Enable tweet-geolocation and don’t drive ERTs crazy! Improving situational awareness using Twitter

Juan Sixto, Oscar Peña, Bernhard Klein and Diego López-de-Ipiña

DeustoTech–Deusto Institute of Technology, Universidad de Deusto, Avenida de las Universidades 24, 48007, Bilbao (Spain)
{jsixto, oscar.pena, bernhard.klein, dipina}@deusto.es

Abstract

When traditional communication services are down during an emergency event, Twitter has proven to provide first-hand information to emergency response teams. The lack of geotagged tweets complicates these teams labour when trying to pin-point the events on a map. A rapid identification of situational awareness on incidents may help reduce the number of casualties and damages thanks to an efficient management. In this paper, we present a new approach to improve geolocation accuracy in Twitter posts, relying on NLP techniques and online geolocation APIs, providing the most trusted event location from a Twitter stream.

1 Antecedents

Geographic Information Retrieval (GIR) is a field of information retrieval which aims to extract location names in texts requiring some semantic data to be present in them [1]. The longer the texts to analyse are, the better GIR systems perform to detect place names, as sentences provide a better-formed context from which deeper semantic structure can be analysed [2]. GIR also involves a disambiguation step, as many place names are also related to person names (e.g., San Francisco), or have multiple matches around the world (e.g., Springfield). Disambiguation issues have been addressed in several works, as shown in [3], [4], [5] and [6].

2 Introduction

Twitter has become a very popular communication tool among Internet users for opinion and event broadcasting, as well as following news and facts around the world. Each day, more than 175 million tweets (micro-posts of a maximum length of 140 characters) are published in the social network, issuing significant events in their more than 200 million users daily life [7]. This real time information sharing opens new research opportunities to analyse Twitter’s stream in order to apply data mining techniques to extract relevant information. Studies like [8] and [9] have addressed the capacity to track emergency events and how they evolve, as people usually first post news on Twitter, and are later broadcasted by traditional media corporations [10]. Alerts can be send as soon as an emergency event is detected (known as First Story Detection - FSD [11]), providing relevant information gathered from the conversations around the incident to the corresponding emergency response teams. One of the biggest drawbacks of this process is identifying the location where the emergency is taking place. Twitter allows the possibility to add users location to each tweet, but enabling this feature is not widely spread, so other techniques must be applied in order to try to position the event on a map [12]. We present an
approach to detect locations in Twitter conversations, providing the most precise information available to response teams.

3 Location Problems

The traditional approach to identify locations programmatically is to extract place names using a Named Entity Recognition (NER) system. NER analysers annotate terms related to different entities (i.e., persons, organizations, locations, etc.), but they miss lots of place names due to two main Twitter characteristics: the short length of the posts (limited by the platform) and the informal language used by many users, which provides limited both syntactic and semantic contexts to analyse, making the annotation process much harder [13], [14].

3.1 Noise reduction in tweets

To increase the number of annotations the system is able to detect in a tweet’s text, a slang cleaner (SC) module has been developed. The objective of this module is to detect common slang words within a post, and replace them with the equivalent expression in plain language, (e.g., “fyi” will become “for your interest”), thus enriching each tweet’s context. Special tags used in Twitter and stop words (SW) are also ignored: words starting with RT (retweets), @ (mentions), # (hashtags) and URLs (links to other resources) are not analysed. This approach is also followed by [15]. Preliminary tests show that correcting slang words in tweets before analysing them with NLP tools increases the number of annotated entities:

<table>
<thead>
<tr>
<th>Location system</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stanford nlp</td>
<td>87.36</td>
<td>90.12</td>
<td>96.05</td>
<td>92.99</td>
</tr>
<tr>
<td>Stanford nlp + SC</td>
<td><strong>87.78</strong></td>
<td>90.48</td>
<td>96.20</td>
<td>93.25</td>
</tr>
<tr>
<td>Twitter nlp tools</td>
<td>76.25</td>
<td>81.43</td>
<td>90.48</td>
<td>85.71</td>
</tr>
<tr>
<td>Twitter nlp tools + SC</td>
<td><strong>82.86</strong></td>
<td>83.87</td>
<td>96.30</td>
<td>89.66</td>
</tr>
</tbody>
</table>

Table 1: Results comparison using Google Reverse Coder

The corpus used for testing was collected during the earthquakes that hit Japan between 15-17 January, 2013. The keywords used to crawl the Twitterverse using the Twitter streaming API 1 were: Earthquake Japan, Earthquake Chiba, Earthquake Katsuura and their variants, forming a total amount of 1013 tweets. Results provided by the system using Google Reverse Coder have been manually verified, considering false positives and wrong location disambiguations as invalid results. Accuracy in Table 1 shows correct percentage of full corpus elements using different location systems. All files involved in this process can be downloaded from the resources folder in Github 2. Location names are uniquely stored, so new instances of “Hokkaido” in the tweets will not launch new queries to geolocation APIs.

Since many false positives are thrown as a result of this process (i.e., annotating entities as locations when they are not), further refinement is needed.

3.2 Disambiguation

Even real place names may not benefit from the geolocation process of an emergency event. Rarely people refer to an incident giving precise addresses, ending with area references as general

---

1 https://dev.twitter.com/docs/streaming-apis  
2 https://github.com/morelab/twitter-nlp
as cities or even whole states. Certain emergency events affect large areas; such as earthquakes or floodings which may spread through various states, but different type of incidents could only affect a neighbourhood, a certain street address, etc. Disambiguation can also be found when different real world locations are named using the same exact word (e.g., given Springfield, about 70 places worldwide show up, 38 of them within the United States). It is easy to imagine the following question will soon arise: which Springfield are users referring to? And if a car crash is being reported, where in the whole city has it taken place? Therefore, the most accurate locations available must be searched for, providing response teams a way to track event spreading through a map.

4 Proposed solution

Other studies have tackled this problem relying on two basis: geotagged tweets and user profiles location information [16]. As many people avoid auto-geotagging due to privacy concerns, and user profile’s information usually does not address the current location of the user, new techniques need to be applied, as this proposal demonstrates.

For this experiment, two different NER systems have been compared: the standard Stanford NER classifier 3, and a specially built NLP tool trained for maximizing results when analysing tweets, described in 4. Different corpora have been gathered for experiments using the Twitter Streaming API 5, querying for different incidents worldwide. This corpora is formed by:

≃ 3M tweets of general purpose incidents on mid-November, 2012
≃ 2k tweets addressing the Evan Cyclone hitting Fiji Islands on December 2012
≃ 40k tweets regarding Tasmanian fires in early January 2013
≃ 1k tweets from earthquakes striking Japan in January 2013

As observed during the crawling process, tweets in emergency events tend to be clustered in conversation graphs. The relationships between nodes (original tweets) are formed by replies and retweets, two Twitter features that allow users to answer other users posts providing new data and spreading the latest news over the network. Thanks to these features, first reporters become reference journalists of the conversation, gaining trustworthiness and aggregating relevant data from other Twitter users, allowing information to persist as the event evolves [18]. As shown in [19], retweets (Twitter’s way to broadcast someone’s post without altering its content) are the de-facto way to propagate emergency-related information within the Twittersverse during emergency events. This behaviour is where our solution relies on, as data gathered from different points of view enriches the whole alert message. As graphs evolve over time (whilst new facts are reported and the twitterverse echoes the alert), their content must be checked repeatedly. For each tweet, the system will try to extract the most accurate available location (if any), using the schema described by OpenStreetMap’s Nominatim service 6, in which each level provides more precise location information than the previous one:

---

3http://nlp.stanford.edu/software/CRF-NER.shtml
4https://github.com/ aritter/twitter_nlp
5https://dev.twitter.com/docs/streaming-apis
6http://wiki.openstreetmap.org/wiki/Nominatim
Tweets are annotated by the previously described tools, and each identified location name is run through 4 different geolocalization services and APIs, which results can be observed in Table 3, in order to determine if a real location can be resolved (i.e., geo/non-geo disambiguation).

Each valid location (with the precision given by its depth in the schema above), will be remembered, so new tweets addressing the same location will not be run through all the query process, but will be used to provide more relevance to it. If a more precise location is detected (e.g., a unique suburb name when all the identified locations work at city level), the new place is checked to analyse if it helps to make more concrete the identified parents. If so, non-matching parents are discarded. Otherwise, the new location is analysed to see if a new event is spreading.

As seen on Figure 2, the algorithm helps both to get rid of non-matching “Londons”, and to move forward to a more precise location information, such as “Picadilly Circus”.

Finally, if tweets within a conversation are geotagged and the up-chain makes sense, the event can be pin-pointed over a map with the highest accuracy. Although, this is an ideal state, and if no geotagged tweets are posted, the system will only provide the lowest-level verified area as the event location, that is, the most trustworthy information that automatic systems can feed to emergency response teams.

5 Results

To evaluate the performance of both NER systems (i.e., the standard Stanford NER classifier and the Twitter NLP tools package), different tweet corpora have been annotated. Table 2 presents the accuracy classification results and the number of entities detected by each tool in the same corpora. In addition, slang cleaner and stop words filters application has also been calculated.
<table>
<thead>
<tr>
<th>Location system</th>
<th>Accuracy (%)</th>
<th>Total locations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stanford nlp</td>
<td>87.36</td>
<td>76</td>
</tr>
<tr>
<td>Stanford nlp + SC</td>
<td>88.89</td>
<td>80</td>
</tr>
<tr>
<td>Stanford nlp + SW</td>
<td>92.50</td>
<td>74</td>
</tr>
<tr>
<td>Stanford nlp + SC + SW</td>
<td><strong>92.68</strong></td>
<td><strong>76</strong></td>
</tr>
<tr>
<td>Twitter nlp tools</td>
<td>77.50</td>
<td>62</td>
</tr>
<tr>
<td>Twitter nlp tools + SC</td>
<td>84.29</td>
<td>59</td>
</tr>
<tr>
<td>Twitter nlp tools + SW</td>
<td>81.01</td>
<td>64</td>
</tr>
<tr>
<td>Twitter nlp tools + SC + SW</td>
<td><strong>90.41</strong></td>
<td><strong>66</strong></td>
</tr>
</tbody>
</table>

Table 2: Results comparison between Stanford NLP and Twitter NLP tools using SlangCleaner (SC) + StopWords Removal(SW)

In Table 3, different geolocalization APIs have been compared. Results have demonstrated that Google Reverse Coder and Yahoo Geoplanet services generate a similar accuracy, higher than the other services, so further improvements can rely on these proved solutions.

<table>
<thead>
<tr>
<th>Rates</th>
<th>Nominatim</th>
<th>Geonames</th>
<th>Google Reverse Coder</th>
<th>Yahoo Geoplanet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tweets</td>
<td>1.184</td>
<td>1.184</td>
<td>1.184</td>
<td>1.184</td>
</tr>
<tr>
<td>Detected locations</td>
<td>55</td>
<td>31</td>
<td>75</td>
<td>79</td>
</tr>
<tr>
<td>Undetected locations</td>
<td>27</td>
<td>51</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Location Rate (%)</td>
<td>67.07%</td>
<td>37.80%</td>
<td>91.46%</td>
<td>96.34%</td>
</tr>
<tr>
<td>Accuracy (%)</td>
<td>64.56%</td>
<td>40.51%</td>
<td><strong>84.81%</strong></td>
<td><strong>84.81%</strong></td>
</tr>
</tbody>
</table>

Table 3: Results comparison between geolocation tools using “Stanford N + SC + SW”

6 Conclusions and future work

Results have demonstrated that a geolocation improvement system based on both NER analysis and geolocation web services provides better performance regarding situational-awareness than current techniques, which rely just on geotagged tweets and profile’s information in the Twitterverse. Results also suggest that the developed system can be used as a valid GIR tool in different social networks or mobile phone short messages. Future work should improve the system’s performance and accuracy by defining probability formulae where each location name appearance will be given a balanced weight according to its relevance within the actual state of the conversation graph and the best online geolocalization APIs results. As this graphs evolve, re-computation of the relevance weights should derive on accuracy increases.

7 Acknowledgements

This research was funded by the SABESS project, Innpacto Project Funding of the Spanish government, grant agreement no. IPT-2011-1052-390000.

---

7 [http://nominatim.openstreetmap.org/search](http://nominatim.openstreetmap.org/search)
9 [https://developers.google.com/maps/documentation/geocoding/](https://developers.google.com/maps/documentation/geocoding/)
References


