A Participant-based Approach for Event Summarization Using Twitter Streams

Chao Shen[1]**, Fei Liu**[2]**, Fuliang Weng**[2]**, Tao Li**[1]
[1]School of Computing and Information Sciences, Florida International University
Miami, Florida 33199, USA
[2]Research and Technology Center, Robert Bosch LLC
Palo Alto, California 94304, USA
{cshen001, taoli}@cs.fiu.edu
{fei.liu, fuliang.weng}@us.bosch.com

Abstract

Twitter offers an unprecedented advantage on live reporting of the events happening around the world. However, summarizing the Twitter event has been a challenging task that was not fully explored in the past. In this paper, we propose a participant-based event summarization approach that "zooms-in" the Twitter event streams to the participant level, detects the important sub-events associated with each participant using a novel mixture model that combines the "burstiness" and "cohesiveness" properties of the event tweets, and generates the event summaries progressively. We evaluate the proposed approach on different event types. Results show that the participant-based approach can effectively capture the sub-events that have otherwise been shadowed by the long-tail of other dominant sub-events, yielding summaries with considerably better coverage than the state-of-the-art.

1 Introduction

Twitter has increasingly become a critical source of information. People report the events they are experiencing or publish comments on a wide variety of events happening around the world, ranging from the unexpected natural disasters, regional riots, to many scheduled events, such as sports games, political debates, local festivals, and even academic conferences. The Twitter data streams thus cover a broad range of events and broadcast these information in a live manner. Event summarization in this paper aims to generate a representative and concise textual description of the *scheduled* events that are being lively reported on Twitter, providing people with an alternative means of observing the world beyond the traditional journalism. Specifically, we investigate scheduled events of different types, including six of the NBA (National Basketball Association) sports games and a representative conference event, namely the Apple CEO's keynote speech in the Apple Worldwide Developers Conference (WWDC 2012)[1]. All these events have excited great discussion among the Twitter community.

Summarizing the Twitter event is a challenging task that has yet been fully explored in the past. Most previous summarization studies focus on the well-formatted news documents, as driven by the annual DUC[2] and TAC[3] evaluations. In contrast, the Twitter messages (a.k.a., tweets) are very short and noisy, containing nonstandard terms such as abbreviations, acronyms, emoticons, etc. (Liu et al., 2011b; Liu et al., 2012; Eisenstein, 2013). The noisy contents also cause great difficulties to the traditional NLP tools such as NER and dependency parser (Ritter et al., 2011; Foster et al., 2011), limiting the possibility of applying finer-grained event analysis tools. In nature, the event tweets are closely associated with the timeline and are drastically different from a static collection of news documents. The tweets converge into text streams that pulse along the timeline and cluster around the important moments or sub-events. These "sub-events" are of crucial importance since they represent a surge of interest from the Twitter audience and the correspond-

[1] https://developer.apple.com/wwdc/

[2] http://duc.nist.gov/

[3] http://www.nist.gov/tac/

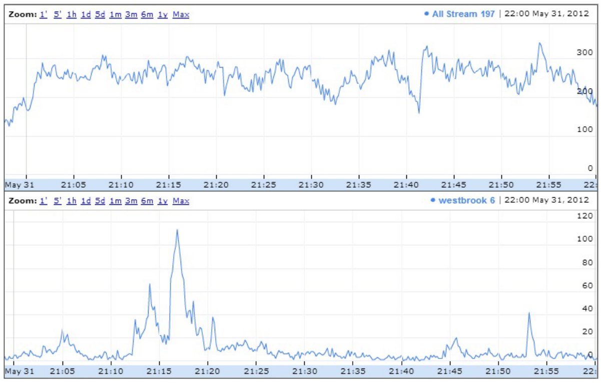

Figure 1: Example Twitter event stream (upper) and participant stream (lower). Event stream contains tweets related to an NBA basketball game (Spurs vs Thunder) scheduled on May 31, 2012; participant stream contains tweets corresponding to the player Russell Westbrook in team Thunder. X-axis denotes the timeline and y-axis represents the number of tweets per 10-second interval.

ing key information must be reflected in the event summary. As such, event summarization research has been focusing on developing accurate sub-event detection systems and generating text descriptions that can best summarize the sub-events in a progressive manner (Chakrabarti and Punera, 2011; Nichols et al., 2012; Zubiaga et al., 2012).

In Figure 1, we show an example Twitter event stream and one of its "participant" streams. The event stream contains all the tweets related to an NBA basketball game Spurs vs Thunder; while the participant stream contains only tweets corresponding to the player Russell Westbrook in this game. Previous research on event summarization focuses on identifying the important moments from the coarse-level event stream. This may yield several side effects: first, the spike patterns are not clearly identifiable from the overall event stream, though they are more clearly seen if we "zoom-in" to the participant level; second, it is arguable whether the important sub-events can be accurately detected based solely on the tweet volume change; third, a popular participant or sub-event can elicit huge volume of tweets which dominant the event discussion and shield less prominent sub-events. For example, in the NBA games, discussions about the key players (e.g., "LeBron James", "Kobe Bryant") can heavily shadow other important participants or sub-events, resulting in an event summary with repetitive descriptions about the dominant players.

In this work, we propose a novel participant-based event summarization approach, which dynamically identifies the participants from data streams, then "zooms-in" the event stream to participant level, detects the important sub-events related to each participant using a novel time-content mixture model, and generates the event summary progressively by concatenating the descriptions of the important sub-events. Results show that the mixture model-based sub-event detection approach can efficiently incorporate the "burstiness" and "cohesiveness" of the participant streams, and the participant-based event summarization can effectively capture the sub-events that have otherwise been shadowed by the long-tail of other dominant sub-events, yielding summaries with considerably better coverage than the state-of-the-art approach.

2 Related Work

Mining Twitter for event information has received increasing attention in recent years. Many research studies focus on identifying the trending events from Twitter and providing a concise and dynamic visualization of the information. The identified events are often represented using a set of keywords. (Petrovic et al., 2010) proposed an algorithm based on locality-sensitive hashing for detecting new events from a stream of Twitter posts. (O'Connor et al., 2010; Becker et al., 2011b; Becker et al., 2011a; Weng et al., 2011) proposed demo systems to display the event-related themes and popular tweets, allowing the users to navigate through their topic of interest. (Zhao et al., 2011) described an effort to perform data collection and event recognition despite various limits to the free access of Twitter data. (Diao et al., 2012) integrated both temporal information and users' personal interests for bursty topic detection from the microblogs. (Ritter et al., 2012) described an open-domain event-extraction and categorization system, which extracts an open-domain calendar of significant events from Twitter.

With the identified events of interest, there is an ever-increasing demand for event summarization, which distills the huge volume of Twitter discussions into a concise and representative textual description of the events. Many studies start with the text summarization approaches that have been shown to perform well on the news documents and develop adaptations to fit these methods to a collection of event tweets. (Sharifi et al., 2010b) proposed a graph-based phrase reinforcement algorithm to build a one-sentence summary from a collection of topic tweets. (Sharifi et al., 2010a; Inouye and Kalita, 2011) presented a hybrid TF-IDF approach to extract one- or multiple-sentence summary for each topic. (Liu et al., 2011a) proposed to use the concept-based ILP framework for summarizing the Twitter trending topics, using both tweets and the webpages linked from the tweets as input text sources. (Harabagiu and Hickl, 2011) introduced a generative framework that incorporates event structure and user behavior information in summarizing multiple microblog posts related to the same topic.

Regarding summarizing the data streams, (Marcus et al., 2011) introduced a “TwitInfo” system to visually summarize and track the events on Twitter. They proposed an automatic peak detection and labeling algorithm for the social streams. (Takamura et al., 2011) proposed a summarization model based on the facility location problem, which generates summary for a stream of short documents along the timeline. (Chakrabarti and Punera, 2011) proposed an event summarization algorithm based on learning an underlying hidden state representation of the event via hidden Markov models. (Louis and Newman, 2012) presented a method for summarizing a collection of tweets related to a business. The proposed procedure aggregates tweets into subtopic clusters which are then ranked and summarized by a few representative tweets from each cluster. (Nichols et al., 2012; Zubiaga et al., 2012) focused on real-time event summarization, which detects the sub-events by identifying those moments where the tweet volume has increases sharply, then uses various weighting schemes to perform tweet selection and finally generates the event summary.

Our work is different from the above research studies in three folds: first, we propose to “zoom-in” the Twitter event streams to the participant level, which allows us to clearly identify the important sub-events associated with each participant and generate a balanced event summary with comprehensive coverage of all the important sub-events; second, we propose a novel time-content mixture model approach for sub-event detection, which effectively leverages the “burstiness” and “cohesiveness” of the event tweets and accurately detects the participant-level sub-events. Third, we evaluate the participant-based event summarization system on different event types and demonstrate that the proposed approach outperforms the state-of-the-art method by a considerable margin.

3 Participant-based Event Summarization

We propose a novel participant-centered event summarization approach that consists of three key components: (1) “Participant Detection” dynamically identifies the event participants and divides the entire event stream into a number of participant streams (Section 3.1); (2) “Sub-event Detection” introduces a novel time-content mixture model approach to identify the important sub-events associated with each participant; these “participant-level sub-events” are then merged along the timeline to form a set of “global sub-events”[4], which capture all the important moments in the event stream (Section 3.2); (3) “Summary Tweet Extraction” extracts the representative tweets from the global sub-events and forms a comprehensive coverage of the event progress (Section 3.3).

3.1 Participant Detection

We define event participants as the entities that play a significant role in shaping the event progress. “Participant” is a general concept to denote the event participating persons, organizations, product lines, etc., each of which can be captured by a set of correlated proper nouns. For example, the NBA player “*LeBron Raymone James*” can be represented by $\{$*LeBron James, LeBron, LBJ, King James, L. James*$\}$, where each proper noun represents a unique mention of the participant. In this work, we automatically identify the proper nouns from tweet streams, filter out the infrequent ones using a threshold ψ, and cluster them into individual event participants. This process allows us to dynamically identify the key participating entities and provide a full-coverage for these participants in the event summary.

[4] We use “**participant sub-events**” and “**global sub-events**” respectively to represent the important moments happened on the participant-level and on the entire event-level. A “global sub-event” may consist of one or more “participant sub-events”. For example., the “steal” action in the basketball game typically involves both the defensive and offensive players, and can be generated by merging the two participant-level sub-events.

We formulate the participant detection in a hierarchical agglomerative clustering framework. The CMU TweetNLP tool (Gimpel et al., 2011) was used for proper noun tagging. The proper nouns (a.k.a., mentions) are grouped into clusters in a bottom-up fashion. Two mentions are considered similar if they share (1) lexical resemblance, and (2) contextual similarity. For example, in the following two tweets "*Gotta respect Anthony Davis, still rocking the unibrow*", "*Anthony gotta do something about that unibrow*", the two mentions *Anthony Davis* and *Anthony* are referring to the same participant and they share both character overlap ("anthony") and context words ("unibrow", "gotta"). We use $sim(c_i, c_j)$ to represent the similarity between two mentions c_i and c_j, defined as:

$$sim(c_i, c_j) = lex\_sim(c_i, c_j) \times cont\_sim(c_i, c_j)$$

where the lexical similarity ($lex\_sim(\cdot)$) is defined as a binary function representing whether a mention c_i is an abbreviation, acronym, or part of another mention c_j, or if the character edit distance between the two mentions is less than a threshold θ[5]:

$$lex\_sim(c_i, c_j) = \begin{cases} 1 & c_i(c_j) \text{ is part of } c_j(c_i) \\ 1 & \text{EditDist}(c_i, c_j) < \theta \\ 0 & \text{Otherwise} \end{cases}$$

We define the context similarity ($cont\_sim(\cdot)$) of two mentions as the cosine similarity between their context vectors $\vec{v}_i$ and $\vec{v}_j$. Note that on the tweet stream, two temporally distant tweets can be very different even though they are lexically similar, e.g., two slam dunk shots performed by the same player at different time points are different. We therefore restrain the context to a segment of the tweet stream $|S_k|$ and then take the weighted average of the segment-based similarity as the final context similarity. To build the context vector, we use term frequency (TF) as the term weight and remove all the stopwords. We use $|D|$ to represent the total tweets in the event stream.

$$cont\_sim_{|S_k|}(c_i, c_j) = \cos(\vec{v}_i, \vec{v}_j)$$

$$cont\_sim(c_i, c_j) = \sum_k \frac{|S_k|}{|D|} \times cont\_sim_{|S_k|}(c_i, c_j)$$

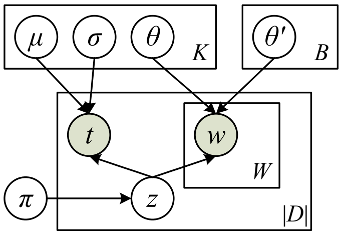

Figure 2: Plate notation of the mixture model.

Similarity between two clusters of mentions are defined as the maximum possible similarity between a pair of mentions, each from one cluster:

$$sim(C_i, C_j) = \max_{c_i \in C_i, c_j \in C_j} sim(c_i, c_j)$$

We perform bottom-up agglomerative clustering on the mentions until a stopping threshold δ has been reached for $sim(C_i, C_j)$. The clustering approach naturally groups the frequent proper nouns into participants. The **participant streams** are then formed by gathering the tweets that contain one or more mentions in the participant cluster.

3.2 Mixture Model-based Sub-event Detection

A sub-event corresponds to a topic that emerges from the data stream, being intensively discussed during a short period, and then gradually fades away. The tweets corresponding to a sub-event thus demand not only "temporal burstiness" but also a certain degree of "lexical cohesiveness". To incorporate both the time and content aspects of the sub-events, we propose a mixture model approach for sub-event detection. Figure 2 shows the plate notation.

In the proposed model, each tweet d in the data stream D is generated from a topic z, weighted by π_z. Each topic is characterized by both its content and time aspects. The content aspect is captured by a multinomial distribution over the words, parameterized by θ; while the time aspect is characterized by a Gaussian distribution, parameterized by μ and σ, with μ represents the average time point that the sub-event emerges and σ determines the duration of the sub-event. These distributions bear similarities with the previous work (Hofmann, 1999; Allan, 2002; Haghighi and Vanderwende, 2009). In addition, there are often background or "noise" topics that are being constantly discussed over the entire

[5] θ was empirically set as $0.2 \times min\{|c_i|, |c_j|\}$

event evolvement process and do not present the desired "burstiness" property. We use a uniform distribution $U(t_b, t_e)$ to model the time aspect of these "background" topics, with t_b and t_e being the event beginning and end time points. The content aspect of a background topic is modeled by similar multinomial distribution, parameterized by θ'. We use the maximum likelihood parameter estimation. The data likelihood can be represented as:

$$L(D) = \prod_{d \in D} \sum_{z} \{\pi_z p_z(t_d) \prod_{w \in d} p_z(w)\}$$

where $p_z(t_d)$ models the timestamp of tweet d under the topic z; $p_z(w)$ corresponds to the word distribution in topic z. They are defined as:

$$p_z(t_d) = \begin{cases} N(t_d; \mu_z, \sigma_z) & \text{if } z \text{ is a sub-event topic} \\ U(t_b, t_e) & \text{if } z \text{ is background topic} \end{cases}$$

$$p_z(w) = \begin{cases} p(w; \theta_z) & \text{if } z \text{ is a sub-event topic} \\ p(w; \theta'_z) & \text{if } z \text{ is background topic} \end{cases}$$

where both $p(w;\theta_z)$ and $p(w;\theta'_z)$ are multinomial distributions over the words. Initially, we assume there are K sub-event topics and B background topics and use the EM algorithm for model fitting. The EM equations are listed below:

E-step:

$$p(z_d = j) \propto \begin{cases} \pi_j N(d; \mu_j, \sigma_j) \prod_{w \in d} p(w; \theta_j) & \text{if } j <= K \\ \pi_j U(t_b, t_e) \prod_{w \in d} p(w; \theta'_j) & \text{else} \end{cases}$$

M-step:

$$\pi_j \propto \sum_d p(z_d = j)$$

$$p(w; \theta_j) \propto \sum_d p(z_d = j) \times c(w, d)$$

$$p(w; \theta'_j) \propto \sum_d p(z_d = j) \times c(w, d)$$

$$\mu_j = \frac{\sum_d p(z_d = j) \times t_d}{\sum_{j=1}^{K} \sum_d p(z_d = j)}$$

$$\sigma_j^2 = \frac{\sum_d p(z_d = j) \times (t_d - \mu_j)^2}{\sum_{j=1}^{K} \sum_d p(z_d = j)}$$

To process the data stream D, we divide the data into 10-second bins and process each bin at a time. The peak time of a sub-event was determined as the bin that has the most tweets related to this sub-event. During EM initialization, the number of sub-event topics K was empirically decided by scanning through the data stream and examine tweets in every 3-minute stream segment. If there was a spike[6], we add a new sub-event to the model and use the tweets in this segment to initialize the value of μ, σ, and θ. Initially, we use a fixed number of background topics with $B = 4$. A topic re-adjustment was performed after the EM process. We merge two sub-events in a data stream if they (1) locate closely in the timeline, with peaks times within a 2-minute window; and (2) share similar word distributions: among the top-10 words with highest probability in the word distributions, there are over 5 words overlap. We also convert the sub-event topics to background topics if their σ values are greater than a threshold β[7]. We then re-run the EM to obtain the updated parameters. The topic re-adjustment process continues until the number of sub-events and background topics do not change further.

We obtain the "**participant sub-events**" by applying this sub-event detection approach to each of the participant streams. The "**global sub-events**" are obtained by merging the participant sub-events along the timeline. We merge two participant sub-events into a global sub-event if (1) their peaks are within a 2-minute window, and (2) the Jaccard similarity (Lee, 1999) between their associated tweets is greater than a threshold (set to 0.1 empirically). The tweets associated with each global sub-event are the ones with $p(z|d)$ greater than a threshold γ, where z is one of the participant sub-events and γ was set to 0.7 empirically. After the sub-event detection process, we obtain a set of global sub-events and their associated event tweets.[8]

3.3 Summary Tweet Extraction

We extract a representative tweet from each of the global sub-events and concatenate them to form an informative event summary. Note that our goal in this work is to identify all the important moments

[6] We use the algorithm described in (Marcus et al., 2011) as a baseline and ad hoc spike detection algorithm.

[7] β was set to 5 minutes in our experiments.

[8] We empirically set some threshold values in the topic re-adjustment and sub-event merging process. In future, we would like to explore more principled way of parameter selection.

Event		Date	Duration	#Tweets
NBA	Lakers vs Okc	05/19/2012	3h10m	218,313
	Celtics vs 76ers	05/23/2012	3h30m	245,734
	Celtics vs Heat	05/30/2012	3h30m	345,335
	Spurs vs Okc	05/31/2012	3h	254,670
	Heat vs Okc (1)	06/12/2012	3h30m	331,498
	Heat vs Okc (2)	06/21/2012	3h30m	332,223
Apple's WWDC'12 Conf.		06/11/2012	3h30m	163,775

Table 1: Statistics of the data set, including six NBA basketball games and the WWDC 2012 conference event.

for event summarization, but not on proposing new methods for tweet selection. We thus use the Hybrid TF-IDF approach (Sharifi et al., 2010a; Liu et al., 2011a) to extract the representative sentences from a collection of tweets. In this approach, each tweet was considered as a sentence. The sentences were ranked according to the average TF-IDF score of the consisting words; top weighted sentences were iteratively extracted, while excluding those that have high cosine similarity with the existing summary sentences. (Inouye and Kalita, 2011) showed the Hybrid TF-IDF approach performs constantly better than the phrase reinforcement algorithm and other traditional summarization systems.

4 Data Corpus

We evaluate the proposed event summarization approach on six NBA basketball games and a representative conference event, namely the Apple CEO's keynote speech in the Apple Worldwide Developers Conference (WWDC 2012)[9]. We use the heterogeneous event types to verify that the proposed approach can robustly and efficiently produce summaries on different event streams. The tweet streams corresponding to these events are collected using the Twitter Streaming API[10] with pre-defined keyword set. For NBA games, we use the team names, first name and last name of the players and head coaches as keywords for retrieving the event tweets; for the WWDC conference, the keyword set contains about 20 terms related to the Apple event, such as "wwdc", "apple", "mac", etc. We crawl the tweets in real-time when these scheduled events are taking place; nevertheless, certain non-event tweets could be mis-included due to the broad coverage of the used keywords. During preprocessing, we filter out the tweets containing URLs, non-English tweets, and retweets since they are less likely containing new information regarding the event progress. Table 1 shows statistics of the event tweets after the filtering process. In total, there are over 1.8 million tweets used in the event summarization experiments.

Time	Action (Sub-event)	Score
9:22	Chris Bosh misses 10-foot two point shot	7-2
9:22	Serge Ibaka defensive rebound	7-2
9:11	Kevin Durant makes 15-foot two point shot	9-2
8:55	Serge Ibaka shooting foul (Shane Battier draws the foul)	9-2
8:55	Shane Battier misses free throw 1 of 2	9-2
8:55	Miami offensive team rebound	9-2
8:55	Shane Battier makes free throw 2 of 2	9-3

Table 2: An example clip of the play-by-play live coverage of an NBA game (Heat vs Okc). "Time" corresponds to the minutes left in the current quarter of the game; "Score" shows the score between the two teams.

We use the play-by-play live coverage collected from the ESPN[11] and MacRumors[12] websites as reference, which provide detailed descriptions of the NBA and WWDC events as they unfold. Table 2 shows an example clip of the play-by-play descriptions of an NBA game. Ideally, each item in the live coverage descriptions may correspond to a sub-event in the tweet streams, but in reality, not all actions would attract enough attention from the Twitter audience. We use a human annotator to manually filter out the actions that did not lead to any spike in the corresponding participant stream. The rest items are projected to the participant and event streams as the goldstandard sub-events. The projection was manually performed since the "game clock" associated with the goldstandard (first column in Table 2) does not align well with the "wall clock" due to the game rules such as timeout and halftime rest. To evaluate the participant detection performance, we ask the annotator to manually group the proper noun mentions into clusters, each cluster corresponds to a participant. The mentions that do not correspond to any participant are discarded. The goldstandard event summaries are generated by manually selecting one representative tweet from each of the groundtruth global sub-events. We choose not to use the play-by-play descriptions as reference summaries since their vocabulary is rather limited and do not overlap with the tweet language.

[9] https://developer.apple.com/wwdc/

[10] https://dev.twitter.com/docs/streaming-apis

[11] http://espn.go.com/nba/scoreboard

[12] http://www.macrumorslive.com/archive/wwdc12/

Example Participants - NBA game
westbrook, russell westbrook
stephen jackson, steven jackson, jackson
james, james harden, harden
ibaka, serge ibaka
oklahoma city thunder, oklahoma
gregg popovich, greg popovich, popovich
kevin durant, kd, durant
thunder, okc, #okc, okc thunder, #thunder

Example Participants - WWDC Conference
macbooks, mbp, macbook pro, macbook air,...
google maps, google, apple maps
wwdc, apple wwdc, #wwdc
os, mountain, os x mountain, os x
iphone 4s, iphone 3gs, iphone

Table 3: Example participants automatically detected from the NBA game Spurs vs Okc (2012-5-31) and the WWDC'12 conference.

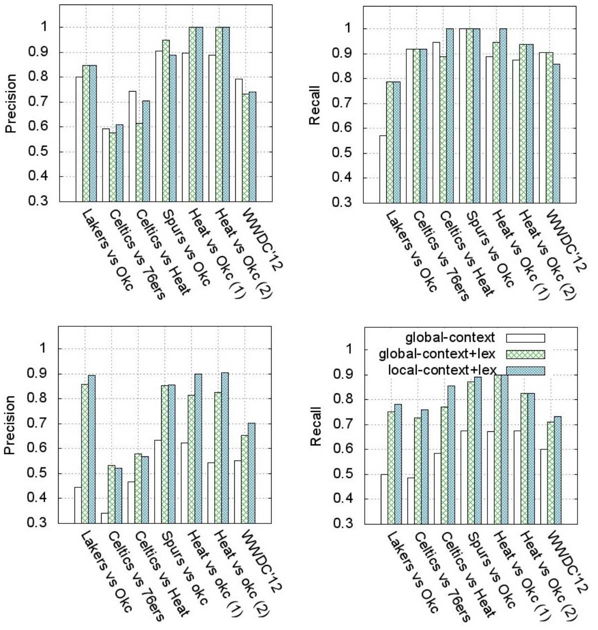

Figure 3: Participant detection performance. The upper figures represent the participant-level precision and recall scores; while the lower figures represent the mention-level precision and recall. X-axis corresponds to the six NBA games and the WWDC conference.

5 Experimental Results

We evaluate the participant-based event summarization in a cascaded fashion and present results for each of the three components, including the participant detection (Section 5.1), sub-event detection (Section 5.2), and quantitative and qualitative evaluation of example event summaries (Section 5.3).

5.1 Participant Detection Results

In Table 3, we show example participants that were automatically detected by the proposed hierarchical agglomerative clustering approach. We note that the clusters include various mentions of the same event participant, e.g., "*gregg popovich*", "*greg popovich*", and "*popovich*" are both referring to the head coach of the team Spurs; "*macbooks*", "*macbook pro*", "*mbp*" are referring to a line of products from Apple. Quantitatively, we evaluate the participant detection results on both participant- and mention-level. Assume the system-detected and the goldstandard participant clusters are T_s and T_g respectively. We define a **correct participant** as a system detected participant with more than half of its associated mentions are included in a goldstandard participant (referred to as the **hit participant**). As a result, we can define the participant-level precision and recall as below:

$$\text{participant-prec} = \#\text{correct-participants}/|T_s|$$
$$\text{participant-recall} = \#\text{hit-participants}/|T_g|$$

Note that a correct participant may include incorrect mentions, and that more than one correct participants may correspond to the same hit participant, both of which are undesired. In the latter case, we use **representative participant** to refer to the correct participant which contains the most mentions in the hit participant. In this way, we build a 1-to-1 mapping from the detected participants to the groundtruth participants. Next, we define **correct mentions** as the union of the overlapping mentions between all pairs of representative and hit participants. Then we calculate the mention-level precision and recall as the number of correct mentions divided by the total mentions in the system or goldstandard participant clusters.

Figure 3 shows the participant- and mention-level precision and recall scores. We experimented with different similarity measures for the agglomerative clustering approach[13]. The "global context" means that the context vectors are created from the entire data stream; this may not perform well since different participants can share similar global context. E.g., the terms "shot", "dunk", "rebound" can appear in the context of any NBA players and are not

[13] The stopping threshold δ was set to 0.15, local context length is 3 minutes, and frequency threshold ψ was set to 200.

Event	**Participant-level Sub-event Detection**									**Global Sub-event Detection**									
	#P	#S	Spike			MM			#S	Spike			Participant + Spike			Participant + MM		
			R	P	F	R	P	F		R	P	F	R	P	F	R	P	F
Lakers vs Okc	9	65	0.75	0.31	**0.44**	0.71	0.39	**0.50**	48	0.67	0.38	0.48	**0.94**	0.19	0.32	**0.88**	0.40	**0.55**
Celtics vs 76ers	10	88	0.52	0.39	**0.45**	0.53	0.43	**0.47**	60	0.65	0.51	0.57	**0.72**	0.18	0.29	**0.78**	0.39	**0.52**
Celtics vs Heat	14	152	0.53	0.29	**0.37**	0.50	0.38	**0.43**	67	0.57	0.41	0.48	**0.97**	0.21	0.35	**0.91**	0.28	**0.43**
Spurs vs Okc	12	98	0.78	0.46	**0.58**	0.84	0.57	**0.68**	81	0.41	0.42	0.41	**0.88**	0.35	0.50	**0.91**	0.54	**0.68**
Heat vs Okc (1)	15	123	0.75	0.27	**0.40**	0.72	0.35	**0.47**	85	0.41	0.47	0.44	**0.94**	0.20	0.33	**0.96**	0.34	**0.50**
Heat vs okc (2)	13	153	0.74	0.36	**0.48**	0.76	0.43	**0.55**	92	0.41	0.33	0.37	**0.88**	0.21	0.34	**0.87**	0.38	**0.53**
WWDC'12	10	56	0.64	0.14	**0.23**	0.59	0.33	**0.42**	43	0.53	0.26	0.35	**0.77**	0.14	0.24	**0.70**	0.31	**0.43**
Average	**12**	**105**	0.67	0.32	**0.42**	0.66	0.41	**0.50**	**68**	0.52	0.40	0.44	**0.87**	0.21	0.34	**0.86**	0.38	**0.52**

Table 4: Sub-event detection results on both participant and the event streams. “Spike” corresponds to the spike detection algorithm proposed in (Marcus et al., 2011); “MM” represents our proposed time-content mixture model approach. “#P” and “#S” list the number of participants and sub-events in each event stream.

discriminative enough. We found that adding the lexical similarity measure greatly boosted the clustering performance, especially on the mention-level, and that combining the lexical similarity with the local context is even more helpful for some events. We notice that two events (celtics vs 76ers and celtics vs heat) yield relatively low precision on both participant- and mention-level. Taking a close look at the data, we found that these two events accidentally co-occurred with other popular events, namely the TV program “American Idol” finale and the NBA Draft. The keyword based data crawler thus includes many noisy tweets in the event streams, leading to some false participants being detected.

5.2 Sub-event Detection Results

We compare our proposed time-content mixture model (noted as “MM”) against the spike detection algorithm proposed in (Marcus et al., 2011) (noted as “Spike”) . The spike algorithm is based on the tweet volume change. It uses 10 seconds as a time unit, calculates the tweet arrival rate in each unit, and identifies the rates that are significantly higher than the mean tweet rate. For these rate spikes, the algorithm finds the local maximum of tweet rate and identify a window surrounding the local maximum. We tune the parameter of the “Spike” approach (set $\tau = 4$) so that it yields similar recall values as the mixture model approach. We then apply the “MM” and “Spike” approaches to both the participant and event streams and evaluate the sub-event detection performance. Results are shown in Table 4. A system detected sub-event is considered to match the goldstandard sub-event if its peak time is within a 2-minute window of the goldstandard.

We first apply the “Spike” and “MM” approach to the participant streams. The participant streams on which we cannot detect any meaningful sub-events have been excluded, the resulting number of participants are listed in Table 4 and denoted as “#P”. In general, we found the “MM” approach can perform better since it inherently incorporates both the “burstiness” and “lexical cohesiveness” of the event tweets, while the “Spike” approach relies solely on the “burstiness” property. Note that although we divide the entire event stream into participant streams, some key participants still own huge amount of discussion and the spike patterns are not always clearly identifiable. The time-content mixture model gains advantages in these cases.

We apply three settings to detect global sub-events on the data streams. “Spike” directly applies the spike algorithm on the entire event stream; the “Participant + Spike” and “Participant + MM” approaches first perform sub-event detection on the participant streams and then merge the detected sub-events along the timeline to generate global sub-events. Note that there are fewer goldstandard sub-events (“#S”) on the global streams since each global sub-event may correspond to one or multiple participant-level sub-events. Because of the averaging effect, spike patterns on the entire event stream is less obvious than those on the participant streams. As a result, few spikes have been detected on the event stream using the “Spike” algorithm, which leads to low recall as compared to other participant-based approaches. It also indicates that, by dividing the entire event stream into participant streams, we have a better chance of identifying the sub-events that have otherwise been shadowed by the dominant sub-events or participants. The two participant-based methods yield similar recall but “Participant

+ Spike" yields slightly worse precision, since it is very sensitive to the spikes on the participant-level, leading to the rise of false alarms. The "Participant + MM" approach is much better in precision, which is consistent to our findings on the participant streams.

5.3 Summarization Results

Summarization evaluation has been a longstanding issue in the literature (Nenkova and Mckeown, 2011; Liu and Liu, 2010). There are even less studies focusing on evaluating the event summaries generated from data streams. Since the summary annotation takes quite some effort, we sample a 10-minute segment from each of the seven event streams and ask a human annotator to select representative tweets for each segment. We then compare the system summaries against the manual summaries using the ROUGE-1 (Lin, 2004) metric. The quantitative results and qualitative analysis are presented in Table 5 and Table 6 respectively. Note that the ROUGE scores are based solely on the n-gram overlap between the system and reference summaries, which may not be the most appropriate measure for evaluating the Twitter event summaries. However, we do notice that the accurate sub-event detection performance can successfully translate into a gain of the ROUGE scores. Qualitatively, the participant-based event summarization approach focus more on extracting tweets associated with the targeted participants, which could lead to better text coherence.

Event	Method	R(%)	P(%)	F(%)
NBA Average	Spike	14.73	23.24	**16.87**
	Participant + Spike	54.60	14.65	**22.40**
	Participant + MM	54.36	23.06	**31.53**
WWDC Conf.	Spike	26.58	39.62	**31.82**
	Participant + Spike	49.37	25.16	**33.33**
	Participant + MM	42.77	31.73	**36.07**

Table 5: ROUGE-1 scores of summarization

Method	Summary
Manual	Good drive for *durant* Pretty shot by *Duncan* Good 3 point *tony parker* Nice move *westbrook* Good shot *Westbrook*
Spike	Game 3. Spurs vs. OKC Okc and spurs game.
Participant + Spike	OKLAHOMA CITY THUNDER vs san antonio spurs!! YA I hope okc win the series. Ill hate too see the heat play San Antonio we aint in San Antonio anymore. NBA: SA 0 OKC 8, 9:11 1st.#TeamOkc San antonio spurs for 21 consecutive win? #nba Somebody Should Stop *Tim Duncan*. Pass the damn ball *Westbrook* Good 3 pointer *tony parker*!
Participant + MM	*Tim Duncan* shot is so precise *Tim Duncan* is gettin started Good 3 pointer *tony parker*! Sefalosa guarding *tony parker*. Good fucking move coach brooks *Westbrook* = 2 Fast 2 Furious Niggas steady letting *Tim Duncan* shoot *Westbrook* mid range shot is automatic

Table 6: Example summaries for an event segment. Participants are marked using *italicized* text.

6 Conclusion and Future Work

In this work, we made an initial attempt to generate event summaries using Twitter data streams. We proposed a participant-based event summarization approach which "zooms-in" the Twitter event streams to the participant level, detects the important sub-events associated with each participant using a novel mixture model that incorporates both the "burstiness" and "cohesiveness" of tweets, and generates the event summaries progressively. Results show that the proposed approach can effectively capture the sub-events that have otherwise been shadowed by the long-tail of other dominant sub-events, yielding summaries with considerably better coverage. Without loss of generality, we report results on the entire event streams, though the proposed approach can well be applied in an online fashion.

There are many challenges left in this line of research. Having a standardized evaluation metric for event summaries is one of them. In the current work, we employed ROUGE-1 for summary evaluation, since it has been shown to correlate well with the human judgements on noisy text genres (Liu and Liu, 2010). We would like to explore other evaluation metrics (e.g., ROUGE-2, -SU4, Pyramid (Nenkova et al., 2007)) and the human evaluation in future. We will also explore better ways of integrating the sub-event detection and summarization approaches.

Acknowledgments

Part of this work was done during the first author's internship in Bosch Research and Technology Center. The work is also partially supported by NSF grants DMS-0915110 and HRD-0833093.

References

James Allan. 2002. Topic detection and tracking: Event-based information organization. *Kluwer Academic Publishers Norwell, MA, USA*.

Hila Becker, Feiyang Chen, Dan Iter, Mor Naaman, and Luis Gravano. 2011a. Automatic identification and presentation of twitter content for planned events. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 655–656.

Hila Becker, Mor Naaman, and Luis Gravano. 2011b. Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 438–441.

Deepayan Chakrabarti and Kunal Punera. 2011. Event summarization using tweets. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 66–73.

Qiming Diao, Jing Jiang, Feida Zhu, and Ee-Peng Lim. 2012. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 536–544.

Jacob Eisenstein. 2013. What to do about bad language on the internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL/HLT)*.

Jennifer Foster, Ozlem Cetinoglu, Joachim Wagner, Joseph Le Roux, Stephen Hogan, Joakim Nivre, Deirdre Hogan, and Josef van Genabith. 2011. #hardtoparse: POS tagging and parsing the twitterverse. In *Proceedings of the AAAI Workshop on Analyzing Microtext*, pages 20–25.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 42–47.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-Document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 362–370.

Sanda Harabagiu and Andrew Hickl. 2011. Relevance modeling for microblog summarization. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 514–517.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proceedings of Uncertainty in Artificial Intelligence (UAI)*.

David Inouye and Jugal K. Kalita. 2011. Comparing twitter summarization algorithms for multiple post summaries. In *Proceedings of 2011 IEEE Third International Conference on Social Computing*, pages 290–306.

Lillian Lee. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 25–32.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Workshop on Text Summarization Branches Out*.

Feifan Liu and Yang Liu. 2010. Exploring correlation between ROUGE and human evaluation on meeting summaries. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1):187–196.

Fei Liu, Yang Liu, and Fuliang Weng. 2011a. Why is "SXSW" trending? Exploring multiple text sources for twitter topic summarization. In *Proceedings of the ACL Workshop on Language in Social Media (LSM)*, pages 66–75.

Fei Liu, Fuliang Weng, Bingqing Wang, and Yang Liu. 2011b. Insertion, deletion, or substitution? Normalizing text messages without pre-categorization nor supervision. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 71–76.

Fei Liu, Fuliang Weng, and Xiao Jiang. 2012. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1035–1044.

Annie Louis and Todd Newman. 2012. Summarization of business-related tweets: A concept-based approach. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*.

Adam Marcus, Michael S. Bernstein, Osama Badar, David R. Karger, Samuel Madden, and Robert C. Miller. 2011. Twitinfo: Aggregating and visualizing microblogs for event exploration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 227–236.

Ani Nenkova and Kathleen Mckeown. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2–3):103–233.

Ani Nenkova, Rebecca Passonneau, and Kathleen Mckeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing*, 4(2).

Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. 2012. Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM Interntional Conference on Intelligent User Interfaces (IUI)*, pages 189–198.

Brendan O'Connor, Michel Krieger, and David Ahn. 2010. TweetMotif: Exploratory search and topic summarization for twitter. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 384–385.

Sasa Petrovic, Miles Osborne, and Victor Lavrenko. 2010. Streaming first story detection with application to twitter. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 181–189.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1524–1534.

Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1104–1112.

Beaux Sharifi, Mark-Anthony Hutton, and Jugal K. Kalita. 2010a. Experiments in microblog summarization. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, pages 49–56.

Beaux Sharifi, Mark-Anthony Hutton, and Jugal K. Kalita. 2010b. Summarizing microblogs automatically. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 685–688.

Hiroya Takamura, Hikaru Yokono, and Manabu Okumura. 2011. Summarizing a document stream. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval (ECIR)*, pages 177–188.

Jui-Yu Weng, Cheng-Lun Yang, Bo-Nian Chen, Yen-Kai Wang, and Shou-De Lin. 2011. Imass: An intelligent microblog analysis and summarization system. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 133–138.

Siqi Zhao, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. 2011. Human as real-time sensors of social and physical events: A case study of twitter and sports games. *Technical Report TR0620-2011, Rice University and Motorola Labs*.

Arkaitz Zubiaga, Damiano Spina, Enrique Amigó, and Julio Gonzalo. 2012. Towards real-time summarization of scheduled events from twitter streams. In *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, pages 319–320.