:\{u,v\}\in E} \Pr[\ell(v)]\ \Pr[\{u,v\} \in E \mid \ell(u), \ell(v)],
\end{aligned}$$

where $\boldsymbol{\ell}_{N(u)} = \{\ell(v) : \{u,v\} \in E\}$ are the locations of the neighbors of u and the summation goes over all different candidate locations of each neighbor. In the above derivation we have assumed independence for the prior probabilities of different locations and for the different edges. As already discussed, we further assume that the likelihood of an edge being present, given the locations of the adjacent vertices, $\Pr[\{u,v\} \in E \mid \ell(u), \ell(v)]$, only depends on the distance $d(\ell(u), \ell(v))$ of the locations. By reordering the terms, we then get

$$\Pr[E \mid \ell(u)] = \prod_{v:\{u,v\}\in E} \sum_{\ell(v)} \Pr[\ell(v)]\ \Pr[\{u,v\} \in E \mid d(\ell(u), \ell(v))]. \tag{1}$$

The maximum a posteriori estimate for $\ell(u)$ is then given by

$$\hat{\ell}(u) = \arg \max_{\ell(u)\in\mathcal{L}(u)} \Pr[\ell(u)]\ \Pr[E \mid \ell(u)].$$

If no information is provided for the items and the prior is set to the uniform distribution the MAP estimate corresponds to the *maximum-likelihood estimate* (MLE).

In the above estimation we assumed that the term $\Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))]$ is known. This probability function can be learned from training data containing items with known locations and some edges between them. Learning the edge probability function includes also the case that there are edges of different types. In this case the edge probability function may depend on the edge type. It should be expected that the edge probability function is a monotonically decreasing function of the distance, and indeed, this is the case in all three of our case studies. However, the model does not require monotonicity.

3.3 Estimating multiple dependent locations

We now show how to extend the method to find MAP or MLE location estimates for all items jointly. The likelihood of the edges in the graph, given all locations is given by

$$\Pr[E \mid \boldsymbol{\ell}] = \prod_{\{u,v\}\in E} \Pr[\{u, v\} \in E \mid \boldsymbol{\ell}] = \prod_{\{u,v\}\in E} \Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))].$$

This function is not maximized by simply computing the MLE for each location individually using Eq. (1). The reason is that if the distribution of $\ell(u)$ is updated, it will potentially change the MLEs of u's neighbors.

In order to get an approximate MLE, we use a simple iterative method. In each iteration, for each item u the prior $\Pr[\ell(u)]$ is recomputed using the current estimate for the locations of the neighbors of u. The recomputation of $\Pr[\ell(u)]$ is done by computing $\Pr[E \mid \ell(u) = j]$, using Eq. (1), for each candidate location $j \in \mathcal{L}(u)$ and normalizing. The method terminates when the estimates converge or when a maximum number of iterations is reached. The method is illustrated in Algorithm 1.

Note that an alternative way of getting an approximate solution for the locations of all items could be to define an inference problem for an *undirected graphical model*, where the items would correspond to the vertices of the graph, and use methods such as loopy belief propagation for estimating the locations. However, in our early experimentation we noticed that the estimation of conditional density functions $\Pr[\ell(u) \mid \ell(v)]$ used in graphical models is more challenging than the estimation of edge probabilities, since the former seems to depend more heavily on geographical characteristics, like oceans or metropolitan areas.

3.4 Baseline methods

The method closest to our approach is the algorithm proposed by Backstrom et al. [2] for the problem of determining the locations of Facebook users. They also compute a MLE for each user and then iterate the computation step with the user locations updated in a batch. However, there are two key differences

Alg. 1. Approximate maximum likelihood estimation for multiple locations.

```
Input: Graph $G = (V, E)$, prior distributions $\Pr[\ell(v)]$ for each $v \in V$, and items
        whose locations we want to estimate $U \subseteq V$.
Output: Locations for each $u \in U$.
Initialize a list of lists $T$ ;    // Stores likelihood of each candidate location of each user
for $i \leftarrow 1$ to max_iter do
    foreach $u \in U$ do
        foreach $j \in \mathcal{L}(u)$ do
            $T_{u,j} \leftarrow \Pr[E \mid \ell(u) = j]$ ;                    // Use Eq. (1)
    foreach $u \in U$ do
        $T_{u,:} \leftarrow \frac{T_{u,:}}{\sum_j T_{u,j}}$ ;                  // Normalize distribution
        $\Pr[\ell(u)] \leftarrow T_{u,:}$ ;                         // Update priors in a batch
return $\arg\max_{\ell(u)} \Pr[\ell(u)]$ for each $u \in V$;
```

compared to our method: First, they consider only friends whose location is known exactly and after each iteration the users are assigned to their most likely locations. If there are multiple almost as probable locations for a user, assigning her to a single location seems harsh, which is why we have designed our method to work with location distributions. Second, they include an additional term which is a product over all edges not being present. However, they state that this term typically plays a small role and is expensive to compute, so in our experiments we run a slightly simplified version of their algorithm by omitting the additional term. This baseline method is referred to as Backstrom*.

As another baseline, we use the method proposed by Jurgens [5]. This method is designed for cases when only a small fraction of locations is known initially, and the idea is to propagate location "labels" in the network until all users have been geolocated. The author experimented with different ways for selecting the user location $\ell(u)$ based on the known neighbor locations $\ell(v)$, and the best performance was obtained by selecting the geometric median

$$\arg\min_{\ell(u)} \sum_{\ell(v)} d(\ell(u), \ell(v)).$$

We note that this term can be rewritten as

$$\arg\max_{\ell(u)} \prod_{\ell(v)} e^{-d(\ell(u), \ell(v))},$$

which, quite interestingly, shows that the method is equivalent to Backstrom* given that

$$\Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))] \propto e^{-d(\ell(u), \ell(v))}.$$

We refer to this baseline method as Jurgens.

4 Experiments

The problem of determining the geographical location of an entity—such as a person in a social network, a tweet, a photo, or virtually any piece of information—is an integral part of many services. We present three very different types of geolocation problems which can be all represented and solved under the same general framework described in Section 3.1. Our experiments are performed on publicly available data, and on data collected via public APIs. To facilitate reproducibility of our results, we have made the software used for the experiments available at: `https://github.com/ekQ/geolocation`.

4.1 Predicting social network user home locations

In this experiment, we use a similar setup that was used in Backstrom et al. [2] and vary the fraction of users, whose locations are initially known exactly. For the remaining users, we try to find the best location from a candidate set consisting of the known or the most probable locations of the neighboring users. Then we start iterating and update the candidate sets in the beginning of each iteration. This method is compared with BACKSTROM* and JURGENS presented in Section 3.4.

Data. We use a location-based social network called Brightkite [3]. This dataset contains 58 228 users, 214 078 edges between the users, and 4 491 143 check-ins by the users. In order to estimate the ground truth locations of the users, we simply compute the median latitude and longitude of their check-ins. The users are randomly split into a training set (50%), used for fitting the models, and a testing set (50%), used for evaluating the geolocation performance. Any edges between the training and testing users are ignored.

Social network experimental results. The proposed method, which keeps track of the uncertainty in the estimates, is compared to two other recently proposed methods which assign a single location to a user instead of a distribution. If a user already initially has some friends whose exact location is known, then it is not clear that keeping track of the location distributions should improve the estimation. And even if none of the user's friends have a known location, they will eventually be assigned one if the graph is connected, as the estimates will propagate throughout the graph, enabling location estimation for each user. Nevertheless, the results in Figure 2 show that the proposed method outperforms BACKSTROM* and JURGENS. Accuracy is defined as the fraction of users geolocated within 40 km from the ground truth location and the average accuracy improvement over BACKSTROM* is 0.5 percentage points. McNemar's test confirms that the improvements are statistically significant ($p < 0.001$) up till fraction 0.6. After this point, quantifying the uncertainty does not help anymore since most of the neighbor locations are known exactly. Compared to JURGENS, the difference is more clear, the average improvement being 4.7 percentage points. Figure 3 shows a power-law fit for the term $\Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))]$ employed in MLE and BACKSTROM*.

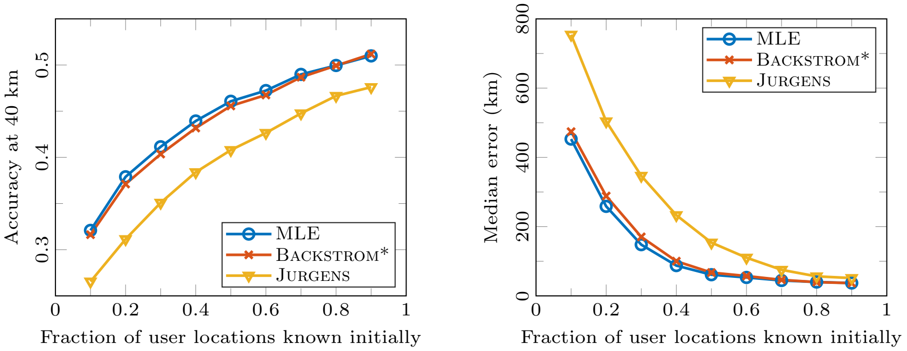


Fig. 2. Brightkite user geolocation performance with the proposed method (MLE) and two other recently proposed methods.

4.2 Geotagging historical church records

We consider a big historical dataset containing digitized church records from Finland. The digitalization from the original hand-written documents has been obtained by volunteers in the "HisKi" project.[1] This data can be considered an early population register, which was kept by the Evangelical Lutheran Church, the national church of Finland. The data contains millions of records of births, deaths, marriages and migration, spanning approximately three hundreds years from the 1600s to the late 1800s. The coverage of the church records was originally close to full, but the digitized material covers only parts of the complete dataset (some material is not digitized yet, and some is lost).

As part of the exploratory data analysis of the HisKi dataset, our interest in this paper is to attach geographical coordinates to the records in the data by geolocating the village associated with each record. This is not a trivial problem since most village names are not unique. In addition to the village name, the records contain the associated parish name, and the name of the estate/farm.

Data. The dataset contains 9 410 villages and 521 parishes in total. Village names can be matched against the Finnish geographic name database,[2] which contains coordinates for all villages in contemporary Finland, but in most cases, the match is not one-to-one since there are lots of duplicate village names. Furthermore, some of the village names have slightly changed over time, so we find the matching candidate villages by employing Damerau–Levenshtein string edit distance with a cutoff value of two.

Finding the correct village location among the candidates would not be feasible without additional information. However, we can leverage the fact that we know which villages belong to the same parish and that these villages are likely

[1] The Genealogical Society of Finland has an online interface to the HisKi data: `http://hiski.genealogia.fi/hiski?en`

[2] Open data provided by the National Land Survey of Finland: `http://www.maanmittauslaitos.fi/en/digituotteet/geographic-names`

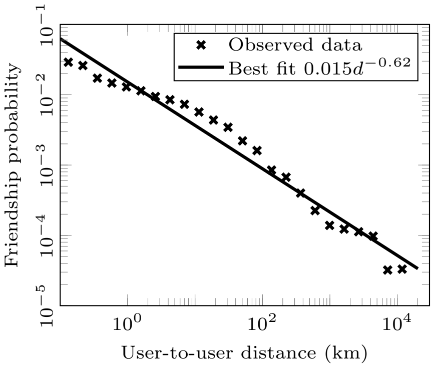


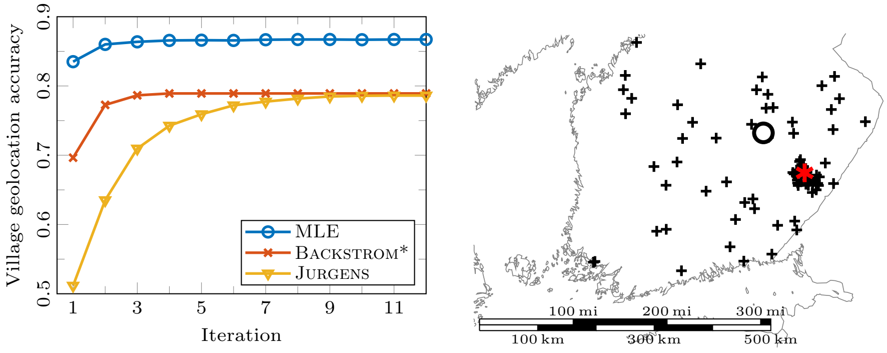
Fig. 3. Probability of two Brightkite users being friends given their distance and the power law fit to the data.

to be located nearby. This insight can be captured under the proposed framework by representing the village data as a graph, containing a node for each village and an edge between each pair of villages belonging to the same parish. The prior distribution for a village node is given by the locations of the matching villages in the geographic name database.

In this experiment, we consider only the 427 parishes with a known location and their 4 574 member villages whose names have at least two matches in the geographic name database. The average number of matches is 7.2. The ground truth location of a village is defined as the candidate location nearest to the associated parish. The parishes are split into training (30%) and testing (70%) parishes.

Village geolocation results. The locations of all village objects are initially unknown. The proposed method can readily model this uncertainty but Backstrom* and Jurgens rely on the assumption that at least a part of the locations are initially known. However, we can also apply the latter two methods by first assigning each village randomly to a candidate location and then running the methods. Since some of these initial guesses will be correct, the algorithms might be able to gradually find more and more correct locations. This is indeed what happens as is shown in Figure 4a. The term $\Pr[\{u,v\} \in E \mid d(\ell(u), \ell(v))]$ is learned from the data for the proposed method and Backstrom*, whereas Jurgens inherently assumes that it follows an exponential distribution. Hence Backstrom* obtains a higher accuracy in the first iterations than Jurgens but they both stagnate to the same accuracy of 79%. The proposed method clearly outperforms these by achieving an 87% accuracy, which is remarkable given that none of the locations are initially known and randomly assigning the villages to candidate locations would yield an accuracy of only 25%.

Geolocating parishes. In addition to villages, we can use the proposed method for geolocating the parishes whose coordinates are not recorded in the HisKi database. One way of achieving this, is to build a graph where each village is



(a) Performance

(b) The "Sääminki" parish

Fig. 4. (a) Fraction of correctly geolocated villages considering the 4 574 villages with at least two village name matches. **(b)** Location of a historical parish called "Sääminki" as identified by our method (✱) and the location of an island called Sääminki in contemporary Finland (○). Candidate village locations are shown by black crosses.

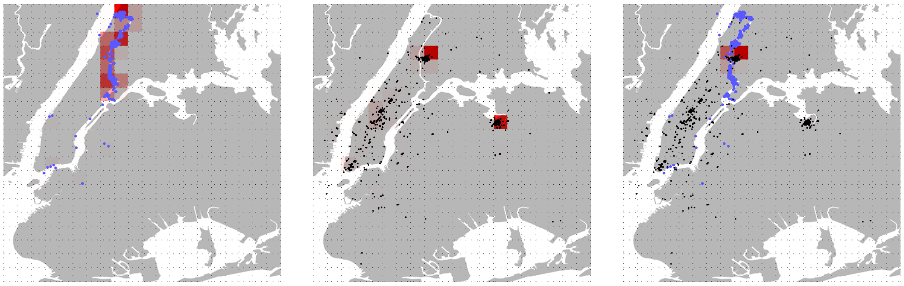
linked only to its parish. The HisKi database contains also information about neighboring parishes, so we can additionally draw edges between parishes.

Due to lack of space, we do not present the full results of this experiment here, but instead we provide some anecdotal evidence that the locations found by the method are useful. Let us study a parish called *Sääminki* for which there is only one match in the geographic name database. However, it turns out that this match is 200 km away from the location found by the algorithm. By searching for some background information on this parish, we find out that there used to be a municipality called Sääminki but nowadays it belongs to the city of Savonlinna. However, 200 km away from Savonlinna, there is still an island called Sääminki, where we might erroneously locate the parish if we do not consider the information about the member villages. The member village candidate locations and the two different Sääminki locations are visualized in Figure 4b.

4.3 Geotagging Flickr photos

Finally, we apply the proposed method to a different application domain and show that it can be employed, not only to network-based, but also to content-based geolocation. The specific task we aim to accomplish is to estimate the locations of photos uploaded in the Flickr photo-sharing system.[3] Our input consists of a set of photos specified by their IDs, timestamps, and user-provided tags for each photo. We present a mapping of this problem to our general framework, and find out that the obtained maximum likelihood solution corresponds to a method proposed by Serdyukov et al. [12], showing that their method is a special case of our framework. Additionally, the framework allows us to link con-

[3] `http://www.flickr.com/`



(a) Tag ‘harlemriver’ **(b)** Tag ‘stadium’ **(c)** Consecutive tags

Fig. 5. (ab) Prior distributions for Flickr tags ‘harlemriver’ (blue) and ‘stadium’ (black) around New York City with a grid size of 0.5 km. **(c)** A photo with the tag ‘stadium’ can be narrowed down to the neighborhood of the Yankee Stadium, using MLE, if the next photo has been tagged with ‘harlemriver’.

secutive, presumably nearby photos of a user to estimate their locations jointly. This idea is shown to improve the maximum likelihood estimate.

The set of items V consists of all photos and all tags. We only focus on estimating the photo locations, so the set U contains only the photo items. As the candidate locations for each photo and tag, we define an $N \times N$ grid over the city, giving a total of N^2 candidate locations. The prior distributions of the tags are learned by counting the occurrences of the tags in different grid cells from a set of training photos with known locations. For the photos, we employ a uniform prior over the whole grid.

Edges are created between a photo h and all its tags a. We assume a tag to be located in the same cell where its photo was taken, and thus, if $d(\ell(h), \ell(a)) > 0$, we set $\Pr[\{h, a\} \in E \mid d(\ell(h), \ell(a))] = 0$. Otherwise, when $d(\ell(h), \ell(a)) = 0$, the term $\Pr[\{h, a\} \in E \mid d(\ell(h), \ell(a))]$ simply corresponds to the probability of tag a at location $\ell(h)$, which can be estimated as the fraction of training photos at $\ell(h)$ having tag a. Considering only the edges between photos and tags, would lead to the multiplication of tag probabilities in each cell, which is equivalent to the maximum likelihood solution proposed by Serdyukov et al. [12]. They show that geolocation accuracy can be improved by applying various smoothing techniques, but for simplicity, we have only applied Laplace smoothing, adding a dummy count of 0.1 to each grid cell.

Furthermore, we create an edge between two consecutive photos u and v taken by the same user, within a 5-minute interval. The underlying assumption is that such photos have been taken in nearby locations. Term $\Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))]$ corresponds to the edge probability between two photos by a single user given their distance. This term peaks at distance 0 and then decreases monotonically. In this experiment, distances are measured by the Manhattan distance. Figure 5 illustrates the advantage of using edges between consecutive photos. Dots correspond to individual photos with a given tag and heatmaps show the estimated probabilities.

Computation. Next we show how to evaluate Eq. (1) used in Algorithm 1 conveniently using matrix operations due to shared candidate locations (recall that we assume that each photo can be located in any grid cell). First, we define a linear indexing from 1 to N^2 for the grid cells. Second, we have two different types of edges: "photo–tag" and "photo–photo." For a photo h we denote by $N_t(h)$ the tags of the photo, and by $N_p(h)$ the neighboring photos of h (which can be 0, 1, or 2, as we assume a linear order induced by time with a 5-min cutoff). Then, the evaluation of Eq. (1) for a location $j \in \{1, \ldots, N^2\}$ takes the following form:

$$\begin{aligned}
\Pr[E \mid \ell(h) = j] &= \prod_{v \in N(h)} \sum_{\ell(v)} \Pr[\ell(v)] \; \Pr[\{h, v\} \in E \mid d(j, \ell(v))] \\
&= \prod_{v_t \in N_t(h)} \sum_{i=1}^{N^2} \Pr[\ell(v_t) = i] \; \Pr[\{h, v_t\} \in E \mid d(j, i)] \\
&\quad \times \prod_{v_p \in N_p(h)} \sum_{i=1}^{N^2} \Pr[\ell(v_p) = i] \; \Pr[\{h, v_p\} \in E \mid d(j, i)] \\
&=: \; A(h, j) \prod_{v_p \in N_p(h)} B(v_p, j).
\end{aligned}$$

Let us first look at term $A(h, j)$. As pointed out earlier in this section, the edge probability is nonzero only when $d(i, j) = 0$ and thus we get rid of the summation

$$A(h, j) = \prod_{v_t \in N_t(h)} \Pr[\ell(v_t) = j] \; \Pr[\{h, v_t\} \in E \mid d(j, j)] = C \prod_{v_t \in N_t(h)} \Pr[\ell(v_t) = j],$$

where C is a constant.

Let P be the number of photos and $\boldsymbol{A} \in \mathbb{R}^{P \times N^2}$ a matrix defined by $A_{v,j} = A(v, j)$. The matrix $\boldsymbol{A}$ can be precomputed before starting Algorithm 1, since the location distributions of the tags are not updated.

Then let matrix $\boldsymbol{T} \in \mathbb{R}^{P \times N^2}$ denote the uniform prior probabilities of the photo locations, given by $T_{v,j} = \Pr[\ell(v) = j] = \frac{1}{N^2}$, and $\boldsymbol{P} \in \mathbb{R}^{N^2 \times N^2}$ the edge probabilities given the locations of the adjacent vertices $P_{\ell(u),\ell(v)} = \Pr[\{u, v\} \in E \mid d(\ell(u), \ell(v))]$. Now we notice that we can compute $B(v_p, j)$ using the following vector multiplication

$$B(v_p, j) = \sum_{i=1}^{N^2} \Pr[\ell(v_p) = i] \; \Pr[\{h, v_p\} \in E \mid d(j, i)] = \boldsymbol{T}_{v_p,\cdot} \boldsymbol{P}_{\cdot,j}$$

and thus

$$\Pr[E \mid \ell(h) = j] = A(h, j) \prod_{v_p \in N_p(h)} B(v_p, j) = A_{h,j} \prod_{v_p \in N_p(h)} \boldsymbol{T}_{v_p,\cdot} \boldsymbol{P}_{\cdot,j}.$$

Using this formula, we can execute Algorithm 1 efficiently and conveniently, updating the matrix $\boldsymbol{T}$ at every iteration.

Data preprocessing. The Flickr service allows users to geolocate their photos at different accuracy levels. The dataset we use contains only the photos with the highest accuracy level. Furthermore, in Flickr, it is possible to upload multiple photos with the same set of tags in a bulk. To reduce noise in the data due to bulk uploads, we only keep the first photo if there are multiple photos from the same user, on the same date, and with exactly the same tags. Additionally, we filter out the tags that have been used by fewer than three users. The photos originate from a 10 km × 10 km area in the center of New York City, and in total we have 727 457 photos, 42 519 users, and 59 190 unique tags, after the aforementioned preprocessing steps.

Geotagging results. In our experimental setup, the grid size is set to 10×10 so that each cell is 1 km by 1 km. From each user, we take only the photos that have an edge to another photo to understand the effect of linking photos in more detail. A 10-fold cross-validation over users is employed to evaluate the performance of the methods.

A majority-vote baseline, computed by predicting the grid cell with the largest number of training photos, yields an accuracy of 23.5%. The method by Serdyukov et al. [12] obtains a clearly higher accuracy of 46.6%. This is further improved by the MLE method, which converges in two iterations, yielding an accuracy of 47.1%. The 0.5% improvement is statistically significant ($p < 0.001$) according to McNemar's test, suggesting that the information regarding consecutive photos could prove useful when designing methods tailored for the photo geotagging problem.

5 Conclusions and discussion

We have presented a probabilistic framework for inferring the geographical locations of objects. We assume that the objects are linked in a graph structure, and a prior distribution of the object locations is available. We showed that these assumptions are mild, and many application scenarios fit the proposed setting. To demonstrate the generality of the proposed method and to evaluate its performance, we presented detailed experiments for three different types of geolocation problems. Our evaluation indicated that the proposed method outperforms two other recently proposed network-based geolocation methods, BACKSTROM* and JURGENS.

An important novelty of the method is that it can manage large degrees of uncertainty in the data. Unlike the existing approaches we are aware of, our method does not need to assume the exact locations for any of the objects in the data. This is convincingly demonstrated in all three of our case studies.

Several future directions are worth exploring. In some cases it is more natural to treat location as a continuous variable. Thus we could try to find the maximum likelihood estimate in the continuous case, employing gradient-based methods. Also, it would be interesting to compare the proposed approach with undirected graphical models (Markov random fields), in which approximate inference can be achieved, for instance, by adopting loopy belief propagation.

6 Acknowledgments

We would like to thank the team in the Genealogical Society of Finland and Jouni Malinen for making the HisKi data available to us. We also thank Géraud Le Falher for sharing the Flickr dataset.

References

1. Ahmed, A., Hong, L., Smola, A.: Hierarchical geographical modeling of user locations from social media posts. In: Proceedings of the 22nd International Conference on World Wide Web. pp. 25–36. ACM (2013)
2. 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)
3. 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)
4. Crandall, D.J., Backstrom, L., Huttenlocher, D., Kleinberg, J.: Mapping the world's photos. In: Proceedings of the 18th International Conference on World Wide Web. pp. 761–770. ACM (2009)
5. Jurgens, D.: That's what friends are for: Inferring location in online social media platforms based on social relationships. In: Proceedings of the 7th International AAAI Conference on Weblogs and Social Media. pp. 273–282 (2013)
6. 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 International Conference on Knowledge Discovery and Data Mining. pp. 1023–1031. ACM (2012)
7. Mahmud, J., Nichols, J., Drews, C.: Where is this tweet from? Inferring home locations of Twitter users. In: Proceedings of the 6th International AAAI Conference on Weblogs and Social Media. pp. 511–514. ACM (2012)
8. 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)
9. O'Leary, M.: The mathematics of geographic profiling. Journal of Investigative Psychology and Offender Profiling 6(3), 253–265 (2009)
10. Rout, D., Bontcheva, K., Preoţiuc-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)
11. Sadilek, A., Kautz, H., Bigham, J.P.: Finding your friends and following them to where you are. In: Proceedings of the 5th ACM International Conference on Web Search and Data Mining. pp. 723–732. ACM (2012)
12. Serdyukov, P., Murdock, V., Van Zwol, R.: Placing Flickr photos on a map. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 484–491. ACM (2009)
13. Yamaguchi, Y., Amagasa, T., Kitagawa, H., Ikawa, Y.: Online user location inference exploiting spatiotemporal correlations in social streams. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. pp. 1139–1148. ACM (2014)