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

Thomas Steiner[1], Arnaud Brousseau[1,2*], and Raphaël Troncy[2]

[1] Google Germany GmbH, ABC-Str. 19, 20354 Hamburg, Germany
{tomac,arnaudb}@google.com
[2] EURECOM, Sophia Antipolis, France
raphael.troncy@eurecom.fr

Abstract. The Twitter Trends feature 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 show the possibility to complete the functionality provided by Twitter Trends via 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 uses 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 the reported named entities by Google Analytics data, e.g., users' geographic locations.

1 Introduction

The Twitter Trends feature was introduced by Twitter in September 2008 with the objective to reflect "what's happening in my world right now" in the beginning only globally[3], but since January 2010 also locally[4]. 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 terms/hashtags to determine whether they are part of a "trend":

- The **quantity**, i.e, the absolute number of appearances of a given term/hashtag among all users' tweets.

---

[3] http://blog.twitter.com/2008/09/twitter-trends-tip.html

[4] http://blog.twitter.com/2010/01/now-trending-local-trends.html

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

- 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` would be given priority over an old and generic tag like `#awesome` to be featured as "trend".

Twitter Trends analysis is a fascinating field to many, with both an academic or an industry (or even social media hobbyist) background (e.g., [3] by Kannan et al. gives a good overview of Twitter Trends analysis and 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 the Twitter Trends API[6] (i.e., use Twitter's pre-calculated trends data), or use the Twitter Streaming API[7] (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, this does not necessarily mean that tweet consumers also read these tweets. Currently the closest practical approximation to estimate whether a tweet (on Twitter.com, not in third-party applications) 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 alone is not a sufficient condition). A different approach would be to only count tweets as read if potentially contained URLs have been clicked. This leaves out many potentially read tweets that either do not contain URLs, or only unclicked URLs.

In this paper, we suggest a solution for enabling tweet tracking combined with named entity extraction (NEE) for the Twitter.com page. This solution allows to combine data from classic Web analysis (such as user location) with our interpretation of Twitter trends, which is based on named entities. Others[8] 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 their current physical location, but oftentimes their permanent hometown.

The remainder of the paper is structured as follows: Section 2 introduces the two background technologies Google Chrome extensions and Google Analytics that are required for our experiment. Section 3 introduces our Twitter Swarm NLP Google Chrome extension. Section 4 contains an evaluation of our experimentation results so far. Section 5 gives an overview on related work. Section 6 finalizes the paper with an outlook on future work, and gives a conclusion.

2   Required Technologies

We begin with an overview of the used technologies for this experiment, namely we introduce Google Chrome extensions for the Google Chrome browser, and give a brief summary of the Web analysis solution Google Analytics.

---

[6] `http://dev.twitter.com/doc/get/trends/`

[7] `http://dev.twitter.com/pages/streaming_api`

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

2.1   Google Chrome Extensions

Google Chrome extensions[9] 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 get usually (but not necessarily) distributed through the Chrome Web Store[10]. 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 via dynamic code injection, similar to the Firefox Greasemonkey extension[11]. By using the standard Document Object Model (DOM), they can read or modify details of the Web pages a user visits. An example of such modification is, e.g., removing background images for better legibility. Content scripts can run on any website, or be limited to just certain websites. This can be controlled by the extension developer with a so-called manifest file in JSON format. When a user installs an extension, the access rights of the extension are displayed and must be acknowledged by the user.

2.2   Google Analytics

Google Analytics[12] is Google's Web analysis solution allowing for detailed statistics about the visitors of a website. The software is implemented by adding an invisible snippet of JavaScript code on the to-be-tracked pages of a website. This code collects visitor data through requests for a specific 1 x 1 pixel image on Google's servers, during which the page and user data is reported 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.

3   Twitter Swarm NLP Extension

We have developed a Google Chrome extension called Twitter Swarm NLP[13] which with we inject JavaScript code via a content script into the Twitter.com homepage. By installing this extension, users explicitly 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 to Twitter.com, and if so, retrieves the tweets of the current user's timeline (`http://twitter.com/#`), or search result page (e.g., `http://twitter.com/#!/search/%23semweb`), or profile page (e.g., `http://twitter.com/`

---

[9] `http://code.google.com/chrome/extensions/index.html`

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

[11] `http://www.greasespot.net/`

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

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

`#!/timberners_lee`) on a one-by-one basis, and performs Named Entity Extraction (NEE) via Natural Language Processing (NLP) using a remote NLP Web service (see Section 3.1) 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 Section 4).
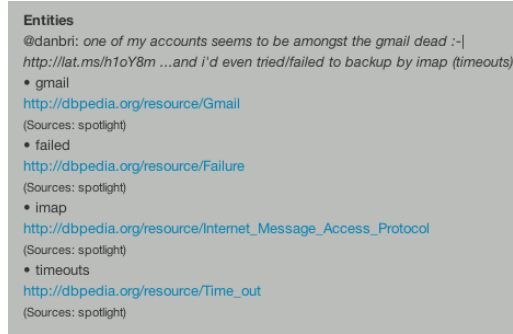
Entities
@danbri: *one of my accounts seems to be amongst the gmail dead :-|*
http://lat.ms/h1oY8m ...and i'd even tried/failed to backup by imap (timeouts)
• gmail
http://dbpedia.org/resource/Gmail
(Sources: spotlight)
• failed
http://dbpedia.org/resource/Failure
(Sources: spotlight)
• imap
http://dbpedia.org/resource/Internet_Message_Access_Protocol
(Sources: spotlight)
• timeouts
http://dbpedia.org/resource/Time_out
(Sources: spotlight)

Fig. 1. Screenshot of the original tweet above, and the thereof extracted entities below.

3.1 Twitter Swarm NLP Web Service

For our extension, we have created a wrapper NLP Web service that merges results from existing third-party NLP Web services, namely from OpenCalais[14], Zemanta[15], AlchemyAPI[16], and DBpedia[1] Spotlight[17]. All NLP Web services return named entities together with their types and potential subtypes, names, relevance, and URIs that link into the Linked Open Data (LOD) cloud[18]. The problem is that each service has implemented its own type system (e.g., the type "city" is represented by `http://dbpedia.org/ontology/City` in DBpedia vs. `http://rdf.freebase.com/ns/location.citytown` in Freebase). Providing mappings for all of them would be a rather time-consuming task. However, as all services offer links into the LOD cloud, the type information can be pulled from there if need be, in a true Linked Data manner. In order to clarify what data our wrapper service provides, the merged results for the sample query "Google Translate" are depicted below. For the sake of clarity, we just show a shortened form (the complete result can be seen at the URL `http://tomayac.no.de/entity-extraction/combined/Google%20Translate`).

```
[
  {
    "name": "Google Translate",
```

---

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

[15] `http://www.zemanta.com/`

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

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

[18] `http://lod-cloud.net/`

```
    "relevance": 0.7128319999999999,
    "uris": [
      {
        "uri": "http://dbpedia.org/resource/Google_Translate",
        "source": "alchemyapi"
      }
    ],
    "source": "alchemyapi"
  }
]
```

3.2   Technical Implementation Of the Extension

Twitter.com is an Ajax-dependent website, which makes use of so-called hash-bang URIs[19]. When the page Twitter.com gets loaded, the static parts of the page can be cached, while the dynamic parts – controlled by the content of the hashbang URI – get pulled in via JavaScript. Currently the extension is implemented in a way to run once upon page load, however, not upon Ajax refresh events on the page.

We overload Google Analytics event tracking[20] for the purpose of tracking named entities. We interpret a Google Analytics event as the occurrence of a named entity in a tweet never seen before. In its original sense, event tracking is meant to track on-page events, like, e.g., playing or stopping an embedded video on a page. The anatomy of an event is as follows: an event is defined by its `category` (req.), `action` (req.), `label` (opt.), and `value` (opt.). In our concrete case, the `category` is always "Entities", the `action` is any of the particular named entity's URIs (as they all represent the same thing, we simply choose the first URI of the list). The `label` is the particular entity's name and the sources of that named entity in parenthesis (as outlined in Section 3.1), and finally for the `value` we pass the timestamp of the moment where the entity extraction happened (which is closest to when we assume the tweet was read). A concrete call to log an occurrence of a named entity as injected onto Twitter.com by the content script looks like this (the `_gaq` array is the Google Analytics queue for asynchronous tracking):

```
_gaq.push(['twitter_swarm_nlp._trackEvent',
    'Entities',
    'http://dbpedia.org/resource/JSON',
    'json (zemanta,opencalais)',
    1299776578]);
```

Currently the extension displays all extracted named entities grouped by originating tweets and roughly in the tweet order at the righthand-column (see Figure 1 for a screenshot). In order not to flood the Twitter Swarm NLP Web service (and in turn the underlying third-party NLP Web services), the NLP requests get sent in intervals of one second. Dependent on the response time for each particular request, the extracted named entities get added to the righthand-column. Each named entity is displayed with its label, one representing URI (out of potentially several URIs), and the sources.

---

[19] See `http://www.jenitennison.com/blog/node/154` for a Twitter hashbang URI analysis.

[20] `http://code.google.com/apis/analytics/docs/tracking/eventTrackerGuide.html`

3.3   Avoiding Duplicate Tracking

Each tweet gets sent one-by-one to the Twitter Swarm NLP Web service, as described in Section 3.2. 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 does not falsify the overall 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 (we assume each tweet gets read at most one time). Twitter has introduced a new system to generate unique tweet IDs called snowflake[21], which guarantees k-sorted IDs[22] with at least a 1s bound. 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.

4   Evaluation And Discussion

In this section we first provide some raw statistics followed by an evaluation of our experiments, and discuss the approach and its criticisms in the second part. If not noted differently, all statistics are for February 24 to March 11, 2011.

4.1   Raw Statistics

The extension was live since February 24, 2011. It was announced via Twitter[23] on that day, and according to Bit.ly statistics[24] generated *102 clicks* till March 11 (see Figure 2(a)). If we compare these click statistics of the announcement link with the pageview statistics according to Google Analytics (see Figure 2(b)), we can see interesting patterns. In the Bit.ly statistics there are two spikes on February 24 and 26, in the prior case originating from initial clicks on the link in the announcement tweet plus same-day retweets, and in the latter case originating from delayed retweets from people who found the extension interesting after testing it, together with a second announcement tweet on February 26. The top usage spike according to Analytics is around February 28 to March 2, where on February 28 a new version of the extension was released via the Chrome Web Store, and also announced via Twitter. As of March 11 the extension reached *1,009 pageviews* as reported by Analytics, and had *35 all-time users* and *28*

---

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

[22] See http://portal.acm.org/citation.cfm?id=70413.70419

[23] http://twitter.com/#!/tomayac/status/40806491510276096

[24] http://bit.ly/eBsjQu+

seven-day active users. This translates to a rate of a 27.45% of seven-day active users per accumulated clicks on the announcement link. All in all, the extension has detected *1,533 unique different named entities* in total. Figure 2(c) shows the detailed distribution of the number of occurrences, e.g., 1,294 of all 1,533 unique named entities appeared only once, and only 12 named entities appeared more than five times (see Table 1 for the most frequent named entities). This can be interpreted as users reading about very diverse topics.
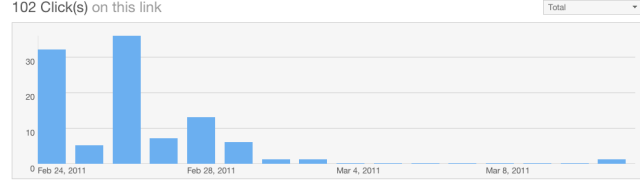
(a) 102 accumulated clicks on the announcement link as of Bit.ly.

(b) 1,009 absolute pageviews of the extension as of Analytics.
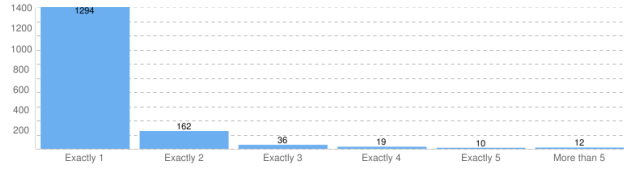
(c) Distribution of named entity occurrences as of Analytics.

Fig. 2. Raw statistics.

4.2   Top Read-about Named Entities

In this section we present some statistics on the top read-about named entities (consumers' view), which, again, does not necessarily correspond to the most tweeted-about named entities (producers' view, see Section 5.4). It is to be noted that our findings are at this point *not* statistically significant due to a too small number of seven-day active users of our extension (also see Section 4.5), however, nevertheless reveal interesting trends when the results are interpreted with a grain of salt. Table 1 shows the top 10 read-about named entities from February 24 to March 11. Looking at the data, we can see that the topic around the events in "Lybia" (rank 1 and 6) and its dictator "Gaddafi"[25] (rank 10) were of

---

[25] See `http://en.wikipedia.org/wiki/2011_Libyan_uprising` for a summary of the events.

high interest to the tweet readers. Rank 7 is taken by the named entity "iPad", which makes sense given the introduction of the iPad 2 on March 2, 2011[26]. Other named entities are not necessarily easily mappable to concrete events, however, the data suggests that "Twitter" (rank 2), "Google" (rank 3), and "YouTube" (rank 5) are companies that tweet readers show a general interest in, which seems reasonable. For the named entity "Linked Data" (rank 8), this is probably due to the social Twitter environment of the authors, but given the too small installation base is not representative. The named entity "glossary of graph theory" (rank 9) is definitely an incorrectly extracted named entity, as the underlying term was always "successor". We have no explanation for the popularity of "United States dollar" (rank 4), however, suppose that it is a correctly recognized, albeit relatively meaningless because out of context, named entity, as the underlying term was in all cases "USD".

Rank	Named Entity	Total Occurrences
1	http://dbpedia.org/resource/Libya	23
2	http://dbpedia.org/resource/Twitter	21
3	http://dbpedia.org/resource/Google	13
4	http://dbpedia.org/resource/United_States_dollar	10
5	http://dbpedia.org/resource/YouTube	10
6	http://rdf.freebase.com/ns/en/libya	10
7	http://dbpedia.org/resource/IPad	9
8	http://rdf.freebase.com/ns/en/linked_data	9
9	http://dbpedia.org/resource/Glossary_of_graph_theory	8
10	http://dbpedia.org/resource/Muammar_al-Gaddafi	8

Table 1. Top 10 read-about named entities (for rank 4, the original URL was http://d.opencalais.com/genericHasher-1/736a403d-157b-3e80-86a7-acc404607cb2)

4.3   Tracking Of Named Entities Over Time

Using Google Analytics, named entities (i.e., events in the sense of Google Analytics, as outlined in Section 3.2) can be easily tracked over time. As an example, Figure 3(a) shows the occurrences over time for the named entity "iPad". Albeit the numbers are not statistically significant, the peak of interest is on March 2, the day where the iPad 2 was revealed, and the second highest peak is on March 1, the day before the announcement, where media was atwitter with expectation of the device. Another example is given in Figure 3(b), which shows the occurrences of the named entity "tsunami". Japan was hit by an earthquake follwed by a tsunami on March 11, exactly where the peak is on the graph. Hence, the occurrence graphs indeed correspond to what we would expect.

---

[26] See http://en.wikipedia.org/wiki/Ipad_2#History for the details.
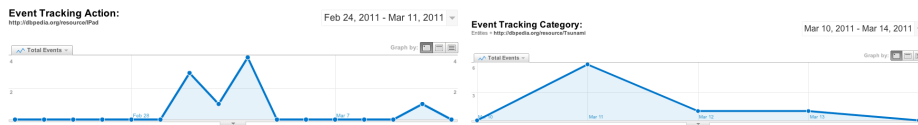
(a) Popularity of the named entity "iPad" (actually "iPad 2") over time (February 24 - March 11). The device was revealed on March 2, 2011.

(b) Popularity of the named entity "tsunami" over time (March 10 - March 14). Japan was hit by an earthquake followed by a tsunami on March 11.

Fig. 3. Popularity of named entities over time.

4.4 Top Read-about Named Entities Pivoted By Country

Japan was hit by an earthquake followed by a tsunami on March 11. As one would expect, this was reflected in the most read-about named entities for the period March 11 to March 14. The top 3 most read-about named entities for this period of time were "tsunami" (8 occurrences), "Japan" (7 occurrences), and "earthquake" (4 occurrences). Table 2 shows the occurrences distribution pivoted by country. Again, our data set is too small to be statistically significant, however, the potential for this data to reveal new insights is promising. Given enough data, we could, e.g., provide an answer to the question whether among Twitter users the tsunami caused more interest in the American, or the European continent. As we use URIs as named entity identifiers, there is no ambiguity, and no language barriers (the English word "earthquake" and the Spanish "terremoto" will both be represented by, e.g., `http://dbpedia.org/resource/Earthquake`).

Entity	Total	Germany	Finland	United States	Chile	India	Netherlands	Italy
`dbp:Tsunami`	8	3	2	1	1	0	1	0
`dbp:Japan`	7	2	3	1	0	0	0	1
`dbp:Earthquake`	4	2	1	0	0	1	0	0

Table 2. Top 3 most read-about named entities (March 10 - March 14) pivoted by country, sorted by 'Total". The prefix `dbp` stands for `http://dbpedia.org/resource`.

4.5 Criticism Of the Approach

The quality of the extracted named entities from tweets rises and falls with the quality of the Natural Language Processing step. Not being in the domain of NLP ourselves, we try to address the uncertain quality by using several NLP services in parallel. We found the approach to work well, with only few false results. As outlined in Section 3.2, we always track the source and entity name in the

`label` field of the overloaded Analytics event, which allows us to go back from a named entity's representing URI to the originating terms, which oftentimes already permits to judge the quality of the named entity and its relevance.

We found the biggest issue to be to get a critical mass of Twitter users to install the extension in order to get more reliable data. We were very open in our extension description that data was tracked via Analytics, which might have caused privacy-aware users to stay away from it. Twitter itself uses Google Analytics (as per the Twitter.com source code with the account ID "UA-30775-6"), however, states this fact only in a discreetly mentioned privacy policy[27] as required by the Google Analytics terms and conditions[28]. One could think of a similar approach for the extension.

Currently the user base also is still biased by the authors' social structure (more people have found the extension via Twitter than via the Chrome Web Store, which is the primary discovery channel for Chrome extensions). From a user's point of view, this is also due to the extension's primary purpose (display extracted named entities in tweets), which apart from users interested in the Semantic Web or NLP technologies is not an appealling reason for average Twitter users to install such extension. One could imagine bundling (hiding) the tweet analysis part of the extension with a more generic use case for a Twitter extension, like, e.g., a short URL unshorter, or a Twitter auto-refresher extension.

5   Related Work

As stated in Section 1, Twitter Trends is both an active field of academic and industry research, but also a playground for social media hobbyists. In the following we present four exemples from these categories.

5.1   Linked Open Social Signals (TWARQL)

In the Linked Open Social Signals project [4] 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 explicitly 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 authors maintain a client-side JavaScript application[29] allowing for users to search for tweets matching a customizable SPARQL query or to

---

[27] `http://twitter.com/privacy`

[28] `http://www.google.com/intl/en/analytics/tos.html`

[29] `http://knoesis1.wright.edu/twarql/query.html`

subscribe to tweet streams in a realtime "push" way, filtered according to the user's request (so-called concept feeds).

5.2   Twitris 2.0

With Twitris 2.0 [2], 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[30]. Using this set of search terms, related hashtags are then searched on Twitter 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 location hotspot a tag cloud with related tags is displayed, and the data can be sliced by time.

5.3   Semantic-MicROBlogging (SMOB)

SMOB [5] 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[31] and SIOC[32]. 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 the use of meaningful hashtags such as `#dbp:Eiffel_Tower`, or `#geo:Paris_France`, in the style of RDF prefixes for DBpedia 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[6].

5.4   Twopular

Twopular[33] 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 to 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.

---

[30] `http://www.google.com/insights/search/`

[31] `http://www.foaf-project.org/`

[32] `http://sioc-project.org/`

[33] `http://twopular.com/`

6   Future Work And Conclusion

In this paper, we have suggested a solution for obtaining a consumers' point of view on Twitter trends. Rather than measuring the "trendiness" of terms/hashtags in tweets being produced, we measure the "trendiness" of named entities in tweets being consumed using Google Analytics. This is not only interesting *per se*, but together with the "by-product" of classical Web analysis allows for even richer insights into, e.g., the location of users interested in a certain trend, sliceable back to any point (or period of time) in history where there is data available.

Future work can go in different directions. First, it is desirable to reach a broader user base through bundling the data analysis part of the extension with a more generic Twitter extension that solves a broader use case. Second, parts of the collected data could be fed back to the users (top n read-about entities during the last x hours in a *region*). Third, the commercial potential of the collected data is promising, albeit regionally different privacy laws may apply that prevent use of such data. There are potential new social media analysis tools that could be based on tweet consumers' points of view on Twitter trends. We have barely scratched the surface of what is possible.

References

1. 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.
2. 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`.
3. 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`.
4. 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.
5. 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`.
6. 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.