From Event Detection to Storytelling on Microblogs

Janani Kalyanam*, Sumithra Velupillai†, Mike Conway‡, and Gert Lanckriet*
* University of California, San Diego,
{jkalyana@, gert@ece.}ucsd.edu
† KTH, Stockholm and King's College, London,
sumithra@kth.se
‡ University of Utah, Salt Lake City
mike.conway@utah.edu

***Abstract*— The problem of detecting events from content published on microblogs has garnered much interest in recent times. In this paper, we address the questions of what happens after the outbreak of an event in terms of how the event gradually progresses and attains each of its milestones, and how it eventually dissipates. We propose a model based approach to capture the gradual unfolding of an event over time. This enables the model to automatically produce entire timeline trajectories of events from the time of their outbreak to their disappearance. We apply our model on the Twitter messages collected about Ebola during the 2014 outbreak and obtain the progression timelines of several events that occurred during the outbreak. We also compare our model to several existing topic modeling and event detection baselines in literature to demonstrate its efficiency.**

I. Introduction

Detecting events from social media microblogs has garnered much interest in the past decade [1]. Several approaches, models and methodologies have been proposed to address the task of detecting events from a continuous stream of microblog posts. However, monitoring an event's development over time, and naturally building a story-line of all the topics that occurred during the course of an event still remains a challenge. While many studies have addressed the problem of event detection from micro-blogging data [2, 3, 4, 5], little effort has been dedicated to investigating the evolution of an event over time. The works which focus on event detection have given little attention to what happens *after* the initiation of an event. How does the event progress, and what are its important milestones? How does it eventually terminate?

During the pre-microblogging era (especially in the early 2000s), topic tracking algorithms [6, 7, 8] were used to track stories over time. Many approaches from this era were developed to track news stories from traditional media (newspapers). The scenario of microblogs is a very different domain in terms of its content when compared to traditional media. Microblogs are used by the general public and hence, the content is not curated for language, grammar, punctuation, spelling etc. In addition, microblogs are extremely short text documents unlike newspaper articles. Hence, the topic modeling and tracking algorithms developed in the pre-microblogging era [9, 10, 11] which rely on documents being fairly long in order to learn patterns in the data, do not easily lend themselves to the microblogging scenario.

IEEE/ACM ASONAM 2016, August 18-21, San Francisco, CA 978-1-5090-2846-7/16/$31.00 2016 IEEE

We introduce a model which aims at detecting events from a stream of microblogs, and building a timeline for each event, summarizing it as a sequence of sub-events (or topics) with a beginning, middle and end. This model aims at capturing the *gradual unfolding* of an event over its lifetime, and identifies how and when the gradual changes occurs. The problem with existing keyword-based methods for event detection is that, as an event evolves over time, it is likely that keyword-based methods (all of which rely on monitoring keyword frequencies over time, and assume the sudden burst in the frequency of a few keywords to be onset of an event) will classify the newer developments of the event as entirely new events. Specific examples of this phenomenon have also been illustrated in the examples given in the works of [2, 5]. However, in order to extract the entire story pertaining to an event, it is important for the model to identify and adapt to these 'turning' points in event.

Our model is based on the unsupervised learning approach of non-negative matrix factorization (NMF), with additional terms that explicitly model the *turning points* (if any) of an event over time, while also accounting for the distinctive characteristics of microblogs being very short, and sparse documents. Our model identifies a new emerging event, the several milestones of the event during its evolution and the disappearance of the event from the stream. We evaluate our model by studying tweets (Twitter messages) collected about Ebola during the 2014 Ebola outbreak in the United States. Our model produces a timeline of all events with the important milestones composing each event. This result matches closely with the timeline available on Wikipedia and other verified news sources about Ebola. and verified on several news outlets. In addition, we also compare the performance of our model to existing methods ([3, 12, 13, 14, 15]) on topic detection tasks to demonstrate its efficacy over them.

II. Model and Optimization

Our goal is to build a model that learns the trajectory of an event, and naturally builds a timeline from it. We clarify the usage of the terms *topics* and *events*. A *topic* refers to a coherent group of keywords detected by our model. In some scenarios, we also refer to a discrete probability distribution over the vocabulary as a *topic*. An *event*, on the other hand, is a real world occurrence in space and time. All events (and their milestones) come from meaningful topics. However, not all topics necessarily need to correspond to real world occurrences or events. For example, some topics generated by our model could be just noise, and thereby do not correspond to any real world event. In general, an event can be summarized as a sequence of chronologically ordered meaningful topics.

A. Model

Our model is built on an NMF (non-negative matrix factorization) based topic models approach. The traditional documents-by-words matrix $\mathbf{X}$ tends to be sparse for a corpus with short documents. However, it is a commonly observed phenomenon that, for corpora of very short documents, the size of the vocabulary grows at a much slower rate w.r.t the number of documents [16]. In some sense, a very large number of documents are being generated from a relatively small vocabulary. This results in the word co-occurrence matrix $\mathbf{K}$ (a square matrix, with dimension equal to the size of the vocabulary) being comparatively denser than the documents-by-words matrix $\mathbf{X}$. Hence, we consider factoring such a co-occurrence matrix in order to infer the topics. Factorizing this matrix would result in:

$$\mathbf{K} \approx \mathbf{Q}^T \mathbf{Q}. \tag{1}$$

Equation 1 can be thought of as a clustering of commonly co-occurring words into coherent topics. In our paper, we consider the squared loss $||\mathbf{K} - \mathbf{Q}^T\mathbf{Q}||_F^2$ and factorize the m-by-m word co-occurrence matrix $\mathbf{K}$ symmetrically into a k-by-m topic matrix $\mathbf{Q}$, where k is the number of topics. Like in any topic model, the parameter k of the number of topics needs to be set manually (generally by cross-validation).

In the subsequent paragraphs, we crystallize what it means to model the *gradual unfolding* of an event. In our view, the model should inform if there have been any updates to the current status of the event. This includes specifying if an existing event has remained status quo, or has had any advancements (and if so, what they are), or if the event is about to fade. It should also inform about any emerging events. We will revisit these notions again in Section III-C when we demonstrate through an example how our model learns and infers these notions of unraveling of events.

In order to accomplish modeling this view of the evolution of an event over time, we introduce a tracking matrix $\mathbf{T}$. The objective in Equation 1 is modified to take the following form (in terms of the loss function):

$$L = ||\mathbf{K}^t - {\mathbf{Q}^t}^T \mathbf{Q}^t||_F^2 + ||\mathbf{K}^t - {\mathbf{Q}^t}^T \mathbf{T}^t \mathbf{Q}^{t-1}||_F^2 + \alpha(R). \tag{2}$$

In Equation 2, note that the notation has been modified to accommodate for the scenario where we have a continuous stream of microblogs. The matrix $\mathbf{K}^t$ is the word-cooccurrence matrix for all the microblogs at time-t. The matrix $\mathbf{Q}^{t-1}$ is the k-by-m topics-by-words matrix obtained in the previous timestep. While minimizing for L, $\mathbf{Q}^{t-1}$ and $\mathbf{K}$ are considered known. The unknowns are the topic matrix $\mathbf{Q}^t$, and the current tracking matrix $\mathbf{T}^t$.

The first term in Equation 2 detects topics based on the microblogs of the current time-t. The second term detects topics based on both the microblogs at time-t ($\mathbf{K}^t$) and the topics of the previous time step ($\mathbf{Q}^{t-1}$). In some sense, it models the relationship between $\mathbf{Q}^t$ and $\mathbf{Q}^{t-1}$ through the tracking matrix $\mathbf{T}^t$. To see this more clearly, one can view this relationship as

$$\mathbf{Q}^t \approx \mathbf{T}^t \mathbf{Q}^{t-1}. \tag{3}$$

Note that $\mathbf{T}^t$ is a k-by-k matrix with all positive entries. Hence, the i-th row of $\mathbf{T}^t$ helps explain the i-th topic for time-t as a mixture of the previous topics. If a particular row of $\mathbf{T}^t$ is $\mathbf{0}$, it means that particular topic is new (since it cannot be explained by any of the previous topics). If a particular column of $\mathbf{T}^t$ is $\mathbf{0}$, it means that this particular topic is about to disappear since it has not found its presence anywhere at time-t. If $\mathbf{T}^t$ happens to be a permutation matrix (after row normalization), it indicates no evolution from $t-1$ to t. From row-i, one can essentially deduce how this topic relates to the topics from the past. In Section II-C, we further solidify how to go from this model to explicitly inferring changes in an event, and producing an actual timeline. We give our model the name Modeling Event Progression (MEP). The idea of using such matrices to *track* quantities over subsequent time steps has been used before [17, 18]. However, to our knowledge, this is the first time it is being proposed in the context of extracting timelines of events from streaming microblogging content.

B. Optimization

The optimization problem can be written as

$$\begin{aligned} \underset{\mathbf{T}^t, \mathbf{Q}^t}{\text{minimize}} \quad & L \\ \text{subject to} \quad & \mathbf{Q}^t \geq \mathbf{0}, \mathbf{T}^t \geq \mathbf{0}, \end{aligned} \tag{4}$$

where L is the loss from Equation 2[1].

The loss function in Equation 2 is not convex w.r.t. the variables of interest. In addition, the fact that the first term of L (which is quartic non-convex) involves the square of the variable $\mathbf{Q}^t$ complicates approaching the optimization via traditional NMF solvers. To address this, we treat the $\mathbf{Q}^t$s in the first term as two different matrixes: $\mathbf{Q}^t$, and $\mathbf{P}^t$. The loss function can be re-written as:

$$L = ||\mathbf{K}^t - {\mathbf{Q}^t}^T \mathbf{P}^t||_F^2 + ||\mathbf{K}^t - {\mathbf{Q}^t}^T \mathbf{T}^t \mathbf{Q}^{t-1}||_F^2 + \alpha(R). \tag{5}$$

This version of the loss function makes it readily available for NMF solvers using multiplicative update equations [15]. The updates are made for each variable, assuming the others are known (the variables are initialized with random positive values to begin with).

Solving for $\mathbf{P}^t$ is straight forward, since the objective reduces to a non-linear least squares problem. A caveat is

[1] Here the $\geq$ stands for element-wise greater-than-or-equal-to; $\mathbf{0}$ stands for the all-zeros matrix

that we require all the entries of $\mathbf{P}^t$ to be positive. So, we solve for $\mathbf{P}^t$ under the conventional least squares setting, and set all the negative values in the resulting $\mathbf{P}^t$ to zero. While this procedure lacks convergence theory, it is shown to be quite effective in practice [19]. Once we know $\mathbf{P}^t$, the update equations for the other variables can be derived following a similar procedure as in [15]. The actual algorithm for obtaining the topic matrix $\mathbf{Q}^t$ and the tracking matrix $\mathbf{T}^t$ is summarized in Algorithm 1. Note: the [.] convention implies element-wise division of the values in a matrix[2].

Data: $\mathbf{K}^t, \mathbf{Q}^{t-1}, \epsilon, \alpha,$ `maxiter`;
Result: $\mathbf{Q}^t$,$\mathbf{T}^t$.
$\mathbf{Q}^t \leftarrow$ rand
$\mathbf{T}^t \leftarrow$ rand
`iter_no` $\leftarrow 0$
while *iter_no* $<$ *maxiter* & $L > \epsilon$ **do**

$$\mathbf{P}^t[i,j] \leftarrow \max((\mathbf{Q}^{t^T}\mathbf{Q}^t)^{-1}\mathbf{Q}^{t^T}\mathbf{K}[i,j], 0)$$

$$\mathbf{Q}^t \leftarrow \quad \mathbf{Q}^t \odot \frac{[\mathbf{P}^t\mathbf{K}^{t^T} + \mathbf{T}^t\mathbf{Q}^{t-1}\mathbf{K}^T]}{[\mathbf{P}^t\mathbf{P}^{t^T} + \mathbf{T}^t\mathbf{Q}^{t-1}\mathbf{Q}^{t-1^T}\mathbf{T}^t + \alpha\mathbf{e}\mathbf{e}^T]}$$

$$\mathbf{T}^t \leftarrow \quad \mathbf{T}^t \odot \frac{[\mathbf{Q}^t\mathbf{K}^t\mathbf{Q}^{t-1}]}{[\mathbf{Q}^t\mathbf{Q}^{t^T}\mathbf{T}^t\mathbf{Q}^{t-1}\mathbf{Q}^{t-1^T} + \alpha\mathbf{e}\mathbf{e}^T]}$$

`iter_no` $\leftarrow$ `iter_no` $+ 1$
$L \leftarrow L(\mathbf{P}^t, \mathbf{Q}^t, \mathbf{T}^t)$

end

Algorithm 1: Optimization

C. Timeline Generation

Recall that, in our view of modeling the progression of an event, we want the model to explicitly inform us whether an event has remained status quo, if the event has progressed (and if so, what the advancements are), or if it is about to fade.

In this section, we will present how this gradual unfolding can be inferred from this model. The tracking matrix attempts to relate each row in $\mathbf{Q}^t$ as a linear combination of the rows in $\mathbf{Q}^{t-1}$ (Equation 3), thus informing how each topic at time-t is composed of topics from time-$(t-1)$. Note that, since all the values are positive, one can normalize each row, as such:

$$\mathbf{T}^t[i,:] = \frac{\mathbf{T}^t[i,:]}{\sum_{j=1}^{j=k}\mathbf{T}^t[i,j]}.$$

Each row of $\mathbf{T}^t$ is now a probability mass function (pmf) over k items. We can think of the topic associated with row-i at time-t as a random variable $T_{t,i}$. The entropy of the $T_{t,i}$ is defined as:

$$H(T_{t,i}) := -\sum_j \mathbf{T}^t[i,j]\log(\mathbf{T}^t[i,j]).$$

This entropy quantifies how much topic-i has metamorphosized from time-$t-1$ to time-t. We consider three scenarios.

- If row-i has all its mass concentrated in one cell, the resulting $H(T_{t,i})$ would be very low. This suggests that it is likely that topic-i is a continuation of an earlier topic without much change. Hence, this event has remained status quo.
- If the un-normalized values of row-i happen to be very small, it indicates that topic-i cannot be represented well in terms of any of the earlier topics (likely marking the beginning of a new event). This makes the pmf of $T_{t,i}$ to be somewhat equally distributed across all the cells, resulting in a high value of $H(T_{t,i})$.
- A moderate value of $H(T_{t,i})$ indicates that $T_{t,i}$ has its pmf spread over a few elements suggesting that topic-t is a combination of a few topics from the past. This informs about the evolution of the event at time-t from time-$(t-1)$.

At each instance, the metamorphosis of the events are quantified and deduced from the entropy information. In addition, if an entire column of $\mathbf{T}^t$ has small values, it is indicative of a dissipating event as it has not found its presence anywhere in time-t. This timeline generation process is explained with an example in Section III. Note: A *topic* refers to a coherent group of keywords (as learned by the rows of $\mathbf{Q}$). An *event* is an temporal ordering of topics.

III. Experiments

We begin by providing details about the dataset used in this study. Subsequently, we delve into the experiments that (a) compare our model to various baselines on a topic detection task, and (b) demonstrate through an example how the gradual unfolding of an event is captured by our model, and illustrate all the events and their progressions detected by our algorithm.

A. Twitter-Ebola Dataset

From the Twitter-Ebola Dataset [20], all the non-English tweets were discarded using the `user_lang` and the `lang` functionality. The text of the tweets were preprocessed to remove any URLs, special characters and hashtags. A vocabulary count was built from the remaining tweets, and all the word types that appeared fewer than 41 times were discarded. Tweets with only one token were also discarded. The resulting dataset spans from $2014/10/4$ to $2014/10/31$ totalling 10.5M tweets and 29000 word types. This amounts to approximately 375K tweets per day. After all the data pre-processing such as stopword and infrequent word removal, the average length of a tweet is approximately 5 words. In order to apply our MEP algorithm, we split the tweets into smaller blocks pertaining to a particular time period.

B. Topic Detection

The goal of the first set of experiments is to compare MEP to several existing baselines in the event detection literature and topic modeling literature on topic detection tasks. Ideally, a groundtruth for such a task can be obtained by having human annotators mark the topics for each tweet. In this study, we choose to use the *hashtags* as proxy for such annotations [21][3].

[2]An implementation of our algorithm can be found here: `https://github.com/kjanani/matrix_factorization`

[3]As mentioned in Section III-A, all hashtags were removed from the data and have no bearing on the learning.

$k = 5$

model type	model	NDCG	MAP
[Ours]	MEP	0.2027	0.0953
event detection	trend-detect	0.1823	0.0862
	o-cluster	0.1677	0.0892
topic modeling	O-BTM	0.1745	0.091
	nmf	0.1722	0.0864
	lda	0.1245	0.0589

$k = 7$

model type	model	NDCG	MAP
[Ours]	MEP	0.1626	0.0706
event detection	trend-detect	0.1502	0.0539
	o-cluster	0.1310	0.0534
topic modeling	O-BTM	0.1459	0.0569
	nmf	0.1306	0.0565
	lda	0.0837	0.0366

$k = 10$

model type	model	NDCG	MAP
[Ours]	MEP	0.1430	0.0696
event detection	trend-detect	0.1379	0.0667
	o-cluster	0.1320	0.0606
topic modeling	O-BTM	0.1271	0.0412
	nmf	0.1057	0.0463
	lda	0.0660	0.0164

TABLE I

SUMMARY OF RESULTS FROM TOPIC DETECTION TASK.

In order to facilitate a common ground for evaluating our model and all the baselines, we choose to describe each topic obtained by the models as a ranking of the most representative 10 words present in it. Our model produces the topic matrix $\mathbf{Q}^t$ at every timestep. From this topic matrix, one can immediately obtain a ranking of the top-10 words in each topic. As we will see shortly, an equivalent ranking for each topic can be obtained from all the baselines as well. A comparison between the rankings obtained by each model, and a ranking provided by the groundtruth would then quantify the performance of each model. These rankings are compared using the scores Normalized Cumulative Discounted Gain (NDCG), and Mean Average Precision (MAP). We obtained the groundtruth ranking by first averaging the bag-of-words representation of all the tweets containing a particular hashtag, and then identifying the top words from this average representation.

Now, we briefly describe each of the baselines, and explain how a ranking of the top 10 words was obtained for each topic. The baselines can be broadly classified into two subgroups of works: one stemming from the event detection literature, and another from the topic modeling literature.

1) Event Detection Baselines: [3] (o-cluster) use an incremental online clustering approach to detect events. We used the metric and threshold for similarities as suggested by the authors. Once all the clusters were obtained using incremental online clustering, a ratio of inter-tweet-distances to the intra-tweet-distances was calculated for each cluster. The k clusters with highest scores were considered to be the topics detected, and a ranking of the most frequent 10 words were obtained for each. This model lacks the notion of time. [12] (trend-detect) use an online probabilistic topic models approach for trend detection. We used the code provided by the authors to obtain the topics at each time step. The code automatically provides a ranking of the top-10 words in each topic. This model incorporates a notion of time, but it is primarily designed to facilitate a better starting point for the next time slice, than to actually categorically identify the changes that take place in an event over time.

2) Topic Modeling Baselines: [22] (O-BTM) use a modification of latent dirichlet allocation to detect trending topics from tweets. As proposed originally, this model lacks a notion of time. However, in their follow-up work of [13], they introduced a online version of which uses the knowledge about topics discovered in the previous timestep to its new parameters in the current timestep. In addition to the above models, we compare to classical algorithms like NMF [15], and LDA [14].

The experiments were performed for k values of $5, 7$, and 10. The results are summarized in Table I. In general, the performance decreases with increase in k. This corroborates the phenomenon in any multiclass class problem that, as the number of classes increases, the model finds it more challenging to learn to distinguish between these classes simultaneously. Secondly, for the topic modeling baselines, we note that BTM performs better than LDA and NMF. BTM is a model based on LDA with modifications specifically geared towards short text documents. Our dataset has documents with 5 words/document on average. This could explain why a model specifically designed for short texts clearly has an edge over the others.

Our model MEP generally seems to outperform all the baselines. An important reason that we attribute to the success of MEP is that, our dataset exhibited those characteristics that we initially had set out to model. The baselines had different goals. The algorithm o-cluster [3] sets out to *identify* events from social media, and not necessarily model their progression over time. The Twitter-Ebola dataset was collected in October 2014, during the height of the Ebola 2014 crisis in the United States. As we will see in the next section, our dataset consisted of events and stories that span over multiple days. Hence, for a dataset that has such strong temporal characteristics, explicitly modeling them seems to definitely help in terms of detecting topics.

C. The Story Telling Experiment

topic at $(t-1)$	top words
1	ur, watching, disney, channel, africa, people, retweet
2	patient, cdc, fighting, condition, homeless, officials
3	simpsons, oct, 1997, predicting, goodluck
4	ur, watching, disney, channel
5	dallas, patient, critical, homeless
topic at (t)	**top words**
1	cat, giving, speech
2	patient, nebraska, hospital, experimental, treatment
3	ur, watching, channel, disney
4	africa, people, retweet, spain, nurse, positive
5	watching, disney, channel, ur

TABLE II

THE TOP FEW WORDS FROM EACH OF THE 5 TOPICS OBTAINED ON DATES 2014/10/5 (REPRESENTED AS TOPIC-$t-1$) AND 2014/10/6 (REPRESENTED AS TOPIC-(t). THIS TABLE MUST BE STUDIED IN CONJUNCTION WITH FIGURE 2.

.

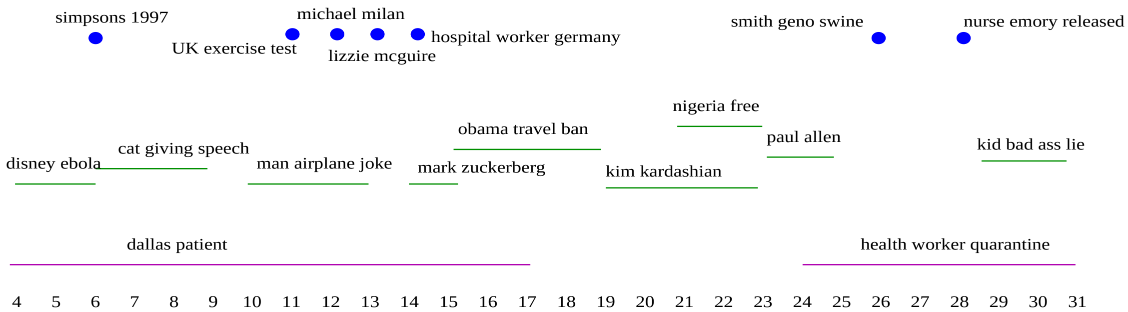

Fig. 1. Timeline of events discovered from the Twitter Ebola Dataset.

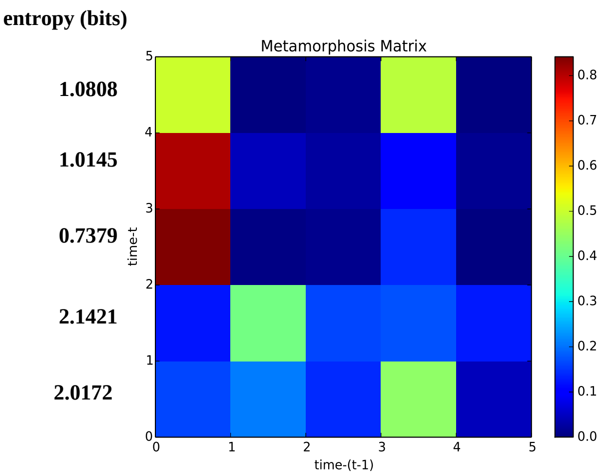

Fig. 2. A colored heatmap of the matrix $\mathbf{T}^t$, where $t = 6$. The x-axis represents topics at time-$t-1$, and the y-axis represents topics at time-t. The top words from these topics are shown in Table II.

We start by demonstrating an example from the Twitter-Ebola dataset of how the gradual unfolding of events are learned by the model. The example is illustrated though Figure 2 and Table II. Figure 2 displays the heatmap of the tracking matrix $\mathbf{T}^t$ learned on $2014/10/6$. Each row of $\mathbf{T}^t$ is the distribution of the random variable $T_{i,t}$, and represents the topics on $2014/10/06$ in terms of the topics of $2014/10/05$. The numbers that appear with each row (on the left) is the entropy of the $T_{i,t}$s. The top words describing each of the topics at time-t and time-$(t-1)$ are summarized in Table II. The number of topics k was set to 5.

Recall from Section II-A that in our view of modeling an event's progression, we expect the model to inform us about new events, or about the status changes of existing events (if any, and what they are), or if an event is about to fade. In Figure 2, $T_{3,t}$ has the lowest entropy (note: largest possible entropy in this case is approximately 2.8). It also has the largest mass in the first cell. This suggests that topic-3 of time-t is perhaps just a continuation of topic-1 of time-$(t-1)$. The top words of topic-3 in time-t and topic-1 in time-$(t-1)$ (see Table II) also have a large portion of overlapping words ('ur','watching','disney','channel'). Hence, this perhaps indicates that the event associated to these topics is continuing to the next timestep without much change. As it turns out, this event actually corresponds to a meme that spread widely on Twitter during the Ebola outbreak [23]

$T_{1,t}$ and $T_{2,t}$ have the two highest entropies. It is also evident from the figure that the pmf for these two random variables is much more evenly spread out over the previous topics. This suggests that that these topics are perhaps new, and could be indicative of emerging events since they cannot be represented well by any combination of topics from the previous timestep. This is corroborated by examining the top words corresponding to these topics from Table II. For topic-1 at time-t, the set of top words ('cat','giving','speech') do not appear anywhere in time-$(t-1)$. Hence, this is indeed a new topic. And as it turns out, the event related to this topic was also a meme that was wide spread on Twitter during the Ebola outbreak. It had indeed emerged on the $2014/10/06$ [23]. Similarly, for topic-2 at time-1, the top words ('nebraska','hospital','experimental','treatment') do not appear anywhere in time-$(t-1)$. Hence, this also seems to be a new emerging event. It turns out on $2014/10/06$, an experimental treatment was carried out for one of the Ebola patients in Nebraska.

$T_{4,t}$ and $T_{5,t}$ have entropies that fall somewhere in between. For $T_{5,t}$, the pdf is shared predominantly between cells 1 and 4. The top words from the table also yield the same finding that topic-5 at time-t emanates from both topics 1 and 4 in time-$(t-1)$. Similarly for $T_{4,t}$, both the figure and table indicate that the topic is moving from the meme about ('disney','channel') to one about the spanish nurse contracting Ebola. These types of changes are the ones that indicate the gradual unfolding that occurs during the lifetime of an event.

Recall that a particular column of the heatmap explains the presence of a certain topic at time-t. While examining the heatmap in Figure 2, observe that columns 3 and 5 have all values close to zero. This indicates these topics (and the related events) are vanishing. This is also supported by the words in Table II; the words of topics-3 and 5 at time-$(t-1)$ do not appear anywhere in time-t.

Using this insight derived from the tracking matrix and the entropies, we categorically identify emerging, gradually changing, continuing and dissipating events at every time step. Specifically, for $k = 5$, the following thresholds on entropy were used to identify the gradual changes: (a) $H(.) \leq 1$

indicated continuing events; (b) $1 \leq H(.) \leq 2$ indicated evolving events; (c) $H(.) \geq 2$ indicated new events. If a topic from time-$(t-1)$ did not find its presence in time-t, that event was categorized as fading. In addition, to obtain the predominant composition of evolving events, we identified the main topics form the previous timestep that contributed to at least 80% of the current topic. Using these rules at each time step, we produced a demarcation for all topics. After repeating this for all the timesteps, we produced the timelines for each event by following the demarcations.

We discussed the challenge of validating topic detection tasks in Section III-B. Now, we again face a similar, albeit a lot harder challenge of validating the timeline of entire events. To the best of our knowledge, there are no existing ways to quantify the task of evaluating the gradual changes and progression of events. Hence, we chose a more qualitative and manual approach to evaluating the timelines. Once all the timelines were obtained, we validated them by verifying on news sources and Wikipedia to make sure that the events and their timelines were indeed true. We present the timeline obtained from our Twitter Ebola dataset in Figure 1. Each event is represented by a colored line and a few words that best describe it. What we noticed from the process of creating this timeline is that certain events (predominantly memes) do not evolve much over the course of their lifetime. There was a Twitter meme with the following text: "More Americans have been married to Kim Kardashian than have died from Ebola". This meme persisted for close to 5 days on twitter. The text corresponding to this topic did not change much from one time step to the next. Hence, this topic was constantly categorized as 'continuing' in our model. Similar observations held true for other memes such as the ('cat','giving','speech') meme, or ('disney','watching','channel').

Certain other events, which were generally long term events (lasting more than 4 consecutive days) had several variations and milestones in their lifetime. Take the example of the ('dallas','patient') event. This event followed the topics surrounding the first case of Ebola detected in the US soil. This event lasted approximately 11 days. The top words detected by our model about this event are summarized in Table III. By examining these words, we observes that they corresponds well to what took place during that time. For example, according to news sources, the patient died on $2014/10/08$. And on $2014/10/10$, the first news about the nurse who treated the patient possibly having contracted Ebola emerged, and so on.

In conclusion, modeling the progression of events and extracting their story can lead to interesting insights regarding the phenomena associated with an event. To the best of our knowledge, while substantial effort has been dedicated to detecting events, ours is the first work to focus on following the trajectories of events and extracting their timelines.

2014/10/07	kidney, dialysis
2014/10/08	thomas, eric, duncan, died, first, patient
2014/10/09	died, patient
2014/10/10	duncan, fever, nurse
2014/10/11	nurse, symptoms
2014/10/12	health, care, worker, positive
2014/10/13	health, care, worker, protocol
2014/10/14	nurse, dallas, nina, pham
2014/10/15	health, care, worker, 2nd, positive
2014/10/16	nurse, flight, ohio
2014/10/17	virus, flight, nina, pham

TABLE III

THIS TABLE LISTS SOME OF THE TOP WORDS THAT OCCURRED IN THE EVENT DALLAS PATIENT ACROSS THE DAYS.

REFERENCES

[1] F. Atefeh and W. Khreich, "A survey of techniques for event detection in twitter," *Comput. Intell.*

[2] R. Li, K. H. Lei, R. Khadiwala, and K. C.-C. Chang, "Tedas: A twitter-based event detection and analysis system," in *ICDE 2012*.

[3] "Learning similarity metrics for event identification in social media," in *WSDM '10*, 2010.

[4] M. Naaman, H. Becker, and L. Gravano, "Hip and trendy: Characterizing emerging trends on twitter," *JASIST 2011*.

[5] M. Mathioudakis and N. Koudas, "Twittermonitor: Trend detection over the twitter stream," in *SIGMOD 2010*.

[6] "Topic detection and tracking evaluation overview," in *Topic Detection and Tracking*, ser. The Information Retrieval Series, J. Allan, Ed., 2002, vol. 12, pp. 17–31.

[7] V. Lavrenko, J. Allan, E. DeGuzman, D. LaFlamme, V. Pollard, and S. Thomas, "Relevance models for topic detection and tracking," in *HLTR 2002*.

[8] T. Leek, R. Schwartz, and S. Sista, "Probabilistic approaches to topic detection and tracking," in *Topic Detection and Tracking*, 2002.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[10] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature 1999*.

[11] D. M. Blei and J. D. Lafferty, "Dynamic topic models," ser. ICML '06, 2006.

[12] J. H. Lau, N. Collier, and T. Baldwin, "On-line trend analysis with topic models:\# twitter trends detection topic model online." in *COLING 2012*.

[13] X. Cheng, X. Yan, Y. Lan, and J. Guo, "Btm: Topic modeling over short texts," *TKDE 2014*.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *JMLR*.

[15] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *NIPS*, 2000.

[16] X. Yan, J. Guo, S. Liu, X. Cheng, and Y. Wang, *Learning Topics in Short Texts by Non-negative Matrix Factorization on Term Correlation Matrix*.

[17] J. Kalyanam, A. Mantrach, D. Saez-Trumper, H. Vahabi, and G. Lanckriet, "Leveraging social context for modeling topic evolution," in *KDD 2015*. ACM.

[18] C. K. Vaca, A. Mantrach, A. Jaimes, and M. Saerens, "A time-based collective factorization for topic discovery and monitoring in news," ser. WWW '14, 2014.

[19] M. W. e. Berry, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational statistics & data analysis*, 2007.

[20] J. Kalyanam, S. Velupillai, S. Doan, M. Conway, and G. Lanckriet, "Facts and fabrications about ebola: A twitter based study," *ACM KDD Workshop on Connected Health in the Big Data Era*, 2015.

[21] O. Tsur and A. Rappoport, "What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities," in *WSDM, 2012*.

[22] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *WWW 2013*.

[23] Bustle, http://www.bustle.com/articles/45810-17-ebola-internet-meme.