Incremental and Adaptive Topic Detection over Social Media

Konstantinos Giannakopoulos(✉) and Lei Chen(✉)

Department of Computer Science and Engineering,
Hong Kong University of Science and Technology,
Clear Water Bay, Kowloon, Hong Kong
{kgaa,leichen}@cse.ust.hk

Abstract. Social media like Twitter and Facebook are very popular nowadays for sharing users' interests. However, the existing solutions on topic detection over social media overlook time and location factors, which are quite important and useful. Moreover, social media are frequently updated. Thus, the proposed detection model should handle the dynamic updates. In this paper, we introduce a topic model for topic detection that combines time and location. Our model is equipped with incremental estimation of the parameters of the topic model and adaptive window length according to the correlation of consecutive windows and their density. We have conducted extensive experiments to verify the effectiveness and efficiency of our proposed Incremental Adaptive Time Location (IncrAdapTL) model.

1 Introduction

In recent years, the use of online social networks like Twitter have been spread. Hundreds of thousands of short messages are exchanged between users. Research has been done on detection of topics on messages that users publish. Each tweet, consists of the main text message, and additional useful information like timestamp and location coordinates. All this information is used by researchers in order to incorporate time and location in the proposed topic detection models.

In this paper, we propose a generative, LDA-based topic model for topic detection in tweets. Our model incorporates time-zones and location regions. We process input data with sliding windows with incremental re-evaluation of the topic model parameters and adaptive window lengths for faster processing.

Most of the previous existing works on this field that propose generative topic models use either only location or only time separately. However, we combine both. In addition, previous works do not handle incremental updates of model's parameters. We propose incremental updates where we do not need to process all the tweets in the sliding time windows. It is not necessary to process the same tweets in consecutive windows. Moreover, we propose adaptive window lengths. There are time periods where more tweets are posted and time periods where less tweets appear. We take advantage of sparse windows by increasing the window

© Springer International Publishing AG, part of Springer Nature 2018
J. Pei et al. (Eds.): DASFAA 2018, LNCS 10827, pp. 460–473, 2018.
https://doi.org/10.1007/978-3-319-91452-7_30

length. This improves accuracy of detected topics in sparse windows. As far as we are concerned, previous research works do not use any mechanism to handle this situation.

The main contributions of our proposed model are the following:

- Firstly, we introduce incremental update of the model parameters between consecutive windows. Our proposed model used sliding windows for processing messages. It does not need to process the old messages of each window. It processes only the new tweets and we do not need to re-evaluate from scratch the model parameters in each window.
- Secondly, we introduce adaptive window lengths for processing data. We observe that more tweets and different topics are posted during day-time than during night-time. So, we adapt the window length according to the correlation of consecutive windows and according to their density for faster processing.

The rest of the paper is organized as follows. In Sect. 2 we review the related work, in Sect. 3 we present our topic model, in Sect. 4 we evaluate our approach.

2 Related Work

In this section we review some previous research papers that have proposed topic models for topic detection. Topic models are based on the original LDA that is introduced in [2].

Firstly, we present *temporal topic models* that were proposed in previous works. A nonparametric mixture model for topic modeling over time is introduced in [5]. TOT [11] is a non-Markov continuous-time model of topical trends. In this model, words and continuous time are generated by a topic associated with a user. Dynamic Topic Model (DTM) [1] captures the evolution of topics over time. It shows topic distribution in various time intervals.

Secondly, we discuss *spatial topic models* that were introduced in previous works. Geographical topic discovery and comparison in presented in [13]. It presents three models: a location-driven model where GPS documents are clustered into topics based on their locations, a text-driven model where geographical topics are detected based on topic modeling with regularization by spatial information, a location-text joint model, a.k.a. LGTA (Latent Geographical Topic Analysis), which combines geographical clustering and topic modeling into one framework. GLDA (Geo Latent Dirichlet Allocation) [9] extends LDA for location recommendation. Paper [7] addresses the problem of modeling geo-graphical topical patterns on Twitter by introducing a sparse generative model.

Thirdly, we show few research works that combine time and location in *Spatio-Temporal topic models*. Paper [10] processes users' check-in. It detects topics and proposed a POI recommendation system with spatial and temporal information of user movements and interests. It proposes two models: USTTM and MSTTM for local (within a city) and global area (between cities) respectively. A Spatio-Temporal Topic (STT) model for location recommendation is

presented in [8]. It processes users' check-ins to combine geographical influence and temporal activity patterns.

In addition, topic detection has been achieved through *wavelet analysis*. A lightweight event detection using wavelet signal analysis of hashtag occurrences in the twitter public stream is presented in [4].

Moreover, LDA-based methods for topic detection are SparseLDA [12] and O-LDA [3]. These methods describe real-time approaches to detect latent topics in data streams. In addition, topic mixtures estimated from an LDA model [6] are used to identify hot and cold topics.

3 Approach

We propose an LDA-based generative model for topic detection that incorporates time and location, that we call 'IncrAdapTL'. We identify two time-zones according to tweet time-stamps: day-time [6am–6pm] and night-time [6pm–6am]. The collected locations are the districts from the city of Hong Kong.

Our proposed model processes input data with sliding windows. We introduce incremental update of model's parameters between consecutive windows, and adaptive window lengths. We call our model Incremental Adaptive Time Location (IncrAdapTL) model and we present it in Algorithm 2.

3.1 Generative Process

In Table 1 we list the notations of parameters that we use. In Fig. 1 we present the topic model of IncrAdapTL and in Algorithm 1 its generative process. Our model consists of four distributions: word multinomial distribution per topic ϕ, topic multinomial distribution per tweet θ, timezone multinomial distribution per tweet ω, and location multinomial distribution per tweet ψ.

For each word w of each tweet message m, first we draw a timezone t from a multinomial distribution ω of timezones per tweet message, then we draw a location l from a multinomial distribution ψ of locations per tweet message, and finally we draw a topic z using the sampling process described in Sect. 3.2.

Algorithm 1. Generative Process

```
for each tweet m do
    for each word w of the tweet m do
        Draw a timezone t ~ Mult(ω);
        Draw a location l ~ Mult(ψ);
        Draw a topic z ~ p(k|t, l);
    end
end
```

Table 1. Notation of parameters

Variable	Notation
ϕ	Word distribution per topic
θ	Topic distribution per tweet message
ω	Timezone distribution per tweet message
ψ	Location distribution per tweet message
t	A chosen timezone
l	A chosen location
z	A chosen topic
m	A tweet message
M	Total number of tweets
N	Total number of words for each tweet
V	Vocabulary size
K	Total number of topics
T	Total number of timezones
L	Total number of locations
$n_{w,k}$	Occurrences of a word w given a topic k
$\sum_{w=1}^{V} n_{w,k}$	Total number of words assigned to topic k
$n_{m,k}$	Occurrences of a topic k given a tweet m
$\sum_{k=1}^{K} n_{m,k}$	Total number of topics assigned to tweet m
$n_{t,m}$	Occurrences of a timezone t given a tweet m
$\sum_{t=1}^{T} n_{t,m}$	Total number of timezones assigned to a tweet m
$n_{l,m}$	Occurrences of a location l given a tweet m
$\sum_{l=1}^{L} n_{l,m}$	Total number of locations assigned to tweet m

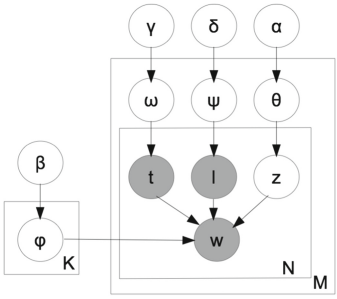


Fig. 1. Topic model

3.2 Sampling

Each drawn topic depends on the sampled timezone and on the sampled location by estimating the following probability:

$$p(k|t,l) \sim \frac{n_{w,k}+\beta}{\sum_{w=1}^{V} n_{w,k}+V\beta} \times \frac{n_{m,k}+\alpha}{\sum_{k=1}^{K} n_{m,k}+K\alpha} \times \frac{n_{t,m}+\gamma}{\sum_{t=1}^{T} n_{t,m}+T\gamma} \times \frac{n_{l,m}+\delta}{\sum_{l=1}^{L} n_{l,m}+L\delta} \quad (1)$$

3.3 Incremental

The IncrAdapTL model uses incremental re-estimation of topic model parameters. In the following algorithms we notate 'increm' mode when we estimate parameters from the previous window incrementally without processing all the tweets of each window. We notate 'estim' mode when we estimate parameters non-incrementally. In the latter, we re-estimate the parameters from scratch, by processing all the tweets of every window.

At this part, we explain the IncrAdapTL Algorithm 2. Our algorithm processes a stream of tweet data using sliding windows. In the first window of the stream of data [lines: 3–7] we run our model in the 'estim' mode (initialization and sampling). There are no previous parameters saved on the model, we run in non-incremental model because we need to process all the tweets of the first window.

In the rest windows of the stream [lines: 8–15]. Firstly, we load intermediate results from the previous window for incremental update of the model parameters. Secondly, we make decision of adaptive window length (we describe this in detail in Algorithm 5 of Sect. 3.4). Thirdly, we run our model in the 'increm' mode (initialization and sampling). Finally, in both modes, we save the window intermediate results [lines: 16–17].

Both modes, 'estim' and 'increm', have two steps: initialization and sampling. The sampling step is the same in both modes. During sampling, for each word of every tweet document a timezone, a location, and a topic are assigned. The initialization step of each mode differs.

Initialization of the 'estim' mode is presented in Algorithm 3. First we initialize the counters that are used in the estimation of the probabilities: $n_{w,k}, n_{m,k}, \sum_{w=1}^{V} n_{w,k}, \sum_{k=1}^{K} n_{m,k}$, with 0. We pass through all the tweets of the current window (old and new). We cannot benefit from the tweets that already existed in the previous window. Then, for every word of each tweet message we randomly choose a topic k and we increment the proper counters above by 1.

Initialization of the 'increm' mode is presented in Algorithm 4. Our model processes tweet datasets with sliding windows. In order to avoid passing through the tweets that existing in the previous window, we keep an tweet index in the stream of tweets, i.e. tweetIndex, from the previous window [lines: 3, 4]. So, we load the counters, $n_{m,k}$ and $\sum_{k=1}^{K} n_{m,k}$, that are related with the topic-tweet

Algorithm 2. Window Process of Incremental Adaptive Time Location (IncrAdapTL) model

```
Global Initialization - Collection of global dictionary, locations, timezones;
for each window do
    if windowCounter == 1 then
        /* Run in 'estim' mode. */
        Initialization of 'estim' mode;
        Sampling;
    end
    else
        Load intermediate results from previous window;
        /* Adaptive window length. */
        Adaptive window length decision;
        /* Incremental parameter estimation. Run in 'increm' mode. */
        Initialization of 'increm' mode;
        Sampling;
    end
    /* For incremental update. */
    Save window intermediate results;
end
```

Algorithm 3. Initialization of 'estim' mode

```
/* Initialize counters with zero. */
n_{w,k} = 0, sum_{w=1}^{V} n_{w,k} = 0, n_{m,k} = 0, sum_{k=1}^{K} n_{m,k} = 0;
for each tweet m do
    for each word w do
        /* Draw a topic k randomly. */
        k = Random(K);
        /* Increment proper counters by one. */
        n_{w,k} += 1; sum_{w=1}^{V} n_{w,k} += 1; n_{m,k} += 1; sum_{k=1}^{K} n_{m,k} += 1;
    end
end
```

distribution θ. In addition, when we have slided the window we have updated the counters, $n_{w,k}$ and $\sum_{w=1}^{V} n_{w,k}$ that are related with the word-topic distribution ϕ. So, in [line: 2] of Algorithm 4, we load the updated values of $n_{m,k}$ $\sum_{k=1}^{K} n_{m,k}, n_{w,k}$ and $\sum_{w=1}^{V} n_{w,k}$. These counters contain the information of the overlap between consecutive windows.

Algorithm 4. Initialization of 'increm' mode

```
1  /* Load counters from previous window. */
2  Load previous n_{m,k}, Σ_{k=1}^{K} n_{m,k}, n_{w,k}, Σ_{w=1}^{V} n_{w,k};
3  /* Load the index in tweet stream. */
4  Load tweetIndex;
5  for each tweet m after tweetIndex do
6  |   for each word w do
7  |   |   /* Draw a topic k randomly. */
8  |   |   k = Random(K);
9  |   |   /* Increment proper counters by one. */
10 |   |   n_{w,k} += 1; Σ_{w=1}^{V} n_{w,k} += 1; n_{m,k} += 1; Σ_{k=1}^{K} n_{m,k} += 1;
11 |   end
12 end
```

Then, in [lines: 5–12] we process only the new tweets of the current window. For each word of every tweet after the tweetIndex, we update $n_{w,k}, n_{m,k}, \sum_{w=1}^{V} n_{w,k}, \sum_{k=1}^{K} n_{m,k}$ as before.

After initialization we perform sampling as we have mentioned above in Algorithm 2. We have described the sampling method in Sect. 3.2. The sampling process remains the same in both 'estim' and 'increm' modes.

3.4 Adaptive Window

Our second contribution is that the IncrAdapTL model uses adaptive window lengths. We have observed that the number of posted tweets varies between night-time and day-time in particular districts and in total. Throughout a day, there are sparse and dense windows. The tweet density of windows affects the performance of a topic model. Thus, in sparse windows we increase the window length in order to process more tweets. On the other hand, in dense windows we decrease the window lengths, so that we can focus on smaller time period.

So, we introduce different window lengths for more efficient processing of input stream in terms of time and accuracy. We start with window of 2 h length and we double it until it reaches the length of 8 h. Hence, we have three window lengths: windows of 2 h, 4 h, 8 h. In each case, the overlap with the previous window has length of 1 h.

As we have shown above in Algorithm 2, during processing of each window our model decides adaptively the length of the next window $i+1$ [lines: 10–11]. This decision is made as follows: First, we sample r% of the current window i. $r = \frac{\#\text{tweets in window}}{\text{window length in hours}} * 0.001$. We observe that the number of tweets per hour, i.e. $\frac{\#\text{tweets in window}}{\text{window length in hours}}$, ranges between 100 and 300. So, we transform this number into a percentage between 10% and 30%. We use high sampling

ratio for dense windows and low sampling ratio for sparse windows. This is how our sampling rate is estimated in every window.

After we have collected the samples of the current window i, we compare the topic distribution of the samples $sample_i$ with the topic distribution of the previous window $increm_{i-1}$ by estimating the $\chi^2 - test$. We present this in Algorithm 5.

Algorithm 5. Adaptive window length decision

```
1 /* sample r% sample_i mode */
2 Run the topic model in 'sample' mode;
3 /* sample_i ~ increm_{i-1} */
4 χ² − test for topic-tweets distributions comparison of sample_i and increm_{i-1};
```

In Algorithm 6 we explain the steps for applying the $\chi^2 - test$. First [line: 1], we map similar topics between the 'sample' mode in current window i and the 'increm' mode of the previous window, $i-1$. We use the Jaccard distance for this topic similarity. We detect 15 topics in every mode and every topic consists of 10 words. Then, in [lines: 2, 3], we collect the tweet-topic distributions in $sample_i$ and in $increm_{i-1}$.

Algorithm 6. $\chi^2 - test$ for topic-tweets-distribution in each window i

```
1  Map similar topics between sample_i and increm_{i-1};
2  Collect tweets-per-topic distribution in sample_i;
3  Collect tweets-per-topic distribution in increm_{i-1};
4  Estimate the χ² − test between sample_i and increm_{i-1};
5  if χ² > critical value then
6  |  /* Reject H_0 */
7  |  if current window i is more dense than window i − 1 then
8  |  |  /* more dense, smaller window */
9  |  |  make next window (i + 1) length half;
10 |  end
11 |  else
12 |  |  /* more sparse, larger window */
13 |  |  make next window (i + 1) length double;
14 |  end
15 end
16 else
17 |  /* Insufficient evidence to reject H_0 */
18 |  keep same window length;
19 end
```

Then, in [line: 4], we use the $\chi^2 - test$ in order to test if tweet-topic distributions of $sample_i$ and $increm_{i-1}$ are similar. We consider null hypothesis H_0 that they come from same distribution, with significance level: $\alpha = 0.05$.

H_0: tweet-topic distributions of $sample_i$ and $increm_{i-1}$ are similar.
H_1: not H_0.

Then, in [lines: 5–15], if the χ^2 is larger than the critical value, then we reject the H_0. In this case, the distributions are not similar and we change the length of the window. If the current window i is more dense than the previous window $(i-1)$, then we make next window $(i+1)$ length half [lines: 7–10]. Otherwise, if the current window i is more sparse, then we make next window $(i+1)$ length double [lines: 11–14]. The window length ranges between 2 and 8 h. The overlap between consecutive windows is fixed to 1 h. Density metric is the comparison of tweets per hour $\frac{\#\text{tweets in window}}{\text{window length in hours}}$ between current and previous window.

When we have insufficient evidence to reject H_0 [lines: 16–19], we consider that the distributions are similar and we keep the same window length for next window.

4 Evaluation

In this section we present the experiments for the evaluation of our Incremental Adaptive Time Location (IncrAdapTL) model. We perform two sets of experiments. In the first set we compare the running times between IncrAdapTL and its non-incremental and non-adaptive version (TL). We show that IncrAdapTL processes the same dataset faster. In the second set of experiments, we show how the accuracy of IncrAdapTL changes in relationship with window lengths.

4.1 Characteristics of Datasets

Firstly, we present details on the datasets we use. We have crawled tweets from Hong Kong. We identify 22 districts, and two time-zones: day-time [6am–6pm], night-time [6pm–6am]. We use three datasets. As we present in Table 2, dataset A consists of 73K tweets crawled from the 21st December, 2015 to the 3rd January, 2016; dataset B includes 47K tweets from the 15th January to the 25th January; dataset C contains 77K from the 28th January to the 14th February.

We crawl tweets from the internet Twitter4j API[1] and Snowball[2]. We collect data from the area of Hong Kong. The goal of our work is the detection of discussed topics in different districts of the city, in different time-zones. We separate a day period into two time-zones: day-time [6am–6pm], night-time [6pm–6am].

[1] http://twitter4j.org.
[2] http://snowball.tartarus.org.

Table 2. Datasets

Dataset	Dates	Number of tweets
A	21/12/2015–03/01/2016	73,192
B	15/01/2016–25/01/2016	47,585
C	28/01/2016–14/02/2016	77,974

4.2 Execution Time

In the first set of experiments, we compare the execution times in milliseconds of our Incremental Adaptive Time Location (IncrAdapTL) model, as we presented in Algorithms 2 and 6 with the non-incremental and non-adaptive version of our model (TL). In the TL model, every window has a fixed length of two hours (non-adaptive) and in every window we run the 'estim' mode, i.e. estimation of the model parameters from scratch by processing all the tweets of each window, (non-incremental), as we described in Sect. 3.3.

We show that our proposed model, IncrAdapTL, can process the same datasets in less total execution time. We present the results for each dataset in Fig. 2 and in Table 3.

We observe that dataset A is processed by IncrAdapTL in 987 s, and in 1,214 s by TL. IncrAdapTL needs 81% of the TL's time. Similarly, IncrAdapTL processes dataset B in 629 s, and TL in 782 s. The difference is 80%. Also,

Fig. 2. Execution times in (ms)

Table 3. Execution times in (ms)

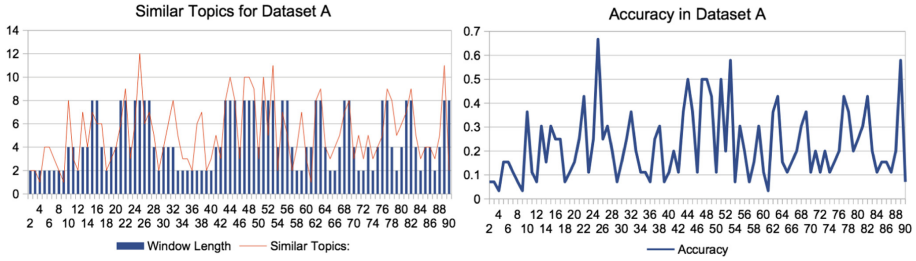
	'IncrAdapTL'	'TL'
Dataset A	987,219	1,214,082
Dataset B	629,736	782,205
Dataset C	1,161,124	1,496,003

IncrAdapTL processes dataset C in 1,161 s, whereas TL in 1,496 s. This is the 77% of TL's time. The experiments show that IncrAdapTL is better. The trend also shows that our method can scale well to very large data sets.

4.3 Accuracy

In the second set of experiments, we estimate the accuracy of our model. In every window, we compare our Incremental Adaptive Time Location (IncrAdapTL) model, as we presented in Algorithms 2 and 6, with the 'estim' mode, i.e. estimation of the model parameters from scratch (non-incremental). The result of the 'estim' mode is our ground truth, because in this mode processes all the tweets of every window and estimate the parameters from scratch. In these experiments, each window length of 'estim' mode (non-incremental) and 'increm' mode (incremental) are the same.

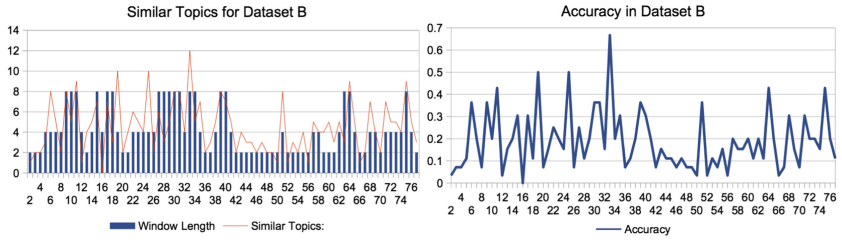
Results for dataset A are presented in Fig. 3; for dataset B in Fig. 4; and for dataset C in Fig. 5. In every graph we observe how our model's window length changes during the processing of the stream of data (adaptive). We see the sparse windows with 8 h length and the dense windows with 2 h length. Also, we see



a. Similar Topics in Dataset A

b. Accuracy in Dataset A

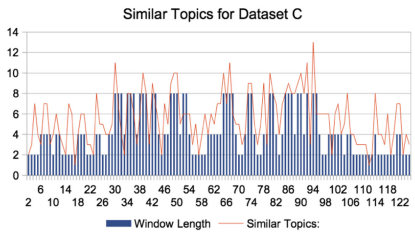
Fig. 3. Dataset A



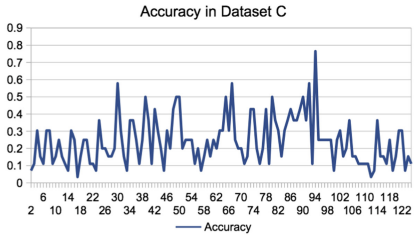
a. Similar Topics in Dataset B

b. Accuracy in Dataset B

Fig. 4. Dataset B



a. Similar Topics in Dataset C



b. Accuracy in Dataset C

Fig. 5. Dataset C

consecutive windows with the same length when the χ^2 value is small, and there is high correlation with the previous window, as we described in Algorithm 6.

Also, we observe that the number of similar topics is improved in the case of sparse windows, then the window length grows to 8 h. The number of common topics follows the length of windows in all datasets.

4.4 Qualitative Analysis

In addition, we show some concrete examples in each dataset that our model can detect some interesting topics. Dataset A presented in Table 4 contains tweet related with travel, Christmas and New Year. Dataset B in Table 5 includes few fashion events and entertainment trends. Dataset C in Table 6 contains topics related with travel, Chinese New Year, entertainment.

Table 4. Similar topics in dataset A

Mode	Topic	Keywords
estim	Christmas	christmas, eve, dinner, everyone, restaurant, kowloon, #HongKong, first
increm	Christmas	christmas, eve, #christmas, restaurant, hongkong, going, #HongKong, city
estim	Travel	#HongKong, #Asia, #Travel, #2015, #Exploration, #Christmas, disney, #Adventure
increm	Travel	#Asia, #Travel, #2015, #Exploration, #Adventure, #Holiday, super
estim	New Year	hong, kong, #2016, #happynewyear, love, #HongKong, #HappyNewYear, posted
increm	New Year	#2016, #happynewyear, #HongKong, #HappyNewYear, #hongkong, #HK, park, #newyear, #hk, peak
estim	New Year	good, first, countdown, morning, #2016HK, fireworks#, hi, hope, people, guys
increm	New Year	love, countdown, posted, #2016HK, fireworks#, hi, bye, life
estim	New Year	year, new, happy, first, #2016, best, start, everyone
increm	New Year	year, new, happy, hk, 2016, photo, posted, see, day, first

Table 5. Similar topics in dataset B

Mode	Topic	Keywords
estim	Fashion	out, new, life, fox, fur, #furry, doing, design, vest
increm	Fashion	central, new, fox, fur, #furry, year, design, vest, #furs, #furvest
estim	Ocean Park	posted, photo, kong, park, hong, ocean, adventure, kok, hotel,
increm	Ocean Park	posted, photo, park, ocean, adventure, away, sure, please, #travel,
estim	Entertainment	devonseron, day, bemylady, #BMLAngSimula, devon, central, seron, china, onitsshowtime
increm	Entertainment	devonseron, bemylady, #BMLAngSimula, devon, seron, onitsshowtime, itsshowtime, tweet, guangzhou, ever
estim	Fashion	sha, tsim, tsui, people, collection, fashionally, #6, womenswear, #fashion, #fashionally
increm	Fashion	time, collection, fashionally, #6, womenswear, #fashion, two, #fashionally, class, side
estim	Sports	#NBAVote, kobe, bryant
increm	Sports	#NBAVote, kobe, bryant, big
estim	Career	#Hiring, #CareerArc, our, #Jobs, #job, see, #HongKong, team, #Zhuhai, latest
increm	Career	#Hiring, #CareerArc, our, #Jobs, #job, #HongKong, team, #Zhuhai, latest, opening

Table 6. Similar topics in dataset C

Mode	Topic	Keywords
estim	Location	hong, kong, airport, international, hkg, islands, district, station, disneyland,
increm	Location	hong, kong, airport, international, hkg, islands, district, disneyland, ocean,
estim	Entertainment	#MrAndMrsSotto, ako, best, wishes, congrats, ang, first
increm	Entertainment	#MrAndMrsSotto, ako, wishes, bossing, congrats, forever, #Shenzhen
estim	Entertainment	#HBLPSL, #AbKhelKeDikha, #PSLT20, runs, overs, wright, new, russell, balls, batsman
increm	Entertainment	#HBLPSL, #AbKhelKeDikha, #PSLT20, runs, overs, gone, bowling, new, batsman, imran
estim	Travel	#travelling, #travelgram, #wanderlust, #travel, #wanderer, #explore, furniture, world
increm	Travel	#travelling, #travelgram, #wanderlust, #travel, #wanderer, #explore, last, miss
estim	Chinese New Year	new, year, happy, chinese, eve, lunar, 2016, year's, coming
increm	Chinese New Year	year, happy, one, lunar, home, spring, market, first
estim	Chinese New Year	new, year, happy, chinese, monkey, eve, batsman, night, lunar, everyone
increm	Chinese New Year	new, year, happy, chinese, monkey, eve, everyone, family, year's, friends
estim	Chinese New Year	year, new, happy, chinese, monkey, lunar, eve, wish, central, #gathering
increm	Chinese New Year	year, new, chinese, lunar, eve, time, wish, hotel, #familydinner, #qualitytime

5 Conclusion

In this paper we propose an Incremental Adaptive Time Location (IncrAdapTL) topic model for topic detection in tweets. This is an LDA-style generative topic model that incorporates time-zones (taken from time-stamps) and locations extracted from tweet stream API. We propose an incremental way of updating the parameters between consecutive windows and an adaptive window length in

relationship with the correlation of consecutive windows and density, for faster processing. We evaluate IncrAdapTL by comparing total execution time and accuracy using three tweet datasets.

Acknowledgment. The work is partially supported by the Hong Kong RGC GRF Project 16207617, National Grand Fundamental Research 973 Program of China under Grant 2014CB340303, the National Science Foundation of China (NSFC) under Grant No. 61729201, Science and Technology Planning Project of Guangdong Province, China, No. 2015B010110006, Webank Collaboration Research Project, and Microsoft Research Asia Collaborative Research Grant.

References

1. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 113–120. ACM (2006)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**(Jan), 993–1022 (2003)
3. Canini, K., Shi, L., Griffiths, T.: Online inference of topics with latent Dirichlet allocation. In: Artificial Intelligence and Statistics, pp. 65–72 (2009)
4. Cordeiro, M.: Twitter event detection: combining wavelet analysis and topic inference summarization. In: Doctoral Symposium on Informatics Engineering, pp. 11–16 (2012)
5. Dubey, A., Hefny, A., Williamson, S., Xing, E.P.: A nonparametric mixture model for topic modeling over time. In: Proceedings of the 2013 SIAM International Conference on Data Mining, pp. 530–538. SIAM (2013)
6. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Natl. Acad. Sci. **101**(suppl 1), 5228–5235 (2004)
7. Hong, L., Ahmed, A., Gurumurthy, S., Smola, A.J., Tsioutsiouliklis, K.: Discovering geographical topics in the twitter stream. In: Proceedings of the 21st International Conference on World Wide Web, pp. 769–778. ACM (2012)
8. Hu, B., Jamali, M., Ester, M.: Spatio-temporal topic modeling in mobile social media for location recommendation. In: 2013 IEEE 13th International Conference on Data Mining (ICDM), pp. 1073–1078. IEEE (2013)
9. Kurashima, T., Iwata, T., Hoshide, T., Takaya, N., Fujimura, K.: Geo topic model: joint modeling of user's activity area and interests for location recommendation. In: Proceedings of the sixth ACM International Conference on Web Search and Data Mining, pp. 375–384. ACM (2013)
10. Liu, Y., Ester, M., Qian, Y., Hu, B., Cheung, D.W.: Microscopic and macroscopic spatio-temporal topic models for check-in data. IEEE Trans. Knowl. Data Eng. **29**, 1957–1970 (2017)
11. Wang, X., McCallum, A.: Topics over time: a non-Markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 424–433. ACM (2006)
12. Yao, L., Mimno, D., McCallum, A.: Efficient methods for topic model inference on streaming document collections. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 937–946. ACM (2009)
13. Yin, Z., Cao, L., Han, J., Zhai, C., Huang, T.: Geographical topic discovery and comparison. In: Proceedings of the 20th International Conference on World Wide Web, WWW 2011, pp. 247–256. ACM, New York (2011)