Georeferencing Flickr resources based on textual meta-data

Olivier Van Laere[a], Steven Schockaert[b], Bart Dhoedt[a]

[a]*Department of Information Technology, Ghent University, IBBT, Belgium*
[b]*School of Computer Science & Informatics, Cardiff University, United Kingdom*

Abstract

The task of automatically estimating the location of web resources is of central importance in location-based services on the Web. Much attention has been focused on Flickr photos and videos, for which it was found that language modeling approaches are particularly suitable. In particular, state-of-the art systems for georeferencing Flickr photos tend to cluster the locations on Earth in a relatively small set of disjoint regions, apply feature selection to identify location-relevant tags, then use a form of text classification to identify which area is most likely to contain the true location of the resource, and finally attempt to find an appropriate location within the identified area. In this paper, we present a systematic discussion of each of the aforementioned components, based on the lessons we have learned from participating in the 2010 and 2011 editions of MediaEval's Placing Task. Extensive experimental results allow us to analyze why certain methods work well on this task and show that a median error of just over 1 kilometer can be achieved on a standard benchmark test set.

Keywords: Text mining, Metadata, Geographic Information Retrieval, Classification, Semi-structured Data

1. Introduction

With the rising popularity of smartphones and tablet computers, location plays an ever increasing role on the web. Many applictions, including search engines, try to adapt the services they offer to the current location of the user. This requires that resources (e.g. web pages in the case of search engines) be associated with a geographic scope. Such geographic information can be obtained in various ways. One way of learning information about places is to encourage users to explicitly share information about their whereabouts with their friends and contacts. This is the case with Foursquare[1], on which users can compete

Email addresses: olivier.vanlaere@intec.ugent.be (Olivier Van Laere), S.Schockaert@cs.cardiff.ac.uk (Steven Schockaert), bart.dhoedt@intec.ugent.be (Bart Dhoedt)

[1] http://www.foursquare.com

each other for points they earn for each "check-in" at a certain place, or Twitter[2] where the user's current location can be attached to the tweet. Secondly, a gazetteer can be consulted as a source of geographical information. Gazetteers (for example GeoNames[3] or Yahoo! Geoplanet[4]) are essentially lists or indexes containing information about a large number of known places, described by different features such as geographical coordinates and semantic types. Creating and maintaining such a gazetteer is mostly expert driven and a cumbersome and time-consuming task. Gazetteers clearly provide a valuable source of geo-information, if one is able to disambiguate between the possibly multiple entities with the same name. For instance, if one needed details on an entity described as "Paris", a gazetteer would normally contain at least two entities: one for Paris, France and one for Paris, Texas. In absence of any additional information, it is hard to disambiguate between these two entities, although in this example using a "default sense" heuristic (based on for instance the population count) would in most cases return the correct meaning.

In our work, we focus on yet another way of gathering geographical information. As the amount of user-contributed textual data on the Web is growing every day (e.g. by means of status updates on social networks, comments, reviews, ratings, blog posts, tagged photo and video uploads), and as many of those contributions also include geographical coordinates, there is a vast amount of textual information available for automated mining of geographical knowledge. More specifically, in this paper, we show how such automatically obtained geographic knowledge allows us to estimate geographical coordinates for Flickr photos and videos, using only the textual information from their Flickr tags. To this end, we train a classifier from the tags of Flickr photos with known coordinates (i.e. the location where the photo was taken), which is capable of selecting the area in which a previously unseen photo or video has most likely been taken. In a subsequent step, our system tries to find a precise location within that area, by identifying the photos from the training data that are most similar to the photo or video we want to localise.

Several approaches to this problem of georeferencing Flickr resources have already been proposed in the literature [28, 27, 32, 36, 35, 34] . To facilitate the comparison of different solutions, the Placing Task has been introduced in 2010 as part of the MediaEval[5] evaluation campaign. This task requires participants to georeference Flickr videos based on the associated tags, visual features, and user profile information. Both in 2010 and 2011, our system came out as the best performing one. The research goal of this paper is to analyze the results of our system and to perform an in-depth evaluation of the contributions of each of the different steps in our approach to the overall results.

The remainder of this paper is organized as follows: Section 2 summarizes

[2] http://www.twitter.com
[3] http://www.geonames.org
[4] http://developer.yahoo.com/geo/geoplanet/
[5] http://www.multimediaeval.org

related work. A general overview of our approach to extracting implicit geographical information from Flickr is presented in Section 3, as well as the description of the techniques used to employ this textual information in estimating the location of Flickr photos and videos. Next, Section 4 provides an in-depth analysis of different approaches of each individual component of the georeferencing process along with experimental results. Finally, Section 5 states the conclusions and discusses future work.

2. Related work

2.1. Finding locations of resources

The task of deriving geographic coordinates for photos has recently gained in popularity; see e.g. [26]. [44] published a survey on recent research and applications on the topic of georeferencing resources. Most existing approaches are based on clustering, in one way or another, to convert the task into a classification problem. For instance, in [6] locations of unseen resources are determined using mean shift clustering, a non-parametric clustering technique from the field of image segmentation. The advantage of this clustering method is that an optimal number of clusters is determined automatically, requiring only an estimate of the scale of interest. Specifically, to find locations, the difference is calculated between the density of photos at a given location and a weighted mean of the densities in the area surrounding that location. To assign locations to new images, both visual (keypoints) and textual (tags) features were used. Experiments were carried out on a sample of over 30 million images, using both Bayesian classifiers and linear support vector machines, with slightly better results for the latter. Two different resolutions were considered corresponding to approximately 100 km (finding the correct metropolitan area) and 100 m (finding the correct landmark). It was found that visual features, when combined with textual features, substantially improve accuracy in the case of landmarks. A similar conclusion follows from the multimodal approach demonstrated and evaluated in [11]. In contrast, [24] discusses method using only visual information. A novel high-level representation for videos, called bag-of-scenes, is proposed. In this approach, each component of the representation has a self-contained semantics that can be directly related to a specific place of interest. Experiments were conducted in the context of the MediaEval 2011 Placing Task, using the same dataset that we will use in this paper. In [15], another approach is presented which is based purely on visual features. For each new photo, the 120 most similar photos with known coordinates are determined. This weighted set of 120 locations is then interpreted as an estimate of a probability distribution, whose mode is determined using mean-shift clustering. The resulting value is used as a prediction of the image's location. Around 16% of the resources in the test set can be estimated within 200 km of their actual location.

The idea that when georeferencing images, the spatial distribution of the classes (areas) could be utilized to improve accuracy has been suggested in [32]. Their starting point is that typically not only the correct area will receive

a high probability, but also the areas surrounding the correct area. Indeed, the expected distribution of tags in these areas will typically be quite similar. Hence, if some area a receives a high score, and all of the areas surrounding a also receive a relatively high score, we can be more confident in a being approximately correct than when all the areas surrounding a receive a low score. Motivated by this intuition, [32] proposes a location-aware form of smoothing when estimating probabilistic language models.

In addition to georeferencing Flickr photos, several authors have recently focused on finding the location of other web resources such as Twitter posts or Wikipedia pages. For instance, in [3], a probabilistic framework based on maximum likelihood estimation was used to estimate the location of users based on the content of their tweets. In particular, a generative probabilistic model proposed in [2] is used to determine words with a geographic scope within a tweet, and a form of neighborhood smoothing is employed to refine the estimations. For 51% of the users, a location was obtained that is within a 100 mile radius of their true location. Next, [40] looked into georeferencing Wikipedia articles as well as Twitter posts. After laying out a grid over the Earth's surface (in a way similar to [32]), for each grid cell a generative language model is estimated. To assign a test item to a grid cell, its Kullback-Leibler divergence with the language models of each of the cells is calculated. In [7], we have shown how Wikipedia pages can be georeferenced using language models that are trained from Flickr, taking the view that the relative sparsity of georeferenced Wikipedia pages does not allow for sufficiently accurate language models to be trained, especially at finer levels of granularity.

Interestingly, some recent language modeling approaches have combined the idea of topic models with location-dependent language models. For instance, [9] proposes geographic topic models with the aim of simultaneously capturing linguistic variation across different regions and different topics.

2.2. Using locations of resources

When available, the coordinates of a photo may be used in various ways. In [1], for instance, coordinates of tagged photos are used to find representative textual descriptions of different areas of the world. These descriptions are then put on a map to assist users in finding images that were taken in a given location of interest. Their approach is based on spatially clustering a set of geotagged Flickr images, using k-means, and then relying on (an adaptation of) tf-idf weighting to find the most prominent tags of a given area. Similarly, [23] looks at the problem of suggesting useful tags, based on available coordinates. The relevance of a given tag is measured in terms of the number of users that have used it to describe photos located within a certain radius of the current photo's coordinates. A refinement of this method only looks at tags that occur with visually similar photos, which is shown to improve the quality of the proposed tags. Along similar lines, our method could be used to suggest coordinates when users are tagging their photos and videos, automating the process that

is now carried out manually using Suggestify[6], a web application that enables people to suggest a location for ungeotagged Flickr photos of someone else. This could contribute to making a larger fraction of the photos and videos on Flickr associated with an explicit location[7]. As a related use case, we can consider the problem of making search engines aware of spatial constraints in users' queries. For example, to allow users to specify a geographic scope for their query, Google introduced an option to search *nearby*[8] in February 2010. Implementing such a method involves a correct interpretation of the spatial constraint (e.g. based on a gazetteer in combination with location information obtained from the user's IP address for disambiguation) and a mechanism to identify the geographic scope of a website [18]. This latter problem could be solved using a combination of different methods. Web pages containing explicit mentions of addresses could be localised using standard techniques for geocoding (e.g. by comma group resolution [22]). In general, however, the textual content of the web page needs to be used as evidence. While traditionally gazetteer-based methods have been used to this end, initial results have shown that our model for georeferencing based on language models trained from Flickr can successfully be used to georeference resources such as Wikipedia pages [7].

Some authors have looked at using geographic information to help diversify image retrieval results [19, 25]. Finally, in [16], GeoSR is presented as a way of measuring the semantic relatedness of Wikipedia articles based on their geographic context, allowing users to explore information in Wikipedia that is relevant to a particular location. In [41], one would like to discover points of interests based on geotagged photos by applying a form of spectral clustering. The problem with this approach is that there is no unified way for determining the appropriate parameters for the clustering algorithm. For that purpose, a self-tuning clustering approach is proposed.

To conclude the discussion of related work, we describe a number of techniques that also treat the problem of extracting knowledge about toponyms from Flickr, but for the goal of learning geographic knowledge per se, e.g. as a method of enriching existing gazetteers. In our approach, in addition to toponyms, various other types of tags may provide useful evidence. For example, the tag "pepsi" has no relevance when compiling or enriching gazetteers, but, since it will occur more frequently in some countries or states than in others, it may be helpful to disambiguate the meaning of other terms.

Geotagged photos are useful from a geographic perspective, to better understand how people refer to places, and overcome the limitations and/or costs of existing mapping techniques [12]. For instance, by analyzing the tags of georeferenced photos, [17] found that the city toponym was by far the most essential reference type for specific locations. Moreover, evidence is provided suggesting

[6] `http://suggestify.appspot.com/`

[7] `http://www.flickr.com/map/` shows that around 178M photos are geotagged of over 6.97 billion photos (`http://www.flickr.com/explore`) on Flickr. Accessed on March 14th, 2012.

[8] `http://googleblog.blogspot.com/2010/02/refine-your-searches-by-location.html`

that the average user has a rather distinct idea of specific places, their location and extent. Despite this tagging behaviour, the conclusion was that the data available in the Flickr database meets the requirements to generate spatial footprints at a sub-city level. Finding such footprints for non-administrative regions (i.e. regions without officially defined boundaries) using georeferenced resources has also been adressed in [31] and [39]. Another problem of interest is the automated discovery of which names (or tags) correspond to places. Especially for vernacular place names, which typically do not appear in gazetteers, collaborative tagging-based systems may be a rich source of information. In [28], methods based on burst-analysis are proposed for extracting place names from Flickr. Finally, note that to some extent, ontologies, and in particular ontologies of places may be derived from Flickr tags [30]. The approach differs substantially from the one presented in this work, as the authors do not use geographic coordinates for deriving the ontologies; these are induced from the Flickr tags vocabulary using a subsumption-based model.

3. Georeferencing framework

3.1. Overview

In this paper we present our approach to georeferencing resources from the Web purely based on textual meta-data. Given an unseen resource x described by a certain set of tags $\mathcal{T}$, we estimate a location based on the information contained in $\mathcal{T}$. In particular, we consider the scenario of estimating the location (i.e. in actual latitude/longitude coordinates) of Flickr photos, based on the tags associated with them. This approach is purely text based and no visual or other features are used in the process, although existing approaches described in literature do leverage these features, as described in Section 2.

A common approach to georeferencing is by resolving toponyms (place names) in the given text with the help of gazetteers or named entity recognition (NER). Although this may seem straightforward, it is complicated in practice due to the ambiguity of toponyms. For full-text documents, named entity taggers can be used to detect the words in a phrase that represent place names, while their coordinates can be resolved from a gazetteer. In the case of Flickr tags however, linguistic context and capitalization is missing, hence heuristics need to be used to determine whether names such as "turkey" or "nice" refer to places or to the common words in English.

To avoid explicitly disambiguating tags, we interpret the problem of georeferencing as a classification problem, by partitioning the locations on Earth into a finite number of areas, of which the most likely area for a given resource, represented as its set of tags $\mathcal{T}$, is determined. This method avoids seeking specifically for toponyms and the need of any form of (explicit) disambiguation. A first drawback, however, is that the result is an *area*, consisting of multiple photos and their locations, rather than a single pair of coordinates. Another drawback is that the partitioning of the training data into a finite set of areas superimposes a certain factor of scale to the results: when the partitioning results in a relatively small number of areas, say 500, they are likely to cover a

larger area of the world's surface. Depending on the textual information available, such a partitioning can be too coarse for one resource whereas it is too fine-grained for another resource. Take the following example: consider a photo with only one tag *elbulli*, referring to a restaurant in Spain. It is very unlikely that starting from 500 areas, one would be able to pinpoint the location of the restaurant within 1 kilometer of its actual location. On the other hand, for a photo annotated with the tags *germany, europe*, one would rather think of a larger area consisting of some of the 500 areas, actually requiring a coarser scale for this kind of resource. There clearly is no single scale that will perform best for all photos we would like to georeference. In our approach we present, in Section 3.8, two different methods for converting the resulting area into a pair of coordinates, resolving the first issue. The similarity based area refinement we propose addresses the second issue in particular. It allows using a coarser scale while still being able to accurately estimate locations by finding similar items within this coarse clustering.

The general architecture of the georeferencing framework we propose is outlined below:

1. Starting from a (preprocessed) geotagged training set, i.e. a dataset that contains the *true location* of the resources (where *true location* is to be considered as the location provided by the owner of the resource), a clustering algorithm is applied to cluster the locations of the resources into a finite set of disjoint areas $\mathcal{A}$.
2. Next, by applying feature selection, a vocabulary V consisting of discriminative tags is compiled, i.e. tags that are likely to be indicative of geographic location.
3. In a subsequent step, we train a language model. Given a unseen resource x, identified by its set of tags $\mathcal{T}$ after feature selection, a classifier will rank the areas $\mathcal{A}$ at a given scale and determine the area a that is considered to be the most likely area to contain the resource x.
4. To convert this area a into an actual location estimate, we search training items contained in this area that are most similar based on their tags. The location of these training items is then used to derive a location estimate.

We now discuss each of these steps in more detail.

3.2. Data preprocessing

The training sets we use consist of meta-data from Flickr photos. For each photo that is uploaded to its website, Flickr maintains several types of meta-data, which can be obtained via a publicly available API. In this paper, three types of meta-data will be relevant: descriptive tags that have been provided by the photo owners, the user's location (as provided by the user in her profile as free text, e.g. "Ghent, Belgium"), and information about where the photos were taken. The location information includes a geographical coordinate (latitude and longitude), and information about the accuracy of the location, encoded as a number between 1 (world-level) and 16 (street-level). Starting from a raw dataset, a number of preliminary filtering steps are carried out on this data:

1. Photos that do not contain any tags or have invalid coordinates are removed from the collection.
2. In order to retain only those photos that provide meaningful information w.r.t. within city or sub-city scale location, only photos whose location accuracy is at least 12 (viz. city level accuracy) are retained.
3. Users on Flickr can upload content in bulk, i.e. uploading multiple photos with the same information at once. In order to reduce the impact of these bulk uploads, as pointed out in [32], for photos containing the same upload date, an identical tag set and the same coordinates, only a single instance is retained.

The photos that remains after these filtering steps are used for obtaining clusters of locations, and for estimating language models.

3.3. Clustering the training data

In order to interpret the problem of georeferencing resources as a classification problem, we cluster the locations of the training data into sets of disjoint areas $\mathcal{A}$ over which language models can be trained.

Different approaches have been described in literature. In [6], a mean shift procedure is used to find highly photographed locations based on the density of photos. The authors found that this procedure was effective in determining these places at different scales (a metropolitan scale of 100 km and a landmark-level scale of 100 m). In contrast to most clustering approaches, mean shift does not require the number of clusters to be predetermined, but rather relies on a scale parameter to choose the number of clusters implicitly. In [32] a fixed grid overlay is placed over to the Earth. In this work, the authors considered varying grid sizes (and thus scales) comparing to location cells of roughly 1, 5, 10, 50 and 100 kilometers long over their sides. In [19], k-means clustering is used to identify famous locations in collections of geo-tagged photos from Flickr. In our previous work [36] we also used k-medoids (partitioning around medoids) clustering to obtain areas of interest. An alternative to clustering would be to use boundaries of administrative divisions such as cities, provinces, and countries. However, such boundaries are not freely available for every country, and usually no information about areas at the sub-city scale is available.

In what follows, we provide an overview of a number of techniques for obtaining a clustered representation of locations. An experimental comparison of these techniques will be provided in Section 4.2.

3.3.1. k-medoids clustering

Partitioning Around Medoids (PAM) or k-medoids is a clustering technique closely related to the well-known k-means algorithm; the algorithm partitions the data into groups of data points while the objective is to minimize the squared error, which is the sum of the distances between each individual point in a cluster and the cluster center (the medoid). The k-medoids algorithm is more robust to noise and outliers than k-means. Distances are calculated using the geodesic (great-circle) distance measure. The algorithm is an iterative process.

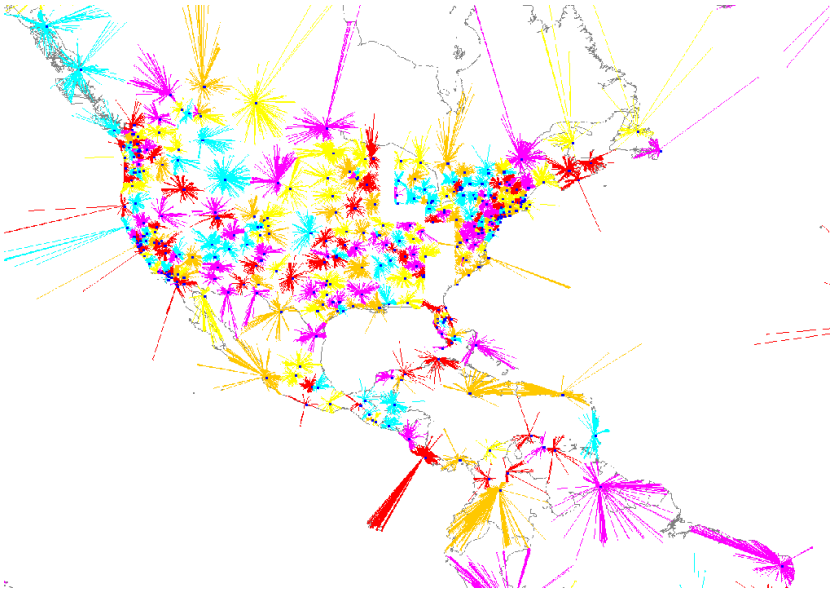


Figure 1: Sample clustering of a part of the main training set using Partition Around Medoids, $k = 1000$.

Also, increasing the number of data points results in a quadratically increasing computing cost ($O(n^2)$). We therefore apply sampling during the optimization of each individual cluster. Per cluster, a maximum of 512 data points are swapped with the medoid point m in every iteration. An example clustering of our main training set using this algorithm ($k = 1000$) is shown in Figure 1. As can be seen, metropolitan areas on both the Northeast and the West coast of the US are covered by a large number of smaller clusters, in contrast to little clusters covering large parts of northern South-America. This shows that the granularity of the clusters is based on the amount of information available in these regions.

3.3.2. Grid based clustering

A second possibility is to use a grid. Intuitively, the idea is to lay a grid of square cells over the surface of the Earth. This clustering method is straightforward and computationally inexpensive ($O(n)$). A single run over all data points is sufficient to assign them to their corresponding cluster based on the geographical coordinates of the points. When clustering the data, only cells that actually contain at least one image are considered as a cluster.

Note that, when a cell size of 1 degree in latitude and longitude is considered for each of the sides of the grid cells, this roughly corresponds to a side of 111 km in latitude and 111 km in longitude near the equator. However, the length of the longitude side converges to 0 km at the geographic poles, making it impossible to map equally sized cells when using only one parameter value to

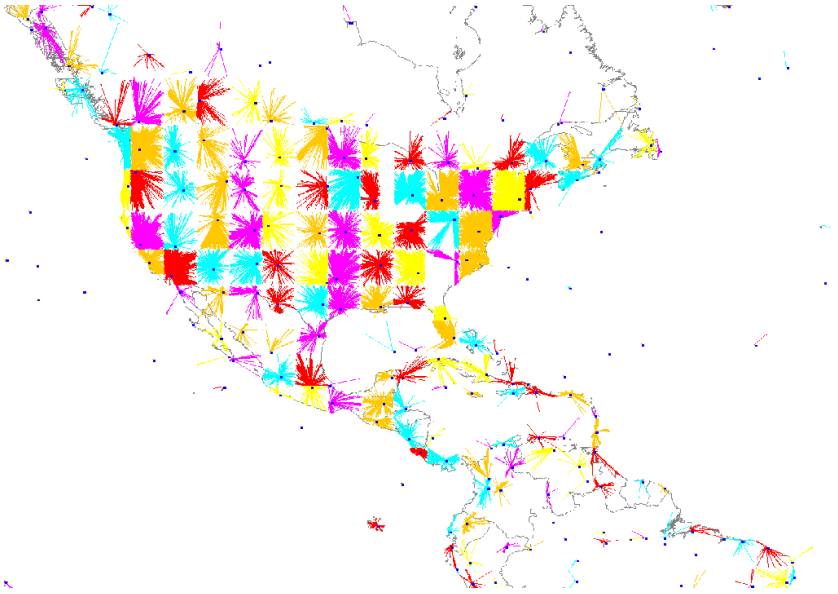


Figure 2: Sample clustering of a part of the main training set using the grid clustering approach. The side of each cell are 4.375 degrees latitude and longitude, resulting in 1001 clusters.

simultaneously define the length of both sides of the grid cells.

An example clustering of our main training set using this algorithm is shown in Figure 2. In this example, grid cells are considered using a cell size of 4.375 degrees of latitude and longitude, as this value resulted in a configuration of 1001 clusters, facilitating a comparison with Figure 1.

3.3.3. Mean shift clustering

A third and final clustering algorithm we discuss is mean shift clustering [5]. As opposed to k-medoids and grid-based clustering, which require specifying the desired number of clusters beforehand, mean shift clustering requires a parameter h that is considered the *scale of observation*. The number of resulting clusters emerges from the choice of this scale factor. Mean shift clustering is again an iterative process.

For the approach outlined in this paper, we need clusterings at different levels of granularity. The reason is that depending on the nature of the training data, coarser or finer grained clusterings will lead to an optimal performance. Initial experiments have revealed, however, that changing the scale parameter does not substantially reduce the overall number of clusters. Figure 3 illustrates this: in this example, there exist a number of small clusters located close to the West and East coasts of Northern America. These clusters are outside the influence range (defined by the scale parameter h) of other clusters. One possible solution could be to increase the scale parameter, but due to these isolated clusters

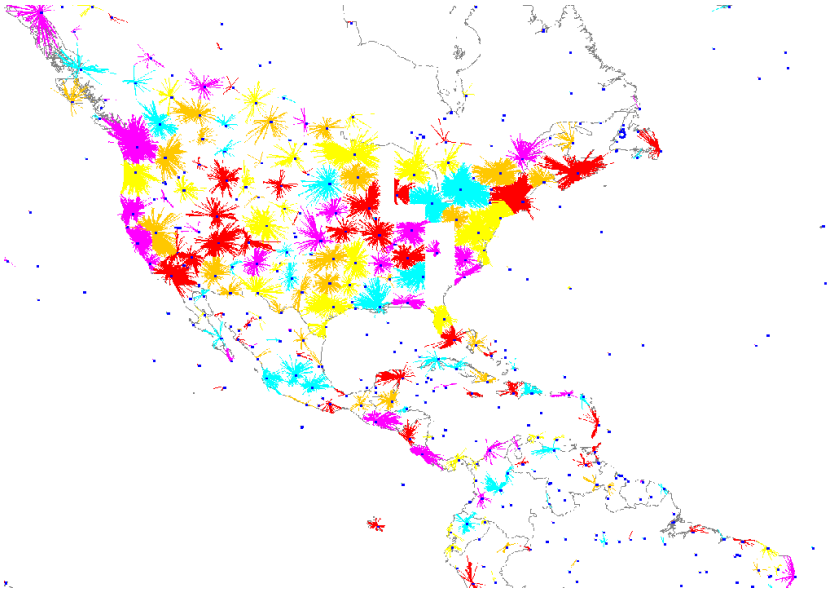


Figure 3: Sample clustering of a part of the main training set using the mean shift algorithm, $h = 150$, resulting in 2349 clusters in total.

this parameter needs to be increased substantially, again resulting in a coarse clustering.

To cope with this effect, we consider a variant of the mean shift algorithm. Once the mean shift procedure finishes producing a set of clusters $\mathcal{A}$, the following additional steps are taken:

Initialize a set of data points $\mathcal{P}$ to the empty set
for each cluster a in the set of clusters $\mathcal{A}$ **do**
  if $|a| < t$ **then**
    Add all of the data points p of a to $\mathcal{P}$
    Remove a from $\mathcal{A}$
  end if
end for
for each data point p in $\mathcal{P}$ **do**
  Assign p to the closest cluster a in $\mathcal{A}$ where closest is defined as the minimum geodesic distance between p and the medoid of a
end for

In order to avoid introducing additional parameters we keep the value of t fixed at 10 throughout the experiments; we thus only use the scale parameter h to change the number of clusters. Figure 4 illustrates the clustering obtained when merging smaller clusters with their closest neighbors.

The difference between Figures 3 and 4 is clear: the small clusters close to the coasts of the North American contintent are merged with other clusters. In

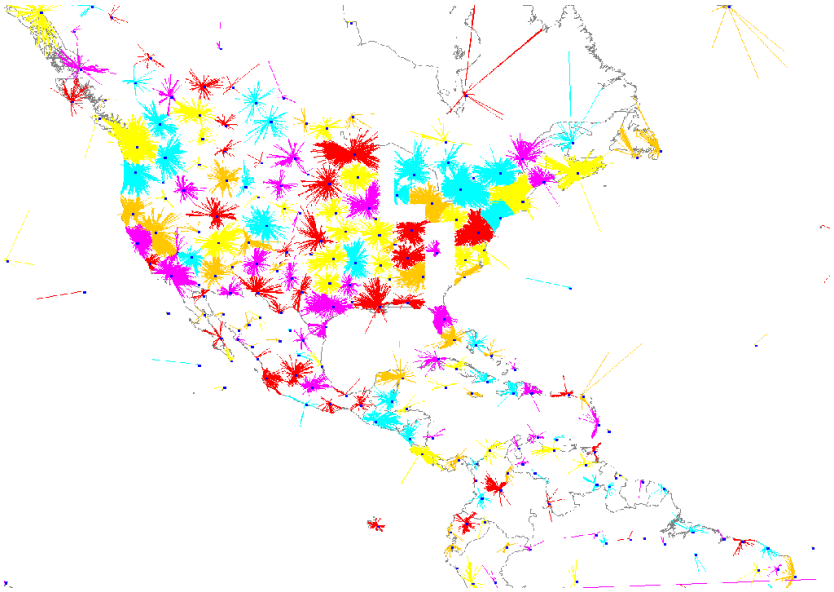


Figure 4: Sample clustering of a part of the main training set using the mean shift algorithm with merges, $h = 150$, $t = 10$, resulting in 965 clusters in total.

general, most of the isolated clusters are merged: of the 2349 original clusters the standard mean shift algorithm produced, 1384 clusters with less than 10 photos were merged with with their nearest neighbours. Of these 1384 clusters, more than 1100 clusters contained even less than 5 photos. It is important for our approach that each cluster represents a certain minimal amount of information if one wants to train reliable language models based on that information.

3.4. Feature selection

In order to train a language model for a specific scale, a set of tags (vocabulary V) is needed, consisting of tags that are likely to be indicative for the geographic location. A comparative study on feature selection techniques used in text classification in general can be found in [42]. To the best of our knowledge, no similar comparison has been carried out in literature focused on the effect of different feature selection approaches in georeferencing. These six feature selection methods will be evaluated in Section 4.3.

3.4.1. χ^2

Let $\mathcal{A}$ be the set of areas that is obtained after clustering the data into k clusters. Then for each area a in $\mathcal{A}$ and each tag t assigned to photos in a, the χ^2 statistic is given by:

$$\chi^2(a,t) = \frac{(O_{ta} - E_{ta})^2}{E_{ta}} + \frac{(O_{t\bar{a}} - E_{t\bar{a}})^2}{E_{t\bar{a}}} + \frac{(O_{\bar{t}a} - E_{\bar{t}a})^2}{E_{\bar{t}a}} + \frac{(O_{\bar{t}\bar{a}} - E_{\bar{t}\bar{a}})^2}{E_{\bar{t}\bar{a}}} \quad (1)$$

where O_{ta} is the number of photos in area a where tag t occurs, $O_{t\overline{a}}$ is the number of photos outside area a where tag t occurs, $O_{\overline{t}a}$ is the number of photos in area a where tag t does not occur, and $O_{\overline{t}\overline{a}}$ is the number of photos outside area a where tag t does not occur. Furthermore, E_{ta} is the number of occurrences of tag t in photos of area a that could be expected if occurrence of t were independent of the location in area a, i.e. $E_{ta} = N \cdot P(t) \cdot P(a)$ with N the total number of photos, $P(t)$ the probability that a photo contains tag t and $P(a)$ the probability that a photo is located in area a, the latter two probabilities being estimated using maximum likelihood:

$$P(t) = \frac{\sum_{a \in \mathcal{A}} O_{ta}}{\sum_{t' \in V} \sum_{a \in \mathcal{A}} O_{ta}} \tag{2}$$

$$P(a) = \frac{|a|}{N} \tag{3}$$

Similarly, $E_{t\overline{a}} = N \cdot P(t) \cdot (1 - P(a))$, $E_{\overline{t}a} = N \cdot (1 - P(t)) \cdot P(a)$ and $E_{\overline{t}\overline{a}} = N \cdot (1 - P(t)) \cdot (1 - P(a))$.

The most relevant features for a given area can then be selected by choosing the features with the highest value for the χ^2 statistic. To select a vocabulary V containing the v most discriminative features, we need to aggregate the rankings obtained for every area a into a single ranking. This is accomplished by first selecting the best tag from each of the rankings, then the tags at position 2, etc.

3.4.2. Maximum χ^2

Maximum χ^2 (max χ^2) is similar to χ^2 except that when constructing the overall ranking, each tag is ranked according to its highest χ^2 value over all areas a. In other words, not only the ranking imposed by the χ^2 statistic plays a role here, but also the actual value. In principle, even the highest ranked tag for a given area may not be selected if its χ^2 value is too low (e.g. because the area corresponds to a small cluster where none of the photos bears any tags that are descriptive of the location).

3.4.3. Log-Likelihood

As an alternative to the χ^2 statistic, we consider Dunning's *log-likelihood* statistic [8]. For each term t and area $a \in \mathcal{A}$, the log-likelihood is given by:

$$\begin{aligned} G^2(a,t) = \quad & 2(O_{ta} \log O_{ta} + O_{t\overline{a}} \log O_{t\overline{a}} + O_{\overline{t}a} \log O_{\overline{t}a} + O_{\overline{t}\overline{a}} \log O_{\overline{t}\overline{a}} \\ & + N \log N \\ & - (O_{ta} + O_{t\overline{a}}) log(O_{ta} + O_{t\overline{a}}) - (O_{ta} + O_{\overline{t}a}) log(O_{ta} + O_{\overline{t}a}) \\ & - (O_{t\overline{a}} + O_{\overline{t}\overline{a}}) log(O_{t\overline{a}} + O_{\overline{t}\overline{a}}) - (O_{\overline{t}a} + O_{\overline{t}\overline{a}}) log(O_{\overline{t}a} + O_{\overline{t}\overline{a}})) \end{aligned} \tag{4}$$

where $O_{ta}, O_{t\overline{a}}, O_{\overline{t}a}$ and $O_{\overline{t}\overline{a}}$ are defined as in Section 3.4.1 and N is the total number of photos in the training data. Similarly, the most relevant features for a given area can then be selected by choosing the features with the highest value for the G^2 statistic. In order to obtain the vocabulary V, the same method as described in Section 3.4.1 is used.

3.4.4. Information Gain

Whereas the χ^2 based methods are rooted in statistics, information gain uses information theory to select informative terms. It measures the change in entropy when learning about the presence or absence of the tag. The information gain of the tag t is defined as:

$$\begin{aligned} G(t) = \quad & -\sum_{a \in \mathcal{A}} P(a) \log P(a) \\ & +P(t) \sum_{a \in \mathcal{A}} P(a|t) \log P(a|t) \\ & +P(\bar{t}) \sum_{a \in \mathcal{A}} P(a|\bar{t}) \log P(a|\bar{t}) \end{aligned}$$

with $P(a)$, the probability for area a, is estimated by Equation (3). Similarly, the probability for $P(t)$ is estimated by dividing the number of occurrences of the tag by the total number of tag occurrences (Equation (2)). $P(a|t)$ is estimated as the number of tag occurrences of tag t in area a, divided by the total number of occurrences of tag t:

$$P(a|t) = \frac{O_{ta}}{\sum_{a' \in \mathcal{A}} O_{ta}}$$

$P(\bar{t})$ and $P(a|\bar{t})$ are defined likewise, using the number of occurrences of all tags but t.

Note that information gain immediately produces a single ranking, in contrast to the χ^2 statistic, which produces a ranking per area. Hence, we can simply choose the vocabulary V by selecting the v tags with the highest information gain.

3.4.5. Most frequently used (MFU)

A particularly simple term selection technique that is sometimes used consists of selecting the terms that occur in the largest number of documents. Despite the simplicity of the method, it often performs remarkably well in practice [42].

3.4.6. Geographical spread (geospread)

As a sixth and final feature selection method, we describe a *geographic spread filtering* feature selection method presented in [13] and applied in [14]. In this work, a score is proposed that captures to what extent the occurrences of a tag are clustered around a small number of locations. The geographical spread score is calculated as follows:

Place a grid over the world map with each cell having sides of 1 degree latitude and longitude
for each unique tag t in the training data **do**
  for each i, j **do**
    For cell $c_{i,j}$, determine $|t_{i,j}|$, the number of training items containing the tag t
    if $|t_{i,j}| > 0$ **then**
      for each $c_{i',j'} \in \{c_{i-1,j}, c_{i+1,j}, c_{i,j-1}, c_{i,j+1}\}$, the neighbouring cells of $c_{i,j}$, **do**

Determine $|t_{i',j'}|$
if $|t_{i',j'}| > 0$ and $c_{i,j}$ and $c_{i',j'}$ are not already connected **then**
Connect cells $c_{i,j}$ and $c_{i',j'}$
end if
end for
end if
end for
$count$ = number of remaining connected components
$score(t) = count/max(|t_{i,j}|)$, with $max(|t_{i,j}|)$ over all original cells $c_{i,j}$.
end for

In the algorithm, merging of neighbouring cells is necessary in order to avoid penalizing geographic terms that cover a wider area.

Given the definition of the geographical spread score, a clear distinction should come to light between terms that are quite location bound on the one hand, and very general tags on the other hand. The smaller the resulting score for a tag t, the more specific its geographic scope and thus the more it is coupled to a specific location. We will refer to this method as *geospread*.

3.4.7. Qualitative evaluation of the feature selection methods

Table 1 presents an overview of the 10 highest ranking features according to each of the term selection algorithms discussed in this section. The features selected by χ^2, *max* χ^2 and *log-likelihood* depend on a specific clustering of the training data (in this case, $k = 2500$), while the other methods construct a ranking over all the training data, independent a specific clustering.

Considering the features selected by χ^2, we observe that the list only consists of toponyms: a country, cities and regions are mentioned, as well as a name of a Russian conference center: *igromir*. Note that the top ranking tags in this example are a random sample of the best ranking tags for each of the 2500 areas used for creating the ranking. However, when analyzing the first 2500 terms (and beyond) of the entire feature ranking, the behaviour witnessed persists.

The *max* χ^2 method returns a seemingly similar ranking, but this time, the geographical entities are, all but two (*bahiabrazil* and *bolodecasamento*), referring to islands. The top ranking tag, *bolodecasamento*, is in fact non-geographical related and represents the Portuguese concept of a "wedding cake". This term immediately propagated to the top of the ranking because it occurred only once in the training data, within a given area containing only a single photo with this tag. By chance, the regular χ^2 method could have also ranked this term at the top, instead of at position 2372, if it started processing the areas with the area specifically containing this tag. This behaviour can be explained by the use of the χ^2 measure (1) in general, which awards such a very specific case with a maximum score.

In general, the ranking favors tags that frequently occur in a single cluster (cfr. the islands) and rarely outside it over discriminative terms for certain areas that also occur elsewhere: e.g. *andorra*, ranked in position 347, has 314 occurrences in a single cluster, whereas *canada* is ranked in position 67 463 while it occurs 29 141 times, albeit spread out over many clusters.

Table 1: Overview of the top 10 terms according to different feature selection methods applied to the training data.

	χ^2	max χ^2	Log-likelihood
1	gijón	bolodecasamento	roma
2	lhaviyani	seychelles	hsinchu
3	montauk	vanuatu	medellin
4	wolfsburg	elhierro	korea
5	igromir	bahiabrazil	nara
6	saintebaume	lanyu	valdaosta
7	hartford	galapagos	alps
8	bulgaria	isleofman	nef
9	rochester	bermuda	snowymountains
10	mendoza	madagascar	stalbans
	Information Gain	Most frequently used	Geospread
1	california	geotagged	kaohsing
2	australia	2008	haninge
3	france	2009	greatermanchester
4	italy	california	hsinchu
5	japan	2007	antwerpen
6	canada	nikon	nikone3700
7	germany	beach	algarve
8	scotland	nature	sinpu
9	spain	canon	hsinpu
10	taiwan	travel	oxfordshire

The list of features obtained by *log-likelihood* contains words that describe admimistrative entities such as cities or countries, while the tags *alps*, *valdaosta* and *snowymountains* describe mountains or valleys. The tag *nef* refers to the raw file format for photos taken with Nikon cameras. It is included because it occured 295 times in the training data, of which 210 occurrences are by the same user in the same area. Methods to combat such problems would be to only use tags that have been used by a sufficiently large number of users, or only consider one occurrence per tag per user, for feature selecting purposes. In practice, however, such methods tend to worsen results in the Placing Task setting, as training and test data may contain resources by the same user. In such a case user-specific tags are often helpful.

Inspecting the table further, we observe that information gain (IG), provides a list of country names, whereas "most frequently used" (MFU) returns a list of tags that rarely contain any reference to a place in particular (except for *california*). However, while tags like *beach* or *nature* are not toponyms, they might help in disambiguating cases where one needs to decide if a photo was taken near the sea or in the city.

Finally, the geospread measure presents a list of terms that it considers to have a very specific spatial scope. All but one tag in the list can indeed be

easily located on a map. After analyzing the training data, the occurrence of *nikone3700* at position 6 out of more 1.13M in the list (details of the dataset can be found in Section 4.1) can be explained by the fact that a single user tagged 443 photos with the model of his camera in the same surroundings (the *greatermanchester* area, a tag also occuring in the top 3). As the geospread measure favors terms with a small geographical footprint, this term popped up as it can be tied to a very small region.

A quantitative evaluation of the different methods presented here follows in Section 4.3.

3.5. Language models

Given a previously unseen image x, we now attempt to determine in which area x was most likely taken. In this paper, we use a (multinomial) Naive Bayes classifier, which has the advantage of being simple, efficient, and robust. Initial results in [32] have shown good results for this classifier. Specifically, we assume that an image x is represented as its set of tags $\mathcal{T}$. Using Bayes' rule, we know that the probability $P(a|x)$ that image x was taken in area a is given by

$$P(a|x) = \frac{P(a) \cdot P(x|a)}{P(x)}$$

Using the assumption that the probability $P(x)$ of observing the tags associated with image x is fixed among all areas a, we find

$$P(a|x) \propto P(a) \cdot P(x|a)$$

Characteristic of Naive Bayes is the assumption that all features are independent. Translated to our context, this means that the presence of a given tag does not influence the presence or absence of other tags. Writing $P(t|a)$ for the probability of a tag t being associated to an image in area a, we find

$$P(a|x) \propto P(a) \cdot \prod_{t \in \mathcal{T}} P(t|a) \tag{5}$$

After moving to log-space to avoid numerical underflow, this leads to identifying the area a^* where x was most likely taken by:

$$a^* = \arg\max_{a \in \mathcal{A}} (\log P(a) + \sum_{t \in \mathcal{T}} \log P(t|a))$$

In this final equation, the prior probability $P(a)$ and the probability $P(t|a)$ remain to be estimated. In general, the maximum likelihood estimation can be used to obtain a good estimate of the prior probability but alternative approaches that include available meta-data are also possible, as we will show in Section 3.6. When estimating $P(t|a)$, a form of smoothing is needed to avoid a zero probability when a certain tag t does not occur in area a. We discuss different forms of smoothing in Section 3.7.

3.6. Estimating the prior probability

In this section, we discuss four possible ways of estimating the prior probability for the language models. An experimental comparison of these methods will be provided in Section 4.4.1.

3.6.1. Maximum likelihood and uniform prior

A common way of estimating the prior probability for the language models is using the maximum likelihood estimation:

$$P(a) = \frac{|a|}{N} \tag{6}$$

in which $|a|$ represents the number of training items contained in area a, and N represents the total number of training items as before.

A second, rather simple, way of estimating the prior probability might be to assign a uniform probability to all areas in $\mathcal{A}$.

$$P(a) = \frac{1}{|\mathcal{A}|} \tag{7}$$

One could also think of incorporating information from recent uploads from the same user, but this was not considered in this paper.

3.6.2. Using home location information from the user

The owner of a Flickr photo can provide a textual description of his home location in his profile, which can be retrieved using the public API. Most of the photo owners on Flickr have actually provided such a description, although it is not always precise or accurate. For example, in the best cases, the description looks like "San Francisco, CA, United States" or "Cava de' Tirreni, Italia", pointing unambiguously to a known city whereas in the worst cases, the users describe their home location as "to infinity and beyond" or "homeless, US".

When the location information is present, we geocode this information (as provided by the user) using the Google Geocoding API[9] to convert the textual description to coordinates by extracting the *location* information returned from the Google API. For the example of "Cava de' Tirreni, Italia", this returns

```
"location" : {
   "lat" : 40.70205550,
   "lng" : 14.7065740
},
```

which indeed corresponds to the center of the town.

Although this example yields an interesting source of location information, this is however not the case for a large part of the descriptions. It might be clear that informal descriptions provided by the users present the Geocoding

[9] `http://code.google.com/apis/maps/documentation/geocoding/`

API with an unresolvable task. As will be explained in Section 4.1, in the case of the datasets considered in our experiments, the home location could be geocoded for 65% to 85% of the photos.

For an unseen resource x, when available, the information about the home location of the owner, $loc_{home}(x)$, can be used to estimate the prior probability as follows:

$$P(a) \propto (\frac{1}{d(m_a, loc_{home}(x)) + 0.001})^w \tag{8}$$

where $m_a(x)$ is the medoid of the area a and where $d(x, y)$ is the geodesic distance between the locations of points x and y. The parameter w allows to vary the influence of the home location on the prior probability. In the denominator a fixed value of 0.001 is introduced to avoid division by zero in the case that $loc_{home}(x)$ and m_a coincide. Using the home location of a user in this way corresponds to an assumption that all things being equal, locations within a reasonable distance from a user's home are more likely than locations at the other side of the world, even though we cannot exclude the latter case altogether.

3.6.3. Gaussian mixture models

Another way of using the home location when estimating the prior probability is to use a Gaussian mixture model (GMM) [29]. A Gaussian mixture model is a parametric probability density function represented as a weighted sum of Gaussian densities:

$$P(x|\lambda) = \sum_{i=1}^{M} w_i g(x|\mu_i, \Sigma_i) \tag{9}$$

$$\lambda = \bigcup_{i=1}^{M} \{w_i, \mu_i, \Sigma_i\}$$

where $x \in \mathbb{R}$ represents some numerical feature, $w_i = 1, \ldots, M$ are the mixture weights and $g(x|\mu_i, \Sigma_i), i = 1, \ldots, M$ are the component Gaussian densities. The mixture weights are required to sum up to 1: $\sum_{i=1}^{M} w_i = 1$. In our case, the Gaussian mixture model is used to estimate the prior probability of an area a, given the distance between a and the home location of the user. The feature x then corresponds to a distance.

The underlying idea is that there may be several types of relations between the home location of the user and the location of the photo:

1. With a certain probability w_1, the photo is taken nearby the house of the owner, in which case the prior probability of an area quickly decreases as the distance from the home location of the user increases.
2. With a probability w_2, the photo was taken on a day trip by the user.
3. With a probability w_3, the photo was taken on a holiday.

Using a Gaussian mixture model, we can jointly describe these scenarios, using the probabilities w_1, w_2 and w_3 as the mixture weights, and using one Gaussian to describe each scenario. Of course, neither the mixture weights nor the parameters of the Gaussians are known a priori. However, they can be estimated from the training data using the expectation-maximization (EM) procedure [29].

3.7. Smoothing methods

To avoid a zero probability when an unseen resource x contains a tag that does not occur with any of the photos from area a in the training data, smoothing is needed when estimating $p(t|a)$ in (5). Let O_{ta} be the occurrence count of tag t in area a. The total tag occurrence count O_a of area a is then defined as follows:

$$|O_a| = \sum_{t \in V} O_{ta} \tag{10}$$

where V is the vocabulary that was obtained after feature selection, as explained in Section 3.4.

One possible smoothing method is Bayesian smoothing with Dirichlet priors, in which case we have ($\mu > 0$):

$$P(t|a) = \frac{O_{ta} + \mu \, P(t|V)}{|O_a| + \mu} \tag{11}$$

where the probabilistic model of the vocabulary $P(t|V)$ is defined using maximum likelihood:

$$P(t|V) = \frac{\sum_{a \in \mathcal{A}} O_{ta}}{\sum_{t' \in V} \sum_{a \in \mathcal{A}} O_{ta}} \tag{12}$$

Another possibility is to use Jelinek-Mercer smoothing, in which case (11) becomes ($\lambda \in [0, 1]$):

$$P(t|a) = \lambda \frac{O_{ta}}{|O_a|} + (1 - \lambda) \, P(t|V) \tag{13}$$

with $P(t|V)$ defined as in (12). For more details on these smoothing methods for language models, we refer to [43]. The performance of both smoothing methods will be experimentally assessed in Section 4.4.1.

3.8. Finding a location within the chosen area

The previous steps result in the selection of an area a among those in $\mathcal{A}$ where the photo (or video) x has been taken (recorded). The final step that remains is converting this area a into an actual location, i.e. resolve the latitude and longitude coordinates for the resource x. We discuss two ways of accomplishing this: by determining the medoid of the area a, and by performing similarity search. Both methods are evaluated in Section 4.2.

3.8.1. Medoid based location estimation

The most straightforward way of converting an area a into actual coordinates is to choose the location of the medoid m_a, defined as:

$$m_a = \arg\min_{x \in \mathcal{A}_k} \sum_{y \in \mathcal{A}_k} d(x, y) \tag{14}$$

where $d(x, y)$ is the geodesic distance between the locations of photos x and y.

Clearly, the location estimates that are obtained in this way will mainly be useful when a sufficiently fine-grained clustering is used.

3.8.2. Similarity based location estimation

As an alternative, we explore the idea of using the location of the most similar resources from the training set that are known to be located in the chosen area a. Specifically, let $y_1, ..., y_n$ be the n most similar photos from our training set. We then propose to estimate the location of x as a weighted center-of-gravity of the locations of $y_1, ..., y_n$:

$$loc(x) = \frac{1}{n} \sum_{i=1}^{n} sim(x, y_i)^{\alpha} \cdot loc(y_i) \tag{15}$$

where the parameter $\alpha \in]0, +\infty[$ determines how strongly the result is influenced by the most similar photos only. The similarity $sim(x, y_i)$ between resources x and y_i was quantified using the Jaccard measure:

$$s_{jacc}(x, y) = \frac{|x \cap y|}{|x \cup y|}$$

where we identify a resource with its set of tags *without feature selection*, to make full use of all the originally associated tags. In principle, Jaccard similarity may be combined with other types of similarity, e.g. based on visual features.

In (15), locations are assumed to be represented as Cartesian (x, y, z) coordinates rather than as (lat, lon) pairs. In practice, we thus need to convert the (lat_i, lon_i) coordinates of each photo y_i to its Cartesian coordinates.

4. Experimental results

In this section, we present a ground-truth based evaluation of each of the individual components of our georeferencing framework presented before. In general, after running an experiment using a given configuration, we will obtain an estimated location for each of the test items. We then analyze the results using two metrics:

1. **Acc@X**: number of location estimates within X km of the actual location, as defined by the ground truth, divided by the total number of items in the test set. The accuracy is determined for the following values of X: 1 km, 5 km, 10 km, 50 km, 100 km, 1000 km and 10 000 km.

2. **Median error distance (MER)**: median over all test items of the distance between the estimated and the true location.

The first metric was used in the evaluation of the Placing Task initiative, and provides a detailed view on the performance of a given method. However, in most cases, we also use the second metric, as it summarizes the performance of a method as a single value. A median error distance of for example 5 km (which is equal to an $Acc@5$ of 50%), would indicate that half of the test set could be georeferenced with an error distance smaller than 5 kilometers.

The methodology of the experiments is as follows:

1. Using a baseline configuration, we will examine the performance of the different clustering approaches presented in Section 3.3. At the same time, we evaluate both area refinement approaches discussed in Section 3.8.
2. Next, using the best outcome of the initial experiment, we investigate the influence of the different feature selection algorithms, outlined in Section 3.4, on the results of the georeferencing use case.
3. Again adopting the feature selection method yielding the best result, we analyze the impact of applying different forms of smoothing (Section 3.7) and different ways of calculating the prior probability (Section 3.6).

At the end of this multi-step, greedy, way of experimentation we provide an overview of these different experiments and their potential improvements. Finally, in Section 4.6, we discuss the influence of adding more training data.

Before elaborating on the individual experiments, we provide a clear overview of the datasets used in this paper.

4.1. Datasets

For all experiments in this paper, the collection of test items is the same. This collection consists of the development and test data provided for the 2011 edition of the Placing Task, which is available with the Task organizers. The data consists of Flickr videos and their meta-data (which is represented in the same way as Flickr photos). Bearing in mind that some experiments need the home location of the owner of the videos, we filtered out those videos for which this information could not be retrieved. The final test set therefore contained the data for 13 390 Flickr videos.

With respect to the training data, we have used the dataset that was available to the Placing Task participants for most of the experiments carried out in this paper. This dataset constists of 3 185 343 georeferenced Flickr photos and their meta-data. As mentioned in Section 3.2, we preprocessed this dataset by removing photos with invalid coordinates, with missing tag information and items originating from a batch upload. On this dataset, no particular accuracy filtering was imposed, i.e. the accuracy level of the photos varies from 1 to 16, where 1 corresponds to accurate at world-level, 12 at city-block level and 16 at

street-level[10]. This resulted in a dataset of 2 096 712 Flickr photos covering more or less the entire world; it is referred to as the training set throughout the remainder of this paper, unless specified otherwise (viz. in the case of the experiments in Section 4.6). Figure 5 presents a geographical mapping of this dataset.

As some experiments require additional data, we crawled Flickr for data in April 2011 using the public API. The goal of the crawl was to fetch data about as many geotagged photos as possible. We were able to retrieve the meta-data of 105 118 157 photos being, at that time, over 70% of all geotagged photos. Again, we preprocessed the data obtained by removing the photos containing no valid coordinates or containing no tags, and we removed the bulk uploads. This resulted in a collection of 43 711 679 photos. Among these photos, we extracted those that reported an accuracy level of 16, which corresponds to a street level accuracy. This final step resulted in a set of 17 169 341 photos. This dataset was split into 16M and 1 169 341 photos. From the latter set, we randomly selected 10 000 photos whose owners have no other photos in the training set. This set of 10K photos is used as the *development set*, and will be used to optimize the parameters for the prior and smoothing techniques, independent of the actual test set. Of the remaining 16M photos, training sets of the first 1M, 2M, ..., 10M photos are extracted to provide the necessary training data for the experiments in Section 4.6.

Table 2 provides information on the different datasets and the number photos in each set, as well as information on the mean number of tags associated to the photos and the standard deviation of the number of tags.

4.2. Clustering and area refinement

The goal of this first experiment is to find out which clustering approach performs best and what is the optimal number of clusters, by comparing the results of the different clustering algorithms discussed in Section 3.3. At the same time, we compare both area refinement methods described in Section 3.8. The setup of this experiment is as follows:

- We use the training set consisting of 2 096 712 training items and 13 390 test items respectively.
- We cluster the training dataset into a predefined number of clusters k, varying from 500 to 20000 clusters.
- For the clustering algorithms that do not allow to fix the number of clusters beforehand (i.e. grid clustering and mean shift clustering), we set their respective parameters such that we can obtain a number of clusters that is more or less comparable to the predefined value for the PAM algorithm.

[10]For details on the Flickr accuracy values, please refer to `http://www.flickr.com/services/api/flickr.photos.search.html`

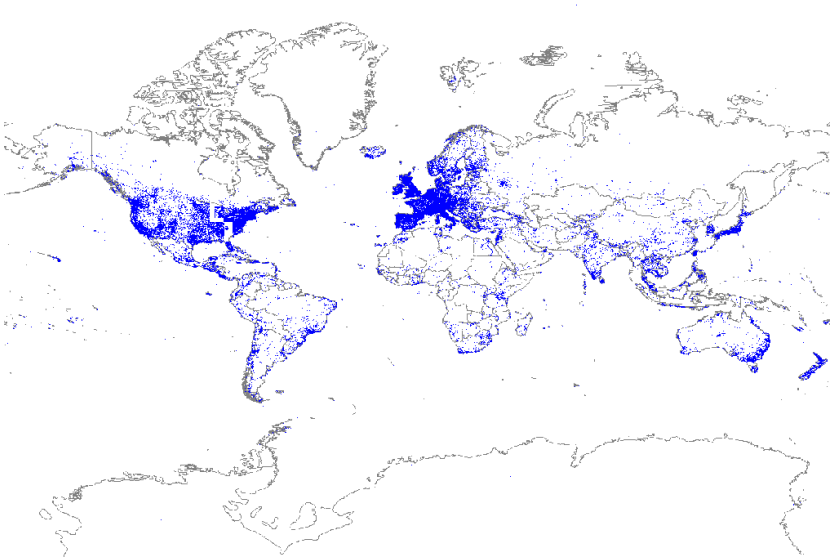
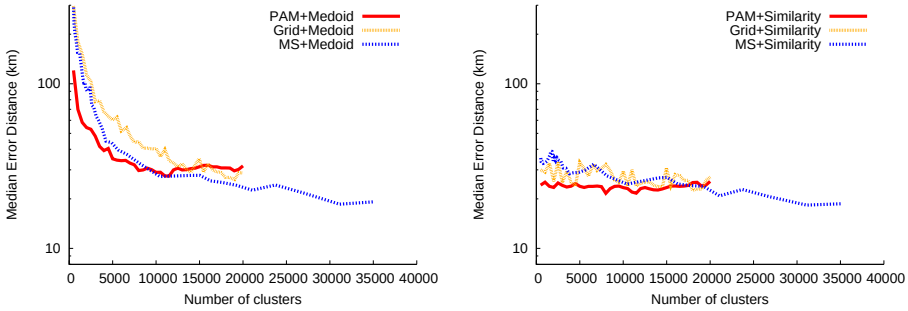


Figure 5: A plot of the photo data, after preprocessing, in the main training dataset from the Placing Task.

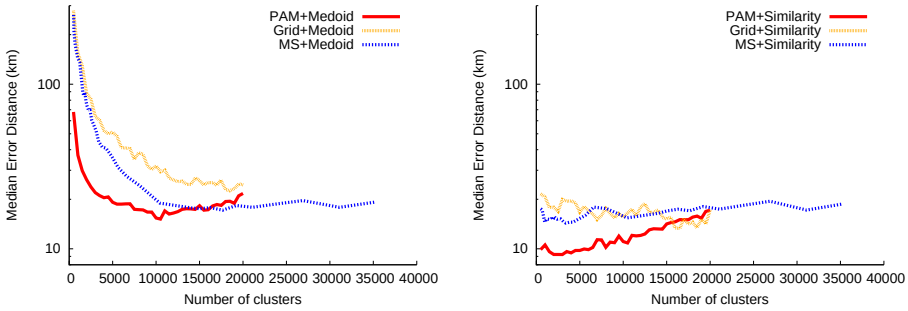
Table 2: Statistics of the considered datasets. Apart from the number of photos N in each of the datasets, the mean number of tags $\mu(|\mathcal{T}|)$ associated with each data item and the standard deviation $\sigma(|\mathcal{T}|)$ of this value are reported.

Dataset	N	$\mu(\lvert\mathcal{T}\rvert)$	$\sigma(\lvert\mathcal{T}\rvert)$	Type
General experiments				
Training set	2 096 712	7.801	7.491	photos
Test set	13 390	9.514	8.348	videos
Parameter optimization				
Development set	10 000	8.515	8.614	photos
Training experiments				
Training set 1M	1 000 000	8.745	8.463	photos
Training set 2M	2 000 000	8.746	8.462	photos
Training set 3M	3 000 000	8.747	8.456	photos
Training set 4M	4 000 000	8.747	8.457	photos
Training set 5M	5 000 000	8.748	8.461	photos
Training set 6M	6 000 000	8.749	8.463	photos
Training set 7M	7 000 000	8.749	8.465	photos
Training set 8M	8 000 000	8.750	8.464	photos
Training set 9M	9 000 000	8.750	8.465	photos
Training set 10M	10 000 000	8.751	8.466	photos



(a) Medoid based area refinement (b) Similarity based area refinement

Figure 6: Comparing the median error distance for 3 different clustering methods using a fixed number of features, $v = 45\ 000$.



(a) Medoid based area refinement (b) Similarity based area refinement

Figure 7: Comparing the resulting median error distance of 3 different clustering methods using a fixed amount of memory (16 GB).

- In order to eliminate any side-effects introduced by the choice of the feature selection method, the *most frequently used* feature selection method (as introduced in Section 3.4.5) is used. This method is independent of the underlying clustering.
- The baseline language model is applied with the maximum likelihood prior (6) and Bayesian smoothing with Dirichlet priors (11), $\mu = 1750$.

Figures 6a and 6b present the results of the experiment for a fixed number of features, $v = 45000$, using a log scale on the Y-axis. Two interesting conclusions can be drawn from this data. First, not surprisingly, using similarity search (Figure 6b) to convert an area to a precise location clearly performs better than returning the medoid of the areas (Figure 6a), especially when the number of clusters is small. Second, mean shift clustering is most effective to

reduce the median error over the test set when the number of areas used is large (both Figures 6a and 6b). One should note that the results of this experiment are somewhat misleading: when using more clusters, more memory is required. Thus, when using a smaller number of clusters, we could include more features. Therefore, in a second experiment, we keep the amount of memory used fixed and choose the maximum number of features feasible for each clustering. When looking at Figures 7a and 7b, containing the results when using 16 GB of memory and maximizing the number of features per number of clusters, we see that similarity search (Figure 7b) again outperforms the medoid based location conversion (Figure 7a). However, here PAM outperforms both other clustering algorithms substantially, and this at very low values for the number of clusters (Figure 7b). The optimal value is $k = 3000$ (and comparable results are found at $k = \{2500, 3000, \ldots, 4500\}$) with a median error distance of 10.89 km. Table 3 gives an idea of the total number of features we can include at different clustering scales.

The conclusion of this experiment is two-fold:

1. In order to convert an area to a precise location, a similarity based conversion (15) clearly outperforms a medoid based conversion.
2. In configurations that only allow a small number of features to be retained, mean shift clustering delivers the best performance. As soon as a sufficiently large number of features can be used, PAM outperforms both grid based and mean shift clustering algorithms, although we were unable to compare the algorithms in cases where a large number of clusters can be constructed using all features.

For the remainder of the paper, we will only consider PAM based clusterings combined with similarity based area refinement. To give an idea of the physical dimensions of the clusters generated by PAM, we included an overview of the (average) cluster size in kilometers (*size*) and standard deviation of the cluster size (σ) for a number of different values of k in Table 4. The average size of a cluster is defined as the sum of the distances between each datapoint and the medoid, divided by the number of datapoints.

4.3. Quantitative evaluation of the feature selection methods

In a second series of experiments, we evaluate the feature selection methods described in Section 3.4. Because of the outcome of the clustering experiment (Section 4.2), the clusterings are created using the PAM algorithm and similarity search will be used to convert the selected areas to a precise location, while the number of features is determined with respect to a fixed amount of memory (i.e. use as many features as possible for the experiment, given 16 GB of memory. For details, see Table 3). Also, as Figure 7 showed the optimal results to be obtained for a lower number of clusters, we will vary the cluster size from 500 to 10000.

Figure 8 depicts the results from this experiment. It is clear that for a large number of choices for the number of clusters, the *geospread* method outperforms all others and also results in the best performance overall, when the number

Table 3: Number of features $|V|$ that can be retained when using k clusters in the fixed memory configuration of our framework (16 GB of memory).

k	$\lvert V\rvert$	k	$\lvert V\rvert$
500	1 500 000	5500	275 000
1000	1 500 000	6000	250 000
1500	1 000 000	6500	225 000
2000	750 000	7000	200 000
2500	625 000	7500	200 000
3000	525 000	8000	200 000
3500	450 000	8500	175 000
4000	400 000	9000	175 000
4500	350 000	9500	150 000
5000	300 000	10000	150 000

Table 4: Statistics regarding the physical dimensions of clusters generated by the PAM algorithm.

k	size (km)	σ (km)
500	100.00	92.04
5000	20.76	20.21
10000	12.94	12.89
15000	9.47	9.50
20000	7.56	7.68

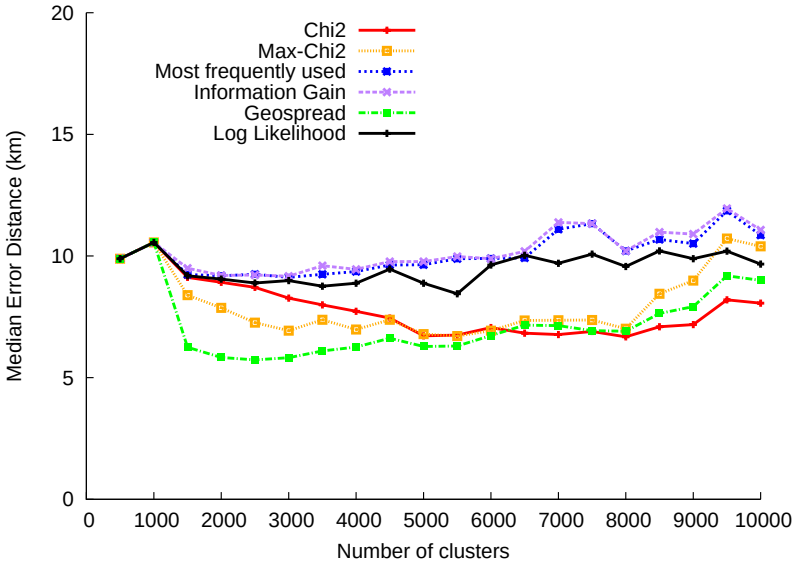


Figure 8: Median error distance over the test collection when estimating locations using different feature selection methods.

of clusters is 2500. Somewhat surprisingly, the *most frequently used* method behaves similarly to the Information Gain (IG) approach. Both perform substantially worse than the other methods. Note that all three aforementioned techniques are independent of the number of clusters k used, in constrast to χ^2, max χ^2 and *log-likelihood*. Also, χ^2 is surpassed in performance by the max χ^2 variant when the number of clusters is sufficiently small. Overall, the χ^2 based methods yield better results than *IG* or *most frequently used*. The *log-likelihood* measure mainly differs from χ^2 in the treatment of terms with only few occurrences, which leads to worse results in this scenario. The overall results deteriorate for an increasing number of areas k, while the best results, with the exception of χ^2 and *log-likelihood*, can be found around $k = 2500$.

We conclude by noticing that the *geospread* feature selection technique achieves a median error distance for the test set of 5.75 km. Applying a good feature selection technique thus improves the best results from the first experiment (9.23 km) by over 35%. Henceforth, we will apply *geospread* feature selection.

4.4. Language models

In the following experiment, we investigate two possible improvements to the baseline language modeling step. First, we investigate how different smoothing methods influence the results. In a subsequent experiment, we hope to find out which of the different implementations of the prior probability $P(a)$ outlined in Section 3.6 performs best.

4.4.1. Smoothing methods

Before we start, let us outline the configuration used for the smoothing experiment. When optimizing the parameters, the regular test set of 13 390 test items is replaced by the *development set* introduced in Section 4.1, containing 10 000 previously unseen test photos. This avoids taking advantage of information in the regular test set when determining optimal parameter values.

Figures 9 and 10 present the median error distance of the evaluation over the *development set*. When using Jelinek-Mercer smoothing, we can see that varying parameter λ only has a limited impact on the results. We also observe that for each individual clustering scale k, the optimal parameter value differs. In these results, these values are 0.6, 0.3 and 0.3 for k equal to 2500, 5000 and 7500 respectively.

The results in Figure 10 reveal that the choice of the parameter μ has a stronger influence on the performance of Dirichlet smoothing. The main conclusion that we can draw from these results is that the optimal value for μ decreases when the number of clusters increases. Indeed, when the number of clusters increases, there are fewer tag occurrences per cluster, so intuitively we need a smaller value of μ for the same amount of smoothing.

Overall, when comparing the results from the Jelinek-Mercer smoothing method and Bayesian smoothing with Dirichlet priors, we see that the results are quite similar. In the best cases, just under 7 km of median error is measured, with a slightly better result for the Bayesian smoothing with Dirichlet

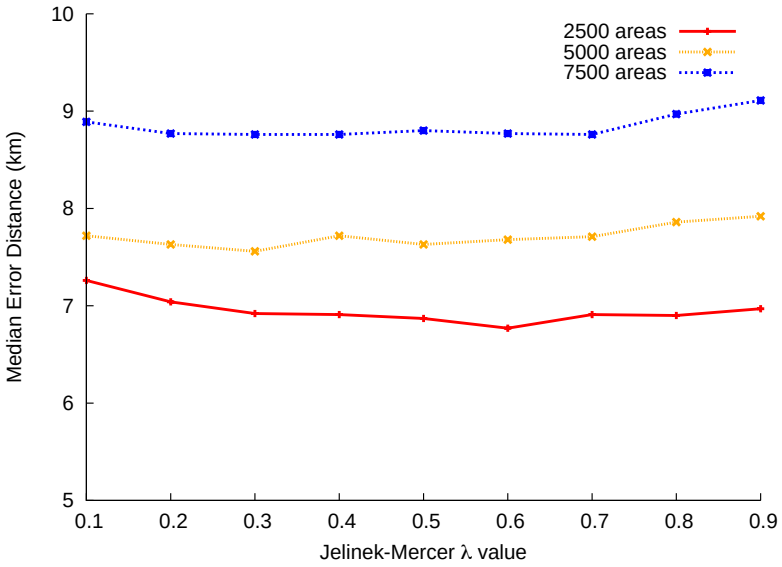


Figure 9: Median error distance over the *development* set when estimating locations with 2500, 5000 and 7500 clusters using different λ values for the Jelinek-Mercer smoothing method.

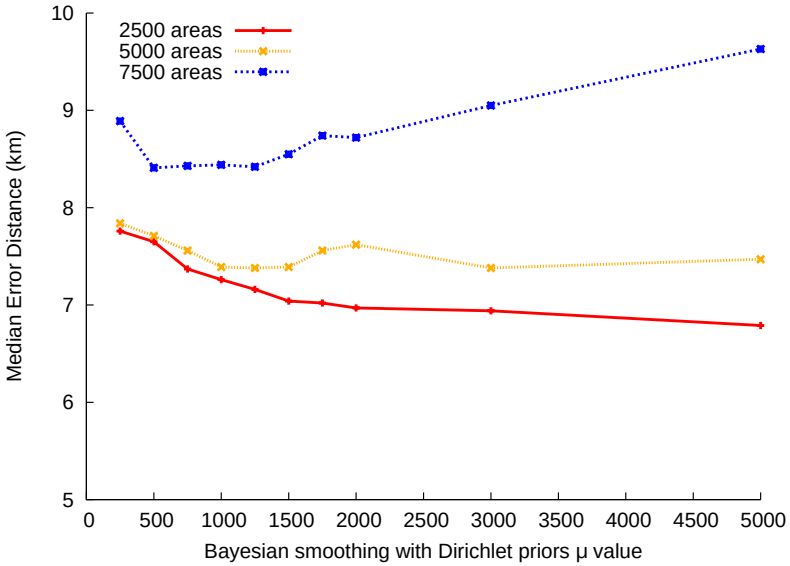


Figure 10: Median error distance over the *development* set when estimating locations with 2500, 5000 and 7500 clusters using different μ values for the Bayesian smoothing method with Dirichlet priors.

Table 5: Optimal μ values for Bayesian smoothing with Dirichlet priors for different values of clusters k, obtained after evaluation of a separate development set.

k	μ		k	μ
500	15000		5500	1000
1000	15000		6000	3000
1500	15000		6500	3000
2000	10000		7000	1500
2500	12500		7500	750
3000	5000		8000	750
3500	12500		8500	1000
4000	3000		9000	1000
4500	3000		9500	1000
5000	1750		10000	500

Table 6: Median error distance over the test collection when estimating locations with 500, 2500, 5000 and 7500 clusters, using different priors in the language models.

k	Uniform	ML	Home	ML+Home	GMM4	ML+GMM4
500	9.21	8.74	5.92	5.79	5.73	**5.61**
2500	5.38	5.34	**3.33**	3.34	3.96	3.65
5000	6.31	6.28	**2.92**	3.12	4.19	3.80
7500	7.23	6.75	**3.10**	3.21	4.88	3.63

priors. For $\lambda = 0.6$, Jelinek-Mercer smoothing produces a median error distance of 6.77 km, whereas Bayesian smoothing with Dirichlet priors results in 6.74 km at $\mu = 5000$. These findings confirm experimental results in other areas of information retrieval [43, 33], and to earlier work on georeferencing Flickr photos [32].

As our goal is to improve the overall performance of the framework, we will adopt the Bayesian smoothing method with Dirichlet priors for the remainder of our experiments, using optimized parameter values μ for each individual clustering level. These optimal parameter values are reported in Table 5.

4.4.2. Prior probability

Next, we determine the most suitable way of estimating the prior probability. In particular, we are interested in the results of the georeferencing process when using a maximum likelihood prior (ML), a uniform prior, the prior in (8), a prior based on Gaussian mixture models (GMM) with 1 to 5 component densities (GMM1 to GMM5), a combination of the ML and *Home* prior (ML+Home) and a combination of the ML and GMM1-5 priors (ML+GMMx).

Note that for this experiment, the regular test set (13 390 items) was used.

Table 6 presents the results of several of these configurations. The results of

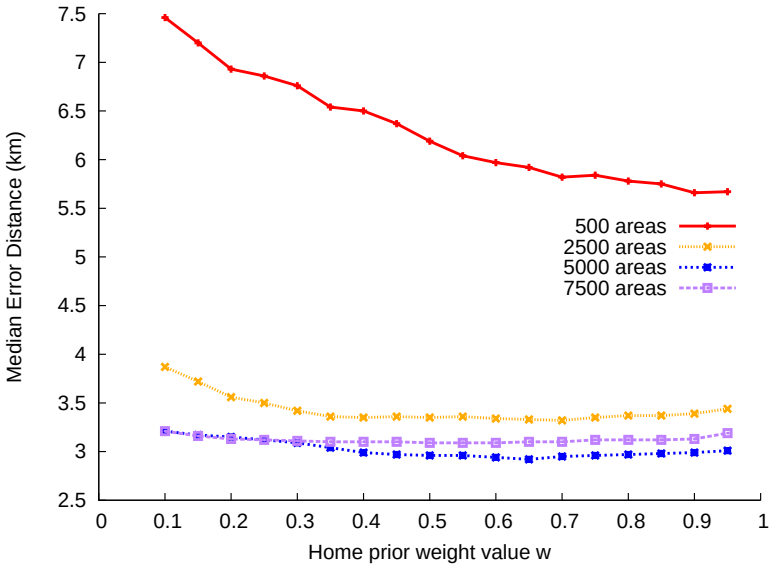


Figure 11: Median error distance over the test collection when estimating locations with 500, 2500, 5000 and 7500 clusters, using different weight values w for the home prior in the language models.

GMM1 to GMM3 are not presented, as these are all situated between the ML results and the GMM4 results. The results of GMM4 and GMM5 are identical and we therefore omitted GMM5 from this table. In the case of the *Home* prior, we set the parameter $w = 0.65$, a value that was experimentally found to be optimal. A discussion on this parameter value will follow shortly hereafter.

The optimal result can be found at $k = 5000$ when using a *Home* prior, resulting a median error distance of 2.92 km. The improvement over the baseline ML prior is clearly noticeable. When combined with the ML prior, the results of the Gaussian mixture model based prior are further improved. Interesting to note is that even though combining ML with the mixture models improves the overall performance, combining the Home prior with ML does not lead to a similar result. We can conclude that the *Home* prior, as defined in (8), is the best choice for optimizing the performance of our language modeling approach.

We investigated the robustness of the parameter w, controlling the influence of the distance between the suggested area and the home location of the photo owner. Figure 11 shows the results, confirming that our default parameter choice of $w = 0.65$ (based on initial experiments) turned out to be more or less in the middle of a range of good results. The figure also confirms that the infuence of the parameter w is rather limited, except for a small number of areas (e.g. $k = 500$).

Table 7: Summarizing the results of optimal configurations of the framework in terms of accuracy at certain error distances and median error distance (in km) over the test collection of 13 390 items.

Configuration	Acc@1	Acc@10	Acc@100	Acc@1000	MER
clustering	22.15	46.3	59.24	69.02	15.16
+ similarity search	34.59	50.61	60.69	69.81	9.23
+ geospread	35.05	53.91	65.15	72.65	5.75
+ smoothing + home prior	38.21	65.58	83.24	92.05	2.92

Table 8: Comparison of the optimal configuration of this paper and the submissions to the 2011 Placing Task, evaluated over the 5347 test videos for 2011.

	1 km	10 km	100 km	1000 km	10000 km
Li et al. [21]	0.21%	1.12%	2.71%	12.16%	79.45%
Krippner et al. [20]	9.86%	21.49%	29.79%	43.26%	84.16%
Ferres et al. [10]	14.61%	42.66%	56.65%	68.64%	94.93%
Choi et al. [4]	20.00%	38.20%	52.60%	66.30%	94.20%
Hauff et al. [13]	17.20%	50.76%	**70.77**%	82.61%	97.21%
Van Laere et al.[37]	**24.20**%	**51.49**%	63.27%	**85.62**%	**97.85**%
This work	**25.04**%	**53.53**%	**75.16**%	**87.21**%	**99.01**%

4.5. Summarizing improvements and results

Table 7 summarizes the result of optimizing the various components of the georeferencing framework and presents detailed accuracies for each of the configurations. Each transition to a better configuration is statistically significant[11] with a p-value $< 2.2 \times 10^{-16}$. The first substantial improvement is witnessed when using a similarity based area refinement instead of returning the location of the medoid of an area (Section 3.8.2). Although accuracies improve overall, the difference is most pronounced at smaller error distances. When the *geospread* method is used instead of choosing the most frequently occurring tags, the median error distance is further reduced. Finally, using Dirichlet smoothing with optimized values of the parameter μ and taking the home location of the photo owner into account if available, yields another significant improvement in accuracies and median error, which further decreases from 5.75 km to 2.92 km.

The optimal configuration presented here is an improved version of the baseline system that we used in the Placing Task benchmark that already outperformed other systems. The results presented in this paper show further improvements over the alternative approaches. Table 8 compares the optimal configuration of this paper to all the participants of the 2011 Placing Task.

[11] To evaluate the statistical significance, we used the sign test as the Wilcoxon signed-rank test is unreliable in this situation due to its sensitivity to outliers.

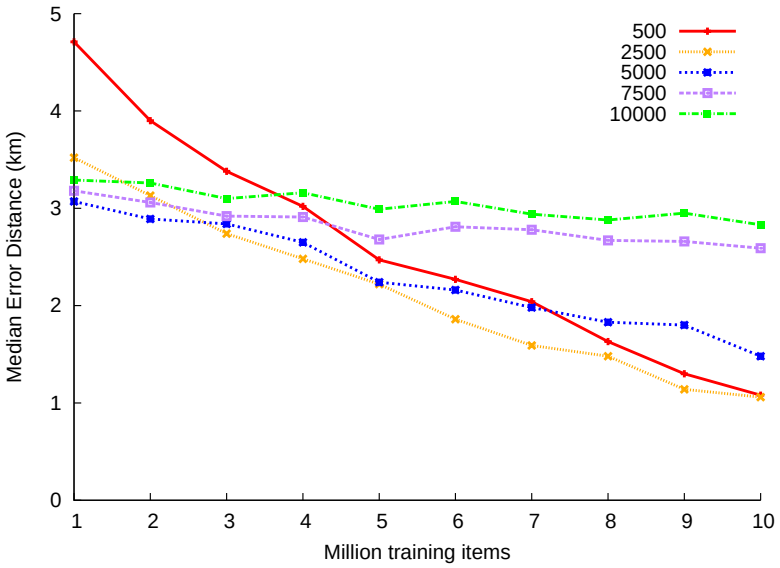


Figure 12: Median error distance of the 13 390 test items when estimating their locations at the 500, 2500, 5000, 7500 and 10 000 scales using an optimally tuned framework and a varying amount of training data.

4.6. The influence of training data

For this final experiment, we use the optimal configuration of the framework discovered so far. We start with a training set of 1M photos, and gradually increase the size of the training set in steps of 1M photos, establishing a trade-off between the amount of training data used by the system and the results it achieves with it. The difference in results between each pair of configurations is statistically significant with a p-value $< 2.2 \times 10^{-16}$.

Figure 12 presents the results of this experiment in terms of median error distance. Similar to the conclusion in Section 4.5, the best result is achieved at a scale of $k = 2500$ areas. In this case, making use of the full 10M training items results in a median error distance of 1.06 km. It is interesting to note that the coarsest scale $k = 500$ performs equally well, with a median error of only 1.08 km. More generally, we can notice that adding more training data has a larger effect when the number of clusters is smaller. Due to the large amount of training data available, the similarity search within an area performs very well.

It is important to understand why a two-step approach to georeferencing is necessary. Using only a (global) search for similar images, we will soon run into trouble. If there is no training photo available that has a tag set that is almost equal to the one we are looking for, there is no way for the similarity search to differentiate among the tags (some tags provide strong geographical clues), treating them all equally important. By starting the similarity search from the area that was obtained after classification, which implicitely resolves ambiguity

Table 9: Detailed results in terms of accuracy at certain error distances and median error distance (in km) for the optimal results when using 10M training items, using the optimal configuration of the framework.

	Acc@1	Acc@10	Acc@100	Acc@1000	MER
1M	36.33	62.23	84.41	92.46	3.52
2M	38.40	63.81	84.71	**93.23**	3.13
3M	40.13	64.16	84.85	92.74	2.74
4M	41.78	65.29	84.93	92.78	2.48
5M	43.31	66.00	84.81	93.02	2.22
6M	45.18	66.42	85.22	92.86	1.86
7M	45.97	67.48	**85.32**	93.09	1.59
8M	46.80	67.50	84.73	92.67	1.48
9M	48.71	**69.37**	85.29	93.07	1.14
10M	**49.63**	68.96	85.08	93.22	**1.06**

among terms, this problem will likely be resolved in many cases.

As the amount of training data increases, it becomes more likely that a training photo will be present that largely resembles the tag set we are looking for, improving the effectiveness of a (global) similarity search. This effect is clearly visible in Figure 12 for the configuration using 500 clusters.

Also, as can be concluded from Table 9, a larger amount of training data enables the framework to improve the location estimations within the sub 10 kilometer range. If a developer is satisfied with an error distance of for example maximum 100 kilometer for an application, the results are largely independent of the amount of training data used.

5. Conclusions and future work

Converting the problem of georeferencing Flickr resources based on textual meta-data into a classification problem is a popular approach in literature. After this initial classification step, a similarity search is performed in the area identified by the classifier. After a thorough experimental evaluation of this approach, we conclude the following:

- To achieve good results at sub-city scales (i.e. less than 10 kilometer of error distance), a similarity search component is essential.
- Information about the (home) location of the user is useful evidence for georeferencing Flickr resources.
- Among the clustering algorithms we have tested, k-medoids clustering performs best, due to its tendency to produce smaller scale clusters in areas of the world for which more training data is available.
- Applying a feature selection technique that is able to exploit the geographical aspect of the underlying data outperforms traditional methods.

- If we increase the amount of training data, the optimal number of clusters decreases due to an improved similarity search. Also, using more training data substanially improves accuracy in locating items within 10 km from their true location, while the results at an error margin of 100 km or 1000 km remain rather constant.

We see a number of opportunities for future work. Current approaches to georeferencing train models on the same type of data as the resources for which a location needs to be found. We believe that the language models trained from Flickr can be successfully used to estimate locations for other types of textual resources, without the need for a gazetteer. Initial experiments in [7] show promising results to this end. Second, as has been demonstrated in the experiments in Section 4.3, using an appropriate feature selection method is essential. Although the geographical spread filtering method introduced in [13] is a good example of a method that takes the spatial distribution of the tags into account, we believe that there is still scope for improvement in this aspect. Next, in our current approach, all features are weighted equally in the similarity search step. It is clear that not all available features associated with a Flickr photo have an equal importance. Research should be carried out to find similarity measures that better reflect this than the Jaccard measure. Further, there may be other sources of information that could provide additional evidence for georeferencing Flickr resources. For example, intuitively it seems clear that in one way or another, gazetteers may help to improve the results, although a good way for disambiguating tags would be needed. Another idea is to use the timestamp of a photo in combination with some visual features to find out during what moment of the day a photo was taken (e.g. night, midday, or in between) may help us to narrow the possible locations down to a number of time zones. Finally, current georeferencing approaches focus on returning a specific location for each query, although this is not meaningful in all cases. If the only tag available for a photo is "France", it makes more sense to return the boundaries of the country instead of a pre-defined geographical coordinate in the city centre of Paris. As a partial solution to this problem, [38] introduces a method to automatically identify what is the most appropriate level of granularity at which a photo should be localized.

Acknowledgements. The authors would like to thank Claudia Hauff for her help on the implementation of the geographical spread feature selection technique she presented at the MediaEval2011 workshop. We would also like to thank the organizers of the MediaEval workshop and the Placing Task organizers in particular for providing us with the Flickr dataset.

References

[1] S. Ahern, M. Naaman, R. Nair, J. H.-I. Yang, World explorer: visualizing aggregate data from unstructured text in geo-referenced collections, in: Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries, 2007, pp. 1–10.

[2] L. Backstrom, J. Kleinberg, R. Kumar, J. Novak, Spatial variation in search engine queries, in: Proceedings of the 17th International Conference on World Wide Web, 2008, pp. 357–366.

[3] Z. Cheng, J. Caverlee, K. Lee, You are where you tweet: a content-based approach to geo-locating Twitter users, in: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, 2010, pp. 759–768.

[4] J. Choi, H. Lei, G. Friedland, The 2011 icsi video location estimation system, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Choi_ICSI_Placing_me11wn.pdf, 2011.

[5] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 603–619.

[6] D. J. Crandall, L. Backstrom, D. Huttenlocher, J. Kleinberg, Mapping the world's photos, in: Proceedings of the 18th International Conference on World Wide Web, 2009, pp. 761–770.

[7] C. De Rouck, O. Van Laere, S. Schockaert, B. Dhoedt, Georeferencing Wikipedia pages using language models from Flickr, in: Proceedings of the Terra Cognita 2011 Workshop, 2011, pp. 3–10.

[8] T. Dunning, Accurate methods for the statistics of surprise and coincidence, Comput. Linguist. 19 (1) (1993) 61–74.

[9] J. Eisenstein, B. O'Connor, N. A. Smith, E. P. Xing, A latent variable model for geographic lexical variation, in: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, 2010, pp. 1277–1287.

[10] D. Ferres, H. Rodriguez, Talp at mediaeval 2011 placing task: Georeferencing flickr videos with geographical knowledge and information retrieval, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Ferres_UPC_Placing_me11wn.pdf, 2011.

[11] G. Friedland, J. Choi, A. Janin, Video2gps: a demo of multimodal location estimation on flickr videos, in: Proceedings of the 19th ACM international conference on Multimedia, 2011, pp. 833–834.

[12] M. Goodchild, Citizens as sensors: the world of volunteered geography, GeoJournal 69 (2007) 211–221.

[13] C. Hauff, G.-J. Houben, WISTUD at MediaEval 2011: Placing Task, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Hauff_WISTUD_Placing_me11wn.pdf.

[14] C. Hauff, G.-J. Houben, Placing images on the world map: a microblog-based enrichment approach, in: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, 2012, pp. 691–700.

[15] J. H. Hays, A. A. Efros, IM2GPS: Estimating geographic information from a single image, in: Proceedings of the 21st IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.

[16] B. Hecht, M. Raubal, GeoSR: Geographically explore semantic relations in world knowledge, in: L. Bernard, A. Friis-Christensen, H. Pundt (eds.), 11th AGILE International Conference on Geographic Information Science, 2008, pp. 95–114.

[17] L. Hollenstein, Capturing vernacular geography from georeferenced tags, Master's thesis, University of Zurich (2008).

[18] C. B. Jones, A. I. Abdelmoty, D. Finch, G. Fu, S. Vaid, The SPIRIT spatial search engine: Architecture, ontologies and spatial indexing, in: Proceedings of the Third International Conference on Geographic Information Science, 2004, pp. 125–139.

[19] L. Kennedy, M. Naaman, Generating diverse and representative image search results for landmarks, in: Proceedings of the 17th International Conference on World Wide Web, 2008, pp. 297–306.

[20] F. Krippner, G. Meier, J. Hartmann, R. Knauf, Placing Media Items Using the Xtrieval Framework, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Krippner_CUT_Placing_me11wn.pdf.

[21] L. T. Li, J. Almeida, R. da S. Torres, Recod working notes for placing task mediaeval 2011, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Li_UNICAMP_Placing_me11wn.pdf, 2011.

[22] M. D. Lieberman, H. Samet, J. Sankaranayananan, Geotagging: using proximity, sibling, and prominence clues to understand comma groups, in: Proceedings of the 6th Workshop on Geographic Information Retrieval, 2010, pp. 6:1–6:8.

[23] E. Moxley, J. Kleban, B. Manjunath, Spirittagger: a geo-aware tag suggestion tool mined from Flickr, in: Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval, 2008, pp. 24–30.

[24] O. A. B. Penatti, L. T. Li, J. Almeida, R. da S. Torres, A visual approach for video geocoding using bag-of-scenes, in: Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, 2012, pp. 53:1–53:8.

[25] A. Popescu, I. Kanellos, Creating visual summaries for geographic regions, in: IR+SN Workshop (at ECIR), 2009.

[26] A. Rae, V. Murdock, P. Serdyukov, P. Kelm, Working Notes for the Placing Task at MediaEval2011, in: Working Notes of the MediaEval Workshop, Pisa, Italy, September 1-2, 2011, CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-807/Rae_Placing_me11overview.pdf.

[27] T. Rattenbury, N. Good, M. Naaman, Towards automatic extraction of event and place semantics from flickr tags, in: Proceedings of the 30th Annual International ACM SIGIR Conference, 2007, pp. 103–110.

[28] T. Rattenbury, M. Naaman, Methods for extracting place semantics from Flickr tags, ACM Transactions on the Web 3 (1) (2009) 1–30.

[29] D. Reynold, Gaussian mixture models, Tech. rep., MIT Lincoln Laboratory (2008).

[30] P. Schmitz, Inducing ontology from Flickr tags, in: Proceedings of the Collaborative Web Tagging Workshop, 2006, pp. 210–214.

[31] S. Schockaert, M. De Cock, Neighborhood restrictions in geographic IR, in: Proceedings of the 30th Annual International ACM SIGIR Conference, 2007, pp. 167–174.

[32] P. Serdyukov, V. Murdock, R. van Zwol, Placing Flickr photos on a map, in: Proceedings of the 32nd Annual International ACM SIGIR Conference, 2009, pp. 484–491.

[33] M. D. Smucker, J. Allan, An investigation of Dirichlet prior smoothing's performance advantage, Tech. Rep. IR-445, University of Massachusetts (2005).

[34] O. Van Laere, S. Schockaert, B. Dhoedt, Combining multi-resolution evidence for georeferencing Flickr images, in: Proceedings of the 4th International Conference on Scalable Uncertainty Management, 2010, pp. 347–360.

[35] O. Van Laere, S. Schockaert, B. Dhoedt, Towards automated georeferencing of flickr photos, in: Proceedings of the 6th Workshop on Geographic Information Retrieval, 2010, pp. 5:1–5:7.

[36] O. Van Laere, S. Schockaert, B. Dhoedt, Finding locations of Flickr resources using language models and similarity search, in: Proceedings of the 1st ACM International Conference on Multimedia Retrieval, 2011, pp. 48:1–48:8.

[37] O. Van Laere, S. Schockaert, B. Dhoedt, Ghent university at the 2011 Placing Task, in: Working Notes of the MediaEval Workshop, 2011.

[38] O. Van Laere, S. Schockaert, B. Dhoedt, Georeferencing flickr photos using language models at different levels of granularity: An evidence based approach, Web Semantics: Science, Services and Agents on the World Wide Web.

[39] F. Wilske, Approximation of neighborhood boundaries using collaborative tagging systems, in: Proceedings of the GI-Days, 2008, pp. 179–187.

[40] B. Wing, J. Baldridge, Simple supervised document geolocation with geodesic grids, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011, pp. 955–964.

[41] Y. Yang, Z. Gong, L. H. U, Identifying points of interest by self-tuning clustering, in: Proceedings of the 34th International ACM SIGIR Conference, 2011, pp. 883–892.

[42] Y. Yang, J. O. Pedersen, A comparative study on feature selection in text categorization, in: Proceedings of the 14th International Conference on Machine Learning, 1997, pp. 412–420.

[43] C. Zhai, J. Lafferty, A study of smoothing methods for language models applied to Ad Hoc information retrieval, in: Proceedings of the 24th Annual International ACM SIGIR Conference, 2001, pp. 334–342.

[44] Y.-T. Zheng, Z.-J. Zha, T.-S. Chua, Research and applications on georeferenced multimedia: a survey, Multimedia Tools and Applications 51 (2011) 77–98.