JOURNAL OF SPATIAL INFORMATION SCIENCE
Number 9 (2014), pp. 37–70
doi:10.5311/JOSIS.2014.9.170

RESEARCH ARTICLE

# Geocoding location expressions in Twitter messages: A preference learning method

Wei Zhang and Judith Gelernter

School of Computer Science, Carnegie Mellon University, Pittsburgh, USA

*Received: March 6, 2014; returned: May 12, 2014; revised: June 3, 2014; accepted: June 27, 2014.*

**Abstract:** Resolving location expressions in text to the correct physical location, also known as geocoding or grounding, is complicated by the fact that so many places around the world share the same name. Correct resolution is made even more difficult when there is little context to determine which place is intended, as in a 140-character Twitter message, or when location cues from different sources conflict, as may be the case among different metadata fields of a Twitter message. We used supervised machine learning to weigh the different fields of the Twitter message and the features of a world gazetteer to create a model that will prefer the correct gazetteer candidate to resolve the extracted expression. We evaluated our model using the F1 measure and compared it to similar algorithms. Our method achieved results higher than state-of-the-art competitors.

**Keywords:** geocoding, toponym resolution, named entity disambiguation, geographic referencing, geolocation, grounding, geographic information retrieval, Twitter

## 1 Introduction

News tweet: "Van plows into Salvation Army store in Sydney."[1]

The obvious guess is that this tweet refers to an incident in Australia. In fact, the tweet refers to an incident that happened on the Cape Breton Island in Nova Scotia, Canada. It requires examining the information associated with the tweet text to determine which geographic place the name "Sydney" refers to. How we solve this problem algorithmically is the topic of this paper.

[1]Official Twitter account of CBC Nova Scotia@ CBCNS 1pm May 20 2014

© by the author(s) Licensed under Creative Commons Attribution 3.0 License

Place names are known as toponyms. Multiple instances of the same toponym in different geographic regions create a disambiguation problem. Leetaru [24] found that Northern Africa and the Middle East have fairly low levels of toponym duplication, while North and South America and South-Eastern Asia have higher-than-normal concentrations of name duplication. The higher the level of name duplication, the higher is the potential geocoding error.

Disambiguation clues from other locations mentioned in the text, as well as clues from the gazetteer used to resolve those locations, have been used for what is termed toponym resolution [25]. The toponym resolution in text problem is complicated when the text is limited to 140 character messages known as tweets. However, a Twitter message is also accompanied with information from its JavaScript Object Notation (JSON) file. We refer to this as metadata. The user profile, user registration date, user location, number of friends and followers in the social network, GPS coordinates, and time that the tweet was posted, are some of the fields that might help disambiguate location information. Frequently, these metadata fields are blank. In our random sample of 1 million English-language tweets from January 2013, for example, the GPS coordinates of the mobile device at the time the message was tweeted appear in only a tiny fraction tweets.

*Research problem*

The problem is how to determine which geographical location (that is, which entry in the gazetteer) is the correct match with the location word(s) found in tweet text. We borrow methods used for toponym resolution in text to handle toponym resolution in tweets. A number of heuristics as to which gazetteer location is the most probable match for the location in text have been used with some success [25], but balancing priorities among known heuristics is complex.

New methods are required to handle short tweets and take advantage of relevant Twitter metadata. Our approach is to incorporate different sources of information from the gazetteer, as well as surface similarity between tweet toponym and gazetteer candidates, tweet context, and tweet metadata. We use a preference learning model to rank gazetteer candidates in order to find the geographical location that corresponds to the place word in the tweet.

Tweet metadata helps to solve the problem. Examples of metadata fields relevant to location are user registered location information, the user time zone, user description, and the tweet coordinates that indicate where the tweet was posted (Table 1).

Our data set from a three-month period in 2013 is constructed from English-language tweets, which was filtered to find those tweets that contain toponyms that were the most ambiguous (that is, with toponyms that had the most duplication in the gazetteer). We deliberately did not collect tweets for a particular crisis because we wanted the toponyms to be spread randomly across the world, so that the resulting model would be maximally generalizable.

*Significance of solving the research problem*

While tweet mapping has been used for predicting elections and tracking disease, for example, our particular application has been in Twitter for disasters. Victims of tornadoes and typhoons are increasingly tweeting calls for help that refer to locations. After the re-

| Field Name | Source for information in the field | Example |
|---|---|---|
| Tweet text | User input at the time of posting the tweet | "The crowd in baltimore when ray lewis comes out of the tunnel for the last time will erupt! #OneOfTheBest" |
| Coordinates | User mobile device at the time of the posting of the tweet | 37.59628575, -82.24814532 |
| User location | User input at the time of opening a Twitter account | Belfry, Ky |
| Time zone | Twitter-entered time zone of the user at the time of the posting of the tweet | Atlantic Time (Canada) |
| User description | User input at the time of opening a Twitter account | "17 years old, plays basketball for belfry high, senior c/o 2013" |

Table 1: Tweet metadata fields that are relevant to location (example from our January 2013 data set).

cent Japanese earthquake, for instance, when land lines collapsed, Twitter and Facebook were used to post information such as shelter locations to those who were in trouble [40]. The Philippine government broadcast a hash-tag for its citizens to follow during the 2012 typhoon[2], encouraging its citizens to use particular terms in their tweet-calls for help.

Getting location information out of crisis tweets will help disaster recovery. To do this, we need to identify the correct location in order to send help to the right place. Tweets that include locations are re-tweeted more often than other tweets [31], suggesting that Twitterers themselves find these tweets more important.

Our geocoding algorithm attaches geographic coordinates to place names and is the first step in allowing those place names to be mapped. For crisis tweets, the ambiguity problem will be less when the number of potential overlap names is confined to a particular region, but the penalty for resolving a name incorrectly will be substantially higher.

*Toponym resolution consists of geoparsing and geocoding*

Geoparsing, or recognizing a word as a location, here is prior to geocoding. Geoparsing is also termed the *named entity recognition* problem. We have experimented with geoparsing [12] and multilingual geoparsing [14]. The geoparsing problem is confounded by non-geographical words that are also geographical words (e.g., Turkey on Thanksgiving as opposed to Turkey on the Mediterranean). The geoparsing problem intensifies when the language is informal, or contains non-grammatical microtext that may contain uncommon abbreviations or slang.

Measurement of geocoding error is aggravated by geoparsing, and by the gazetteer which may include distinct political and economic regions with different coordinates but closely related names (see Table 2). We do not measure the geoparsing error in our experiments in this paper because our concern is on the method to disambiguate place names. We measure the disambiguation problem.

[2]http://irevolution.net/2014/07/01/filipino-official-strategy-crisis-hashtags/

| Location Name | Feature Type | Population | Lat, Longitude | ID |
|---|---|---|---|---|
| City of London | PPLA3 | 7,556,900 | 51.51279, −0.09184 | 2643741 |
| London | PPLC | 7,556,900 | 51.50853, −0.12574 | 2643743 |
| City of London | ADM3 | 8000 | 51.51334, −0.08901 | 2643744 |

Table 2: Excerpt from the GeoNames gazetteer that shows name ambiguity for the same city. The place types, according to GeoNames are: PPLA3 is the seat of the 3rd order administrative division, PPLC is a capital, and ADM3 is the 3rd order administrative division.

Our research aims to determine which place name is referred to in text, recognizing recurrences of the same name in different parts of the world are common. So the candidate place name that is output by our system is not invariably the correct candidate. We can, however, use our gazetteer ranking method to attach the probability that a particular location has been resolved correctly.

The output of our model assumes that the place names listed in the gazetteer are correct. We use the GeoNames gazetteer, even though any gazetteer with English place name entries could be used, and errors of irregular name coverage (place names missing) and geolocation inaccuracy have been found in the GeoNames [3]. Thus, the output of any geocoding algorithm is only as exact as the knowledge base that underlies it.

There are numerous ways that people describe places that are not toponyms. Streets, buildings, and rooms in buildings are some common ways that people define their location [35]. Because our interest is in geocoding, however, we limit our definition of location to toponyms.

*Organization of this article*

Others' toponym resolution approaches are reviewed in Section 2. The phases of our research are detailed in Section 3. We annotated our own Twitter data due to the novelty of our research question, and we describe our approach to data annotation in Section 4. The machine learning algorithm used to select the correct gazetteer candidate to resolve the toponym(s) in a tweet is described in Section 5. The experiments in Section 6 show the relative importance of features in our machine learning algorithm in determining the correct location with respect to a baseline, and the relative importance of the training data set size. We perform an analysis of errors in Section 7 based on a run of unseen data in order to show strengths of the model and where it can be improved. We use the same data to compare our method to others' in Section 8. The next phase of research is proposed in Section 9, and the most important contributions of this research are stated in conclusion.

## 2 Related work

Geocoding includes the problem of how to associate geographic coordinates with a place name, also called a toponym. It becomes a word disambiguation problem in that many places have the same name. The question is: how to determine the sense (that is physical location) of the word mentioned in the text that corresponds to the multiple words of similar form in the knowledge base (gazetteer)? The solution involves determining what

information can be used to associate a geographical location with a reference in text, and then how that association can be computed automatically.

This section presents several approaches to geocoding. The section begins with the geocoding of an address rather than a toponym. Then it reviews data expansion methods that find clues to disambiguate related data, and language models that find clues in words used in context. The section also covers heuristic methods in which clues are drawn both from context and from the gazetteer, and machine learning methods in which heuristics become statistical probabilities rather than rules.

## 2.1 Geocoding that does not require toponym resolution

The practical importance of this problem is demonstrated by the variety of geocoders that, given a postal address, can output geographical coordinates. Texas A&M[3], Google[4], and TeleAtlas[5] geocoders are just a few examples. The GeoNames API can also resolve postal code[6].

Location words may be resolved using context, such as accompanying street addresses or postal codes, rather than toponym lookup [10]. The geolocation of a street address may be accomplished with the help of street indexes and postal codes (for example, [21]). Buildings too may be geolocated if they can be associated with a street address.

Geocoding may be accomplished even more simply using the GPS coordinates of a user's mobile device, as is the case with most currently available digital tweet maps. For example, Twittervision[7] shows tweets and photos from around the world, with Trendsmap[8] showing topics that are trending on Twitter. The One Million Tweet Map[9] shows the relative locations of a million recent tweets, and A World of Tweets[10] shows the tweets descending onto a world map dynamically. Larger-scale Twitter maps focus on communities, such as Twitter Mapper/Gathering Point; topics, such as on GeoChirp[11] and Tweography[12]; or friends' tweets, as on MyTweetMap[13]. GPS coordinates of tweets have been used also for sentiment maps of Twitter users. One recent sentiment analysis of tweets, for example, selected GPS-located tweets only for mapping [37]. But relatively few tweets have geographical coordinates, which strongly biases the sampling. The capability of moving from GPS location to map is available in most of the tools for making a map from tweets (for example TweettoMap[14] or Maptimize[15]).

Below we discuss four approaches to toponym resolution: data expansion, language models, heuristics and machine learning.

[3]http://geoservices.tamu.edu/Services/Geocode/
[4]http://developers.google.com/maps/documentation/geocoding/
[5]http://www.ffiec.gov/Geocode/
[6]http://www.geonames.org/export/free-geocoding.html
[7]http://twittervision.com
[8]http://trendsmap.com
[9]http://onemilliontweetmap.com
[10]http://aworldoftweets.frogdesign.com
[11]http://www.geochirp.com
[12]http://www.tweography.com
[13]http://mytweetmap.com
[14]http://tweettomap.com
[15]http://maptimize.com

## 2.2 Geocoding by means of data expansion

The problem of toponym geocoding has been considered in the "Knowledge Base Population" track at the 2010 Text Analysis Conference by the National Institute of Standards and Technology [22]. The aim is to supplement potentially ambiguous words in a document with words in other documents. The knowledge base used for named entity disambiguation has been Wikipedia, [7]. One approach selected the candidate referents, and two disambiguation modules determined the most probable referent independently. If the two modules disagreed, then no candidate was assigned [30].

Data expansion provides can provide more location cues to aid in toponym resolution. Expansion is based on direct or indirect links from the context of the toponym to an information source that may include information that helps to disambiguate the toponyms. Links could be to articles, comments, tweets from the same user, or friends in social networks, to name a few. Ireson and Ciracegna (2010) [20] compared how different quantities of user-generated content would resolve the toponyms in Flickr data. They used support vector machines (SVMs), and found that information beyond the item itself was not particularly helpful. This approach is similar to ours because we are using the tweet metadata, which is an "expansion" of the tweet text. Tweet coordinates in the metadata are considered to be a direct source of location information. Locations found in the user-registered location field of a Twitter message are more indirect' sources of information.

Others have used non-location words or events help classify location words, as in [36] and [39]. A candidate toponym in a document could be disambiguated with the help of other place references in the same document, as in Lieberman and Samet [27]. We adapt this to Twitter by looking for place references in other metadata fields of the Tweet JavaScript Object Notation (JSON) format. Kauppinen et al. [23] used geo-ontology to help disambiguate the locations. They focus on local toponyms in Finland and on the historical change.

Other external sources, such as events, have been used to help disambiguate toponyms. For example, when the 2012 Summer Olympics is mentioned, we can infer London, UK as a location. Nothman et al. [33] linked events in local news articles to enrich a limited corpus. The authors saw an increase in average accuracy of toponym grounding in their limited corpus, but did not see an increase in toponym resolution. Roberts et al. [36] built a model that combined geospatial and temporal evidence that improved the resolution accuracy.

Social networks have been used to locate users geographically [9]. The data must be gathered in such a way as to preserve the social network, however, if this method is to be replicated.

## 2.3 Geocoding by means of language models

A language model is made under the assumption that specific locations employ specific words or word forms. For example, Sano et al. [32] built a corpus for different areas based the tf-idf (term frequency—inverse document frequency) statistic and toponym blocks. Tf-idf is often used as a factor in weighting since the statistic is larger when a word appears more frequently in a document, but this is moderated by the number of times that word appears in the corpus. As an example of a toponym block, a place name ending with "isaki" is more likely to be in Japan. Evaluation of place words is a subset of information retrieval, and so standard precision and recall measures are used, as well as the harmonic mean of precision and recall in the F-measure (see Section 6.1 for further explanation of

these measures). Sano et al. [32] reported an average precision of 85.54% and recall of 91.92%, but the toponyms were disambiguated to the country level only. Hosokawa [19] also built non-geographic features of relevant places from news articles based on the tf-idf statistic. Using such techniques alone is not suitable for finding the best single candidate for toponym resolution, although it might provide the several candidates that are most promising. Adams and Janowicz [1] geolocated locations with text by association of the text with topics learned from geocoded Wikipedia and travel blogs.

## 2.4 Geocoding by means of heuristics and graph-based methods

Leidner [25] lists heuristics used in toponym disambiguation research, including population, level of spatial hierarchy, geometric minimality (to minimize distance among toponyms mined from the text), and frequency of occurrence in the text. The problem becomes how to know which rules dominate so as to know which to follow when they conflict. Rather than weighting the different rules, Smith and Crane [38] tested multiple toponyms to see if some candidates are distributed around some centroids. Candidates far from the centroids are eliminated and grouping is performed among nearby candidates under the one sense per co-location assumption. Lieberman, Samet, and Sankarayananan [28] recognized that comma groups of three or more place names in a row tend to be siblings, that is, on the same level of the geospatial hierarchy.

Buscaldi [6] created a metric of the geographic distance between multiple ambiguous toponyms as combined with the frequency of the toponyms that appear in the texts to set the weights. Their work achieved a precision and recall of 88%. Li et al. [26] assigned higher weights for candidates within the same state and lower weights for candidates within the same continent. They used spanning trees to combine multiple toponyms based on the hierarchical similarities in order to find the candidates with the most significant similarities. The reported accuracy was 96%, but the corpus was very small. Wang et al. [41] used Dempster-Shafer theory for multiple-evidence combination, and preliminary results showed the co-occurrence method indeed improved performance.

Goldberg et al. [15] compared several geocoding approaches. They framed the problem as: given multiple reference sources with different classes of geographic objects at different geographic scales (when we use just one reference source, a gazetteer), which is the correct physical location? They found that preferring one reference source as the most authoritative to resolve toponyms is less accurate than other approaches, including: "uncertainty," in which they select the gazetteer candidate with the smallest centroid as it has the least uncertainty; "gravitation," in which a center is calculated based on candidates from multiple reference sources (but this might pull away from the correct location when a smaller candidate is contained within a larger; and "topological" in which the topological features of the candidates are used to determine the probable output location. Goldberg et al [15] present a method to show the relative likelihood that a given geocoded location is accurate.

## 2.5 Machine learning approaches

Scientists have used machine learning to decide on the best candidate among a set of gazetteer candidates. Lieberman et al. [27] used context features to construct a window of varied size (proximity) and depth (number of candidates), and implemented random forests to classify toponyms based on gazetteer features that included population and ex-

onyms (names by which a place is known in other areas and languages). In their domain of news, precision was over 95%, and the best recall ranged from 61% to 90%.

Blessing [5] built an SVM classifier, adding words and bigrams to feature vectors. The F-measure for German text was around 90%, although the method is not detailed. Agirre et al. [2] described general word-sense disambiguation based on the $k$-nearest neighbor machine learning algorithm, as well as SVMs, with moderate success. Wing and Baldridge [42] used supervised methods to geographically locate Wikipedia and Twitter data. They divided the Earth into small cells by latitude/longitude and then classified the documents to fit into these cells in order to determine their distribution by particular words (since terms like barbecue and wine may have higher density in particular regions). They measured the similarity of the testing document and the term distribution in each cell to determine the candidate with the smallest divergence. For Wikipedia data the mean error was 221km and median was 11.8km. For tweets with far less context, the mean error was higher at 967km and median of 479km. Speriosu and Baldridge [39] used document-level geo-tags as training data labels. With auto-created training data, their result out-performed the population baseline on the data set they use. However, their improvement is so significant because of the error introduced by auto-labeling, and the lack of rich information to help.

Han et al [17] used metadata and context words as well as the language of the tweet to geolocate the Twitterer. Our work is similar in using metadata, but we geolocate toponyms in the tweet instead of the person who posted the tweet.

# 3 Methodology

Our objective, given the text of the 140-character Twitter message, is to create a model to resolve toponyms in the tweet text and output the physical location (geographic coordinates) that correspond to the place named. Our approach used the following steps:

*Data collection and annotation*

We began the study by collecting English-language tweets within a specified period of time. We filtered these to contain tweets with toponyms that are ambiguous, with the result that the data would be more informative for training than would a random sample. In each tweet, we manually identified and attached coordinates to each toponym. These coordinates are the labels that the supervised machine learning model used to predict labels for unseen data. This is detailed in Section 4.

*Geoparsing*

We used a tool created in our earlier research to identify which words in tweet text were locations [14]. The result is a non-ranked list of gazetteer candidates for each location word (Section 5).

*Geocoding*

Our method to create a model that resolves toponyms found in tweet text by ranking the gazetteer candidates identified in the geoparsing step is detailed in Section 5. The model classifies gazetteer candidates as good or bad matches for each extracted toponym.

The characteristics of the text message, of the Twitter metadata that accompanies the message, and of the gazetteer candidates are used by the algorithm to help discriminate between gazetteer candidates. These characteristics are known as "features" in a machine learning algorithm. SVMs are used for machine learning because of their effectiveness when training data is limited. Note that "features" as a way to code patterns seen in data should not be confused with "features" as a type of data in a gazetteer.

*Geocoding algorithm: internal testing*

In Section 6 we added new features one at a time to identify which carry the most information, so that we could use the best features to make the best possible model. We compared these features to gazetteer features used as a baseline to ensure we could measure performance gains per feature. Also, we demonstrated empirically that the amount of data we have used to train the model is sufficient. These experiments show the extent to which we were able to optimize the model, given the training data and our choice of features.

We evaluated this optimized model on a data set in Section 7, and evaluate the results in terms of the F1 measure and the spatial hierarchy. We performed an error analysis on the results to determine the types of mistakes that were made so that we know how to improve the geocoder in the future, as considered in Section 9.

*Geocoding algorithm: external evaluation*

Evaluation experiments in Section 8 compare our Carnegie Mellon algorithm to similar algorithms from Berico Technologies and from Yahoo. The results demonstrate our algorithm's strength in tweet geocoding.

# 4 Data and the gold standard for geo-tagging

Most toponym resolution experiments have been conducted using news articles, such as the SpatialML data set or LGL data set from [27]. SpatialML, for example, is a set of news documents in which each toponym has been hand-assigned a latitude and longitude[16] A Twitter data set for which all the tweets include GPS coordinates is also in circulation[17], but it does not include the full metadata that accompanies every tweet.

We manually annotated a set of 956 tweets by noting toponyms and geographical coordinates. We will make our hand-annotated tweets available for research purposes according to the parameters allowed by the Twitter company. The tweets in our sample were posted between 1 January 2013 and 1 April 2013. The tweets were downloaded from the Carnegie Mellon archive with permission of Brendan O'Connor. Within this set, we selected only those tweets that contained toponyms that were most ambiguous. We did this by geoparsing the tweets first[18]. Then we listed the toponyms that appear. Using the number of repetitions of a toponym entry in the gazetteer, we selected the 1000 most ambiguous toponyms in this list (that is, the 1000 toponyms with the most name duplication in the

[16]SpatialML data is from the Linguistic Data Consortium http://catalog.ldc.upenn.edu/LDC2011T02. The LGL data is available through the authors: Lieberman, Samet, Sankaranarayanan [27].

[17]http://www.ark.cs.cmu.edu/GeoTwitter

[18]http://github.com/geoparser/geolocator

gazetteer). Next we used these toponyms to search the January–April tweets. Those tweets that remained had enough ambiguity to train a model that would be discriminating. Of those tweets, we retained only the ones that also had GPS coordinates or Twitter "place" fields (which also include coordinates). Table 5 presents the statistics for this data set.

Our geoparser contains a machine learning algorithm called a "named entity recognizer" which allows it to recognize as locations some entries in text which do not appear in the gazetteer. Consequently, some tweets containing toponyms not in the gazetteer were included in the data set. This has no ramifications for our study because we do not evaluate this aspect in the present research. In this study, we are interested only in location expressions that manifest ambiguity; that is, location expressions that appear more than once in the gazetteer.

*Annotation procedure*

We tagged each toponym found in tweet text, including hash-tag index words such as #LosAngeles. We also tagged demonyms that morphologically match the corresponding location, so that for example, "Puerto Rican" would be tagged as Puerto Rico, and "Dutch" would be found as the Netherlands. We were able to match hash-tag expressions without white space because we indexed both the white space, and non-white space version of each. Table 3 gives an example of the data with tags for toponym resolution. We did not perform the degrees-minutes-seconds to decimal degrees conversion on every GeoNames entry when we created our gold standard in order to save ourselves an extra calculation. Using latitude, longitude in integers was adequate for the resolution required here. Our repeated reviewing of output after iterative runs of a manageable-sized training set ensured that there were no mistakes in the manual annotations.

The set of potential gazetteer candidates for each toponym was mined from tweet text (see Table 4). We calculated the geographical distance between the coordinates of the true candidate in the gazetteer, and those of the mined toponym.

| **Text** | **Toponym** |
|---|---|
| Damn I miss Cebu City! Good morning Pinas! | tp{**Pinas**[14,121]}tp<br>tp{**Cebu City**[10,124]}tp |
| Ringing in the new year with two of my new best friends from 2012 at @republicMN in Minneapolis ☺ | tp{**Minneapolis**[45,-93] 5037649}tp |

Table 3: An example of the tags using the coordinate-wise tagging (upper example) and identification-wise tagging (lower example).

All candidates within some minimum threshold distance with the correct geographical features (such as population, state) were labeled as true. In some cases, multiple gazetteer candidates match one extracted toponym (see for example Table 4, as well as the Table 2 excerpts from the GeoNames gazetteer).

*Statistics*

In those 956 tweets in our data set containing locations and GPS coordinates, we found 1,393 toponym expressions (Table 5). The table shows that there were in the data set 779 unique toponym expressions, for which there were 877 gazetteer candidates that corre-

| Tweet text | Toponym candidates* | Pop. | Number alternative names | GeoNames ID | Latitude | Longitude |
|---|---|---|---|---|---|---|
| Ringing in the new year with two of my new best friends at @republicMN in Minneapolis ☺ | Minneapolis us ks | 2032 | 2 | 4275586 | 39.1219 | −97.7067 |
| | Minneapolis us nc | 0 | 2 | 4479740 | 36.0993 | −81.9871 |
| | Minneapolis us mn | 382,578 | 40 | 5037649 | 44.98 | −93.2638 |

* City, country, state.

Table 4: An example of the tagged gazetteer candidates for Minneapolis.

spond lexically to those toponyms (a ratio of 1 toponym in tweet text to 1.35 gazetteer candidates, shown in the table by GeoNames IDs). This demonstrates the potential for ambiguity in this data. Some of the same places are mentioned more than once in the tweet set, bringing the total number of toponyms to 1393 (Table 5), which widens the ambiguity further.

| | |
|---|---|
| Number of tweets | 956 |
| Unique toponyms | 779 |
| Gazetteer candidates (unique GeoNames IDs) | 877 |
| Total toponyms | 1393 |
| Gazetteer candidates (total GeoNames IDs) | 1877 |

Table 5: Statistics on our Twitter data set.

# 5 A classifier for toponym resolution in tweets

## 5.1 Geoparsing as pre-requisite to geocoding

We built our own geoparsing algorithm in the course of earlier research to identify location expressions in tweets [14]. The geoparser relies both on rule-based and machine learning methods to determine which words are locations. Some of the features and natural language processing tools are designed specifically for tweets. The locations recognized from the geoparsing algorithm are buildings (BD), streets (ST), toponym (TP), and abbreviation (AB).

Most existing named entity recognition models for locations have been trained on news articles or formal text. In our domain of social media text, by contrast, the sentence structure may be informal, with words abbreviated or in slang, or with careless spelling or phrases squeezed (that is, without proper spacing between words).

Although the geoparser is gazetteer-based, there is no guarantee that every location extracted will be in the gazetteer. Consequently, some locations cannot be geocoded. We

have separately created an algorithm that will automatically add local place references to a gazetteer [13]. Examples of toponyms that can be recognized but not geocoded include abbreviations (such as "Cali" for "California"), "Vegas" (for "Las Vegas"), and possessives lacking an apostrophe (such as "Detroits").

## 5.2 Classification and features

The procedure for toponym classification has four steps: (1) recognize the toponym mentions in the text (which requires the geoparser, Section 5.1); (2) look up the candidate(s) in the gazetteer according to features (Sections 5.3–5.4); (3) use machine learning to classify each candidate as either a good match (true) or not a good match (false), (as described in Sections 5.5–5.7); (4) use the gazetteer coordinates of the true match to determine the output. This is accomplished by aggregating the coordinates of the true candidates based on the confidence value.

### 5.2.1 Features: gazetteer

The GeoNames gazetteer[19] includes attributes for each entry, such as population, alternative names, and level in the geopolitical hierarchy. We use population, alternative names, and geographical features as machine learning features (cf. [25]). These features help to indicate the likelihood that a gazetteer candidate is a correct match with the extracted location expression.

GeoNames presents difficulties for our research due to inconsistencies [3]. For example, some large places do not include a population or set of alternative names. These inconsistencies are partially responsible for the ability to resolve some toponyms in the baseline (see Table 9, for example).

The gazetteer features for the baseline are population (that is, the number of people living in a toponym-region) and alternative names (the number of alternative names listed for that toponym). Gazetteer features become the baseline for the construction of features for our model not only because they have been used in earlier toponym resolution research, but also because they are independent of any particular data sample. Establishing a baseline allows us to measure performance gains for each context and metadata feature against a stable, understandable number. These gazetteer features for the baseline are presented in Tables 6–9; a combination of other features with the baseline appears in Tables 10–13.

*Population*

The GeoNames gazetteer includes population statistics for many populated places, at the level of country, state, and city. That means that oceans, rivers, train stations, and other types of entries are given a population of zero. Population is a good indicator to determine whether a place in the gazetteer matches that found in a tweet because the larger places (as indicated by population size) are more likely to be mentioned.

In order to select the places with larger population, we simply rank the gazetteer candidates in decreasing order by population value, and use the numerical ranking values in the range $[1, \infty]$ as the "utility" of the population feature. We have compared the population absolute value feature and ranking feature on our data set of 390 tweets, and although the

[19] http://www.geonames.org/

recall is 26% by using the absolute population value, the recall reaches 69% for population ranking among gazetteer candidates.

*Alternative names*

The GeoNames gazetteer includes exonyms for some of its place entries. Exonyms are names for the same place in other languages. For instance, for the city of Delft in Holland, GeoNames includes exonym renditions of the city name in English, Greek, Russian, Hebrew, Japanese, and Chinese. We have combined the exonyms for a single gazetteer entry with alternative listings in the gazetteer (such as "London" and "City of London," shown in Table 2) to create the feature "alternative names."

The assumption behind using alternative names as an indicator of which gazetteer candidate is a good match for the extracted toponyms is that the best known places have the most alternative names. The number of alternative names is thus a correlate of that place's wider familiarity. The feature value we use is the rank of the alternative name counts as we did for population, which is also within range $[1, M]$, where $M < \infty$. We designate the toponyms that do not have alternative names as $-1$ in order to widen the gap between non-populated and populated places.

*Geographical attribute*

In the GeoNames gazetteer, the geographical attribute is either a location type (populated place, administrative division, sea, lake, hotel, for example) or a political entity feature (country or state code, for example, which appears as ADM1, ADM2, ADM3, ADM4). For instance, the United States of America will have only a country code of 00. Pittsburgh, a city in the state of Pennsylvania, USA, will have a country code "us," ADM1 code "pa." Because the ADM1 code is not empty for Pittsburgh, it can be recognized as a city-level toponym. We can use the geographical type to infer the level of the spatial hierarchy for locations, provided the type is given.

In addition, we experimented with including the feature values into the classification model to see whether the most frequently-occurring feature types would affect gazetteer candidate selection. The total number of geographical feature types in the GeoNames gazetteer is 667. We used a vector of 667 binary values to represent the feature vector, with the "one hot encoding" technique [8].

By using this method, we learned the specific types of locations to choose as candidates. This method suffers from the "curse of dimensionality" problem, meaning that we have insufficient data for too many features. In our training data we only observed around 200 features that are actually recorded as candidate attributes for all the toponyms in our dataset, reducing the feature size to around 200. However, this reduction is still not enough to rule out the problem. We investigated the effectiveness of both feature value and feature hierarchy in experiments to determine whether the locations higher in the geospatial hierarchy or the locations with most common features are favored by more data. The experimental results are shown in Section 6.

#### 5.2.2 Features: geospatial context for the location expression

Contextual features, in the general case, are other toponyms extracted from data surrounding the toponym mentions in tweet text. Contextual features could mean location-related

terms, such as the "Steelers"—a football team from Pittsburgh. In our case, however, our knowledge base is a gazetteer, so our contextual features are limited to toponyms. This is similar to the heuristic of spatial autocorrelation [4], which assumes that each location mentioned in a particular context is not independent, but rather is spatially correlated. The relation could be "hierarchical," which implies that a location entity is contained within another location entity. The relation could instead be "sibling," in which the two entities appear in the same geospatial level of the hierarchy, such as two cities in the same state. We use the agreement among the co-occurring toponyms to disambiguate one another. This method works extremely well when the adjacent locations are hierarchically related, such as:

*"I'm at The Teapot (Oldbury, Sandwell) http://t.co/QamAYjsA"*

The gazetteer includes Oldburys with country codes "au," "gb," "jm," "zw," and Sandwells with "gb" and "us." We take the intersection of the two countries to determine that the best candidate is among the cities in Great Britain, because "gb" appears as a country code for both Oldbury and Sandwell gazetteer candidates.

Here is another example,

*"@ITFCbrooksy11 Is that where we are likely to go? Holland or Germany, maybe Norway?"*

The toponyms extracted from this tweet do not agree with respect to the gazetteer country code, so we cannot use the country code to help disambiguate locations. However, we have coded this as a rule although it is not encoded into the feature vector. The comma group rule is triggered when we have multiple toponyms in context that cannot be resolved by the machine learning algorithm [28].

#### 5.2.3 Features: Twitter metadata

Users have the option of entering a home location upon registering for a Twitter account. This user-registered location appears in the metadata accompanying every tweet that the user posts. Table 1 contains an example of what may be found in the user location field. Note that if the user moves to another city, a mismatch will be found between actual location and location registered formerly. It has been found that only 66% of users fill in the user location field [18].

We employ data from the user location field only when it is needed, and we disable the feature when it is not. We do this by considering when the user location and a toponym in the tweet are possibly in the same country or same state, then we assume that user location is valuable. If not, we disregard the user location field. Our feature thus indicates whether the user location and extracted toponym share a country and state. If we have multiple toponyms in a tweet, and multiple location expression in the user-registered location field, then our strategy is: 1) find the common country for the toponyms in the user location field; and then 2) see whether that country matches each toponym in the tweet text. We incorporate this as a feature, and let the optimization procedure decide whether or not it is reliable.

The metadata in each tweet could carry geographical information about the user in the fields of user registered location, time zone, user description (a short biography of whatever the person wants to share), or GPS coordinates of the device that posted the tweets. These fields are not invariably filled in by the user, although time zone is supplied automatically by Twitter. This section describes how we turned some of this information into features.

We chose not to mine location information from the user description field. This is because the field contains common words that introduce false positives in matching with place names. Adding this error does not balance the meager benefit from rare location information in this metadata field.

*Tweet coordinates*

The Twitter "place" fields allow more precise location information for each tweet. A bounding box of the location is provided for each place. Figure 1 shows how the place pin on the interface of the drop-down menu below the tweet box corresponds to the location in the tweet.


Figure 1: This screen-shot shows a user checking that the default location Pittsburgh, PA corresponds to the message.

In order to determine whether the Twitter "place" fields would be useful features, we calculated the Euclidean distance between these Twitter "place" field coordinates (when known) and the coordinates for each candidate for the toponym mention. We then rank each candidate by its distance to the actual gazetteer coordinates. However, owing to the sparseness of the data, we did not incorporate this into the final model.

Just how sparse are the Twitter "place" fields in the Twitterverse? Informally, we downloaded 1 million English language tweets from January 2, 2013. Of these tweets, there were 2169 tweets with location mentions in the tweet, but only 51 tweets with Twitter "place" fields (.17%), and only 50 tweets with the GPS coordinates of the user's mobile device (.16%). This information is so rare that we cannot use it to train a model.

Time zone is added by Twitter at the time of the message posting. Time zone represents the whole longitude for that time zone. We experimented by adding time zone into the feature set along with population, alternative names, and string similarity features. Because time zone only decreased the accuracy, we did not add it to the model. We considered using the time of day of the tweet to help disambiguate a location. We would assume that

tweet traffic will be fairly low between 2am–5am on weekdays. However, some tweets will not fit the traffic pattern, so we did not attempt to use it as part of the model.

Only one out of 1000 times do people include geographical coordinate in their tweets. However, once this information exits in the metadata, it is a useful source to determine the location in the text. The assumption is that when the coordinates are available, the tweet text is related to that geographical location.

#### 5.2.4 Features: word form

We looked for the morphological similarity between the extracted toponym and the gazetteer candidates. We used normalized edit distance to denote the similarity. The distance measure is shown below:

$$sim(T, C) = 1 - \frac{edit(T, C)}{min(len(T), len(C))}$$

where $edit(T, C)$ measures the edit distance between string $T$ and $C$. If the target word is identical to the candidate, then the similarity is 1.

There are some cases in which we should not apply the similarity measurement, as in the case of US state abbreviations. CA and California share the same GeoNames ID. Searching the gazetteer for CA will, however, yield a low string similarity value. We know that the abbreviation is a match with the state name because of rules to identify the abbreviations.

The introduction of alternative names from the gazetteer improves recall, while decreasing precision. This word form feature helps neutralize the precision side-effects caused by improving recall.

### 5.3 Using features to select the best match among gazetteer candidates

Using candidate-by-candidate classification to generate one candidate per toponym allows us to rely on standard precision and recall measures.

Use of classification for bipartite ranking may produce conflict. Instead of generating a single candidate, two sets of candidates my be generated, where one set ranks higher than the other based on the predicted feature values. Moreover, the gazetteer candidates in the highest-ranked set may or may not agree as to the location that matches the extracted expression. When the candidates do agree (as in the "London" and "City of London" example in Table 2), we choose the entry with the largest population or with the common geographical feature type (as appears in the GeoNames gazetteer). When those candidates do not agree, we have generated a new method to resolve the conflict and determine which candidate is the best. This is done by generating a weight for each candidate. During the process of classification, each candidate receives a predicted label and the corresponding probability related to the distance from the data point of the candidate to the margin of the decision boundary between the two sets. The further the data point is from that margin, the more confident is its accuracy.

Suppose we have $m$ candidates that are predicted as true. Here $C_t = \{C_1, \ldots, C_m\}$ is an indicator function that generates one or zero, depending upon the argument. Each $C_i$ has a corresponding coordinate $G_i$ and the probability $P_i$ that is generated by the SVM. Among those $m$ predictions of coordinates, suppose we have $n$ unique coordinates, $Cord_1 - Cord_n$.

Then for each $Cord_j$, the probability that each data point is correct is given by

$$P(Cord_j) = \frac{\sum_{C_i \in C_t} I(G_i \sim Cord_j) P(C_i)}{\sum_{C_i \in C_t} P(C_i)}$$

where $I(argument)$ is the indicator function determined by the truth of the argument, and $G_i \sim Cord_j$ indicates that $G$ is close to $Cord$ geographically. Then we select the best prediction about the coordinates by:

$$Cord^* = \underset{j \in (1,n)}{\operatorname{argmax}} P(Cord_j).$$

The best choice $Cord^*$ is the one that maximizes $P(Cord_j)$. The features such as country code, state code, population, time zone, that correspond to the candidates with coordinate $Cord^*$ in the gazetteer can be used to disambiguate the extracted toponym.

## 5.4 Preference learning for the classifier

We categorize our machine learning method as preference learning because it induces a predictive preference model from the data. "Learning to rank" [29] is the preference learning problem that has attracted most attention in recent years.

Three types of preference learning problems are label ranking, instance ranking, and object ranking [11]. Our work falls into the category of instance ranking. This is because the ground truth in our training data can be seen as ordinal with only two values (1 for the true candidate, and 0 for the false candidate) for the bipartite, or instance ranking. That is, the prediction results are grouped into two sets, and within each set, the items are not ordered.

The goal of instance ranking is to minimize not the error in classification, but rather the error in ranking. In order to minimize the ranking error, we transformed the features in such a way that the ranking information is incorporated into the classification model. This allows us to use the classification framework to do the ranking without changing the classification model per se.

We thus define the problem as classification of every candidate entry from the gazetteer candidate retrieved by the toponym expression. We investigate to what extent the information from the tweet metadata can help, and also whether the gazetteer and geospatial context information that are widely used in general toponym resolution research are effective for toponym resolution in tweets.

## 5.5 Classification method

The classification step could be replaced with a ranking function, where each candidate can be assigned a value that is calculated by several heuristic rules, and then the top candidate is selected as the resolution result. The problem is that there are multiple rules and we do not know how much each rule contributes to the candidate ranking score. Here we tested each rule individually by converting them into features. We use SVMs to help us optimize the best combination of the features based on the training data.

SVMs are a type of supervised machine learning algorithm that uses features made from characteristics in labeled data to predict classifications for non-labeled data. The data are represented as vectors, along with the classification labels from the training data. The

classification is binary: a given instance is either in or out of a category. An SVM is a linear classifier. The use of kernel functions within the model allows non-linear data to be treated as linear. Here, we used the radial basis function kernel.

First we provide a definition of terms to clarify our method. A toponym mention in the text is generated from the toponym recognition phase is denoted as $T$. The $n$ candidates that are matched in the gazetteer are $C_T = \{C_{T_1}, C_{T_2}, \ldots, C_{T_n}\}$, where each $C_{T_i}$ is a combination of ID and a vector of $k$ features, $C_{T_i} = \langle ID, f_{T_i} \rangle$, where $f_{T_i} = \langle f_{T_1}, f_{T_2}, \ldots, f_{T_k} \rangle$. $L(T)$ is the ground truth label set for toponym $T$ where $L(T) = \{C_{Tl1}, \ldots, C_{Tlm}\} \in C$, $|L(T)| = m$.

Below we describe the features that we use for the model, and how we transform the feature values to enable the candidate ranking through classification. Those features also try to handle the gazetteer problem (with multiple ID labels), incorporating metadata, and handling missing fields. Several indicator functions are used as features to help manage missing data in the user location or tweet coordinates field. We transform the feature absolute values into feature relative ranking values for learning the candidate preferences, and perform learning and prediction using the SVM. Finally, we find the correct geographical features and coordinates by choosing the top-ranked candidate when multiple candidates are classified as true.

# 6 Experiments on features and data set size

The purpose of these experiments is to show 1) the effectiveness of the ranking features over the absolute value; 2) the score increase enabled by adding metadata information; 3) how the evaluation scores change with the training data size, so as to find the minimal training size needed, and observe the stability of the classifier through cross validation.

We used 300 tweets for training and 100 for testing for Tables 6–13 concerning the population and alternative names features, user location, and tweet coordinates. These experiments do not require much training data because there are only a few features, which brings a low risk of overfitting. For Figures 2–4, we increased the amount of data to use 79% of the full 956 tweets for training, and the remaining 21% (200 tweets) for testing. Moreover, we used cross-validation on experiments 3 and 4, so that the training data scales to between 50 and 750. We evaluated the mean and variance across different runs with the same amount of training and test data to make the results more reliable.

We established a baseline combination of gazetteer features before we ran the experiments. To test the effectiveness of the other features, we added each of the contextual features or metadata features separately to the baseline, to see to what extent each feature contributes to the overall scoring.

The geocoding result is measured only on in-gazetteer entries (that is, we do not measure geoparsing: whether all locations in the tweet were found correctly). This is because the out-of-gazetteer locations actually measure the first stage of geocoding which consists of geoparsing.

## 6.1 Evaluation metrics

Geocoding is evaluated using the same precision, recall, and F1 measures as other types of disambiguation tasks, such as word sense disambiguation. In this experiment, we use the

standard definition of precision, but adapt recall as follows:

$$Precision = \frac{\text{\# of correctly predicted positive candidates}}{\text{\# of total positive candidates predicted}}$$

$$Recall = \frac{\text{\# of correctly predicted toponyms}}{\text{\# of gold standard toponyms}}$$

Due to the transformation of the toponym resolution problem into a candidate classification problem, the tagging result for each toponym may not be unique. For example, sometimes a toponym could correspond to several true candidates, or several mixed true and false candidates. This occurs because of the candidate-by-candidate classification setting of the problem. Thus, we used a precision metric to evaluate the precision on a candidate level instead of on a toponym level. This is because SVM will only generate candidates, so only by evaluating on the candidate level can we determine the true performance of the SVM. So we used the candidate precision so as to make sure that we are able to evaluate the machine learning algorithm only, without the interference of non-machine learning aspects. However, in order to make the system practical, we had to make a candidate selection method that helps us to select the single candidate when multiple results conflict. For recall, we were able to evaluate on both candidate and toponym level, without help from the candidate selection method. The strategy is that once we found a true candidate for the location extracted, that location was tagged as correct. The recall metric reflects this.

Notice also that our gazetteer candidate precision and recall will be lower than that of the precision and recall for the extracted location. For recall, that is because there might be more than one correct candidate, and the algorithm might not select all of them. For precision also, the selection process will introduce errors. Calculation of the extraction error, on the other hand, is straightforward without additional sources of error.

If we use the gazetteer candidate level precision and recall, we are evaluating the machine learning algorithm. However, we would be then unable to evaluate without the help of the toponym selection strategy just how each extracted location is recognized. On the other hand, if we use the extracted location precision and recall, we will introduce the error from the location selection strategy, and we will not be able easily to determine the source of the error. So the reason for choosing the candidate level precision and toponym level recall is to narrow the gap between "evaluating the toponyms" and "evaluating the machine learning method."

Researchers sometimes include the spatial element of the error, or the offset, as a physical measurement (for example, 366km as the difference between the actual and expected answer). However, the number of on-the-ground units between the actual and expected answer is less important than whether the geospatial hierarchy was preserved, since a relatively small offset in Europe might mean that the toponym was resolved to the wrong country—a worse mistake than resolving to the right country but the wrong state. So while presenting our results in the standard format, we also introduce a method of evaluation that is more logical to the data in question than standard methods.

## 6.2 Results for the gazetteer features

We considered how each feature affected the final score. Each feature was trained separately on the same 250 tweets, and tested on the same 140 tweets. We test (1) gazetteer features: population, alternative names, and geographical hierarchy; (2) surface similarity

with respect to morphology; (3) tweet text features; and (4) tweet metadata features (coordinates and user-registered location).

*Population*

We experimented with three population features for training and testing, namely relative population ranking, relative population ranking with normalization, and absolute population value. Relative population ranking is generated by ranking the population values of the list of gazetteer candidates, and population value is converted to the corresponding ranking value. We created a feature for normalized population by setting any gazetteer entries that did not include population estimates to -1 instead of ranking them with the non-zero population values, so as to increase the gap between them and the non-zero population candidates. We created a separate feature for the absolute value of the population.

| Feature | Precision | Recall |
|---|---|---|
| Population rank normalized (so that 0 population becomes -1) | 82.43 | 73.49 |
| Population rank | 79.39 | **78.92** |
| Population absolute value | **93.22** | 26.19 |

Table 6: Comparison of three population feature types trained on 250 tweets and tested on 140 tweets.

The comparison of three feature value settings is in Table 6. The table shows that when using the feature for population rank, recall increased substantially. Although the precision dropped about 10%, we still favor the population ranking feature because of the high F1 value. The normalized population gives higher precision but lower recall compared to the population feature because distinguishing the zero population case helps us better discriminate zero population and non-zero population candidates. However, the recall drops for the same reason.

*Alternative names*

We added alternative names to the population feature to see whether there would be a change in scores, and to compare different feature values. Similar to population, we examined the alternative names as a count ranking, alternative names as a binary (either zero or one/multiple names), and alternative names as an absolute value (when the population was missing, we assigned it $-1$, in distinction to a population that the gazetteer has assigned as zero). The results appear in Table 7.

| Feature | Precision | Recall |
|---|---|---|
| Population + alternative names count ranking | 76.88 | **80.72** |
| Population + zero or non-zero alt names (binary) | **79.39** | 78.92 |
| Population + alternative names absolute value | 77.24 | 72.29 |

Table 7: Comparison of different alternative names ranking combined with population feature, trained on 250 tweets and tested on 140 tweets.

The training and testing used the same data as in Table 6 for population. Table 7 shows that the alternative names ranking value, and the binary ranking value is better than using the population feature alone. Alternate names help to identify a place, for example in cases where the toponym does not have a population listed in the gazetteer (such as a tourist site), where the number of alternate names might be abundant.

*Geographical features*

We experimented with two different geographical feature types. One is the geographical attribute as given in the GeoNames (such as ADM1), which is converted into a vector (using "one hot encoding"). The other is the geographical feature hierarchy (ADM1 would be a higher level than ADM3). The results are in Table 8.

| Feature | Precision | Recall |
|---|---|---|
| Baseline + geographical feature vector | **0.8421** | 0.6987 |
| Baseline + geographical hierarchy type | 0.8 | **0.7108** |

Table 8: The baseline (gazetteer features of population, alternative names and string similarity) added to the geographical feature vector, and the spatial hierarchy feature.

In Table 8, we can see that the addition of these features does not outperform the use of the population and alternative name features in Tables 5 and 6. These geographical features, therefore, were not integrated into the model.

## 6.3 String similarity and string surface form indicators

The string similarity feature combines string similarity with: (a) an indicator for abbreviation and (b) indicators of whether there is a single candidate in the gazetteer as a potential match and whether there is a single candidate extracted from the tweet. This string similarity feature is used to adjust the model to favor the candidates whose name is the most similar to that of the extracted toponym in the tweet, taking into consideration abbreviations as a special case. String similarity represents a good feature combination that improved the overall precision and recall.

Table 9 compares features in the data using five-fold cross validation. The variance is shown by the standard deviation added in parentheses. The variance of precision and recall increases when the similarity features are added.

| Features | Precision | Recall |
|---|---|---|
| Population | 79.08 (2.50) | 78.76 (2.44) |
| Population and alternative names | 80.12 (2.97) | 80.54 (3.62) |
| Baseline: Population, alternative names, and String similarity and abbreviations | **82.60 (2.84)** | **81.75 (3.47)** |

Table 9: Mean (with standard deviation in parentheses) of precision and recall for the gazetteer features of population and alternative names, and the string similarity and abbreviation feature.

## 6.4 Context features

The contextual features represent a search for common country and state in other location expressions found in the tweet text. We tested separately the feature for common country and common state for every gazetteer candidate.

Table 10 is generated by five-fold cross validation of the data set. As we can see, the common country increased the baseline accuracy. However, the recall is lower than the increase made possible by common state. From the standard deviation comparison of common country and common state, we can see that, using the common state feature can generate a more stable result, while increasing the mean. This means that the common state is able to better show the influence of the neighboring toponyms, because entries for states in the gazetteer include additional information about the country, and so carry more information that could be used for disambiguation than the common country feature.

When we added indicators for both common country and common state to the baseline, the precision is in between that of using common country only and adding common state only. Recall increased more than when we added either of the features individually.

| Feature | Precision | Recall | F1 |
|---|---|---|---|
| Baseline + Common country feature | 84.14 (2.76) | 83.29 (3.64) | 83.71 |
| Baseline + Common state feature | **86.39 (2.01)** | 84.13 (2.67) | **85.25** |
| Baseline + Common country + Common state | 84.99 (2.6) | **85.45 (1.98)** | 85.22 |

Table 10: Mean (standard deviation) of precision and recall for the contextual features plus the baseline of gazetteer features of population and alternative names.

## 6.5 Metadata features

*Tweet coordinates*

The tweet coordinates feature indicates the location of the user's device when he or she posted a tweet, and not necessarily a location mentioned in the tweet message. However, we have found by inspection of 956 tweets differences between GPS-coordinates and location mentioned in the tweet are rare (see Table 11). This enables us to use the geographic distance between the GPS coordinates and the coordinates of the location mentioned in the tweet as a feature. It suggests also that users typically tweet about events and places that are relevant to where they are standing with a mobile device.

Precision and recall jump to another level with the addition of the tweet coordinates, as shown in Table 11. Precision increases without decreasing recall, which means that the feature shows no sign of overfitting. Remember that coordinates are attached to a very small proportion of what is actually tweeted.

| Feature | Precision | Recall | F1 |
|---|---|---|---|
| Baseline + Coordinates | 90.25 (3.14) | 85.47 (2.54) | 87.79 |

Table 11: Mean(standard deviation) after adding the coordinates feature to the baseline. Five-fold cross validation with 300 tweets for training, 100 for testing.

*User location*

We tested three user location features along with the baseline (Table 12). One represents the availability of user location data (whether there is any user location data or not). Another feature called "User Location Overlap—Country" is an indicator of whether the user location overlaps with the toponym candidate on the country level. A further feature called "User Location Overlap—State" is an indicator of overlap on the state level.

The state overlap conveys more information than the country overlap (see Table 12). However, when we added at the same time country and state, the output for recall was almost the same. This suggests that the loss of the recall comprehends only a few toponyms. Compared to the baseline, we did not see an improvement with the user location features overall, although the variance was reduced greatly by adding the "User Location Overlap—State" feature only. One possible explanation is that the user location field adds more noise than disambiguation benefit.

| Feature | Precision | Recall | F1 |
|---|---|---|---|
| Baseline, User Location Overlap—Country | 80.51 (2.75) | 78.34 (3.96) | 79.41 |
| Baseline, User Location Overlap—State | 80.89 (1.78) | 79.63 (2.33) | 80.26 |
| Baseline, User Location Overlap—Country, User Location Overlap—State | **82.11 (3.13)** | **79.66 (3.04)** | **80.86** |

Table 12: User location information overlap.

## 6.6 Summary of best-performing features

Table 13 shows the outcome of combining features that performed best. The most effective combination is the baseline + context (common country or state of other locations found in tweet text). The Table 13 experiment is done on 300 tweets for training and 100 for testing, with five-fold cross validation in order to compare the features in a uniform setting.

| Feature | Precision | Recall | F1 |
|---|---|---|---|
| Best combination (Baseline + Context of Common country + Common state) | **84.99 (2.6)** | **85.45 (1.98)** | **85.22** |
| Baseline + Context + User Location Overlap—Country, User Location Overlap—State [Note: these are in the final model] | 84.88 (1.27) | 80.01 (3.25) | 80.01 |

Table 13: Combination of features, and comparison to those that are in the final model.

The features we implemented in the current version of our geocoding algorithm, then are (1) the gazetteer features of population, alternative names, and geographical hierarchy; (2) surface similarity with respect to morphology; (3) tweet context features; and (4) tweet metadata features (coordinates and user-registered location). These features reflect the tests described above. The features also reflect how we gathered the tweets (we did not have a tweet stream for a single user available for testing, for example). We show that some of the features we tried did not improve geocoding performance, so we did not include them in the machine learning model.

In 400 tweet-size experiments it was the combination of the following features that yielded the highest accuracy in geocoding locations in tweet text:

1. Gazetteer features (baseline)
   - population of place in gazetteer entry.
   - number of alternative names within an entry, and among matching entries.
2. Morphology (baseline)
   - string match between extracted location and location candidates in gazetteer, also taking into account abbreviations in the text.
3. Tweet context (context)
   - other location expressions mentioned in the tweet (country or state in common with location in text).
4. Tweet metadata
   - GPS coordinates from user mobile device.

## 6.7 Size of training data set

Data annotation for geocoding is time consuming. In the case of this data, the annotation procedure consisted of finding the correct physical location(s) in the gazetteer to correspond to the extracted location expression in the tweet, and assigning latitude and longitude coordinates.

We wanted to see how the training data set would correlate with the overall performance of the algorithm to test whether the models fit the data. We can tell this by the F1 scores. First we scaled the training set in increments from 50 to 300 tweets. Figure 2 shows the F1 performance for the top-ranked gazetteer candidate per toponym extracted on the following feature combinations:

- $baseline$ = population, alternative names, and string similarity and the indicator of whether the expression is an abbreviation.
- $baseline + context$ = population, alternative names, and string similarity and the indicator of whether the expression is an abbreviation, plus using any other toponym expressions in the tweet to disambiguate the one in question.
- $baseline + user\ location$ = population, alternative names, and string similarity and the indicator of whether the expression is an abbreviation, plus any other toponym expressions in the user location field.
- $baseline + context + user\ location$ = population, alternative names, and string similarity and the indicator of whether the expression is an abbreviation, plus using any other toponym expressions in the tweet to disambiguate the one in question, plus any other toponym expressions in the user location field.
- $baseline + GPS\ coordinates$ = population, alternative names, and string similarity and the indicator of whether the expression is an abbreviation, plus the GPS coordinates of the user's mobile device.

Examination of Figure 2 shows that the performance order of the five feature clusters beyond 100 tweets generally remains the same as the data size increases. Recall that we defined the F1 by taking the score for only the top-ranked gazetteer candidate per toponym

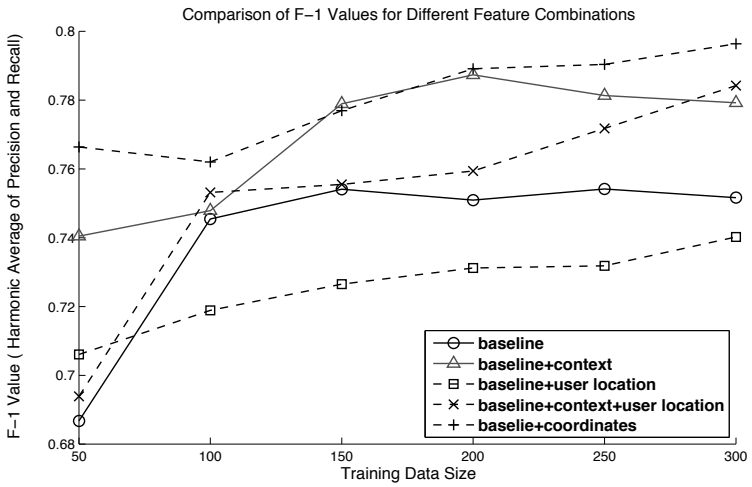



Figure 2: The dynamics of F1 to the size of the dataset.

extracted. For the baseline feature, which is the combination of all the gazetteer features, the F1 stops increasing at 150 tweets, as does the baseline + context feature combination. This indicates that the gazetteer and context features do not need much training data to reach the best performance. However, there are three lines (baseline + user location, baseline + context + user location, and baseline + coordinates) that continue to improve as the size of the training data set increases. At 300 training tweets, the baseline + context + user location outperforms baseline + context, which means that the introduction of the user location information is valuable.

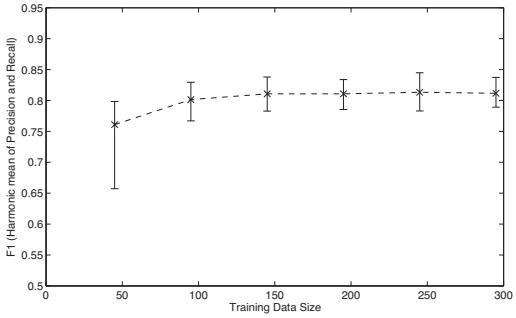
We also provide the specific F1 mean, minimum, and maximum for each training set size to show the reliability of the feature combinations. Notice from the five charts of Figure 3, that: (1) when an additional feature is added to the model, (compare Figures 3a, b, c, and e) the variance increases (Figure 3d uses a different feature set); (2) the optimum amount of training data differs depending upon the feature set.

Based on the experiments with the 400 tweets, we wanted to learn how much data is sufficient for the accuracy to stabilize. We saw from Figure 2 that the highest performing feature combination is baseline + user location + context. This is probably because the representation is richer so there is more potential to attain a better result than simpler feature combinations. We increased the data size to 750, and saw an interesting error rate reduction, as shown in Figure 4.

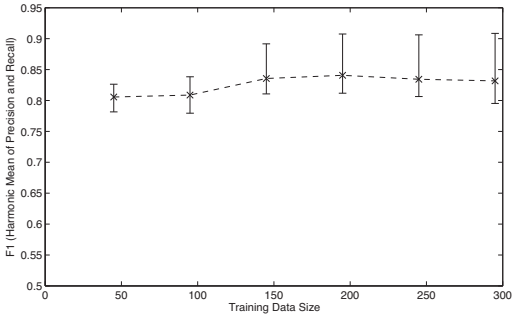
# 7 Error analysis

## 7.1 Evaluation of geocoding output

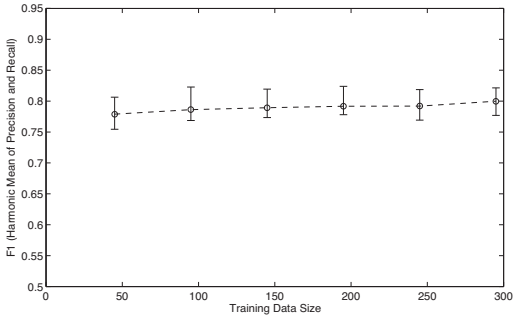
A geocoding algorithm can be evaluated in two phases: 1) whether the extracted expression was mentioned in the gazetteer or not (geoparsing); and 2) whether the extracted location was matched with the correct gazetteer candidate (geocoding). For example, “Vegas,” the



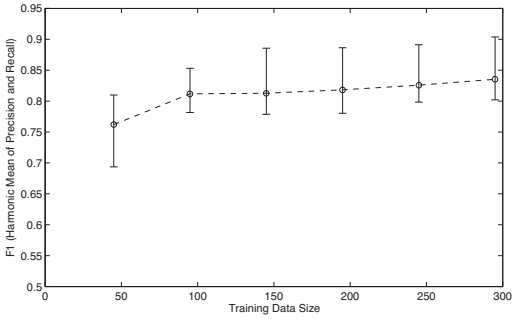

(a) Baseline with the mean, min, and max values for F1.



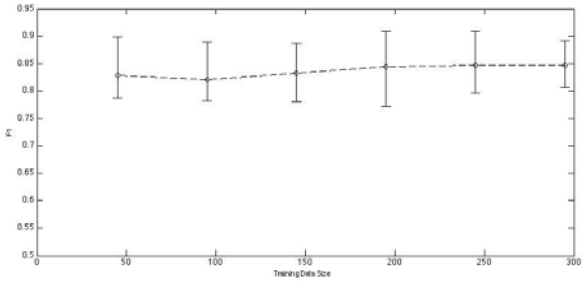

(b) Baseline + context with the mean, min, and max values for F1.




(c) Baseline + user location with the mean, min, and max values for F1.




(d) Baseline + user location + context with mean, min, and max values for F1.



(e) Baseline and coordinates showing the mean, min, and max for the F1.

Figure 3: F1 mean, minimum, and maximum for each training set size with each feature combination

short form of an in-gazetteer toponym, did not match with candidate "Las Vegas" in the gazetteer. In this paper, we only measure the geocoding result on the in-gazetteer entries, that is phase two.

Table 14 shows a division of our F1 errors based on the spatial hierarchy. Note that we have included continent-level error at the country level.

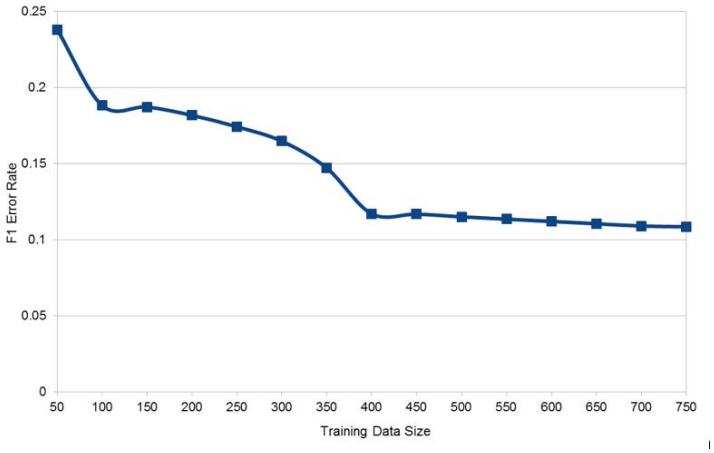



Figure 4: Error rate as shown in declining F1 value with different numbers of training tweets. Feature combination shown is baseline + user location + context.

| **Level of hierarchy** | **Location was left out** | **Location was resolved incorrectly (false positives)** |
|---|---|---|
| Country (or continent) | 47.62% | 54.55% |
| State | 14.28% | 31.82% |
| City | 38.10% | 13.64% |

Table 14: F1 errors based on level of the spatial hierarchy.

## 7.2 Explanation of geocoding errors

We used our best feature combination, and observed the errors made during cross validation. We were looking only for toponyms (political and natural features), and not for buildings and streets. Here are some types of mistakes that recurred, and how we might solve them.

**Error type 1.** *Country level mistakes*

"finland, south korea, hong kong, japan and singapore are all better educated than the uk"

In this tweet, the algorithm makes errors when it uses one place name to disambiguate another. So we created some additional rules which are invoked when a candidate from the gazetteer is not generated by machine learning: a "comma group" approach, that uses sentence syntax (nouns followed by commas) to infer spatial hierarchy [28], and a maximum population rule that uses the maximum population to infer which candidate is correct.

**Error type 2.** *Not enough use of non-geographic words.*

"rt @eu_re_ka_c: i actually want the lakers to win tonight b/c i hate the heat a little more than la. #thelesseroftwoevils"

In this example, "la" is resolved to the place higher in the geospatial hierarchy, the Lao People's Democratic Republic, instead of Los Angeles. But any basketball fan would know that their team, the Lakers, come from Los Angeles.

**Error type 3.** *Mistakes in balancing the features.*

"I'm at Chermine Bahous (Ajaltoun, Mount Lebanon) http://t.co/FnlsWgFs"

Rarely, the model has all the information to make the correct selection, but selects the wrong choice due to an imbalance of features. In this example, the locations are Chermine Bahous, Ajaltoun, Mount Lebanon, and the user location is Ajaltoun. Chermine Bahous is not in the GeoNames gazetteer. But we have two Lebanese locations in the tweet, as well as a Lebanese location in the user location field.

# 8 Comparison of our geocoder to similar algorithms

Our model draws geographic information from other fields in the JSON (JavaScript Object Notation) file of each tweet. We compared our method to that of Yahoo Placemaker and CLAVIN (Table 15). We used 390 test tweets supplied to each system for a geocoding evaluation. We provided to Placemaker and CLAVIN all the metadata that is in every tweet in the test set, by concatenating the user location, user description, and coordinates with the original tweet message. We know that Placemaker is able to use the additional information because we observed the difference of disambiguation between adding and not adding that information.

We would have compared our geocoder to the Microsoft Bing Geocoder. But Bing Geocoder does not accept a sentence as input to perform geoparsing and geocoding; it only accepts only the word itself. We also tried to compare our approach to software called GeoMaker. However, the results were identical to those of Yahoo Placemaker, so we infer that the GeoMaker uses the API from Yahoo Placemaker.

*Scoring*

We scored each toponyms' location separately, even though the Yahoo system outputs locations in chunks. For instance, a tweet "Supply Cha…– #Hartford , CT (http://t.co/KXkPEJcR) Get Supply Chain Management Jobs" has two locations "Hartford" and "CT." Yahoo Placemaker will output the "#Hartford, CT" and resolve them as a whole. And the resolved tag is only for "Hart-ford." However, we count it as two correctly recognized locations. The outcome is shown in Table 15. We include the F1 only for the final geocoding.

For toponym recognition, the recall of Placemaker is 89.41%. It handled locations preceded by hashtags, noisy content, and abbreviations. Even so, our system outperformed Placemaker and also CLAVIN, especially on toponym resolution accuracy and recall. Remember that geocoding for toponyms, as is reported in Table 15, is aligned with toponym selection (that is, geoparsing), so the geoparsing results may affect the geocoding score.

# 9 Future research

*More data to improve geocoding accuracy*

| | **Finding place names (geoparsing)** | | **Disambiguation (geocoding)** | | |
|---|---|---|---|---|---|
| | Precision | Recall | Precision | Recall | F1 |
| Yahoo! Placemaker | 93.37 | 89.41 | 81.82 | 75.14 | 78.33 |
| Berico CLAVIN | 92.35 | 16.23 | 42.79 | 17.86 | 25.2 |
| Our System: Carnegie Mellon Geolocator 2.0 | 91.43 | 90.83 | 85.6 | 81.64 | 83.54 |

Table 15: Comparison of the geoparsing precision, recall, and geocoding precision, recall, and F1 among three tested systems.

Mining locations from a series of tweets by the same user could be helpful in determining which location is meant. Alternatively, we could infer the geographical location of a message based on the Twitterers social network, by considering the tweet text and user metadata information of friends. A similar algorithm could be constructed using other social networks, such as Facebook. We might trace a web page mentioned in a tweet for location-related information (see [44]). Others have used non-location words or events to help classify location words [36, 39].

*Geocoding more precisely*

If we merge geoparsing and geocoding, we should be able to get better results because we will draw additional information from the gazetteer to aid in geoparsing. Conversely, improving geoparsing will improve the geocoding. We could expand the geoparsing to include direction cues such as "north of" a place (even though determining what "north of" means has been shown to vary from person to person [16]). The boundaries of where a place begins and ends, say along a coastline, has been considered by [34]. GeoLocate[20], a platform for geographical referencing natural history collections, is able to infer the coordinates in such contexts.

Research that incorporates natural language understanding, such as phrases that include distances, has been termed research in "spatial relations" [43]. For example, in the sentence "[t]hey had a meeting 200 kilometers west of Washington," the resolved toponym would be West Virginia rather than Washington D.C. This would be especially useful to geographically locate events involving small-scale toponyms not in the gazetteer that contain, intersect, or abut other regions.

*More generalizable model*

The advantage of reliance on an unsupervised rather than on a supervised learning model is that it would not tie the results to a particular data sample. We would use a deep learning algorithm, such as a belief network, or a Boltzmann machine, to create the model. Model creation would be slow, but the features learned from unannotated data could potentially build a model that is more generalizable.

[20] http://www.museum.tulane.edu/geolocate/

## 10 Conclusion

Geolocation research concerns attaching geographical coordinates to location expressions mined from text. Geocoding research mostly has been conducted on news articles, in part owing to the availability of the training data. We collected a set of tweets, and annotated them with the location that appears in the tweet text, along with coordinates for each location. The many potential gazetteer matches for each toponym prompted our adaption of preference learning as a method to rank gazetteer candidates and determine the best match for the extracted location. Thus, our preference learning for multi-candidate classification uses the relative rank of candidates as feature values. This ranking allowed us to achieve high recall as well as high precision.

Geocoding location expressions found in a tweet is more effective when we rely on tweet metadata to provide additional information. We have created a rule-based method in addition to the machine learning model to bring out location cues from the metadata that are useful, and to silence cues that conflict and so cannot aid in disambiguation. Similar research could be conducted with informal text in other social networks, such as Facebook.

## Acknowledgments

We thank Brendan O'Conner for access to the stored tweets from his archive, downloaded from the Twitter Gardenhose/Decahose, at Carnegie Mellon University.

## References

[1] ADAMS, B., AND JANOWICZ, K. On the geo-indicativeness of non-georeferenced text. In *Proc. 6th International AAAI Conference on Weblogs and Social Media* (Dublin, Ireland, 2012), J. G. Breslin, N. B. Ellison, J. G. Shanahan, and Z. Tufekci, Eds., The AAAI Press, pp. 375–378.

[2] AGIRRE, E., AND DE LACALLE, O. L. UBC-ALM: Combining k-NN with SVD for WSD. In *Proc. 4th International Workshop on Semantic Evaluations* (Stroudsburg, PA, 2007), Association for Computational Linguistics, pp. 342–345.

[3] AHLERS, D. Assessment of the accuracy of geonames gazetteer data. In *Proc. 7th Workshop on Geographic Information Retrieval* (Orlando, Florida, 2013), ACM, pp. 74–81. doi:10.1145/2533888.2533938.

[4] ANSELIN, L. Local indicators of spatial association—LISA. *Geographical Analysis 27*, 2 (1995), 93–115. doi:10.1111/j.1538-4632.1995.tb00338.x.

[5] BLESSING, A., AND SCHÜTZE, H. Automatic acquisition of vernacular places. In *Proc. 10th International Conference on Information Integration and Web-based Applications & Services* (New York, NY, 2008), ACM, pp. 662–665. doi:10.1145/1497308.1497437.

[6] BUSCALDI, D., AND MAGNINI, B. Grounding toponyms in an italian local news corpus. In *Proc. 6th Workshop on Geographic Information Retrieval* (New York, NY, 2010), ACM, pp. 15:1–15:5. doi:10.1145/1722080.1722099.

[7] CUCERZAN, S. Large-scale named entity disambiguation based on wikipedia data. In *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (Prague, Czech Republic, 2007), pp. 708–716.

[8] DAHL, G. E., ADAMS, R. P., AND LAROCHELLE, H. Training restricted boltzmann machines on word observations. *Computing Research Repository (arXiv)* (2012), abs/1202.5695. http://arxiv.org/abs/1202.5695.

[9] DAVIS JR., C. A., PAPPA, G. L., DE OLIVEIRA, D. R. R., AND DE L. ARCANJO, F. Inferring the location of twitter messages based on user relationships. *Transactions in GIS 15*, 6 (2011), 735–751. doi:10.1111/j.1467-9671.2011.01297.x.

[10] EDWARDS, S. E., STRAUSS, B., AND MIRANDA, M. L. Geocoding large population-level administrative datasets at highly resolved spatial scales. *Transactions in GIS 18*, 4 (2013), 586–603. doi:10.1111/tgis.12052.

[11] FÜRNKRANZ, J., AND HÜLLERMEIER, E., Eds. *Preference Learning*. Springer-Verlag, 2010. doi:10.1007/978-3-642-14125-6.

[12] GELERNTER, J., AND BALAJI, S. An algorithm for local geoparsing of microtext. *GeoInformatica 17*, 4 (2013), 635–667. doi:10.1007/s10707-012-0173-8.

[13] GELERNTER, J., GANESH, G., KRISHNAKUMAR, H., AND ZHANG, W. Automatic gazetteer enrichment with user-geocoded data. In *Proc. 2nd ACM SIGSPATIAL International Workshop on Crowdsourced and Volunteered Geographic Information* (New York, NY, 2013), pp. 87–94. doi:10.1145/2534732.2534736.

[14] GELERNTER, J., AND ZHANG, W. Cross-lingual geo-parsing for non-structured data. In *Proc. 7th Workshop on Geographic Information Retrieval* (New York, NY, 2013), ACM, pp. 64–71. doi:10.1145/2533888.2533943.

[15] GOLDBERG, D. W., WILSON, J. P., AND COCKBURN, M. G. Toward quantitative geocode accuracy metrics. In *Proc. 9th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences, July* (Leicester, UK, 2010), pp. 20–23.

[16] HALL, M. M., AND JONES, C. B. Quantifying spatial prepositions: An experimental study. In *Proc. 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACMGIS)* (New York, NY, 2008), ACM, pp. 62:1–62:4. doi:10.1145/1463434.1463507.

[17] HAN, B., COOK, P., AND BALDWIN, T. Text-based twitter user geolocation prediction. *Journal of Artificial Intelligence Research 49* (2014), 451–500. doi:10.1613/jair.4200.

[18] HECHT, B., HONG, L., SUH, B., AND CHI, E. H. Tweets from Justin Bieber's heart: The dynamics of the location field in user profiles. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI)* (New York, NY, 2011), ACM, pp. 237–246. doi:10.1145/1978942.1978976.

[19] HOSOKAWA, Y. Improving vertical geo/geo disambiguation by increasing geographical feature weights of places. In *Proc. 2012 ACM Research in Applied Computation Symposium (RACS)* (New York, NY, 2012), ACM, pp. 92–99. doi:10.1145/2401603.2401625.

[20] IRESON, N., AND CIRAVEGNA, F. Toponym resolution in social media. In *Proc. 9th International Semantic Web Conference on The Semantic Web* (Berlin, 2010), vol. 1 of *Lecture Notes in Computer Science*, Springer, pp. 370–385. doi:10.1007/978-3-642-17746-0_24.

[21] JAMEEL, M. S., AND CHINGTHAM, T. S. Compounded uniqueness level: Geo-location indexing using address parser. *International Journal of Computer Theory and Engineering 1*, 1 (2009).

[22] JI, H., AND GRISHMAN, R. Knowledge base population: Successful approaches and challenges. In *Proc. 49th Annual Meeting of the Association for Computational Linguistics* (Stroudsburg, PA, 2011), vol. 1, Association for Computational Linguistics, pp. 1148–1158.

[23] KAUPPINEN, T., HENRIKSSON, R., SINKKILÄ, R., LINDROOS, R., VÄÄTÄINEN, J., AND HYVÖNEN, E. Ontology-based disambiguation of spatiotemporal locations. In *Proc. 1st International Workshop on Identity and Reference on the Semantic Web (IRSW)* (Tenerife, Spain, 2008).

[24] LEETARU, K. Fulltext geocoding versus spatial metadata for large text archives: Towards a geographically enriched Wikipedia. *D-Lib Magazine 18*, 9–10 (2012).

[25] LEIDNER, J. L. *Toponym resolution in text: Annotation, evaluation, and applications of spatial grounding of place names*. PhD thesis, University of Edinburgh, UK, 2008.

[26] LI, H., SRIHARI, R. K., NIU, C., AND LI, W. InfoXtract location normalization: A hybrid approach to geographic references in information extraction. In *Proc. HLT-NAACL 2003 Workshop on Analysis of Geographic References* (2003), vol. 1, Association for Computational Linguistics, pp. 39–44.

[27] LIEBERMAN, M. D., AND SAMET, H. Adaptive context features for toponym resolution in streaming news. In *Proc. 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Portland, Oregon, 2012), ACM, pp. 731–740. doi:10.1145/2348283.2348381.

[28] LIEBERMAN, M. D., SAMET, H., AND SANKARANAYANANAN, J. Geotagging: Using proximity, sibling, and prominence clues to understand comma groups. In *Proc. 6th Workshop on Geographic Information Retrieval* (Zurich, Switzerland, 2010), ACM, pp. 6:1–6:8. doi:10.1145/1722080.1722088.

[29] LIU, T.-Y. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval 3*, 3 (2009), 225–331.

[30] MIHALCEA, R., AND CSOMAI, A. Wikify! Linking documents to encyclopedic knowledge. In *Proc. 16th ACM Conference on Information and Knowledge Management (CIKM)* (Lisbon, Portugal, 2007), ACM, pp. 233–242. doi:10.1145/1321440.1321475.

[31] MUNRO, R. Subword and spatiotemporal models for identifying actionable information in Haitian Kreyol. In *Proc. 15th Conference on Computational Natural Language Learning* (Portland, Oregon, 2011), Association for Computational Linguistics, pp. 68–77.

[32] NOBESAWA, S. H., OKAMOTO, H., SUSUKI, H., MATSUBARA, M., AND SAITO, H. Robust toponym resolution based on surface statistics. *IEICE Transactions on Information and Systems 92*, 12 (2009), 2313–2320.

[33] NOTHMAN, J., HONNIBAL, M., HACHEY, B., AND CURRAN, J. R. Event linking: Grounding event reference in a news archive. In *Proc. 50th Annual Meeting of the Association for Computational Linguistics* (Jeju Island, Korea, 2012), Association for Computational Linguistics, pp. 228–232.

[34] PURVES, R., CLOUGH, P., AND JOHO, H. Identifying imprecise regions for geographic information retrieval using the web. In *Proc. 13th Annual GIS Research UK Conference* (Glasgow, UK, 2005), pp. 313–318.

[35] RICHTER, D., WINTER, S., RICHTER, K.-F., AND STIRLING, L. How people describe their place: Identifying predominant types of place descriptions. In *Proc. 1st ACM SIGSPATIAL International Workshop on Crowdsourced and Volunteered Geographic Information* (Redondo Beach, California, 2012), ACM, pp. 30–37. doi:10.1145/2442952.2442959.

[36] ROBERTS, K., BEJAN, C. A., AND HARABAGIU, S. M. Toponym disambiguation using events. In *Proc. 23rd International Florida Artificial Intelligence Research Society Conference (FLAIRS)* (Daytona Beach, Florida, 2010).

[37] SCHILDER, F., VERSLEY, Y., AND HABEL, C. Extracting spatial information: grounding, classifying and linking spatial expressions. In *Proc. Workshop on Geographic Information Retrieval at SIGIR* (Sheffield, UK, 2004).

[38] SMITH, D. A., AND CRANE, G. Disambiguating geographic names in a historical digital library. In *Research and Advanced Technology for Digital Libraries*, vol. 2163 of *Lecture Notes in Computer Science*. Springer, 2001, pp. 127–136. doi:10.1007/3-540-44796-2_12.

[39] SPERIOSU, M., AND BALDRIDGE, J. Text-driven toponym resolution using indirect supervision. In *Proc. 51st Annual Meeting of the Association for Computational Linguistics (ACL)* (Sofia, Bulgaria, 2013), pp. 1466–1476.

[40] WALLOP, H. Japan earthquake: How twitter and facebook helped. *The Telegraph* (2011). http://www.telegraph.co.uk/technology/twitter/8379101/Japan-earthquake-how-Twitter-and-Facebook-helped.html.

[41] WANG, X., ZHANG, Y., CHEN, M., LIN, X., YU, H., AND LIU, Y. An evidence-based approach for toponym disambiguation. In *Proc. 18th International Conference on Geoinformatics* (Beijing, China, 2010), IEEE, pp. 1–7. doi:10.1109/GEOINFORMATICS.2010.5567805.

[42] WING, B. P., AND BALDRIDGE, J. Simple supervised document geolocation with geodesic grids. In *Proc. 49th Annual Meeting of the Association for Computational Linguistics* (Portland, Oregon, 2011), vol. 1, Association for Computational Linguistics, pp. 955–964.

[43] YUAN, Y. Extracting spatial relations from document for geographic information retrieval. In *Proc. 19th International Conference on Geoinformatics* (Shanghai, China, 2011), IEEE, pp. 1–5. doi:10.1109/GeoInformatics.2011.5980797.

[44] ZHANG, Q., JIN, P., LIN, S., AND YUE, L. Extracting focused locations for web pages. In *Web-Age Information Management*, vol. 7142 of *Lecture Notes in Computer Science*. Springer, 2012, pp. 76–89. doi:10.1007/978-3-642-28635-3_7.