Torpedo: Topic Periodicity Discovery from Text Data

Jingjing Wang[a] Hongbo Deng[b] and Jiawei Han[a]

[a]University of Illinois, 201 N Goodwin Ave, Urbana IL, USA;
[b]Yahoo! Labs, 701 First Avenue, Sunnyvale CA, USA

ABSTRACT

Although history may not repeat itself, many human activities are inherently periodic, recurring daily, weekly, monthly, yearly or following some other periods. Such recurring activities may not repeat the same set of keywords, but they do share similar topics. Thus it is interesting to mine topic periodicity from text data instead of just looking at the temporal behavior of a single keyword/phrase. Some previous preliminary studies in this direction prespecify a periodic temporal template for each topic. In this paper, we remove this restriction and propose a simple yet effective framework `Torpedo` to mine periodic/recurrent patterns from text, such as news articles, search query logs, research papers, and web blogs. We first transform text data into topic-specific time series by a time dependent topic modeling module, where each of the time series characterizes the temporal behavior of a topic. Then we use time series techniques to detect periodicity. Hence we both obtain a clear view of how topics distribute over time and enable the automatic discovery of periods that are inherent in each topic. Theoretical and experimental analyses demonstrate the advantage of `Torpedo` over existing work.

Keywords: Topic periodicity, text data, time dependent topic modeling

1. INTRODUCTION

Identifying periodic patterns in time series has been extensively studied. However, periodic *topic* discovery in *text data* is less explored.

Text data can reflect many periodic patterns, such as festivals, fashion trends, disease outbreaks and popular research topics, which can be of great interest to sociologists, advertisement providers, scientific researchers, etc. For example, Fig. 1 shows a snapshot of the trend of `wedding+gift` in Google search*. It demonstrates a very neat yearly periodic pattern of queries which contain the keywords `wedding` and `gift`. This may indicate that wedding ceremonies often take place in the summer. With this information, advertisement providers can incorporate a periodic pattern based feature to their recommender system. This feature will promote products which are well suited for wedding gifts in the summer time and thus help maximize the revenue. With the ever-increasing speed of web developments, electronic text together with temporal information such as news, online forums and digital libraries opens up the opportunity to discover periodic patterns from text data.

Keyword-based or query-based periodicity discovery in text data has been discussed in.[1–3] However, instead of periodic *topic* discovery, they check the time series of each individual keyword/query to detect periodicity. Similar study[4] has also been performed on the time series formed by the frequency of hashtags in Twitter. In these studies, the time varying frequency is readily available as a representative time series for each token (keyword/query/hashtag), where new time series algorithms are proposed to detect periodicity.

In this paper, we go beyond individual token based methods and propose a *topic* based periodicity discovery framework named `Torpedo`. A topic is characterized by both a word distribution over the entire vocabulary as in standard topic models such as PLSA[5] and LDA,[6] and a time distribution over the time range which the corpus covers. We argue that individual tokens suffer from poor explaining power. Single keywords may be either too narrow or too broad, while a topic conveys more accurate information of the periodic pattern. In the rest of this paper, we use periodic topics and periodic patterns interchangeably when there is no confusion.

Further author information: (Send correspondence to Jingjing Wang)
Jingjing Wang: E-mail: jwang112@illinois.edu
Hongbo Deng: E-mail: hbdeng@yahoo-inc.com
Jiawei Han: Email: hanj@illinois.edu
*http://www.google.com/trends/

Next-Generation Analyst III, edited by Barbara D. Broome, Timothy P. Hanratty, David L. Hall, James Llinas,
Proc. of SPIE Vol. 9499, 94990A · © 2015 SPIE · CCC code: 0277-786X/15/$18 · doi: 10.1117/12.2180097

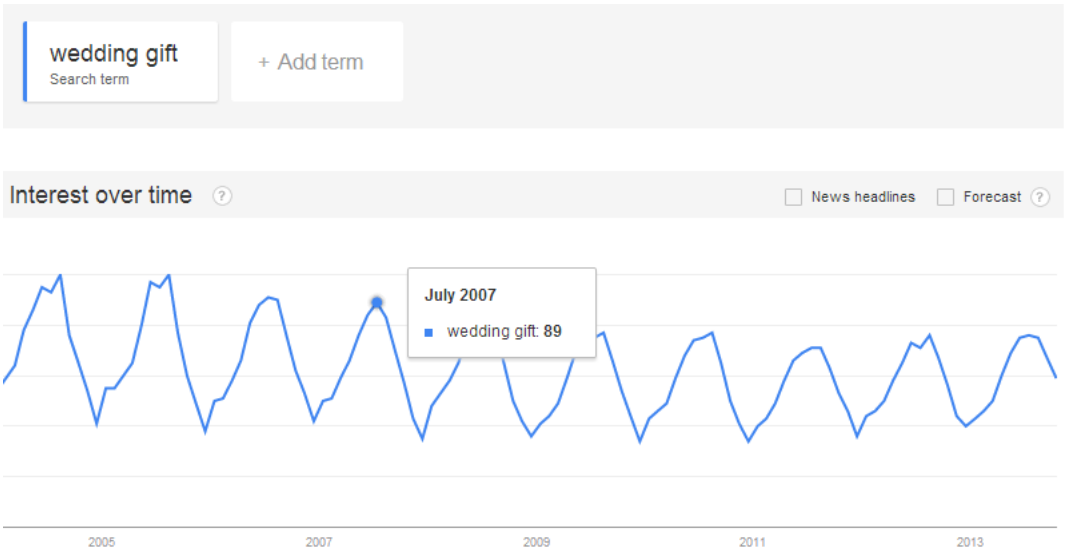

Figure 1. The trend of `wedding+gift` in Google Search

To approach the problem of periodic topic discovery, we propose a two-step generic framework which takes advantage of both advanced topic modeling technology and the well-studied time series periodicity discovery techniques. First we transform time-stamped text data into topic specific time series by a time dependent topic modeling module. Then we analyze the time series to find any inherent periodicity for each topic by a periodicity discovery module. Each module can accommodate various state-of-art techniques to meet the requirements from various applications. Example 1 illustrates our overall framework.

EXAMPLE 1 (PERIODIC TOPIC DISCOVERY). *Suppose we have the query log of an online search engine for 4 years from Jan. 2010 to Dec. 2013. We discretize the time range into 48 months denoted by* $[1, 2, 3, ..., 48]$.

By the time dependent topic modeling module, we discover a set of topics, each of which is represented by a word distribution and a time distribution. Let us say one topic has a word distribution $\{$*wedding:*0.3*, gift:*0.3*, ceremony:*0.2*, friend:*0.1*,...*$\}$ *and a time distribution* $\{p(t)\}_{t=1,2,3,...,48}$ *as follows:*

$$\begin{cases} p(7) = p(19) = p(31) = p(43) = 0.1 \\ p(t) = 0.01364, t \in [1, 2, 3, ..., 48], \text{ and } t \neq 7, 19, 31, 43 \end{cases}$$

Another topic has a word distribution $\{$*japanese:*0.4*, tsunami:*0.3*, earthquake:*0.2*,...*$\}$ *and a time distribution as follows:*

$$\begin{cases} p(15) = 0.5,\ p(16) = 0.3,\ p(17) = 0.1 \\ p(t) = 0.0022, t \in [1, 2, 3, ..., 48], \text{ and } t \neq 15, 16, 17 \end{cases}$$

The time distribution of each topic is viewed as a standard time series. Now we apply the periodicity discovery module to detect the period in the time series. The first topic can be identified as an yearly periodic pattern while the second one has no periodicity.

The core of our problem is how to instantiate the two modules. The time dependent topic modeling module is to bridge the gap between topic discovery and periodicity discovery, which is the main focus of this paper. We carefully design this module with the following intuitions: 1) Top ranked words in a topic should often co-occur in documents. 2) Top ranked words in a topic should often co-occur around the same time. 3) The time distribution of a topic should accommodate the imbalance of the corpus over time. 4) The model should be robust to noise and able to exclude non-discriminative words. For the time series periodicity discovery module, we adapt a state-of-art method named `AUTOPERIOD` which involves a combination of Discrete Fourier transform (DFT) and circular autocorrelation proposed in.[7] Nevertheless, any standard method can well achieve the goal. Further discussion of the time series techniques on periodicity discovery is beyond the scope of this paper. We demonstrate the effectiveness of `Torpedo` with three real world datasets.

Contributions We make the following contributions:

- We identify the novel problem of periodic topic discovery and propose a simple yet effective generic framework to discover periodic topics in text data.
- We propose novel instantiations of the time dependent topic modeling module which elegantly bridge the gap between topic discovery and periodicity discovery.
- Comprehensive experiments on three real world datasets are conducted and the results demonstrate the effectiveness of `Torpedo`.

2. RELATED WORK

Token Based Periodicity Discovery The time series of the frequencies of individual tokens are utilized to discover periodicity in this line of work. Ishida[1] uses autocorrelation. Shokouhi[2] applies Holt-Winters decomposition.[8] Preotiuc-Pietro *et al.*[4] models the time series by Gaussian processes. However, a single token such as a keyword, a query or a hashtag has limited explaining power and they are very sensitive to noise. Synonyms may also blur the periodic patterns.

Time Dependent Topic Modeling There has been extensive study on time dependent topic discovery which contains two categories.

- Topics have fixed word distributions, but with different strength over time.[9–14]
- Topics are evolving over time, *i.e.*, they have different word distributions as time goes by.[15–17]

The nature of our task classifies it into the first category since we want to find periodic/recurring topics. Within this category, although existing work tackle the problem of discovering time-dependent topics, none of them directly deals with periodic/recurrent topic discovery. The continuous time distributions (such as Gaussian distribution or Beta distribution) they use to model the temporal behaviors of topics are unimodal and cannot capture the multiple peaks of a periodic pattern. Fitting a model with unimodal time distributions will either miss the periodic topics or bury them in other topics.

Periodicity Discovery in Standard Time Series Periodicity discovery in pure time series has been studied for long. Fourier transform and auto-correlation are the two most popular methods to detect periods[18] in general. In recent years, various methods[7,19–26] are also developed to accommodate specific applications. This paper is not targeted to develop new periodicity detection methods for time series, but takes advantage of the mature periodicity detection study. We adapt the method named `AUTOPERIOD` for our periodicity discovery module.

A Preliminary Study on Topic Periodicity The study by Yin *et al.*[14] directly relates to our work. It constructs a temporal template by a mixture Gaussian distribution for each topic, and then fits these templates to the data.There are two major issues unaddressed. First, it requires an accurate specification of the number of periodic and non-periodic topics to construct the templates beforehand, which is usually hard to obtain. Second, it requires an accurate specification of the period for each template beforehand, such as a year, a week or a day. If a periodic topic does not exactly follow its pre-specified template due to noise, or the number for each type is not accurate, this model would fail to detect meaningful topics. In sum, this model requires large amount of human intervention which is difficult to control. On the contrary, `Torpedo` automates the entire process and minimizes the unreliable human intervention.

3. PROBLEM FORMULATION

We formulate the problem of periodic topic discovery in this section.

DEFINITION 3.1 (TIMESTAMPED DOCUMENT). *A timestamped document is a document d with a timestamp t_d.*

DEFINITION 3.2 (TOPIC). *A topic in a collection of timestamped documents D is characterized by 1) a distribution over words from vocabulary V and 2) a distribution over the time range which the collection D covers. Formally, we represent a topic k by a multinomial distribution $\phi_k = \{\phi_{kv}\}_{v \in V}$ s.t. $\sum_{v \in V} \phi_{kv} = 1$. and a time distribution ψ_k. We will ellaborate on how to specify ψ_k in the following section.* Given the definitions of documents and topics, our problem is defined as follows.

PROBLEM 1 (PERIODIC TOPIC DISCOVERY). *Given a collection of timestamped documents D, discover the periodic topics, each of which is characterized by a word distribution ϕ_k, a time distribution ψ_k, and the period T_k.*

4. A GENERIC FRAMEWORK FOR PERIODIC TOPIC DISCOVERY

4.1 Time Dependent Topic Modeling Module

The goal of this module is to jointly learn the word distributions and time distributions of topics discussed in a timestamped document collection.

4.1.1 Overview

We consider the following factors when we instantiate this module.

1. Co-occurrence Assumptions. The most salient intuition in standard topic modeling is that top ranked words in a topic should often co-occur in documents. Once we want to model topics over time, it is natural to encode the analogous intuition that top ranked words in a topic should often co-occur around the same time. Therefore, we adopt the hierarchical generative structure to generate ⟨word, time⟩ pairs as in.[13] The document timestamp is appended to each word in this document.

2. The Form of Time Distributions. Unlike existing work which use continuous unimodal distributions, we propose to model the temporal behavior of a topic by a discrete multinomial distribution, which is able to capture the multiple peaks of a periodic pattern. Hence we have $\psi_k = \{\psi_{kt}\}\ s.t. \sum_t \psi_{kt} = 1$ where t goes through all the discretized time indices.

3. Dealing with Imbalanced Text Collection. The time distribution of a topic should accommodate the imbalance of the corpus over time. Consider a corpus which has a huge number of documents in one time interval t_1 but only has a few documents in another time interval t_0. It is very likely that t_1 bears more probability mass for all topics than t_0. In the extreme case where all the documents lie in t_1 and no documents in t_0, we should have $\psi_{kt_1} = 1, \psi_{kt_0} = 0$ for every topic k. To this end, we impose the collection's time distribution† as a Dirichlet prior smoothing[27] to the time distribution of each topic.

4. Dealing with Background Noise. Following the philosophy in,[28] we introduce a background topic B to attract common words and thus makes other topics concentrated on more discriminative words. Note that this background topic B is not only described by the collection's word distribution ϕ_B, but also involves the collection's time distribution ψ_B.

4.1.2 Instantiation

Now we are ready to present the instantiation of the time dependent topic modeling module. The graphical representation is shown in Fig. 2. The generative process is described as follows.

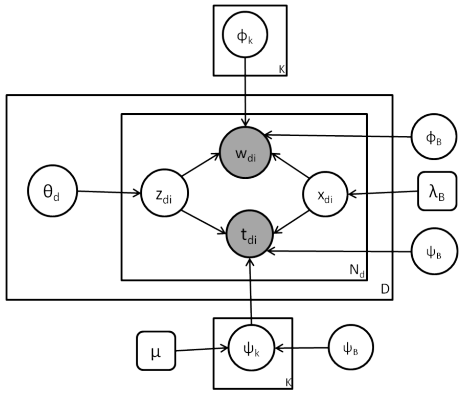

Figure 2. Graphic Model Representation

To generate each word-time tuple $\langle w_{di}, t_{di} \rangle$,

†The collection's time distribution ψ_B satisfies: ψ_{Bt} is proportional to the number of words in the time interval t:

$$\psi_{Bt} = \frac{\#\ words\ in\ the\ interval\ t}{\#\ words\ in\ the\ collection} = \frac{\sum_{v,d} n(v,t,d)}{\sum_{v,t,d} n(v,t,d)}$$

1. Draw K time distributions ψ_k from the Dirichlet prior defined by ψ_B, one for each topic k: $\psi_k \sim Dir(1 + \mu\psi_B)$. μ controls how strong our confidence is on the prior;
2. Draw a switch variable x_{di} from a Bernoulli distribution $x_{di} \sim Bernoulli(\lambda_B)$. λ_B is the topic proportion of the background topic B.
3. If $x_{di} = 1$,

   draw a word w_{di} from the background topic B: $w_{di} \sim Multi(\phi_B)$;
   draw a time index from the background topic B: $t_{di} \sim Multi(\psi_B)$

   Else,

   draw a topic z_{di} from the topic distribution θ_d: $z_{di} \sim Multi(\theta_d)$

   draw a word w_{di} from the topic z_{di}: $w_{di} \sim Multi(\phi_{z_{di}})$;
   draw a time index from the topic z_{di}: $t_{di} \sim Multi(\psi_{z_{di}})$

A Maximum A Posterior (MAP) inference can be naturally applied to this model, which can be solved by the Expectation-Maximization(EM) algorithm.[29]

4.1.3 Analysis

We provide several insights into this module.

1. The regularization effect from the temporal dimension. Applying the EM algorithm for the MAP estimator, we obtain the following posterior distribution $p(k|v, t, d)$ at each update:

$$p(k|v, t, d) = \frac{\theta_{dk}\phi_{kv}\psi_{kt}}{\sum_{k'=1}^{K} \theta_{dk'}\phi_{k'v}\psi_{k't}},\ k = 1, 2, ..., K$$

When ψ_k is uniformly distributed, the posterior probability is exactly the same as that derived from PLSA. In fact, for the calculation of $p(k|v, t, d)$, the word weight ϕ_{kv} for each topic is re-weighted by the time weight ψ_{kt} of the same topic. A word-time-document tuple $\langle v, t, d\rangle$ would be more likely to be assigned to a topic k only when the following three conditions are satisfied simultaneously: 1) The word v ranks high in topic k; 2) Topic k has a high probability mass in the time interval indexed by t; and 3) Topic k has a high weight in document d. The temporal influence on the posterior probability is then propagated in the EM updates.

2. Connection to Dividie-and-Conquer Methods. Discrete time topic modeling in a divide-and-conquer manner[17] is a very popular method. It splits a document collection into sub-collections according to discretized time intervals, finds topics in each sub-collection separately and finally 'links' them by topic similarity to form a view of the topics along the time line. It is interesting to notice that this model is a special case of our instantiation with a special initialization setting.

At the initialization stage, if we set $\psi_{kt^k} = 1$ and $\psi_{kt} = 0$ when $t \neq t^k$, which means each topic k is only active at one time interval t^k, we are actually spliting the topic set to subsets corresponding to each interval t: $S_t = \{k|\psi_{kt} = 1\}$. For each interval t, in the E-step, the words in documents with time t will only contribute to the posterior of the topics in S_t. Then in the M-step, documents with time index t will get non-zero θ_{dk}'s only for topics in S_t. Then when we go back again to the E-step, regardless of how the updated $\{\psi_k\}$ distribute, $p(k|\cdot, \cdot, d)(k \notin S_{t_d})$ would be zero. Thus each sub-collection of documents within one time interval is forming a "close" set in terms of topic distribution propagation. And the initial assignments of $\{\psi_k\}$ determine the number of active topics in each time interval. If these numbers are aligned with the number of topics in the aforementioned divide-and-conquer model, they would be equivalent.

3. Complexity Analysis. Since all the words in each document share the same timestamp, they are associated with the same time index t_d. Therefore the variable t is not adding complexity although it introduces a new dimension. The complexity for the inference of the generative model is $O(Iter \cdot K(|V| + |W| + |D| + |I|))$, where $Iter$ is the number of EM iterations, K is the number of topics, $|V|$ is the vocabulary size, $|W|$ is the total number of words in the collection, $|D|$ is the number of documents and $|I|$ is the number of discretized time indices.

4. Time Discretization With the proposed smoothing technique for ψ_k, the time discretization does not necessarily have to be uniform.

Suppose the document collection covers a time range $[p, q]$[‡] and we want to discretize it into $|I|$ sub-intervals. Then any r_i's which satisfy $\sum_{i=1}^{|I|} r_i = q - p, r_i > 0$ can be used for discretization. The $|I|$ sub-intervals are $\{[p, p + r_1], [p + r_1, p +$

[‡] p, q are unix timestamps

Table 1. Statistics of the Datasets

	$\|D\|$	$\|V\|$	period	topics
DBLP	4,070	2,132	1 year	6 conferences
Flickr	84,244	7,524	1 year	5 music festivals
QryLogA	397,149	28,884	-	-
QryLogB	63,884	12,240	-	-

$r_1 + r_2], ..., [p + \sum_{i=1}^{|I|-1} r_i, q(= p + \sum_{i=1}^{|I|} r_i)]\}$. This flexibility allows us to try different time discretizations to understand the data in different scales. In practice, the most interesting periods can be a day, a week, or a year. The discretization can be adjusted accordingly to accommodate the different scales. If there is completely no prior knowledge, we can try different discretizations and output all the results for investigation.

4.2 Periodicity Discovery Module

The per-topic time distributions $\{\psi_k\}$ represent the temporal behaviors of the topics. Each ψ_k is a discrete time signal. The nature of such signals requires us to carefully address two difficulties: 1) the sequence with periodicity can be noisy due to the fact that topics could appear at any time, could be missing for some intervals, and could deviate from the regular periodic paths; and 2) the time series might be short, *i.e.*, only lasts for a few periodic intervals, which will cause a simple fourier transform not applicable.

To address these issues, we adapt a method named `AUTOPERIOD` proposed in[7] which involves a combination of Discrete Fourier transform (DFT) and circular autocorrelation. First, DFT is applied to the time series to generate a set of candidate periods by extracting peaks from the periodogram, then autocorrelation follows to identify the true period out of the false positives by examining if the period is located on a hill (as opposed to a valley) on the autocorrelation function (ACF). This approach to a large extent alleviates the major problems when applying DFT or autocorrelation alone for periodicity detection. Specifically, for DFT, false positives caused by spectral leakage[30] can be filtered out by the ACF validation step; for ACF, the huge number of candidate periods (peaks on the ACF) is avoided by first using DFT to generate candidates. Since both DFT and ACF can be computed by Fast Fourier Transform (FFT),[31] the overall computational complexity is $O(n \log(n))$, where n is the length of the original time sequence.

5. EXPERIMENT

5.1 Datasets

We use three datasets in this paper. We have ground truth of the periods and topics for the first two datasets while we have no information for the third one other than the plain text. The statistics of the datasets are given in Table 1.

1. DBLP This dataset consists of titles taken from six conferences - WWW, SIGMOD, SIGIR, KDD, VLDB and NIPS in the time span 2003 to 2007. The timestamps of the paper titles are determined according to the conference programs.

2. Flickr This dataset is constructed from the photo sharing website Flickr[§]. Photos from 2006 to 2010 including keywords SXSW (South by Southwest), Coachella, Bonnaroo, Lollapalooza and ACL (Austin City Limits) are crawled. They correspond to five music festivals. The tag of a photo is considered as document text, while the time when it was taken is used as the document timestamp.

3. QryLog This dataset is obtained by sampling the query log of a commercial search engine. It contains two sub datasets:
1) *QryLogA* contains a sample of queries containing the keywords "thanksgiving" or "christmas" between Sep. 2009 and Jul. 2014.
2) *QryLogB* contains a sample of queries containing the keywords "weekend" or "Friday night" or "tv shows" between Jan. 2014 and Jul 2014. The time the query was issued is used as the timestamp for the query.

[§]http://www.flickr.com/

Table 2. Word Distributions of the Discovered Topics for *DBLP* after Time Swap (5 Topics). Red: words which are highly indicative in *other topics*

Torpedo					LPTA				
Topic 1 DM	Topic 2 IR	Topic 3 ML	Topic 4 WEB	Topic 5 DB	Topic 1 -	Topic 2 -	Topic 3 -	Topic 4 -	Topic 5 -
mining 0.0744	retrieval 0.0719	learning 0.0519	web 0.1487	data 0.0631	data 0.0442	retrieval 0.0247	learning 0.0291	web 0.0836	data 0.0445
data 0.0531	text 0.0266	models 0.0148	semantic 0.0351	xml 0.0391	mining 0.0185	data 0.0197	based 0.0116	search 0.0223	query 0.0228
clustering 0.0252	search 0.0228	bayesian 0.0145	search 0.0341	query 0.0319	based 0.0175	query 0.0145	models 0.0113	based 0.0216	database 0.0186
detection 0.0214	based 0.0217	classification 0.0135	services 0.0189	database 0.0301	clustering 0.0141	based 0.0144	classification 0.0105	semantic 0.0212	xml 0.0183
time 0.0178	document 0.0213	gaussian 0.0132	based 0.0185	queries 0.0282	efficient 0.0140	search 0.0129	model 0.0105	xml 0.0109	processing 0.0150
patterns 0.0168	relevance 0.0162	kernel 0.0113	peer 0.0123	processing 0.0243	large 0.0139	web 0.0128	retrieval 0.0090	services 0.0102	efficient 0.0132
discovery 0.0148	language 0.0157	model 0.0095	content 0.0117	databases 0.0240	xml 0.0135	mining 0.0116	analysis 0.0085	data 0.0083	queries 0.0129
frequent 0.0139	query 0.0151	inference 0.0092	engine 0.0116	efficient 0.0176	databases 0.0131	text 0.0112	gaussian 0.0080	content 0.0074	databases 0.0129
privacy 0.0105	evaluation 0.0138	markov 0.0089	applications 0.0109	streams 0.0166	web 0.0102	detection 0.0102	data 0.0080	approach 0.0067	streams 0.0119
efficient 0.0098	feedback 0.0132	sparse 0.0085	service 0.0106	management 0.0162	queries 0.0100	learning 0.0101	clustering 0.0075	service 0.0067	web 0.0107

5.2 Performance Studies

Token based periodicity discovery methods lack explaining power and are sensitive to noise. Standard topic models such as PLSA and LDA do not consider the temporal dimension. We can expect worse performances from them on the periodic topic discovery task. A comparative study of (periodic) topics discovered by token based methods, PLSA and LDA is given in[14] thus we do not involve the comparison with them here. In this paper, we focus our empirical study on the comparison with the LPTA model[14] and new knowledge discovery.

By using the first two datasets which have ground truth, we validate our approach and compare Torpedo with LPTA. With the third dataset, we provide a sample of the periodic topics discovered from the query log. Due to the lack of verifiable "true" results, most of our results are concluded from qualitative study[¶] [‖].

5.2.1 Comparison with LPTA

With the correct period and number of topics, both Torpedo and LPTA are able to correctly identify the periodic topics. However, if the periodic patterns become less straightforward, Torpedo exhibits strong advantage over LPTA. We simulate the following two scenarios.

(1) *Occasional deviation:* We swap the timestamps of NIPS'2006 documents and SIGIR'2005 documents in the *DBLP* dataset to simulate this scenario. The topics are now recurring over time with no strict periods. We set the topic number to be 5. In this new setting, results obtained from Torpedo and LPTA are shown in Table 2. The five yearly periodic topics can still be identified by Torpedo. In contrast, the topics identified by LPTA contain many non-discriminative words which make topics similar to each other. Keywords in several areas appear simultaneously in the same topic and common words such as *data* ranks high in almost every topic. We can hardly identify semantically coherent topics.

(2) *A mixture of periodic topics with non-periodic topics:* A subset from the *Flickr* dataset is constructed in which all photos related to SXSW and ACL festivals from 2006 to 2010 are kept but those related to Coachella and Lollapalooza are kept only if they were taken in the year 2009. This simulates a dataset with 2 periodic topics and 2 non-periodic topics. While LPTA requires an accurate specification of the number of topics for each type of pattern, Torpedo does not have this requirement. We set the total number of topics as 4 and run Torpedo. Results are shown in Table 3. Two periodic topics with period 1 year and two non-periodic topics are identified. However, if we use the parameters of 1 non-periodic topic and 3 periodic topics for LPTA, the resulting topics are completely messed up. Results are shown in Table 3.

In the above new settings, LPTA fails to extract meaningful topics while Torpedo still maintains good performance. In fact, for periodic topics, LPTA forces the time distribution to be a mixture Gaussian distribution based on a prespecified period, where each Gaussian component captures an aggregate peak of the topic. Therefore, when the periodic patterns become noisy or the period is incorrectly specified, a prespecified template no longer fits the data. In contrast, the time distribution in Torpedo can capture very flexible behaviors and automatically identify the period.

5.2.2 Periodic Topic Discovery in the Query Log Dataset

We set the number of topics to be 50 and 20 respectively for *QryLogA* and *QryLogB*. A sample of the discovered periodic topics are shown in Table 4 and 5. To make more sense out of the topics, we employ simple heuristics[**] to obtain a set of

[¶] Perplexity is not a good metric here because our model is not designed to predict but to discover.

[‖] For all the experiments, the background topic proportion λ_B is set to 0.75 based on the empirical study in.[28,32]

[**] We employ simple rules such as a) the query contains top ranked words in a topic; b) the query was issued around the peak of the periodic topic's time distribution; c) the query d has high proportion of the topic k, *i.e.* θ_{dk} is large.

Table 3. Word Distributions of the Discovered Topics for *Flickr* (4 Topics). Blue: highly indicative words

Torpedo				LPTA			
Topic 1 Coachella	Topic 2 ACL	Topic 3 Lollapalooza	Topic 4 SXSW	Topic 1 -	Topic 2 -	Topic 3 -	Topic 4 -
Coachella 0.24857	acl 0.11969	Lollapalooza 0.1797	SXSW 0.1797	Coachella 0.1068	SXSW 0.0904	austin 0.0611	acl 0.1010
indio 0.069337	austin 0.088545	chicago 0.098066	austin 0.098066	Lollapalooza 0.0800	austin 0.0834	acl 0.0606	austin 0.0874
california 0.048874	music 0.059572	grantpark 0.037875	texas 0.037875	music 0.0458	texas 0.0715	austincitylimits 0.0579	music 0.0743
music 0.031708	austincitylimits 0.057641	concert 0.034442	southbysouthwest 0.034442	chicago 0.0436	music 0.0663	music 0.0502	limits 0.0471
desert 0.018342	limits 0.053187	august 0.023338	music 0.023338	concert 0.0338	southbysouthwest 0.0426	texas 0.0457	city 0.0471
art 0.013509	city 0.05303	illinois 0.012338	live 0.012338	indio 0.0298	live 0.0349	SXSW 0.0438	austincitylimits 0.0462
musicfestival 0.0090553	texas 0.042465	lolla 0.010142	atx 0.010142	california 0.0210	concert 0.0280	concert 0.0331	texas 0.0455
Coachellamusicandartsfestival 0.0085571	zilker 0.022432	stevegalli 0.0083197	concert 0.0083197	live 0.0178	atx 0.0162	live 0.0290	concert 0.0239
live 0.0079914	photos 0.01916	backstagegallery 0.0083197	downtown 0.0083197	grantpark 0.0169	downtown 0.0118	southbysouthwest 0.0182	live 0.0198
party 0.007934	ron 0.018891	performance 0.0081966	gig 0.0081966	august 0.0104	gig 0.0112	zilker 0.0145	zilker 0.0185

Table 4. Top Ranked Words and Queries for A Sample of Yearly Periodic Topics Discovered in *QryLogA*. The labels in the top row is annotated manually by examining the results.

thanksgiving dinner		thanksgiving sale		nightmare before christmas (a movie) & related products		christmas tree	
Topic1	Top Ranked Queries	Topic2	Top Ranked Queries	Topic3	Top Ranked Queries	Topic4	Top Ranked Queries
thanksgiving	prepare thanksgiving dinner	day	walmart thanksgiving specials	nightmare	the nightmare before christmas bedding	gifts	used christmas tree baler
dinner	prepare for thanksgiving dinner	thanksgiving	walmart thanksgiving day sale	store	the nightmare before christmas stockings	tree	christmas tree drop off gilbert az
decorating	thanksgiving holiday ideas	sale	day after thanksgiving sale	stockings	nightmare before christmas sally wig	activities	yarn christmas tree
holiday	thanksgiving dinner on a budget	clip	walmart day after thanksgiving	patterns	the nightmare before christmas stickers	lighted	gorgeous christmas tree
turkey	thanksgiving turkey dinner take out	specials	thanksgiving specials in new york	christmas	director of nightmare before christmas	school	rocking christmas tree
ideas	turkey and thanksgiving	macy's	thanksgiving day sale at walmart	cactus	characters of the nightmare before christmas	bulbs	how to fluff a christmas tree branches
park	turkey thanksgiving	art	clip art thanksgiving day	clipart	nightmare before christmas sunglasses	girl	sears christmas tree
market	thanksgiving turkey disguise	walmart	what will walmart have on sale the day after thanksgiving	shops	nightmare before christmas sally costumes	2012	mini christmas tree bulbs
chords	thanksgiving turkey dinner ideas	city	kmart day after thanksgiving sale	town	sally from nightmare before christmas costumes	holidays	lighted spiral christmas tree
childrens	thanksgiving dinner reservations	york	macy's thanksgiving parade new york city	costumes	nightmare before christmas patterns	invitation	how to donate christmas tree

Table 5. Top Ranked Words and Queries for A Sample of Weekly Periodic Topics Discovered in *QryLogB*. The labels in the top row is annotated manually by examining the results.

weekend furniture store in st. louis		online tv shows		wwe smackdown (a tv program on friday night)		beach weekend getaway	
Topic1	Top Ranked Queries	Topic2	Top Ranked Queries	Topic3	Top Ranked Queries	Topic4	Top Ranked Queries
weekends	weekends only furniture store in st. louis	episodes	hulu free tv shows full episodes	night	wwe friday night smackdown results for last night	nyc	things to do this weekend nyc
louis	weekends only furniture mo	full	cbs full episodes tv shows	friday	friday night smackdown wwe	weekend	things to do long island this weekend
st	furniture sales this weekend	packages	tv shows with new episodes	lights	what channel is friday night smackdown	beach	events in daytona beach this weekend
furniture	weekends only furniture st peters mo	hulu	hulu online tv shows full episodes	smackdown	wwe smackdown wwe friday night smackdown	va	weekend beach getaway packages
deals	weekends only furniture st louis mo	season	best tv shows on hulu	live	wwe smackdown on friday night	miami	what to do in daytona beach this weekend
mo	weekends furniture st louis	cbs	full season tv shows	fox	friday night smackdown live	long	things to do miami this weekend
freedom	weekend only furniture st louis	1990s	hulu tv shows	activities	friday night smackdown results 2014	events	things to do in miami this weekend
st.	weekends only furniture store st. louis mo	4	tv shows online full episodes	tykes	last friday night smackdown	houston	urban beach weekend
streaming	weekends only furniture in st. louis	3	tv shows on hulu plus	wwe	results wwe last friday night smackdown	getaway	weekend getaway packages
cape	weekends only furniture st. louis mo	paid	cbs tv shows online full episodes	spa	wwe friday night smackdown results	things	weekend getaway in nyc

relevant queries for each topic. It is interesting to see that the model can discover many meaningful periodic topics. Each topic corresponds to a very coherent set of queries. These knowledge can naturally benefit a broad spectrum of search engine applications such as query auto-completion and query clustering providing the periodicity dimension.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we study the problem of periodic topic discovery. We propose a generic framework including a time dependent topic modeling module and a periodicity discovery module. Novel instantiations of the time dependent topic modeling module are proposed and thoroughly studied both theoretically and experimentally. For future work, we would like to explore how to benefit different applications with the discovered periodic topics.

ACKNOWLEDGMENTS

This work was sponsored in part by the U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), the Army Research Office under Cooperative Agreement No. W911NF-13-1-0193, National Science Foundation IIS-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, NIH Big Data to Knowledge (BD2K) (U54), MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC, and U.S. Office of Naval Research (ONR) contracted with Intelligent Automation Inc. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

REFERENCES

[1] Ishida, K., "Periodic topic mining from massive amounts of data," in [*TAAI*], 379 –386 (nov. 2010).

[2] Shokouhi, M., "Detecting seasonal queries by time-series analysis," in [*Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*], *SIGIR '11*, 1171–1172, ACM, New York, NY, USA (2011).

[3] Vlachos, M., Meek, C., Vagena, Z., and Gunopulos, D., "Identifying similarities, periodicities and bursts for online search queries," in [*Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*], *SIGMOD '04*, 131–142, ACM, New York, NY, USA (2004).

[4] Preotiuc-Pietro, D. and Cohn, T., "A temporal model of text periodicities using gaussian processes," in [*EMNLP*], 977–988 (2013).

[5] Hofmann, T., "Probabilistic latent semantic indexing," in [*SIGIR*], 50–57 (1999).

[6] Blei, D. M., Ng, A. Y., and Jordan, M. I., "Latent dirichlet allocation," in [*NIPS*], 601–608 (2001).

[7] Vlachos, M., Yu, P. S., and Castelli, V., "On periodicity detection and structural periodic similarity," in [*SDM*], (2005).

[8] Goodwin, P., "The holt-winters approach to exponential smoothing: 50 years old and going strong," *Foresight* , 30–34 (2010).

[9] Jo, Y., Hopcroft, J. E., and Lagoze, C., "The web of topics: discovering the topology of topic evolution in a corpus," in [*WWW*], 257–266, ACM, New York, NY, USA (2011).

[10] Hong, L., Dom, B., Gurumurthy, S., and Tsioutsiouliklis, K., "A time-dependent topic model for multiple text streams," in [*KDD*], 832–840, ACM, New York, NY, USA (2011).

[11] Mei, Q., Liu, C., Su, H., and Zhai, C., "A probabilistic approach to spatiotemporal theme pattern mining on weblogs," in [*WWW*], 533–542, ACM, New York, NY, USA (2006).

[12] Iwata, T., Yamada, T., Sakurai, Y., and Ueda, N., "Online multiscale dynamic topic models," in [*KDD*], 663–672, ACM, New York, NY, USA (2010).

[13] Wang, X. and McCallum, A., "Topics over time: a non-markov continuous-time model of topical trends," in [*KDD*], 424–433, ACM, New York, NY, USA (2006).

[14] Yin, Z., Cao, L., Han, J., Zhai, C., and Huang, T. S., "Lpta: A probabilistic model for latent periodic topic analysis," in [*ICDM*], 904–913 (2011).

[15] Wang, C., Blei, D. M., and Heckerman, D., "Continuous time dynamic topic models," in [*UAI*], 579–586 (2008).

[16] Blei, D. M. and Lafferty, J. D., "Dynamic topic models," in [*ICML*], 113–120 (2006).

[17] Mei, Q. and Zhai, C., "Discovering evolutionary theme patterns from text: an exploration of temporal text mining," in [*KDD*], 198–207, ACM, New York, NY, USA (2005).

[18] Harris, B., [*Spectral analysis of time series*], Wiley (1967).

[19] Rasheed, F. and Alhajj, R., "Periodic pattern analysis of non-uniformly sampled stock market data," *Intell. Data Anal.* **16**(6), 993–1011 (2012).
[20] Rasheed, F., Al-Shalalfa, M., and Alhajj, R., "Efficient periodicity mining in time series databases using suffix trees," *IEEE Trans. Knowl. Data Eng.* **23**(1), 79–94 (2011).
[21] Elfeky, M. G., Aref, W. G., and Elmagarmid, A. K., "Warp: Time warping for periodicity detection," in [*ICDM*], 138–145, IEEE Computer Society (2005).
[22] Elfeky, M. G., Aref, W. G., and Elmagarmid, A. K., "Periodicity detection in time series databases," *IEEE Trans. Knowl. Data Eng.* **17**(7), 875–887 (2005).
[23] Keogh, E. J., Wei, L., Xi, X., Vlachos, M., Lee, S.-H., and Protopapas, P., "Supporting exact indexing of arbitrarily rotated shapes and periodic time series under euclidean and warping distance measures," *VLDB J.* **18**(3), 611–630 (2009).
[24] Vlachos, M., Yu, P. S., Castelli, V., and Meek, C., "Structural periodic measures for time-series data," *Data Min. Knowl. Discov.* **12**(1), 1–28 (2006).
[25] Yang, J., Wang, W., and Yu, P. S., "Mining asynchronous periodic patterns in time series data," *IEEE Trans. Knowl. Data Eng.* **15**(3), 613–628 (2003).
[26] Glynn, E. F., Chen, J., and Mushegian, A. R., "Detecting periodic patterns in unevenly spaced gene expression time series using lomb-scargle periodograms," *Bioinformatics* **22**(3), 310–316 (2006).
[27] Zhai, C. and Lafferty, J. D., "A study of smoothing methods for language models applied to ad hoc information retrieval," in [*SIGIR*], 334–342 (2001).
[28] Zhai, C., Velivelli, A., and Yu, B., "A cross-collection mixture model for comparative text mining," in [*KDD*], 743–748 (2004).
[29] Dempster, A. P., Laird, N. M., and Rubin, D. B., "Maximum likelihood from incomplete data via the em algorithm," *J. R. Stat. Soc.* **39**(1), 1–38 (1977).
[30] Harris, F. J., "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proc. of the IEEE* **66**(1), 51–83 (1978).
[31] Cooley, J. and Tukey, J., "An algorithm for the machine calculation of complex fourier series," *Mathematics of Computation* **19**(90), 297–301 (1965).
[32] Mei, Q., Cai, D., Zhang, D., and Zhai, C., "Topic modeling with network regularization," in [*WWW*], 101–110 (2008).