Georeferencing Wikipedia documents using data from social media sources

OLIVIER VAN LAERE, Yahoo Labs, Barcelona
STEVEN SCHOCKAERT and VLAD TANASESCU, Cardiff University
BART DHOEDT, Ghent University - iMinds
CHRISTOPHER B. JONES, Cardiff University

Social media sources such as Flickr and Twitter continuously generate large amounts of textual information (tags on Flickr and short messages on Twitter). This textual information is increasingly linked to geographical coordinates, which makes it possible to learn how people refer to places by identifying correlations between the occurrence of terms and the locations of the corresponding social media objects. Recent work has focused on how this potentially rich source of geographic information can be used to estimate geographic coordinates for previously unseen Flickr photos or Twitter messages. In this paper, we extend this work by analysing to what extent probabilistic language models trained on Flickr and Twitter can be used to assign coordinates to Wikipedia articles. Our results show that exploiting these language models substantially outperforms both (i) classical gazetteer-based methods (in particular, using Yahoo! Placemaker and Geonames) and (ii) language modelling approaches trained on Wikipedia alone. This supports the hypothesis that social media are important sources of geographic information, which are valuable beyond the scope of individual applications.

Categories and Subject Descriptors: H.3 [**INFORMATION STORAGE AND RETRIEVAL**]: Information Search and Retrieval; I.2.6 [**ARTIFICIAL INTELLIGENCE**]: Learning

General Terms: Experimentation, Measurement

Additional Key Words and Phrases: Geographic Information Retrieval, Language Models, Semi-structured Data

ACM Reference Format:
Van Laere, O., Schockaert, S., Tanasescu, V., Dhoedt, B. and Jones, C. B. 2012. Georeferencing Wikipedia documents using data from social media sources. ACM Trans. Inf. Syst. V, N, Article A (January YYYY), 33 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

Location plays an increasingly important role on the Web. Smartphones enable users around the world to participate in social media activities, such as sharing photos or broadcasting short text messages. In this process, the content that is added by a given user is often annotated with its geographical location (either automatically by a GPS device or manually by the user). As a result, more and more georeferenced content is becoming available on the web. At the same time, due to the popularity of location-

Author's addresses: O. Van Laere, Yahoo Labs, Barcelona, Spain; email: vanlaere@yahoo-inc.com; B. Dhoedt, Department of Information Technology, Ghent University, iMinds, Belgium; email: Bart.Dhoedt@intec.ugent.be; S. Schockaert and V. Tanasescu and C. B. Jones, School of Computer Science & Informatics, Cardiff University, United Kingdom.; email: {S.Schockaert, V.Tanasescu, C.B.Jones}@cs.cardiff.ac.uk.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

based services, the demand for georeferenced content has also become stronger. Applications such as Foursquare[1] or Google Places[2], for instance, allow users to find nearby places of a given type, while applications such as Wikitude[3] provide information about a user's surroundings by using georeferenced Wikipedia articles, among others.

Several authors have investigated how geotagged Flickr photos (i.e. Flickr photos that are associated with coordinates) can be used to estimate coordinates for photos without geotags [Crandall et al. 2009; Serdyukov et al. 2009; Van Laere et al. 2011]. Although some authors have exploited visual features from the actual pictures, the dominant approach consists of training language models for different geographic areas, and subsequently using these language models to estimate in which area a photo was most likely taken. More recently, similar approaches have been proposed to georeference Twitter messages [Eisenstein et al. 2010; Cheng et al. 2010; Kinsella et al. 2011] and Wikipedia articles [Wing and Baldridge 2011; Roller et al. 2012]. A key aspect of the aforementioned approaches is that the considered language models are always trained on the type of resources that are georeferenced (e.g. Flickr photos are used to train a system for georeferencing Flickr photos). While this makes sense in the case of Flickr and Twitter, it is less clear whether an approach for georeferencing Wikipedia articles can be truly effective in this way. Indeed, since different users may take photos of the same places, given a new photo to be georeferenced, it will often be the case that several photos from the same place are contained in the training data. Hence, if we can identify these photos from the training data, accurate coordinates for the new photo can be found. In contrast, given a new Wikipedia article about a place, there should normally not be any other articles about that place in Wikipedia, implying that at most only approximate coordinates can be inferred (e.g. by discovering in which city the described place is located). On the other hand, there may be georeferenced photos on Flickr of the place, or georeferenced Twitter messages that describe the specific Wikipedia article.

In addition to supporting location-based services such as Wikitude, the ability to accurately georeference Wikipedia articles, or web pages in general [Amitay et al. 2004], is an important requirement for developing geographic information retrieval (GIR) systems [Purves and Jones 2011]. For many queries, the relevance of search results is determined in part by their geographic scope. For example, users searching for information about primary schools may only be interested in nearby schools. In the field of GIR, most mainstream approaches determine the geographic scope by looking for mentions of place names and by looking up the corresponding coordinates in a gazetteer (along with other methods, such as recognizing telephone prefixes, postcodes, etc.).

In this paper, our central hypothesis is that a system for georeferencing Wikipedia articles about places can substantially benefit from using sources such as Flickr or Twitter. As the number of georeferenced Flickr photos and Twitter messages is increasing at a fast pace, if confirmed, this hypothesis could form the basis of a powerful new approach for georeferencing Wikipedia articles, and web documents in general, continuously improving its performance as more training data becomes available.

The results we present in this paper strongly support our hypothesis. In particular, using a language model trained using 376K Wikipedia documents, we obtain a median error of 4.17 km, while a model trained using 32M Flickr photos yields a median error of 2.5 km. When combining both models, the median error is further reduced to 2.16 km. Repeating the same experiment with 16M tweets as the only training data results in a median error of 35.81 km. When combined with the Wikipedia training data

[1] https://foursquare.com/
[2] http://www.google.com/places/
[3] http://www.wikitude.com/

the median error decreases to 3.69 km. Combining all three models results in a median error of 2.18 km, suggesting that while Twitter is useful in the absence of Flickr data, the evidence it provides is superseded by the evidence encoded in the Flickr models. The methodology we adopt in this paper to training and using the language models is based on our earlier work in the area of georeferencing Flickr photos [Van Laere et al. 2013]. However, to the best of our knowledge, apart from a preliminary analysis in [De Rouck et al. 2011], this paper is the first to analyse the performance of language models trained on social media for georeferencing full text documents.

The remainder of this paper is organized as follows: Section 2 summarizes related work in the field of georeferencing textual resources. Section 3 describes the different data sources we consider and summarizes the datasets we use in our evaluation. Next, Section 4 describes how we estimate and combine language models from Flickr, Twitter and Wikipedia. Our evaluation is discussed in detail in Section 5. In Section 6 we provide a discussion about our main result. Finally, Section 7 states the conclusions and discusses future work.

2. RELATED WORK

We review two areas of work on georeferencing: gazetteer-based methods in Section 2.1, followed by language modelling based methods in Section 2.2.

2.1. Gazetteer based methods

Gazetteers are essentially lists or indices containing information about a large number of known places, described by features such as geographic coordinates, semantic types, and alternative names. Examples of gazetteers are Yahoo! GeoPlanet[4] and Geonames[5], the latter being freely available for download and containing information about over 8.1 million different entities worldwide. The data contained in a gazetteer is mostly manually selected and reviewed by domain experts and thus tends to be of high quality. However, manual moderation is a time-consuming and cumbersome task, which implies in particular that most gazetteers have an especially limited coverage of spots, i.e. small scale features that can be represented appropriately by a single point coordinate.

In an effort to address this issue as well as the limited coverage of some gazetteers, [Popescu et al. 2008] proposes a method for automatically constructing a gazetteer from different sources using text mining. [Manguinhas et al. 2008] produced a gazetteer service that accesses multiple existing gazetteer and other place name resources, using a combination of manual resolution of feature types and automated name matching to detect duplicates. [Smart et al. 2010] access multiple gazetteers and digital maps in a mediation architecture for a meta-gazetteer service using similarity matching methods to conflate the multiple sources of place data in real-time. In [Twaroch et al. 2008], a method is proposed to discover new place names, by analysing how people describe their home location on the Gumtree website. The approach we propose in this paper can be seen as an alternative to enriching existing gazetteers. Instead of using web sources for discovering lists of places and use these lists to implement systems for georeferencing text documents, we aim to directly estimate geographical location, without the intermediate step of constructing or enriching a gazetteer.

Given access to a comprehensive gazetteer, a natural way to discover the geographic scope of a web page consists of identifying place names and looking up their coordinates in the gazetteer. In practice, however, this method is complicated by the fact that many place names are highly ambiguous. A well known-example is “Springfield”:

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

at least 58 populated places with this name are listed in Geonames. Georeferencing methods using a gazetteer have to cope with this. In [Amitay et al. 2004], gazetteers are used to estimate the locations of toponyms mentioned in text and a geographical focus is determined for each page. During this process, two different types of ambiguities are described: geo/geo, e.g. the previous example of "Springfield", or geo/non-geo, such as "Turkey" or "Bath", which are also common nouns in English. Heuristic strategies to resolve both type of ambiguities are proposed. [Weinberger et al. 2008] presents a probabilistic framework that is able to propose additional tags capable of disambiguating the meaning of the tags associated to a Flickr photo. For instance, given the tag "washington", adding "dc" or "seattle" resolves the possible ambiguity. [Lieberman et al. 2010] investigated toponym resolution based on the understanding of comma groups, such as the previous example of "Washington, DC", to determine the correct interpretation of the place names. [Tobin et al. 2010] resolves toponyms against a number of gazetteers, and tackle the problem of ambiguity using a number of heuristics based on an in-depth analysis carried out in [Leidner 2007]. In addition to all aforementioned types of ambiguity, place names are sometimes used in a nonspatial sense (e.g. "Brussels" refers to a political entity in a sentence such as "According to Brussels, the proposed measures have been ineffective"). This form of ambiguity can, in principle, be addressed using standard techniques for named entity recognition (NER), although it is a non-trivial problem.

Another limitation of gazetteer based methods is that people often use vernacular names to describe places, which tend to be missing from gazetteers. For instance, "The Big Apple" is used when referring to "New York City". To cope with this [Jones et al. 2008] extracts knowledge of vernacular names from web sources by exploiting co-occurrence on the web with known georeferenced places.

2.2. Language modelling based methods

Over the past few years considerable research has focused on georeferencing Flickr photos on the basis of their tags. The tendency for particular tags to be clustered spatially, and hence to provide strong evidence for the place at which a photo was taken, was studied by [Rattenbury et al. 2007; Rattenbury and Naaman 2009] who compared alternative clustering techniques and demonstrated the benefits of hybrid approaches. Most existing georeferencing methods exploit the clustering properties in one way or another to convert the georeferencing task to a classification problem. For instance, in [Crandall et al. 2009] locations of unseen resources are determined using the mean shift clustering algorithm, a non-parametric clustering technique from the field of image segmentation. The advantage of this clustering method is that the number of clusters is determined automatically from a scale parameter. To assign locations to new images, both visual (keypoints) and textual (tags) features have been used in [Crandall et al. 2009]. 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). The authors found that visual features, when combined with textual features, substantially improve accuracy in the case of landmarks. In [Serdyukov et al. 2009], the idea is explored that whenever a classifier suggests a certain area where an image was most likely taken, the surrounding areas could be considered as well to improve the results. Their observation is that typically not only the correct area will receive a high probability, but also surrounding areas will exhibit similar behaviour. This idea was further elaborated on in [Van Laere et al. 2012], where the evidence for a certain location from models trained at different levels of granularity is combined using Dempster-Shafer evidence theory to determine the most likely location

where a certain photo was taken and to assess the spatial granularity for which this estimation is meaningful. Finally, [Van Laere et al. 2011] showed that approaches using classification benefit from a second step, in which a suitable location is determined within the area that was found by the classifier, by assessing the similarity (here the Jaccard measure was used to assess similarity) between the photo to be georeferenced and the photos from the training data that are known to be located in that area. The interest of the research community into this problem resulted in the Placing Task, an evaluation framework focussing on the problem of georeferencing Flickr videos [Rae and Kelm 2012], as part of the MedialEval benchmarking initiative[6].

In parallel and using similar techniques, researchers have looked into georeferencing Twitter messages. Due to their limited length, Twitter messages are much harder to georeference than for instance Wikipedia articles or general web pages. For example, when an ambiguous term occurs, it is less likely that the surrounding words will provide sufficient context for accurate disambiguation. However, as tweets are rarely posted in isolation, previous messages from the same user can be exploited as context information. Following such a strategy, [Eisenstein et al. 2010] shows that it is possible to estimate the geographical location of a Twitter user using latent topic models, an approach which was shown to outperform text regression and supervised topic models. [Cheng et al. 2010] proposes a method to determine the city in which a Twitter user is located (among a pre-selected set of cities). Each city is modelled through a probabilistic language model, which can be used to estimate the probability that the user's tweets were written by a resident of that city. While this baseline model only found the correct city for 10% of the users, substantial improvements were obtained when using a term selection method to filter all terms that are not location-relevant, leading to a 49.8% accuracy on a city scale. [Kinsella et al. 2011] trains language models over geotagged Twitter messages, and rely on Kullback-Leibler divergence to compare the models of locations with the models of tweets. The results show that around 65% of the tweets can thus be located within the correct city (among a pre-selected set of 10 cities with high Twitter usage) and around 20% even within the correct neighbourhood (in this case, within the spatial scope of New York only). In comparison, the effectiveness of gazetteer based methods for georeferencing Twitter messages was found to amount to 1.5% correctly georeferenced messages on the neighbourhood scale (in this experiment Yahoo! Placemaker was used).

When it comes to georeferencing Wikipedia documents, the work of [Wing and Baldridge 2011] is of particular interest. After laying out a grid over the Earth's surface (in a way similar to [Serdyukov et al. 2009]), for each grid cell a generative language model is estimated using only Wikipedia training data. To assign a test item to a grid cell, its Kullback-Leibler divergence with the language models of each of the cells is calculated. Results are also reported for other approaches, including Naive Bayes classification. The follow-up research in [Roller et al. 2012] improved this method in two ways. First, an alternative clustering of the training data is suggested: by using k-d trees, the clustering is more robust to data sparsity in certain clusters when using large datasets. Indeed, most of the datasets are not uniformly distributed and using a grid with equal-sized cells will ignore the fact that certain parts of the world can be covered quite densely or sparsely with training data, depending on the location. In this paper, we use k-medoids clustering for a similar purpose. A second improvement is that instead of returning the center of the grid cell, the centre-of-gravity is returned of the locations of the Wikipedia pages from the training data that are located in the cell. The significance of this latter improvement is confirmed by our earlier results in

[6] http://www.multimediaeval.org/

[Van Laere et al. 2011], in the setting of georeferencing Flickr photos, and is described in Section 4.5.

In this paper, we will investigate the use of mixed data sources to georeference Wikipedia documents. The approaches outlined above indeed all use the same type of information for training and test data. First efforts in this area include our previous work [De Rouck et al. 2011] where a preliminary evaluation has been carried out of the effectiveness of georeferencing Wikipedia pages using language models 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. In addition some evaluations have been carried out that use data from multiple sources. Finally, for the task of georeferencing Flickr photos, [Hauff and Houben 2012] introduces the idea of using evidence from Twitter messages by the same user within a given time interval around the time stamp of the photo.

3. DATASETS

We will evaluate our techniques using two test collections of Wikipedia articles. The first test set, discussed in detail in Section 3.1, is used to compare our approach against earlier work in [Wing and Baldridge 2011] and [Roller et al. 2012], but has a number of shortcomings. For this reason, we constructed a second test set of Wikipedia documents, as described in Section 3.2. Our training data will consist of Wikipedia articles, in addition to Flickr photos and Twitter messages, as detailed in Sections 3.3 to 3.5.

3.1. Wing and Baldrigde (W&B) Wikipedia training and test set

The training and test data from [Wing and Baldridge 2011] has been made available on the TextGrounder website[7]. Using this dataset enables us to compare the results reported in [Wing and Baldridge 2011] and [Roller et al. 2012] to the results we obtain using our approach. The dataset originates from the original English-language Wikipedia dump of September 4, 2010[8], which was pre-processed as described in [Wing and Baldridge 2011], and divided into 390 574 training articles and 48 589 test articles. In [Roller et al. 2012] a slightly modified version of this dataset has been used. Accordingly, we filtered the dataset for the 390 574 Wikipedia training documents and 48 566 Wikipedia test documents that have been used in [Roller et al. 2012].

However, this test set has a number of shortcomings:

— No distinction is made between Wikipedia articles that describe a precise location on the one hand (e.g. the Eiffel tower), and Wikipedia articles whose geographic scope cannot reasonably be approximated by a single coordinate, such as large geographical entities (e.g. rivers, trails or countries) or Wikipedia lists (e.g. “List of the KGB chairmen”), on the other hand.
— To create the ground truth, the Wikipedia dump used was filtered for pages that mention a geographical coordinate, while the page itself has no explicitly assigned coordinates. As an example, for the article on “List of shipwrecks in 1964”[9], the ground truth location was set to 44°12’N 08°38’E, which is mentioned in the article in relation to October 14, 1964, the day the ship *Dia* sank south of Savona, Italy.
— As part of the preprocessing considered by [Wing and Baldridge 2011], all information about word ordering has been removed from the original document. This seriously disadvantages our method which relies on n-grams, because Flickr tags often correspond to the concatenation of several terms.

[7] http://code.google.com/p/textgrounder/wiki/WingBaldridge2011
[8] http://download.wikimedia.org/enwiki/20100904/enwiki-20100904-pages-articles.xml.bz2
[9] http://en.wikipedia.org/wiki/List_of_shipwrecks_in_1964

We have therefore also evaluated our method on a newly crawled test collection, as discussed next.

3.2. The Wikipedia spot training and test set

Constructing a dataset from raw dumps of Wikipedia pages requires pre-processing as these pages contain fragments of markup language that are not relevant in this context. On the other hand, certain markup codes provide meaningful information that we would like to keep, such as captions of links to files, images or tables. Our pre-processing script converts the example raw Wikipedia fragment:

```
[[Image:Abbotsford Morris edited.jpg|thumb|300px|right|Abbotsford in 1880.]]
'''Abbotsford''' is a [[historic house]] in the region of the [[Scottish
Borders]] in the south of [[Scotland]], near [[Melrose]], on the south
bank of the [[River Tweed]]. It was formerly the residence of
[[historical novel]]ist and [[poet]], [[Walter Scott]].
It is a Category A [[Listed Building]].
```

to the following text: “Abbotsford in 1880. Abbotsford is a historic house in the region of the Scottish Borders in the south of Scotland, near Melrose, on the south bank of the River Tweed. It was formerly the residence of historical novelist and poet, Walter Scott. It is a Category A Listed Building”.

To construct the test set, we downloaded the DBPedia 3.7 “Geographic Coordinates” English (nt) Wikipedia dump[10], containing the geographical coordinates and Wikipedia ID’s (e.g. “Abbotsford_House”) of 442 775 entities. From these, we retained the 47 493 documents whose coordinates are located within the bounding box of the United Kingdom. The raw XML version of these documents have been be obtained by posting the (encoded) ID’s against Wikipedia’s `Special:Export`[11] function.

Wikipedia contains numerous documents that are hard to pinpoint to a precise location, discussing for example architectural styles, schools of thought, people or concepts. As we consider techniques for estimating precise coordinates, it is useful to restrict the evaluation to articles that have a limited spatial extent, such as landmarks, buildings, schools, or railway stations. Although DBPedia lists the coordinates of the documents, it does not provide any information on the “type” or “scale” of the coordinates. However, this information can be extracted from the XML documents by scanning for the Wikipedia coordinate template markup (i.e. `{{coord*}}`) and parsing its contents. After extracting this information, we have further filtered the dataset, keeping only the documents whose coordinates either refer to a location of type “railwaystation, landmark or edu”[12] (being the only types that refer to spots), or have a reported scale of 1:10000 or finer.

The result is a set of 21 839 Wikipedia test documents. This dataset, along with the pre-processing script, has been published online[13]. To make this set compatible with the W&B training set, we removed any occurrences of our test documents from the W&B training data, resulting in a training set of 376 110 Wikipedia documents. This reduced training set is used whenever our “spot” test set is used. When evaluating the W&B test set, we still use the full training set.

Note that, while the spot dataset only contains Wikipedia articles that are located within the bounding box of the UK, our method does not exploit this information.

[10] `http://downloads.dbpedia.org/3.7/en/geo_coordinates_en.nt.bz2`
[11] `http://en.wikipedia.org/wiki/Special:Export`
[12] For a full list of Wikipedia GEO types, see `http://en.wikipedia.org/wiki/Wikipedia:GEO#type:T`
[13] Our pre-processing script, along with the original XML and processed test set are made available online at `https://github.com/ovlaere/georeferencing_wikipedia`

The restriction on the UK is motivated by the possibility of future work, which could consider additional country-specific evidence, such as local news articles.

3.3. Flickr training set

In April 2011, we collected the meta-data of 105 118 157 georeferenced Flickr photos using the public Flickr API. We pre-processed the resulting dataset by removing photos with invalid coordinates as well as photos without any tags. For photos that are part of bulk uploads, following [Serdyukov et al. 2009] we removed all but one photo. This resulted in a set of 43 711 679 photos. Among these photos, we extracted only those that reported an accuracy level of 12 at least, which means that the geographical coordinates of the photos we use are accurate at a city block level. This final step resulted in a set of 37 722 959 photos, of which 32 million photos served as training data for this paper.

3.4. Twitter training set

Twitter provides samples of the tweets published by its users[14]. We monitored the "Gardenhose" stream using the `statuses/filter` API method in combination with a bounding box parameter covering the entire world. This allowed us to track only Twitter messages with a geographical coordinate attached to them. Doing so for a period from March to August 2012 resulted in a dataset of 170 668 054 tweets.

In order to avoid an unfair bias in the number of word occurrences at certain locations caused by a single user, we aggregated all tweets from a given user at the same location into a single document. The resulting document is represented as a set of terms, i.e. multiple occurrences of the same term at the same location by the same user are only counted once. For example:

```
52.135978 -0.466651  Bonus 4 - Olympic torch http://t.co/q3yNthcj
52.135978 -0.466651  Bonus 3 - Olympic torch http://t.co/wZUH4a5B
52.135978 -0.466651  Bonus 6 - Olympic torch http://t.co/M9Tm6Ow0
52.135978 -0.466651  Bonus 5 - Olympic torch http://t.co/HWqiTDZy
52.135978 -0.466651  Bonus 9 - Olympic torch http://t.co/2ovhQdPu
52.135978 -0.466651  Bonus 8 - Olympic torch http://t.co/iIRvEe5C
52.135978 -0.466651  Bonus 7 - Olympic torch http://t.co/hO8PAsf1
```

then becomes:

```
52.135978 -0.466651 Bonus 3 4 5 6 7 8 9 - Olympic torch
http://t.co/q3yNthcj http://t.co/wZUH4a5B http://t.co/M9Tm6Ow0
http://t.co/HWqiTDZy http://t.co/2ovhQdPu http://t.co/iIRvEe5C
http://t.co/hO8PAsf1
```

Next, we only retained those documents in which at least one hashtag (e.g. #empirestatebuilding) occurs, further reducing the dataset to 18 952 535 documents. In this paper we used a subset of 16 million of these documents as training data.

3.5. Data compatibility

Further pre-processing was needed to arrive at a meaningful combination of Wikipedia, Flickr and Twitter data. For example, while Wikipedia documents contain capitalized words, the Flickr tags are all lowercase and moreover often correspond to the concatenation of several words, e.g. photos on Flickr may be tagged as "empirestatebuilding". This has implications in two steps of our approach:

[14] `https://dev.twitter.com/docs/streaming-apis/streams/public`

(1) the estimation of a language model from Flickr or Twitter data while test documents are taken from Wikipedia. (Section 4.3).
(2) the comparison of similarity between a test document and training items from the selected area are compared, in the procedure from Section 4.5.

3.5.1. Wikipedia documents and Flickr data. To make the Wikipedia test data compatible with the Flickr training data, we can "translate" the documents to Flickr tags. This can easily be achieved by converting the Wikipedia test articles to lowercase, and scanning for terms or concatenations of up to 5 consecutive terms that correspond to a Flickr tag from the training data.

3.5.2. Wikipedia documents and Twitter documents. To facilitate comparison between Wikipedia test data and Twitter training data, we convert all terms to lowercase and for each of the occurring hashtags, we remove the leading "#" sign. Again, we scan the Wikipedia documents for terms or concatenations of up to 3 consecutive terms that correspond to any term occuring in the Twitter training data, as especially hashtags may correspond to the concatenation of several terms.

4. ESTIMATING LOCATIONS USING LANGUAGE MODELLING

Probabilistic (unigram) language models have proven particularly effective to estimate the location of textual resources [Serdyukov et al. 2009; Wing and Baldridge 2011; Roller et al. 2012]. In this section we will detail the approach adopted, which is based on the algorithm outlined in [Van Laere et al. 2011]. The fundamental addition to this method consists of the fact that the models were trained using a combination of Wikipedia, Flickr and Twitter data. This implies two modifications to the approach from [Van Laere et al. 2011]:

(1) There is need for a way to combine different language models
(2) The last phase of our approach involves assessing the similarity between the item to be georeferenced and the items in the training set. This means that we need a way of measuring the similarity between e.g. a Wikipedia article and a Flickr photo.

Our approach consists of two main steps. First, we treat the problem of estimating the location of an unseen document $\mathcal{D}$ as a text classification problem. To this end, the coordinates appearing in the training data, here an aggregation of Wikipedia, Flickr and Twitter data, are clustered into k distinct areas a, that make up the clustering $\mathcal{A}_k$. After clustering, a feature selection procedure is applied aimed at removing terms that are not spatially relevant (e.g. removing tags such as *birthday* or *beautiful*). In particular, we select a vocabulary $\mathcal{V}$ of m features. Given a specific clustering $\mathcal{A}_k$ and the vocabulary of features $\mathcal{V}$, language models for each cluster can be estimated. In this paper, we estimate a separate language model from Wikipedia, Twitter and Flickr. This process is illustrated in Figure 1.

Given a document $\mathcal{D}$ for which we want to determine suitable geographical coordinates, we first calculate the probabilities that each of the language models has generated that document. Then the language models from Wikipedia, Twitter and Flickr are combined to produce a final probability estimate, which is used to select the cluster $a \in \mathcal{A}_k$ that is most likely to contain the location of $\mathcal{D}$. In the second step, once an area a has been chosen, we estimate the location of $\mathcal{D}$ as the location of the training item from area a that is most similar to $\mathcal{D}$. This process is illustrated in Figure 2. Next, we discuss each of these steps in more detail.

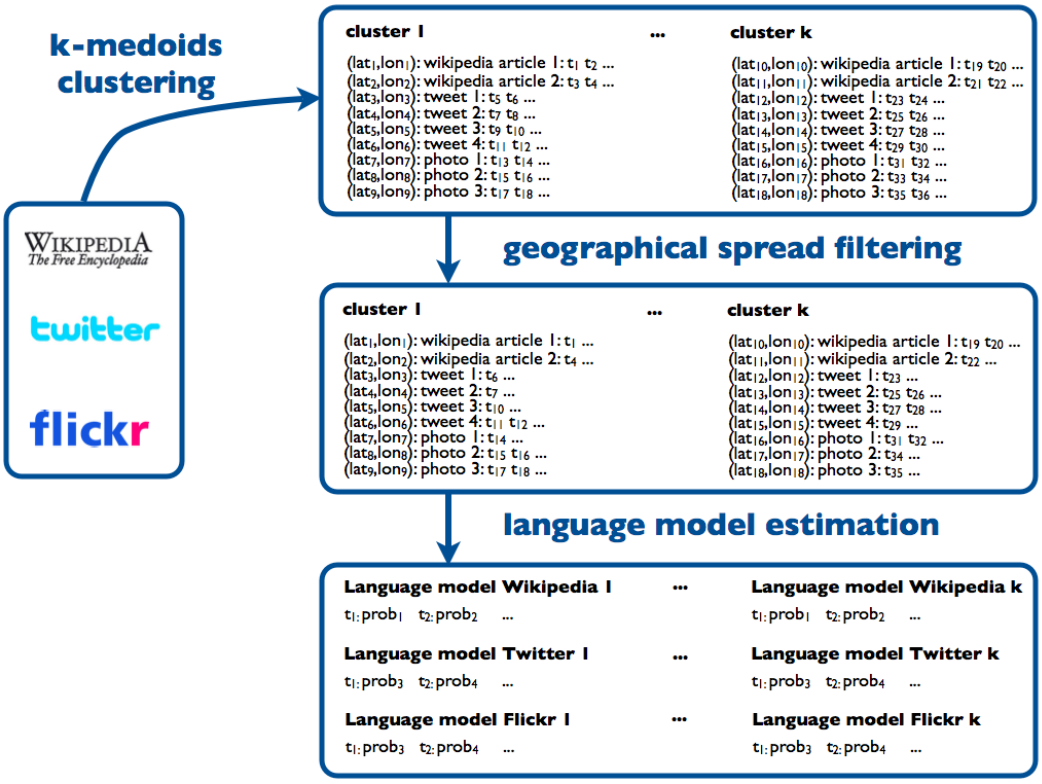


Fig. 1: Overview of the training phase: after clustering the locations of the items in the training set, language models are estimated for each of these clusters. For each considered source (Wikipedia, Twitter and Flickr) a separate language model is estimated.

4.1. Clustering

To cluster the training data, we have used the k-medoids algorithm, which is closely related to the well-known k-means algorithm but is more robust to outliers. Distances are evaluated using the geodesic (great-circle) distance measure. Other authors have used a grid-based clustering or mean-shift clustering [Crandall et al. 2009], but experiments in [Van Laere et al. 2013] have shown k-medoids to be better suited for this task. A grid clustering ignores the fact that certain grid cells contain much more information than others, allowing more precise location estimations in that part of the world. Mean-shift clustering has a similar issue, and results in clusters which are all of approximately the same scale, independent of the amount of training data that is available for that region of the world. In contrast, k-medoids yields smaller clusters when the data density is higher and larger clusters when data is sparser. Figures 3(b) to 3(c) illustrate this difference, which is clearly visible when looking at the coastal regions in the East and the West.

The performance of our method will depend on an appropriate choice of the number of clusters k. If k is set too high, the classifier will not be able to reliably choose the correct area. However, if k is set too low, then the area which is found by the classifier in step 1 will be too large, and it becomes challenging to choose a reasonable location within that area in step 2. We have analysed this effect in detail in [Van Laere et al.

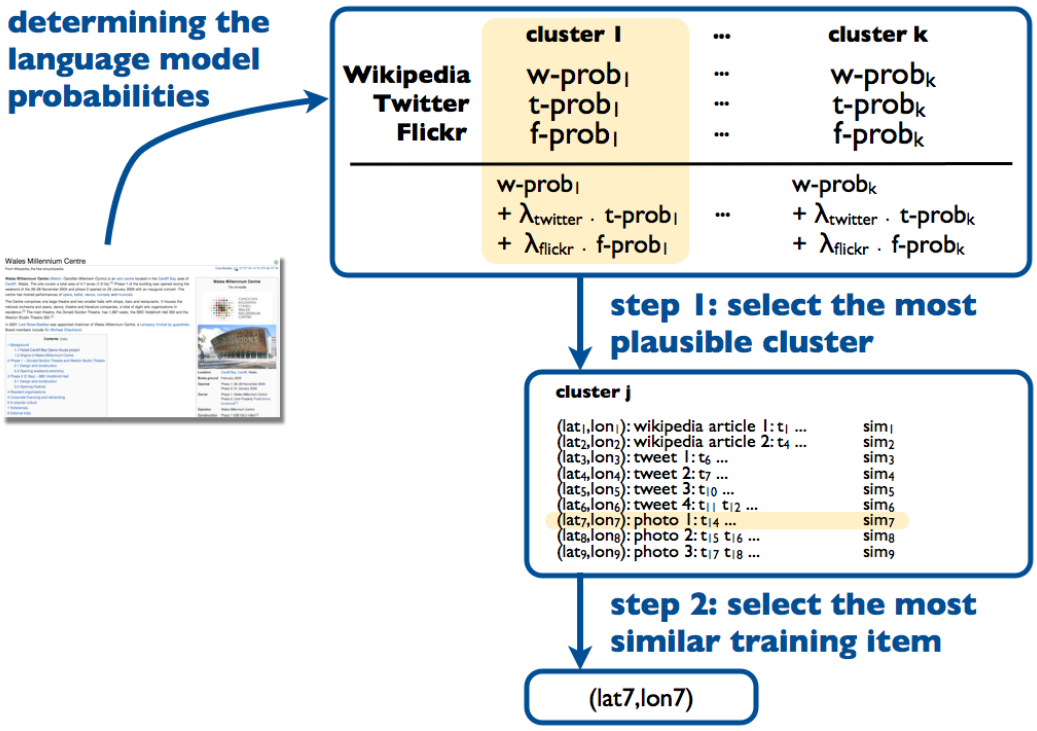


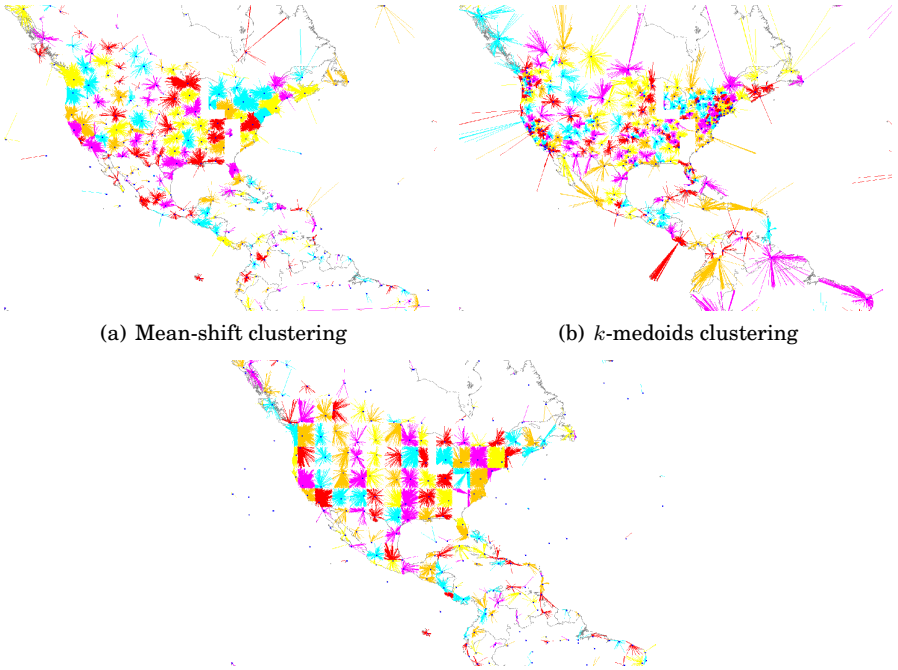
Fig. 2: Overview of the two step methodology to determine coordinates for a Wikipedia article. First, the probabilities from the language models are used to determine the cluster corresponding to the geographic area which is most likely to contain the location of the article. Then, in the second step, the Wikipedia article is compared with training items in the selected cluster to find the most appropriate location in the corresponding area.

2013], in the context of georeferencing Flickr photos. We found that when more training data is available, choosing a reasonable location within a given area can be done much more reliably and it becomes beneficial to choose a larger number of clusters k.

4.2. Feature selection

Many of the tags that have been assigned to photos are not related to their location. By ignoring such terms, our approach can be made more efficient as well as more effective. The task of identifying relevant terms, however, is challenging. In [Van Laere et al. 2013] we compared a number of traditional term selection techniques such as χ^2 and information gain against the *geographic spread filtering*, which was proposed in [Hauff and Houben] for the specific task of georeferencing Flickr photos. Since the latter method clearly outperformed classical term selection techniques, we will adopt it in this paper.

The geographic spread filtering method determines a score that captures to what extent the occurrences of a term are clustered around a small number of locations. Algorithm 1 explains how the geographical spread score is calculated. In the algorithm, merging neighbouring cells is necessary in order to avoid penalizing geographic terms



(a) Mean-shift clustering (b) k-medoids clustering

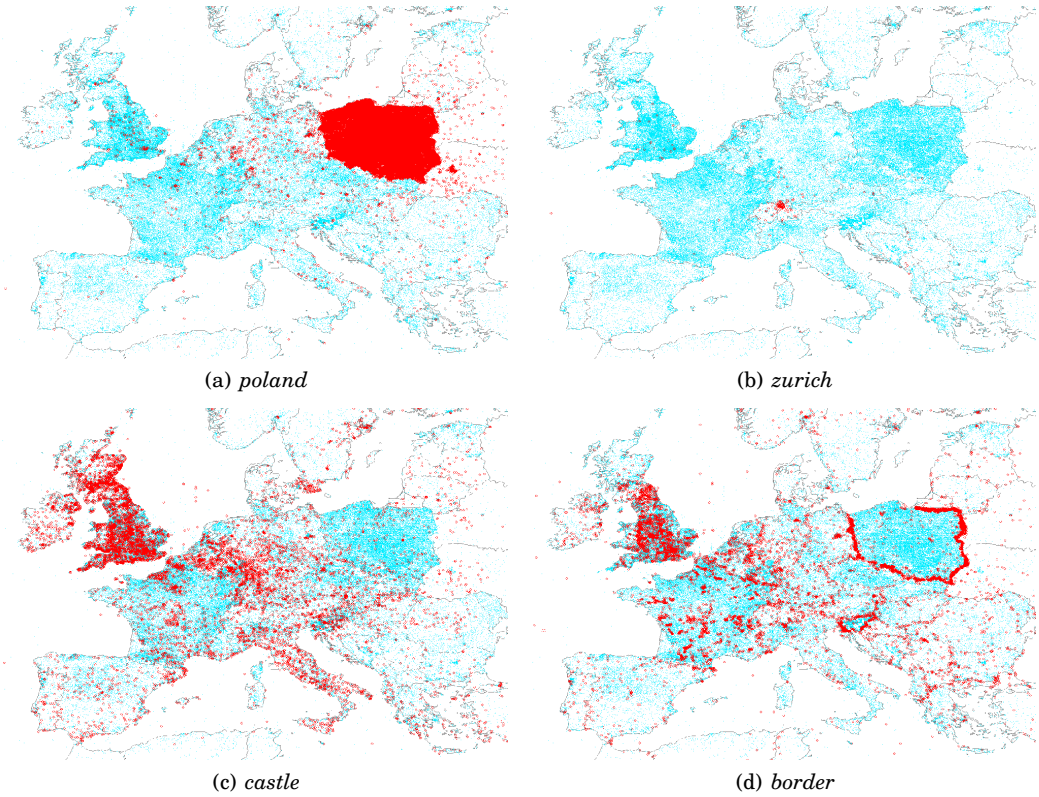
(c) Grid clustering

Fig. 3: Comparison of three different clustering algorithms on the same subset of data.

that cover a wider area. The smaller the score for a term t, the more specific its geographic scope and thus the more it is coupled to a specific location. Figures 4(a) to 4(d) illustrate terms with both a high and low geographical spread score. In our experiments (in Section 5), we rank the features in decreasing order of their geographical spread score.

ALGORITHM 1: Geographic spread filtering.

Place a grid over the world map with each cell having sides of 1 degree latitude and longitude
for each unique term t in the training data **do**
  for each cell $c_{i,j}$ **do**
    Determine $|t_{i,j}|$, the number of training documents containing the term 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
  $score(t) = \frac{count}{\max_{i,j} |t_{i,j}|}$
end for



(a) *poland* (b) *zurich*

(c) *castle* (d) *border*

Fig. 4: Examples of occurrences (highlighted in red) in the Wikipedia training data of two terms with a low geographical spread, *poland* and *zurich*, and two more general terms with a high spread, *castle* and *border*.

4.3. Language modelling

Given a previously unseen document $\mathcal{D}$, we now attempt to determine in which area $a \in \mathcal{A}_k$ it most likely relates. We use a (multinomial) Naive Bayes classifier, which has the advantage of being simple, efficient, and robust. Note that the classes of this classifier are the clusters that have been obtained by using k-medoids, as explained before. As these clusters correspond to geographic areas, the result of applying the Naive Bayes classifier will essentially be an approximate location. A more precise location will then be obtained in the subsequent step. Results from [Serdyukov et al. 2009] have shown good performance for Naive Bayes classifiers. Specifically, we assume that a document $\mathcal{D}$ is represented by a collection of term occurrences $\mathcal{T}$. Using Bayes' rule, we know that the probability $P(a|\mathcal{D})$ that document D was associated with area a is given by

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

Using the assumption that the probability $P(\mathcal{D})$ of observing the terms associated with document $\mathcal{D}$ does not depend on the area a, we find

$$P(a|\mathcal{D}) \propto P(a) \cdot P(\mathcal{D}|a)$$

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

$$P(a|\mathcal{D}) \propto P(a) \cdot \prod_{t \in \mathcal{T}} P(t|a) \tag{1}$$

After moving to log-space to avoid numerical underflow, this leads to identifying the area a^*, to which $\mathcal{D}$ was most likely associated with, by:

$$a^* = \arg\max_{a \in \mathcal{A}} \left(\log P(a) + \sum_{t \in \mathcal{T}} \log P(t|a) \right) \tag{2}$$

In Equation (2), 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:

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

in which $|a|$ represents the number of training documents contained in area a, and N represents the total number of training documents. This reflects the bias of the considered source. For instance, all things being equal, a photo on Flickr has more likely been taken in Western Europe than in Africa. In our setting, in which test data are Wikipedia articles and training data may be taken from Flickr, Twitter and Wikipedia, the justification for the maximum likelihood estimation may appear less strong. However, it should be noted that the geographic bias of Flickr, Twitter and Wikipedia is quite similar, as Figures 5(a) to 5(c) show, illustrating the coverage of our datasets over Africa. In other contexts, where test items may have a different geographic bias, a uniform prior probability could be more appropriate.

To avoid estimating unreliable probabilities, when only a limited amount of information is available, and to avoid a zero probability when $\mathcal{D}$ contains a term that does not occur with any of the documents from area a in the training data, smoothing is needed when estimating $P(t|a)$ in Equation (1). Let O_{ta} be the number of times t occurs in area a. The total term occurrence count O_a of area a is then defined as follows:

$$O_a = \sum_{t \in \mathcal{V}} O_{ta} \tag{4}$$

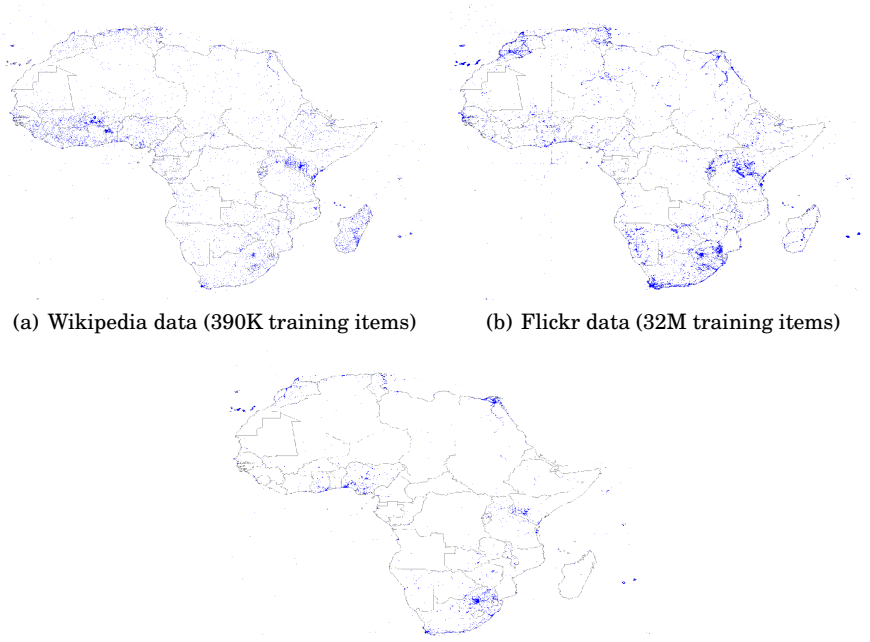
where $\mathcal{V}$ is the vocabulary that was obtained after feature selection, as explained in Section 4.2. When using Bayesian smoothing with Dirichlet priors, we have ($\mu > 0$):

$$P(t|a) = \frac{O_{ta} + \mu \, P(t|\mathcal{V})}{O_a + \mu} \tag{5}$$

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

$$P(t|\mathcal{V}) = \frac{\sum_{a \in \mathcal{A}} O_{ta}}{\sum_{t' \in \mathcal{V}} \sum_{a \in \mathcal{A}} O_{ta}} \tag{6}$$

For more details on smoothing methods for language models, we refer to [Zhai and Lafferty 2001].



(a) Wikipedia data (390K training items) (b) Flickr data (32M training items)

(c) Twitter data (16M training items)

Fig. 5: A qualitative comparison of the data coverage of the different sources of training data over Africa.

4.4. Combining language models

To combine language models estimated from different sources $\mathcal{S}$, e.g. $\mathcal{S} = \{Wikipedia, Flickr, Twitter\}$, (2) can be modified to include weight factors λ_{model}:

$$a^* = \arg\max_{a \in \mathcal{A}_k} \Big(\sum_{model \in \mathcal{S}} \lambda_{model} \cdot \log(P_{model}(a|\mathcal{D})) \Big) \tag{7}$$

The area a maximizing expression (7), using the probabilities produced by all the different models in $\mathcal{S}$, is then chosen as the area that is most likely to contain the given test document $\mathcal{D}$. The parameters λ_{model} can be used to control the influence of each model on the overall probability for a given area a. In particular, if a given model is less reliable, e.g. because it was trained on a small amount of training data or because the training data is known to be noisy (e.g. many tweets talk about places that are not at the associated location of the user), λ_{model} can be set to a small value.

In practice, we compute the models in memory. This makes it unfeasible to store the probabilities for each of the k areas for each test document and for each of the language models, at the same time. To cope with this, we compute each model separately and store the top-100 areas with the highest probabilities for each test document $\mathcal{D}$ in the given model. By doing so, probabilities $P_{model}(a|D)$ for certain areas $a \in \mathcal{A}_k$ will be missing in Equation (7), which we estimate as follows:

$$P^*_{model}(a|\mathcal{D}) = \begin{cases} P_{model}(a|\mathcal{D}) & \text{if } a \text{ in top-100} \\ \min_{a' \text{ in top-100}} P_{model}(a'|\mathcal{D}) & \text{otherwise} \end{cases}$$

4.5. Location estimation

We consider three different ways of choosing a precise location, once a suitable area a has been found.

4.5.1. Medoid. The most straightforward solution is to choose the location of the medoid m_a, defined as:

$$m_a = \underset{x \in \mathit{Train}(a)}{\arg\min} \sum_{y \in \mathit{Train}(a)} d(x, y) \tag{8}$$

where $\mathit{Train}(a)$ represents the set of training documents located in area a and $d(x, y)$ is the geodesic distance between the locations of documents x and y. This comes down to the idea of selecting the location of the training document that is most centrally located among all documents in a. While this method is rather straightforward, it can still give reasonable location estimates when the number of clusters k is sufficiently large.

4.5.2. Jaccard similarity. Another solution consists of returning the location of the most similar training document in terms the Jaccard measure:

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

where we identify a document with its *set* of terms, *without* considering feature selection. Using feature selection here would be harmful as there may be rare terms (e.g. the name of a local restaurant) or terms without a clear geographic focus (e.g. *castle*) that could be very helpful in finding the exact location of a document.

4.5.3. Lucene. A third and final solution is to use Apache Lucene. The fact that Jaccard similarity does not take multiple occurrences of a given feature into account is not an issue when considering Flickr tags. However, when the test and/or training data consists of Wikipedia documents, this could potentially be a shortcoming. Also, [Krippner et al.] has shown that Lucene can be effective in finding similar Flickr photos as well. To find the training document in area a that is most similar to $\mathcal{D}$, we use Lucene search with its default scoring mechanism[15].

5. EXPERIMENTAL EVALUATION

5.1. Methodology

In this section, we discuss the results of experiments addressing the research questions stated in Section 1. In Sections 5.2 and 5.4, we establish baseline results for both of the Wikipedia test sets. To this end, we georeference the test documents using only language models trained using other Wikipedia documents. Subsequently, we evaluate the results when using language models trained only using Flickr or Twitter data. After describing the baseline approach, we discuss the effect of combining different language models in Sections 5.3 and 5.5. Sections 5.6 to 5.9 provide detailed insights in the results. Finally, in Section 5.10, we compare the results of our method on both test sets against Yahoo! Placemaker, which is a gazetteer-based service for georeferencing arbitrary web documents, and a against a method using Geonames.

Baseline approach. The approach outlined in Section 4 requires several parameters, including the number of features to select and a parameter controlling the amount of smoothing. A detailed analysis of the influence of each of these parameters is beyond

[15]For details on this scoring function, we refer to `http://lucene.apache.org/core/3_6_1/api/all/org/apache/lucene/search/Similarity.html`

the scope of this paper, as a detailed study was conducted in [Van Laere et al. 2013]. To focus on the core research questions of this paper, we have therefore fixed the following parameters:

— the number of features used by the feature selection algorithm (Section 4.2) was set to 250 000 features for the Wikipedia training data, and 150 000 features for the Flickr and Twitter training sets.
— the smoothing parameter μ, used for the Bayesian smoothing with Dirichlet priors in the language models (Section 4.3), was set to 15 000.

We evaluate the results of the experiments using the following metrics:

(1) The **accuracy** of the classifier for the given clustering. This is given by $\frac{P}{P+N}$ where P is the number of test documents that have been assigned to the correct cluster and N is the number of documents that have not.
(2) For each test document, the distance error is calculated as the distance between the predicted and the true location. The **median error** distance is used as an evaluation metric. This allows us to observe, using a single value, the overall scale of the errors made for a given test collection.
(3) From the aforementioned error distances, following [Hays and Efros 2008] we also calculate the percentage of the test items that were predicted within 1 m, 10 m, 100 m, 1 km, 10 km, 100 km and 1000 km of their true location, which we refer to as **Acc@Kkm**, with K being the threshold distance in kilometer.

Note that the accuracy of the classifier is only meaningful to compare the relative performance of different versions of the Naive Bayes classifier which are based on the same clustering. It is not meaningful as a general measure to evaluate how well we can georeference Wikipedia articles.

5.2. Baseline results for the W&B dataset

Table I presents the baseline results on the W&B dataset (Section 3.1). The optimal results are highlighted in light-blue. The values for the approach taken by [Roller et al. 2012] are gathered by parsing and evaluating the log files as provided by the authors. These results were obtained using a *k-d*-based clustering of the training data and finding the cluster which is most similar to the Wikipedia document in terms of the Kullback-Leibler divergence.

Overall, the approach taken by [Roller et al. 2012] achieves better results at the higher error distance thresholds (most notably at 10 km and 100 km), whereas our approach achieves better results at the lower thresholds (most notable at 0.1 km and 1 km), both when using the same training data from Wikipedia and when using training data from Flickr. This difference with [Roller et al. 2012] can be explained as follows. By returning the centre-of-gravity of the area that was found by the classifier, [Roller et al. 2012] takes a rather cautious approach, as the centre is reasonably close to most elements in the area. Our method, on the other hand, tries to identify the exact location within an area; cases for which this is successful explain why we do better at the lower thresholds and cases for which this step fails are partially responsible for the worse results at the higher accuracy levels. Differences in the clustering method and the use of Kullback-Leibler instead of Naive Bayes may also lead to some changes in the results. For example, when using fewer clusters, more emphasis is put on the similarity search step which in general is more errorprone. This effect may explain why using 50000 clusters yields better results than using 2500 clusters at the 1 km and 10 km thresholds for the Wikipedia training data.

Table I: Comparison between the results from [Roller et al. 2012] and our framework from Section 4 when trained using Wikipedia, Flickr and Twitter documents separately (W&B dataset). The different k-values represent the number of clusters used while the maximal values across all three models in the table are highlighted for each of the different accuracies, as well as the minimal median error.

Wikipedia	Roller et al	$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error	13.36 km	22.25 km	19.26 km	19.13 km	19.58 km
Accuracy	N/A	64.18%	49.02%	35.72%	26.31%
Acc@0.001 km	0.1%	1.1%	1.06%	1.03%	0.99%
Acc@0.01 km	0.1%	1.15%	1.12%	1.09%	1.05%
Acc@0.1 km	0.16%	1.58%	1.58%	1.55%	1.48%
Acc@1 km	3.53%	5.62%	6.05%	6.28%	6.34%
Acc@10 km	42.75%	32.42%	35.58%	36.19%	36.01%
Acc@100 km	86.54%	79.34%	80.1%	79.01%	77.77%
Acc@1000 km	97.42%	95.73%	95.6%	94.97%	94.21%
Flickr 32M		$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error		51.14 km	48.94 km	50.77 km	53.32 km
Accuracy		44.26%	29.29%	20.64%	15.22%
Acc@0.001 km		0.02%	0.02%	0.02%	0.03%
Acc@0.01 km		0.21%	0.2%	0.19%	0.16%
Acc@0.1 km		2.61%	2.39%	2.14%	1.88%
Acc@1 km		11.25%	10.15%	9.18%	8.4%
Acc@10 km		26.26%	26.75%	25.94%	25.11%
Acc@100 km		62.78%	63%	62.24%	61.2%
Acc@1000 km		88.6%	87.78%	87.09%	86.35%
Twitter 16M		$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error		350.58 km	406.7 km	427.58 km	469.68 km
Accuracy		24.02%	14.61%	9.72%	7.14%
Acc@0.001 km		0%	0.01%	0%	0%
Acc@0.01 km		0.01%	0.01%	0.02%	0.02%
Acc@0.1 km		0.04%	0.07%	0.11%	0.16%
Acc@1 km		0.66%	1.32%	1.71%	1.94%
Acc@10 km		8.56%	11.82%	12.69%	12.74%
Acc@100 km		36.31%	36.01%	35.34%	34.55%
Acc@1000 km		61.05%	59.23%	58.36%	57.05%

Interesting to see in Table I is that the highest Acc@0.1 km and Acc@1 km values are obtained using a language model trained using 32M Flickr photos, with the difference at 1 km being especially pronounced. This result is all the more remarkable because the Flickr model cannot be used to its full potential given that the W&B dataset only supports the use of unigrams (see Section 3.1). Even though the results from using a model trained on 16M Twitter documents are worse than the two other models, it is noteworthy that is still allows to locate 1.94% of the Wikipedia documents within 1 km of their true location.

Finally, Table I also mentioned the classifier accuracies for the results based on our framework. First, note that these accuracies are only meaningful when comparing between configurations based on the same number of clusters. In particular, it is interesting to see that for each k, the accuracy of the classifier trained on Wikipedia performs

substantially better than the classifier trained on Flickr (which, in turn, performs substantially better than the classifier trained on Twitter). This means that the better performance on the Acc@1 km measure in the case of Flickr is due to being more effective at finding a suitable coordinate within the discovered cluster (i.e. step 2 of the methodology), which in this case offsets the worse performance by the classifier in step 1.

5.3. Combining language models using training data from social media (W&B dataset)

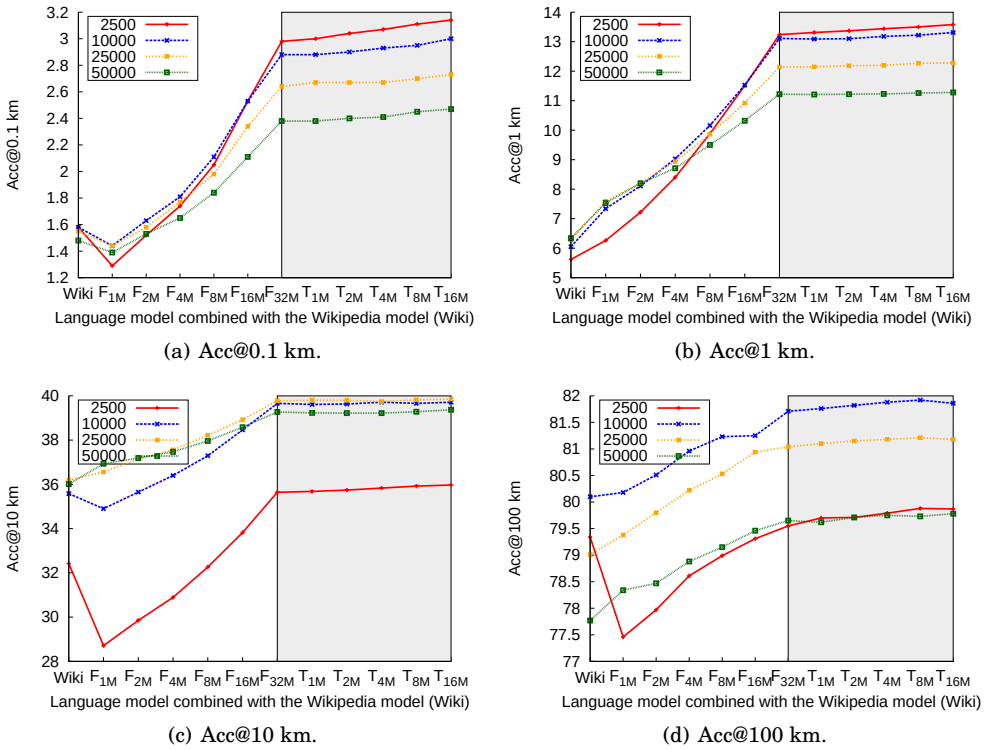
5.3.1. Wikipedia + Flickr + Twitter. Figure 6 shows the result of combining the language models from Wikipedia, Flickr and Twitter, using $\lambda_{flickr} = 0.5, \lambda_{twitter} = 0.15$[16]. The graphs consist of two parts. On the left, we start with a pure Wikipedia model (*Wiki*) and combine this model with different Flickr models trained using a gradually increasing amount of training Flickr photos (up to 32M) (F_{1M} to F_{32M}). In the center of the graphs, where the shaded area begins, we start with the $Wiki + F_{32M}$ model and continue to combine with language models from Twitter trained using up to 16M documents (T_{1M} to T_{16M}). The location estimate returned for each test document is the location from the most similar training item overall (i.e. a Wikipedia document[17], a Flickr photo or a Twitter document) in the cluster selected by the classifier. As for the results in Table I, the Jaccard similarity is used for this purpose. As before, results are evaluated on the W&B test data and the number of clusters is varied from 2500 to 50000.

The combination $Wiki + F_{32M}$ in Figure 6(a) only shows an increase of 1.4%, which is somewhat dissappointing. We assume this is partially due to the fact that not all test documents from the W&B dataset correspond to a spot. For instance, it does not make sense to estimate an exact coordinate for a test document such as "Sante Fe Trail"[18].

As Figure 6(b) to Figure 6(d) show, the optimal number of clusters (k) to use depends on the accuracy level we aim for, although choosing $k = 10000$ seems to be a reasonable choice in all cases. Moreover, note that as more training data is considered, the optimal number of clusters tends to decrease. This is particularly obvious in Figure 6(a), where choosing 2500 clusters is the worst of the four considered choices when 1M Flickr photos are considered, but it is the best choice with 32M Flickr photos. It should also be noted that for Acc@0.1 km and Acc@1 km the optimal number of clusters is lower than for Acc@10 km and Acc@100 km. This is due to the fact in cases where the predicted coordinates are 100m or even within 1 km of the true location, there is often a training item which is very similar to the Wikipedia article. Due to its high similarity, this training item will be selected in step 2, provided that it is contained in the cluster that has been selected in step 1. It thus becomes beneficial to choose a lower number of clusters to minimize classifier errors in step 1.

All the graphs also show a deterioration of the results when extending the Wikipedia training data with 1M Flickr photos. When too few Flickr photos are available, probabilities in the language models can apparently not be estimated in a reliable way. In such as case, linearly interpolating the language model from Wikipedia with a poorly performing language model from Flickr will lead to worse results than simply using the model from Wikpedia, as is illustrated in Table II. This table shows the individual accuracies for the $Wiki$ and F_{1M} model, as well as the combined $Wiki + F_{1M}$ model, that are used in Figure 6.

[16] A detailed discussion of the influence of these parameter values follows in Section 5.7.
[17] In fact, we only use the title of Wikipedia documents during similarity search. We will come back to this in Section 5.9.
[18] `http://en.wikipedia.org/wiki/Santa_Fe_Trail`



(a) Acc@0.1 km.

(b) Acc@1 km.

(c) Acc@10 km.

(d) Acc@100 km.

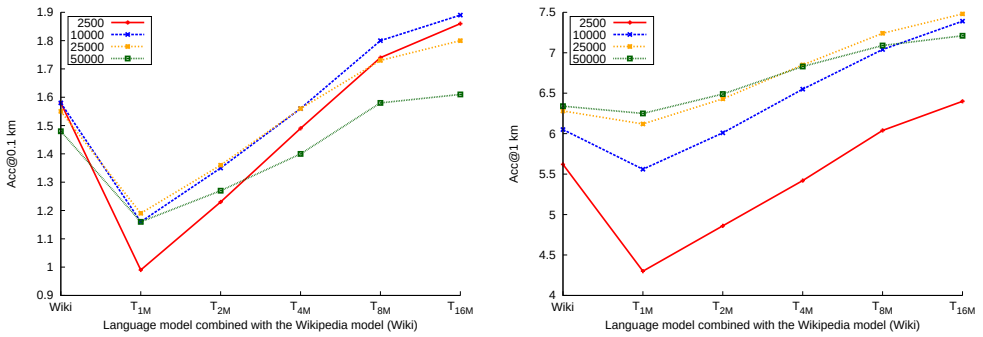
Fig. 6: Percentage of the test documents located within different error distances on the W&B test set, when combining the language model from Wikipedia with Flickr (on the left side) and subsequently with Twitter models (in the shaded area) trained over an increasing amount of information and for different numbers of clusters k.

Table II: Individual accuracies for the $Wiki$ and F_{1M} model, as well as the combined $Wiki + F_{1M}$ model, that are used on the W&B test set in Figure 6.

Model	Acc@0.1 km	Acc@1 km	Acc@10 km	Acc@100 km
$Wiki$	1.58%	5.62%	32.42%	79.34%
F_{1M}	0.58%	3.74%	14.08%	43.87%
$Wiki + F_{1M}$	1.29%	6.26%	28.71%	77.46%

Looking at the right side of the graphs, it seems that the Twitter data is nearly obsolete: only minor improvements are achieved. It should however be noted that the number of georeferenced tweets made available each day is substantially larger than the number of georeferenced Flickr photos, which offers opportunities to training language models from hundreds of millions of tweets, which would likely allow for a more substantial contribution. However, it should be noted that many georeferenced tweets do not describe the current location of the user, and simply increasing the number of tweets in the training data may not be sufficient. One idea might be to train a classifier that can detect tweets which describe a location and then limit the training set to such tweets.

5.3.2. Wikipedia + Twitter. Using a similar configuration as the previous experiment, we combine the Wikipedia language model with Twitter models trained over up to 16M documents. The results are shown in Figure 7.



(a) Acc@0.1 km. (b) Acc@1 km.

Fig. 7: Percentage of the test documents located within error distances of 0.1 km and 1 km on the W&B test set, when combining the language model from Wikipedia with Twitter models trained over an increasing amount of information and for different numbers of clusters k.

As Twitter documents are generally less informative than the tags associated to Flickr photos, the deterioration on the results when using too few training documents is even more pronounced in Figure 7(a) than it was in Figure 6(a). Still, when sufficient Twitter data becomes available, significant improvements[19] can be obtained in comparison with only using Wikipedia training data.

5.4. Baseline results for the spot dataset

Figure 6 showed that adding Twitter and especially Flickr has the potential to substantially improve the results. However, as we discussed in Section 3.5, the W&B test data ignores word ordering, which is a disadvantage for our approach because Flickr tags and Twitter terms may correspond to concatenations of terms in a Wikipedia document. Therefore, and also in view of the shortcomings described in Section 3.1, we propose an evaluation based on another test set.

We establish the baseline results using the spot dataset consisting of 21 839 Wikipedia test documents, in combination with a filtered training set consisting of 376 110 Wikipedia documents, as described in Section 3.2. Table III depicts the results of our framework, using the same parameter settings as for Table I. Again, the maximal values across all three models in the table are highlighted for each of the different accuracies, as well as the minimal median error. The results presented under Roller et al have been obtained by running their textgrounder framework [20] on this dataset using a grid configuration of 0.1 degree per cell side.

As could be expected given the nature of the test data, the accuracies presented in Table III are much higher than those for the W&B test set in Table I. A relatively large fraction of the documents can be localized within 1 km of their true location (35.73%

[19]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. The results are significant with a p-value $< 2.2 \times 10^{-16}$.
[20]The textgrounder framework can be downloaded from `https://github.com/utcompling/textgrounder/wiki/RollerEtAl_EMNLP2012`.

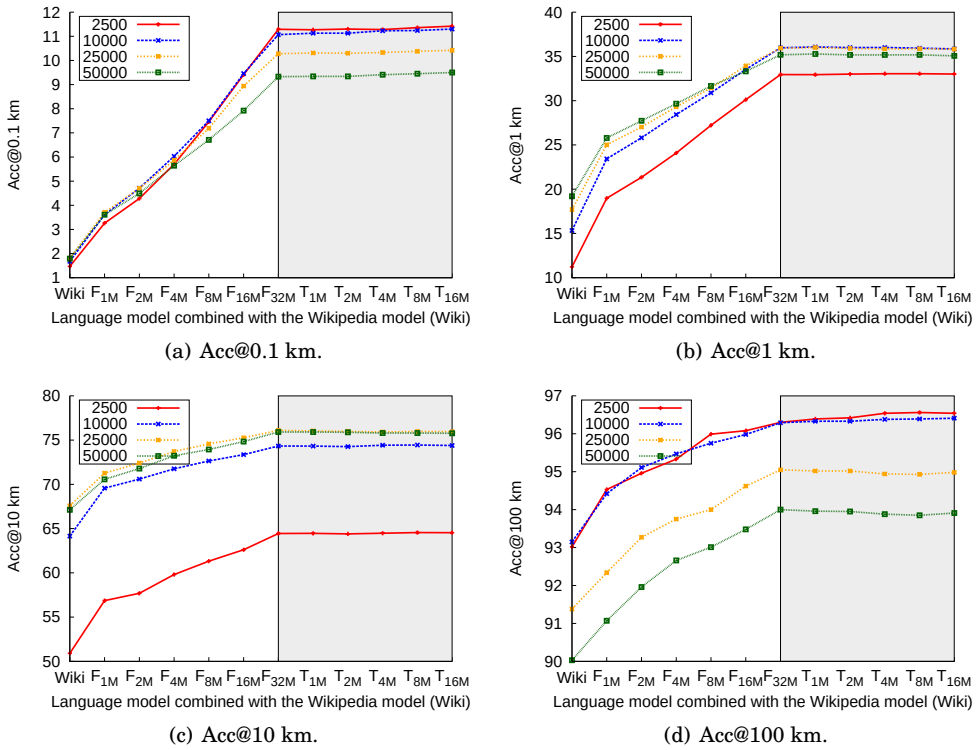
Table III: Comparison between the results from [Roller et al. 2012] and our framework from Section 4 when trained using Wikipedia, Flickr and Twitter documents separately (spot dataset). The different k-values represent the number of clusters used while the maximal values across all three models in the table are highlighted for each of the different accuracies, as well as the minimal median error.

Wikipedia	Roller et al	$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error	8.12 km	9.68 km	5.86 km	4.64 km	4.17 km
Accuracy	N/A	70.11%	57.99%	46.21%	36.32%
Acc@0.001 km	0.02%	0.34%	0.34%	0.33%	0.33%
Acc@0.01 km	0.02%	0.38%	0.39%	0.38%	0.38%
Acc@0.1 km	0.10%	1.47%	1.68%	1.81%	1.79%
Acc@1 km	4.17%	11.23%	15.33%	17.7%	19.2%
Acc@10 km	53.11%	50.91%	64.15%	67.58%	67.12%
Acc@100 km	75.98%	93.02%	93.15%	91.38%	90.03%
Acc@1000 km	92.36%	98.02%	98.34%	97.77%	97.35%
Flickr 32M		$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error		3.7 km	2.6 km	2.44 km	2.5 km
Accuracy		76.97%	63.24%	51.6%	42.35%
Acc@0.001 km		0.06%	0.07%	0.05%	0.06%
Acc@0.01 km		0.95%	0.94%	0.92%	0.84%
Acc@0.1 km		11.52%	11.5%	11.05%	10.49%
Acc@1 km		33.24%	35.45%	35.73%	35.17%
Acc@10 km		63.4%	71.32%	72.44%	71.29%
Acc@100 km		96.47%	96.47%	95.98%	95.48%
Acc@1000 km		98.84%	98.77%	98.63%	98.5%
Twitter 16M		$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error		25.21 km	24.47 km	29.57 km	35.81 km
Accuracy		43.03%	26.83%	18.3%	13.36%
Acc@0.001 km		0%	0%	0%	0%
Acc@0.01 km		0.01%	0.01%	0.01%	0.02%
Acc@0.1 km		0.15%	0.24%	0.23%	0.33%
Acc@1 km		3.14%	6.21%	7.75%	8.28%
Acc@10 km		29.52%	36.98%	36.07%	33.39%
Acc@100 km		72.66%	69.69%	66.7%	64.17%
Acc@1000 km		94.91%	94.23%	93.1%	92.88%

as opposed to 11.25%). Using the Flickr model results in a median error of 2.44 km, compared to 4.64 km for the Wikipedia model. This Flickr model outperforms the two other models at the classification accuracies and at all threshold accuracies except Acc@0.001 km. Again, the results from the Twitter model are worse, except for the fact that 8.28% of the test set can be localised within 1 km of their true location.

5.5. Combining language models using training data from social media (spot dataset)

5.5.1. Wikipedia + Flickr + Twitter. Similar to the experiment carried out on the W&B dataset in Section 5.3, we combine the language models obtained from Wikipedia, Flickr and Twitter and evaluate using the spot test collection of 21 839 Wikipedia documents. The results are presented in Figure 8.



(a) Acc@0.1 km. (b) Acc@1 km.

(c) Acc@10 km. (d) Acc@100 km.

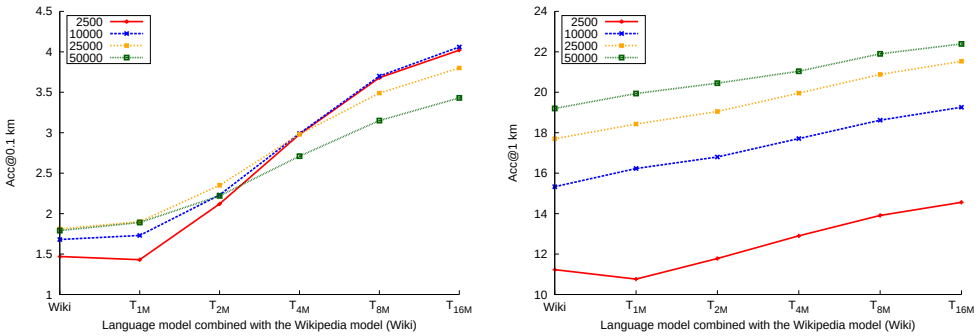
Fig. 8: Percentage of the test documents located within different error distances on the spot test set, when combining the language model from Wikipedia with Flickr (on the left side) and subsequently with Twitter models (in the shaded area) trained using an increasing amount of information and for different numbers of clusters k.

Overall, the relative performance of the different configurations in Figure 8 is qualitatively similar to the results for the W&B test set in Figure 6, although the magnitude of the improvements is much higher. Given this better performance of the Flickr models, Twitter does not seem to be helpful at all anymore.

5.5.2. Wikipedia + Twitter. Figure 9 presents the results of combining the Wikipedia language model with Twitter models trained over different amounts of data. In contrast to Figure 7, these graphs clearly demonstrate that improvements can be obtained at error margins of 1 km and below by extending the Wikipedia model with only Twitter data. This is remarkable given the difference in structure between a Wikipedia training document and a Twitter message. Also, the deteriorating effect for small amounts of training data is only slightly noticed when using $k = 2500$ clusters.

5.6. Training data analysis

It may seem that, by adding for example 32 million Flickr photos to the training data, we are increasing the number of training items by an order of magnitude. However, the amount of textual information that is actually added is comparable to the initial Wikipedia training data, as can be seen in Table IV. This is because a Wikipedia training document generally provides a significantly larger amount of textual information (mean of $\simeq$387 tokens) compared to a Flickr training photo (mean of $\simeq$8 tokens). A



(a) Acc@0.1 km. (b) Acc@1 km.

Fig. 9: Percentage of the test documents located within error distances of 0.1 km and 1 km on the spot test set, when combining the language model from Wikipedia with Twitter models trained using an increasing amount of information and for different numbers of clusters k.

Table IV: Comparison of the number of tokens in each of the different training sets (before and after feature selection (FS)). The number of unique tokens is reported, along with the total number of token occurrences, before and after feature selection (see Section 4.2).

Dataset	# items	Unique tokens	Total before FS	Total after FS
Wikipedia	390 574	2 817 660	151 325 949	53 134 473
Flickr	1 000 000	563 707	8 395 186	4 829 997
	2 000 000	972 484	17 163 282	8 705 356
	4 000 000	1 732 867	35 597 819	14 667 027
	8 000 000	3 087 690	71 395 087	25 474 723
	16 000 000	5 362 086	143 592 337	44 930 446
	32 000 000	9 269 494	279 109 442	79 968 463
Twitter	1 000 000	2 678 380	18 184 767	7 256 169
	2 000 000	4 667 761	35 581 577	13 796 968
	4 000 000	8 055 391	69 235 192	26 335 231
	8 000 000	13 823 337	136 203 621	51 779 462
	16 000 000	23 077 992	264 632 000	99 964 037
TwitterHashtags	1 000 000	454 884	1 514 359	466 028
	2 000 000	805 521	3 083 544	989 408
	4 000 000	1 428 268	6 188 443	1 937 579
	8 000 000	2 532 145	12 298 065	3 765 776
	16 000 000	4 529 912	24 132 042	6 770 206

similar argument holds for Twitter documents with a mean of $\simeq$16 tokens. Table IV provides further details on the unique tokens (words) that occur in the datasets, the total number of tokens in the initial datasets, and the number of tokens that remained after feature selection (see Section 4.2).

In addition to our standard Twitter dataset, we included the *TwitterHashtags* variant in Table IV, which consists of only the hashtags encountered in the Twitter document. As can be seen from the table, the number of token occurrences is significantly reduced in this dataset, with a mean of $\simeq$0.4 tokens per document. We have omitted

Table V: Comparison of the percentage of the test documents located within error distances of 0.1 km and 1 km on the spot test set, when combining the language model from Wikipedia with Twitter models, containing all terms and only Hashtags, trained using an increasing amount of information and for different numbers of clusters k.

k	2500	10000	25000	50000	2500	10000	25000	50000
Acc@0.1 km	Twitter				TwitterHash			
Wiki	1.47%	1.68%	1.81%	1.79%	1.47%	1.68%	1.81%	1.79%
T_{1M}	1.43%	1.73%	1.90%	1.89%	1.41%	1.73%	1.88%	1.88%
T_{2M}	2.12%	2.23%	2.35%	2.22%	2.10%	2.19%	2.32%	2.18%
T_{4M}	2.98%	2.99%	2.98%	2.71%	2.94%	2.99%	2.91%	2.62%
T_{8M}	3.68%	3.70%	3.49%	3.15%	3.65%	3.69%	3.43%	3.01%
T_{16M}	4.02%	4.06%	3.80%	3.43%	4.00%	4.02%	3.74%	3.28%
Acc@1 km	Twitter				TwitterHash			
Wiki	11.23%	15.33%	17.70%	19.20%	11.23%	15.33%	17.70%	19.20%
T_{1M}	10.76%	16.23%	18.43%	19.94%	10.73%	16.15%	18.27%	19.87%
T_{2M}	11.78%	16.80%	19.05%	20.45%	11.76%	16.72%	18.88%	20.34%
T_{4M}	12.90%	17.71%	19.96%	21.04%	12.88%	17.69%	19.84%	20.91%
T_{8M}	13.91%	18.62%	20.88%	21.90%	13.97%	18.59%	20.70%	21.61%
T_{16M}	14.56%	19.26%	21.53%	22.39%	14.52%	19.20%	21.34%	22.25%

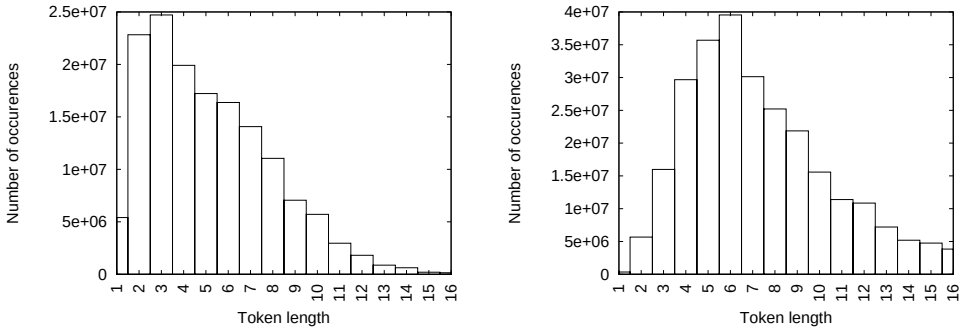
the results of this variant in the previous sections, as this dataset produces similar results as the standard Twitter dataset, as can be seen in Table V. This is interesting by itself, as the amount of information used to achieve those results is less than 7.5% of the original Twitter dataset.

Figure 10 further summarises some characteristics of the training data, comparing the length of tokens in the different training sets. Note that the mode in Figures 10(b) and 10(c) is higher than in Figures 10(a) and 10(d), which is consistent with the idea that tags are more descriptive and therefore likely to be longer, and the view that tags often are concatenation of several words. The latter point is more pronounced in the case of Twitter than in Flickr, as the distribution in Figure 10(c) is skewed more towards higher token lengths. The slight difference between Figures 10(a) and 10(d) in the proportion of tokens of lengths 2 and 3 may be due to the tendency to omit determiners in tweets.

5.7. Influence of the λ_{model} parameters when combining different models

As outlined in Section 4.4, the parameter λ_{model} which weighs the different models in Equation (7) can play an important role in the results. In Figure 11(a), we show, on the spot dataset, for each datapoint the λ_{flickr} value that is optimal when combining the Wikipedia model with each of the Flickr models. As can be expected, the models obtained by using a larger amount of training data prove to be more reliable, allowing to increase the weight λ_{flickr}. The accuracy value for $k = 2500$ at F_{1M} is 75.71% while it increases to 82.15% at F_{32M}.

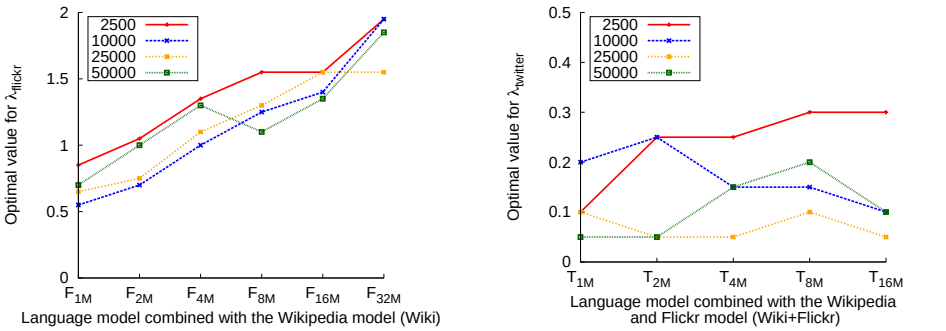
Figure 11(b), shows for each datapoint the $\lambda_{twitter}$ value that was optimal when combining the Wikipedia+F_{32M} model with each of the Twitter models. Unsurprisingly, the $\lambda_{twitter}$ values are low, even for a relatively large amount of training data. For $k = 2500$, it seems that the results become more reliable for more training data. The accuracy value for $k = 2500$ at T_{1M} is 78.58% while it only increases to 79.01% at T_{16M}. In fact, it is hard to discover any meaningful trend in Figure 11(b), which serves as another illustration that the Twitter data in its current form is much harder to use effectively than the Flickr data.



(a) Wikipedia (b) Flickr

(c) Twitter Hashtags (d) Twitter All

Fig. 10: Histograms of the distribution of the word length (up to 16 characters) for the different sources of information, without feature selection.



(a) Optimal λ_{flickr} for the spot test set (b) Optimal $\lambda_{twitter}$ for the spot test set

Fig. 11: Comparing the optimal values for λ under different configurations

Table VI: Comparison of the results when using different n-grams on the spot dataset. The language model was obtained by combining the Wikipedia, Flickr F_{32M} and Twitter T_{16M} models ($k = 10000$, $\lambda_{flickr} = 2, \lambda_{twitter} = 0.1$).

	1-gram	2-gram	3-gram	4-gram	5-gram
Accuracy	67.05%	69.71%	69.90%	69.92%	69.90%
Median Lucene	3.22 km	2.97 km	2.98 km	2.98 km	2.98 km
Median Similarity	2.31 km	2.05 km	2.03 km	2.02 km	2.02 km

Table VII: Comparing the results of retrieving the most similar training item using Lucene and Jaccard similarity. These results are shown, using the combined Wikipedia + Flickr (32M) + Twitter (16M) language model and $k = 10000, \lambda_{flickr} = 0.5, \lambda_{twitter} = 0.15$, for both the W&B (left) and spot (right) test set.

	W&B test set		Spot test set	
	Lucene	Jaccard	Lucene	Jaccard
Median Error	16.37 km	17.03 km	3.28 km	2.37 km
Accuracy	50.89%		66.87%	
Acc@0.001 km	0.55%	0.21%	0.18%	0.07%
Acc@0.01 km	0.75%	0.42%	0.84%	0.91%
Acc@0.1 km	2.71%	3.00%	8.38%	11.3%
Acc@1 km	10.64%	13.31%	29.65%	35.85%
Acc@10 km	39.62%	39.71%	74.92%	74.39%
Acc@100 km	82.15%	81.86%	96.44%	96.41%
Acc@1000 km	96.37%	96.34%	99.03%	99.04%

5.8. n-grams and similarity search

Table VI illustrates the impact of concatenating words from the Wikipedia training documents to make them compatible with the Flickr and Twitter training data. In this table, we compare the performance of our method when concatenations are not allowed, or limited to a fixed number of consecutive words. We used the spot test set for this table, while the language model was obtained by combining the Wikipedia, Flickr F_{32M} and Twitter T_{16M} models ($k = 10000$, $\lambda_{flickr} = 2, \lambda_{twitter} = 0.1$). The results present both the Lucene similarity and Jaccard similarity to obtain the location estimates for the test documents. As can be seen from the table, allowing sequences of a few words to be concatenated yields higher accuracies and lower median errors, for both similarity methods. In all the experiments for this paper, we used $n = 3$ as the effect of longer sequences does not seem to influence the results substantially.

Table VI shows that the median errors obtained using Jaccard similarity are lower than when using Lucene. Table VII compares using Lucene and Jaccard similarity in more detail. These results are based on the combined Wikipedia + Flickr (32M) + Twitter (16M) language model and $k = 10000, \lambda_{flickr} = 0.5, \lambda_{twitter} = 0.15$. Results for both the W&B (left) and spot (right) test set are reported, while the best results for both datasets are highlighted. As can be seen in the table, the results are somewhat mixed.

5.9. Similarity search: full content vs. title only

In many cases, the title of a Wikipedia document about a place will be the name of that place. If enough training data from Flickr are available, photos about that place will often be in the training data, and we may try to match the title of the Wikipedia page to

Table VIII: Comparison between using full wikipedia documents and using titles during similarity search

W&B test set (48 566 items)	$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error Title only	22.43 km	17.14 km	16.85 km	17.4 km
Median Error Full	24.8 km	19.84 km	18.83 km	18.76 km
Acc@0.001 km Title Only	0.17%	0.23%	0.27%	0.31%
Acc@0.001 km Full	0.52%	0.54%	0.58%	0.59%
Acc@1 km Title Only	13.24%	13.11%	12.14%	11.22%
Acc@1 km Full	3.31%	4.39%	5.23%	5.68%
Spot test set (21 839 items)	$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error Title only	3.54 km	2.34 km	2.17 km	2.16 km
Median Error Full	9.26 km	5.40 km	4.00 km	3.31 km
Acc@0.001 km Title Only	0.07%	0.08%	0.06%	0.06%
Acc@0.001 km Full	0.18%	0.20%	0.21%	0.27%
Acc@1 km Title Only	32.95%	35.98%	35.94%	35.19%
Acc@1 km Full	8.65%	13.75%	17.73%	21.01%

the photos in the training data, ignoring the body of the document. Table VIII shows the result of using only the page titles for the Jaccard similarity search, compared to using the full document. It should be noted that in the classification step, the full document is used in both cases. The results have been obtained using the combination of the Wikipedia and the F_{32M} Flickr model ($\lambda_{flickr} = 0.5$). We observe the change in median error and Acc@0.001km and Acc@1km, as these are the values that are mainly influenced by the similarity search, whereas the results for the thresholds above 1 km are mostly influenced by the performance of the classifier. As can seen in the table, we observe a substantial improvement when restricting to the title of a Wikipedia page for both datasets. For all the experiments in this paper, the similarity search was carried out using only the Wikipedia page title.

5.10. Comparing the results to gazetteer based methods

In this section we investigate how the performance of our method relates to two gazetteer based methods. First, we compare the result of our combined model (Wikipedia, Flickr and Twitter, $\lambda_{flickr} = 0.5, \lambda_{twitter} = 0.15$), and Yahoo! Placemaker, a freely available webservice capable of georeferencing documents and webpages. Placemaker identifies places mentioned in text, disambiguates those places and returns the centroid for the geographic scope determined for the document. Note that this approach uses external geographical knowledge such as gazetteers and other undocumented sources of information. We have evaluated the performance of the Placemaker based on the full text of each Wikipedia article in the spot test set. We have not considered the W&B test set, as test documents in this set are represented as bag-of-words, which prevents the use of a named entity tagger. Moreover, Placemaker was not able to return a location estimate for all of the documents in our test sets. In these cases, a default coordinate in London (51.507334, -0.127683) was used as an informed guess.

As a second baseline, we have implemented a method which uses the Geonames gazetteer. For each of the test items, we used a combination of the Natural Language Toolkit (NLTK[21]) for Python and the Stanford Named Entity Recognizer[22] to extract entities that refer to a location. Subsequently, the Geonames gazetteer is used to re-

[21] http://nltk.org/
[22] http://nlp.stanford.edu/software/CRF-NER.shtml

Table IX: Comparison of Yahoo! Placemaker (PM), Geonames (Geo), Roller et al. [Roller et al. 2012] (RO) and our approach on the spot test set (21 839 items).

	PM	Geo	RO	$k = 2500$	$k = 10000$	$k = 25000$	$k = 50000$
Median Error	30.17 km	24.05 km	8.12 km	3.57 km	2.37 km	2.19 km	2.18 km
Acc@0.001 km	0.00%	0.01%	0.02%	0.07%	0.07%	0.07%	0.05%
Acc@0.01 km	0.03%	0.10%	0.02%	0.91%	0.91%	0.84%	0.71%
Acc@0.1 km	0.27%	0.90%	0.10%	11.42%	11.30%	10.42%	9.50%
Acc@1 km	4.14%	9.95%	4.17%	33.01%	35.85%	35.82%	35.07%
Acc@10 km	27.57%	34.63%	53.11%	64.54%	74.39%	75.95%	75.77%
Acc@100 km	73.48%	63.67%	75.98%	96.54%	96.41%	94.98%	93.91%
Acc@1000 km	97.80%	73.40%	92.36%	99.04%	99.04%	98.67%	98.41%

trieve coordinates for each of these entities. In case of ambiguity (i.e. when multiple entries are found in the gazetteer for a single name), several possible locations for a place may be available. To assign coordinates to the Wikipedia article, we choose the medoid of the places that were found, choosing for each place the nearest coordinates in case of ambiguity. This corresponds to a standard heuristic for disambiguating place names. Again, in case this baseline could not extract any named entities and thus provide a prediction, an informed guess is made using the London coordinate. The results are presented in Table IX where the optimal results are highlighted. The location estimates for our results are again obtained by using the Jaccard similarity.

In this table, all configurations of our method considerably outperform both the Yahoo! Placemaker and the Geonames based method. The Placemaker appears to have a particularly bad coverage for spots, which helps to explain why it performs poorer than the Geonames based methods for e.g. Acc@0.1 km, Acc@1 km and Acc@10 km. Note that the Geonames based method also outperforms the approach from Roller et al. at short error ranges, e.g. Acc@1 km. This supports the hypothesis that using Wikipedia for training data is not suitable for finding exact coordinates, as there usually are no other Wikipedia articles about the same place as the test instance.

6. DISCUSSION

In addition to general classification errors made by our framework, errors that could potentially be avoided by using more training data, we also noted the following particular issues.

6.1. Extraterrestrial coordinates

One of the first anomalies we encountered when processing the Wikipedia training data from the W&B dataset is that certain coordinates had values beyond the expected ranges of latitude ($[-90, 90]$) and longitude ($[-180, 180]$). Table X provides examples of this. As can be seen from this table, this concerns coordinates that refer to celestial bodies other than the earth. A closer inspection of the training set revealed over 1000 of these extraterrestrial coordinates.

6.2. Automated error detection of coordinates

In the spot test set, there is a document about the "Erasmushogeschool Brussel"[23]. The system reported an error of 616.01 km when predicting the location of this test document. Closer inspection revealed that the ground truth for this Wikipedia page was incorrect, and our predicted location was actually the correct location for the place.

[23] http://en.wikipedia.org/wiki/Erasmushogeschool_Brussel

Table X: Example Wikipedia training documents with unexpected values for their geographical coordinates

Wikipedia name	Latitude	Longitude	Reason
Medusae_Fossae_Formation	-5.0	213.0	On Mars
Quetzalpetlatl_Corona	68.0	357.0	On Venus
Pele_(volcano)	-18.7	-255.3	On Jupiter's moon Io

In particular, the coordinates were reported as `50.7998N 4.4151W` instead of an *eastern* longitude which is likely to be due to a manual error.

This example suggests an idea to automatically detect errors in coordinates. If one or multiple sources in which we are highly confident claim that a document is located somewhere else than the current coordinates state, the framework could automatically correct the Wikipedia page. In the spot test collection, we detected three such errors, of which two have since been corrected on Wikipedia (as can be observed in their editing history): "Erasmushogeschool Brussel", which still has the incorrect coordinates online, "Monmouth Hospital"[24] and "Barryscourt Castle"[25].

6.3. Exact matches

Following the idea that no two Wikipedia documents cover exactly the same topic, we would expect not to find any two documents sharing the exact same coordinates. However, looking at the results of Tables I and III, there are a number of test documents that can be georeferenced to the exact correct location. After manually assessing these cases, we can divide the exact matches into the following categories:

— **Generic coordinates**: Generic coordinates are assigned to different pages that have something in common. For instance, the Wikipedia pages for *Liberia* (in the training data), the West-African country, and its capital *Monrovia* (in the test data), have the same coordinates. The reason for this is that the coordinates in the W&B dataset are obtained by processing the Wikipedia dump data and the coordinate of Monrovia is the first one mentioned in the raw page of Liberia. A similar argument holds for the pages of *Geography of Albania* (test) and *Albania* (training).
— **Identical concepts known by multiple names**: Certain training and test documents actually describe the same location. Apart from concepts known by different names, this can also be due to a change of name over time. This results in duplicates that are sometimes overlooked by Wikipedia authors. Some examples of changes over time are *Tunitas, California* (training) which is a ghost town that changed its name to into the town of *Lobitos* (test). Another example is the former *Free City of Danzig* (test), now known as *Gdańsk*.
— **Different concept but related coordinates**: This category hosts the most interesting matches. For example, the system managed to determine the location of the *MV Languedoc* (test) by providing the coordinates of the *SS Scoresby* (training). Both ships were torpedoed by the U-48 submarine and sunk in the same location. Another example of items that fall into this category are concepts that have their own Wikipedia page but are actually part of a more well-known concept, such as *Queen Elizabeth II Great Court* (test) as part of the *British Museum* (training) or *Larmer Tree Gardens* (test) that hosts the *Larmer Tree Festival*. [Wing and Baldridge 2011] also provides a brief discussion of this category of examples.

[24] `http://en.wikipedia.org/wiki/Monmouth_Hospital`
[25] `http://en.wikipedia.org/wiki/Barryscourt_Castle`

7. CONCLUSIONS

In this paper, we have presented an approach to georeferencing Wikipedia documents that combines language models trained over different sources of information. In particular, we combine Wikipedia training data with models trained using Flickr and Twitter, to account for the fact that the places described in a Wikipedia article may already be described in Flickr or Twitter. Overall, we have found that language models trained from Flickr can have a substantial impact on the quality of the produced geotags. As the number of Flickr photos increases every day, the potential of this method continuously increases, although the law of diminishing returns is likely to apply. For this reason, it may be important to consider a broader set of sources. The results we obtained for Twitter were less encouraging: unless language models are trained using billions of tweets, the use of Twitter does not offer substantial performance benefits. It should be noted, however, that various improvements for Twitter may be conceived. In particular, it may be possible to identify messages that are about the current location of the user (e.g. messages beginning with "I'm at") and training models from such messages may be more effective. As part of future work, we intend to look at other sources, such as local news stories, although exact coordinates are usually not available for such resources. As part of a solution, we may envision a system which uses the names of geographic regions as primitive locations instead of coordinates. This also relates to the challenge, discussed in [Van Laere et al. 2012], of finding the most appropriate level of granularity at which to estimate the location of a resource. Given the Wikipedia page for the Tour de France[26], for instance, identifying a precise coordinate does not make much sense. Rather, a system that can identify "France" as the most appropriate location estimate may be used (or a polygon which more or less covers France). This would bring the approach also closer to how documents are indexed in a spatially-aware search engine [Purves et al. 2007; Chen et al. 2006].

ACKNOWLEDGMENTS

The authors would like to thank Benjamin Wing, Jason Baldridge and Stephen Roller for providing us with their datasets and with the details of their experimental set-up.

REFERENCES

AMITAY, E., HAR'EL, N., SIVAN, R., AND SOFFER, A. 2004. Web-a-where: geotagging web content. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 273–280.

CHEN, Y.-Y., SUEL, T., AND MARKOWETZ, A. 2006. Efficient query processing in geographic web search engines. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*. SIGMOD '06. ACM, New York, NY, USA, 277–288.

CHENG, Z., CAVERLEE, J., AND LEE, K. 2010. You are where you tweet: a content-based approach to geolocating Twitter users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. 759–768.

CRANDALL, D. J., BACKSTROM, L., HUTTENLOCHER, D., AND KLEINBERG, J. 2009. Mapping the world's photos. In *Proceedings of the 18th International Conference on World Wide Web*. 761–770.

DE ROUCK, C., VAN LAERE, O., SCHOCKAERT, S., AND DHOEDT, B. 2011. Georeferencing Wikipedia pages using language models from Flickr. In *Proceedings of the Terra Cognita 2011 Workshop*. 3–10.

EISENSTEIN, J., O'CONNOR, B., SMITH, N. A., AND XING, E. P. 2010. A latent variable model for geographic lexical variation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. 1277–1287.

HAUFF, C. AND HOUBEN, G.-J. 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.

[26] `http://en.wikipedia.org/wiki/Tour_de_France`

HAUFF, C. AND HOUBEN, G.-J. 2012. 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*. 691–700.

HAYS, J. H. AND EFROS, A. A. 2008. IM2GPS: Estimating geographic information from a single image. In *Proceedings of the 21st IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1–8.

JONES, C. B., PURVES, R. S., CLOUGH, P. D., AND JOHO, H. 2008. Modelling vague places with knowledge from the web. *International Journal of Geographical Information Science 22*, 1045–1065.

KINSELLA, S., MURDOCK, V., AND O'HARE, N. 2011. "i'm eating a sandwich in glasgow": modeling locations with tweets. In *Proceedings of the 3rd international workshop on Search and mining user-generated contents*. 61–68.

KRIPPNER, F., MEIER, G., HARTMANN, J., AND KNAUF, R. 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.

LEIDNER, J. L. 2007. *Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding of Place Names*. PhD thesis, School of Informatics, University of Edinburgh.

LIEBERMAN, M. D., SAMET, H., AND SANKARANAYANANAN, J. 2010. Geotagging: using proximity, sibling, and prominence clues to understand comma groups. In *Proceedings of the 6th Workshop on Geographic Information Retrieval*. 6:1–6:8.

MANGUINHAS, H., MARTINS, B., AND BORBINHA, J. 2008. A geo-temporal Web gazetteer integrating data from multiple sources. In *Proceedings of the 3rd International Conference on Digital Information Management*. 146–153.

POPESCU, A., GREFENSTETTE, G., AND MOËLLIC, P. A. 2008. Gazetiki: automatic creation of a geographical gazetteer. In *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*. 85–93.

PURVES, R. AND JONES, C. 2011. Geographic information retrieval. *SIGSPATIAL Special 3,* 2, 2–4.

PURVES, R. S., CLOUGH, P., JONES, C. B., ARAMPATZIS, A., BUCHER, B., FINCH, D., FU, G., JOHO, H., SYED, A. K., VAID, S., AND YANG, B. 2007. The design and implementation of spirit: a spatially aware search engine for information retrieval on the internet. *International Journal of Geographical Information Science 21,* 7, 717–745.

RAE, A. AND KELM, P. 2012. Working Notes for the Placing Task at MediaEval2012. In *Working Notes of the MediaEval Workshop*. CEUR-WS.org, ISSN 1613-0073, online http://ceur-ws.org/Vol-927/mediaeval2012_submission_6.pdf.

RATTENBURY, T., GOOD, N., AND NAAMAN, M. 2007. Towards automatic extraction of event and place semantics from flickr tags. In *Proceedings of the 30th Annual International ACM SIGIR Conference*. 103–110.

RATTENBURY, T. AND NAAMAN, M. 2009. Methods for extracting place semantics from Flickr tags. *ACM Transactions on the Web 3,* 1, 1–30.

ROLLER, S., SPERIOSU, M., RALLAPALLI, S., WING, B., AND BALDRIDGE, J. 2012. Supervised text-based geolocation using language models on an adaptive grid. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. 1500–1510.

SERDYUKOV, P., MURDOCK, V., AND VAN ZWOL, R. 2009. Placing Flickr photos on a map. In *Proceedings of the 32nd Annual International ACM SIGIR Conference*. 484–491.

SMART, P. D., JONES, C. B., AND TWAROCH, F. A. 2010. Multi-source toponym data integration and mediation for a meta-gazetteer service. In *Proceedings of the 6th international conference on Geographic information science*. GIScience'10. Springer-Verlag, Berlin, Heidelberg, 234–248.

TOBIN, R., GROVER, C., BYRNE, K., REID, J., AND WALSH, J. 2010. Evaluation of georeferencing. In *Proceedings of the 6th Workshop on Geographic Information Retrieval*. 7:1–7:8.

TWAROCH, F. A., SMART, P. D., AND JONES, C. B. 2008. Mining the web to detect place names. In *Proceedings of the 2nd international workshop on Geographic information retrieval*. GIR '08. ACM, New York, NY, USA, 43–44.

VAN LAERE, O., SCHOCKAERT, S., AND DHOEDT, B. 2011. Finding locations of Flickr resources using language models and similarity search. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*. 48:1–48:8.

VAN LAERE, O., SCHOCKAERT, S., AND DHOEDT, B. 2012. Georeferencing flickr photos using language models at different levels of granularity: An evidence based approach. *Journal of Web Semantics*.

VAN LAERE, O., SCHOCKAERT, S., AND DHOEDT, B. 2013. Georeferencing flickr resources based on textual meta-data. *Information Sciences Volume 238*, 52–74.

WEINBERGER, K. Q., SLANEY, M., AND VAN ZWOL, R. 2008. Resolving tag ambiguity. In *Proceedings of the 16th ACM international conference on Multimedia*. 111–120.

WING, B. AND BALDRIDGE, J. 2011. Simple supervised document geolocation with geodesic grids. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 955–964.

ZHAI, C. AND LAFFERTY, J. 2001. A study of smoothing methods for language models applied to Ad Hoc information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference*. 334–342.

Received November 2012; revised October 2013; accepted X