Exploring Multiple Evidences to Infer Users Location in Twitter

Departamento de Estatística, Universidade Federal de Ouro Preto, Diogo de Vasconcelos, 122, Ouro Preto, Brazil
Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Avenida Antônio Carlos, 6627, Belo Horizonte, Brazil

Abstract

Social networks are valuable sources of information to monitor real-time events, such as earthquakes and epidemics. For this type of surveillance, users location is an essential piece of information, but a substantial number of users choose not to disclose their geographical information. However, characteristics of the users’ behavior, such as the friends they associate with and the types of messages published may hint on their spatial location. In this paper, we present a method to infer the spatial location of Twitter users. Unlike the approaches proposed so far, we incorporate two sources of information to learn geographical position: the text posted by users and their friendship network. We propose a probabilistic approach that jointly models the geographical labels and Twitter texts of users organized in the form of a graph representing the friendship network. We use the Markov random field probability model to represent the network and learning is carried out through a Markov chain Monte Carlo simulation technique to approximate the posterior probability distribution of the missing geographical labels. We show the accuracy of the model in a large dataset of Twitter users, where the ground truth is the location given by the GPS position. The method is evaluated and compared to two baseline algorithms that employ either of these two types of information. The results obtained are significantly better than those of the baseline methods.

Keywords: Network Learning; Geographic Targeting; Geolocation Estimation.

1. Introduction

Social networks, such as Twitter and Facebook, were initially conceived as tools to encourage social interactions. However, with time they became powerful real-time sensors, gathering information about what people think and do all over the world. As a consequence, they started being employed as monitoring tools, used to provide information about events as diverse as earthquakes [1], epidemics [2] or elections [3].

When monitoring events, knowing where the information comes from is really valuable. Although users from social networks can fill up profiles with their personal information, this data is not always available or can be trusted. For example, [4] reported that, on average, only 35% of Facebook users declare location, while [5] showed that, among US Twitter users, 75% of them fill up the corresponding field. However, because the location field does not follow any patterns, a large volume of invalid (Mars) or low precision (Brazil) locations are often reported.

Apart from the declared location, user location in Twitter can be provided in two other ways: obtaining the geographic location from the computer IP address or from the GPS coordinates of mobile devices. The location given by the IP address is not very reliable and needs to be continually updated. In Brazil, for example, this service correctly locates 72% of IPs with precision within a radius of 40 kilometers [6]. GPS provided locations are the ones with best accuracy and reliability, since they are restricted to users posting from mobile devices and that allow such information to be disclosed. However, experimental results showed that, in countries such as Brazil, under 10% of tweets provide GPS data.

Given the importance of geolocation for event monitoring, studies on inferring user location based on other public available data are growing. This is also the main goal of this paper: to present a new method for inferring users location in Twitter by combining two sources of evidence: the tweets of the user [7, 8] and their relationships in the network [9, 10]. Although either tweets content or the friendship network have been previously used to infer the location of a Twitter user, to the best of our knowledge, the first method to propose a way to integrate these two sources of evidence was published in [11], and is extended in this paper.
The method proposed in [11] is based on a probabilistic graphical model, where the information of users friendships is represented as an undirected graph, with vertexes representing users and edges representing relationships. A friendship is defined by a mutual Twitter relationship among two users. Each vertex in the graph is also associated with the text of the tweets a user posts, and labeled with the geographical location coming from a GPS.

Other location sources are ignored, as they represent less reliable sources. Note that the spatial locations are only partially observed, with a substantial proportion of the users missing GPS geographical information. Based on a joint, integrated stochastic model relating different information sources, an algorithm is derived to learn the missing spatial positions. The inference is based on the posterior distribution of the possible geographical locations given all the available information: the entire network structure, the tweets’ contents for every user, and the observed spatial locations of some users.

We modeled the non-relational data as a Markov random field with neighborhood structure given by the social network links. A maximum a posteriori estimator is adopted as a point estimate but uncertainties probabilities associated with the estimated labels can be easily obtained as a side product from our learning algorithm. We used the Gibbs Sampler algorithm, a specific member of the Markov chain Monte Carlo (MCMC) algorithm class [12], in order to get the posterior probabilities estimates.

Apart from a more detailed description of the method previously proposed, this paper evaluates the sensitivity of the temperature parameter of a probabilistic Potts model [13] in the method, and also assesses the deterioration of the learning algorithm as the proportion of missing spatial locations increases. Finally, it presents new experiments with an updated dataset with 3 and 10 cities, respectively, with 8,397 and 11,850 users and more than one hundred thousand connections.

The results were compared to methods using either the tweets content or the friendship graph, namely Naive Bayes and the MultiRankWalk (MRW). They show that the aggregation of the two evidence sources in a single stochastic model to learn the missing locations improves the values of $f_1$ from 0.75 to 0.82 for 3 and 0.49 to 0.58 for 10 cities. The rest of the paper is organized as follows. Section 2 summarizes the current state of the art to learn the geographical location of social network agents. Section 3 describes our probabilistic model and Section 4 presents the experimental results. In Section 5, we present the main conclusions and future work.

2. Related Work

This section reviews recent works in the literature that consider the text, information of user profile or the relationship graph when predicting the geographical location of a user.

We start by discussing the works that consider only the text associated with each user. Cheng et al [8] used a classifier to automatically identify words within tweets that are strongly related to a local geographic scope, and then estimate users locations using a smoothing model that searches for the identified words in the messages.

Mahmud et al. [7], in contrast, added up information about tweeting behavior (volume of tweets per time unit) and external location knowledge (e.g., dictionary containing names of cities and states) together with text to infer users location. They used an ensemble of classifiers to explore the aforementioned features. The use of location dictionaries (aka gazetteers) is a very popular approach when looking for locations in Web text in general, as showed by [14].

Li et al. [15] inferred tweets’ location based on their text and moment of creation. For each place of interest (such as a movie theater, restaurant or gym), the method builds a unigram-based textual model and a temporal model. The temporal model estimates a probability distribution for tweets along the day, month and year. Given a new tweet, both models are used to rank points of interest as to their probability of being referred to by the message. Similarly, Ikawa et al. [16] identified tweets posted using services such as Foursquare, and associated their terms with the mentioned location. Remaining tweets were then associated with the identified locations according to the closest moment of creation, and classification is performed by finding the minimum distance between a new message and location-associated lists of terms.

With respect to works using the Twitter friendship network, a comprehensive review about learning from graph relationships can be found in [10]. Regarding works focused on the Twitter graph, [9] proposed an estimation method whereby a user’s location is set as the one which is the most frequent among his friends.

Lin and Cohen [17] proposed MRW (Multi-RankWalk), a method able to classify sites in a semi-supervised manner, i.e., where only few vertices of the graph need to be labeled. Their idea is to use an algorithm similar to PageRank [18] to infer the labels of the vertices, creating multiple rankings using random walks from seed (labeled) instances. This approach is the one we compare to the method proposed here.
Crandall et al. [19] proposed a very different approach where, knowing that two users have been in approximately the same geographic location at approximately the same time, on multiple occasions, it estimates their probability of knowing each other.

Finally, Gonzales et al. [20] investigated the effects of locality in Twitter, focusing specially on user/followers location relations. One of their results shows that, in countries where English is not the first language, there is a high intra-country locality among users and their followers, while English-speaking countries suffer from what they call external locality effect, having many of their followers in the U.S. In this sense, Backstrom et al. [21] analyzed a user graph collected from Facebook. They show that most users have at least one friend whose address is available, and that users who have more than four friends with known location can be more precisely located by inference over the graph than by GeoIP.

3. A probabilistic model for inferring location

This section describes the algorithm proposed to learn users’ location based on the content of his tweets and his friendship graph. Let $N$ be the total number of users, $\theta_{i}$ the $i$-th user location and $\theta = (\theta_{1} \cup \theta_{i})$, where $\theta_{i} = (\theta_{1}, \theta_{2}, \ldots, \theta_{k})$ is the set of $k$ users with known location (labeled nodes) and $\theta_{i} = (\theta_{k+1}, \theta_{k+2}, \ldots, \theta_{N})$ the set of users with unknown location (unlabeled nodes). Denote by $\theta_i$ the $(N - 1)$-dimensional vector with all the location labels except that of the $i$-th user. The friendship graph, our first data source, is defined by $G = (V, E)$, where $V$ represents the set of $N$ users (vertices) and the edges $E$ represent the mutual follower relationships between pairs of users.

We want a model allowing for geographically close users to be connected more often than not. We also need a model that relates probabilistically the $i$-th user text $w_i$ to his geographical label $\theta_i$.

To specify the joint distribution of all nodes in $\theta_{U}$ is a very difficult task due to the intricate mutual interaction between the individual variables. Because of that, we resort to the Brook’s lemma [22], which guarantees that the joint distribution is completely determined by the specification of consistent individually defined conditional distributions. That is, to obtain the joint distribution (1), all we need is to provide the set of $N - k$ univariate conditional probability distributions

$$P(\theta_{i} | \theta_{1}, \theta_{k+1}, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_{N}, w_{1}, \ldots, w_{N}, G) \quad (2)$$

for $i = k + 1, \ldots, N$. These conditional distributions are called full conditional distributions as the conditioning event at the $i$-th distribution is composed by all remaining variables but the $i$-th one. The importance of this result is that it allows the user to to specify (conditional) models for univariate random variables $\theta_i$ rather than having to deal with the simultaneous correlation among the set of $N - k$ variables of the vector $\theta_{U}$ if one is required to specify directly the joint distribution (1).

An additional advantage of specifying the probability distribution of $\theta_{U}$ through the set of $N - k$ univariate full conditional distributions in (2) is the possibility of using the Gibbs Sampler [12] algorithm to learn the hidden geographical labels. We return to this issue after completing the model specification.
For each $\theta_i \in \Theta_U$, we can be factorize the full conditional distribution

$$P(\theta_i|\theta_{-i}, w_i) \propto P(\theta_i|\theta_{-i})P(w_i|\theta_i, \theta_{-i}).$$

This decomposition breaks down the specification problem into two distinct tasks. First, we need to specify a probability distribution for the hidden label $\theta_i$ conditionally on the geographical labels of all other vertices in the graph, ignoring the documents. Note that this specification is conditional on the knowledge of all other node labels, including the hidden ones in the vector $\theta_U$. The second task is to specify a generative probability model for the document associated with the $i$-th user given the entire set $\Theta$ of node labels.

For the first task, we reduce the probability distribution of the geographical locations assuming a sparse representation for $\Theta$ based on a Markov random field model. The sparsity is induced by the assumption that, providing the information about the location of a user’s friends, the rest of the information contained in the network can be ignored.

Thus, $P(\theta_i|\theta_{-i})$ is simplified to

$$P(\theta_i|\theta_{-i}) = P(\theta_i|\theta_D)$$

where the vector $\theta_D$ contains the location of all the neighbors of $i$. Note that, typically, $\theta_D$ is composed by elements from the observed labels in $\Theta_L$, as well as from the hidden vector $\Theta_U$. Indeed, in some cases, $\theta_D$ can be entirely composed of hidden nodes contained in $\Theta_U$. The estimation procedure will need to somehow probabilistically propagate the information contained in $\Theta_U$ to fill this missing data.

As the possible locations form a finite set of labels, we modeled the Markov random field using the Potts Model [13], a distribution well known by image restoration researchers, and which is a generalization of the celebrated Ising model for black-and-white images. According to the Potts’ model, the joint probability of a given configuration depends on its energy measured by the degree of similarity between neighboring sites. The induced conditional probability that a site belongs to a particular class is an increasing function of the number of his neighbors which pertain to the same class, i.e., neighboring sites tend to belong to the same class. Since the sum factor is non-negative, a large value of $\beta$ puts more probability mass on configurations with equal labels among neighboring vertices. On the other extreme, if we take $\beta$ equal to zero, we obtain a completely uniform probability distribution, where any labels’ configuration is equally likely and there is no increased probability for neighbors to share the same label. Negative values for $\beta$ are theoretically possible but unlikely to make sense as it induces a negatively correlated configuration: a given label for a node would induce different labels on its neighbors. Therefore, $\beta$ is an important parameter and we have two options to set its value. One is to assume a flat prior distribution over the range of possible values and enlarge the probability model with this incremental uncertainty. The second one, which we adopted, is to optimally tune this parameter by cross-validation. This has the additional advantage of allowing us to study the sensitivity of the algorithm to different values of this hyper-parameter.

The Markov random field allows for the correlation among labels exploring the graph connectivity between users. To appreciate the usefulness of this probability model, consider a situation where a user has no geographical label and is surrounded in the connection graph by users with no geolocation information. The Markov model can still infer the geolocation of the central user by looking at farther apart neighbors and propagating this more distant information down to the specific user. During this process, it automatically takes into account the entire graph topology, such that not every user enters equally likely in predicting a given label.

Focusing on the second task, we need to specify the term $P(w_i|\theta_i, \theta_{-i})$. In our generative model, to predict a user’s text, the geographic information of his friends is conditionally independent given that we know his own location $\theta_i$. Thus, this probability is simplified to

$$P(w_i|\theta_i, \theta_{-i}) = P(w_i|\theta_i).$$

To find the value of $P(w_i|\theta_i)$ we use the bag of words model by considering the successive words posted by the user as independent of one another. That is,

$$P(w_i|\theta_i) = \prod_j P(w_i|\theta_i).$$

where $w_{ij}$ denotes the $j$-th word published by the $i$-th user. This assumption is obviously only an approximation. Despite its simplicity, the method obtained very good results when this information was coupled with the graph structure summarized in the Markov random field model, as we show in Section 4. Each probability $P(w_i|\theta_i)$ is estimated by the empirical frequency that
the word $w_{ij}$ appears among all the words published by all users residing in location $\theta_i$.

With these assumptions, the conditional probability $P(\theta_i|\theta_{-i}, w_i)$ turns out to be

$$P(\theta_i|\theta_{-i}, w_i) \propto P(\theta_i|\theta_{w}) \prod_j P(w_{ij}|\theta_i).$$

The values of $\theta_i$ for those users whose location is unknown are updated by the Gibbs Sampler algorithm [12], belonging to the class of Markov chain Monte Carlo algorithms. In this algorithm, a probability distribution for the entire set of hidden labels is obtained, conditioned on those labels that are known and on the text produced by the users. For a given user with missing location $\theta_i$, its marginal conditional probability will assign probabilities for all possible labels and the inference can be based on the maximum a posteriori since the possible values for the geographical labels are categorical values. More specifically, giving an initial and arbitrary configuration for $\theta_i$, the Gibbs Sampler algorithm loops over its $N - k$ elements sampling a location $\theta_i$ from the univariate complete conditional distribution

$$P(\theta_i|\theta_{-i}, w_i).$$

After a certain number of burn-in simulations, the sampled values will be approximately selected from the joint probability distribution $P(\theta_{i:}|\theta_{-i}, w_1, \ldots, w_N)$. The estimated label $\widehat{\theta}_i$ is given by

$$\widehat{\theta}_i = \arg \max_{\theta_i} P(\theta_{i:}|\theta_{-i}, w_1, \ldots, w_N).$$

4. Experimental Results

In order to verify the performance of our probabilistic model, tests were performed using a Twitter dataset with 11,850 users from 10 different Brazilian cities, collected in the first semester of 2011 using a breath-first search. The seeds for the search were users using the terms “dengue” and “aedes aegypti”, as the objective was to improve localization of users talking about dengue outbreaks in Brazil. As we expect the number of available labels (cities) as well as the number of missing examples to have an impact in the method, experiments were performed in two phases.

First, a subset of 8,477 users from three big cities in Brazil, namely Belo Horizonte (BH), Rio de Janeiro (RJ), and São Paulo (SP), were selected. In a second phase, the whole dataset, with 10 cities, was evaluated. The city of a user is set as the city with the most frequent number of GPS located messages. The 11,850 users form a graph composed of 125,145 edges, shown in Figure 2. Note that there is a strongly connected component in this graph and many isolated users, with less than two friends.

Apart from the users and their connections, the text of the users’ most recent 200 tweets was also retrieved. We consider 200 tweets because that is the number the Twitter API allows us to collect without repaginating. We consider unigrams of the text of the tweets, and removed punctuation, stop words and URLs. Words with frequency smaller than three are also removed from the dataset. After this process, we ended up with 103,498 distinct terms for 3 cities and 353,926 for 10 cities. The frequency of the terms in the users’ tweets is used as the input $w_i$ to the proposed model.

In the experiments, 70% of the users were randomly selected to compose the training set and the remaining 30% the validation set. The results obtained were compared to two other approaches: MRW, a method proposed by [17] and based on the friendship graph, and Naive Bayes (NB) using the tweets content. MRW was implemented using the Igraph package of R [23], and the probability of teleportation set to 0.5.

4.1. Experiments with 3 cities

This section shows results obtained when using 8,477 users in three cities. The data distribution is unbalanced, with 17.66%, 43.98% and 38.36% users in BH, RJ and SP, respectively. As the model results depend strongly on the value of the temperature $\beta$, we first calibrate this parameter. Figure 3 shows the accuracy of the method (success rate) for different values of the parameter. The graph shows that the model performance increases with $\beta$ until it reaches a plateau, where it stabilizes. Note that...
Table 1: Results obtained by the proposed method (Integrated-Data Approach - IDA) and two baselines that consider the information about the two data sources independently

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision (%)</th>
<th>Acc</th>
<th>f1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BH</td>
<td>RJ</td>
<td>SP</td>
</tr>
<tr>
<td>MRW</td>
<td>56.98</td>
<td>88.78</td>
<td>72.65</td>
</tr>
<tr>
<td>NB</td>
<td>40.49</td>
<td>80.84</td>
<td>74.86</td>
</tr>
<tr>
<td>IDA (tf)</td>
<td>83.57</td>
<td>89.40</td>
<td>73.09</td>
</tr>
<tr>
<td>IDA (tf-idf)</td>
<td>78.20</td>
<td>81.49</td>
<td>78.6</td>
</tr>
</tbody>
</table>

The fluctuations in the success rate for $\beta$ in the interval from 20 to 50 are purely random. The results reported here used $\beta$ equals to 30.

Table 1 shows the results obtained by the three methods considering the 2,544 users in the validation set. We report the precision for each class followed by the overall accuracy and the macro-f1 metric. Macro-f1 corresponds to the harmonic mean of the precision and recall values. The results show that MRW correctly predicted the location of 75.06% of users, while Naive Bayes correctly predicted 67.42%. Hence, the information from the graph showed to be more relevant to predict the location of a user than the text. Notice that either of these two baseline methods has an increasing probability of correctly predicting the locations as the users’ population increase.

The results obtained by the proposed method are reported in the last two lines of Table 1 (IDA-Integrated-data approach). The location the $i$-th user is estimated by that with the highest probability among all possible locations. Initially, we focus on the results of IDA (tf), which considers the frequency of the words in the tweets. Overall, this method correctly infers the location of 82.14% of users, which is substantially greater than our baseline algorithms performance. For users living in BH, RJ and SP the proportion of corrected inference were, respectively, 83.57%, 89.40% and 73.09%. Combining both types of data improved the overall accuracy and f1 at the cost of a smaller precision for SP, which decreased from 74.86% to 73.09% when compared to the Naive Bayes algorithm. This may be due to sampling variance only, not reflecting any intrinsic aspect of the problem.

In order to provide a more detailed analysis of the results, Table 2 presents how the results of precision and recall were distributed among classes using a confusion matrix-like representation. The lines correspond to the real classes and the columns to the predicted classes. The first matrix, reporting precision, shows how many of the users were correctly classified in class $i$ and, when misclassified, to which cities they were attributed to. For example, SP has the lowest precision, with 73.09%, but the highest recall, 87.55%. When predicting SP (last column), 20.8% of users from Rio were misclassified as being from SP and 6.12% from BH. This may be due to the great interaction among users from RJ and SP, which are strongly connected. Conversely, from the recall data, we can observe that, from all users in SP (last line of second part of the table), 87.55% were attributed to SP, and the remainder were mostly assigned to RJ (8.55%).

A disadvantage of using only the frequency of the
terms when describing different users is that these are not necessarily discriminative. Frequent terms in all cities, for example, are not good at discriminating them. Hence, we considered another way to assign weights to terms, replacing the term frequency by the $tf-idf$ method (term frequency-inverse document frequency) [24]. The value of $tf-idf$ is high when the term is rare across the dataset but very common in the document under analysis. The $tf-idf$ is composed of two parts. The first is the term frequency and the second the inverse document frequency.

In our case, we consider as if one city was a document. Consider $C$ the complete set of cities available in the dataset. Let $t$ and $c$ be a specific term and city in the dataset. The term frequency is defined as the relative frequency of term $t$ in tweets of users belonging to city $c$, and is defined as

$$tf(t,c) = \frac{f(t,c)}{\max\{f(t,c) : c \in C\}}$$

where $f(t,c)$ is the frequency of term $t$ in tweets from users belonging to city $c$. The second part is the inverse document frequency and is defined as

$$idf(t, C) = \log \frac{|\{d \in C : t \in c\}|}{|C|}$$

where $|C|$ is the total number of cities and $\{c \in C : t \in c\}$ is the number of cities where term $t$ occurs. The $tf-idf$ is given by a combination of these two terms, defined as

$$tfidf(t,d) = tf(t,d) \times idf(t,D).$$

The last line in Table 1 shows the values of accuracy and $f1$ achieved using $tf-idf$, together with the precision per class. Note that the results obtained are similar to those reported without any weighting factor, but with $f1$ dropping from 0.82 to 0.79. These results indicate that a selection of the most relevant terms might be more effective for the proposed model than proposing different weighting schemes.

Finally, we also evaluated how the model performance deteriorates as we decrease the size of the training set. A smaller number of labeled nodes should make it harder for the method to correctly predict the missing vertices’ labels. Figure 4 shows the results obtained for the IDA and Naive Bayes algorithms. The graph shows that the accuracy of the proposed method falls slowly as we increase the proportion of the vertices with hidden labels to test the classifier. Regarding Naive Bayes, we observe that its results are uniformly worse than those of IDA, with a difference of at least 0.3 and a very sharp drop at the beginning. Therefore, Naive Bayes presents a worse performance than IDA at any missing information rate. IDA, in contrast, is able to predict the labels correctly even when only a few of them are known.

### 4.2. Experiments with 10 cities

The previous section showed how the method behaves when a small number (three) of labels can be assigned to vertices. It also considers three cities with a high absolute number of vertices available to learn from. This section adds to the dataset seven other cities, some of them with frequencies as low as 1.77% (see last line of Table 4).

The results obtained by the three methods are summarized in Table 3, where we show the accuracy and $f1$ obtained by IDA using only term frequency and MRW and Naive Bayes. As observed, again the performance of IDA is much better than those of the baselines, with an $f1$ of 0.58 against 0.496 of MRW and 0.367 of Naive Bayes.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>$f1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRW</td>
<td>54.23</td>
<td>0.496</td>
</tr>
<tr>
<td>NB</td>
<td>27.62</td>
<td>0.367</td>
</tr>
<tr>
<td>IDA (tf)</td>
<td>80.59</td>
<td>0.58</td>
</tr>
</tbody>
</table>
This experiment reflects better the reality of Twitter, where we do have cities with a very small number of users. Table 4 shows the results using the confusion matrix-like representation introduced in the previous section. Note that the smallest precisions are for cities including Fortaleza (44%), POA (42.86%), Recife (28%) and Vitoria (VIT) (51.72%), which are also the less frequent cities. In this case, an under-sampling of the other cities might benefit the method.

Regarding the recall, we observe that 43.55% of users from POA, 36% of users from Fortaleza and 18.37% of users from VIT were misclassified as SP. This might be the cause because, apart from being very frequent, the relationships of all users with those in Sao Paulo are very intense.

For the four cities with label distributions smaller than 2%, we did not obtain results with similar quality to those obtained for cities with a greater number of nodes to learn from. However, even for these cities, the results obtained are much superior than those of the baseline methods. For the other 3 cases of small samples, the methods showed that the built-in propagation of the local information to farther apart locations in the graph is a very important feature of our algorithm and one that we think makes it more powerful than alternate methods.

5. Conclusions and Future Work

This work presented a probabilistic model to infer the location of Twitter users. It extends original work presented in Rodrigues et al (2013) [11], where we first propose to integrate information from Twitter users’ texts and friendship network. The method, which was previously tested in a dataset with three cities, was retested in an updated dataset with up to ten cities. The accuracies obtained, when compared to baselines using only one source of information, were better, with improvements in /1 from 0.72 to 0.82 for 3 cities and 0.496 to 0.58 to 10 cities.

The model that integrates information from both sources is based on a Markov random field distribution for the geographical labels, which are partially observed. For the text, we assume a bag-of-words model and also tested a weighted version considering a city-based tf-idf model. One novelty in this work is the use of the Markov random field model, a much more sophisticated probability description than those proposed so far by other researchers. In this model, even farther apart users in the connection graph can impact one’s label probability distribution. The model automatically takes into account the distance in the neighborhood graph as well as the presence of other neighboring and redundant users.

We also found that the proposed methodology is robust to reductions in the size of the training base, and robust to the parameter of the method. This means that we need to know the location of relatively few users to be able to infer about others when we have the friendship graph and the text associated with all the users.

As future work, we can add other sources of information to the model, such as data from the social network Foursquare, to deal specially with data sparsity. Borrowing information from other sources could have the same leverage effect as we found in using the text posted by users.

Another direction is to consider not only GPS data when building the graph, but also incorporate those obtained by IP addresses or declared by users. In this case, the idea would be to assign different levels of confidence for labels coming from different sources.

Furthermore, the methodology used in this work can also be extended to analyze data from other social networks such as Facebook, Flickr, and Instagram. Therefore, our proposed probabilistic model have other applications beyond the specific Twitter case study we presented here. In all these possible applications, the main driving idea is to borrow information from one information source to help infer other hidden variables. The degree of success will depend on the correlation degree of the two information sources.

Acknowledgments

The authors would like to thank CNPq, Capes, FAPEMIG, and InWeb – National Institute of Science and Technology for the Web for financial support.

References

Table 4: Confusion matrices obtained by the method with 10 classes

<table>
<thead>
<tr>
<th>Real</th>
<th>Pred.</th>
<th>BH</th>
<th>BSB</th>
<th>CTB</th>
<th>Fortaleza</th>
<th>Manaus</th>
<th>POA</th>
<th>Recife</th>
<th>RJ</th>
<th>SP</th>
<th>VIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BH</td>
<td></td>
<td>88.5</td>
<td>2.44</td>
<td>2.17</td>
<td>4</td>
<td>1.15</td>
<td>11.43</td>
<td>8</td>
<td>1.03</td>
<td>2.68</td>
<td>6.9</td>
</tr>
<tr>
<td>BSB</td>
<td>1.07</td>
<td>68.29</td>
<td>2.17</td>
<td>8</td>
<td>0.76</td>
<td>5.71</td>
<td>12</td>
<td>1.32</td>
<td>0.92</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>CTB</td>
<td>1.07</td>
<td>0</td>
<td>78.26</td>
<td>4</td>
<td>0.38</td>
<td>2.86</td>
<td>8</td>
<td>1.32</td>
<td>3.22</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>Fortaleza</td>
<td>0.53</td>
<td>7.32</td>
<td>4.35</td>
<td>44</td>
<td>0.38</td>
<td>2.86</td>
<td>12</td>
<td>0.44</td>
<td>1.38</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>Manaus</td>
<td>1.34</td>
<td>4.88</td>
<td>3.26</td>
<td>44</td>
<td>88.17</td>
<td>14.29</td>
<td>4</td>
<td>1.1</td>
<td>1.3</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>POA</td>
<td>1.07</td>
<td>0</td>
<td>1.09</td>
<td>8</td>
<td>0.38</td>
<td>42.86</td>
<td>8</td>
<td>0.59</td>
<td>2.07</td>
<td>6.9</td>
<td></td>
</tr>
<tr>
<td>Recife</td>
<td>0.8</td>
<td>0</td>
<td>1.09</td>
<td>8</td>
<td>2.67</td>
<td>2.86</td>
<td>28</td>
<td>0.88</td>
<td>0.92</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>RJ</td>
<td>2.41</td>
<td>9.76</td>
<td>1.09</td>
<td>12</td>
<td>2.29</td>
<td>5.71</td>
<td>8</td>
<td>89.65</td>
<td>15.54</td>
<td>13.79</td>
<td></td>
</tr>
<tr>
<td>SP</td>
<td>1.6</td>
<td>2.44</td>
<td>5.43</td>
<td>8</td>
<td>2.67</td>
<td>5.71</td>
<td>8</td>
<td>2.94</td>
<td>71.29</td>
<td>3.45</td>
<td></td>
</tr>
<tr>
<td>VIT</td>
<td>1.6</td>
<td>4.88</td>
<td>1.09</td>
<td>0</td>
<td>1.15</td>
<td>5.71</td>
<td>4</td>
<td>0.73</td>
<td>0.69</td>
<td>51.72</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Real</th>
<th>Pred.</th>
<th>BH</th>
<th>BSB</th>
<th>CTB</th>
<th>Fortaleza</th>
<th>Manaus</th>
<th>POA</th>
<th>Recife</th>
<th>RJ</th>
<th>SP</th>
<th>VIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BH</td>
<td>83.80</td>
<td>0.25</td>
<td>0.51</td>
<td>0.25</td>
<td>0.76</td>
<td>1.01</td>
<td>0.51</td>
<td>3.54</td>
<td>8.86</td>
<td>0.51</td>
<td></td>
</tr>
<tr>
<td>BSB</td>
<td>5.41</td>
<td>37.84</td>
<td>2.70</td>
<td>2.70</td>
<td>2.70</td>
<td>2.70</td>
<td>4.05</td>
<td>24.32</td>
<td>16.22</td>
<td>1.35</td>
<td></td>
</tr>
<tr>
<td>CTB</td>
<td>2.82</td>
<td>0.00</td>
<td>50.70</td>
<td>0.70</td>
<td>0.70</td>
<td>0.70</td>
<td>1.41</td>
<td>12.68</td>
<td>29.58</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>Fortaleza</td>
<td>4.00</td>
<td>6.00</td>
<td>8.00</td>
<td>22.00</td>
<td>2.00</td>
<td>2.00</td>
<td>6.00</td>
<td>12.00</td>
<td>36.00</td>
<td>2.00</td>
<td></td>
</tr>
<tr>
<td>Manaus</td>
<td>1.78</td>
<td>0.71</td>
<td>1.07</td>
<td>0.36</td>
<td>82.21</td>
<td>1.78</td>
<td>0.36</td>
<td>5.33</td>
<td>6.05</td>
<td>0.36</td>
<td></td>
</tr>
<tr>
<td>POA</td>
<td>6.45</td>
<td>0.00</td>
<td>1.61</td>
<td>3.23</td>
<td>1.61</td>
<td>24.19</td>
<td>3.23</td>
<td>12.90</td>
<td>43.55</td>
<td>3.23</td>
<td></td>
</tr>
<tr>
<td>Recife</td>
<td>6.52</td>
<td>0.00</td>
<td>1.61</td>
<td>4.35</td>
<td>15.22</td>
<td>2.17</td>
<td>15.22</td>
<td>26.09</td>
<td>26.09</td>
<td>2.17</td>
<td></td>
</tr>
<tr>
<td>RJ</td>
<td>0.62</td>
<td>0.27</td>
<td>0.07</td>
<td>0.21</td>
<td>0.41</td>
<td>0.14</td>
<td>0.14</td>
<td>83.92</td>
<td>13.95</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>SP</td>
<td>0.60</td>
<td>0.10</td>
<td>0.50</td>
<td>0.20</td>
<td>0.70</td>
<td>0.20</td>
<td>0.20</td>
<td>4.01</td>
<td>93.38</td>
<td>0.10</td>
<td></td>
</tr>
<tr>
<td>VIT</td>
<td>12.24</td>
<td>4.08</td>
<td>2.04</td>
<td>0.00</td>
<td>6.12</td>
<td>4.08</td>
<td>2.04</td>
<td>20.41</td>
<td>18.37</td>
<td>30.61</td>
<td></td>
</tr>
</tbody>
</table>

Distr. 9.49 1.94 2.76 1.77 7.55 1.55 1.75 36.33 34.89 1.97


