Geoinformatica (2013) 17:635–667
DOI 10.1007/s10707-012-0173-8

An algorithm for local geoparsing of microtext

Judith Gelernter · Shilpa Balaji

Received: 22 March 2012 / Revised: 12 October 2012
Accepted: 5 November 2012 / Published online: 27 January 2013

Abstract The location of the author of a social media message is not invariably the same as the location that the author writes about in the message. In applications that mine these messages for information such as tracking news, political events or responding to disasters, it is the geographic content of the message rather than the location of the author that is important. To this end, we present a method to geo-parse the short, informal messages known as microtext. Our preliminary investigation has shown that many microtext messages contain place references that are abbreviated, misspelled, or highly localized. These references are missed by standard geo-parsers. Our geo-parser is built to find such references. It uses Natural Language Processing methods to identify references to streets and addresses, buildings and urban spaces, and toponyms, and place acronyms and abbreviations. It combines heuristics, open-source Named Entity Recognition software, and machine learning techniques. Our primary data consisted of Twitter messages sent immediately following the February 2011 earthquake in Christchurch, New Zealand. The algorithm identified location in the data sample, Twitter messages, giving an F statistic of 0.85 for streets, 0.86 for buildings, 0.96 for toponyms, and 0.88 for place abbreviations, with a combined average F of 0.90 for identifying places. The same data run through a geo-parsing standard, Yahoo! Placemaker, yielded an F statistic of zero for streets and buildings (because Placemaker is designed to find neither streets nor buildings), and an F of 0.67 for toponyms.

Keywords Geoparse · Microtext · Microblogs · Twitter · Social media · Geographic information retrieval · Geo-IR or GIR · Toponym · Data mining · Local search · Location

1 Introduction

This study examines how to geo-parse social media data to make it more readily usable for applications such as tracking news events, political unrest, or disaster response by providing a geographic overview. Our algorithm suite could be a companion to ones that mine social media streams for user opinions, health trends, or political opinions, for example.

J. Gelernter (✉) · S. Balaji
Language Technologies Institute, #6416, School of Computer Science, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213, USA
e-mail: gelern@cs.cmu.edu

Microblogs are one form of social media. Microblogging services include Orkut, Jaiku, Pownce, Yammer, Plurk and Tumblr, as well as Twitter. Their text is "micro" because the entries are short. To conform to space constraints, the writing is often abbreviated, and informal. The Twitter limit is 140 characters, or about 25 words. We use data from Twitter, the second most popular social network as of this writing, and the ninth most popular site on the entire web.[1]

1.1 Geoparsing location and significant applications

Geoparsing is the process of automatically identifying locations named within text. Examples of geoparsers are the Yahoo! Placemaker, MetaCarta, the geoparser for the Drupal content management system, and the Unlock system from the University of Edinburgh.[2] These parse mostly toponyms, which we define as gazetteer-type entries of towns, cities, states, provinces and countries.

"Location" comes from the Latin *lōcatiō*, which translates roughly as place or site of something that happens. This is relevant since our premier application is disaster response, and many disaster response-related tweets give location to show where a disaster happened. We held a preliminary investigation into precisely what constitutes a location in a Twitter message. We asked people to tag locations in 300 messages, and we developed a definition of location in part based on the results of this pilot study [6].

The solution our algorithm provides is novel. It does what others do not by identifying not only toponyms, but also local streets and buildings. It is for this reason that we do not use one of the standard geo-tagged corpora as are mentioned in [14]. Applications such as the system to mine microblogs for local event information [34], and the earthquake detection system that uses Twitter to map the bounds of an earthquake [23], recognize the need for greater location precision. In light of the informal nature of the medium, we identified places that might be misspelled or abbreviated, as might appear in Twitter messages [6].

Our algorithm parses location names. Other researchers have parsed words that have location significance, such as people's names (Angela Merkel for Germany), demonyms (Irish for Ireland), events (Summer 2012 Olympics for London, England), dialect ("grandpappy," used in Appalachian region of the U.S.). Even words without locational properties may be connected with regions [39]. These methods are legitimate means to locate messages, and at a later stage, might be combined with our place name approach to geo-locate a larger number of messages.

The more precise the location, the more precise the geographical maps describing events can be. Our focus and data sample concern crisis informatics. It has been shown that information search and spread intensifies during emergency events, and that the information produced by social media may be heterogeneous and scattered [32]. A review of social media for crisis informatics appears in [37].

1.2 Finding location of the social message author

We can geo-locate the author of the tweet by consulting the user-registered location or GPS-coordinates associated with the tweet. Data mining methods regularly resort to the user-

[1] These statistics date to February 2012, from http://www.ebizmba.com/articles/social-networking-websites

[2] These are found at the following web addresses as of February 7, 2012: Yahoo Placemaker at http://developer.yahoo.com/geo/placemaker/, Metacarta geoparser at http://www.metacarta.com/products-platform-queryparser.htm; Drupal at http://geoparser.andrewl.net/, and the Unlock system at http://unlock.edina.ac.uk/texts/introduction.

registered location that accompanies a tweet. However, one study indicated that that field is completed by only 66 % of users, and when they do register, they might complete it at the level of city or state [7].

Those who tweet on GPS-enabled mobile devices may have precise latitude and longitude associated with the tweet if they opt for the service.[3] We found, however, that geographic coordinates accompanied only 0.005 % of a sample of the New Zealand earthquake tweets.[4] Those tweets that do come with geographic coordinates tend to be from platform-dependent applications such as UberSocial for the Android (formerly UberTwitter and Twidroyd), and Echofon for the iPhone and Mac.

1.3 Methodology: finding location of message content

We follow the artificial intelligence approach that combines intelligence from many shallow methods [21]. Our methods presently do not find hypotheses that compete because our work is still in progress and we are still experimenting with new techniques. If different techniques to find location did come up with competing hypotheses, we would find a confidence value for each result, to give the result location that exhibited the highest score the location that would be associated with the Twitter message. Our methods are lexico-semantic pattern recognition to identify streets and abbreviations, lexico-semantic matching enriched with gazetteer for spell checking and toponym identification, and machine learning for abbreviation disambiguation and to find buildings (through a third-party algorithm).

1.4 Importance of place references that are local

Types of tweets that tend to be rich in place names are news, commentary, and notices about events or problem areas. What types of tweets tend to include place names that are local? We have informally examined sets of 4000 or more tweets that were mined for a city (Pittsburgh in 2011), an event of large scale (2011 hurricane Irene that crossed several U.S. states), and a disaster of city scale (2011 earthquake in Christchurch, NZ; 2011 fire in Austin, Texas; 2010 and 2011 fires in California). We have found that only the city-scale crises are rich in reference to places that are local.

In what context do people tweet about local streets and buildings? Table 1 gives examples of disaster-related tweets with references to local places. Tweets were sent just following the February 2011 earthquake in Christchurch, New Zealand.

While the total number of these local references is small, as shown in Table 2, the information could be important to disaster response. Preliminary inspection has shown that many of these are info-bearing messages that have been re-posted (re-tweeted), and are therefore shown by the Twitter cohort themselves to be significant or reliable. This could be a fruitful area of further study. A tweet can contain any number of terms in any of our four categories, but our Table 2 statistics assume that each term belonged to only one category.

[3] http://thenextweb.com/socialmedia/2010/04/14/twitter-announces-annotations-add-metadata-tweet-starting-quarter-2/

[4] Our data consists of about 300,000 tweets (1 out of every 1000 tweets of about 300,000,000 per hour) sampled from 1 h of tweets. The tweets were dated right after the earthquake. Takahashi, Abe, Igata, "Can Twitter be an alternative of real-world sensors" LNCS 6763, 2011, found that 0.6 % of tweets had GPS coordinates.

Table 1 Typical local place references in crisis-related tweets. Christchurch earthquake, New Zealand, 2011

Streets	Buildings
Massive amount damage along bealy ave. #eqnz	RT @starrjulie: RT @kalena: Phone lines congested. If anyone knows parents of kids at Diamond Harbour school, pls let them know all are …
Reports of deaths on Colombo Street—bad, bad, bad!!	Any word on St Andrews school? #eqnzContact #eqnz

1.5 Questions guiding research

Our purpose is to find locations in Twitter messages, even if those locations are misspelled or abbreviated. We are particularly interested in geo-locating at the level of city and within a city. We seek references to places that are geo-locatable: named streets and addresses, buildings and urban spaces, in addition to neighborhoods, city, state and country toponyms, and abbreviations for any of these.

We combine lexico-semantic pattern recognition for the identification of streets and some buildings and abbreviations, along with conditional random fields (in third-party Named Entity Recognition software), and geo-matching from gazetteer resources to identify places. Our hybrid approach encompasses techniques that others have treated separately. Papers have appeared on what a location is [35], how to identify abbreviations [2, 19] vs. how to identify acronyms [4, 20], how to identify possible disambiguation text [9] and how to choose the best disambiguation expansion [31].

Questions guiding our research are:

- Can we automatically identify streets and street addresses?
- Can we automatically identify geo-locatable local buildings and urban spaces?
- Can we automatically identify local places referenced by abbreviations as might be found in space-constrained, informal microtext?

Key contributions of this paper include a multi-faceted approach to identifying local streets, buildings and place abbreviations in Twitter messages. The paper proceeds with a review of related work. Next we describe the data we used for this study, and we introduce our research with our working definition of location. Then we present the architecture of our geoparsing algorithm, followed by a detailed description of how the algorithm works (with a step-by-step presentation in Appendix 1 and 2). We present sample output that demonstrates strengths and weaknesses of our algorithm and discuss means to optimize. We describe our evaluation on unseen tweets, and compare the results to that produced by a standard geoparser, Yahoo Placemaker, along with statistics showing algorithm effectiveness. We acknowledge limitations, and conclude with suggestions for future work.

Table 2 In tweets following the February 2011 Christchurch, New Zealand earthquake, percentages show how many of the 2000-tweets sampled include references to local streets and buildings, toponyms or abbreviations and acronyms

Christchurch, NZ Tweets (N=2000)	
Street are named in tweet	1.6 %
Buildings are named in tweet	6.2 %
Toponyms are named in tweet	26.9 %
Abbrev & Acronym in tweet	9.2 %

2 Related work: identifying location

Geo-parsing is a form of information retrieval (Geo-IR). There are various approaches to retrieve, or in this case, identify, locations. This section is organized around the question of how location words are identified: according to syntax (NER), terms or objects or people associated with a location, exact match with location words in a gazetteer, or inference from encyclopedia reference or by probabilistic matching between location abbreviation or acronym and the extended word or phrase that serves to disambiguate.

2.1 Geo-locating text based on classifying noun types (NER)

Geo-parsing entails identifying types of locations. Identifying locations is a sub-problem of identifying all named entities, and so extracting location is often discussed in the context of Named Entity Recognition (NER). The proper nouns which represent locations may be extended to languages, events or landmarks associated with locations such as "French" or "Eiffel Tower" for France [13, 34] and may be identified by combining a K-Nearest Neighbor classifier with a linear Conditional Random Fields classifier to find named entities. The Liu et al. method achieved an F1 of 78.5 % for location entities in tweets [17, p.365]. Named Entity Recognition evaluation is typically cited in terms of recall and precision. Some systems allow the recall–precision spectrum to be shifted toward either end of the spectrum, since setting one factor high tends to sacrifice the other. Standard Named Entity Recognition tools perform less well on microtext than on text, and a Latent Dirichlet Algorithm has been found to achieve fairly good results [30].

2.2 Geo-locating text based on language models

Kinsella et al. [12] draw upon the language modeling approach of Ponte and Croft [29] to create a function to describe probabilistic distribution. The Kinsella group estimated the distribution of terms associated with a location, and then estimated the probability that a tweet was associated with that location. Their language model approach succeeded at the city level at up to 65 % accuracy, but returned results at the neighborhood level in only 24 % of cases (pp. 65–66). Eisenstein et al. built a model to predict the region of the tweet author according to author's choice of vocabulary and slang. Their model could identify authors to the correct state in 24 % of cases [5]. Cheng et al. used a language model to identify the region of the tweet's author to within 100 miles of the author's actual location, and the model worked for 51 % of authors [3].

2.3 Geo-locating text based on gazetteer matching

Lieberman et al. provided a survey of geolocation methods for text [15], although there are specific methods that have been used for Twitter. Paradesi combined Named Entity Recognition and gazetteer methods in her TwitterTagger [26]. The system first assigned part-of-speech tags to find proper nouns, and then compared noun phrases per tweet to the United States Geological Survey gazetteer to identify locations. The system identified nouns that seemed to be places by looking for a spatial indicator such as a preposition found before the location name. TwitterTagger research does not consider what sorts of places are found in tweets, however, and therefore does not account for abbreviations.

2.4 Geo-locating text by association with related geo-tagged documents

Watanabe et al. identified local places to the level of specificity of a building, generating their own gazetteer of places with geographic coordinates by extracting place names from geo-tagged Japanese tweets [38]. They used the information from the geo-tagged tweets to identify places named in tweets that do not have geotags, and they grouped tweets according to shared topic keywords that were generated within a short time and within a limited geographic area. Their system detected local events to an accuracy of 25.5 %. Jung proposed that location for a tweet could be inferred by merging Twitter conversations between people into a single document and using associations among individual tweets to improve recognition of location and other entities [10].

2.5 Geo-locating by association with author's geographic coordinates

Event-based detection systems that use Twitter may rely on individual tweet geo-referencing, as in the Mapster system [18], and the TwitInfo system [19]. The problem is that this Twitter-provided feature is voluntary and few people use it presently, so only a tiny fraction of tweets include latitude and longitude.

2.6 Geo-locating text based on abbreviations and acronyms

Geo-locating text given only location abbreviation or acronym entails first identifying abbreviations and acronyms, and then disambiguating them. An earlier paper by Park and Byrd [27] considered the combination of finding and disambiguating abbreviations, although identifying and disambiguating abbreviations and acronyms are commonly separate research topics.

2.6.1 Identifying abbreviations

Abbreviations in microtext may differ from those in full length documents in that the microtext abbreviations might be lower case without punctuation, and might squeeze non-standard word shortenings to fit the microtext space limit. Pennell and Liu [28, p.5366] defined three forms of abbreviation: those made by character deletion (ex: tmor—for "tomorrow"), substitution (2nite—for "tonight"), or some combination of deletion and substitution (2sday—for "Tuesday"). In the geographical abbreviations that are our focus, our data exhibits mostly the abbreviation by deletion with letters missing.

The often non-standard form of microtext abbreviations makes creating a match list an imperfect strategy, although that was the method used for document abbreviations by Ammar et al. who created a list of abbreviations plus their expansions from the Internet [2], and Vanopstal et al. who disambiguated medical abbreviations based on each article's abstract [36]. A match list of standard abbreviations and Twitter abbreviations is of limited help.

Instead, to identify location abbreviations and acronyms, we followed the method of Adriani and Paramita [1]. Our algorithm checks before and after each word for cues such as prepositions (in, near, to), or compass direction (west, south), or distance (5 km from). We save instances found with these heuristics to use in a second pass over the same data in order to find abbreviation instances that do not benefit from context.

2.6.2 Disambiguating abbreviations and acronyms

Figuring out what an abbreviation stands for is called abbreviation expansion or disambiguation. Difficulties with this task include that a single abbreviation may stand for more than

one concept. Worse, in Twitter, the full form of the abbreviation might be stated nowhere. One approach is that of Jung, who linked tweets to find disambiguation data [10]. This approach has already proven its utility in that Ireson and Ciravegna [9] showed that they could achieve better results resolving locations in social media when they included social network data. In our case, we mine for candidate disambiguation word(s) in any tweet from 1 to 5 days prior to the tweet with the abbreviation or acronym.

Pairing the mined abbreviation/acronyms with disambiguation candidates is generally accomplished by supervised learning. We created a system which learns rules from training examples of how to pair abbreviations and acronyms with their expansions. The problem has been attempted using Conditional Random Fields [16], Maximum Entropy modeling [25], Hidden Markov Models [33], the Tilburg Memory-Based Learner [4], and Support Vector Machines and other classifiers [22]. The selection of the appropriate long form for the short form has been accomplished in the limited domain of programming code using a most frequent expansion (MFE) technique so see how many times a short form was matched to a long form [8]. It has also been solved using scored rules [24], but this was in medical texts where the abbreviations are mostly standard.

3 Study data

Because we intend that one of the uses of local parsing of tweets will be to aid in disaster response and recovery, we selected tweets from a 2011 earthquake in Christchurch, New Zealand, and a 2011 wildfire in Austin, Texas in the United States. Our data represent a random sample from Twitter's publically available Spritzer feed, that itself represents only a fraction of Twitter messages. The data include some repetitive posts of the same message, by the same or different people, called "retweets". Since even a small alteration in a retweet precludes it from duplicating an earlier tweet, we did not remove retweets. In addition, retweets can provide us with more information about the significance of the topic being tweeted because if many people post the same message, it is likely important.

Our Christchurch tweets were collected using either the #eqnz hashtag[5] or tweets whose user-registered location is Christchurch, New Zealand, and time-stamped from noon (a little more than an hour before the earthquake), to 5:24 pm local time after the earthquake. Our annotated data was just less than 4000 tweets following the Christchurch, New Zealand earthquake. We developed the algorithm based on 1987 of these New Zealand tweets, leaving 2000 tweets for algorithm evaluation. Our Austin tweets were collected on the basis of at least one of the keywords "TX, Texas, Austin, Bastrop, evacuate, fire" that were tweeted between September 5–7, 2011.

4 Our definition of place in tweet context

4.1 Arriving at a definition of location in tweet text

Our definition of location in a tweet is based upon our preliminary study [6]. We used a sort of grounded theory approach in arriving at a definition, so that instead of hypothesizing a definition, we let the data speak for itself. We gave participants a few hundred tweets and asked them to tag what they believed to be locations. Then we discussed discrepancies

[5] Twitter users developed their own indexing practices of using a "#" symbol, called a hashtag, to label tweets of a topic.

among their resulting tags. From this study, we (1) arrived at a definition of place in a tweet, and (2) developed instructions as to how to assign location to a tweet to guide further annotations and ensure consistency.

4.2 Examples of location in a tweet

Locations may appear as nouns (sometimes misspelled), or adjectives, or possessives. Examples below show each of our location categories as they appear in actual tweets. Distance or direction are included along with the building, toponym or abbreviation for added precision. [6]

Streets or addresses

- 18 Bismark Dr.
- The 4 avenues

Buildings or urban spaces

- BNZ in Riccarton
- Art Gallery bus stop

Toponyms

- Wisconsin's
- New Zealand News Service
- #Christichurch
- Canterbury residents
- Dunedin City council
- Takapuwahia
- "Christchurch" welcomes you
- 10 miles SE of Newhall

Place abbreviations or acronyms

- LA
- AKL (Auckland)
- U.K.
- 10 km SE of Chch

The above examples are clearly recognizable as place names—except for the metonym ("Christchurch welcomes you"). Metonyms are figures of speech in which one concept substitutes for another. Metonyms for place names are particularly common, in that place names may substitute, for example, for the people who live in a place, or for the government of a place. Leveling and Hartrumpf [13] have a method to recognize metonyms, but it requires context, which is thin in Twitter. We believe that the artificial intelligence required to disentangle metonym from place name given the limited tweet context would be considerable, and so present research considers these as place names. Besides, human annotators did not invariably distinguish between metonym and actual place, so we allow the algorithm to do the same. Lieberman and Samet [14] also considered metonyms to be toponyms.

[6] We would like to add time as representative of distance, since presently we miss the radius around San Bruno in a tweet like "about an hr and a half from San Bruno"

4.3 What does not constitute a location for the purposes of data mining?

Excluded from our definition of location are vague place references such as “city center”, “uptown” or “downtown.” They are not readily geo-locatable because their boundaries are not easily agreed upon. Our algorithm therefore does not mine such references.

Places that cannot be geo-located without more information

- central city
- in the burbs
- welfare centres
- tower junction
- cordoned off area
- a garden
- a dead end street
- a Christchurch mansion

Part of a URL or @mention

- @SkyNewsAust (“Aust” for Australia is not a place)

Demonyms

- Aussies

Co-references

- “city” (when it is implied but not stated that the city refers to Christchurch)
- places preceded by a possessive pronoun (mine, their), relative pronoun (which, what), demonstrative pronoun (this)

Our definition of location presently does not include instances of “city,” even though we know by reading the tweet that, in most cases, “the city” refers to the city where the event is occurring. This problem is known as co-reference analysis, and has been handled in the general case by off-the-shelf packages such as the Illinois co-reference package, or the BART co-reference resolution package if sufficient context is available.[7]

5 Method

The diagram (Fig. 1) shows the flow of tweet data through the geoparsing algorithm. The diagram starts with an interface to allow users to enter search parameters for the tweets, although presently the tweets are pre-collected. Processing includes the identification of streets, buildings and toponyms, and location abbreviations. These steps are sequential. The next version of the algorithm, however, has been designed so that the steps execute concurrently for faster execution (see footnote 18). The output consists only of those tweets which have mined locations, along with the location word(s).

The street, building and spell-check are in Java, with an abbreviation script in Python, since this is a coding language well-suited to text manipulation. The tweets are in JSON originally, and we use a .txt file with one tweet per line as input to our algorithm. We run the algorithms in sequence, and each time the data is processed, we send results listing the tweet and all location

[7] Illinois co-reference package: http://cogcomp.cs.illinois.edu/page/software_view/18; BART at http://www.bart-coref.org/

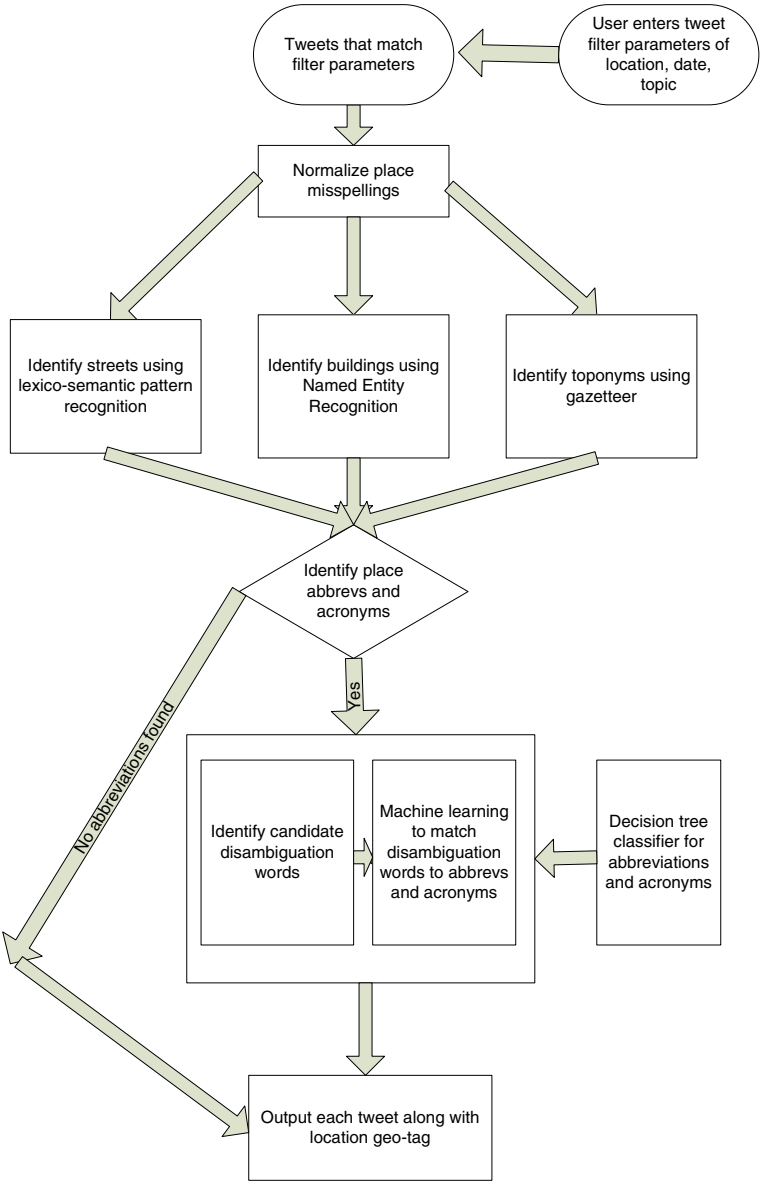

Fig. 1 Diagram shows flow of tweets through our algorithm

matches we were able to find. The abbreviation module output indicates which abbreviation disambiguates to which long word.

6 Local geo-parsing

The purpose of our algorithm, as mentioned in the Introduction, is to geo-locate tweet content so that a tweet can be associated with as precise a location as possible. This section

describes our algorithm's external resources and processing method. It then gives examples from preliminary processing to show what worked initially and what we improved before we ran an evaluation.

6.1 Lexico-semantic approach for streets, buildings, toponyms (detail in Appendix 2)

Three separate processes identify streets, buildings and toponyms. The streets and to some extent buildings are identified by means of lexico-semantic pattern recognition. The toponyms are identified by means of gazetteer matching as well as open-source Named Entity Recognition software. Mis-spellings are corrected through an open-source spell check program.

External resources We have selected resources for their compactness rather than their comprehensiveness to optimize processing. Selection of more comprehensive resources should yield as least as good if not better results, although additional optimization strategies would be needed to gain processing speed.

Dictionaries, etc. External resources include dictionaries and word lists. We use an English dictionary to distinguish between location abbreviations and words that are fewer than six characters. We use also an abbreviation dictionary, a Twitter dictionary, and a list of building types.[8] The place list contains all entries from New Zealand and Australia from the National Geospatial Intelligence Agency (NGA) gazetteer that is used in conjunction with a filter list of common words. This is so that place names that are common words, such as Lawrence, New Zealand, will not be mistaken for the first name Lawrence which might occur more frequently in the data than the place of the same name.

Third party programs that are part of the algorithm include the Named Entity Recognition software OpenCalais, a Part of Speech Tagger developed specifically for Twitter, (see footnote 19) and a spell check algorithm.

Spell check The procedure starts with a third-party spell correction algorithm. We experimented with the Java implementation of the Norvig algorithm,[9] and our preliminary tests have shown it to work well identifying mis-spellings in tweets. We fortify the spell check with (a) gazetteer entries from the county or counties of the event and with (b) buildings, urban spaces and streets from the data set that appear several times (we require three or more repetitions of the same place name so that we do take a mis-spelling as a name).

Examples of how the spell correction algorithm is working:

```
waikoora is corrected to waikoura
lyttleton is corrected to lyttelton
hemilton is corrected to hamilton
welington is corrected to wellington
hristchurch is corrected to christchurch
```

[8] We use the dictionary that loads with every Linux operating system as a dictionary of the English language. We use a dictionary of abbreviations common to Twitter called the Twittonary, which we were granted permission to use in research. We refer also to some minor word lists, such as the buildings list from Wikipedia, and a list of saints' names (to distinguish saints from streets) from http://www.catholic.org/saints/stindex.php

[9] http://developer.gauner.org/jspellcorrect

Our algorithm retains both the given and the corrected spelling of a word to check against the gazetteer as potential matches. That way, if the spell check algorithm made a change, even if the change were wrong, we would have an alternative spelling to look for matches with the gazetteer.

OpenCalais This is a Named Entity Recognition (NER) open source software with a web service API from Thomson Reuters. We use it to find buildings, or what OpenCalais calls "facilities," as well as toponyms. We supplemented it with a building types list from Wikipedia so that it would find a wider range of buildings.[10] OpenCalais is useful because it can identify locations that aren't in our gazetteer and it can automatically disambiguate standard location abbreviations (e.g. UK to United Kingdom). Problems with OpenCalais are that it seems to rely very heavily on capitalization of words, and capitalization is not always grammatical in Twitter messages. Because the "micro" shortenings of microblogs encourage the use of clipped, ungrammatical sentences, aspects of OpenCalais that rely on sentence structure tend to fail. We found that we were able to improve results by matching against our own building list.

6.2 Machine learning for abbreviations and acronyms (detailed in Appendix 1)

Identify short words The algorithm disqualifies as abbreviations those short words that match to the dictionary, and it disqualifies abbreviations that are not place-related by matching against dictionaries of abbreviations. It uses tweet context to indicate which abbreviations are place-related. Cues are preceding prepositions, semantic proximity to a cardinal direction (NE, south, etc.), or semantic proximity to a distance term (yard, mile, kilometer, etc.). Once an abbreviation is recognized as place abbreviation, it is retained and added to a match list so that the same abbreviation lacking context will be identified correctly.

Identify disambiguation phrases We identify candidate disambiguation phrases according to time, mining tweet text that is time stamped *before* the time stamp of the tweet with the abbreviation. Candidate phrases that include verbs, according to the part-of-speech tagger, are disqualified, as we are searching for location names, of which most are nouns. Preliminary examination indicated that location names include verb only rarely. Note that this method of gathering disambiguation phrases has been discontinued in the next version of the algorithm since an inadequate number of disambiguation phrases was found.

The next step was to use a classifier to associate the mined abbreviations and acronyms with the correct disambiguation text. The New Zealand tweet sample contained insufficient examples to train a classifier. Because we sought non-standard location abbreviations as are found in the space-constrained Twitter rather than using abbreviation lists for states, countries and postal codes, we needed to create our own examples. Some of our training data is included in Appendix 3.

Machine learning attributes We do not know automatically whether the short word mined is an abbreviation (bldg. → building) or acronym (ESB → Empire State Building). Hence, we created attributes for both, and included both in the training data. A full list of the attributes we devised appears in Appendix 1. Examples of attributes are "first letter match" (for either abbreviation or acronym), "second letter word match" in which the second letter of the short

[10] http://en.wikipedia.org/wiki/list_of_building_types

word corresponds to the second letter of the disambiguation phrase (for acronym), and "same order of letters" (for either abbreviation or acronym).

Machine learning training data Abbreviations and acronyms we created follow rules of abbreviation such as the short word preserves the order of the full word, and rules of acronym such as the first letter of each word of the long corresponds to each letter of the short, minus stop words.[11] We created 406 non-standard abbreviations for locations. We aimed for abbreviations and acronyms that were not entirely novel, so we checked by searching Twitter and the web to verify that the abbreviation we had created had been used by at least one person previously.

Machine learning algorithm We used the C4.5 decision tree algorithm (the earlier version of J48 in Weka) to classify short words with candidate long words, and create a model we can use to pair short with long words. We then use the per-node probabilities in the decision tree to rank matches at every node such that each abbreviation will have a best match disambiguation, aiming for the correct disambiguation to be ranked the highest. Creating a classifier with the same attributes but with a much larger set of training data would create a classification model that is more generalizable.

Classifier Our classifier model achieves 87.9 % accuracy, with the number of instances at 406. The accuracy statistic is (True positive + True negative)/(True positive + True negative + False positive + False negative). Weka produces a kappa statistic corresponding to the accuracy of 0.748. A statistic of 1.0 would be complete agreement of instances with classes, so our model is quite good.

6.3 Approach examined via error analysis of training data

We ran training data iteratively, performing a separate error analysis for identification of buildings, streets, toponyms and abbreviations. When we conducted each error analysis, we considered whether we could correct not just a particular error (which would be overfitting), but whether we could anticipate and prevent similar errors, without introducing other types of errors.

In Tables 3, 4, 5 and 6 below for streets, buildings, toponyms and abbreviations, we provide an equal number of type 1 and type 2 errors as demonstration, although this balance is not statistically representative of errors found in our training data.

Table 3 shows errors in identifying buildings. Overall, our building errors were mostly type 1 omissions. We missed in many cases the first of two buildings that were named in conjunctive pairs (building y and building z, for example). Our algorithm made type 2 errors when it mined non-specific buildings from the building types list that we did not consider geolocatable. We reduced this error by adding a rule that we do not mine a building word if it is unmodified and is the first word in a tweet.

The complete analysis of streets showed that we were identifying every instance of the abbreviation of "saint" as a street name. This we fixed by downloading a list of saints' names[12] and adding the rule that if a saint's name is found after "st," the word should not be

[11] U.S. airports are found in tweets. But they do not make good training data because U.S. airport abbreviations are forced into a 3-letter mold, and are not supposed to repeat around the country so that many do not follow customary abbreviations rules. For example, LAX stands for the Los Angeles, California airport, and EWR represents the Newark, New Jersey airport. We therefore avoided this sort of abbreviation for training the classifier.

[12] http://www.catholic.org/saints/stindex.php

Table 3 Error analysis of buildings from the New Zealand training set, providing tweets and building names correctly and incorrectly identified, as well as building names missed by our algorithm

Tweet	Correctly identified	Omitted (type 1 error)	Incorrectly identified (type 2 error)
Buildings			
The Ashburton BNZ store will have a BBQ today from 11am. Grab a sausage and help support the people of Canterbury #eqnz	BNZ store		
BREAKING: Camera in CTV building finds 15 people trapped but alive #goodnews #eqnz	CTV building		
Wellingtonians: Donations of non perishable food items and toiletries can be dropped at WGTN or HUTT libraries - the Curly As ... #eqnz		Hutt Libraries	
MT @wilsonvoight: #Welfare Centre: Hagley Park FULL. Make way to Pioneer or Cowles Stadiums #loc Cowles Stadiums #eqnz #chch		Pioneer or Cowles Stadiums	
Liquefaction on Papanui Rd - NOT MINE - #eqnz http:\/\/flic.kr\/p\/9k8Tv4			Not mine
Impressed by one of the guys at the office. He's flying down to chch to help out the farmers cleaning up tomorrow #eqnz			The office

identified as a street. Saints' names will not be a point of confusion in all countries, so this is not a core problem for algorithm generalizability.

Some errors with the building as well as the street routines were solved by greater attention to punctuation. Initially, we stripped the original tweets of punctuation before processing, but this led to mining phrases illogically. Once we preserved splits made by periods, colons, semicolons, and parentheses, we took fewer false positives.

Table 4 Error analysis of streets from the New Zealand training set that shows street names correctly and incorrectly identified, as well as streets that have been missed by our algorithm

Tweet	Correctly identified	Omitted (type 1 error)	Incorrectly identified (type 2 error)
Streets and Addresses			
WATER being handed out on Colombo St (opp Mitre 10 Sydenham). Second truck on it's way. #Christchurch #eqnz quench your thirst ChCh x	Columbo St	Mitre St	
\"oh my gosh\" RT @publicaddress: Citizen video in Lichfield St	Lichfield St		
eeek looks like mum may be stuck @ work now due to road closures this morning #Tirednurse #eqnz			[due to] road
Lyttelton doesn't look too bad at all. 60% damage to one street isn't 60% demolished #eqnz http:\/\/bit.ly\/eNzUEC			One street
New welfare centres have been set up at Pioneer Stadium (75 Lyttelton Street)	75 Lyttelton Street		
Anyone know if 183 Hereford is ok? The Alliance Fran\u00e7ais is there. #eqnz		183 Hereford	

Table 5 Error analysis of toponyms from the New Zealand training tweets that were correctly and incorrectly identified, as well as toponyms that our algorithm missed

Tweet	Correctly identified	Omitted (type 1 error)	Incorrectly identified (type 2 error)
Toponyms (Countries, cities, etc.)			
Hosting a tourist who was brought to Auckland because of #eqnz. Poor thing is frightened	Auckland		
Has received hundreds of corded phones instore at @TelecomNZ to be donated to Christchurch. Keep them coming Wellington! #eqnz	Christchurch; Wellington		
Spare room at my place in Hornby for anyone who needs it.		Hornby	
looking for my nephew #ruok Shuichi Okamoto lives at #loc Ferry Road Ferrymead #chch #eqnz #con @atok98 http:\/\/yfrog.com\/h2cnjdp		Ferrymead	
I wish those media at the #eqnz press conference would stop digging for a reponse on fatality numbers. No point in speculating.			Wish
Looks like our NZ Uni buddies Eion and Hunter are alive and well. #tweetfleet #eqnz			Hunter

A more difficult problem is algorithmic identification of streets and roads that are non-specific. The heuristic that we do not mine a phrase as a street if the street indicator word is preceded by a preposition or short word prevents only a fraction of these errors. So, for

Table 6 Location abbreviations and acronyms from the New Zealand training tweets that our algorithm has mined correctly and incorrectly

Tweet	Correctly identified	Omitted (type 1 error)	Incorrectly identified (type 2 error)
Location Abbreviations			
@BexieLady @Naly_D has a free ticket for the 3.40pm @flyairnz flight from ChCh to AKL. #eqnz	ChCh; AKL		
#eqnz just read on stuff that rescuers were heading back towards CTV building!!!	CTV / Canterbury Television		
Direct link to CNN interview #1 http:\/\/t.co\/PC6PO2Q. On air again at 8:30pm NZT. #eqnz			CNN
Mmmmm..wonder if this means I can't fly out of CHCH to Welly for the conference. I have a lot to say about the power of social media!!	CHCH	Welly	
If you know of a Chc biz that needs to get back online to function or survive		Chc	
New Zealand: Ban Ki-moon statement on #earthquake stresses readiness of #UN to contribute in any way needed: http:\/\/bit.ly\/g5jeUu #eqnz			UN

example, none of these were identified as streets or roads: "down to street," "end street," "due to road," "information about road". Even with these fixes, our errors in street identification are predominantly false positives. Some examples from the training set appear in Table 4.

The algorithm finds toponyms both using OpenCalais and the National Geospatial Intelligence Agency gazetteer. In the case of identifying toponyms and abbreviations, our errors are mostly omission errors (type 1), as shown in Tables 5 and 6. Toponym misses mostly are the result of references to toponyms (such as Hornby and Ferrymead in Table 5) that are not in the gazetteer. We added a local gazetteer to reduce the number of omissions.

Our approach to identifying abbreviations performed well, but in some cases was unable to distinguish which abbreviations were locations. We added a line of code so that an abbreviation identified more than three times as a place (by following a place preposition or proximal to a cardinal direction or distance word) is added to a match list of place abbreviations. Examples of errors from our training set appear in Table 6.

Our approach to disambiguating the abbreviations was flawed by the fact that the correct disambiguation result might be present, without invariably listing first. To bring the best disambiguation word or phrase to the top, therefore, we introduced part of speech tagging, where disambiguation phrases with verbs were excluded,[13] and disambiguation phrases were preferred that included geographical regions (gazetteer matches) or geographical features (hill, mount, heights, etc.).

7 Evaluation experiments

7.1 Creating the gold standard for tweet geo-tags

A manually-created gold standard for locations that appear in the Christchurch, New Zealand, and Austin, Texas, USA tweet data sets is used to score the algorithm. This section describes how we created the gold standard, and how we demonstrated coding reliability.

Each of two participants was given the same set of tweets and a blank spreadsheet with columns for street, building, toponym, and abbreviation. Each participant was given also the same instructions as to what constitutes a location in a tweet (as arrived at by preliminary testing), and examples of what to include in each category. They completed their location coding independently. (Tweets without locations have no location codes.) Then their codes were assembled into a spreadsheet for tweet-by-tweet comparison.

An independent adjudicator determined the location(s) when the two participants did not assign the same location(s) to a tweet. The adjudicator thus considered more carefully the tweets in which locations were found than those in which locations were not found, and this might have been the source of some omission error. The adjudicator decided discrepancies between the two participants' codes based upon the instructions (also given the participant coders) that defined a location in a tweet. Many of the discrepancies, however, seem to be that one coder had simply overlooked a location that the other coder had noticed. The adjudicated list of locations in the tweets is referred to as the gold standard.

[13] Part of speech tagger for Twitter by Noah Smith et al., is at http://www.ark.cs.cmu.edu/TweetNLP/

Table 7 Evaluation of per-tweet intra-coder consistency for the adjudicated annotations with a 6-week interval between the coding sessions (*N*=500)

Street identification (S)	100 %
Building identification (B)	99.9 %
Toponym identification (T)	97 %
Abbreviation identification (A)	98 %

7.2 Demonstrating the gold standard's reliability with intra-coder agreement

We wish to show that the adjudicated annotations are consistent among themselves within a tweet set, measuring *intra*- rather than *inter*-coder agreement. The same adjudicator worked with both tweet sets. Thus, reliability with one set suggests reliability among both sets.

We tested reliability by asking the adjudicator to annotate the same 500-tweet subset of the participants' coded data at two different times, with a 6-week interval between. The interval between the coding sessions was long enough that the tweet codes would not remain in memory.

We allowed partial agreement to count favorably in considering whether the adjudicated codes from the two sessions were actually in agreement. We count 60 % similarity or higher as agreement, as is found in larger decision-making where a majority rather than a unanimity is required.[14] Note that in comparing the codes for these two sessions, most of the codes did match exactly, and in some instances, the codes matched, but their categorization (as toponym vs. building, for example) did not. These were included within the 60 % partial agreement in that what the adjudications are measuring ultimately are not categories but the coded locations themselves. The algorithm does not output location in categories at all—the categories are introduced only to compare relative accuracy among the different types of location-mining.

The percentage accuracy between the two sessions appears in Table 7.

Percentage accuracy is a fairly weak measure of reliability in that it does not account for agreement owing to chance in this particular data set. Nor does percentage accuracy account for the uneven distribution of location codes across the street, building, toponym and abbreviation categories.

We used percentage accuracy because the kappa statistic often used to measure rater agreement is not recommended to measure reliability owing to the nature of the codes and how we score the codes, and owing to the nature of the data itself. The nature of the codes is such that there are very many different location codes in especially the toponym category which might affect the calculation of kappa. Per-tweet scoring of location codes allows a partial match (with 60 % agreement) to be counted as a match if it does not fit kappa rigor. In addition, the many more tweets without location than with location, and hence without codes, would reduce the value of kappa, in an effect known as "prevalence".[15]

The number of people performing the coding independently, the adherence to a formerly arrived at definition of location in a tweet, along with the demonstration of reliability of intra-coder adjudications demonstrate the gold standard's reliability.

[14] "Consensus decision-making" in Wikipedia, Retrieved July 24, 2012, from http://en.wikipedia.org/wiki/Consensus_decision-making

[15] Kilem Gwet (2002). Kappa statistic is not satisfactory for assessing the extent of agreement between raters. Retrieved July 15, 2012 from http://www.agreestat.com/research_papers/kappa_statistic_is_not_satisfactory.pdf; Julius Sim, Chris Wright (2005). The kappa statistic in reliability studies: Use, interpretation and sample size requirements. Phys, Ther. 85(3):257–68.

7.3 Scoring the algorithm output

Precision and recall Information retrieval can be scored by match with an accepted standard using precision and recall statistics, and their combination into the F measure. Definitions are:

$\text{Precision} = \frac{tp}{tp+fp}$ tp = true positive, fp = false positive
$\text{Recall} = \frac{tp}{tp+fn}$ tp = true positive, fn = false negative

$$F = \frac{2\ PR}{P + R}$$

Location categories for scoring The algorithm output is tweet + location(s). However, we divided the manual output results into separate categories for streets, buildings, toponyms, and abbreviations so that we could score each part of the algorithm separately. After which, the four separate parts are averaged in a combined score.

Difficulties in scoring by category are that some names are found in multiple location categories. For example, Stone Oak simultaneously names a street (Stoneoak Drive in Texas), a building (the Stone Oak Ranch apartments in Austin, Texas), and a toponym (a neighborhood in Round Rock, Texas). The statistics must be viewed with this in mind.

7.4 Results of our geo-parsing algorithm on unseen data

The algorithm attained the results in Table 8 on the New Zealand test set of tweets. The algorithm attained an F of .85 for streets, an F of .86 for buildings, an F of .96 for toponyms, and an F of .92 for abbreviations. We could take statistics separately for the spell correction portion of the algorithm. We explained above that the spell checker, fortified with a gazetteer, identifies words and place names it believes to be misspelled, as well as words that are clipped or multiple words together squashed without spaces as are sometimes found in Twitter. If we count words like "christch" and "christc" as misspellings (rather than abbreviations), we have a recall of 0.935. The spell check algorithm identified "Iadho" and corrected to "Idaho" one of the location words in the tweets, bringing its precision to 1. The spell check algorithm therefore performed at an F of 0.966 for the New Zealand tweet set.

7.4.1 Comparison of our algorithm to Yahoo Placemaker

Yahoo Placemaker is a geo-parsing service that tags location words in free text. We ran the same tweets through Placemaker to compare to our algorithm. Results appear in Table 9.

We are unable to score the precision (and therefore also the F measure) for Placemaker in identifying abbreviations because of the way that algorithm works. Placemaker does not output location abbreviations; instead, it outputs the toponym that correspond to those abbreviations. We can use the manual annotations to measure true positives ("place abbreviations found that are actually place abbreviations") and false negatives ("place abbreviations that should have been found but were not") that

Table 8 Tweets from the New Zealand testing set (N=2000) evaluated against manually geo-tagged data with respect to Recall, Precision, and F Measure

	Recall	Precision	F Measure
Street identification (S)	0.85	0.85	0.85
Building identification (B)	0.97	0.78	0.86
Toponym identification (T)	0.94	0.99	0.96
Abbreviation Identification (A)	0.95	0.90	0.92
Combined (BSTA)	0.93	0.88	0.90

Note that our abbreviation identification score would be even higher, save for a bug in our program (which we have already corrected in the next version) that counts cut-off words at the end of tweets as abbreviations. So for example, "ther" in the example following was incorrectly identified as an abbreviation

```
RT @nzff: RT @rialtocinemas: Does anyone know of someone that was at
Rialto Cinemas in CHCH? Still haven't heard from our team down ther ...
```

we require for recall. But we cannot determine the false positives ("abbreviations found that are not abbreviations for place") that we require for measuring precision because these are not Placemaker output.

7.5 Our algorithm with another data set to show generalizability

We wish to show that our algorithm is effective when applied to another set of crisis tweets. We had hand-annotated a set of tweets from a 2011 fire in Austin, Texas, using the same annotation method as for the New Zealand set. The adjudicated annotations were used for scoring the algorithm. Results of the algorithm are in Table 10. The combined F for streets, buildings, toponyms and abbreviations and acronyms was 0.71. Our recall is low with respect to streets because our algorithm does not find Texas highways in the tweets, and such highways were absent in the training set we used to create the street-identification of the algorithm. Our recall is low for buildings because our heuristics are largely semantic and do not rely on syntax. We are correcting this difficulty in the subsequent version of the algorithm.

7.6 Results of the abbreviation disambiguation algorithm on unseen data

The use of precision in word sense disambiguation (WSD) systems is often scored using partial credit. We introduced partial credit of 0.5, between 1 (correct) and 0 (incorrect), in

Table 9 Yahoo Placemaker results on the same New Zealand data set (N=2000) that we used to evaluate our algorithm

	Recall	Precision	F Measure
Street identification (S)	0	0	0
Building identification (B)	0	0	0
Toponym identification (T)	0.82	0.56	0.67
Abbreviation Identification (A)	0.28	Not applicable	Not applicable

Table 10 Austin fire tweets from 2011 (N=3331) demonstrates algorithm generalizability (combined BSTA of 0.71)

Evaluation with Austin fire tweets			
	Recall	Precision	F Measure
Street identification (S)	0.46	0.93	0.62
Building identification (B)	0.49	0.99	0.66
Toponym identification (T)	0.87	0.88	0.87
Abbreviation Identification (A)	0.68	0.71	0.69

scoring for abbreviation disambiguation. We would give partial credit, for example, when the correct disambiguation word appears within the 5-word phrase presented as the result.

Our score for abbreviation disambiguation is low because the correct disambiguation word or phrase is not found in most tweets as a candidate to match to the abbreviation. Our results for the New Zealand tweet set (N=2000) were 0.51 recall, and 0.49 precision, which gives an F of 0.50. We have removed this part of the routine in the subsequent version of the algorithm.

8 Discussion

8.1 Improving identification of places

Errors of omission (type 1) Our heuristics in the building/street/toponym sub-routines do not use semantics to the extent that they could. Therefore, certain words that have non-geographic alternate meanings such as “park” or “square” must be omitted so that we do not find false positives. The result is that we do not find tweets that include references to such places.

Perhaps most important for the goal of the algorithm, we miss local places that are not found in the National Geospatial Agency gazetteer. We tried to correct this by including a more specific gazetteer from the domain region,[16] but this introduced a great many false positives, many of which were caught by our filter lists. Our next version of the algorithm uses a much more comprehensive gazetteer which will miss fewer local toponyms.

Errors of commission (type 2) In one intended use of this algorithm for finding tweets with location information that are relevant to a crisis, errors of omission are more serious than errors of commission. This is because finding commission errors would entail finding places incorrectly output as errors by the algorithm, whereas finding omission errors would require returning to the tweets themselves to look for locations that were missed.

Errors of redundancy We use both OpenCalais and an excerpt from the National Geospatial-Intelligence Agency gazetteer to find toponyms. Occasionally, the same toponym is found twice. But because we also collect locations on different hierarchical levels, such as Lyttelton Port, and Lyttelton, we cannot remove such repetitions.

[16] Official New Zealand gazetteer of place names, at http://www.linz.govt.nz/placenames/find-names/nz-gazetteer-official-names as of January 31, 2012.

Redundancy errors may be corrected at the application level with the heuristic that if a tweet is mapped to a particular location, it should not map the same tweet to the same location more than once.

8.2 Extent to which our work will generalize

Our algorithm performed better on the testing set of Christchurch crisis tweets that resembled the training set (combined street, building, toponym, abbreviation F=0.90) than on the crisis tweets from Austin, Texas (combined street, building, toponyn, abbreviation F=0.71). We can improve results in the next generation of our algorithm by using techniques that rely more on syntax and machine learning than on lexico-semantic pattern recognition.

We expect that our work will be greatly useful to those mining microtext. The Text Retrieval Conference (TREC) added a microblog track in 2011, complete with a large database of tweets that may be downloaded for research purposes. Studies of Named Entity Recognition (NER) particular to Twitter are becoming more commonplace, and improved location recognition will help. We offer our hand-annotated tweets to other researchers for continued study.[17]

9 Future work

We welcome others to propel this research by refining our work or going beyond. We plan to follow many of the research directions here.

9.1 User input

Present experiments use a pre-collected data set. A serviceable program, however, will ask the user to specify search parameters in the actual Twitter stream. For example, the user might input a city of interest, and time period and keyword or phrase. Tweets conforming to those parameters would then be geo-parsed.

9.2 Widening methods to geo-locate tweets

Many tweets have no indication of place. In the 3331-tweet subset of our Austin data set for which we have full metadata, 39.2 % of tweets have no location in tweet text, in user-registered location field or in GPS coordinates. We are beginning initial tests to use the social network to geo-locate messages.

9.3 Results display/visualization

Geo-coding The ultimate goal of this work is to place tweets on a map. The assignment of geographic coordinates is called geo-coding. We are experimenting with methods to assign the correct coordinates to a named place based on gazetteer lookup. How this will be done for streets and buildings requires further investigation.

[17] Write to gelern@cs.cmu.edu for use of the geo-tagged 2011 earthquake tweets from Christchurch, New Zealand, or the geo-tagged 2011 fire tweets from Austin, Texas.

Mapping to show relevance and uncertainty We will be able to associate a location with a tweet to a degree of certainty. Moreover, some tweets have more than one location. Our map should reflect uncertainty in the tweet location, and it would be useful to limit the map to tweets that are relevant.

9.4 Run in close to real time

We intend to modify the algorithm so that it is able to process tweets in close to real time. Initial results have been encouraging, as we have been able to reduce processing time from minutes to seconds. Scaling up has required alterations to the two-module design, external resource management, and data load balancing.[18]

9.5 Generalizability

Our approach could be adjusted to identify locations in tweets of other languages. Even so, differences in naming of streets and addresses among cultures must be respected. Many streets in Japan are not named, for example, and Japanese addresses might be written from largest to smallest geographical entity as opposed to the way addresses are written in the West.

Acknowledgements We are grateful for the support of the Director of Language Technologies Institute, Jaime Carbonell. Our data was collected from an archive maintained by Brendan O'Connor at Carnegie Mellon University. Nikolai Mushegian, Corinne Meloni, Josh Swanson, Niharika Ray, Andrew Minton, Marielle Saums, and Christa Hester were among the tweet annotators who helped us arrive at a ground truth for scoring the results of our algorithm. Our open-source resources were supplemented by data from Abbreviations.com and from the online Twitter dictionary, Twittonary. Finally, we appreciate our discussions with doctoral candidate in statistics Cong Lu regarding intra-coder reliability.

Appendix 1 Geoparsing for location abbreviations

Preparation steps

External resource preparation

Download and make available for processing:

- Twittonary
- Dictionary (supplemented with standard technical abbreviations such as tv and iphone)
- Excerpt from National Geospatial-Intelligence Agency gazetteer for the domain (here, for New Zealand)
- List of geographical features

[18] We reported results of testing the second version of the algorithm at the high performance computing (XSEDE'12) conference in Chicago, Illinois, USA, this July 2012.

- List of building terms
- List of prepositions
- Location indicators for direction: (N S E W, NE, NW, north, northeast, northwest, etc.)
- Location indicators for distance: (mi, mile, km, kilometer, etc.)

Data preparation

1. Normalize tweet: tokenize (word per word), remove articles (a, an, the), and remove punctuation from beginning and end of word and at the end of a sentence.
2. Part of speech processing for tweets using third-party, open source software[19]
3. Find date and time of tweet creation

Processing steps

Identify place abbreviations and acronyms using heuristics that specify how to match external resources to data. Consider a text word to be a place abbreviation if it:

* has between 2 and 6 characters and is not in Dictionary or Twittonary is preceded by preposition or direction or distance term
* matches to a confirmed place abbreviation (that is, an abbreviation that matches the heuristic above
* matches with an abbreviation known to be a place, either from location abbreviations in Abbreviations.com or from an earlier pass over the tweets

But skip as an abbreviation if it is preceded by # or @

Identify candidate disambiguation words

Data for candidate disambiguation words and phrases are drawn from tweets posted 1 to 5 days before the date of the tweet with the corresponding abbreviation. The disambiguation words and phrases are selected according to the following heuristics:

- first word begins with the same letter as the abbreviation
- if an abbreviation has *n* letters, take long multi-words of length *n-2* words, *n-1* words, and *n* words, and *n* with the addition of up to 2 stop words
- no verbs
- no hashtags or @mentions
- no colon, semi-colon, question mark or exclamation point within the phrase

[19] http://www.ark.cs.cmu.edu/TweetNLP/

Attributes of abbreviations and acronyms and their expansions

Based on inspection of the data, we have arrived at the following attributes that characterize a match between location abbreviation or acronym, and candidate expansion word or phrase. We defined these attributes as:

FIRST LETTER MATCH (i.e. first letter of abbreviation and disambiguation match)
SECOND LETTER WORD MATCH
SECOND LETTER WORD MATCH, NO STOPWORDS
SECOND LETTER MATCH
THIRD LETTER WORD MATCH
THIRD LETTER WORD MATCH, NO STOPWORDS (i.e., DRC ◊ Democratic Republic of Congo)
THIRD LETTER MATCH
INDEX OF FIRST LETTER (i.e., value that indicates where first letter in the short form occurs in the long form. If that letter in the short form does not occur in the long form, it gets a "1" index value)
INDEX OF SECOND LETTER
INDEX OF THIRD LETTER
LAST LETTER MATCH
FIRST LETTER CORRESP (i.e., all the letters in the abbreviations correspond to the first letters in each word in the long form. Example pgc -> pyne gould corporation)
NUM VOWELS IN SHORT (i.e., number of vowels in the short form)
NUM VOWELS IN LONG (i.e., number of vowels in the long form)
RATIO NUM VOWELS IN SHORT /LONG (for example, Toba → Manitoba, where Toba has 2 vowels and Manitoba has 4. The ratio would be 2/4 = .5)
SAME ORDER OF LETTERS
SAME NUMBER LETTERS AS WORDS (Most abbreviations expand to one word rather than many, while acronyms sometimes have the same number of letters as words.)
SAME NUMBER LETTERS AS WORDS, NO STOPWORDS
NUMBER OF STOPWORDS
HAS NON-LETTER CHARACTER (for example, p&g for Proctor & Gamble)
FULL SHORT IN LONG
SHORT AT BEGINNING (Entire short word is included verbatim at the beginning of the word (Pitt → Pittsburgh; Calif → California)
NUMBER WORDS IN LONG
NUMBER CHARS IN LONG (Pittsburgh has 10 characters)
NUMBER CHARS IN SHORT
RATIO OF NUMBER CHARS SHORT / NUMBER CHARS LONG
RATIO OF NUMBER CHARS SHORT / NUMBER WORDS LONG
CONTAINS LOC (long word contains geographical features, buildings or toponyms)

Create a decision tree model based on attributes

Decision tree algorithms classify unseen data based on data they have previously learned. Given a set of attributes and how these attributes match to the correct classes (and mis-match to other classes), the algorithm will discover how unseen data are classified based on their attributes. The algorithm maximizes the information gain at each decision tree branch test, and makes a model based on the training data. Tests are decided upon during the training phase on the basis of entropy, or the measure of disorder of the data. The model can be considered as a branching set of decisions, or tests. Each test decision branches into a subtree until it reaches a leaf end node. Unseen data is sent through the tree by undergoing a series of binary tests until it reaches a leaf.

Weka's J48 is a version of an earlier algorithm developed by J. Ross Quinlan, the popular C4.5. We select the J48 option to "prune", or simplify results. Pruning operates on the decision tree while it is being induced. It works by compressing a parent node and child nodes into a single node whenever a split is made that yields a child leaf that represents less than a minimum number of examples from the data set. Pruning can be used as a tool to correct for potential overfitting, and so an unpruned tree might perform slightly better than a pruned one [11].

Our tree (Appendix 4) has many attributes, and therefore many nodes. The root node must effectively split the data, and the best split is the one that provides the most information gain.

Each split attempts to pare down a set of instances (the actual data) until all have the same classification. Each node is tested to determine whether it has a particular value based on which attributes are represented. The data is then routed accordingly.

Given our training data comprised of abbreviations and acronyms and their correct and incorrect disambiguation words and phrases, the machine learning algorithm uses statistics to assign weights to the different features according to their importance. Our decision tree model with 52 leaves appears in Appendix 4.

Because our model is based on our training data as well as these attributes, another model made with different data would obtain different results. We received 87.9 % accuracy with this model, with a corresponding kappa of 0.748, indicating that the model performs much better than chance (which would produce a Kappa of zero).

Features that are significant in correctly classifying the abbreviation or acronym with its disambiguation expression are shown by their location near the root, and also their recurrence in the tree (possibly with different values in different branches). These are:

`NumCharsLong`, number of chraracters in the disambiguation work/phrase
`NumCharsShort`, number of characters in the abbreviation/acronym
`IndexOfSecond`, value that indicates in what position the second letter of abbreviation occurs in the disambiguation word/phrase
`ThirdLetterLetterMatch`, third letter of the abbreviation matches third letter of disambiguation word
`ContainsLoc`, whether the disambiguation word/phrase contains a word in our gazetteer, building lexicon or direction word list
`SecondLetterWordMatch`, second letter of abbreviation matches second letter in disambiguation phrase
`SameNumbLettersAsWords`, same number of letters in the acronym as words in the disambiguation phrase
`SecondLetterMatch`, second letter of the abbreviation matches second letter of disambiguation word

Rank abbreviation – expansion pairs

Run each abbreviation along with its potential match file through the decision tree. Weka outputs ranking along with true or false and a corresponding error prediction (1 indicates 100 % confidence that the abbreviation corresponds to the match expansion). We wrote a short script that sorts the true values according to error level. We take the top five disambiguation phrases for the sake of error analysis to see whether the correct disambiguation appears near the top, even if it was not selected first.

Appendix 2 Geoparsing streets, buildings and toponyms

User Input (theoretically; data was pre-collected for this study)

- city and county of the data set, possibly also nearby countries
- that country's abbreviation and the abbreviation of nearby countries

Ex. For our data set on the Christchurch, New Zealand earthquake, the user enters:

Christchurch (Chch)
New Zealand (NZ), and
Australia (Aus)

External resources:

- Gazetteer excerpt for region of inquiry
- List of common words that are also place names to filter the gazetteer (list was manually generated)

- Enhanced building list (http://en.wikipedia.org/wiki/list_of_building_types) minus a few ambiguous words such as "wall" and "place"

Data (tweet) preparation

1. Save the original tweets
2. Make a copy of the tweets for NLP preparation.
3. Remove hashtag that was used to retrieve the data set (and recurs repeatedly throughout) in copy of tweets, and tokenize.
4. Remove @mentions and replace by XXX. (This is so that we can preserve the original word count).
5. Remove tweets in the copy set of tweets that contain unicode characters
6. Run the copy of tweets through spell-check algorithm. Retain words that are mis-spelled as well as those that are corrected, in case the spell check alters what is actually correct.

Data (tweet) processing

1. Run original tweets through OpenCalais
    a. Retain locations (city, country, continent, etc.)
    b. Retain natural features (mountain, etc.)
    c. Retain facilities
    d. Ignore all other entities found by OpenCalais
2. Find toponyms
    a. Make all data lower case
    b. Match against our own gazetteer
    c. Do not include partial matches (Example: do not take Eiffel if it needs Tower)
    d. Do not include toponyms in @mentions
    e. Allow matches when a space is missing between the two words (newzealand)
3. Find buildings
    a. Look at each word in the tweet individually (tokenize)
    b. Match against building list to find additional facilities
    c. If building word is found, take two words before and capture the string as output. Also, hard code pairs such as "X and Y buildings", "X or Y buildings" and multi-word building names such as "W X and Y Z buildings" [20]
    d. To filter non-specific buildings, we do not take those preceded by the article "a", possessive pronouns "I, my, mine, our, his, hers, yours, theirs", or relative pronouns "which, what" or demonstrative pronouns "this, that"
    e. Do not identify as a building if words are found across punctuation mark of period, comma, semi-colon, brackets, parentheses
    f. Do not identify as a building word, even if it matches an entity named on the buildings list, if it is the first word of a tweet.
    g. Do not identify as a building if the building phrase contains a placeholder "XXX"
4. Find streets within the tweets
    a. Street identification words: st, street, ln, lane, dr, drive, boulevard, blvd, road, rd, avenue, ave, pl, way, wy
    b. Check for an Arabic numeral two to three spaces before the street identification word. If a number is found, mine everything from the number to the street

[20] These have been replaced in the next version of the algorithm that will be presented at the XSEDE'12 conference in July 2012

identification word. If there is no number, only take the word immediately before the street identification word. Take 3 words ahead of the street word (including a number) plus the street indicator

c. To filter possibility of "st" meaning saint instead of street, we use a list of saints' names.[21] Matches with the list indicate that the phrase is not a street. Future versions of the algorithm should rely also on word order.

d. Do not identify as a street if street phrase is found across punctuation: period, comma, semi-colon, brackets, parentheses

Example "14 East. Street that is a dead end"

Do not mine this as 14 East Street

5. Output: tweet—location

Appendix 3 Samples of training data for the abbreviation and acronym classifier

Actual abbreviations and acronyms for location

Park	Park Avenue
Lex	Lexington Ave.
Sau Ar	Saudi Arabia
Zimb	Zimbabwe
Pac	Pacific
Papua	Papua New Guinea
N.A.	North America
Dom Rep	Dominican Republic
Mainz	Mainz am Rhein
SG	St. Gaullen
SG	St. Gaul
Qnborough	Queenborough
Pborough	Peterborough
L.I. sound	Long Island Sound
Hunt	Huntington
N Bay Shore	North Bay Shore
Jersey Shore	New Jersey Shore
Ronk	Ronkonkoma
Rio	Rio de Janeiro
Kab	Kabambare
Kago	Kagoshima
T&T	Trinidad and Tobago
TT	Trinidad and Tobago
U.A.E.	United Arab Emirates
W. Sam	Western Samoa
Sol. Is.	Solomon Islands
S.L.	Sierra Leone
PNG	Papua New Guinea
SOS	Southend on Sea
EXM	Exmouth Gulf Airport
H.H.	Head of the Harbor
MTA	Metropolitan Transport Authority

[21] List of Saints' Names: http://www.catholic.org/saints/stindex.php

FI	Fire Island
HH	Hoek van Holland
Xmas Island	Christmas Island
Congo	Democratic Republic of Congo
D.R.C.	Democratic republic of congo
TdF	Tierra del Fuego
SP	Sao Paulo

False Abbreviations & Acronyms

Park	Parking lot
Lex	Last expression
Sau Ar	Sad argument
Zimb	Zoo in my basement
Pac	Plenty in cabinet
N.A.	not available
SG	Southern Georgia
SG	so geography
SG	several grains
L.I. sound	lighter sound
Hunt	Hunting
N Bay Shore	not by the shore
Ronk	Rings on new keys
Ronk	Rubber on knives
Ronk	Recalling our kids
Rio	Ring in an oval
Rio	Rinse in oil
Kab	kneel and bend
Kab	knots and bends
Kago	kick and go under
T&T	Trains and transportation
TT	tractor trailer
U.A.E.	Under All Empires
U.A.E.	Under application employees
Sol. Is.	Sole Ice
S.L.	southern languages
PNG	please not again
SOS	signs of success
EXM	expected money
FI	Finally I
HH	Hello harry
D.R.C.	Daily Ritual Cleaning
DRC	Dr. Classic
DRC	Drab Rubber Chicken
TdF	to do Friday
TdF	Trumpet for December Festival
SP	sudden park
SP	spark
SP	salt and pepper
SP	sensational paper

Appendix 4 Classifier model for abbreviations and acronyms based on the full training set

```
406 instances
28 attributes
J48 pruned tree
------------------

thirdLetterWordMatchNoSW = TRUE
|  numCharsLong <= 18
|  |  shortLongCCRatio <= 0.214286: TRUE (11.0/3.0)
|  |  shortLongCCRatio > 0.214286: FALSE (3.0)
|  numCharsLong > 18: TRUE (24.0)
thirdLetterWordMatchNoSW = FALSE
|  thirdLetterWordMatch = TRUE: TRUE (5.0)
|  thirdLetterWordMatch = FALSE
|  |  firstLetterMatch = TRUE
|  |  |  shortAtBeginning = TRUE
|  |  |  |  numCharsShort <= 5
|  |  |  |  |  indexOfSecond <= 0
|  |  |  |  |  |  containsLoc = TRUE: TRUE (2.0)
|  |  |  |  |  |  containsLoc = FALSE: FALSE (12.0/1.0)
|  |  |  |  |  indexOfSecond > 0
|  |  |  |  |  |  firstLetterCorresp = TRUE
|  |  |  |  |  |  |  secondLetterWordMatch = TRUE: TRUE (31.0/5.0)
|  |  |  |  |  |  |  secondLetterWordMatch = FALSE
|  |  |  |  |  |  |  |  sameNumLettersAsWords = TRUE
|  |  |  |  |  |  |  |  |  secondLetterLetterMatch = TRUE: FALSE (5.0)
|  |  |  |  |  |  |  |  |  secondLetterLetterMatch = FALSE
|  |  |  |  |  |  |  |  |  |  containsLoc = TRUE: TRUE (2.0)
|  |  |  |  |  |  |  |  |  |  containsLoc = FALSE: FALSE (6.0/1.0)
|  |  |  |  |  |  |  |  sameNumLettersAsWords = FALSE
|  |  |  |  |  |  |  |  |  shortInLong = TRUE: TRUE (72.0/15.0)
|  |  |  |  |  |  |  |  |  shortInLong = FALSE
|  |  |  |  |  |  |  |  |  |  secondLetterLetterMatch = TRUE: FALSE (14.0/4.0)
|  |  |  |  |  |  |  |  |  |  secondLetterLetterMatch = FALSE
|  |  |  |  |  |  |  |  |  |  |  numVowelsShort <= 0
|  |  |  |  |  |  |  |  |  |  |  |  indexOfSecond <= 3: TRUE (12.0/3.0)
|  |  |  |  |  |  |  |  |  |  |  |  indexOfSecond > 3
|  |  |  |  |  |  |  |  |  |  |  |  |  indexOfSecond <= 5: FALSE (5.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  indexOfSecond > 5
|  |  |  |  |  |  |  |  |  |  |  |  |  |  numVowelsLong <= 2: TRUE (2.0)
|  |  |  |  |  |  |  |  |  |  |  |  |  |  numVowelsLong > 2: FALSE (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  |  numVowelsShort > 0: TRUE (16.0/2.0)
|  |  |  |  |  |  firstLetterCorresp = FALSE
|  |  |  |  |  |  |  containsLoc = TRUE
|  |  |  |  |  |  |  |  numCharsLong <= 13: FALSE (3.0/1.0)
|  |  |  |  |  |  |  |  numCharsLong > 13: TRUE (9.0/2.0)
|  |  |  |  |  |  |  containsLoc = FALSE
|  |  |  |  |  |  |  |  shortLongCWRatio <= 2
|  |  |  |  |  |  |  |  |  ratioNumVowelsSL <= 0.125
|  |  |  |  |  |  |  |  |  |  secondLetterLetterMatch = TRUE: TRUE (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  secondLetterLetterMatch = FALSE
|  |  |  |  |  |  |  |  |  |  |  numStopwords <= 0: FALSE (2.0)
|  |  |  |  |  |  |  |  |  |  |  numStopwords > 0: TRUE (2.0)
|  |  |  |  |  |  |  |  |  ratioNumVowelsSL > 0.125: FALSE (12.0)
|  |  |  |  |  |  |  |  shortLongCWRatio > 2: TRUE (2.0)
```

```
|  |  |  |  numCharsShort > 5: TRUE (13.0)
|  |  |  shortAtBeginning = FALSE
|  |  |  |  numVowelsShort <= 2
|  |  |  |  |  secondLetterWordMatch = TRUE
|  |  |  |  |  |  numWordsLong <= 3
|  |  |  |  |  |  |  containsLoc = TRUE: TRUE (6.0)
|  |  |  |  |  |  |  containsLoc = FALSE
|  |  |  |  |  |  |  |  numCharsShort <= 5
|  |  |  |  |  |  |  |  |  indexOfSecond <= 8
|  |  |  |  |  |  |  |  |  |  indexOfThird <= 6: FALSE (5.0)
|  |  |  |  |  |  |  |  |  |  indexOfThird > 6: TRUE (3.0/1.0)
|  |  |  |  |  |  |  |  |  indexOfSecond > 8: TRUE (2.0)
|  |  |  |  |  |  |  |  numCharsShort > 5: TRUE (2.0)
|  |  |  |  |  |  numWordsLong > 3: FALSE (3.0)
|  |  |  |  |  secondLetterWordMatch = FALSE
|  |  |  |  |  |  thirdLetterLetterMatch = TRUE
|  |  |  |  |  |  |  sameOrder = TRUE
|  |  |  |  |  |  |  |  lastLetterMatch = TRUE: FALSE (2.0)
|  |  |  |  |  |  |  |  lastLetterMatch = FALSE
|  |  |  |  |  |  |  |  |  containsLoc = TRUE: FALSE (5.0/1.0)
|  |  |  |  |  |  |  |  |  containsLoc = FALSE
|  |  |  |  |  |  |  |  |  |  numWordsLong <= 2
|  |  |  |  |  |  |  |  |  |  |  numVowelsLong <= 5: TRUE (2.0)
|  |  |  |  |  |  |  |  |  |  |  numVowelsLong > 5: FALSE (2.0)
|  |  |  |  |  |  |  |  |  |  numWordsLong > 2: TRUE (3.0)
|  |  |  |  |  |  |  sameOrder = FALSE: TRUE (2.0)
|  |  |  |  |  |  thirdLetterLetterMatch = FALSE
|  |  |  |  |  |  |  numVowelsShort <= 1
|  |  |  |  |  |  |  |  numStopwords <= 0
|  |  |  |  |  |  |  |  |  containsLoc = TRUE
|  |  |  |  |  |  |  |  |  |  shortLongCWRatio <= 2.5: FALSE (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  shortLongCWRatio > 2.5: TRUE (7.0/1.0)
|  |  |  |  |  |  |  |  |  containsLoc = FALSE
|  |  |  |  |  |  |  |  |  |  indexOfSecond <= 2: TRUE (12.0/1.0)
|  |  |  |  |  |  |  |  |  |  indexOfSecond > 2
|  |  |  |  |  |  |  |  |  |  |  numCharsLong <= 19: FALSE (11.0/2.0)
|  |  |  |  |  |  |  |  |  |  |  numCharsLong > 19: TRUE (2.0)
|  |  |  |  |  |  |  |  numStopwords > 0: FALSE (3.0)
|  |  |  |  |  |  |  numVowelsShort > 1: FALSE (15.0/1.0)
|  |  |  |  numVowelsShort > 2: FALSE (8.0)
|  |  firstLetterMatch = FALSE
|  |  |  numWordsLong <= 1: FALSE (14.0)
|  |  |  numWordsLong > 1
|  |  |  |  sameNumLettersAsWords = TRUE: FALSE (5.0/1.0)
|  |  |  sameNumLettersAsWords = FALSE
|  |  |  |  numVowelsShort <= 0: TRUE (3.0)
|  |  |  |  numVowelsShort > 0
|  |  |  |  |  shortLongCWRatio <= 2.5
|  |  |  |  |  |  shortInLong = TRUE
|  |  |  |  |  |  |  lastLetterMatch = TRUE: TRUE (3.0)
|  |  |  |  |  |  |  lastLetterMatch = FALSE: FALSE (6.0/1.0)
|  |  |  |  |  |  shortInLong = FALSE: FALSE (3.0)
|  |  |  |  |  shortLongCWRatio > 2.5: TRUE (3.0)

Number of Leaves  : 52
```

References

1. Adriani M, Paramita ML (2007) Identifying location in Indonesian documents for geographic information retrieval. GIR'07, November 9, 2007, Lisbon, Portugal, pp 19–23
2. Ammar W, Darwish K, El Kahki, A, Hafez, K (2011) ICE-TEA: in-context expansion and translation of English abbreviations. In Gelbukh A (ed) CICLing 2011, Part II, LNCS 6609, pp 41–54
3. Cheng Z, Caverlee J, Lee K (2010) You are where you tweet: a content-based approach to geo-locating Twitter users. CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada, pp 759–768
4. Dannélls D (2006) Automatic acronym recognition. Eleventh Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006), April 3–7, Trento, Italy, pp 167–170
5. Eisenstein J, O'Connor B, Smith NA, Xing E (2010) A latent variable model for geographic lexical variation. In Proceedings of EMNLP, pp 1277–1287
6. Gelernter J, Mushegian N (2011) Geo-parsing messages from microtext. Transactions in GIS 15(6):753–773
7. Hecht B, Hong L, Suh B, Chi EH (2011) Tweets from Justin Bieber's Heart: the dynamics of the "location" field in user profiles, CHI 2011, May 7–12, 2011, Vancouver, BC, Canada, pp 237–246
8. Hill E, Fry ZP, Boyd H, Sridhara G, Novikova Y, Pollock L, Vijay-Shanker K (2008) AMAP: automatically mining abbreviation expansions in programs to enhance software maintenance tools. MSR,'08, May 10–11, 2008, Leipzig, Germany, pp 79–88
9. Ireson N, Cirabegna F (2008) Toponym resolution in social media. PF Patel-Schneider et al. (eds.) ISWC 2010, Part I, LNCS 6496, pp 370–385
10. Jung JJ (2011) Towards named entity recognition method for microtexts in online social networks: a case study of Twitter. 2011 International Conference on Advances in Social Network Analysis and Mining (ASONAM), pp 563–564
11. Khanal N, Kehoe A, Kumar A, MacDonald A, Mueller M, Plaisant C, Ruecker S, Sinclair S Monk Tutorial: Metadata offers new knowledge. Retrieved January 31, 2012 from http://gautam.lis.illinois.edu/monkmiddleware/public/analytics/decisiontree.html
12. Kinsella S, Murdock V, O'Hare N (2011) "I'm eating a sandwich in Glasgow": modelling locations with tweets. SMUC'11, October 28, 2011, Glasgow, Scotland, pp 61–68
13. Leveling J, Hartrumpf S (2008) On metonymy recognition for geographic IR. Int J Geogr Inf Sci 22(3), http://www.geo.uzh.ch/~rsp/gir06/papers/individual/leveling.pdf, accessed 12 January 2012
14. Lieberman MD, Samet H (2011) Multifaceted toponym recognition for streaming news. SIGIR'11. Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, July 2011, pp 843–852
15. Lieberman MD, Samet H, Sankaranarayanan J (2010) Geotagging with local lexicons to build indexes for textually-specified spatial data. IEEE 26th International Conference on Data Engineering (ICDE), pp 201–212
16. Liu J, Chen J, Liu T, Huang Y (2011) Expansion finding for given acronyms using conditional random fields. In: Wang H, et al. (eds) WAIM 2011, LNCS 6897, pp 191–200
17. Liu X, Zhang S, Wei F, Zhou M (2011) Recognizing named entities in tweets. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, Portland Oregon, June 19–24, pp 359–367
18. Liu Y, Piyawongwisal P, Handa S, Yu L, Xu Y, Samuel A (2011) Going beyond citizen data collection with mapster: a mobile+cloud real-time citizen science experiment. Seventh IEEE international conference on e-science workshops, pp 1–6
19. Marcus A, Bernstein MS, Badar O, Karger DR, Madden S, Miller RC (2011) Processing and visualizing the data in tweets. SIMOD Record 40(4):21–27
20. McInnes BT, Pedersen T, Liu Y, Pakhomov SV, Melton GB (2011) Using second-order vectors in a knowledge-based method for acronym disambiguation. Proceedings of the Fifteenth Conference on Computational Natural Language Learning, pp 145–153
21. Moschitti A, Chu-Carroll J, Patwardhan S, Fan J, Riccardi G (2011) Using syntactic and semantic structural kernels for classifying definition questions in jeopardy! Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK, July 27–31, 2011, pp 712–724
22. Nadeau D, Turney PD (2005) A supervised learning approach to acronym identification. In: Kégl B, Lapalme G (eds) AI 2005, LNAI 3501, pp 319–329
23. Okazaki M, Matsuo Y (2009) Semantic Twitter: analyzing tweets for real-time event notification. In: Breslin JG et al. (eds) BlogTalk 2008/2009, LNCS 6045. Proceedings of the 2008/2009 international conference on social software. Springer, Heidelberg, 2010 pp 63–74
24. Okazaki N, Ananiadou S (2006) A term recognition approach to acronym recognition. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pp 643–650

25. Okazaki N, Ananiadou S, Tsujii J (2008) A discriminative alignment model for abbreviation recognition. Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp 657–664
26. Paradesi S (2011) Geotagging tweets using their content. Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference, May 18–20, 2011, Florida, USA, pp 355–356
27. Park Y, Byrd RJ (2001) Hybrid text mining for finding abbreviations and their definitions. Association for Computational Linguistics http://aclweb.org/anthology/W/W01/W01-0516.pdf, Retrieved January 3, 2012
28. Pennell D, Liu Y (2011) Toward text message normalization: modeling abbreviation generation. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May, 2011, pp 5364–5367
29. Ponte J, Croft WB (1998) A language modeling approach to information retrieval. In Proceedings of SIGIR, pp 275–281
30. Ritter A, Clark S, Etzioni M, Etzioni O (2011) Named entity recognition in tweets: an experimental study. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, 1524–1534
31. Roche M, Prince V (2007) AcroDef: a quality measure for discriminating expansions of ambiguous acronyms. In: Kokinov B et al. (eds) Context 2007, LNAI *4635*, pp 441–424
32. Starbird K, Palen L, Hughes A, Vieweg S (2010) Chatter on the red: what hazards threat reveal about the social life of microblogged information. CSCW 2010, February 6–10, 2010, Savannah, Georgia, USA, pp 241–250
33. Taghva K, Vyas L (2011) Acronym expansion via Hidden Markov Models. 2*1st* International Conference on Systems Engineering, 16–18 August 2011, pp 120–125
34. Takahashi K, Pramudiono Il, Kitsuregawa M (2005) Geo-word centric association rule mining. Proceedings of the sixth international conference on Mobile Data Management (MDM) 2005, Ayia Napa, Cyprus, pp 273–280
35. Tanasescu V, Domingue J (2008) A differential notion of place for local search. LocWeb 2008, April 22, 2008, Beijing, China, pp 9–15
36. Vanopstal K, Desmet B, Hoste V (2010) Towards a learning approach for abbreviation detection and resolution. LREC 2010, May 19–21, 2010, Valletta, Malta, pp 1043–1049
37. Vieweg S, Hughes AL, Starbird K, Palen L (2010) Microblogging during two natural hazards events: what Twitter may contribute to situational awareness. In: Proceedings of the 2010 Annual Conference on Human Factors in Computing Systems (CHI 2010), Atlanta, Georgia: pp 1079–1088
38. Watanabe K, Ochi M, Okabe M, Onai R (2011) Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs. CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK, pp 2541–2544
39. Wing BP, Baldridge J (2011) Simple supervised document geolocation with geodesic grids. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, Portland, Oregon, June 19–24, 2011, pp 955–964

Judith Gelernter is a scientist in the Language Technologies Institute of Carnegie Mellon University. Her background is in geospatial technologies and natural language processing, which she has combined in projects such as geoparsing research (for *Transactions in GIS*, and the *International Journal of Applied Geospatial Research*, for example), and in her course that is an Introduction to Geoinformatics.

Shilpa Balaji is an undergraduate student at the School of Computer Science of Carnegie Mellon University. She is also pursuing minors in Language Technologies, offered by the Language Technologies Institute of CMU, and Business Administration, offered by the Tepper School of Business of CMU. Her academic and research interests include natural language processing and cognitive science. This is the first research project she has worked on, as well as the first paper she's been a part of.