Geotagging Tweets Using Their Content

Sharon Paradesi
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
paradesi@csail.mit.edu

Abstract

Harnessing rich, but unstructured information on social networks in real-time and showing it to relevant audience based on its geographic location is a major challenge. The system developed, TwitterTagger, geotags tweets and shows them to users based on their current physical location. Experimental validation shows a performance improvement of three orders by TwitterTagger compared to that of the baseline model.

Introduction

People use popular social networking websites such as Facebook and Twitter to share their interests and opinions with their friends and the online community. Harnessing this information in real-time and showing it to the relevant audience based on its geographic location is a major challenge. The microblogging social medium, Twitter, is used because of its relevance to users in real-time.

The goal of this research is to identify the locations referenced in a tweet and show relevant tweets to a user based on that user's location. For example, a user traveling to a new place would would not necessarily know all the events happening in that place unless they appear in the mainstream media (television, newspaper or online news articles). The system developed, TwitterTagger, geotags tweets in near real-time and shows tweets related to surrounding areas. Experiments show a performance improvement of three orders by TwitterTagger compared to the baseline model.

System Design

Figure 1 shows the architecture of TwitterTagger, the system developed to geotag tweets. Tweets obtained using the Twitter Streaming API are inputs to TwitterTagger. Then, a Part-of-Speech Tagger is used to tag the content of each tweet. After this stage, two disambiguations are performed in order to clarify the connotations of the noun phrases in each tweet, and to associate correct locations with each tweet. The successfully geotagged tweets are then displayed to the user.

Part of Speech (POS) Tagging: The first stage performs POS tagging of the words in a tweet. The system uses Ling-Pipe POS tagger[1]. The noun phrases are then compared with the USGS database[2] of locations. Common noun phrases, such as 'Love' and 'Need', are also place names and would be geotagged. To avoid this, the system uses a greedy approach of phrase chunking. It constructs the largest noun phrase possible and compares it against the USGS database. If there is a match, the system geotags it; otherwise the last word in the phrase is dropped and the shortened phrase is iteratively compared against the database until there are only two words in the noun phrase.

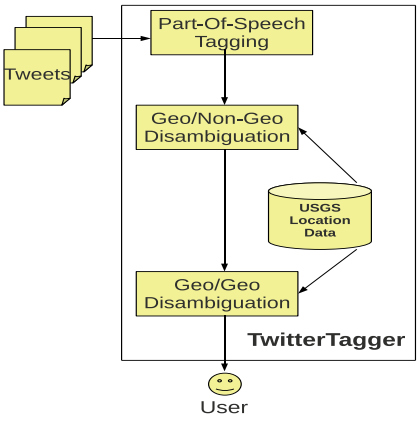

Figure 1: Architecture of TwitterTagger

Geo/Non-Geo disambiguation: The second stage helps distinguish the noun-phrases that are geographic locations from non-geographic references. For example, Sharon is the name of a city in Massachusetts but is also the name of the author. For this type of disambiguation, the following two features are used.

Feature 1: an indicator function used to check whether a 'spatial indicator' occurs before a noun phrase.

A *spatial indicator* is a syntactic construct (usually a preposition) that generally occurs before a location name. For example, the word 'in' in the sentence *She lives in Massachusetts* is a spatial indicator.

[1] http://alias-i.com/lingpipe/
[2] http://geonames.usgs.gov/

Feature 2: an indicator function to check whether other users used a spatial indicator in front of the same noun phrase in their respective tweets. Here, the system discovers if the noun phrase is considered a location by other users.

Geo/Geo disambiguation:

The final stage helps identify noun-phrases that refer to multiple geographic locations. For example, there are thirty-five states in the United States of America that have a city named Springfield or some derivative of it. For this type of disambiguation, the following two features are used.

Feature 1: the distance between the location of the noun phrase in a tweet and that of the user who tweeted it. This is measured by calculating the distance between the latitude and longitude of these two locations.

Feature 2: the distance between the location of noun phrase in a tweet and that of other users who tweeted about that noun phrase. Here, the system finds out whether other users who tweeted about a particular location reside near it.

In Geo/Non-Geo and Geo/Geo disambiguations, the weights for the first feature are learned during the training process and applied during the testing phase. The weights for the second feature, however, are learned during the testing phase. All the weights are then used in the log-linear model (Jurafsky(2008)) to calculate the best possible locations.

Experiments

To evaluate the system, I compared the baseline approach to the Geo/Non-Geo disambiguation module and finally to the entire entire system (with both types of disambiguations).

Baseline: For the baseline system, 253,724 tweets were assigned part-of-speech tags using the POS tagger. The noun phrases were checked against the USGS database to determine whether they were locations. This resulted in positive and negative subsets based on whether there was a geographic match.

Geo/Non-Geo Disambiguation: For the disambiguation between geographic location and non-geographic references, 50,762 tweets were run through a pipeline similar to that of the Baseline experiment. However, the system now filters out non-geographic references using the log-linear model before querying the locations database. Thus, the system eliminates as many false positives as possible.

Geo/Non-Geo + Geo/Geo Disambiguation: The final set-up is very similar to the earlier two set-ups but now includes both the Geo/Non-Geo and Geo/Geo disambiguation modules.

A random sample of 2,000 geotagged tweets was taken from the three set-ups and split into true positives and false positives manually. The precision of all three systems are shown in Table 1. The experiments show that TwitterTagger performs three times better than a baseline model and two time better than the Geo/Non-Geo disambiguation module.

Screenshot: The screenshot (Figure 2) shows tweets that were geotagged as being in and around New York City. The system identifies that Holland Tunnel and Pace University are both in New York and that Church Square Dog Park and Grand Sichuan are nearby in New Jersey.

Metric	Precision (%)
Baseline	4.932
Geo/Non-Geo disambiguation	7.444
Geo/Non-Geo + Geo/Geo disambiguation	15.809

Table 1: Comparison of precision values of the three systems

Figure 2: Screenshot of tweets geotagged as New York

Related Work

Amitay *et al.* (Amitay(2004)) and Li *et al.* (Li(2002)) present techniques to identify important geographic terms in documents. Both the approaches operate on documents of large sizes whereas TwitterTagger works with tweets that are limited to 140 characters.

Eisenstein *et al.* (Eisenstein(2010)) present a way to identify the location of a user based on his or her tweets. However, this work focuses on identifying the location of a tweet rather than a user's. However, profiling users is a useful way of identifying correlations between locations of users and locations of the tweets.

Conclusions

This paper shows a technique to identify the locations that are referred in a tweet. Experiments show that the performance improvement by TwitterTagger is three times when compared to that of the baseline model.

References

Amitay, E.;Har'El, N.; Sivan, R.; and Soffer, A. Web-a-where: geotagging web content. Proceedings of ACM SIGIR , 2004.

Eisenstein, J.; OConnor, B.; Smith, N. A.; and Xing, E. P. A Latent Variable Model for Geographic Lexical Variation. Proceedings of EMNLP , 2010.

Li, H.;Srihari, R. K.; Niu, C; and Li, W. Location Normalization for information extraction. Proceedings of COLING, 2002.

Jurafsky, D.; and Martin, J. H. Speech and Language Processing, Prentice Hall; 2nd edition, 2008.