A Tweet Consumer's Look At Twitter Through Linked Data Goggles Via Google Analytics

Thomas Steiner and Arnaud Brousseau

Google Germany GmbH, ABC-Str. 19, 20354 Hamburg, Germany
{tomac,arnaudb}@google.com

Abstract. Twitter Trends[1] allows for a global or local view on "what's happening in my world right now" from a tweet producers' point of view. In this paper, we discuss the possibility to complete the functionality provided by Twitter Trends by having a closer look at the other side: the tweet consumers' – i.e., readers' – point of view. While Twitter Trends works by analyzing the frequency of terms and their velocity of appearance in tweets being written, our approach is based on the popularity of extracted named entities (in the sense of Linked Data) in tweets being read. Our experimentation architecture takes advantage of the possibility to use a client-side browser extension to harvest and dissect tweets from users' timelines, search result pages, or profile pages; i.e., tweets supposed to be read. Named entities are extracted via several third-party Natural Language Processing (NLP) Web services in parallel, and are then reported to Google Analytics, which is used to store, analyze, and compute trends by pivoting Analytics data, e.g., users' geographic location, with the recorded named entities.

1 Introduction

Twitter Trends analysis is a fascinating field to many, with both an academic or an industry background (e.g., [4] by Kannan et al. has a good overview of Twitter Trends visualization efforts). What all these approaches have in common is that they focus on the tweet producers' point of view. Either they are based on the output from Twitter's Trends API[2] (i.e., use Twitter's pre-calculated trends data), or use the Twitter Streaming API[3] (i.e., calculate trends data on their own). What to the best of our knowledge has been missing so far is the tweet consumers' point of view. If many Twitter users tweet about a topic (e.g., a hashtag like `#typeofsex` that seems to persist in Twitter Trends over a period of several weeks), this does not necessarily mean that tweet consumers also read these tweets. Currently the closest approximation to estimating whether a tweet has been read is to check whether it has ever appeared on someone's timeline, search result page, or profile page (the tweet producer having followers is not a

[1] `http://blog.twitter.com/2008/09/twitter-trends-tip.html`
[2] `http://dev.twitter.com/doc/get/trends/`
[3] `http://dev.twitter.com/pages/streaming_api`

sufficient condition). In this paper, we thus suggest a solution for enabling tweet tracking combined with named entity extraction (NEE) for Twitter.com. This solution allows to combine data from classic Web tracking (such as user location) with our interpretation of Twitter trends, based on named entities. Others[4] have created visualizations of user location combined with trends, however, user location so far has been an approximation of what Twitter users expose on their Twitter profile pages, which is not necessarily the same as the user's physical location.

The remainder of the paper is structured as follows:

1.1 Twitter Trends

Twitter Trends was introduced by Twitter during the summer of 2008. This service was implemented to reflect "what's happening in my world right now" on the micro-blogging platform. To compute trends, Twitter uses an algorithm which analyzes the words and hashtags in tweets. Although the concrete implementation details are kept secret, it is known[5] that the algorithm considers three main characteristics of a term/hashtag to determine if it is part of a "trend":

- the quantity, i.e, the absolute number of appearances of a given term/hashtag among all users' tweets.
- the velocity, i.e, the frequency of appearance of that term/hashtag. The higher the frequency, the more popular the term/hashtag.
- the newness, e.g, the (at the time of writing brand new) hashtag `#ipad2` will be given priority over an old and generic tag like `#awesome` to be featured as "trend". The freshness of trends is also assured by the fact that the frequency is computed over a recent set of tweets, although we have no further details of the exact computation period – Twitter keeps this information secret.

1.2 Google Chrome Extensions

Google Chrome extensions[6] are small software programs that can be installed to enrich the browsing experience with the Google Chrome browser. They are written using a combination of standard Web technologies, such as HTML, JavaScript, and CSS. Chrome extensions bundle all their resources into a single file that gets usually (but not necessarily) distributed through the Chrome Web Store[7]. There are several types of extensions, for this paper we focus on extensions based on so-called content scripts. Content scripts are JavaScript programs that run in the context of Web pages, similar to the Firefox Greasemonkey extension[8]. By using the standard Document Object Model (DOM), they can read

[4] E.g., `http://trendsmap.com/`

[5] `http://blog.twitter.com/2010/12/to-trend-or-not-to-trend.html`

[6] Google Chrome Extensions: `http://code.google.com/chrome/extensions/index.html`. Text partly adapted from the description to be found there.

[7] `https://chrome.google.com/webstore/`

[8] Firefox Greasemonkey extension: `http://www.greasespot.net/`

or modify details of the Web pages a user visits. Examples of such modifications are, e.g., changing hyperlinks to remove potential `@target="_blank"` attributes, or increasing the font size.

1.3 Google Analytics

Google Analytics[9] is Google's free Web analysis solution allowing for detailed statistics about the visitors of a website. The software is implemented by including the Google Analytics Tracking Code (GATC), an invisible snippet of JavaScript code that a webmaster needs to add onto the to-be-tracked pages of a website. This code collects visitor data by requesting a specific 1x1 pixel image on Google's servers, where the page and user data is encoded in the query part of the image's URL. In addition to that, the snippet sets a first party cookie on visitors' computers in order to store anonymous information such as the timestamp of the current visit, whether the visitor is a new or returning visitor, and the referrer of the website that the visitor came from. Part of the shared visitor information is the IP address, which allows for IP-based geolocation of visitors. Depending on the region, the accuracy of IP-based geolocation can be down to urban district level.

2 Twitter Swarm NLP Extension

With our Twitter Swarm NLP extension[10], we inject JavaScript code via a content script into the Twitter.com homepage. By installing this extension, users explicitely opt-in to their data as a Twitter.com visitor being tracked by Google Analytics. The extension first checks if the user is logged in, and if so, retrieves the tweets of the logged-in user's timeline (`http://twitter.com/#`), or search result page (e.g., `http://twitter.com/#!/search/twitter`), or profile page (e.g., `http://twitter.com/#!/timberners_lee`) one-by-one, and performs Named Entity Extraction (NEE) via Natural Language Processing (NLP) using a remote NLP Web service (see Section 2.1) on each of the tweets. The extracted entities are then displayed on the righthand-column of the Twitter.com homepage (see Figure 1), and sent to Google Analytics for further processing (see Figure 2).

2.1 Twitter Swarm NLP Web Service

We have created a wrapper NLP Web service that merges results from existing third-party NLP Web services, namely from OpenCalais[11], Zemanta[12], AlchemyAPI[13], and DBpedia Spotlight[14]. All NLP Web services return entities with their types and/or subtypes, names, relevance, and URIs that link

[9] `http://www.google.com/analytics/`

[10] `https://chrome.google.com/webstore/detail/dpbphenfafkflfmdlanimlemacankjol`

[11] `http://www.opencalais.com/`

[12] `http://www.opencalais.com/`

[13] `http://www.alchemyapi.com/`

[14] `http://dbpedia.org/spotlight`

into the LOD cloud[15]. The problem is that each service has implemented its own type system (e.g., http://dbpedia.org/ontology/City vs. http://rdf.freebase.com/ns/location.citytown), and providing mappings for all of them would be a rather time-consuming task. However, as all services offer links into the LOD cloud, the desired type information can be pulled from there in a true Linked Data manner. The merged results for the sample query "Google Translate" is depicted below. For the sake of clarity, we just show one entity with two URIs, while the original result contained seven entities among which six were directly relevant and one was closely related (the complete result can be seen at the URL http://tomayac.no.de/entity-extraction/combined/Google%20Translate).

```
[
  {
    "name": "Google Translate",
    "relevance": 0.7128319999999999,
    "uris": [
      {
        "uri": "http://dbpedia.org/resource/Google_Translate",
        "source": "alchemyapi"
      },
      {
        "uri": "http://rdf.freebase.com/ns/en/google_translate",
        "source": "zemanta"
      }
    ],
    "source": "alchemyapi,zemanta"
  }
]
```

2.2 Technical Implementation Of the Twitter Swarm NLP Extension

Twitter.com is an Ajax-dependent website, which makes use of so-called hashbang URIs[16]. When the page Twitter.com gets loaded, the static parts can be cached, while the dynamic parts – controlled by the content of the hashbang URI – get pulled in via JavaScript. Using a so-called escaped fragment in the query part of the URL (`?_escaped_fragment_`), the state of the Twitter.com page is still accessible to Web crawlers, as specified in [7]. Currently the extension is implemented in a way to run once upon page load, however, not upon Ajax refresh events on the page.

Each tweet gets sent one-by-one to the Twitter Swarm NLP Web service, as described in Section 2.1. While the extracted named entities of all tweets always get displayed to the user (as shown in Figure 1), only named entities from tweets never seen before get sent to Google Analytics. This is to ensure that duplicate data from the same user do not falsify the statistics. It is to be noted that we do allow the same tweet to be tracked more than once (because one tweet can be read by more than one user), however, we want to exclude the case that the same tweet gets tracked more than once from the same user

[15] http://lod-cloud.net/

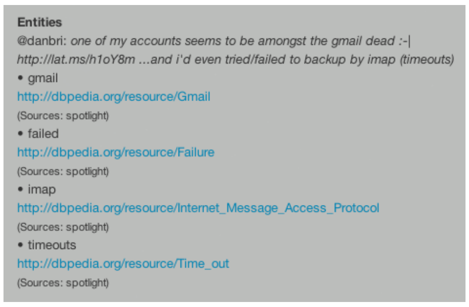
[16] See http://www.jenitennison.com/blog/node/154 for a detailed analysis of Twitter's use of hashbang URIs

(we assume each tweet gets read at most one time). Twitter has introduced a new system to generate unique tweet IDs, called snowflake[17], which guarantees k-sorted IDs (see [1]) with at least a 1s bound. Hence this allows us to store the ID of the latest tweet in the extension's local storage, and check whether the current tweet's ID is greater (i.e., the tweet is younger than the previous latest tweet and has thus not been seen before), and only if so, send its named entities to Analytics. While this strict limitation to only consider new tweets leaves out many old tweets from being tracked in Analytics, it is the only alternative to storing the ID of each and all tweets in local storage to assure duplication-free tracking.

2.3 Visual Presentation Of Named Entities

Currently the extension displays all extracted named entities grouped by originating tweets and roughly in the tweet order at the right column. In order not to flood the Twitter Swarm NLP Web service (and in turn the underlying third-party NLP Web services), the requests get sent with a 1s pause in between two tweets. Dependent on the response time for each request, the extracted named entities get added to the righthand-column. Each named entity gets displayed with its label, one representing URI (out of potentially several URIs), and the sources (as outlined in Section 2.1).

2.4 Dealing With Extracted Entities On the Google Analytics Side

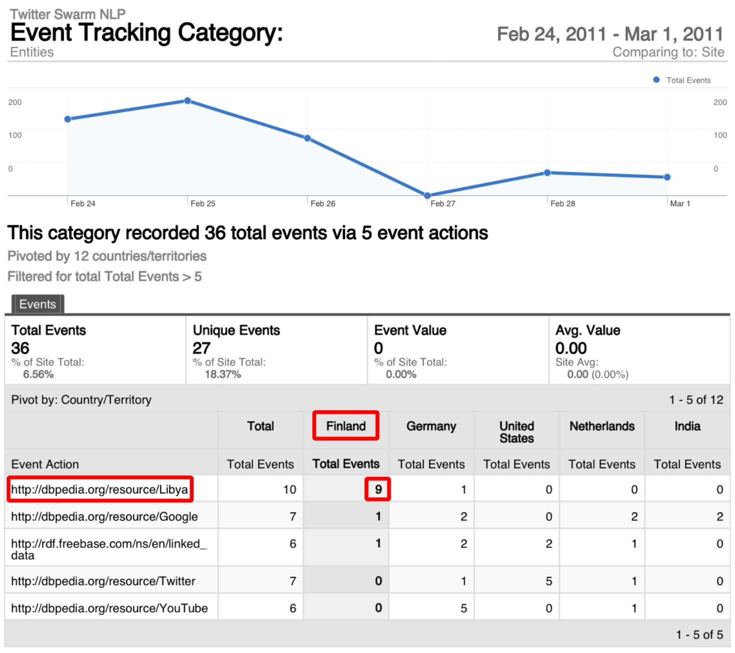


Entities
@danbri: *one of my accounts seems to be amongst the gmail dead :-| http://lat.ms/h1oY8m ...and i'd even tried/failed to backup by imap (timeouts)*

- gmail
http://dbpedia.org/resource/Gmail
(Sources: spotlight)
- failed
http://dbpedia.org/resource/Failure
(Sources: spotlight)
- imap
http://dbpedia.org/resource/Internet_Message_Access_Protocol
(Sources: spotlight)
- timeouts
http://dbpedia.org/resource/Time_out
(Sources: spotlight)

Fig. 1. Screenshot of the extracted entites of a particular tweet as displayed by the Twitter Swarm NLP Extension.

[17] `https://github.com/twitter/snowflake#readme`



Twitter Swarm NLP
Event Tracking Category:
Entities

Feb 24, 2011 - Mar 1, 2011
Comparing to: Site

This category recorded 36 total events via 5 event actions
Pivoted by 12 countries/territories
Filtered for total Total Events > 5

Events

Total Events	Unique Events	Event Value	Avg. Value
36 % of Site Total: 6.56%	27 % of Site Total: 18.37%	0 % of Site Total: 0.00%	0.00 Site Avg: 0.00 (0.00%)

Pivot by: Country/Territory 1 - 5 of 12

	Total	Finland	Germany	United States	Netherlands	India
Event Action	Total Events	Total Events	Total Events	Total Events	Total Events	Total Events
http://dbpedia.org/resource/Libya	10	9	1	0	0	0
http://dbpedia.org/resource/Google	7	1	2	0	2	2
http://rdf.freebase.com/ns/en/linked_data	6	1	2	2	1	0
http://dbpedia.org/resource/Twitter	7	0	1	5	1	0
http://dbpedia.org/resource/YouTube	6	0	5	0	1	0

1 - 5 of 5

Fig. 2. Entities pivoted by country. The entity represented by the URL `http://dbpedia.org/resource/Lybia` appeared in nine tweets on timelines of users located in Finland (red borders in the screenshot).

3 Related Work

3.1 Linked Open Social Signals (TWARQL)

In the Linked Open Social Signals project [5] Mendes et al. investigate the representation of microposts as Linked Open Data (the authors call the opinions, observations, and suggestions contained in microposts "social signals", hence the project name Linked Open Social Signals). Mendes et al. address the problem of information overload caused by the sheer amount of microposts. While the micropost community has come up with hashtags in order to categorize microposts, these hashtags are ambigious and have to be explicitely added to the micropost by the author. Given the typical length limitations of microposts (often 140 characters), sometimes hashtags are left out in favor of more text. The project's main goal is thus to enable collective analysis of social signals for sensemaking by making use of Linked Open Data principles in combination with realtime push models. The approach consists of the following steps:[18]

– Extract content (entity mentions, hashtags and URLs) from microposts

[18] Steps adapted from `http://wiki.knoesis.org/index.php/Twarql`

- Encode content in RDF format using common vocabularies
- Enable SPARQL querying of microposts
- Enable subscription to micropost streams that match a given query
- Enable scalable real-time delivery of streaming data via SparqlPuSH

The authors maintain a client-side JavaScript application[19] allowing for users to search for tweets matching a customizable SPARQL query or to subscribe to tweet streams in a realtime "push" way, filtered according to the user's request (so-called concept feeds).

3.2 Twitris 2.0

With Twitris 2.0 [3], Jadhav et al. present an application to find out what is being said about an event and when, to detect how topics of discussion are changing over a period of time, and finally to check whether there are regional differences in the opinions on a given topic. The approach consists of picking trending hashtags from Twitter, which are then expanded by data from Google Insights for Search[20]. Using this set of search terms, find related hashtags by performing a Twitter search in order to detect topic drifts. In order to locate the origin of a tweet, Twitris uses the approximation of the location given in the user's Twitter profile. This approach works well if there is geocodable data (like "Austin, Texas"), however, fails if there is generic data (like "somewhere under the rainbow"). Tweets about a given set of topics can then be examined on a map view, enriched by relevant photo and video content. For each tweet location hotspot a tag cloud with related tags is displayed, and the data can be sliced by time.

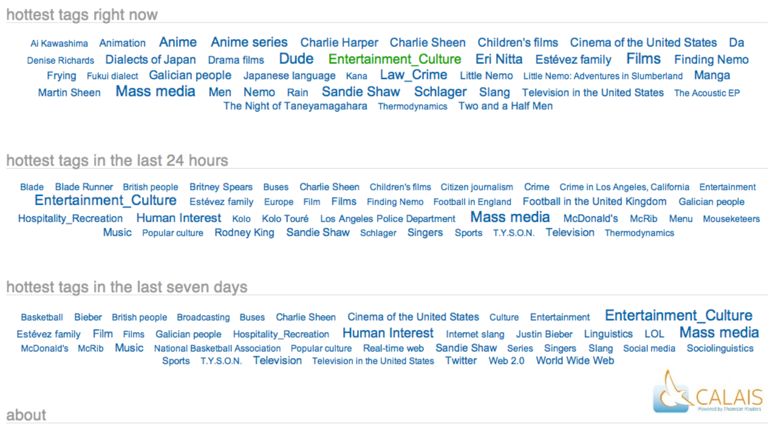
3.3 Semantic-MicrOBlogging (SMOB)

SMOB [6] by Passant et al. is a Semantic MicroBlogging framework that enables a distributed, open and semantic microblogging experience based on Semantic Web and Linked Data technologies. Microposts get annotated with common vocabularies like FOAF[21] and SIOC[22]. SMOB relies on distributed autonomous hubs that communicate with each other to exchange microblog posts and subscriptions, which can also be cross-posted to Twitter. The authors suggest people to use meaningful hashtags such as `#dbp:Eiffel_Tower`, or `#geo:Paris_France`, in the style of RDF prefixes for DBpedia [2] and GeoNames. SMOB allows for manually annotating hashtags with URIs and RDF, with the objective of making microposts accessible for, e.g., lookup services such as Sindice [8].

[19] http://knoesis1.wright.edu/twarql/query.html
[20] http://www.google.com/insights/search/
[21] http://www.foaf-project.org/
[22] http://sioc-project.org/

3.4 Twopular

Twopular[23] is an experiment by Martin Dudek with the objective of analyzing current Twitter trends. Therefore, for a given set of at the particular moment current trends the most recents tweets are obtained, and in an interval of five minutes run through the OpenCalais Web service in order find tags. By having tags for Twitter trends, another way of searching for trends – and more importantly the possibility to interrelate trends based on tag similarity – gets enabled. The author sees the feature more like a "linguistic experiment"[24], however states that the first results seem to be promising. In our tests of the service we could affirm the author's self-assessment, e.g., the Twitter trend (at the time of writing, March 7, 2011, 2PM CET) "Prince Andrew" was mapped to the OpenCalais tag "Prince Andrew, Duke of York", and related tags were, among others, "Sex offender registration", and "X-Offender", where the story behind the trend was that people were tweeting about Prince Andrew of England, whose close friend was found to be a pedophile.


hottest tags right now

Ai Kawashima Animation Anime Anime series Charlie Harper Charlie Sheen Children's films Cinema of the United States Da Denise Richards Dialects of Japan Drama films Dude Entertainment_Culture Eri Nitta Estévez family Films Finding Nemo Frying Fukui dialect Galician people Japanese language Kana Law_Crime Little Nemo Little Nemo: Adventures in Slumberland Manga Martin Sheen Mass media Men Nemo Rain Sandie Shaw Schlager Slang Television in the United States The Acoustic EP The Night of Taneyamagahara Thermodynamics Two and a Half Men

hottest tags in the last 24 hours

Blade Blade Runner British people Britney Spears Buses Charlie Sheen Children's films Citizen journalism Crime Crime in Los Angeles, California Entertainment Entertainment_Culture Estévez family Europe Film Films Finding Nemo Football in England Football in the United Kingdom Galician people Hospitality_Recreation Human Interest Kolo Kolo Touré Los Angeles Police Department Mass media McDonald's McRib Menu Mouseketeers Music Popular culture Rodney King Sandie Shaw Schlager Singers Sports T.Y.S.O.N. Television Thermodynamics

hottest tags in the last seven days

Basketball Bieber British people Broadcasting Buses Charlie Sheen Cinema of the United States Culture Entertainment Entertainment_Culture Estévez family Film Films Galician people Hospitality_Recreation Human Interest Internet slang Justin Bieber Linguistics LOL Mass media McDonald's McRib Music National Basketball Association Popular culture Real-time web Sandie Shaw Series Singers Slang Social media Sociolinguistics Sports T.Y.S.O.N. Television Television in the United States Twitter Web 2.0 World Wide Web

about

Fig. 3. Screenshot of the Twopular trends page.

4 Conclusion

Contributions: time filters (via Analytics), geographical pivoting (via Analytics), global ranking (sliceable per time unit)

As seen in Section 3, semantic analysis of a (real-time) Twitter stream is not new and has been successfully exploited to analyse tweets produced by the Twitter community. What we propose here is an insight into tweets consumers' interests to provide a more accurate view of Twitter trends.

[23] `http://twopular.com/`
[24] `http://twopular.com/blog/?p=308`

References

1. T. Altman and Y. Igarashi. Roughly sorting: sequential and parallel approach. *J. Inf. Process.*, 12:154–158, January 1989.
2. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. G. Ives. DBpedia: A Nucleus for a Web of Open Data. In *6th International Semantic Web Conference (ISWC'07)*, pages 722–735, Busan, Korea, 2007.
3. A. Jadhav, H. Purohit, P. Kapanipathia, P. Ananthram, A. Ranabahu, V. Nguyen, P. N. Mendes, A. G. Smith, M. Cooney, and A. Sheth. Twitris 2.0 : Semantically empowered system for understanding perceptions from social data. Semantic Web Challenge at the 9th International Semantic Web Conference (ISWC2010), Shanghai, China, 2010. `http://www.cs.vu.nl/~pmika/swc/submissions/swc2010_submission_8.pdf`.
4. A. Kannan, J. Patzer, and B. Avital. Trendtracker: A system for visualizing trending topics on twitter. University of California, Berkeley, Visualization Course CS294-10, Visualization Wiki, 2010. `http://vis.berkeley.edu/courses/cs294-10-sp10/wiki/images/archive/d/d4/20100511073322!FinalPaper.pdf`.
5. P. N. Mendes, A. Passant, P. Kapanipathi, and A. P. Sheth. Linked open social signals. *Web Intelligence and Intelligent Agent Technology, IEEE/WIC/ACM International Conference on*, 1:224–231, 2010.
6. A. Passant, T. Hastrup, U. Bojars, and J. Breslin. Microblogging: A semantic and distributed approach. In *Proceedings of the 4th Workshop on Scripting for the Semantic Web, Tenerife, Spain, June 02, 2008, CEUR Workshop Proceedings*, 2008. `http://CEUR-WS.org/Vol-368/paper11.pdf`.
7. K. Probst, B. Johnson, A. Mukherjee, E. van der Poel, L. Xiao, and J. Mueller. Making ajax applications crawlable. Google, 2009. `http://code.google.com/web/ajaxcrawling/docs/getting-started.html`.
8. G. Tummarello, R. Delbru, and E. Oren. Sindice.com: weaving the open linked data. In *6th International Semantic Web Conference (ISWC'07)*, pages 552–565, Busan, Korea, 2007.