On-line Trend Analysis with Topic Models: #twitter trends detection topic model online

JeyHan Lau[1,2] *Nigel Collier*[3] *Timothy Baldwin*[1,2]
(1) Dept of Computing and Information Systems, The University Melbourne, Australia
(2) NICTA Victoria Research Laboratory, Australia
(3) National Institute of Informatics, Japan
jhlau@csse.unimelb.edu.au, collier@nii.ac.jp, tb@ldwin.net

ABSTRACT
We present a novel topic modelling-based methodology to track emerging events in microblogs such as Twitter. Our topic model has an in-built update mechanism based on time slices and implements a dynamic vocabulary. We first show that the method is robust in detecting events using a range of datasets with injected novel events, and then demonstrate its application in identifying trending topics in Twitter.

KEYWORDS: Topic Model, Twitter, Trend Detection, Topic Evolution, Online Processing.

Proceedings of COLING 2012: Technical Papers, pages 1519–1534,
COLING 2012, Mumbai, December 2012.

1 Introduction

In recent years, microblogs such as Twitter (http://www.twitter.com/) have emerged as a highly popular form of social media. Microblogs provide a platform for users to post short messages (a.k.a. "tweets") for followers to read in an on- or off-line fashion. One of the major attractions of microblogs is their contextual immediacy, i.e. they provide "here and now" access to users, including the potential to geotag tweets for the location of the post. Twitter is used in a variety of ways, from posting about personal daily life events to finding jobs to keeping up to date with news events.[1]

As microblogging services have gained in popularity, a myriad of applications that perform analysis of what is "trending" at a given point in time have emerged, with examples including Trendsmap (http://trendsmap.com/), What the Trend (http://whatthetrend.com/), twinitor (http://www.twinitor.com/) and Twendr (http://twendr.com/). There are a number of reasons for tracking trends. At the end-user level, it provides a way for users to identify popular discussions to follow or participate in. From a social science perspective, it provides insights into how new trends emerge, the half-life of trends, and the types of topics commonly discussed in the Twittersphere, etc.

Applications that track trends commonly use a keyword-based approach, and provide output in the form of simple terms, hashtags or term *n*-grams. While these keywords are indicative of the subject of the trending topic, ultimately they fall short in providing users with a fine-grained insights into the nature of the event. This motivated us to look for an alternative means of analysing and presenting trends, and ultimately led us to look at topic models as a potential solution.

Our contributions in this work are as follows. We first describe a topic model that processes documents in an on-line fashion. The model has the important properties that it does not grow over time, and can cope with dynamic changes in vocabulary. We then describe a method to measure shifts in the topic model, in order to track emerging events. We demonstrate the robustness and accuracy of the model using a suite of synthetic datasets based on Twitter and data from the TREC Topic Detection and Tracking (TDT) corpus, and then apply it to a series of Twitter feeds to detect popular topics in particular locations, which we find closely track local and global news events. The associated topics, in the form of a multinomial distribution over terms, are also more descriptive than single hashtags or strings.

2 Background

In recent years, there has been a surge of interest in event detection, due to the ready accessibility of document streams from newswire sources and social media. It has seen applications in many areas, such as the tracking of influenza (Signorini et al., 2011) and harvesting of spatio-temporal information for forest fires (De Longueville et al., 2009). Event detection occurs in two forms: (1) retrospectively, assuming the full document collection as input; and (2) on-line, processing documents dynamically as they arrive.

Retrospective event detection in microblogs provides insights about events that occurred in static sets of historical data. Much of the early work on retrospective event detection took place in the context of the TREC Topic Detection and Tracking (TDT) task (Allan, 2002), e.g. using document clustering and anomaly detection methods. If we wish to detect events happening presently in

[1] http://webtrends.about.com/od/twitter/a/why_twitter_uses_for_twitter.htm

our time, however, we require on-line event detection models. An example application where real-time responsiveness is critical is earthquake detection (Sakaki et al., 2010), and trend analysis also clearly requires on-line processing in order to be of use (Mathioudakis and Koudas, 2010). Most on-line approaches, however, use a relatively simple keyword-based methodology over a pre-defined set of keywords (Culotta, 2010; Lampos and Cristianini, 2010; Weng and Lee, 2011; Zhao et al., 2012) rather than tackling the more challenging task of open-world event detection.

Real-time first story detection (Petrović et al., 2010; Osborne et al., 2012) is the task of detecting the mentions of a breaking story as close as possible in time to its first mention. Here, the system should ideally pick up on the breaking story within seconds or minutes of its first mention in order to have impact, e.g. as an alert system for a newswire agency or intelligence organisation. As such, the methods that are standardly applied to the task tend to be based on analysis of local "burstiness" in the data, e.g. through locality sensitive hashing. Our work differs from theirs in that we wish to identify trends or topics that occur with a significant proportion in the data—which is different from trying to detect the very first mention of these topics. In our case, timeliness of detection is not as critical.

Bursty term analysis has obvious limitations in identifying events, both in that it fails to capture the fact that multiple terms may be involved with the same event (Zanzotto et al., 2011), and requires that at least one term undergoes a sufficiently high jump in relative frequency that the event can be identified. Topic models have been proposed as a means of better capturing events, by way of learning *clusters* of terms that are associated with a given event, as well as modelling changes in term co-occurrence rather than just term frequency. Most work based on topic modelling has been in the form of retrospective event detection models, however (Kireyev et al., 2009; Diao et al., 2012).

Moving to the more general area of the machine learning, several online topic models have been proposed (Hoffman et al., 2010; AlSumait et al., 2008). Hoffman et al. (2010) introduced an online LDA variant that uses variational Bayes as the approximate posterior inference algorithm. The model that is closest in spirit to what we propose is On-Line LDA (OLDA) (AlSumait et al., 2008). Using collapsed Gibbs sampling for approximate inference, OLDA processes documents in an on-line fashion by resampling topic assignments for new documents using parameters from a previously learnt model. We return to compare OLDA with our proposed method in Section 3.3.

3 Methodology

We first provide background on LDA topic modelling in Section 3.1. Next we describe our proposed online variant of LDA in Section 3.2, and contrast it with Online LDA in Section 3.3. Lastly, we explain how our topic model can be used to detect emerging topics in Section 3.4.

3.1 LDA Topic Model

LDA is a generative model that learns a set of latent topics for a document collection (Blei et al., 2003). The input to LDA is a bag-of-words representation of the individual documents, and the output is a set of latent topics and an assignment of topics to every document in the collection. Formally, a topic is a multinomial distribution of words, and a document is associated with a multinomial distribution of topics. A summary of LDA variables is presented in Table 1.

In topic models, the generative process for a word is as follows: first choose a topic, then sample

Variable	Dimension and Type	Description
T	Integer	Number of topics
W	Integer	Number of unique words (vocabulary)
D	Integer	Number of documents
N	Integer	Number of tokens
θ	$D \times T$ of probabilities	Topic distribution in documents
ϕ	$T \times W$ of probabilities	Word distribution in topics
$\boldsymbol{\alpha}$	$D \times T$ of α priors	Dirichlet prior for θ
$\boldsymbol{\beta}$	$T \times W$ of β priors	Dirichlet prior for ϕ
$\boldsymbol{w}$	N-Vector of word identity w	Words in documents
$\boldsymbol{z}$	N-Vector of topic assignment z	Topic Assignment of Words

Table 1: A summary of variables used in LDA.

a word from the given topic. Blei et al. (2003) introduced Dirichlet priors to the generative model, and used variational Bayes to learn θ and ϕ by maximising the probability of words in the collection. Griffiths and Steyvers (2004) proposed using collapsed Gibbs sampling to do approximate inference by considering the posterior distribution over the assignments of words to topics ($P(\boldsymbol{z}|\boldsymbol{w})$). Summarising the derivation steps, the update function in the sampling process for a new topic assignment of a word can be expressed as follows:

$$P(z = t|\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \frac{n(d,t) + \alpha}{n(d,\cdot) + T\alpha} \frac{n(t,w) + \beta}{n(t,\cdot) + W\beta}$$

where $n(d,t)$ is the number of assignments of topic t in document d, and $n(t,w)$ is the number of assignments of word w to topic t; all counts exclude the current assignment z.

3.2 Online Processing Variant

LDA processes the data in a single batch to learn the topic assignments. To facilitate the processing of streamed text, we need a model that: (1) processes the input and updates the model periodically; (2) produces topics that are comparable for different periods so that topic shift/evolution is measurable; and (3) does not grow in size with time (to ensure that it stays sensitive to topic changes over time).

We first introduce a few concepts needed for the model. Time in the model is discretised into slices, and documents (i.e. the input data) are partitioned into time slices. For example, a time slice can be an hour, a day, or a year. Denoting each time slice as k_t, k_0 is the first time slice. L is a sliding window that keeps documents for a fixed number of time slices. As documents in the new time slice arrive, documents in older time slices are discarded, so that length of the window, $|L|$, remains constant. The rationale of this approach is that we require a model that is constant in size; storing the complete document stream history data would cause the model to grow indefinitely over time, and become increasingly insensitive to topic changes.

At the arrival of new documents for time slice k_{t+1}, we update the model by resampling the topic assignments $\boldsymbol{z}$ for all documents in window L (Equation 1), using θ and ϕ from the previous model in slice k_t to serve as Dirichlet priors $\boldsymbol{\alpha}'$ and $\boldsymbol{\beta}'$ in the new model in slice k_{t+1}. The contribution factor, c, determines the degree of contribution of learnt parameters to the priors of the new model. c ranges from 0 to 1: $c = 0$ means the model is run without using any previously learnt parameters. The introduction of c is key in enabling the model to have a set of constantly evolving topics. In other words, it dampens the rich-gets-richer dynamic of the Chinese Restaurant Process in LDA.

1. Initial step:
   (a) Set Dirichlet priors α_0 and β_0; topic number T; contribution factor c; time slice k; and window size $|L|$;
   (b) Given $|L| = l$, window L contains documents from slices $k_0, ..., k_{l-1}$;
   (c) Run LDA for documents in window L;
2. Iterative step for each k_{l+i}:
   (a) Add documents in slice k_{l+i} to window L;
   (b) Remove documents in slice k_i from window L, updating θ and ϕ from the previous model as necessary;
   (c) Re-generate vocabulary for documents in window L;
   (d) Calculate priors $\boldsymbol{\alpha}'$ and $\boldsymbol{\beta}'$ as per Section 3.2;
   (e) Resample $\boldsymbol{z}$ using $\boldsymbol{\alpha}'$ and $\boldsymbol{\beta}'$ for documents in window L as per Equation 1.

Table 2: Work flow of the online processing model.

As a true online model, it would not be appropriate to assume a fixed vocabulary across time. The importance of having a dynamic vocabulary is motivated by the fact that we are interested in detecting emerging events in the data stream, where new words are likely to appear and be associated with new events (e.g. in the form of names of key people or places). To accommodate this, at every update we process the documents in the time window to re-generate the vocabulary, removing words that fall below a pre-defined frequency threshold and adding new words that now satisfy it.[2]

The Dirichlet priors $\boldsymbol{\alpha}'$ and $\boldsymbol{\beta}'$ in the new model in slice k_{t+1} are calculated as follows:

For previously seen documents and words:

$$\alpha'_{dt} = \frac{n(d,t)}{N_{old}} \times D_{old} \times T \times \alpha_0; \qquad \beta'_{tw} = \beta_0 \times (1-c) + \frac{n(t,w)}{N_{old}} \times T \times W_{new} \times \beta_0 \times c \tag{1}$$

For new documents and words:

$$\alpha'_{dt} = \alpha_0; \qquad \beta'_{tw} = \beta_0$$

where α'_{dt} is the prior for topic t in document d; β'_{tw} is the prior for word w in topic t; $n(d,t)$ and $n(t,w)$ are counts from the previous model in slice k_t; α_0 and β_0 are the default uniform prior values for θ and ϕ; and D_{old}, N_{old} and W_{new} are the number of previously processed documents, number of tokens in those documents and number of vocabulary, respectively, in time window L. The rationale behind the normalisation approach is to maintain a constant sum of priors across different batches of processing, i.e. $\sum \boldsymbol{\alpha}' = \sum \boldsymbol{\alpha} = D \times T \times \alpha_0$, and $\sum \boldsymbol{\beta}' = \sum \boldsymbol{\beta} = T \times W \times \beta_0$.

The work flow of the model is presented in Table 2.

3.3 Comparison with On-line LDA

One key difference between our proposed model and On-line LDA (OLDA) is the transfer of parameters from a previously learnt model to the updated model. In OLDA, ϕ counts from the

[2] In all our experiments, we filter our all words that occur less than 10 times across all documents in the window.

previous model are used directly or normalised as priors in the new model. To avoid topics converging after a number of iterations, OLDA removes the the $\boldsymbol{\beta}$ prior counts in ϕ of the previous model before importing them into the new model. Our model handles the converging issue in a more elegant way, by introducing a parameter (contribution factor, c) to dampen the influence from the previous model.

The second difference is that OLDA assumes a fixed vocabulary. While this is a convention in topic models, it is not an appropriate assumption for a genuinely online application, where it is impossible to pre-calculate the vocabulary ahead of time. As emerging events are likely to contain critically-relevant phrases and terms (e.g. the name of a hitherto-unknown key figure in an event, the name of a natural disaster, or the little-known location of an event), the vocabulary of our proposed model is re-generated at each update: new words are added and previously-seen words that drop below our frequency threshold are removed.

3.4 Detection of Novel Topics

At every model update, the word distribution in topics (i.e. ϕ) changes, however a one-to-one correspondence between topics is maintained across adjacent updates (provided that $c \neq 0$). Topics can thus be viewed as constantly evolving as new documents are processed: topics that are rarely or not observed in the updated document set will fade away, replaced by newly-emerged topics.

To detect these novel topics, we calculate the degree of change, or, the *evolution* of a topic using the Jensen-Shannon divergence measure between the word distribution of each topic t before and after an update, and classify a topic as being *novel* if the measure exceeds a threshold.

4 Synthetic Dataset Experiments

4.1 Generation of Synthetic Data

Ultimately we are interested in applying our method to real-world document streams, ideally based on a microblog such as Twitter. For evaluation purposes, however, we require a document stream where we have document-level annotations of: (1) whether it mentions an event of potential interest; and (2) if it mentions an event, what that event is (in the form of an event ID, potentially shared across multiple documents over a time period). In the absence of such a dataset, and given the prohibitive expense in exhaustively annotating such a dataset over the volume of data that comes through Twitter, we created a suite of synthetic datasets. Other than annotation cost, one advantage of using a synthetic dataset is that it gives us the flexibility to generate events with different distributional properties.

Having said that we are resigned to generating a synthetic dataset, we want the data to mimic as closely as possible the actuality of event mentions on Twitter. To this end, we take a document stream from Twitter and replace the message content of tweets nominally relating to a particular event (based on hashtag analysis) with distinct tweet-length mentions of events from the Topic Detection and Tracking corpus (TDT3)[3] dataset. That is, our datasets contain "background events" in the original Twitter document stream that we don't have annotations for, and "novel events" from TDT that we do have annotations for, and that form the basis of our evaluation.

In detail, the following steps were taken to generate the background event dataset:

[3] `http://projects.ldc.upenn.edu/TDT3/`

Event Type	Document Content
Background	@allabouttaurus : be realist we see thing for what they be not what they could be .
Background	ugh i be go to be so sore tomorrow
Background	rt @pagswagxo : next status i see about m . burn and i be gonna go insane .
Background	! ! saatnya mencarus wifi supaya ipod touch gw conect ke internet , dan ngetweet via twitter for iph one ,
Novel	the kosovo information center claim serb police be pass out weapon to serb civilian in the region .
Background	rt @rickyricchi : rt @atikaftri : jan lupa nya jan lupa juga mention yaw ^ ^
Background	@1aurenheilman lol , do you spell "pet peeve " wrong on purpose ?
Background	had2let it be know ! & thanks for txtn back -___ - rt @phliwidapencil lmfao rt @skrillafoccapo : all big booty aint good big booty ! !
Background	rt @desintadict_cb : rtif u want follower (cont) http ://t.co/joej7wfz
Background	well i know where all my christmas money be go . municipal court of jasper .

Table 3: An example of 10 documents in the synthetic dataset for KIM-MILOSEVIC (mapped TDT3 Topic: Holbrooke-Milosevic Meeting).

1. Collect Tweets from September 2011 to January 2012.[4]
2. Identify a set of hashtags that occur over 80% of the days in this 5-month period.[5]
3. Designate these hashtags as the set of candidates for the background events. The treatment of these hashtags as background events is based on the assumption that hashtags are generally associated with events or topics in Twitter.
4. Randomly select a subset of hashtags from the candidate set, biased according to frequency, i.e. popular hashtags are more likely to be selected than rarely-used hashtags.
5. Extract out all messages tagged with the selected hashtags for a given time period (see below), and remove the hashtag.

The novel events were embedded into the dataset as follows:

1. Select a news event that occurred during the time of the crawl, across a range of genres ranging from natural disasters to celebrity deaths to terrorist attacks (see Table 4 for details).
2. Identify a set of Twitter hashtags that correspond to the particular news event. Tweets that contain these hashtags constitute the individual novel documents.
3. Select a TDT3 topic that is of a similar genre to the news event (to keep the injected documents as similar as possible in content to the messages they replace). All topics were events that took place in 1999.
4. Replace each identified tweet with a sentence sampled randomly from the TDT3 documents labelled with the selected event topic (removing the original hashtag in the process).

Our justification for this procedure is two-fold:

- *to generate tweet-level annotations for each event, across a range of event genres* — While the original tweets had one of the pre-identified hashtags, they could have been spam or hijacking the primary topic associated with the hashtag (i.e. we are attempting to ensure the *precision* of the annotations relative to a given event);

[4]We collected the Tweets via the Twitter Streaming API: `https://dev.twitter.com/docs/streaming-api/methods`. The corpus contains approximately 12 million tweets, spanning 1.39 million users.

[5]In this case, we assume that hashtags represent events or topics. This assumption is used only as a means to simplify the process for synthesising dataset. Note that these hashtags (that occur over 80% of the days in the period) are generally popular topics that are frequently discussed and may not necessarily be news events.

- *to guard against the possibility that the background tweets relate to injected event* — In the original dataset, it is highly likely that there are tweets which mention the original news event but aren't tagged with one of the pre-identified hashtags. It is unlikely, however, that there would be background tweets which mention an event from 1999 (i.e. we are attempting to ensure the *recall* of the annotations relative to a given event).

In doing a one-for-one replacement of an original tweet with a TDT3 sentence and maintaining the original timestamp of the tweet, we are additionally achieving an event propagation distribution which is as faithful as possible to actual event mentions in Twitter.

In total, we created 5 datasets, each with a single novel event and 25–150 background events. We initially experimented with 50 background events but later varied the number of background events—hence the variation in the number of background events in the synthetic dataset. Each dataset spans over a period of 9 days, with the novel event injected on the 5th day (and subsequent days at a level defined by the frequency of the original hashtags).[6]All documents are stopped and lemmatised.[7] An example of a sample of generated documents is displayed in Table 3.[8] A summary of the novel events, mapped TDT3 topics and other metadata is presented in Table 4.

4.2 Experiments

In all experiments, we set the time slice to 1 day and length of window $|L|$ to 2 days. Note, however, that the time slice setting is flexible: should more temporally fine-grained analysis be required, the time slice can be set to a shorter time frame, e.g. 1 hour.[9] We set the contribution factor c to 0.5, meaning old and new documents have approximately equal weighting in the 2-day window. We set α_0 to 0.001; a low value is preferred to produce a very sparse topic distribution over documents (each document, i.e. tweet or sentence from TDT3, should be assigned to no more than a few topics). For β_0, we use 0.01. We vary the setting of T, as detailed in Section 4.2.1.

On a daily basis, a new batch of documents is added to window L and the model is updated. At the end of every update, we calculate the topic evolution score for every topic (as per Section 3.4), and identify topics that exceed a set threshold as *novel topics*.[10] Each document in the model is associated with a distribution of topics. To determine the *novel documents*—documents that contain novel topics—we select those that have a novel topic (i.e. topics which have a topic evolution score above the threshold) assigned as its highest-probability topic. Note that only documents from the new time slice can be novel documents.

As the true set of documents that contain the injected novel event are known — they are all

[6]By injecting the novel event on the 5th day we mean to select a period such that the first occurrence of the novel event will always fall on the 5th day in the period. As such, the natural distribution for the novel event is preserved.

[7]We use OpenNLP for POS tagging and Morpha for lemmatisation (Minnen et al., 2001).

[8]The "language" of the novel event may seem different from a standard tweet, but we contend that the topic model does not gain any real advantage from this as the model is not attuned to the quality of the language in the documents. Also, the Twitter stream contains news releases from news and government organisations.

[9]The main bottleneck for shorter time slices is simply the number of documents, in that if the number of documents becomes too few, the topic model is unlikely to model the data well. Note that the time taken to process a 24-hour time slice on a somewhat high-end single-processor Linux machine is of the order of 15 minutes.

[10]We additionally introduce the constraint that novel topics must not contain user mention tokens (i.e. words of @XXXX format) in their top-10 topic words. This constraint is created on the observation that topics that contain user mentions in their top-10 topic words are usually not semantically coherent and hence do not constitute novel topics.

Event Name	VAN-MITCH			
News Event	Van Earthquake, Turkey			
Date of Occurrence	23 October 2011			
Mapped TDT3 Topic	30002: Hurricane Mitch			
Number of Background Events	25	50	100	150
Proportion of Novel Documents on Date of Occurrence	0.4989 (/15611)	0.2403 (/32371)	0.1232 (/63209)	0.1100 (/70749)
Event Name	WASHI-MITCH			
News Event	Tropical Storm Washi, Philippines			
Date of Occurrence	17 December 2011			
Mapped TDT3 Topic	30002: Hurricane Mitch			
Number of Background Events	25	50	100	150
Proportion of Novel Documents on Date of Occurrence	0.1469 (/8203)	0.0452 (/26656)	0.0222 (/54200)	0.0223 (/54098)
Event Name	LIÈGE-PINOCHET			
News Event	Liège Murder-suicide Attack, Belgium			
Date of Occurrence	13 December 2011			
Mapped TDT3 Topic	30003: Pinochet Trial			
Number of Background Events	25	50	100	150
Proportion of Novel Documents on Date of Occurrence	0.0857 (/8971)	0.0245 (/31296)	0.0106 (/72859)	0.0099 (/77494)
Event Name	KIM-MILOSEVIC			
News Event	Death of Kim Jong Il, North Korea			
Date of Occurrence	19 December 2011			
Mapped TDT3 Topic	30015: Holbrooke-Milosevic Meeting			
Number of Background Events	25	50	100	150
Proportion of Novel Documents on Date of Occurrence	0.3965 (/14614)	0.1468 (/39433)	0.0763 (/75858)	0.0750 (/77114)
Event Name	COSTA-SWISSAIR			
News Event	Costa Concordia Disaster, Italy			
Date of Occurrence	14 January 2012			
Mapped TDT3 Topic	30016: SwissAir111 Crash			
Number of Background Events	25	50	100	150
Proportion of Novel Documents on Date of Occurrence	0.1340 (/9875)	0.0418 (/31610)	0.0205 (/64404)	0.0182 (/72489)

Table 4: Metadata for the 5 synthetic datasets. Numbers in parentheses indicate the total number of documents on the day the event occurred.

TDT3 sentences — we can assess the effectiveness of the model by calculating the "tweet"-level precision, recall and F-score on the day the novel event occurred.

4.2.1 Detection of Novel Event over Varying Numbers of Topics T

The number of topics, T, is a key parameter in the model that affects the granularity of the topics. A high value of T allows the model to generate more specialised topics, while a low value of T generates higher level, more general concepts. We experiment with a range of T values to ascertain how sensitive the topic model is to the T setting, and attempt to arrive at a recommendation for an appropriate T setting for general-purpose applications.

In our initial experiments, we vary T and keep the number of background events constant at 50. A summary of the F-scores for the classification of novel documents is presented in Table 5. Encouragingly, we see that the topic model is able to detect the novel event with relatively high reliability when T is greater than 25 for all datasets. Bear in mind that these F-scores are at the message level not the topic level, and are predicated on the detection of the novel event via

Number of Topics T	VAN-MITCH	WASHI-MITCH	LIÈGE-PINOCHET	KIM-MILOSEVIC	COSTA-SWISSAIR
25	0.50	0.00	0.00	0.51	0.00
50	0.74	0.62	0.47	0.72	0.37
100	0.63	0.61	0.55	0.62	0.47
150	0.65	0.45	0.59	0.76	0.46

Table 5: F-scores with varying T (the number of background events is kept constant at 50).

Number of Background Events	VAN-MITCH	WASHI-MITCH	LIÈGE-PINOCHET	KIM-MILOSEVIC	COSTA-SWISSAIR
25	0.77	0.55	0.81	0.80	0.62
50	0.74	0.62	0.47	0.72	0.37
100	0.61	0.53	0.00	0.82	0.45
150	0.45	0.34	0.00	0.70	0.46

Table 6: F-scores with varying number of background events (T is kept constant at 50).

topic shifts. Beyond $T = 50$, the F-scores are largely similar, indicating the model's insensitivity to small changes in the T setting.

4.2.2 Detection of Novel Event with Varying Number of Background Events

In Section 4.2.1, we can observe that the F-scores are generally higher for datasets that have a greater proportion of novel documents (VAN-MITCH, KIM-MILOSEVIC). We similarly see the same trend in topic evolution score: VAN-MITCH and KIM-MILOSEVIC have higher $e(t)$ than LIÈGE-PINOCHET and COSTA-SWISSAIR. This implies that the proportion of novel documents in the data stream has an influence on the detection of topic(s) associated with the novel event.

To better understand this, we next keep T constant at 50 and vary the number of background events. Increasing the number of background events decreases the proportion of novel documents; effectively there will be more mixtures of topics in the data stream and thus it will be harder to detect the novel event.[11] F-scores for classifying the novel documents with varying number of background events are presented in Table 6.

We see that the injected novel event in all datasets except LIÈGE-PINOCHET is detected for all settings of the number of background events.[12] The proportion of novel documents in the LIÈGE-PINOCHET dataset is particularly low—0.0245 at 50 background events (Table 4). It is thus not surprising that the novel event is not detected when the number of background events is increased to 100: the proportion of novel documents drops to a mere 0.0106, the lowest out of all the datasets. Overall, the results are positive and the model has demonstrated its ability to detect novel events, even when the proportion of novel documents is as low as 0.0182 (COSTA-SWISSAIR at 150 background events).

For all the F-scores presented, a threshold over the topic evolution score $e(t)$ is required to determine the set of novel topics on each update. The threshold facilitates the classification of documents that are novel, and impacts directly on the F-score. To choose a suitable setting for the topic evolution score threshold, we plot a graph of F-scores of all datasets with varying number of topics T (Section 4.2.1) and background events (Section 4.2.2) against a range of topic evolution score threshold values in Figure 1. Based on Figure 1, we set the topic evolution

[11] The proportion of novel documents with varying number of background events is summarised in Table 4.

[12] We observe some fluctuation in the F-scores; this is quite possible as the background events are selected independently for each background event setting, i.e. the set of background events in BG=25 is not a subset of those in BG=50.

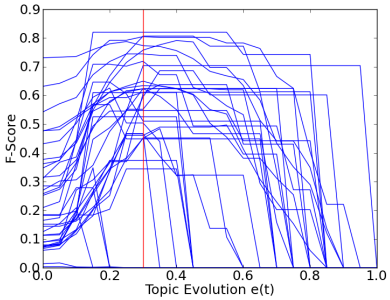

Figure 1: F-scores for the classification of novel documents against threshold values of topic evolution score $e(t)$. Each line represents a single dataset with a fixed number of topics T and background events. The vertical red line indicates the threshold at $e(t) = 0.3$

Injected Novel Events	WASHI-MITCH, LIÈGE-PINOCHET, KIM-MILOSEVIC		
Period	9 October 2012 – 23 October 2012		
Number of Background Events	50		
Novel Event	WASHI-MITCH	LIÈGE-PINOCHET	KIM-MILOSEVIC
Date of Occurrence	17 October 2012	13 October 2012	19 October 2012
Proportion of Novel Documents	0.0452	0.0245	0.1433
on Date of Occurrence	(/26681)	(/31312)	(/40386)

Table 7: Metadata for the synthetic dataset with multiple novel events.

score threshold to 0.3 based on the observation that it provides for consistent and generally optimal performance across all datasets. Note that the F-score results presented thus far are all based on this threshold value.

4.2.3 Detection of Multiple Novel Events

In the previous sets of experiments, each dataset had a single injected novel event. In a real-word setting, however, the number of novel events is unknown and multiple novel events can occur in a single time period. To further test the robustness of our system, and also test our claims about the "fluidity" of the topics under out model, we created another synthetic dataset with a number of novel events injected over a single time period, based on the dates of occurrence of the original events; details of the multi-novel-event dataset are presented in Table 7.

Setting the number of topics T to 50 and using 0.3 for the topic evolution score threshold, we obtain F-scores for novel document classification which are only slightly lower than those for the single-event experiments: 0.49, 0.60, and 0.59 for WASHI-MITCH, LIÈGE-PINOCHET and KIM-MILOSEVIC, respectively, on each event's date of occurrence.

5 Trend Detection in Twitter

The experiments to date have demonstrated the robustness and effectiveness of our methodology in detecting artificially-injected novel events. Ultimately, we are interested in applying the method to online analysis over a microblog such as Twitter to detect emerging trends or events in real-time. To this end, we ran our system for a month in February 2012, over tweets that

Date (UTC)	$e(t)$	Topic ID	Proportion	Topic Words
2012-02-05	0.55	95	0.0093	snow #uks london finally settle look #snow garden nom food
2012-02-06	0.91	256	0.0132	webb howard penalty unite chelsea #mufc game #cfc utd give
2012-02-09	0.77	49	0.0073	capello england fabio resign manager italian job sink ship #capello
2012-02-11	0.62	74	0.0156	suarez evra hand shake racist liverpool #lfc cunt #mufc win
2012-02-12	0.57	160	0.0097	whitney houston rip die dead omg sad amy r.i.p believe
2012-02-12	0.58	168	0.0101	whitney houston sad rip music r.i.p love bong voice remember
2012-02-12	0.61	197	0.0129	whitney houston rip sad r.i.p peace love voice #whitneyhouston song
2012-02-12	0.66	137	0.0134	whitney houston r.i.p rip sad die news #stalbans #harpenden dead
2012-02-13	0.57	91	0.0085	#bafta win film award bafta artist watch meryl #baftas love
2012-02-13	0.71	49	0.0122	zambium penalty ivory coast win #zambia cup zambia miss drogba
2012-02-14	0.46	81	0.0077	happy valentine love &lt xxx ;3&lt dear follow fan load
2012-02-17	0.47	20	0.0077	vagina #replacefilmtitleswithvagina lol war watch funny play movie love wear
2012-02-22	0.50	17	0.0072	win brit #britawards award adele artist british watch international woman
2012-02-22	0.56	251	0.0093	blur adele cut #brits love speech brit sound shit song
2012-02-26	0.60	289	0.0079	goal walcott van super arsenal persie #arsenal wait score game

Table 8: Top-15 trending topics in London in February 2012.

were returned for a geospatially-bound crawl of data from London and New York. We took the two sets of tweets and calculated the topic evolution score $e(t)$ to discover trending topics each day. We set $T = 300$ in each case (to deal with the larger volume of messages as compared to our synthetic experiments). Other parameter values were identical to those used in the synthetic dataset experiments in Section 4.

We display the top-15 February 2012 trending topics in London and New York in Tables 8 and 9, respectively. We filtered out any topics that occurred over less than 0.7% of the stream to show only the popular trends. Note that it is significant that we present both a date and a topic ID: it is possible for the same topic to shift significantly in content at multiple time slices across the topic-modelled time period, although we don't observe this in our limited display of results in Tables 8 and 9.

In London, there is much discussion about soccer (topic IDs 95, 256, 49, 74, 289). A search using the topic words reveal that these topics correspond to real soccer news events. To give a few examples, topic ID 256 is about the controversial penalties awarded by Howard Webb to Chelsea in a Manchester United vs. Chelsea match, topic ID 49 relates to the the resignation of Fabio Capello as England's manager, topic ID 74 is about Suárez refusing to shake Evra's hand before kick off, and topic ID 49 corresponds to Zambia's first victory in the Africa Cup of Nations.

Whitney Houston's death also triggered a massive reaction from Twitter users in London, so much so that it appears in multiple topics (topic IDs 160, 168, 197, 137). Reading over the topics, the difference in these topics seem indistinguishable, and the fact that it occurs across topics is a function of the sheer volume of traffic that mentioned the event; in a deployed setting, it would be relatively trivial to pick up on the fact that the topics are almost identical (Newman et al., 2010; Mimno et al., 2011). Lastly, entertainment award shows are another popular topic in the Twittersphere (topic ID 91, 17, 251), in the form of the BAFTA and Brits awards.

In New York, rather than soccer, American football is the dominant sport and there was a lot of talk of the Super Bowl (topic IDs 267, 50, 88, 60, 207). Similarly to London, Whitney Houston's death drew much attention, split across a number of largely-indistinguishable topics (topic IDs 51, 223, 45), with the exception of topic ID 250, which is related to her funeral. Entertainment award shows are again a popular topic, although New Yorkers are more interested in the Grammy's and Oscars (topic IDs 265, 227, 4, 290, 246).

Date (UTC)	$e(t)$	Topic ID	Proportion	Topic Words
2012-02-06	0.40	267	0.0088	giant 2012 superbowlpocalypse superbowl york fan win move championapocolypse target
2012-02-06	0.46	50	0.0071	#giants #superbowl giant win fuck die touchdown #giantsnation pat root
2012-02-06	0.50	88	0.0081	giant win #superbowl patriot wear shirt jersey fan superbowl today
2012-02-06	0.53	60	0.0101	super bowl giant 2012 champion york fan move target sunday
2012-02-06	0.69	207	0.0089	brady tom elus #superbowl #giants manning giant win catch game
2012-02-12	0.54	51	0.0112	whitney houston rip sad die love #whitneyhouston music r.i.p #rip
2012-02-12	0.58	223	0.0108	whitney houston die rip dead r.i.p sad bobby singer lol
2012-02-12	0.63	45	0.0109	whitney houston rip sad r.i.p dead die omg wow damn
2012-02-13	0.40	265	0.0081	adele win #grammys grammy award tonight watch congrat game artist
2012-02-13	0.42	227	0.0096	chri brown #grammys rihanna bobby love coldplay grammy performance sing
2012-02-13	0.51	4	0.0077	minaj nickus performance nicki #grammys wtf adele lol grammy perform
2012-02-14	0.64	273	0.0125	valentine happy love single &lt today holiday word tomorrow heart
2012-02-19	0.50	250	0.0083	whitney houston rip love #whitneycnn kevin costner funeral r.i.p carter
2012-02-27	0.59	290	0.0075	meryl #oscars streep win violum oscar bradley cooper love lin
2012-02-27	0.76	246	0.0082	#oscars win octavium oscar spencer speech love jlo look dress

Table 9: Top-15 trending topics in New York in February 2012.

In both locations, a new topic for Valentine's Day emerged on February 14th (topic IDs 81 (London) and 273 (New York)).

6 Discussion

The motivation for using a topic model-based approach as opposed to a keyword-based approach is borne out in looking at the detected Twitter trends in Section 5. Some of the detected topics, such as the London football news event examples, would be difficult to capture in a single string of one to three words as used in conventional keyword-based approaches. With our topic model-based detection system, however, the details of an event are summarised more appropriately with a list of associated words.

We note that we did not compare our methodology against existing approaches, such as OLDA or keyword-based approaches. The reasoning for excluding a comparison against OLDA is that in preliminary experiments we found that OLDA was not very effective in detecting the novel topics, as the model uses a fixed vocabulary. Keyword-based approaches are incompatible with our task set up, as they assume knowledge of a fixed information need, which we don't have—our model is looking for the *unknown* in detecting novel events. It is possible to use bursty term analysis (in which case the keywords don't need to be pre-identified), but this often leads to a highly cryptic and potentially misleading representation of the novel event, unlike topic models.

In the synthetic dataset experiments, we measure the model's performance in detecting novel events by calculating the F-scores of novel document classification. The F-score result is a straightforward and objective evaluation, but it is an underestimate of the model's true performance for two reasons. For one, we are actually interested in the detection of the injected event as a newly emerged topic in the data stream rather than correctly classifying every document that contains the injected event. The latter task is the one we evaluate, and a significantly harder task than the former (in fact, if our tweet-level F-score is non-zero, it means we have been successful in detecting the event as a novel topic, meaning we have succeeded in all cases for $T \geq 50$ other than LIÈGE-PINOCHET with the number of background events ≥ 100).

The second reason is that the model could potentially pick up other genuine novel events that occurred in the *background Tweets* that were not related to the injected novel event. Manual inspection on a sample of discovered novel topics revealed that this indeed happened, and the novel topic is often the original news event which was replaced by a TDT3 topic. As an example,

Topic ID	Topic Words
6	kim call korea north jong-il die fire pussy library jong
9	nato kosovo milosevic president force strike crisis military problem unite
13	serb albanian kosovo ethnic kill police hundred rebel home province
19	milosevic nato kosovo holbrooke president richard yugoslav official envoy force
55	kosovo nato force troops milosevic president albanian iraq serb news
72	2011 blue news kosovo maldive top white refugee stand #egypt
73	nato milosevic kosovo military war official international house tax demand
84	kosovo force security monitor mission organization milosevic europe international agreement
86	kim north jong leader die korea korean #fail news christmas

Table 10: Detected novel topics in KIM-MILOSEVIC (50 background events and $T = 100$).

we present a list of detected novel topics for KIM-MILOSEVIC with 50 background events and $T = 100$ topics in Table 10. We see that topic IDs 6 and 86 are related to Kim Jong Il's death, the original news event which was replaced by TDT3's "Holbrooke-Milosevic Meeting" topic. As the documents containing Kim's death do not constitute as part of the *gold* novel document set that have the TDT3 event, the model is penalised for classifying these Tweets as novel documents.

Admittedly, scalability of the model has not been the focus in this work. One reason is that we were initially interested in investigating the accuracy performance of topic models in detecting emerging events, leaving optimisation for future work. The second reason is that for the purposes of tracking *popular* emerging trends, we do not necessarily have to process the full collection of Tweets, as these trends occur in a significant portion of the data. Given that we have set our sights on Twitter, however, this is an obvious area to focus future attentions, given that an estimated 250 million tweets were posted per day on Twitter in 2011. If we were to apply the method to the full Twitter feed, we would use a much finer-grained time granularity and run our method over a larger number of cores than at present (our implementation is already parallelised), and are confident of being able to keep pace with this much greater data volume. Our implementation can be accessed from `http://www.csse.unimelb.edu.au/~tim/etc/online_lda.zip`.

In terms of the sensitivity of the model when scaling down to smaller numbers of documents, in our Twitter trend detection experiments conducted for London and New York, we have demonstrated that even over relatively small numbers of documents —each location has less than 60,000 tweets per day — interesting popular trends and fine-grained news events can be detected.

7 Conclusion

We have proposed a novel topic model-based approach to on-line trend analysis. On every update, we calculate the evolution of topics to detect newly emerged topics in the document collection. We first applied the methodology to a suite of synthetic datasets and demonstrated the model's strength in detecting individual documents describing novel events, and moved on to process raw Twitter data to detect trending topics. The discovered trends were promising and gave insights to the popular culture and events discussed in the Twittersphere.

Acknowledgements

NICTA is funded by the Australian government as represented by Department of Broadband, Communication and Digital Economy, and the Australian Research Council through the ICT centre of Excellence programme. JHL's travel was supported by a Google PhD Travel Prize.

References

Allan, J. (2002). Introduction to topic detection and tracking. In Allan, J., editor, *Topic Detection and Tracking: Event-based Information Organization*, pages 1–16. Kluwer.

AlSumait, L., Barbará, D., and Domeniconi, C. (2008). On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM-08)*, pages 3–12, Washington, DC, USA.

Blei, D., Ng, A., and Jordan, M. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Culotta, A. (2010). Towards detecting influenza epidemics by analyzing Twitter messages. In *Proceedings of the First Workshop on Social Media Analytics*, pages 115–122, New York, NY, USA.

De Longueville, B., Smith, R., and Luraschi, G. (2009). "omg, from here, i can see the flames!": A use case of mining location based social networks to acquire spatio-temporal data on forest fires. In *Proceedings of the 2009 International Workshop on Location Based Social Networks*, pages 73–80, Seattle, Washington.

Diao, Q., Jiang, J., Zhu, F., and Lim, E. (2012). Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL HLT 2012)*, pages 536–544, Jeju Island, Korea.

Griffiths, T. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235.

Hoffman, M., Blei, D. M., and Bach, F. (2010). Online learning for latent dirichlet allocation. In *Advances in Neural Information Processing Systems 23 (NIPS-10)*, pages 856–864.

Kireyev, K., Palen, L., and Anderson, K. (2009). Applications of topics models to analysis of disaster-related Twitter data. In *NIPS Workshop on Applications for Topic Models: Text and Beyond*, Whistler, Canada.

Lampos, V. and Cristianini, N. (2010). Tracking the flu pandemic by monitoring the social web. In *2nd IAPR Workshop on Cognitive Information Processing (CIP 2010)*, pages 411–416, Montreal, Quebec.

Mathioudakis, M. and Koudas, N. (2010). TwitterMonitor: Trend detection over the Twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158, New York, NY, USA.

Mimno, D., Wallach, H., Talley, E., Leenders, M., and McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 262–272, Edinburgh, UK.

Minnen, G., Carroll, J., and Pearce, D. (2001). Applied morphological processing of English. *Journal of Natural Language Processing*, 7(3):207–223.

Newman, D., Lau, J., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, pages 100–108, Los Angeles, USA.

Osborne, M., Petrović, S., McCreadie, R., Macdonald, C., and Ounis, I. (2012). Bieber no more: First story detection using twitter and wikipedia. In *SIGIR 2012 Workshop on Time-aware Information Access*, Oregon, USA.

Petrović, S., Osborne, M., and Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, pages 181–189, Los Angeles, USA.

Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web (WWW-2010)*, pages 851–860, New York, NY, USA.

Signorini, A., Segre, A., and Polgreen, P. (2011). The use of twitter to track levels of disease activity and public concern in the u.s. during the influenza a h1n1 pandemic. *PLoS ONE*, 6:e19467.

Weng, J. and Lee, B. (2011). Event detection in Twitter. In *Proceedings of the Fifth International Conference on Weblogs and Social Media (ICWSM-2011)*, Barcelona, Catalonia, Spain.

Zanzotto, F., Pennacchiotti, M., and Tsioutsiouliklis, K. (2011). Linguistic redundancy in twitter. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 659–669, Edinburgh, United Kingdom.

Zhao, X., Shu, B., Jiang, J., Song, Y., Yan, H., and Li, X. (2012). Identifying event-related bursts via social media activities. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP 2012)*, pages 1466–1477, Jeju Island, Korea.