In *Proceedings of the 20th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Redondo Beach, CA, USA, November 2012.

TweetPhoto: Photos from News Tweets*

Brendan C. Fruin Hanan Samet Jagan Sankaranarayanan†
Center for Automation Research, Institute for Advanced Studies,
Department of Computer Science, University of Maryland
College Park, MD 20742 USA
{brendan, hjs, jagan}@cs.umd.edu

ABSTRACT

TweetPhoto utilizes a map query interface to display news photos from news articles that are extracted from the tweets of 2,000 Twitter users who have been determined to post news related content. These articles are then geotagged and clustered so that a set of locations are associated with a cluster and its associated images. For each of these locations, the images are scored based on the terms associated with the location and the image's caption. This work differs from traditional work in this area as all topic and location extraction is automated without the need for user entered content or GPS coordinates.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Spatial databases and GIS*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces

General Terms

Algorithms, Design, Performance

Keywords

NewsStand, Smartphone, Twitter, App

1. INTRODUCTION

TweetPhoto (see also the related NewsStand [17, 20, 25], STEWARD [13], and TwitterStand [22] systems) is an example application of a general framework being developed at the University of Maryland at College Park for retrieving multimedia data (e.g., text, images, videos) using a map query interface (see also the related systems QUILT [19, 24] and the SAND Browser [18]) from a database of news articles, news photos, and news videos (i.e., by location which differentiates it from Google where the photos are retrieved by their subject matter and often by their constituent image features). The images are news photos from news articles [21] that have been extracted from the tweets posted by 2,000 Twitter users who have been determined to post news-related content. These articles are subsequently geotagged [9, 10, 11, 12, 16] and clustered to that a set of locations are associated with a cluster, and its corresponding images and tweets. For each of these locations, the images are scored based on the terms associated with the location and the image's caption. This work differs from traditional work in this area as all topic and location extraction is automated without the need for user entered content or GPS coordinates. Moreover, the result is that the photos can be accessed on the basis of the places what they are about rather than the locations where the references to their associated articles were tweeted. The photos are accessed given a location or window [1] via a map query interface.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 describes Twitter, how articles are clustered, captions extracted, and images scored. Section 4 describes the user interface. Section 5 describes the statistics of the database, while the future work is outlined in Section 6.

2. RELATED WORK

Most of the work dealing with geotagging of tweets or images has relied on analyzing manually entered user information such as tags (hashtags in Twitter parlance) and location keywords or the location is extracted from location information generated from a GPS enabled device [2, 4, 5, 7, 8, 14, 23, 26] For another approach, see the Google Maps'[5] photos layer which shows a world map overlaid with user submitted images at the location where the images were taken. Similar to TweetPhoto's user interface, the set of displayed photos changes as the viewing window changes. However, the photos layer of Google Maps requires image locations to be tagged for each image while TweetPhoto automatically geotags an image using the article where the image was found. TweetPhoto also focuses on images pertaining to the news which results in photos updating often whereas photos in Google Maps are of geographical landmarks which update less often. For example, the image hosting site Flickr has a map [4] that displays markers for the top geotagged locations in the world. Unlike TweetPhoto, Flickr's map does not update to display new images when the viewing window changes. This means that locations which have images, but are not in the

*This work was supported in part by the National Science Foundation under Grants IIS-07-13501, IIS-08-12377, CCF-08-30618, IIS-09-48548, IIS-10-18475, and IIS-12-19023.

†Current Address: NEC Labs America, 10080 North Wolfe Road SW #3350, Cupertino CA 95014

ACM SIGSPATIAL GIS '12, November 6-9, 2012. Redondo Beach, CA, USA

top geotagged image locations are not shown. Flickr's map also relies on GPS tagged information or manually entered user information. Kinsella et al. [8] geotags Twitter data for varying levels of granularity, but focuses on where the user was when they tweeted as opposed to the content of the message that was tweeted as is the case in TweetPhoto.

In addition, much work has addressed the clustering and scoring of images to produce a relative rank amongst images in a collection (e.g., [3, 6, 15]). Epshtein et al. [3] clustered a set of images based on the location that the picture was taken and the viewing angle. This methodology often groups images of the same landmark or location which would not allow for related but distinct news images, as in our application, to be clustered together. Jaffe et al. [6] proposed a solution to the problem of clustering and ranking images using the image location, image tags, the photographer, and other auxiliary metadata. Unfortunately, this relies on extra information that is often not present in images from news articles. More generally, the main drawback of all the above methods is their reliance on external data sources, while TweetPhoto is able to perform all clustering, ranking and geotagging operations with its own data.

3. METHODS

In this section we describe some of the basic concepts of Twitter which are useful in understanding the design of TweetPhoto. We then briefly describe our article clustering, our method of image extraction, and our image ranking system.

3.1 Twitter

Twitter is a social networking site that allows its users to post and read brief 140 character messages termed *tweets*. As of March 2012, there were over 140 million active users posting 340 million tweets a day [27]. Tweets often contain keywords prepended with a "#" known as a *hashtag*. Tweets can also contain a URL to a news article, image or video.

TweetPhoto and the related TwitterStand system are populated by a hand selected group of 2,000 users known to post news whom we term the *seeders*. The tweets from the seeders usually contain links to news related content. Due to the length restriction on a tweet, URL shortners such as tinyurl.com, bit.ly, and goo.gl are used which can substantially reduce the length of a URL while still directing the user to the appropriate webpage. These shortened URLs are detected by our system and for each page the contents are downloaded and stored in our database. Once the original contents have been downloaded, we clean up the text by removing HTML tags, JavaScript and any unnecessary information that does not pertain to the contents of a news article. In this way, we are able to successfully extract the text of the document along with associated media such as images or videos.

3.2 Article Clustering

We use the *vector space model* of documents which represents a text document as a *term feature vector* in a d-dimensional space, where d is the number of distinct terms in every document in a corpus.

Upon receiving a new article to be clustered, we first normalize the article's content by *stemming* input terms and removing punctuation and other extraneous characters. Next, we extract the article's term feature vector by computing the well-known *Term Frequency-Inverse Document Frequency* (TF-IDF) score for each term in the article. This score emphasizes terms that are frequent in a particular document and infrequent in a large corpus D of documents thereby marking it as significant.

Our clustering algorithm is a variant of leader-follower clustering that permits online clustering in the term vector space. For each cluster, we maintain a *term centroid* and *time centroid*, corresponding to the means of all term feature vectors and publication times of articles in the cluster, respectively. To cluster a new article a, we check whether there exists a cluster where the distance from its term and time centroids to a is less than a fixed cutoff distance ϵ. If one or more candidate clusters exist, then a is added to the closest such cluster, and the cluster's centroids are updated. Otherwise, a new cluster containing only a is created. The term distances between the new article and candidate clusters are computed with a variant of the *cosine similarity measure* which for article a and cluster c is defined as

$$\delta(a,c) = \frac{\overrightarrow{TFV_a} \bullet \overrightarrow{TFV_c}}{||\overrightarrow{TFV_a}||\,||\overrightarrow{TFV_c}||}$$

where $\overrightarrow{TFV_k}$ is the term feature vector of k.

3.3 Caption Extraction

Our image scoring system depends in part on the captions of the images which are textual descriptions of the images. This is part of the image extraction process from the web page of the HTML RSS news feeds which requires processing of the HTML tags in the feeds so that captions can be associated with each of the images in the news articles. Observe that we may not be able to identify a caption for each of the images in a news article (as some may be irrelevant such as advertisements), but from our experience, we are able to do so for a large percentage of them. Even though the captions of images are usually not very descriptive due to their succinctness, they still capture the image's content in the sense that they have terms in common with the text, and hence the captions and text are said to be *similar*.

We examine every image in the HTML page. If we can visualize the HTML as a tree structure, and the image as a node in the tree, then the idea is to look at the children nodes and a few ancestor nodes to try to collect enough text which would serve as the caption of the image. Note that the caption is usually not very long, which means that we can simply discard any text if it is too long. In addition, note that we require that the image have a minimum size and an aspect ratio greater than 1.5, as is typical with images accompanying news articles.

Once we have a caption for the image, we try to match the terms (i.e., words) it contains with the document's cluster's term centroid (described in Section 3.2) to see how many keywords from the cluster are found in the caption. For example, if the feature vector of the document contains "Greece", "EU", "Debt", and "Bailout", then we would expect that one or more of these "features" are present in the caption text. If not, then we simply discard the image. Once the image has been extracted, we also record the caption text and the cluster term centroid in the database.

3.4 Image Scoring

After the images from a news article are extracted and clustered, we use the image's caption for scoring. Each article (and the media contained within the article) is associated with a news cluster which is in turn associated with a set of locations and a set of keywords along with their frequency with which they appear in a given cluster. We use a mod-

ified version of the Jaccard Index where the numerator is the sum of the cardinality of the intersection of the location keyword set with the words in the caption and the frequency of the words in the intersection set. The denominator is the sum of the cardinality of the union of the location keyword set and the caption words and the sum of all the frequencies of the location keywords. The final result is a score, *image_score* between 0 and 1 where 0 indicates that the two sets have nothing in common and 1 corresponds to the case where the two sets are the same. In order to take the recency into consideration when assigning a score, the *image_score* is weighted by adding an integer value known as *days_offset* which is defined to be the number of days since the first image was scored. By doing this, our image scoring process is guaranteed to have more recent images as higher scoring images and the highest scoring image will not permanently remain in this position.

4. TWEETPHOTO USER INTERFACE

The TweetPhoto user interface initially shows a map with images as place markers at the locations that have the most images in the current viewing window. As the viewing window changes by a user panning or zooming, the images are repopulated in order to only retrieve images that are associated with visible latitude and longitude points. This is done by a query to our PostgreSQL database where the results are ordered by the score of the image. Many images may be associated with nearby or the same locations which would result in a cluttered map with overlapping image markers if we were to display all of the results. In order to deal with this issue, only a subset of the images returned from our query may be shown. Since the results of the call to our database is sorted by image score, we know that it is more important to show the earlier images as opposed to the later ones. After the results of the query are returned, we begin by inserting the first image as it is the highest scoring image and there are no other images that it could intersect. We calculate the location and area that the image requires and insert this region into a collection which we term the *bounding_rectangles* which is initially empty. For each image, we calculate the area and determine whether it intersects with any of the rectangles currently in the *bounding_rectangles* displaying the image on the map and inserting it into the *bounding_rectangles* if it does not. The result is a map displaying the highest scoring images where none of the image markers intersect each other. We also allow the user to specify to what extent image markers can intersect in order to show more images that may only intersect in a small region.

If the user selects an image marker, the image marker increases in size, the location name and the image caption are shown along with a right pointing arrow. By clicking (or tapping) the arrow, the interface switches from the map to a grid of all the images associated with the selected location. By default, the interface shows all of the images in descending order of their score. The user can change this by marking the duplicate images, removing the duplicate images or separating the images by topic (their respective clusters). Upon selection of an image, the image is enlarged and its caption is displayed underneath of it. If the selected image is clicked or tapped again, the tweet containing the news article where the image comes from is shown. At any point the user can return to the map view, by selecting the "Map" button located at the upper left hand corner of the screen.

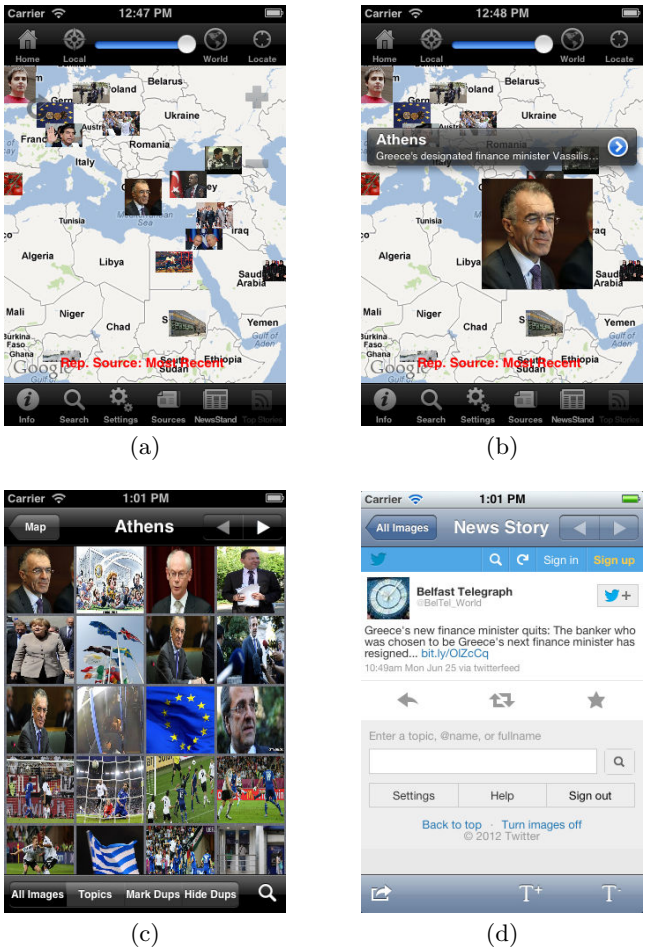

Figure 1: The TweetPhoto user interface as implemented on an iPhone/iPod Touch: (a) map of news photos from tweeted news articles; (b) result of selecting an image from the map of TweetPhoto; (c) grid of images associated with a selected location (Athens) in descending order of score; (d) result of selecting an image from the grid of TweetPhoto and viewing its source tweet.

5. STATISTICS

TweetPhoto and the related TwitterStand systems currently have a PostgreSQL database size of approximately 85 gigabytes. TweetPhoto using its hand selected Twitter users, the *seeders*, downloads over 60,000 news articles extracted from tweets. From these articles, over 15,000 images are downloaded per day, determined to be relevant to the article and scored. Of these 15,000 images, about 35% are unique images while the rest are duplicate or near duplicate images as determined by our system.

6. FUTURE WORK

TweetPhoto is currently limited to images contained within news articles. Many users of Twitter tweet images using Twitter's built in image system or using a third party site such as twitpic.com, flickr.com or yfrog.com. An obstacle in adding these images is that we are limited to only the text of the tweet to determine whether the image is related to news. We also would have to rely on the metadata of these different sites when geotagging the content of the image.

7. REFERENCES

[1] W. G. Aref and H. Samet. Efficient processing of window queries in the pyramid data structure. In *Proceedings of the 9th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 265–272, Nashville, TN, Apr. 1990. Also in *Proceedings of the Fifth Brazilian Symposium on Databases*, pages 15–26, Rio de Janeiro, Brazil, April 1990.

[2] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *WWW'09: Proceedings of the 18th International World Wide Web Conference*, pages 761–770, Madrid, Spain, Apr. 2009.

[3] B. Epshtein, E. Ofek, Y. Wexler, and P. Zhang. Hierarchical photo organization using geo-relevance. In *GIS'07: Proceedings of the 15th International Symposium on Advances in Geographic Information Systems*, Seattle, WA, Nov. 2007.

[4] Flickr. `http://www.flickr.com/map`, 2012.

[5] Google. Google maps. `http://www.google.com/maps`, 2012.

[6] A. Jaffe, M. Naaman, T. Tassa, and M. Davis. Generating summaries and visualization for large collections of geo-referenced photographs. In *MIR'06: Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 89–98, Santa Barbara, CA, Oct. 2006.

[7] D. Joshi, A. Gallagher, J. Yu, and J. Luo. Inferring photographic location using geotagged web images. *Multimedia Tools and Applications*, July 2010. Published online.

[8] S. Kinsella, V. Murdock, and N. O'Hare. "i'm eating a sandwich in glasgow": modeling locations with tweets. In *SMUC'11: Proceedings for the 3rd international workshop on Search and mining user-generated contents*, pages 61–68, Glasgow, Scotland, UK, Oct. 2011.

[9] M. D. Lieberman and H. Samet. Multifaceted toponym recognition for streaming news. In *Proceedings of the 34th International Conference on Research and Development in Information Retrieval (SIGIR'11)*, pages 843–852, Beijing, China, July 2011.

[10] M. D. Lieberman and H. Samet. Adaptive context features for toponym resolution in streaming news. In *Proceedings of the 35th International Conference on Research and Development in Information Retrieval (SIGIR'12)*, pages 731–740, Portland, OR, Aug. 2012.

[11] M. D. Lieberman, H. Samet, and J. Sankaranarayanan. Geotagging: Using proximity, sibling, and prominence clues to understand comma groups. In R. Purves, C. Jones, and P. Clough, editors, *Proceedings of 6th Workshop on Geographic Information Retrieval*, Zurich, Switzerland, Feb. 2010. online proceedings.

[12] M. D. Lieberman, H. Samet, and J. Sankaranarayanan. Geotagging with local lexicons to build indexes for textually-specified spatial data. In *Proceedings of the 26th IEEE International Conference on Data Engineering*, pages 201–212, Long Beach, CA, Mar. 2010.

[13] M. D. Lieberman, H. Samet, J. Sankaranarayanan, and J. Sperling. STEWARD: Architecture of a spatio-textual search engine. In *GIS'07: Proceedings of the 15th ACM International Symposium on Geographic Information Systems*, pages 186–193, Seattle, WA, Nov. 2007.

[14] M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *JCDL'04: Proceedings of the 2004 ACM/IEEE Joint Conference on Digital Libraries*, pages 53–62, Tucson, AZ, June 2004.

[15] S. Overell, B. Sigurbjörnsson, and R. van Zwol. Classifying tags using open content resources. In *WSDM'09: Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, pages 64–73, Barcelona, Spain, Feb. 2009.

[16] G. Quercini, H. Samet, J. Sankaranarayanan, and M. D. Lieberman. Determining the spatial reader scopes of news sources using local lexicons. In A. El Abbadi, D. Agrawal, M. Mokbel, and P. Zhang, editors, *Proceedings of the 18th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 43–52, San Jose, CA, Nov. 2010.

[17] H. Samet, M. D. Adelfio, B. C. Fruin, M. D. Lieberman, and B. E. Teitler. Porting a web-based mapping application to a smartphone app. In *GIS'11: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 525–528, Chicago, Nov. 2011.

[18] H. Samet, H. Alborzi, F. Brabec, C. Esperança, G. R. Hjaltason, F. Morgan, and E. Tanin. Use of the SAND spatial browser for digital government applications. *CACM: Communications of the ACM*, 46(1):61–64, Jan. 2003.

[19] H. Samet, A. Rosenfeld, C. A. Shaffer, and R. E. Webber. A geographic information system using quadtrees. *Pattern Recognition*, 17(6):647–656, November/December 1984.

[20] H. Samet, B. E. Teitler, M. D. Adelfio, and M. D. Lieberman. Adapting a map query interface for a gesturing touch screen interface. In *WWW'11: Proceedings of the 20th International World Wide Web Conference*, pages 257–260, Hyderabad, India, Mar. 2011.

[21] J. Sankaranarayanan and H. Samet. Images in news. In *Proceedings of the 24th International Conference on Pattern Recognition*, pages 3240–3243, Istanbul, Turkey, Aug. 2010.

[22] J. Sankaranarayanan, H. Samet, B. Teitler, M. D. Lieberman, and J. Sperling. TwitterStand: News in tweets. In *GIS'09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 42–51, Seattle, WA, Nov. 2009.

[23] P. Serdyukov, V. Murdock, and R. van Zwol. Placing Flickr photos on a map. In *SIGIR'09: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 484–491, Boston, July 2009.

[24] C. A. Shaffer, H. Samet, and R. C. Nelson. QUILT: a geographic information system based on quadtrees. *IJGIS: International Journal of Geographical Information Systems*, 4(2):103–131, Apr. 1990.

[25] B. E. Teitler, M. D. Lieberman, D. Panozzo, J. S. ayanan, H. Samet, and J. Sperling. NewsStand: A new view on news. In *GIS'08: Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 144–153, Irvine, CA, Nov. 2008.

[26] K. Toyama, R. Logan, A. Roseway, and P. Anandan. Geographic location tags on digital images. In *MM'03: Proceedings of the 11th ACM International Conference on Multimedia*, pages 156–166, Berkeley, CA, Nov. 2003.

[27] Twitter. Twitter turns six. `http://blog.twitter.com/2012/03/twitter-turns-six.html`, Mar. 2012.