Timeline Summarization from Social Media with Life Cycle Models

Yi Chang,[♯] **Jiliang Tang,**[♯] **Dawei Yin,**[♯] **Makoto Yamada,**[†] **and Yan Liu**[‡]
[♯]Yahoo Inc., {yichang,jlt,daweiy}@yahoo-inc.com
[†]Institute for Chemical Research, Kyoto University, myamada@kuicr.kyoto-u.ac.jp
[‡]Department of Computer Science, University of Southern California, yanliu.cs@usc.edu

Abstract

The popularity of social media shatters the barrier for online users to create and share information at any place at any time. As a consequence, it has become increasing difficult to locate relevance information about an entity. Timeline has been proven to provide an effective and efficient access to understand an entity by displaying a list of episodes about the entity in chronological order. However, summarizing the timeline about an entity with social media data faces new challenges. First, key timeline episodes about the entity are typically unavailable in existing social media services. Second, the short, noisy and informal nature of social media posts determines that only content-based summarization could be insufficient. In this paper, we investigate the problem of timeline summarization and propose a novel framework Timeline-Sumy, which consists of episode detecting and summary ranking. In episode detecting, we explicitly model temporal information with life cycle models to detect timeline episodes since episodes usually exhibit sudden-rise-and-heavy-tail patterns on time-series. In summary ranking, we rank social media posts in each episode via a learning-to-rank approach. The experimental results on social media datasets demonstrate the effectiveness of the proposed framework.

1 Introduction

Social media is growing with an explosive rate and it becomes increasingly difficult for online users to locate useful information or obtain high-level digest about entities from massive and high-velocity social media data. For example, no existing social media sites could automatically answer questions as "what are the major activities of Lionel Messi during World Cup 2014?" and "what's the gossips about Jennifer Lopez in 2014 summer?" Timeline, which displays a sequence of episodes (e.g., news or gossips in social media) about an entity (e.g., a celebrity or a company) in chronological order, provides an effective solution to above questions by enabling a more efficient access to understand the entity. Therefore automatically summarizing timeline from rich social media data is in great need.

We have witnessed several timeline summarization algorithms for news corpus [Hu *et al.*, 2011; Yan *et al.*, 2011], however, timeline summarization with social media data faces new challenges. First, a typical process for timeline summarization by humans includes identifying key timeline episodes about the entity and then summarizing each episode; however, key timeline episodes about an entity are often not explicitly available though it is easy to collect a large amount of social media data about the entity. Second, social media posts are usually short, informal and noisy, which indicates that merely making use of content could be insufficient to build an effective timeline summarizer. On the other hand, rich social media data provides unique opportunities for the task of timeline summarization. First, temporal information is pervasively associated with social media data and presents useful patterns complementary to content information. For example, given a popular entity, if we count its frequency on social media for a pre-determined time interval (e.g., an hour), we could generate a time-series over time; the temporal shape of each timeline episode usually exhibits a sudden-spike-and-heavy-tail pattern [Chang *et al.*, 2014] as social media users' interests shift quickly. Second, social media provides various types of signals that can be explored to advance the timeline summarization task such as hashtag information, popularity of each user, and engagement of each social media post. The aforementioned challenges suggest that traditional textual based summarization algorithms are inadequate; while these unique opportunities indicate that it is highly possible to develop novel algorithms for timeline summarization by harnessing the power of social media. There are several very recent timeline summarization algorithms for personalized social media streams [Ren *et al.*, 2013; Li and Cardie, 2014]. However, none of them fully take advantage of opportunities from social media data such as the temporal patterns of timeline episodes.

In this paper, we study the problem of timeline summarization, which targets to provide timeline episode summarization along time axis about an entity. We propose a novel timeline summarization framework, Timeline-Sumy. It consists of episode detecting and summary ranking analogous to a typical process for humans to generate timeline summary. Episode detecting aims to identify key episodes in a timeline; while summary ranking desires to select representative and informative social media posts for each episode via a learning-to-rank (LTR) approach. The major contributions of this paper are summarized as below: 1.) We explicitly model sudden-spike-and-heavy-tail shapes on time-series via life cycle models for timeline summarization; 2.) We develop a novel Bayesian nonparametric model to simultaneously capture content and temporal information for episode detecting; and 3.) We propose a framework Timeline-Sumy which is flexible to integrate various types of signals from social media data for timeline summarization.

2 The Proposed Framework

2.1 Timeline Episode Detecting

We propose a nonparametric generative model where social media posts with temporal information are observations and timeline episodes are latent variables to be detected.

Modeling Temporal Information: Timeline episodes have some unique temporal properties. For example, they present life cycle patterns and they usually emerge or change dramatically. Those properties are seldom captured by existing approaches such as topic evolution approaches [Blei and Lafferty, 2006]. Therefore Timeline-Sumy uses life circle models [Levitt, 1965] to capture temporal information. We use T to denote temporal information for the entity with D posts and adopt a Gamma distribution with parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to model temporal information, which can capture life cycles with sudden-spike-and-heavy-tail patterns [Chang *et al.*, 2014] as:

$$p(\boldsymbol{t}|\alpha_k,\beta_k)=\frac{\beta_k^{\alpha_k}}{\Gamma(\alpha_k)}t^{\alpha_k-1}e^{-\beta_k t},\ \ \Gamma(a)=\int_0^\infty t^{a-1}e^{-t}\mathrm{d}t.$$

Modeling Content Information: Social media posts usually contain hashtag labels; hence there are two types of content, i.e., regular content and hashtag content. When we model content information, we need to distinguish them. The reasons are two-fold. First, regular content is usually short, informal and highly unstructured; while hashtag content provides informative signals to indicate the labels or metadata tags of posts [Romero *et al.*, 2011]. Second, hashtag information is usually very sparse [Kwak *et al.*, 2010]. For example, around 10% of social media posts contain hashtag labels. Therefore the importance of hashtag content could be overwhelmed by regular content if we mix them. For each episode with D social media posts, we use C to denote regular content and L to denote hashtag content. Then we use multinomial distributions with parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ to model C and L, separately as:

$$p(\boldsymbol{c}|\boldsymbol{\theta}_k)=\frac{V!}{\prod_{i=1}^V f(c_i)!}\prod_{i=1}^V \theta_{ki}^{f(c_i)},$$

$$p(\boldsymbol{l}|\boldsymbol{\theta'}_k)=\frac{V!}{\prod_{i=1}^V f(l_i)!}\prod_{i=1}^V {\theta'}_{ki}^{f(l_i)},$$

where V is the number of tokens, $f(c_i)$ refers to the term frequency of token c_i in a post, and $f(l_i)$ is the term frequency of token l_i in a post. In order to handle those posts that do not belong to any timeline episodes, we assume that they belong to a background episode.

Determining the Number of Timeline Episodes: The number of episodes can strongly affect the quality of a timeline, thus automatically determining the number of episodes is crucial for our timeline episode detection task. We adopt a Bayesian nonparametric method to determine the number of episodes K. We assume that Z follows Chinese Restaurant Process with the parameter τ. To make the model fully Bayesian, we place the conjugate priors on the model parameters $\boldsymbol{\theta}$, $\boldsymbol{\theta}'$ and $\boldsymbol{\alpha},\boldsymbol{\beta}$ respectively as: 1.) multinomial distribution $\boldsymbol{\theta}$ has the Dirichlet prior $\mathrm{Dir}(\boldsymbol{\eta})$; 2.) multinomial distribution $\boldsymbol{\theta}'$ has the Dirichlet prior $\mathrm{Dir}(\boldsymbol{\eta}')$ and 3.) $p(\alpha,\beta|\hat{p},\hat{q},\hat{r},\hat{s})$ is the conjugate prior of Gamma distribution: $p(\alpha,\beta|\hat{p},\hat{q},\hat{r},\hat{s})\propto\frac{\hat{p}^{\alpha-1}e^{-\beta\hat{q}}}{\Gamma(\alpha)^{\hat{r}}\beta^{-\alpha\hat{s}}}$. The generative process is described as follows.

for each post j, **do**
  Draw $z_j\sim\mathrm{CRP}(\tau)$
  if z_j is a new episode **then**
    draw $\theta_{z_j}\sim\mathrm{Dir}(\boldsymbol{\eta})$ and $\theta'_{z_j}\sim\mathrm{Dir}(\boldsymbol{\eta}')$
    draw $\alpha_{z_j},\beta_{z_j}\sim p(\alpha,\beta|\hat{p},\hat{q},\hat{r},\hat{s})$
  end if
  Draw $C_j\sim\mathrm{Multinomial}(\theta_{z_j})$
  Draw $L_j\sim\mathrm{Multinomial}(\theta'_{z_j})$
  Draw $t_j\sim\mathrm{Gamma}(\alpha_{z_j},\beta_{z_j})$
end for

Model Inference: Since the calculation of posterior distribution is intractable, we use numerical approximations to infer the model. Gibbs Sampling is a Markov Chain Monte Carlo (MCMC) algorithm, and its basic idea is to sample from conditional distributions of variables iteratively. In this paper, we employ Collapsed Gibbs Sampling to derive the conditional distribution Z and α,β of our proposed nonparametric generative model by integrating out θ and θ'. To sample $\boldsymbol{z}$, following the generative process, for each post j, we have the conditional distribution:

$$\begin{aligned}
& p(z_j=h|\boldsymbol{z}_{-j},\boldsymbol{C},\boldsymbol{L},\boldsymbol{t},\boldsymbol{\tau},\boldsymbol{\eta},\boldsymbol{\eta}',\boldsymbol{\alpha},\boldsymbol{\beta})\\
= & p(\boldsymbol{C},\boldsymbol{L},\boldsymbol{t}|\boldsymbol{z},\boldsymbol{\tau},\boldsymbol{\eta},\boldsymbol{\eta}',\boldsymbol{\alpha},\boldsymbol{\beta})p(z_j=h|\boldsymbol{z}_{-j})\\
= & \int p(\boldsymbol{C}|\boldsymbol{z},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\eta})d\boldsymbol{\theta}\\
& \cdot\int p(\boldsymbol{L}|\boldsymbol{z},\boldsymbol{\theta}')p(\boldsymbol{\theta}'|\boldsymbol{\eta}')d\boldsymbol{\theta}'\\
& \cdot p(\boldsymbol{t}|\boldsymbol{z},\boldsymbol{\alpha},\boldsymbol{\beta})p(z_j=h|\boldsymbol{z}_{-j},\boldsymbol{\tau})\\
\propto & \frac{\beta_h^{\alpha_h}t^{\alpha_h-1}e^{-t\beta_h}}{\Gamma(\alpha_h)}\cdot\frac{\prod_{i=1}^V\frac{\Gamma(C^h_{(-j)i}+\eta_i+C^h_{ji})}{\Gamma(C^h_{(-j)i}+\eta_i)}}{\frac{\Gamma(\sum_{i=1}^V C^h_{(-j)i}+\eta_i+C^h_{ji})}{\Gamma(\sum_{i=1}^V C^h_{(-j)i}+\eta_i)}}\\
& \cdot\frac{\prod_{i=1}^V\frac{\Gamma(L^h_{(-j)i}+\eta'_i+L^h_{ji})}{\Gamma(L^h_{(-j)i}+\eta'_i)}}{\frac{\Gamma(\sum_{i=1}^V L^h_{(-j)i}+\eta'_i+L^h_{ji})}{\Gamma(\sum_{i=1}^V L^h_{(-j)i}+\eta'_i)}}\cdot p(z_j=h|\boldsymbol{z}_{-j},\boldsymbol{\tau}),
\end{aligned}$$

$$p(z_j = h|\boldsymbol{z}_{-j}, \boldsymbol{\tau}) = \begin{cases} \frac{z^h_{(-j)}}{D-1+\tau} & z^h_{(-j)} > 0 \\ \frac{\tau}{D-1+\tau} & h = K^+_{-j} > 0 \\ 0 & \text{otherwise} \end{cases}$$

To sample $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, we derive the conditional probability as:

$$p(\alpha_k, \beta_k|\boldsymbol{t}, \hat{p}, \hat{q}, \hat{r}, \hat{s}, \boldsymbol{z}) \propto \frac{\tilde{p}_k^{\alpha_k - 1} e^{-\beta_k \tilde{q}_k}}{\Gamma(\alpha_k)^{\tilde{r}_k} \beta_k^{-\alpha_k \tilde{s}_k}},$$

$$\tilde{p}_k = \hat{p} P_k \qquad \tilde{q}_k = \hat{q} + S_k$$
$$\tilde{r}_k = \hat{r} + n_k \qquad \tilde{s}_k = \hat{s} + n_k$$
$$S_k = \sum_{j:z_j=k} t_j \qquad P_k = \prod_{j:z_j=k} t_j$$
$$n_k = \sum_{j=1}^{D} \mathbb{I}_{z_j=k}$$

where $\mathbb{I}$ is an indicator and n_k is the number of posts within the k-th episode. With the formula of the conditional probability, the Collapsed Gibbs Sampling algorithm is summarized in Algorithm 1 where M denotes the number of iterations.

Algorithm 1 Gibbs Sampler of the Nonparametric Model

INPUT: hyper-prior parameters $\{\boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\eta}', \hat{p}, \hat{q}, \hat{r}, \hat{s}\}$
OUTPUT: model parameters $\{\boldsymbol{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$

1: Initialize model parameters $\{\boldsymbol{z}^{(0)}, \boldsymbol{\alpha}^{(0)}, \boldsymbol{\beta}^{(0)}\}$
2: **for** $m = 1, ..., M$, **do**
3: **for** $j = 1, ..., D$, **do**
4: Sample z_j for document j, according to:

$$z_j^{(m)} \sim p(z_j = h|\boldsymbol{z}^{(m)}_{<j}, \boldsymbol{z}^{(m-1)}_{>j} \boldsymbol{C}, \boldsymbol{L}, \boldsymbol{t}, \boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\eta}', \boldsymbol{\alpha}, \boldsymbol{\beta})$$

5: **end for**
6: **for** $k = 1, \dots, K$ **do**
7: Using Metroplis-Hastings algorithm [Chib and Greenberg, 1995], sample α_k, β_k according to

$$\alpha_k^{(m)}, \beta_k^{(m)} \sim p(\alpha_k, \beta_k|\boldsymbol{t}, \hat{p}, \hat{q}, \hat{r}, \hat{s}, \boldsymbol{z}^{(m)})$$

8: **end for**
9: **end for**

Fast Burn-in Strategy of Gibbs Sampling: To obtain good approximation of posterior distributions via Gibbs Sampling, we usually need burn-in with a large number of iterations, which is computationally expensive. To speed up the model inference for the framework, we propose a fast burn-in strategy for Gibbs Sampling via setting a good starting point based on temporal bursts. In other words, we analyze the time-series to detect major bursts, and assume that the posts around the same burst are from the same timeline episodes. To be specific, we model each rise and fall pattern with a kernel function, model each time-series as a mixture of kernel functions, and then propose an approach to differentiate true bursts from jagged noisy peaks.

Given a time-series of an entity, we model each rise and fall pattern with a kernel model as: $\boldsymbol{g}(t; \boldsymbol{w}, \boldsymbol{\Gamma}, \mu) = \sum_{\ell=1}^{b} w_\ell \hat{g}(t; \boldsymbol{\gamma}_\ell, \mu)$, where $\hat{g}(t; \boldsymbol{\gamma}, \mu)$ is the basis function of the kernel, μ is the location of a pattern, $\boldsymbol{w}$ is the weight vector for a pattern, and $\boldsymbol{\gamma}_\ell$ is the parameters for ℓ-th basis function.

As spiking time-series usually exhibit sharp rise and slow decay, we make use of this domain knowledge in modeling. Specifically, we employ Gamma function as a basis function:

$$\hat{g}(t; \boldsymbol{\gamma}, \mu) = \begin{cases} Z^{-1}(t-\mu)^{\alpha-1} e^{-\beta(t-\mu)} & (t \geq \mu) \\ 0 & \text{(Others)} \end{cases}$$

where $\boldsymbol{\gamma} = [\alpha, \beta]$ is the predefined model parameters and Z is the normalization factor. Since each time-series tends to contain multiple up and down patterns, we further consider the mixture of kernel models as: $f(t; \boldsymbol{W}, \boldsymbol{\Gamma}, \boldsymbol{\mu}) = \sum_{p=1}^{P} \sum_{\ell=1}^{b} w_{p,\ell} \hat{g}(t; \boldsymbol{\gamma}_\ell, \mu_p)$, where P is the number of rise and fall patterns, μ_k is the location of k-th pattern.

Let us denote a time-series as $\boldsymbol{y} = [y_1, \dots, y_T]^\top$, where T is the length of time-series. To avoid the *non-convexity* problem, we first fix the model parameters of basis functions $\boldsymbol{\Gamma}$ by using prior knowledge of rise and fall patterns and represent each time-series by T mixtures ($T > P$): $f(t; \boldsymbol{W}) = \sum_{p=1}^{T} \sum_{\ell=1}^{b} w_{p,\ell} \hat{g}(t; \boldsymbol{\gamma}_\ell, p)$. Since we place each rise and fall pattern model at every time t, they are highly overlapped with each other over time. However, a time-series with multiple spikes tends to be sparse, that is, only a few $\boldsymbol{w}$ parameters should be non-zero. To this end, we employ the group sparsity constraint for fitting. Then, the final optimization problem can be written as:

$$\min_{\boldsymbol{v}} \quad \sum_{t=1}^{T} (y_t - \sum_{p=1}^{T} \boldsymbol{w}_p^\top \hat{g}(t; \boldsymbol{\Gamma}, p))^2 + \lambda \sum_{p=1}^{T} \|\boldsymbol{w}_p\|_2$$
$$\text{s.t.} \quad w_{p,\ell} \geq 0, \ \ell = 1, 2, \dots, b, \ p = 1, 2, \dots, T,$$

where $\sum_{p=1}^{T} \|\boldsymbol{w}_p\|_2$ is the group regularizer and λ is the regularization parameter. The group regularizer consists of L_2-regularizer for $\boldsymbol{w}$ and L_1 regularizer between groups $\|\boldsymbol{w}_1\|_2, \|\boldsymbol{w}_2\|_2, \dots, \|\boldsymbol{w}_T\|_2$. That is, the estimated parameter $\boldsymbol{w}$ tends to be dense within the group, and only a few groups (i.e., $\boldsymbol{w}$) take non-zero values.

In our proposed Timeline-Sumy framework, we first fit a time-series $\boldsymbol{y}$ using the group lasso based mixture model estimation, compute the magnitude of each estimated group lasso parameter $\boldsymbol{w}$, and select top K-1 group lasso parameters by ranking the magnitude $\boldsymbol{w}$, i.e., $[\|\widehat{\boldsymbol{w}}_1\|_2, \dots, \|\widehat{\boldsymbol{w}}_T\|_2]$, with burst labels 1...K-1. Then, we assign the burst labels to every post with the same time stamp, and assign label K to the remaining posts as the background episode, which is finally used as the initialization of Timeline-Sumy.

With the detected temporal bursts, our Gibbs Sampling algorithm starts from a good initialization, which leads to much faster convergence compared to traditional burn-in process and significantly speeds up the model inference for the proposed framework Timeline-Sumy. For example, our empirical results show that the proposed fast burn-in strategy often converges within 20 iterations compared to more than 500 iterations with traditional burn-in process.

2.2 Summary Ranking

Traditional text-based summarization algorithms are either abstraction-based or extraction-based approaches [Radev *et al.*, 2004; Jones, 2007]. The episode summarizer in Timeline-Sumy is an extraction-based approach. In other words, in the summary ranking phase, Timeline-Sumy aims to select the most representative social media posts as the summary of each timeline episode via ranking. In particular, we capture various types of signals via feature extraction for each social media post, and then utilize learning-to-rank approach [Liu, 2009] to rank these social media posts for each episode. According to different signals, we extract three types of features for each post, i.e., temporal-based, text-based and popularity-based features.

Temporal-based Feature: Intuitively, a representative social media post is issued when its timestamp is exactly match or very close to the temporal peak within the timeline episode. Therefore temporal signals provide valuable information for ranking posts for summarization. In particular, we propose the temporal-based feature of each social post as the temporal gap between its timestamp and the local temporal peak.

Text-based Features: The centroid based method is one of the most effective and robust algorithms for text-based summarization. For each social media post d, we represent it as a TFIDF vector $\vec{d}$, and then compute cosine similarity with the centroid vector $\vec{c}$ that is computed as $\vec{c} = \frac{\sum_{d \in E} \vec{d}}{|E|}$, where E is the set of social media posts in the episode. In addition, we extract another two text-based features: language detection feature based on the existing API[1], which indicates whether the social media post is in English or not; and textual length feature that is the length of the social media post.

Popularity-based Features: Popularity provides very important signals for timeline summarization. For those social media posts with higher popularity, it is likely that they are representative and of high quality, although popularity is not equivalent to quality. We consider four types of signals from tweet popularity: number of replies, number of retweets, number of 'likes', and author popularity (i.e. number of followers for a given tweet's author). As these popularity features are highly skewed, we normalize them with corresponding z-score, $z_i = \frac{x_i - \mu}{\sigma}$, where μ is the mean of the vector $\vec{x} = [x_1, ...]$ and σ denotes its standard deviation.

Learning-to-Rank Algorithm: Given aforementioned three types of heterogeneous features, as well as highly imbalanced training data in reality – very limited social media posts are labeled as positive, while most of the rest are labeled as negative for summary – we train Gradient Boosted Decision Tree (GBDT) algorithm [Friedman, 2001] to rank all candidate social media posts, and select the highest ranked ones as the summary of each timeline episode.

3 Experiments

To demonstrate the effectiveness of the proposed framework Timeline-Sumy, we conduct two groups of experiments. In the first group, we compare Timeline-Sumy with several state-of-the-art timeline summarization algorithms on social media datasets with ground truth. In the second group, we leverage Timeline-Sumy to generate summary from a dataset without labels to illustrate how Timeline-Sumy can help users effectively access and digest useful information from massive social media data about an entity over a period of time.

3.1 Timeline Summarization on Labeled Datasets

Since there are no benchmark datasets for the studied task, we manually label 4 social media datasets for evaluation. We collect 684.9k social media posts about *Andy Murray*, 20.8k posts about *David Ferrer*, 72.9k posts about *Maria Sharapova*, and 336.9k posts about *Roger Federer* from June 22 to August 7, 2012, which overlaps with two premium tennis tournaments: Wimbledon Open Tournament and London Olympics Tournament. We manually label each timeline episode according to each tennis star's major sports activities which are reflected on the corresponding time-series, and finally generate 13, 10, 11, 13 timeline episodes for these four datasets respectively. In each dataset, all social media posts not belonging to any timeline episode are labeled as the background episode. For each timeline episode, we manually label 1 representative social media post as the episode summary. We perform standard preprocessing steps on the datasets such as removing all stop-words and filtering low frequent terms and hashtag labels. Furthermore, the granularity of time-series is per hour, and our data from June 22 to August 7 can be represented as 1128-dimension time-series. We use ROUGE-2 and ROUGE-L [Lin, 2004] as the metrics to assess the quality of timeline summarization.

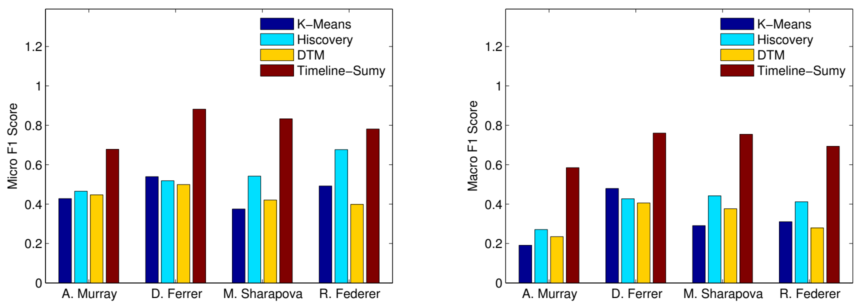
To evaluate timeline summarization, we compare our proposed Timeline-Sumy framework with the following state-of-the-art summarization algorithms: 1.) *LexRank* is a widely used traditionaltext-based summarization algorithm, and it builds a sentence to sentence graph and uses the centrality to select sentences [Erkan and Radev, 2004]; 2.) *ETS* is a timeline summarization for news corpus, and it is a graph based approach with optimized global and local biased summarization [Yan *et al.*, 2011]; 3.) *TPM* is a timeline summarization algorithm based on Twitter data, and it infers dynamic probabilistic distributions over interests and topics for tweets summarization [Ren *et al.*, 2013]. Note that these three baseline approaches are not supervised learning approaches.

One advantage of the episode detecting in the proposed framework is that we can reuse traditional methods to identify key episodes. We choose several representative episode detection methods to replace the nonparametric model in Timeline-Sumy, while re-using the learning-to-rank (LTR) based summary ranking module: 1.) *K-Means + LTR* uses *K-Means* to detect episodes based on content information, while completely ignoring temporal information; 2.) *Hiscovery + LTR* chooses *Hiscovery* [Li *et al.*, 2005] to identify episodes that models content with multinomial distributions and captures temporal information with a mixture of Gaussians. 3.) *DTM + LTR* adopts Dynamic Topical Model [Blei and Lafferty, 2006] to find episodes, and DTM is a state-of-the-art algorithm to model the topical evolving over time. Note that we use DTM to detect episodes by treating each topic as a timeline episode. For these supervised learning approaches, we train learning-to-rank model via cross-validation, i.e. training

[1] https://code.google.com/p/language-detection/

	A. Murray dataset		*D. Ferrer* dataset		*M. Sharapova* dataset		*R. Federer* dataset	
	ROUGE-2	ROUGE-L	ROUGE-2	ROUGE-L	ROUGE-2	ROUGE-L	ROUGE-2	ROUGE-L
LexRank†¶	0.07453	0.19753	0.08421	0.20942	0.15541	0.35570	0.11865	0.26967
ETS†¶	0.09765	0.30739	0.08738	0.26087	0.03580	0.15776	0.10909	0.29779
TMP†¶	**0.24242**	0.44295	0.10614	0.26111	0.12618	0.27586	0.13953	0.29730
LTR‡¶	0.17578	0.36576	0.30612	0.43655	0.23245	0.40964	0.12620	0.34286
K-Means+LTR‡¶	0.12062	0.34496	0.32376	0.44156	0.29146	0.43000	0.28906	0.43580
Hiscovery+LTR‡¶	0.13044	0.40157	0.34626	0.48329	0.44501	0.54453	0.25433	0.44529
DTM+LTR‡¶	0.18255	0.42829	0.37811	0.50990	0.32161	0.50000	0.15663	0.35200
Timeline-Sumy‡§	0.22594	**0.45000**	**0.45078**	**0.59278**	**0.49869**	**0.62663**	**0.35590**	**0.53180**

Table 1: Timeline Summarization Comparison. († indicates an unsupervised learning approach, while ‡ denotes a supervised learning approach; § indicates that the algorithm can learn the number of timeline episodes K automatically, while ¶ means that the algorithm needs to predefine K.)



(a) Micro F1 Comparison

(b) Macro F1 Comparison

Figure 1: Micro F1 and Macro F1 Comparison for Timeline Episode Detection.

on 3 labeled datasets and testing on the remaining dataset.

We demonstrate timeline summarization results in Table 1. Note that *LTR* in the table is a variant of the proposed framework by removing the phase of episode detecting (or *LTR* considers all social media posts as one episode and performs summery ranking to generate timeline summary). We make the following observations: 1.) supervised learning approaches that use label information in general outperform unsupervised learning approaches; 2.) TPM performs best among unsupervised approaches; 3.) removing the phase of episode detecting, the performance of *LTR* degrades a lot, which suggests the necessity of episode detecting for the proposed framework; 4.) Using traditional episode detection methods, the performance of *K-Means* + *LTR*, *Hiscovery* + *LTR* and *DTM* + *LTR* reduces compared to Timeline-Sumy that can further indicate the critical role of timeline episode detecting in timeline summarization.

To deeply understand the reason why Timeline-Sumy outperforms *K-Means* + *LTR*, *Hiscovery* + *LTR* and *DTM* + *LTR*, we further compare the quality of detected timeline episodes by their corresponding episode detection algorithms. We use micro F1 score and macro F1 score as the metrics to evaluate the performance of timeline episode detection. Since episode detection baselines cannot determine the number of episodes automatically, we assume that the number of timeline episodes, K, is known in advance for a fair comparison. We illustrate timeline episode detection results in Figure 1. We observe that the proposed nonparametric generative model often obtains the best performance since it uses the life cycle models to capture unique temporal properties of timeline episodes and it captures content and temporal information simultaneously.

3.2 Timeline Summarization on a Unlabeled Dataset

In this subsection, we utilize Timeline-Sumy to generate summary for an unlabeled dataset as a case study to illustrate how the proposed framework can help users to answer the question as we asked at the beginning of the paper, i.e., "What's the gossips of Jennifer Lopez in 2014 summer?" We conduct this group of experiments on one unlabeled dataset about *Jennifer Lopez*. We collect 601.9k social media posts about *Jennifer Lopez* from June 1 to July 31, 2014. Similar to above labeled datasets, we perform the same preprocesses on this unlabeled dataset.

The summary generated by Timeline-Sumy is shown in Table 2 – 9 timeline episodes from the episode detecting phase with their representative social media posts from the summary ranking phase. In addition to the episode of performing on World Cup 2014 Opening Ceremony, we also identify several key episodes about Lopez such as gossips between Lopez and other celebrities, and Lopez's show and concert. As Jennifer Lopez is a celebrity with many gossips, we don't have any clues about which gossip would attract more attentions.

Date	Timeline Episode	Representative Social Media Post of the Episode
June 5	Lopez's concert in NYC	*Jennifer Lopez Brings 15 Years of Hits Home to The Bronx*
June 10	Lopez on Jimmy Fallon's Show	*The return of "Tight Pants" with Jennifer Lopez #FallonTonight*
June 12	Lopez on 2014 WorldCup Opening Ceremony	*Jennifer Lopez, Pitbull and Claudia Leitte @FIFAWorldCup 2014 Opening Ceremony*
June 16	Gossip between Lopez and Casper Smart	*Casper Smart – I Did Not Cheat on Jennifer Lopez!*
June 25	Gossip between Lopez and Kim Kardashian	*Jennifer Lopez asegura que el trasero de Kim Kardashian no esta de Nah!!*
June 30	Gossip between Lopez and Maksim Chmerkovskiy	*Jennifer Lopez Addresses Maksim Chmerkovskiy Dating Rumors*
July 13	Comparing Lopez with Shakira at WorldCup Closing Ceremony	*Feel proud shakira will be wearing a @CharbelZoe outfit at tonight's ceremony after jennifer lopez at the opening ceremony*
July 18	Indonesian talking about Lopez	*Begini Penampakan Jennifer Lopez Tetap Cantik Tanpa Makeup*
July 24	Lopez's Birthday	*Happy Birthday Jennifer Lopez*

Table 2: Identified Timeline Episodes and their Summaries on the *Jennifer Lopez* dataset.

With the help of Timeline-Sumy, we can effectively discover a chain of timeline episodes about Jennifer Lopez, and automatically summarize the main theme of each episode via its representative post.

4 Related Work

The task of timeline generation or timeline summarization have been mainly studied on news scenario [Hu *et al.*, 2011; Yan *et al.*, 2011] and recently extended to social media [Ren *et al.*, 2013; Chang *et al.*, 2013; Li and Cardie, 2014]. In [Du *et al.*, 2015], both temporal information and content information are leveraged via using a Dirichlet-Hawkes process for event clustering, however, it is assumed the temporal gap between events are known in advance, which doesn't fit our scenario. In addition, timelines are constructed via jointly optimizing both temporal constraints and event constraints in [Do *et al.*, 2012]; a Dirichlet Process model is proposed to mine personal timelines [Li and Cardie, 2014]; and timelines are generated via a time-aware hierarchical Bayesian model together with a learning-to-rank model in [Ge *et al.*, 2015]. However, none of aforementioned frameworks explicitly model the temporal patterns of timeline events.

Several approaches have been proposed to model temporal patterns of online content such as Hawkes process [Matsubara *et al.*, 2012], power law distribution functions [Crane and Sornette, 2008], infinite-state automation approach [Kleinberg, 2003], and Life Cycle model [Chang *et al.*, 2014]. In addition, time-series can be effectively modeled after aligned or shifted such as Dynamic Time Warping (DTW) [Rakthanmanon *et al.*, 2011], clustering time-series aligned by shifting and scaling operations [Yang and Leskovec, 2011]. However, these approaches totally overlook content information.

Futhermore, our timeline episode detection method is inspired by existing work on social media event detection, and majority of related work on this research topic are about new event detection [Sayyadi *et al.*, 2009; Weng and Lee, 2011; Sakaki *et al.*, 2010] or retrospective event detection [Diao and Jiang, 2014]. In addition, dynamic topic models and its variations can be leveraged for topic tracking [Blei and Lafferty, 2006; Wang *et al.*, 2008]. However, most of aforementioned methods cannot handle dramatic event shifting and do not explicitly model life cycle patterns.

5 Conclusion

In this paper, we study the problem of timeline summarization, which is to generate summary for a chain of episodes in a timeline about an entity from social media data. We introduce life cycle models to capture unique temporal patterns include life-cycle patterns and sudden-spike-and-heavy-tail patterns for timeline episodes, and propose a novel timeline summarization framework Timeline-Sumy with an episode detecting phase and a summary ranking phase. In the episode detecting phase, we propose a novel Bayesian nonparametric model which captures regular content, hashtag content and temporal information simultaneously. Gibbs sampling is employed to infer the model parameters, and a fast burn-in strategy based on temporal bursts is further introduced to speed up the model inference. In the summary ranking phase, we introduce a learning-to-rank based approach which is flexible to integrate various types of signals from social media data for timeline summarization. The experimental results demonstrate the effectiveness of Timeline-Sumy. In our current work we only consider each individual entity, and we will extend Timeline-Sumy for multiple correlated entities in our future work.

Acknowledgments

This research is supported by the NSF research grants IIS-1134990, IIS 1254206 and U.S. Defense Advanced Research Projects Agency (DARPA) under Social Media in Strategic Communication (SMISC) program, Agreement Number W911NF-12-1-0034. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agency, or the U.S. Government. We also acknowledge the research grant MEXT KAKENHI #16K16114 from Japan.

References

[Blei and Lafferty, 2006] David Blei and John Lafferty. Dynamic topic models. In *Proceedings of ICML*, 2006.

[Chang *et al.*, 2013] Y. Chang, X. Wang, Q. Mei, and Y. Liu. Towards twitter context summarization with user influence models. In *Preceedings of WSDM*, 2013.

[Chang *et al.*, 2014] Y. Chang, M. Yamada, A. Ortega, and Y. Liu. Ups and downs in buzzes: Life cycle modeling for temporal pattern discovery. In *Preceedings of ICDM*, 2014.

[Chib and Greenberg, 1995] Siddhartha Chib and Edward Greenberg. Understanding the metropolischastings algorithm. *American Statistician*, 49(4), 1995.

[Crane and Sornette, 2008] Riley Crane and Didier Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences*, 2008.

[Diao and Jiang, 2014] Qiming Diao and Jing Jiang. Recurrent chinese restaurant process with a duration-based discount for event identification from twitter. In *Preceedings of SIAM International Conference on Data Mining*, 2014.

[Do *et al.*, 2012] Quang Xuan Do, Wei Lu, and Dan Roth. Joint inference for event timeline construction. In *Preceedings of EMNLP-CoNLL*, 2012.

[Du *et al.*, 2015] Nan Du, Mehrdad Farajtabar, Amr Ahmed, Alexander Smola, and Le Song. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *Preceedings of KDD*, 2015.

[Erkan and Radev, 2004] Gunes Erkan and Dragomir R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research (JAIR)*, 22:457–479, 2004.

[Friedman, 2001] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

[Ge *et al.*, 2015] T. Ge, W. Pei, H. Ji, S. Li, B. Chang, and Z. Sui. Bring you to the past: Automatic generation of topically relevant event chronicles. In *Preceedings of ACL*, 2015.

[Hu *et al.*, 2011] Po Hu, Minlie Huang, Peng Xu, Weichang Li, Adam K. Usadi, and Xiaoyan Zhu. Generating breakpoint-based timeline overview for news topic retrospection. In *Preceedings of ICDM*, 2011.

[Jones, 2007] Karen Sparck Jones. Automatic summarising: The state of the art. In *Information Processing and Management*, pages 1449–1481, 2007.

[Kleinberg, 2003] Jon Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

[Kwak *et al.*, 2010] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue B. Moon. What is twitter, a social network or a news media. In *Proceedings of WWW*, 2010.

[Levitt, 1965] Theodore Levitt. Exploit the product life cycle. *Harvard Business Review*, 1, November 1965.

[Li and Cardie, 2014] Jiwei Li and Claire Cardie. Timeline generation: Tracking individuals on twitter. In *Preceedings of WWW*, 2014.

[Li *et al.*, 2005] Zhiwei Li, Bin Wang, Mingjing Li, and WeiYing Ma. A probabilistic model for retrospective news event detection. In *Proceedings of SIGIR*, 2005.

[Lin, 2004] Chin-Yew Lin. Rouge: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS)*, 2004.

[Liu, 2009] Tie-Yan Liu. *Learning to Rank for Information Retrieval*. Foundation and Trends on Information Retrieval. Now Publishers, 2009.

[Matsubara *et al.*, 2012] Yasuko Matsubara, Yasushi Sakurai, B. Aditya Prakash, Lei Li, and Christos Faloutsos. Rise and fall patterns of information diffusion: Model and implications. In *Proceedings of KDD*, 2012.

[Radev *et al.*, 2004] Dragomir R. Radev, Hongyan Jing, Magorzata Sty, and Daniel Tam. Centroid-based summarization of multiple documents. *Information Processing and Management*, 40(6):919–938, 2004.

[Rakthanmanon *et al.*, 2011] Thanawin Rakthanmanon, Eamonn J. Keogh, Stefano Lonardi, and Scott Evans. Time series epenthesis: Clustering time series streams requires ignoring some data. In *Proceedings of ICDM*, 2011.

[Ren *et al.*, 2013] Zhaochun Ren, Shangsong Liang, Edgar Meij, and Maarten de Rijke. Personalized time-aware tweets summarization. In *Preceedings of SIGIR*, 2013.

[Romero *et al.*, 2011] D. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of information diffusion across topics: Idioms, political hashtags, and complex contagion on twitter. In *Preceedings of WWW*, 2011.

[Sakaki *et al.*, 2010] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of WWW*, 2010.

[Sayyadi *et al.*, 2009] H Sayyadi, M Hurst, and A Maykov. Event detection in twitter. In *Proceedings of ICWSM*, 2009.

[Wang *et al.*, 2008] Chong Wang, David Blei, and David Heckerman. Continuous time dynamic topic models. In *Proceedings of ICML*, 2008.

[Weng and Lee, 2011] J. Weng and B. Lee. Event detection in twitter. In *Proceedings of ICWSM*, 2011.

[Yan *et al.*, 2011] R. Yan, X. Wan, J. Otterbacher, L. Kong, X. Li, and Y. Zhang. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *Preceedings of SIGIR*, 2011.

[Yang and Leskovec, 2011] Jaewon Yang and Jure Leskovec. Patterns of temporal variation in online media. In *Proceedings of WSDM*, 2011.