Ushio: Analyzing News Media and Public Trends in Twitter

Fangzhou Yao, Kevin Chen-Chuan Chang, Roy H. Campbell
Department of Computer Science
University of Illinois at Urbana-Champaign
{yao6, kcchang, rhc}@illinois.edu

***Abstract*—In this information age, Social Networking Services contribute a significant amount of contents in creating a knowledge based society. Nowadays, there are more than 500 million tweets sent per day in Twitter. Such drastic growth of contents brings new opportunities for human beings to discover their surroundings more effectively in a timely manner. Moreover, these types of services evolve not only in a perspective of scalability, but also in the view of indicating more meaningful information regarding what happens in the world. Numerous news agencies are broadcasting breaking news via Twitter and people would like to leave comments with their own opinions as well. However, there are differences between events that news media are more willing to cover and news stories that people are more interested in. Furthermore, as people are becoming the largest sensor network, trending topics are not only led by media, but also by the public, and hence it is worth pondering how they affect each other. In this paper, we focus on studying these concerns by building a system, Ushio, analyzing Twitter streams in both the tweets updated by multiple news agencies and those appearing in the public timeline. We describe our design and implementation of this system, which extracts named entities from the Twitter streams and generates corresponding statistics with its relational model. We then show how we use these data to find trending topics and real focus from both media and the public, as well as discover their related topics along with the correlation indicating the leading role between them for assorted topics.**

I. Introduction

We are living in an age of information, and our world is essentially characterized by the information around. It is important to think about those data, since it plays a crucial role for people to understand this world. Social Networking Services (SNS) have contributed a significant amount of such contents in building this knowledge based society. In 2014, Facebook reaches more than a billion active users, while the world population is around 7.1 billion [2]. Twitter, a microblogging service founded in 2006, now has more than 255 million active users [3].

As more and more people are connected by various types of SNS, the services themselves evolve correspondingly. There are more than 500 million tweets sent per day in Twitter nowadays [4], while the number was only 200 million back to 2011 [5]. Such drastic growth of contents brings new opportunities for people to discover surroundings more effectively in a timely manner, as there would be sufficient amount of data for analysis even in a short time interval.

Moreover, the evolution happens not only in the perspective of scalability to accommodate and analyze much more data, but also in the view of indicating more meaningful information about this world, as the hashtags and geographical information are embedded in tweets and Facebook statuses. There has been researches regarding whether Twitter is still purely a service, which is full of people's updates about their daily life or it is essentially becoming a new type of news media [16], which focuses on more comprehensive topics about what is happening in this world. Lots of news agencies are broadcasting their news via Twitter nowadays, and people would like to participate in these discussions as well. Also, there are many web services including Twitter itself listing trending topics everyday. However, the trends provided by those services do not differentiate which topics are covered more by news media, and which are discussed more by the public. There are still large differences between the events that news media are more willing to cover and stories that people are really interested in.

Furthermore, it has been shown that people are becoming the largest "sensor network" [18], and hence trending topics are not only led by news media anymore, but also by the people. As public opinions are more easily to express and becoming more important, it brings an interesting question to take into account – what kinds of topics are initiated by news media, and what are motivated by the public?

Therefore, in this paper, we propose Ushio to study these concerns and address the above questions with our analysis. Ushio is a system analyzing Twitter streams in both the tweets updated by news agencies and those appearing in the public timeline. This system extracts named entities from tweets in real-time and stores them into a relational data model. Users can then use the query handler or traditional interactive queries to generate statistical data and find trending topics. It also makes it possible to discover related topics from those trending topics, as well as indicate the correlation of the leading role of topics between them, which also reflects media's focuses and people's interests.

The rest of this paper is organized as follows. We first explain our motivation of proposing Ushio in Section II, and then describe the strategy that we use to extract the meaningful data from both media and the public in Section III. Next, we show our design and implementation in Section IV. Furthermore, we evaluate our system through experiments in Section V. Finally, in Section VI, we conclude our work and discuss

future approaches.

II. A Mashed Up But Curated World

The idea of Web 2.0 was introduced more than a decade ago, and User-Generated Content (UGC) has already become the mainstream in today's Internet. As a major platform of UGC, SNS makes its impact to various traditional fields including news media. The significant changes in this field could be concluded as the mash up from various sources, and curated selection of stories with trending topics.

Mashable, a news blog launched in 2005, now has more than 13 million monthly page views [7]. Its primary focus is social media news. Many of its top news stories are oriented by trending topics from SNS, such as Twitter, while editors also cite those trending tweets as sources in their news. Flipboard, an application described as a social magazine, offers content pages generated from data in people's private Facebook and Twitter feeds, as well as assorted sources on specific topics, such as politics and technology from the public timeline of many SNS [8]. Similarly, Pulse, a mobile application originally released in 2010 and acquired by LinkedIn in 2013, offers a mash up of news for users from various sources. The integration of Pulse and LinkedIn also brings its users a more relevant news experience with contents tailored for their professional interests [9].

In 2014, Yahoo presented its news summarizing application, Yahoo News Digest [10], which summarizes top news from different sources, like multiple news agencies, Twitter and Wikipedia. Also, it presents users no more than 10 top stories a time, twice a day, which are curated based on the trending topics on that day. In the same year, The New York Times also released its NYT Now mobile application, which functions similarly as Yahoo's presenting a collection of selected stories, but focusing only on news from its own source [12]. This move marks that traditional news media are changing to meet people's needs in this age. Though most of the NYT Now stories are based on the selection from its editors, there has been several researches focusing on automatically generating digests with given keywords from SNS like Twitter [17] and finding related topics from the web [19].

Therefore, it is obvious to observe that the way to broadcast news has changed greatly. People welcome news services that are tailored for them and comprehensive from different sources, so discovering the correlation between what the news media cover more and what people are really interested in becomes important. Furthermore, since information is increasingly being distributed and presented in real-time streams instead of dedicated sources [11], which include the static web pages, obtaining trending topics in time becomes challenging.

Thus, we would like to build a system that monitors informative streams from both news media and the public, which is able to extract meaningful data and use it to analyze trends in real-time. Moreover, discovering the correspondence between the focuses of news media and people, as well as the leading roles in assorted topics, would be beneficial to help media in building a more relevant news coverage.

III. How Can We Collect Good Data?

This section describes the considerations made to define what good data should be to achieve our objectives. We also show our solution in finding such kind of data.

A. Aggregation and Statistics

Our approach is to generate statistics for various time periods to show trending topic in their corresponding time intervals, which can be as precise as in a few minutes, and as comprehensive as in years. Thus, we tend to collect time-sensitive data that is able to be aggregated.

There has been many researches on discovering trending topics in Twitter. One of them trains a Naive Bayes classifier [28]. However, it takes time and needs a significant amount of pre-labeled data to train such a classifier. Another one uses bursty keywords in tweets to develop a group-burst solution to detect trends [27], while some other one uses Term Frequency - Inverse Document Frequency (TF-IDF) weighting as normalized terms to conduct frequency analysis for trending topics [21]. We would like to simplify this process for aggregation, as users might request a series of results for trending topic in different time intervals.

If we take a simple model for data aggregation into consideration, the tag cloud, we would be able to simply this process in a better way. Tag cloud is a famous visual representation for text data, which is typically used to visualize keywords on websites based on their importance [15]. The keywords are usually single words and their importance mainly depends on the times that they appear in the text. Thus, we can use similar strategy to count extracted keywords from tweets, and hence the more times the same keyword is collected, the more important it would be. When a keyword is extracted, a timestamp is attached to represent the time that it is collected, which also represents the time that a topic is mentioned. The aggregation of identical keywords takes their timestamps in reference, and hence the trending topics could be then calculated for any time intervals.

B. Meaningful and Fine-Grained Entities

We need an effective approach to identify keywords from tweets. The most significant characteristic of a keyword is that it should be meaningful enough and represent the main idea of a sentence. In our system, we focus on English tweets.

Many solutions have been used in data oriented web page searching [19], but for a rather simple data type, the tweets, we decide to use Named Entity Extraction (NER) as our approach, which is turned out to be more performant [22]. Named entity is a type of informative keywords that has been classified into pre-defined categories, such as persons, organizations and locations.

There has been approaches proposed for extracting named entities in Twitter. However, many of them require training for their specific classifiers with certain data sets [23], [24], [25]. This fact results in focusing hashtags too much, which are not often used by news agencies in their updated statuses. Some

also need external tools deployed in the cloud [26], but we prefer keeping our system as simple as possible.

Thus, we would like to use a general NER framework [22] with good performance and accuracy. It should be able to provide fine-grained types for detailed analysis.

C. Data Reliability

Another factor that we take into account is the reliability of collected data. We acquire data from Twitter in two different approaches. One is the Twitter stream containing only tweets from multiple news agencies, and the other is the public timeline consisting of English tweets from all over the world.

On one hand, all Twitter accounts from news agencies that we followed are Twitter verified accounts. Thus, there should be no forged news unless a news agency intends to do so. Even if there is a small amount of fake data, it will not reflect in the final statistical results and hence this potential problem should not become a nontrivial concern. We believe that the data extracted from the majority of news agencies should be genuine, since all news agencies that we selected to follow are the ones with relatively good reputation. Though we followed some Twitter accounts of foreign news media updating in English, we are aware of the fact that most of them tend to report news in their own languages. As we described in previous subsections, we only extract named entities from English tweets and hence this approach might be a limitation for the data set that we are able to collect. However, many breaking events in the world, which eventually become trends for discussions, are also reported frequently by news agencies like CNN and BBC, where English is used as the main language. In consequence, we assume that the eventual size of data set should be sufficient, when we analyze the data acquired for world-wide trending topics.

On the other hand, it might be a problem that we also collect data from the public timeline, since individuals are essentially our data sources in this approach, who make their own observations about the physical world. These observations could be false, and hence it might be misleading. Apollo is one of the projects that focus on the reliability of data collected from Twitter [18]. It develops a model that considers the impact of false information using a maximum-likelihood strategy. However, our system collects over 40 thousand named entities from the public timeline per day in average, as it is shown in Section V, where fake information could be then only a very small portion. Furthermore, we focus on named entities and they are not binary decisions. Thus, it simplifies the process to verify the reliability of the data, and hence we believe that the eventual statistics could represent the trustworthy trends from the public.

In summary, our solution is to use a general NER framework to extract named entities from Twitter streams and store them into a relational model. Thus, we are able to use the data to generate statistics for any requested time intervals. In Section IV, we explain how we design the relational model to make it possible to find relations between named entities.

IV. Design and Implementation

In this section, we show our design for Ushio and our specific decisions for our implementation.

A. Architecture

The design of Ushio has two major components. Figure 1 shows the overview of this system. The first component is our set up on Twitter side. The second component is the application deployed on our machine. It monitors Twitter streams from our account following multiple news media and the public timeline. It also sends the tweet content to the NER framework through a Ruby Java bridge, and parses the extracted named entities with their timestamps into the database, which consists of two tables for the ones from the news media and the public, respectively. Users can interact with the query handler or directly with the database to create corresponding statistics for various types of analysis.

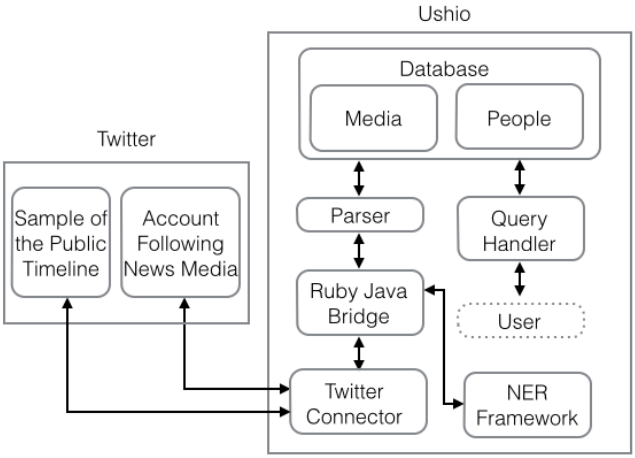

Fig. 1. A High Level View of Ushio.

Though the system is written in Ruby due to its affinity with Twitter and relational databases, the NER framework is written in Java. Thus, we used a Ruby Java bridge to initiate and communicate with the NER framework running in Java Virtual Machine (JVM).

B. Data Collection

We monitors the tweets in a streaming fashion, and hence we need to guarantee that both obtaining data from Twitter API and extracting named entities from tweets are in real-time.

1) Twitter Streaming API: The key requirement of a real-time operation is the ability to ingest content rapidly. Also, the system needs to make its content available immediately, while keeping concurrency, low-latency and high-throughput [20]. Twitter Streaming API is designed to meet these demands [6].

We used the public streaming API to monitor data from the public timeline flowing through Twitter. We did not assign any topic or keyword to the API, and hence it returned us a random sample of all public statuses. Twitter did not have an API for full public timeline streams, since people might need

a roughly identical level of the size of computation clusters as Twitter's to handle that much data. This fact makes it much less useful and unnecessary. Moreover, the public streaming API already satisfies our requirements as it is mentioned in Section V.

As for the news media tweet stream, we created an account `@ushioapp` and followed 115 news agencies. We are able to obtain the complete tweets in real-time with Twitter's user streaming API.

2) Name Entity Recognition: We used Stanford NER framework [13], which achieves an accuracy of 95% in extracting named entities from texts [22]. It is implemented in Java, and hence we used the Ruby Java bridge gem handling all the tasks related to JVM [14], including starting the JVM with specific memory allocation, passing data into a specific Java class and retrieving its outputs.

To guarantee the time bound of named entity extraction from every tweet, since this process should be also in real-time, we terminate any extraction process, if it is not finished in 50ms. As it is mentioned later in Section V as well, we did not discover any termination and the extraction were finished in an average time of 4.00 ms.

In addition, this framework tags every word with its possible properties, such as `PERSON`, `ORGANIZATION`, `LOCATION` and `MISC`. It incurs a problem that phrases like *John Smith* would be parsed into two entities, *John* and *Smith*. This result is definitely not what we want, because it breaks the meaning of the original phrase and might result in misleading aggregation. Thus, we check the entity array parsed from the result and determine adjacent entities of the same type to be one single entity.

C. Data Storage and Analysis

We store our data in a relational model in the database, and hence users can generate statistics per need by using our query handler or directly query from the database. In later production, we will disable the feature to directly invoke the database bypassing the handler for security.

1) Database Schema: The database consists of two tables for named entities parsed from news media and the public timeline. Tables are named as `Media` and `People`, respectively. They are created with the same schema. The tuple has four columns: `entity`, `type`, `time` and `tweet_id`. They represent the content of the named entity, as well as its type, timestamp and original tweet ID correspondingly.

We did not record the original tweet, instead we store the tweet ID, where named entities were extracted. The reason is two-fold. First, we want to minimize the capacity of data collected for later deployment in a web server, or containers like Heroku. Thus, keeping only tweet IDs becomes a better option than keeping the tweets, since each tweet has its unique ID and we can always trace back to the original tweet. Moreover, a tweet ID is a 18-digit integer, which takes much smaller space than 140 characters in a tweet. Second, we keep this property because all named entities extracted from a single tweet would have the same ID, which helps us find the relations between them.

2) Query Handler: We implemented a query handler with pre-made queries to generate statistics used frequently.

```
SELECT entity, count(*) as count
FROM social.People
WHERE time > $a and time < $b [and type = $t]
GROUP BY entity ORDER BY count DESC;
```

Listing 1. Finding Trending Topics

List 1 indicates how we find trending topics from the public. Trending topics could be found by executing similar queries in table `Media`. It ranks the result by aggressing entities in the given time frame from a to b with the optional entity type t.

```
SELECT social.People.entity AS name_entity, count
    (*) as count FROM social.People
WHERE tweet_id IN
  (SELECT social.People.tweet_id FROM social.
      People
  WHERE entity = $e and time > $a and time < $b)
GROUP BY entity ORDER BY count DESC;
```

Listing 2. Finding Related Topics

List 2 indicates how we find related topics given a keyword e. The keyword is usually one of the trending topics. It ranks the result by aggressing entities in the given time frame from those having the same tweet IDs where e also appears.

In addition, users can bypass this handler to execute their own queries. However, this feature will be disable after the deployment in the network for public usage.

V. Evaluation

In this section, we show our experiment environments and basic metrics. We then discuss our data along with the reference of recent news stories.

A. Basic Facts

We conducted our experiments on OS X 10.9.2, MySQL 5.6.17, Java 1.7.0 update 51 and Ruby 2.1.0p0 on a Mac with Intel Core i7 2.93GHz CPU and 32GB memory. The average time to extract named entities from a tweet in this system is around 4.00 ms. The time zone for our experiments is Pacific Daylight Time (PDT).

By May 16th, 2014, we collected 121,226 named entities from the news media, and 11,727,621 named entities from the public timeline, which illustrates that we are able to obtain enough size of data from Twitter's public streaming API. The size of the data set is 780MB. Considering that we started collecting data from Apr 19th, the entities collected is around 30MB per day in average. This fact proves our design to minimize the capacity of data. The data set mentioned in this paper is available and can be downloaded at [1], and it could be directly imported into a MySQL database.

B. Results and Analysis

In this subsection, we explain how we find trending topics for news media and the public and their related topics, as well as discover the differences between these two types of data. In addition, when we discuss an entity in this subsection, its corresponding rank and count are mentioned as "`entity:rank(count)`".

1) What An Exciting Week!: Table I shows the top 10 trending topics from both the news media and the public in the week starting from April 28th to May 4th, 2014. We did not use the types for categorization in this ranking. In addition, we manually removed `RT:1(525)` in the media ranking, since the keyword for retweet is used frequently by news agencies and extracted as an named entity by mistake. The same things happens to `-LRB-:3(10216)` in the public ranking, as we thought that it was actually meaningless after investigation and might be a tag pollution.

Rank	Media	#	Public	#
1	Ukraine	462	Chelsea	11282
2	China	369	EU	10524
3	Donald Sterling	363	God	9913
4	Obama	287	Tribez	9625
5	NBA	282	Justin	9132
6	US	281	Argentina	8848
7	Russia	220	Donald Sterling	8788
8	Clippers	173	Best	8586
9	Apple	168	NBA	6790
10	Oklahoma	153	London	6293

TABLE I
THE WEEK OF 04/28 - 05/04, 2014

We can observe that media are more willing to cover politics, especially the Crimea Crisis during this specific week. Topics are ranked as 1 and 7, while in the public, we only have results showing `Ukraine:40(3145)` and `Russia:67(2165)`. We are aware of the fact that not all entities contributing to the count for *Russia* necessarily refer to Crimea Crisis, but it shows the trend if we also look at the similar count for *Ukraine*. As the national election in India is undergoing, we can see `Modi:13(152)` and `India:14(150)` presented by media, but only `Modi:79(1932)` and `India:33(3516)` by the public. This fact indicates that people have less interest in politics compared with other fields.

In the public trend, sports teams are leading the board because of recent games like English Premier League and Champions League Final. As the World Cup is approaching, we have `Brazil:38(3320)`. `Justin:5(9132)` refers to Justin Bieber mostly after our investigation, and it is the common phenomena that the public is interested in entertainment, especially gossips and scandals. Thus, when scandals meet sports, it becomes the trending topics in both media and the public, as we can find about *Donald Sterling*. In the technology perspective, *Apple* is ranked as 9 in news media trends, but we have `Google:36(3411)` and `Apple:39(3179)` for the public, which indicate that the public do not care about it so much and still might care it more than politics. Besides, *God* are `Jesus:32(3584)` are always mentioned a lot in the public trend.

2) Why Do They Care About It So Much?: Figure 2 shows related topics on April 29th, given the keyword *Microsoft* and *NBA*. The keywords and their related topics are shown in blue and green, respectively. The figure of counts from the media is shown on the left, while the one from the public is on the right.

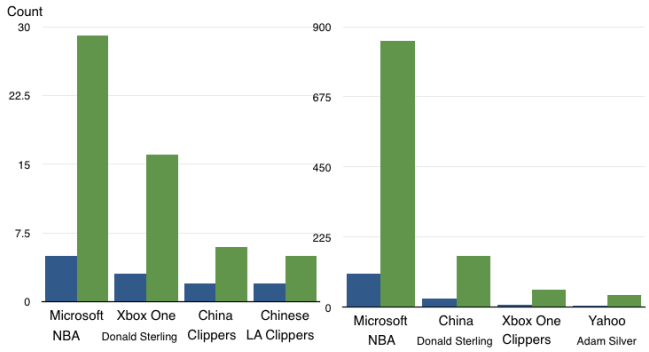

Fig. 2. Related Topics Counts

We find the topic about *Microsoft*, because we discovered that it was ranked higher than usual, and hence we looked for its related topics, which revealed that Microsoft was starting selling *Xbox One* in *China*. The *NBA* racism talk scandal by *Donald Sterling* and *Clippers* was the trend around that day, and hence all counts for these entities overwhelmed the ones about *Microsoft*.

3) Who is the Winner?: Figure 3 shows correlation between media and the public by showing the `PERSON` type entities ranking of *Donald Sterling* from April 26th until May 7th.

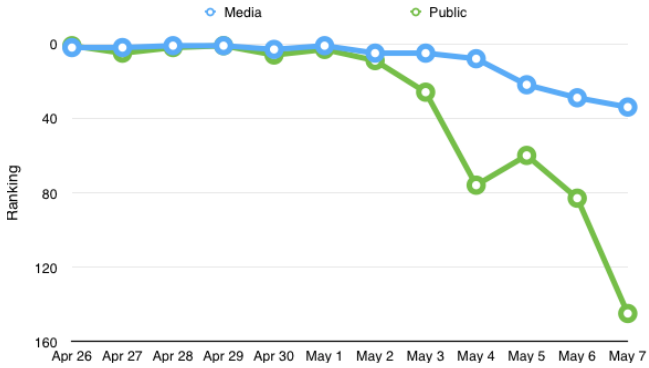

Fig. 3. Correlation on Media and the Public

For this typical event of sports and scandals, the public essentially leads. We can observe that it is the public who lose passion on this topic first, and then news media reduce their related reporting. We can find the difference, especially between May 3rd and 5th, that the ranking in the public dropped greatly, but media did not stopping their coverage so much. Moreover,

in the first few days of this scandal, media covered it more than usual in order to meets people's needs, though people became aware of this event because of the news at the very first day. Our analysis on similar topics including *Justin Bieber* indicates the same conclusion. However, political topics like *Modi*, the public trending explodes after the news announces his election to the Prime Minister of India, and hence media lead in this type of topics. This observation is very similar with the interests from two different sources discovered in previous subsections, though we do not show those data due to the limitation of length.

VI. Conclusions and Future Work

In this paper, we show our design and implementation of Ushio, a system monitoring Twitter streams from both media and the public. It is able to extract meaningful data from tweets in real-time and store them into a relational model for analysis. We demonstrate that our solution is able to find trending topics as well as their related topics. We show that our system is able to help people discover the correlation of the leading role between them, which also reflects news media's focuses and people's interests.

In the future, we plan to conduct more experiments on assorted topics to make our conclusion more sound. Also, we plan to deploy this system with a visualization interface for public accesses and add segregation based on geographical information in tweets.

VII. Acknowledgments

This material is based on research sponsored by the Air Force Research Laboratory and the Air Force Office of Scientific Research, under agreement number FA8750-11-2-0084.

References

[1] F. Yao, K. C. Chang, and R. H. Campbell, *Data Set for This Paper*, [Online]. Available: http://goo.gl/ihNeuD

[2] D. Murphy, *Facebook Tops 1B Monthly Active Users*, [Online]. Available: http://goo.gl/8HcndP

[3] E. Protalinski, *Twitter Passes 255m Monthly Active Users*, [Online]. Available: http://goo.gl/GMu8xX

[4] R. Krikorian, *New Tweets Per Second Record, and How!*, [Online]. Available: http://goo.gl/PJbsYB

[5] Twitter Engineering, *200 Million Tweets Per Day*, [Online]. Available: http://goo.gl/bZQLcS

[6] Twitter Engineering, *The Streaming APIs*, [Online]. Available: http://goo.gl/H87PZ1

[7] K. O. Prior, *Where the Connected Get Clued In*, [Online]. Available: http://goo.gl/TpRBjQ

[8] S. Richmond, *Flipboard: the Closest Thing I've Seen to the Future of Magazines*, [Online]. Available: http://goo.gl/Z6xUg

[9] A. Gupta , *LinkedIn and Pulse Integration: Professional News Tailored to You*, [Online]. Available: http://goo.gl/U4D6j8

[10] C. Newton, *Yahoo's Sleek News Digest App Swims Against the Stream*, [Online]. Available: http://goo.gl/hQMuZ9

[11] A. C. Madrigal, *2013: The Year The Stream Crested*, [Online]. Available: http://goo.gl/a8h670

[12] The New York Times Company, *Introducing NYT Now*, [Online]. Available: http://goo.gl/0Khc5y

[13] The Stanford Natural Language Processing Group, *Stanford Named Entity Recognizer*, [Online]. Available: http://goo.gl/7i62No

[14] Hiki Development Team, *Ruby Java Bridge*, [Online]. Available: http://goo.gl/DzSJ

[15] M. Halvey, and M. T. Keane, *An Assessment of Tag Presentation Techniques*, [Online]. Available: http://goo.gl/6YVFTl

[16] H. Kwak, C. Lee, H. Park, and S. Moon, *What is Twitter, a Social Network or a News Media?*, 19th International World Wide Web Conference, 2010.

[17] M. T. Amin, Tarek Abdelzaher, Dong Wang, Boleslaw Szymanski, *Crowd-sensing with Polarized Sources*, 10th IEEE International Conference on Distributed Computing in Sensor Systems, 2014.

[18] D. Wang, M. T. Amin, S. Li, T. Abdelzaher, L. Kaplan, S. Gu, C. Pan, H. Liu, C. C. Aggarwal, R. Ganti, X. Wang, P. Mohapatra, B. Szymanski, and H. Le, *Using Humans as Sensors: An Estimation-theoretic Perspective*, 13th ACM/IEEE International Symposium on Information Processing in Sensor Networks, 2014.

[19] M. Zhou, T. Cheng, and K. C. Chang, *Data-oriented Content Query System: Searching for Data into Text on the Web*, 3rd ACM International Conference on Web Search and Data Mining, 2010.

[20] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. Lin *Earlybird: Real-Time Search at Twitter*, 28th IEEE International Conference on Data Engineering, 2012.

[21] J. Benhardus, *Streaming Trend Detection in Twitter*, International Journal of Web Based Communities. Vol.9(1), 2013.

[22] J. R. Finkel, T. Grenager, and C. Manning, *Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling*, 43rd Annual Meeting on Association for Computational Linguistics, 2005.

[23] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B. Lee, *TwiNER: Named Entity Recognition in Targeted Twitter Stream*, 35th Annual International ACM SIGIR Conference on Research and Development on Information Retrieval, 2012.

[24] A. Ritter, S. Clark, Mausam, and O. Etzioni, *Named Entity Recognition in Tweets: An Experimental Study*, Conference on Empirical Methods in Natural Language Processing, 2011.

[25] X. Liu, S. Zhang, F. Wei, and M. Zhou, *Recognizing Named Entities in Tweets*, 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011.

[26] T. Finin, W. Murnane, A. Karandikar, N. Keller, and J. Martineau *Annotating Named Entities in Twitter Data with Crowdsourcing*, NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon?s Mechanical Turk, 2010.

[27] M. Mathioudakis, and N. Koudas, *TwitterMonitor: Trend Detection over the Twitter Stream*, 2010 ACM SIGMOD International Conference on Management of Data, 2010.

[28] D. Irani, S. Webb, C. Pu, and K. Li *Study of Trend-Stuffing on Twitter through Text Classification*, 7th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference, 2010.