Identification and Ranking of Event-specific Entity-centric Informative Content from Twitter

Debanjan Mahata[1], John R. Talburt[1], and Vivek Kumar Singh[2]

[1] University of Arkansas at Little Rock, Department of Information Science, USA
[2] South Asian University, Department of Computer Science, New Delhi, India
dxmahata@ualr.edu, jrtalburt@ualr.edu, vivek@cs.sau.in

Abstract. Twitter has become the leading platform for mining information related to real-life events. Not, all tweets are useful and informative. A large amount of the shared content are non-informative spams and informal personal updates. Thus, it is necessary to identify and rank informative event-specific content from Twitter. Moreover, tweets containing information about named entities (like person, place, organization, etc) occurring in the context of an event, generates interest and aids in gaining useful insights. In this paper, we develop a generic model based on the principle of mutual reinforcement, for representing and identifying event-specific, as well as entity-centric informative content from Twitter. A novel algorithm is proposed that ranks tweets in terms of event-specific, entity-centric information content by leveraging the semantics of relationships between different units of the model. Experiments and observations are reported on tweets collected for two real-life events, and evaluated against popular baseline techniques.

1 Introduction

Twitter is a social media platform that has become an indispensable source for disseminating news and real-time information about current events. It is a microblogging application that allows its users to post short messages of 140 characters known as tweets. Twitter is widely accepted as a source for first-hand citizen journalistic content and has been harnessed in detection, extraction and analysis of real-life events [19,17,18].

A significant amount of tweets in Twitter are related to real-life events (e.g, football matches, music shows, etc). Majority of these event related tweets are pointless babbles, personal updates and spams providing no information to the general audience interested to know about an event. On the other hand there are tweets that presents newsworthy content, recent updates and real-time coverage of on-going events. These tweets are informative and are very useful for users who follow an event, and search for related information in Twitter.

Occurrence of a real-life event in general is characterized by participation of entities like people, organizations, or things at a certain place over a period of time [21]. While sharing information about an event in Twitter, users often mention these entities (e.g *Update: Statement from Australian Prime Minister Tony*

Abbott on the Hostage incident #SydneySiege http://t.co/b4tO4A8CQj). We consider such user updates as entity-centric messages related to the event. The consumers of event related information are most often interested in such entity-centric messages in the context of the event. Also, informative content shared about the entities during an event helps in gaining useful insights about the event as well as the related entities.

The main objective of the work presented in this paper is to automatically identify and rank event-specific informative tweets mentioning relevant entities in their content. Towards this objective, we propose a generic model based on principle of mutual reinforcement for representing relationships between event-specific information cues and relevant named entities extracted from the tweet content. We develop a novel algorithm that leverages the mutually reinforcing relationships represented by the model for ranking tweets in terms of event-specific informative content sharing information about entities related to the event. Finally, we evaluate the ranked results against popular baselines, and report the effectiveness of our algorithm in identifying and ranking event-specific informative tweets discussing about event related entities.

2 Related Work

In this section we review existing works related to the ranking of information content in tweets. There are many web based platforms used for searching and retrieving information shared in Twitter [3]. Recency of tweets, popularity based on retweet counts, authority of users and content relevance are the dominant factors used for ranking, in these platforms. A study of different state-of-the-art features commonly used for ranking tweets has been documented by [5]. Seen[4] is a new state-of-the-art platform that uses a proprietary algorithm named *SeenRank* for ranking tweets in terms of event-specific information content for presenting event highlights and summaries to its users. In this work, we consider *SeenRank* as one of our baselines and compare our ranking with it. Apart from the existing real-world search applications, several adaptations of *PageRank* [16] has been proposed by the scientific community for ranking tweets and users in Twitter [25,22]. TweetRank [8] is one such adaptation that ranks tweets by taking into account the direct relationships between tweets in the form of retweets and replies, as well as indirect follower-friend relationships, and usage of similar hashtags. Various learning to rank approaches have been used for ordering tweets retrieved for a given query in terms of their relevance and quality [6,13,23].

Recently researchers have shown interest in microblog summarization. One of the most important part of any summarization framework is to identify the salient posts. Experiments have been conducted using both feature-based and graph-based approaches. However, in the context of our work only graph-based approaches are relevant. A comparison of different Twitter summarization algorithms was performed by [9]. The *phrase graph* algorithm [20] is the most frequently used graph-based approach in microblog summarization. Summarization of tweets for sporting events was performed by [15] using the phrase graph algorithm. Other popularly used graph-based summarization algorithms are *LexRank* [7] and *TextRank* [14].

Our objective is more aligned with techniques used for ranking tweets. We choose our baselines accordingly, as discussed in the *Experimental Settings and Evaluation* section. To our knowledge the framework and algorithm we propose is novel and is

[3] http://mashable.com/2009/04/22/twitter-search-services/

[4] http://seen.co

the first attempt to understand the semantics of relationships between event-specific information cues in Twitter for implementing a graph-based algorithm that ranks event-specific informative content discussing about different entities related to the event.

3 Problem Statement

In this section, we give the definition of an *event* appropriate in the context of our problem, and then present a formal statement of the problem that we want to solve.

Events have been defined from various perspectives and in different contexts. In the context of our work we adopt a definition similar to [3]. An **event** is defined as a real-world occurrence (E_i) with an associated time period T_{E_i} ($t^{start}_{E_i}$-$t^{end}_{E_i}$), and a time ordered stream of tweets M_{E_i}, of substantial volume, discussing the occurrence of the event and posted in time T_{E_i}. While, discussing about the event, the users comment and talk about entities (person, organization, place, facility, etc) relevant to the event. Our aim is to identify and rank event-specific informative tweets that not only shares information about the event but also informs about entities related to it.

Problem: *Given an event E_i, a time ordered stream of n tweets $M_{E_i} = \{m_1, m_2, ..., m_n\}$ related to the event posted in time period T_{E_i}, the problem is to find a ranked set of tweets $\hat{M}_{E_i} = \{m_1 \geq ... \geq m_i \geq m_j \geq ... \geq m_n \mid i < j\}$, ordered in decreasing order of its event-specific informative content sharing information about event related entities.*

4 Methodology

Twitter allows its users to post short messages with a limitation of 140 characters. Users not only post plain textual content in their messages but also share urls, linking to other external websites, images and videos. Apart from curating new content, the users also share content produced by others. This activity is known as *retweeting*, and such tweets are preceded by the special characters *RT*. The messages are normally written by a single person and are read by many. The readers in the context of Twitter are known as *followers*, and the user whom the other users follow is considered as their *friend*. Any user with good intent either share messages that might be of interest to his followers, or for joining conversations on topics of his interest. The '@' symbol followed by the username commonly known as *user mentions*, is used for mentioning other users in tweets for initiating conversation with them.

The concise and informal content of a tweet is often contextualized by the use of a crowdsourced annotation scheme called *hashtags*. Hashtags are a sequence of characters in any language prefixed by the symbol '#' (for e.g. #nldb2015). They are widely used in order to add context to the tweets, categorizing the content based on a topic, join conversations related to a topic, and to make the tweets easily searchable by other interested users. They also act as strong identifiers of topics [12]. When tweeting about real-life events the users also tend to use hashtags in order to post event-specific content.

Based on the mechanism of user interactions and content production in Twitter we assumed that the content of a tweet is primarily composed of hashtags, words for expressing and conveying information, and urls that lead to additional information about the content. While conveying information about an event the users also mention named entities in the textual content of the tweets. For example, the tweet *Update: Statement from Australian Prime Minister Tony Abbott on the Hostage incident #SydneySiege* *http://t.co/b4tO4A8CQj*, not only provides information about the Sydney Siege crisis event, but also informs about "Tony Abbott" in the context of the event. The tweets are posted by users. It is also intuitive that users having high follower count tends to post informative posts, as tweets posted by such users are read by a larger audience. Also, it might be that since they share informative content, they are followed by large number of other users. Therefore, for an event E_i, in order to identify and rank event-specific informative tweets discussing about entities relevant to the event, we consider the following as event-specific information units:

- a set of *hashtags* ($H_{E_i}=\{h_1,h_2,...,h_p\}$) used for annotating the tweets.
- a set of *entities* ($W_{E_i}=\{w_1,w_2,...,w_r\}$) mentioned in the tweets.
- a set of *users* ($U_{E_i}=\{u_1,u_2,...,u_s\}$) posting tweets ($\in M_{E_i}$) about the event.
- a set of *urls* ($L_{E_i}=\{l_1,l_2,...,l_t\}$) linking to external sources related to the event

The above information units might be independent, but in the context of an event, they are not independent of each other in presenting informative content. The informativeness of any information unit depends upon its occurrence with other information units. We define the informativeness of each unit based on the assumptions below. For an event E_i

- a *tweet is considered to be informative* if it is strongly associated with: **(a)** *informative hashtags*, **(b)** *informative entities*, **(c)** *informative users*, **(d)** *informative urls.*
- a *hashtag is considered to be informative* if it is strongly associated with: **(a)** *informative tweets*, **(b)** *informative entities*, **(c)** *informative users*, **(d)** *informative urls.*

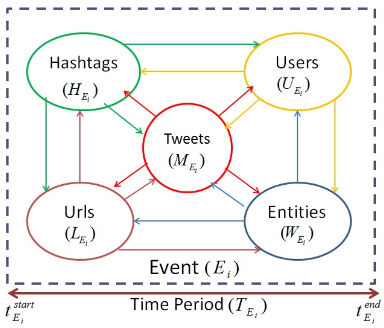

Fig. 1. Mutual Reinforcement Chains in Twitter.

- an *entity is considered to be informative* if it is strongly associated with: **(a)** *informative tweets*, **(b)** *informative hashtags*, **(c)** *informative users*, **(d)** *informative urls.*
- a *user is considered to be informative* if it is strongly associated with: **(a)** *informative tweets*, **(b)** *informative hashtags*, **(c)** *informative entities*, **(d)** *informative urls.*
- a *url is considered to be informative* if it is strongly associated with: **(a)** *informative tweets*, **(b)** *informative hashtags*, **(c)** *informative entities*, **(d)** *informative users.*

The relationships between event-specific *information units* together with their relationships with the tweets for an event E_i forms a *Mutual Reinforcement Chain* [24], as shown in Fig 1. We represent this relationship in a graph $G=(V,D)$, where $V=M_{E_i}\cup H_{E_i}\cup W_{E_i}\cup U_{E_i}\cup L_{E_i}$, is the set of vertices and D is the set of directed edges between different vertices. Whenever two vertices are associated, there are two edges between them that are oppositely directed. Each directed edge is assigned a weight, which determines the degree of association of one vertex with the other. The weights for each edge is calculated according to the conditional probabilities given in Table 1. We do not consider an edge between two vertices of same type.

We assign an initial event-specific score to all the vertices of the graph. The formulations of the scores assigned to the vertices $\in H_{E_i},W_{E_i},U_{E_i},L_{E_i}$ can be found in Table 1. For initializing the tweets $(\in M_{E_i})$ with an informativeness score we develop a logistic

Table 1. Affinity scores and event-specific initialization scores of nodes $\in G$.

Affinity scores between different nodes $\in M_{E_i}, H_{E_i}, W_{E_i}, U_{E_i}, L_{E_i}$:
$P(h_i \vert w_j)=\frac{No.\,of\,tweets\,h_i\,and\,w_j\,occur\,together}{No.\,of\,tweets\,w_j\,occurs}$, $P(w_i \vert h_j)=\frac{No.\,of\,tweets\,w_i\,and\,h_j\,occur\,together}{No.\,of\,tweets\,h_j\,occurs}$,
$P(h_i \vert l_j)=\frac{No.\,of\,tweets\,h_i\,and\,l_j\,occur\,together}{No.\,of\,tweets\,l_j\,occurs}$, $P(l_i \vert h_j)=\frac{No.\,of\,tweets\,l_i\,and\,h_j\,occur\,together}{No.\,of\,tweets\,h_j\,occurs}$,
$P(h_i \vert u_j)=\frac{No.\,of\,tweets\,h_i\,and\,u_j\,occur\,together}{No.\,of\,tweets\,u_j\,occurs}$, $P(u_i \vert h_j)=\frac{No.\,of\,tweets\,u_i\,and\,h_j\,occur\,together}{No.\,of\,tweets\,h_j\,occurs}$,
$P(w_i \vert l_j)=\frac{No.\,of\,tweets\,w_i\,and\,l_j\,occur\,together}{No.\,of\,tweets\,l_j\,occurs}$, $P(l_i \vert w_j)=\frac{No.\,of\,tweets\,l_i\,and\,w_j\,occur\,together}{No.\,of\,tweets\,w_j\,occurs}$,
$P(w_i \vert u_j)=\frac{No.\,of\,tweets\,w_i\,and\,u_j\,occur\,together}{No.\,of\,tweets\,u_j\,occurs}$, $P(u_i \vert w_j)=\frac{No.\,of\,tweets\,u_i\,and\,w_j\,occur\,together}{No.\,of\,tweets\,w_j\,occurs}$,
$P(u_i \vert l_j)=\frac{No.\,of\,tweets\,u_i\,and\,l_j\,occur\,together}{No.\,of\,tweets\,l_j\,occurs}$, $P(l_i \vert u_j)=\frac{No.\,of\,tweets\,l_i\,and\,u_j\,occur\,together}{No.\,of\,tweets\,u_j\,occurs}$,
$P(h_i \vert m_j)=P(m_i \vert h_j)=P(w_i \vert m_j)=P(m_i \vert w_j)=P(u_i \vert m_j)=P(m_i \vert u_j)=P(l_i \vert m_j)=P(m_i \vert l_j)=1.0$
Note: $P(h_i \mid w_j)$ should be read as the probability of occurrence of hashtag h_i given the occurrence of the entity w_j in the stream of tweets M_{E_i} related to event E_i collected over the time period T_{E_i}.
Event-specific initialization scores of nodes $\in H_{E_i}, W_{E_i}, U_{E_i}, L_{E_i}$:
$Score(h_i)=\frac{freq(h_i)}{max\{freq(h_1),freq(h_2),...,freq(h_p)\}}$ (1) $Score(w_i)=\frac{freq(w_i)}{max\{freq(w_1),freq(w_2),...,freq(w_r)\}}$ (2)
$Score(u_i)=\frac{followers(u_i)}{max\{followers(u_1),...,followers(u_r)\}}$ (3), $Score(l_i)=\frac{freq(l_i)}{max\{freq(l_1),freq(l_2),...,freq(l_r)\}}$ (4)
where, $freq(h_i)$ is the frequency of occurrence of the i^{th} hashtag ($\in H_{E_i}$) in the stream of tweets M_{E_i}. Similarly, $freq(w_i)$ denotes the frequency of occurrence of the i^{th} entity ($\in W_{E_i}$) and, $freq(l_i)$ denotes the frequency of occurrence of the i^{th} url ($\in L_{E_i}$). $followers(u_i)$ denotes the number of followers of user $u_i \in (U_{E_i})$.

Table 2. Features and Performance of the logistic regression model

Features for the logistic regression model
Has Url, No. of words, No. of stopwords, No. of feeling words [11], No. of slang words, No. of hashtags, No. of user mentions, Tweet length (No. of characters), No. of unique characters, No. of special characters, Favorite count, Retweet count, Formality[5] [2], Is tweet verified.

Performance	**Precision**	**Recall**	**F1-score**
Non-informative (0)	0.70	0.49	0.57
Informative (1)	0.78	0.90	0.84
Avg/Total	0.76	0.77	0.75
Accuracy =	76.32%		

regression model. For training the model we used an annotated dataset provided by [1]. The tweets labeled as *related and informative* were assigned a score of 1 and all the other tweets labeled as *related - but not informative* and *not related* were assigned a score of 0. Table 2, lists the features selected for each tweet, and reports the performance of the model for 10-fold cross validation. The model was then used for assigning informativeness score between 0 and 1 to all the tweets in the dataset, with 0 being least informative and 1 being most informative. The assigned initial scores gives an initial ranking of the vertices. We aim to refine the initial scores and assign a final score for ranking the vertices by leveraging the relationships between them and propagating the initial scores accordingly, from one vertex to another. Next, we formalize our ranking methodology and present our proposed algorithm step-by-step.

The relationships between two sets of vertices in the graph G is denoted by an affinity matrix. For example, $A_{E_i}^{MH}$ denotes the $M_{E_i}-H_{E_i}$ affinity matrix for event E_i, where $(i,j)^{th}$ entry is the edge weight quantifying the association between i^{th} tweet ($\in M_{E_i}$) and j^{th} hashtag ($\in H_{E_i}$), calculated using Table 1, and so on. The rankings of tweets, hashtags, entities, users and urls in terms of event-specific informativeness, can be iteratively derived from the Mutual Reinforcement Chain for the event. Let $R_{E_i}^M$, $R_{E_i}^H$, $R_{E_i}^W$, $R_{E_i}^U$ and $R_{E_i}^L$ denote the ranking scores for M_E, H_{E_i}, W_{E_i}, U_{E_i}, and L_{E_i}, respectively. Therefore, the Mutual Reinforcement Chain ranking for the k^{th} iteration can be formulated as follows:

$$R_{E_i}^{M(k+1)}=A_{E_i}^{MM(k)}+A_{E_i}^{MH(k)}+A_{E_i}^{MW(k)}+A_{E_i}^{MU(k)}+A_{E_i}^{ML(k)} \tag{1}$$

$$R_{E_i}^{H(k+1)}=A_{E_i}^{HM(k)}+A_{E_i}^{HH(k)}+A_{E_i}^{HW(k)}+A_{E_i}^{HU(k)}+A_{E_i}^{HL(k)} \tag{2}$$

$$R_{E_i}^{W(k+1)}=A_{E_i}^{WM(k)}+A_{E_i}^{WH(k)}+A_{E_i}^{WW(k)}+A_{E_i}^{WU(k)}+A_{E_i}^{WL(k)} \tag{3}$$

$$R_{E_i}^{U(k+1)}=A_{E_i}^{UM(k)}+A_{E_i}^{UH(k)}+A_{E_i}^{UW(k)}+A_{E_i}^{UU(k)}+A_{E_i}^{UL(k)} \tag{4}$$

$$R_{E_i}^{L(k+1)}=A_{E_i}^{LM(k)}+A_{E_i}^{LH(k)}+A_{E_i}^{LW(k)}+A_{E_i}^{LU(k)}+A_{E_i}^{LL(k)} \tag{5}$$

The equations 1-5 can be represented in the form of a block matrix Δ_{E_i}, where,

$$\Delta_{E_i}=\begin{pmatrix} A_{E_i}^{MM} & A_{E_i}^{MH} & A_{E_i}^{MW} & A_{E_i}^{MU} & A_{E_i}^{ML} \\ A_{E_i}^{HM} & A_{E_i}^{HH} & A_{E_i}^{HW} & A_{E_i}^{HU} & A_{E_i}^{HL} \\ A_{E_i}^{WM} & A_{E_i}^{WH} & A_{E_i}^{WW} & A_{E_i}^{WU} & A_{E_i}^{WL} \\ A_{E_i}^{UM} & A_{E_i}^{UH} & A_{E_i}^{UW} & A_{E_i}^{UU} & A_{E_i}^{UL} \\ A_{E_i}^{LM} & A_{E_i}^{LH} & A_{E_i}^{LW} & A_{E_i}^{LU} & A_{E_i}^{LL} \end{pmatrix}$$

Let

$$R_{E_i}=\begin{pmatrix} R_{E_i}^{M} & R_{E_i}^{H} & R_{E_i}^{W} & R_{E_i}^{U} & R_{E_i}^{L} \end{pmatrix}^T$$

then, R_{E_i} can be computed as the dominant eigenvector of Δ_{E_i}.

$$\Delta_{E_i}.R_{E_i}=\lambda.R_{E_i} \tag{6}$$

In order to guarantee a unique R_{E_i}, Δ_{E_i} must be forced to be stochastic and irreducible. We follow the steps taken by [24] in order to make Δ_{E_i} stochastic and irreducible. To make Δ_{E_i} stochastic we divide the value of each element in a column of Δ_{E_i} by the sum of the values of all the elements in that column. This finally makes Δ_{E_i} column stochastic. We now denote it by $\hat{\Delta}_{E_i}$. Next, we make $\hat{\Delta}_{E_i}$ irreducible. This is done by making the graph G strongly connected by adding links from one node to any other node with a probability vector p. Now, $\hat{\Delta}_{E_i}$ is transformed to

$$\overline{\Delta}_{E_i}=\alpha\hat{\Delta}_{E_i}+(1-\alpha)E \tag{7}$$

$$E=p\times[1]_{1\times k} \tag{8}$$

where $0\leq\alpha\leq1$ is set to 0.85 according to *PageRank*, and k is the order of $\hat{\Delta}_{E_i}$. We set $p=[1/k]_{k\times 1}$ by assuming a uniform distribution over all elements. Now, $\overline{\Delta}_{E_i}$ is stochastic and irreducible and it can be shown that it is also primitive by checking $\overline{\Delta}_{E_i}^2$ is greater than 0.

Following steps are taken next,

- We initialize the rank vectors $(R_{E_i}^{M(0)},R_{E_i}^{H(0)},R_{E_i}^{W(0)},R_{E_i}^{U(0)},R_{E_i}^{L(0)})$ for each subset of vertices $(M_{E_i},H_{E_i},W_{E_i},U_{E_i},L_{E_i})$, by calculating their initialization scores. All the scores lie between 0 and 1.
- Then we assign

  $$R_{E_i}^{0}=\begin{pmatrix} R_{E_i}^{M(0)} & R_{E_i}^{H(0)} & R_{E_i}^{W(0)} & R_{E_i}^{U(0)} & R_{E_i}^{L(0)} \end{pmatrix}^T$$

  and normalize $R_{E_i}^0$ such that $||R_{E_i}^0||_1=1$

– Apply power iteration method using the same parameters as used in PageRank with the convergence tolerance set at 1e-08 and λ=0.85.
– At the end of k^{th} iteration we normalize $R_{E_i}^k$ such that $||R_{E_i}^k||_1$=1
– We get the final rank vectors for each subset of the vertices ($R_{E_i}^M$,$R_{E_i}^H$,$R_{E_i}^W$,$R_{E_i}^U$,$R_{E_i}^L$) after convergence. We only choose $R_{E_i}^M$, which gives us the final scores of the tweets.
– We finally obtain the set $\hat{M}_{E_i}$ consisting of the tweets arranged in descending order of their final scores.

The final ordered set of tweets $\hat{M}_{E_i}$ are the tweets ranked in terms of their event-specific informative content sharing information about entities related to the event.

5 Experimental Settings and Evaluation

5.1 Data Collection

Table 3. Details of data collected for the experiment.

Event Name and Query Hashtag	**No. of Tweets**	**Time Period (UTC)**
Millions March NYC (#millionsmarchnyc) ($http://goo.gl/I8WR4B$)	56927	13th Dec, 2014; 20:25:43 - 14th Dec, 2014; 03:30:41
Sydney Siege (#sydneysiege) ($http://goo.gl/qLguvG$)	398204	15th Dec, 2014, 07:21:16 - 15th Dec, 2014; 22:46:45

For implementing and evaluating our proposed algorithm we collected 455,131 tweets from two real-life events, 'Millions March NYC' and 'Sydney Siege', using Twitter Streaming API. Details of the dataset is presented in Table 3. Tweets for each event was collected over the given period of time, by providing a popular hashtag corresponding to each event to the API. The choice of the events was driven by its availability in Seen.co event database, whose ranking scores[6] are used as one of the baselines representing the state-of-the-art technique.

5.2 Data Preparation

We performed a series of data preparation steps before implementing the logistic regression model and our algorithm. Tweets having duplicate content were detected using md5 hashing scheme, and redundant copies were filtered out keeping a single representation of the tweet in our database. Although, the methodology is language independent, we only considered english language tweets, as the manual annotators used for evaluation were only proficient in english.

We used the default parts-of-speech (POS) tagging module provided by NLTK library[7]. A standard list of english stop words was used for eliminating the stop words

[6] Tweets in Seen.co is ranked according to their proprietary algorithm SeenRank and the scores are available in the response of their API found at (http://developer.seen.co/) We used a python wrapper freely available at https://github.com/dxmahata/pySeen for collecting data from Seen.co
[7] http://nltk.org

from tweet text. All the characters of the tweets were converted to lower case. The tweets were tokenized after detecting the POS tags and removing the special characters. We filtered out the user mentions, retweet symbol (RT) and urls from the text during tokenization and did not consider them as tokens. A list of words expressing feelings was obtained from *wefeelfine.org*. Twitter related slang words were obtained from a publicly available document published by United States FBI[8]. A final list of slang words was compiled by adding some more internet slangs. The list would be made available on request. Retweet counts, favorite counts, verification information, user followers count and time information were obtained from the metadata attached with each tweet returned by Twitter API. The urls shared in tweets are generally shortened. Due to the use of different url shortener services, a single url might be represented in different forms by each service. In order to solve this problem, we used AlchemyAPI[9] to expand the urls to their original form. The slang hashtags were removed. We extracted named entities from the tweets using AlchemyAPI. The entities containing slang words were removed. Removal of slang hashtags and entities was done in order to obtain high quality results as intuitively high quality informative tweets should not contain slangs.

5.3 Baselines and Evaluation

In order to evaluate the performance of our algorithm we selected three different techniques that acted as our baselines. One of them is a proprietary algorithm known as *SeenRank* commercially used by Seen.co for generating event summarization and highlights from Twitter. We considered SeenRank as the state-of-the-art technique. Number of retweets is a good measure of popularity of a tweet and is also used by Twitter as well as many other applications for ranking. Therefore, we also considered tweets ordered in decreasing order of number of retweets as one of our baselines. We named this ordering as *RTRank*. The Logistic Regression model that we implemented for initializing the informativeness score of the tweets was considered as the third baseline. We considered it in order to make sure that our algorithm improves upon the initial informativeness score already assigned to the tweets.

We evaluated the rankings obtained using our algorithm on the two datasets by comparing its performance with the selected baselines. A subset of tweets for each event for a given time period (one hour) was selected. The choice of the time period was made on the basis of the intersecton of the time period of the tweets collected by us and that provided by seen.co for the same event. There were 21641 tweets for Millions March NYC and 37429 tweets for Sydney Siege, respectively.

We obtained the ranked tweets for our algorithm as well as the baselines. For all the approaches except *SeenRank* the tweets were sorted in decreasing order on the basis of the ranking scores as the primary key and time of posting as the secondary key. This was done in order to get the recent informative tweets at the top of the order. For *SeenRank* we sorted the tweets in terms of the scores assigned to them by Seen, as showing recent informative tweets for an event is one of the features of their platform. These ranked results were then annotated on an informativeness scale of 1 to 3 (1 being least informative and 3 the most informative) by three graduate students as independent annotators. Necessary background of the events were given to the annotators. In addition to event-specific information, they were also instructed to look for information related to entities relevant to the event.

[8] https://www.documentcloud.org/documents/1199460-responsive-documents.html#document/p1
[9] http://alchemyapi.com

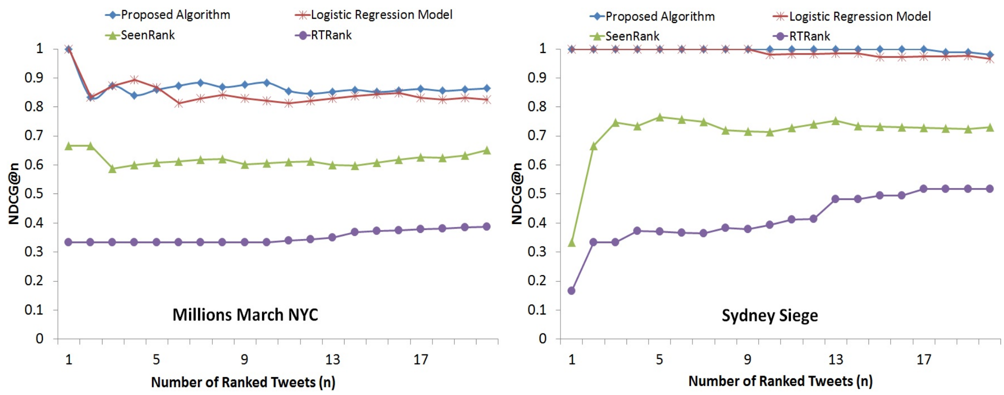

Fig. 2. Performance comparison of ranking techniques using NDCG scores.

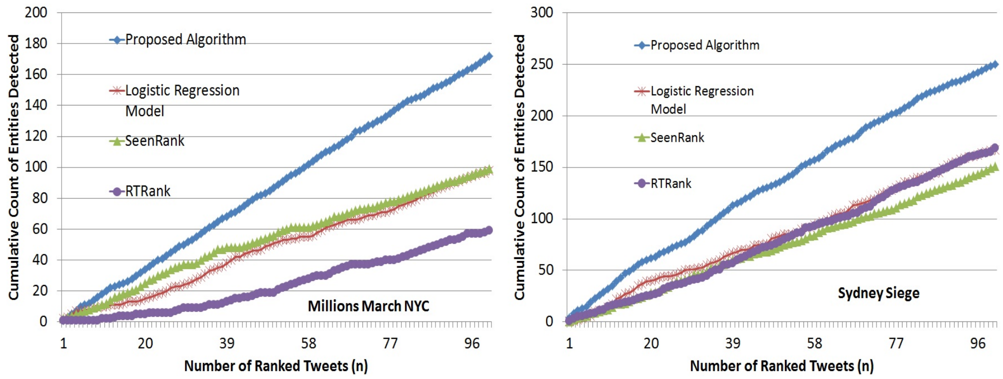

Fig. 3. Cumulative count of entities detected by the annotators for first hundred tweets ranked according to the different techniques.

The annotators browsed the first fifty ranked tweets for all the four approaches for each of the datasets and assigned each of those tweets a rank from among the three ranks 1, 2 and 3. Thereafter, we computed *Inter Indexer Consistency* (IIC) values for the annotations of the two datasets. For the Million March NYC dataset, the average IIC value for annotations of the seven results by three annotators was 0.76. For the Sydney Siege event, the IIC value obtained was 0.83. The IIC values for both the events fall in the acceptable range of accuracy of annotations. The annotators also reported the number of entities they could identify in each tweet.

After being assured about consistency and accuracy of annotations, we moved to compute the *Normalized Discounted Cumulative Gain* (NDCG) [10] values at each of the fifty recall levels. This has been done for all the approaches for each of the datasets. Fig 2 shows the NDCG curves for all the approaches on the Millions March NYC and the Sydney Siege events, respectively, for the first 20 tweets. It is quite evident from the figures that our proposed algorithm outperforms all the baselines including the state-of-the-art approach. We also calculated the cumulative count of entities identified by the annotators in the top hundred ranked tweets for each approach

(Fig 3). In order to assign a single count to entities identified in each tweet, we took the average number of entities identified by the three annotators in each tweet. Our algorithm outperformed all the other approaches as shown in Fig 3. This assured that our proposed algorithm performed better than all the baseline techniques in identifying event-specific informative content sharing information about event related entities.

Table 4. NDCG values at 10, 20, 30, 40 and 50 for ranked tweets of Millions March NYC and Sydney Siege.

Millions March NYC	**@10**	**@20**	**@30**	**@40**	**@50**
Proposed Algorithm	0.884	0.864	0.893	0.894	0.908
Logistic Regression Model	0.821	0.826	0.821	0.812	0.820
SeenRank	0.607	0.651	0.717	0.741	0.758
RTRank	0.333	0.387	0.432	0.487	0.524

Sydney Siege	**@10**	**@20**	**@30**	**@40**	**@50**
Proposed Algorithm	1.000	0.980	0.941	0.918	0.902
Logistic Regression Model	0.981	0.966	0.923	0.924	0.917
SeenRank	0.713	0.729	0.787	0.813	0.817
RTRank	0.394	0.516	0.552	0.588	0.648

We also report the detailed NDCG values for all the approaches at 10,20,30,40 and 50 recall levels, respectively in Table 4 for both the events. Apart from performing better than the other techniques in identifying event-specific informative tweets containing information about event related entities, our proposed model has an additional advantage of identifying and ranking top informative hashtags, entities, urls and users for an event. Due to space constrains, we show the top 5 hashtags, entities and urls for the Sydney Siege event in Table 5. We do not report the users for privacy concerns.

Table 5. Top 5 informative hashtags, entities and urls for Sydney Siege

Event	**Sydney Siege**
Top 5 Informative Hashtags	1. #sydneysiege, 2. #SydneySiege, 3. #Sydneysiege, 4. #MartinPlace, 5. #9News
Top 5 Informative Entities	1. police, 2. sydney, 3. reporter, 4. lindt, 5. isis
Top 5 Informative Urls	1. http://www.cnn.com/2014/12/15/world/asia/australia-sydney-hostage-situation/index.html 2. http://www.bbc.co.uk/news/world-australia-30474089, 3. http://edition.cnn.com/2014/12/15/world/asia/australia-sydney-siege-scene/index.html, 4. http://rt.com/news/214399-sydney-hostages-islamists-updates/, 5. http://www.newsroompost.com/138766/sydney-cafe-siege-ends-gunman-among-two-killed

6 Conclusion and Future Work

In this paper we proposed a novel model for identifying and ranking event-specific informative tweets sharing information about named entities related to the event. We defined event-specific *information units*, and identified mutually reinforcing relationships between them and the tweets produced during an event. A set of named entities extracted from the tweets for an event were considered as one of the information units.

We represented the associations between information units in a graph structure that forms the underlying framework for our ranking algorithm. We also defined and quantified the semantics of the relationships between the vertices of the graph and assigned event-specific scores to the edges and vertices. We proposed an algorithm for ranking the vertices. The algorithm makes use of the mutually reinforcing chains formed between the vertices of the graph for propagating the event-specific scores of a vertex to its neighbors. The accumulated score of the vertices after the convergence of the algorithm is used for ranking streams of tweets produced during two real-life events in terms of their event-specific informative content discussing about event related named entities.

We obtained promising results using our algorithm. The results were evaluated by comparing the performance of our approach with 3 other approaches including the state-of-the-art *SeenRank* algorithm used by Seen.co for ranking tweets displayed in their website. Our approach outperforms all the baselines. Additionally, our technique simultaneously ranked top informative hashtags, entities, users and urls related to an event. Our next step would be to extend the developed framework and implement it in a distributed computing environment, particularly integrating it with mapreduce. We also plan to use our framework for generating event summaries from Twitter, implementing event-centric recommendation in microblog environment and create event identities from the ranked information units for identifying event related content in other social media channels.

References

[1] A. Olteanu, C. Castillo, F. Diaz, S. Vieweg. 2014. CrisisLex: A Lexicon for Collecting and Filtering Microblogged Communications in Crises. In Proceedings of the AAAI Conference on Weblogs and Social Media (ICWSM'14). AAAI Press, Ann Arbor, MI, USA.

[2] Alejandro, Mosquera, and Moreda Paloma. "The use of metrics for measuring informality levels in web 2.0 texts." (2011).

[3] Becker, Hila, Mor Naaman, and Luis Gravano. "Beyond Trending Topics: Real-World Event Identification on Twitter." ICWSM 11 (2011): 438-441.

[4] Castillo, Carlos, Marcelo Mendoza, and Barbara Poblete. "Information credibility on twitter." Proceedings of the 20th international conference on World wide web. ACM, 2011.

[5] Damak, Firas, et al. "Effectiveness of State-of-the-art Features for Microblog Search." Proceedings of the 28th Annual ACM Symposium on Applied Computing. ACM, 2013.

[6] Duan, Yajuan, et al. "An empirical study on learning to rank of tweets." Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, 2010.

[7] Erkan, Gnes, and Dragomir R. Radev. "LexRank: Graph-based lexical centrality as salience in text summarization." J. Artif. Intell. Res.(JAIR) 22.1 (2004): 457-479.

[8] Hallberg, V., et al. "An adaptation of the PageRank algorithm to Twitter world." (2012).

[9] Inouye, David, and Jugal K. Kalita. "Comparing twitter summarization algorithms for multiple post summaries." Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom). IEEE, 2011.

[10] Jrvelin, Kalervo, and Jaana Keklinen. "Cumulated gain-based evaluation of IR techniques." ACM Transactions on Information Systems (TOIS) 20.4 (2002): 422-446.

[11] Kamvar, Sepandar D., and Jonathan Harris. "We feel fine and searching the emotional web." Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011.

[12] Laniado, David, and Peter Mika. "Making sense of twitter." The Semantic WebISWC 2010. Springer Berlin Heidelberg, 2010. 470-485.

[13] McCreadie, Richard, and Craig Macdonald. "Relevance in microblogs: enhancing tweet retrieval using hyperlinked documents." Proceedings of the 10th Conference on Open Research Areas in Information Retrieval, 2013.

[14] Mihalcea, Rada, and Paul Tarau. "TextRank: Bringing order into texts." Association for Computational Linguistics, 2004.

[15] Nichols, Jeffrey, Jalal Mahmud, and Clemens Drews. "Summarizing sporting events using twitter." Proceedings of the 2012 ACM international conference on Intelligent User Interfaces. ACM, 2012.

[16] Page, Lawrence, et al. "The PageRank citation ranking: Bringing order to the web." (1999).

[17] Popescu, Ana-Maria, Marco Pennacchiotti, and Deepa Paranjpe. *"Extracting events and event descriptions from twitter." Proceedings of the 20th international conference companion on World wide web. ACM, 2011.*

[18] Purohit, Hemant, and Amit P. Sheth. *"Twitris v3: From Citizen Sensing to Analysis, Coordination and Action." ICWSM. 2013.*

[19] Sakaki, Takeshi, Makoto Okazaki, and Yutaka Matsuo. *"Earthquake shakes Twitter users: real-time event detection by social sensors." Proceedings of the 19th international conference on World wide web. ACM, 2010.*

[20] Sharifi, Beaux, M-A. Hutton, and Jugal K. Kalita. "Experiments in microblog summarization." Social Computing (SocialCom), 2010 IEEE Second International Conference on. IEEE, 2010.

[21] Shaw, Ryan, Raphal Troncy, and Lynda Hardman. "Lode: Linking open descriptions of events." The Semantic Web. Springer Berlin Heidelberg, 2009. 153-167.

[22] Tunkelang, D. "A twitter analog to pagerank." The Noisy Channel (2009).

[23] Vosecky, Jan, Kenneth Wai-Ting Leung, and Wilfred Ng. "Searching for quality microblog posts: Filtering and ranking based on content analysis and implicit links." Database Systems for Advanced Applications. Springer Berlin Heidelberg, 2012.

[24] Wei, Furu, et al. "Query-sensitive mutual reinforcement chain and its application in query-oriented multi-document summarization." Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2008.

[25] Weng, Jianshu, et al. "Twitterrank: finding topic-sensitive influential twitterers." Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010.