LMPP: A Large Margin Point Process Combining Reinforcement and Competition for Modeling Hashtag Popularity

Bidisha Samanta, Abir De, Abhijnan Chakraborty, Niloy Ganguly
Indian Institute of Technology Kharagpur, India
bidisha@iitkgp.ac.in, {abir.de,abhijnan.chakraborty,niloy}@cse.iitkgp.ernet.in

Abstract

Predicting the popularity dynamics of Twitter hashtags has a broad spectrum of applications. Existing works have primarily focused on modeling the popularity of individual tweets rather than the underlying hashtags. As a result, they fail to consider several realistic factors contributing to hashtag popularity. In this paper, we propose Large Margin Point Process (LMPP), a probabilistic framework that integrates hashtag-tweet influence and hashtag-hashtag competitions, the two factors which play important roles in hashtag propagation. Furthermore, while considering the hashtag competitions, LMPP looks into the variations of popularity rankings of the competing hashtags across time. Extensive experiments on seven real datasets demonstrate that LMPP outperforms existing popularity prediction approaches by a significant margin. Additionally, LMPP can accurately predict the relative rankings of competing hashtags, offering additional advantage over the state-of-the-art baselines.

1 Introduction

To improve the efficacy of Twitter, prediction of hashtag flow (e.g., early detection of trending hashtags [Mathioudakis and Koudas, 2010]) is important. Thus, a large amount of research on viral marketing and information cascades [Cheng *et al.*, 2016; Chakraborty *et al.*, 2016; Sikdar *et al.*, 2016] have focused on analyzing hashtag-popularity and its role in tweet propagation.

Prior works and their limitations: Research on predictive aspects of hashtag-popularity primarily follows two kinds of models: (i) Static feature-based models ([Rosenfeld *et al.*, 2016; Bourigault *et al.*, 2014; Shi *et al.*, 2016] and the citations therein), and (ii) Temporal models ([Zhao *et al.*, 2015; Kobayashi and Lambiotte, 2016; Bi and Cho, 2016; De *et al.*, 2016a; Iwata *et al.*, 2013; De *et al.*, 2014; Kupavskii *et al.*, 2012; Hua-Wei *et al.*, 2014; Shuai and Jun, 2015; Gao *et al.*, 2016; Ferraz Costa *et al.*, 2015; Bao *et al.*, 2015; Gomez-Rodriguez *et al.*, 2011; De *et al.*, 2016b]). In the static models, the temporal properties (e.g., time of posts, no. of retweets) are embedded into feature maps, and the parameters are learned following a supervised approach. However, in practice, future temporal properties are not known in advance, which in turn constrains their forecasting prowess. The temporal models aim to address this shortcoming by modeling the stochastic nature of hashtag-dynamics, using point-process, survival theory, etc. However, most existing temporal models focus on tweet-propagation rather than hashtags, thereby skirting several realistic aspects of hashtag-flow, and resulting in modest prediction performance. More importantly, they are largely unable to reproduce any microscopic feature in hashtag dynamics (e.g., relative popularity variation, sudden trend change etc.).

In general, the popularity of a hashtag depends on two primary factors: (i) hashtag-tweet reinforcement, and (ii) inter-hashtag competitions. Every hashtag has an intrinsic attractiveness, and similarly the tweets bearing the hashtag also have their own appeal. In our earlier work [Samanta *et al.*, 2017], we showed that these two factors often reinforce each other. For example, an unpopular hashtag sometimes becomes viral due to the presence of some popular tweets. Similarly, a not-so-popular tweet may become popular later on due to the popularity of the hashtag it is bearing. Since traditional temporal approaches bank on modeling only tweet-propagation, simply extending these prediction-frameworks to hashtags, would not produce accurate results (our experimental results also emphatically establish that).

However, considering only the hashtag-tweet reinforcement process still leaves a paucity in the realistic modeling of hashtag-flow, that demands a careful consideration of the inter-hashtag competitions. In many cases, a newly popular hashtag may abbreviate the visibility of other popular hashtags. For example, often a breaking-news hails new hashtags which quickly get popularized and in the process, at least for a short span of time, suppress the popularity of other consistently popular hashtags, i.e., in practice, two hashtags often compete, rather than reinforcing each other's dissemination. Some recent works [Valera and Gomez-Rodriguez, 2015; Myers and Leskovec, 2012] attempted to model competitions in different contexts; however, they rely heavily on feature-engineering and underlying networks [De *et al.*, 2013], making them inefficient in online learning.

Present work and road-map: In this paper, we develop **L**arge **M**argin **P**oint **P**rocess (LMPP), a novel probabilistic framework that models the dynamics of hashtag popularity by unifying the above two factors in a principled way. The inherent hashtag-popularity often reinforces the virality of its constituent tweet-chains and vice-versa, which LMPP aims to capture using a generalized triggering kernel. In this process, LMPP considers the hashtag-tweet reinforcement factor to vary widely across the popularity distribution of tweets.

Furthermore, LMPP aims to incorporate competition among the hashtags and its impact on hashtag dynamics. In order to do that, we probe into the variations in the popularity rankings of the concurrent and related hashtags. Therefore, to capture such signals, we suitably curate the parameter space of LMPP that ensures correct ordering of popularity across several time intervals. Such a formulation intuitively articulates the competition process, without drastically changing the model-setting. In fact, this additional trait helps to properly train the model, which in turn enables it to detect sudden drifts in popularity rankings of the competing hashtags.

Our contributions: Summarizing, in this paper, we develop a novel stochastic framework for modeling hashtag dissemination, that unifies the role of hashtag-tweet reinforcement process and inter-hashtag competition in a principled way. LMPP can also be seen as an instance of a *large-margin estimation* where the constraints utilize the popularity differences among competing hashtags, thereby enabling the model to maximize the associated popularity margins along with the corresponding likelihood. Such a setting helps the model to estimate ranks on the fly, a crucial practical challenge that was left unaddressed in the literature. On seven real-world datasets crawled from Twitter, LMPP offers substantial accuracy gains in predicting popularity of hashtags, beyond strong baselines. By accurately considering the competition process, it can successfully model the ranking dynamics over time, of the correlated hashtags, which none of the existing baselines can even consistently trace. Consequently, it can reasonably forecast the abrupt popularity changes of hashtags which, in general, is considered a difficult phenomenon to reproduce.

2 Proposed Model

2.1 Overview

Terminology: We define a **tweet-chain** as the set consisting of a tweet and all its retweet instances. In this work, we represent a tweet by a unique id, and a tweet-chain in terms of the posting times of the tweet and all her retweets. Formally, a tweet-chain C_i, corresponding to tweet i, can be written as $C_i = \{t_j | \text{tweet } i \text{ is (re)tweeted at time } t_j\}$. Similarly, a hashtag $\boldsymbol{H}$ can be expressed in terms of the tweet-chains, $\boldsymbol{H} = \{C_i | C_i \text{ is a tweet-chain bearing } \boldsymbol{H}\}$. Finally, the *history* of the hashtag $\boldsymbol{H}$ until and excluding time t, $\mathcal{H}_{\boldsymbol{H}}(t)$ can be represented as the union of the posting times of the corresponding (re)tweets posted before t.

$$\mathcal{H}_{\boldsymbol{H}}(t) = \cup_{C_i \in \boldsymbol{H}} \{t_j | t_j \in C_i \text{ and } t_j < t\}$$

Computation of hashtag popularity: In a similar spirit to [Zhao *et al.*, 2015], we measure the popularity of a hashtag by the total number of its constituent tweet-posts. To do so, we propose LMPP, that models the temporal dynamics of the constituent tweets of a hashtag in terms of post-rate. While modeling such a post-rate, LMPP combines the role of hashtag-tweet reinforcement and hashtag competitions in the overall popularity dynamics.

Basic generative process for tweet-chains: At the outset, we represent the posting times of the (re)tweets as a point-process model. In particular, given a hashtag $\boldsymbol{H}$, we define the counting variable as $N_{\boldsymbol{H}}(t)$, where $N_{\boldsymbol{H}}(t) \in \{0\} \cup \mathbb{Z}^+$ counts the number of (re)tweets posted until and excluding time t. Then, we characterize the conditional probability of observing an event in infinitesimal time interval $[t, t+dt)]$ as

$$\mathbb{P}(\text{An event triggers in } [t, t+dt) | \mathcal{H}_{\boldsymbol{H}}(t)) = \lambda_{\boldsymbol{H}}(t)\, dt \quad (1)$$

$$\text{i.e., } \mathbb{E}_{dN_{\boldsymbol{H}}(t) \sim \{0,1\}}[dN_{\boldsymbol{H}}(t) | \mathcal{H}_{\boldsymbol{H}}(t)] = \lambda_{\boldsymbol{H}}(t) dt \quad (2)$$

Here $dN_{\boldsymbol{H}}(t)$ indicates the number of (re)tweets in the infinitesimal time-window $[t, t+dt)$ and $\lambda_{\boldsymbol{H}}(t)$ stands for the associated hashtag intensities, which further depends on the history $\mathcal{H}_{\boldsymbol{H}}(t)$. The functional form of $\lambda_{\boldsymbol{H}}(t)$ is chosen to capture the phenomenon of interests that possibly encompass hashtag-competitions, self-exciting dynamics or hashtag-tweet interactions. In the following, we present a specific characterization of $\lambda_{\boldsymbol{H}}(t)$ that captures the self-exciting nature of hashtag dynamics.

Self-exciting dynamics: To capture the mutual excitation between (re)tweet posting events, we rely on *Hawkes process* [Farajtabar *et al.*, 2014; De *et al.*, 2016b]. It is a particular type of functional form used in the growing literature on social activity modeling using point processes [Farajtabar *et al.*, 2014; Valera and Gomez-Rodriguez, 2015]:

$$\begin{aligned}\lambda_{\boldsymbol{H}}(t) =& \lambda_{\boldsymbol{H},0} + \beta \sum_{t_i \in \mathcal{H}_{\boldsymbol{H}}(t)} e^{-\omega_0 (t - t_i)} \\ =& \lambda_{\boldsymbol{H},0} + \beta(\kappa(t) \star dN_{\boldsymbol{H}}(t)) \quad (3)\end{aligned}$$

Here, $\lambda_{\boldsymbol{H},0} \geqslant 0$, models the initial post-rate of (re)tweets, and the second term, with $\beta \geqslant 0$, assigns weight to the influence of the publication of earlier (re)tweets. $\kappa(t) = e^{-\omega_0 t}$ is an exponential triggering kernel indicating the decay of influence of the past events over time, and $\star$ denotes the convolution operation.

2.2 LMPP: Modeling Hashtag-tweet Reinforcement and Inter-hashtag Competitions

Apart from the inherent popularity dynamics of the tweet-chains, our proposed framework LMPP considers two more crucial factors in modeling the popularity dynamics of a hashtag: (i) the mutual reinforcement process between hashtags and tweets, and (ii) the competitions among hashtags.

We observe that, it is the intensity kernel $\kappa(t)$ (in Eq (3)) that accounts for the self-exciting mechanism for the Hawkes process. Therefore, we aim to construct a suitable $\kappa(t)$ which should, along with the self-exciting reinforcement process of the individual tweets, capture the hashtag-tweet reinforcement factor. Note that, the influence of a hashtag is only exposed through the tweets, and *the effect of the hashtag on a tweet is more enunciated via a popular tweet than a rare tweet*. Therefore, the hashtag-tweet reinforcement factor should vary across the popularity distribution of the tweets. Thus, the kernel should be further parameterized by *tweet-popularity index* k (defined in Section 2.3), to have $\kappa_k(t)$. Particularly, $\kappa_k(t)$ should be chosen in such way that:

• Given a hashtag, when a popular tweet is retweeted i.e., when k goes high, the inherent attractiveness of the hashtag heavily influences its propagation process. That is, $\kappa_k(t)$ pushes $\lambda_{\boldsymbol{H}}(t)$ more towards a Hawkes process. Therefore, the overall resulting dynamics become more and more bursty.

• For non-popular tweets, the effect of the resultant influence of the hashtag is very low on its propagation process. Thus, a low value of $\kappa_k(t)$ should fare in a relatively small $\lambda_{\boldsymbol{H}}(t)$.

Considering the above points, we take $\kappa_k(t)$ as

$$\kappa_k(t) = \kappa_\infty(t) e^{-\frac{\omega t}{k}} \tag{4}$$

$\kappa_k(t)$ has two factors. $\kappa_\infty(t)$ indicates the self exciting process of the tweets, while $e^{-\frac{\omega t}{k}}$ stands for the hashtag-tweet influence. Note that, the impact of a tweet on the hashtag grows high, as the popular tweets get posted and vice-versa. Furthermore, we try to approximate $\kappa_\infty(t)$ as a more generalized intensity kernel $\kappa_\infty(t) = \sum_{j=1}^{M} \beta_j e^{-\omega_j t}$ where M is a large integer.

Then, the arrival rate of tweets can be written as

$$\lambda_{\boldsymbol{H}}(t; \boldsymbol{k}(t)) = \lambda_{\boldsymbol{H},0} e^{-\epsilon t} + \sum_{j=1}^{M} \beta_{\boldsymbol{H}}^{j} \sum_{t_i \in \mathcal{H}_{\boldsymbol{H}}(t)} e^{-(\omega_j + \frac{\omega}{k(t_i)})(t - t_i)} \tag{5}$$

Here, $k(t_i)$ is the popularity of tweet posted at time t_i, and $\boldsymbol{k}(t) := \{k(t_i) | t_i \in \mathcal{H}_{\boldsymbol{H}}(t)\}$. For compactness, we denote $\boldsymbol{\omega} = [\omega_1, \omega_2, ...\omega_M]$. Here, an additional decay factor $e^{-\epsilon t}$ is incorporated to diminish the effect of the initial condition, which we found to work well in practice.

2.3 Popularity Distribution

We observe that given a hashtag, the distribution of the popularity indices of individual tweets follows a power-law (figures omitted for brevity), which means that the tweets getting very high re-tweets are very small in number, whereas plenty of tweets are having small number of retweets. The distribution is captured as below:

$$p(k) = ck^{-\alpha} \text{ with } c = \alpha - 1 \tag{6}$$

where k is the popularity of a tweet-chain.

Hence, the expected arrival rate of the process having tweet-chains with *random popularity* can be formulated as:

$$\tilde{\lambda}_{\boldsymbol{H}}(t) = \mathbb{E}_{\boldsymbol{k}}[\lambda_{\boldsymbol{H}}(t; \boldsymbol{k}(t))] = \int_1^\infty c \lambda_{\boldsymbol{H}}(t; \boldsymbol{k}(t)) k^{-\alpha} dk \tag{7}$$

where, c is a constant given by Eq. (6).

2.4 Popularity Ranking in Hashtag Competition

In a tweet-diffusion process, concurrent hashtags often compete with each other for user attention. Such scenarios are usually pronounced through the variations of popularity rankings of the competing hashtags over time. To model it, one may specify $\lambda_{\boldsymbol{H}}(t)$, so that, it detects the variation in their popularity rankings across time. In particular, we say

$$N_{\boldsymbol{H}_1}[t_s, t_f) > N_{\boldsymbol{H}_2}[t_s, t_f) \implies$$
$$\int_{t_s}^{t_f} \lambda_{\boldsymbol{H}_1}(t) dt \geq \int_{t_s}^{t_f} \lambda_{\boldsymbol{H}_2}(t) dt + 1 \; \forall \boldsymbol{H}_1, \boldsymbol{H}_2 \text{ and } t_s < t_f \tag{8}$$

where, $N_{\boldsymbol{H}}[t_s, t_f)$ denotes the number of (re)tweets of hashtag $\boldsymbol{H}$ posted in the interval $[t_s, t_f)$

Datasets	Duration	#Hashtags	#Tweets	Mean rank diversity
Oscars	Feb 24 to Feb 29, '16	15	20,536	0.60
MTV-Awards	Apr 3 to Apr 12, '16	20	7,897	0.75
Nepal-Earthquake	Apr 25 to May 1, '15	25	32,613	0.70
Dem-Primary	Feb to June, '16	15	21,746	0.72
BBD	Oct 6 to Oct 8, '14	20	67,399	0.51
Copa	June 3 to June 26, '16	15	12,853	0.63
T20WC	Mar 8 to Apr 3, '16	10	5,854	0.77

Table 1: Summary of the datasets.

2.5 Parameter Estimation

Given a set of N hashtags $\mathbb{H} = \{\boldsymbol{H}_l | 1 \leq l \leq N\}$, we record a collection of posts $\mathcal{H}_{\boldsymbol{H}_l}(T) = \{t_i\}$ for each hashtag $\boldsymbol{H}_l$ during a time period $[0, T)$. Using these posts, we attempt to find the optimal parameters $\lambda_{\boldsymbol{H},0}$ and $\boldsymbol{\beta_H} = [\beta_{\boldsymbol{H}}^1, \beta_{\boldsymbol{H}}^2, ..., \beta_{\boldsymbol{H}}^M]$ for each hashtag $\boldsymbol{H} \in \mathbb{H}$ by solving a maximum likelihood estimation (MLE) problem. To do so, it is easy to show that the resulting log-likelihood function is

$$\log[L(\boldsymbol{\lambda}_{\mathbb{H},0}, \boldsymbol{B} | \epsilon, \boldsymbol{\omega}, \omega)]$$
$$= \sum_{\boldsymbol{H} \in \mathbb{H}} \sum_{t_i \in \mathcal{H}_{\boldsymbol{H}}(T)} \log \lambda_{\boldsymbol{H}}(t_i) - \sum_{\boldsymbol{H} \in \mathbb{H}} \int_0^T \lambda_{\boldsymbol{H}}(t) dt \tag{9}$$

where, $\boldsymbol{\lambda}_{\mathbb{H},0} := [\lambda_{\boldsymbol{H}_1,0}, \lambda_{\boldsymbol{H}_2,0}, ..., \lambda_{\boldsymbol{H}_N,0}]$ and $\boldsymbol{B} \in \mathbb{R}^{N \times M}$ with $\boldsymbol{B}_{l,i} = \beta_{\boldsymbol{H}_l}^i$ are the variables to be estimated.

To incorporate the effect of competing hashtags, we further restrict $\lambda_{\boldsymbol{H}}(t)$ following Eq (8), by first splitting the interval $[0, T)$, into L sets of small, equal and disjoint subintervals $[0, T_s), [T_s, 2T_s), \ldots, [(L-1)T_s, T)$, where $T_s = T/L$, and then imposing the following constraints:

Whenever, $N_{\boldsymbol{H}}[iT_s, (i+1)T_s) \geq N_{\boldsymbol{H}'}[iT_s, (i+1)T_s)$,

$$\int_{iT_s}^{(i+1)T_s} (\lambda_{\boldsymbol{H}}(t) - \lambda_{\boldsymbol{H}'}(t)) dt \geq 1; \; \boldsymbol{H}, \boldsymbol{H}' \in \mathbb{H}, \; 0 \leq i \leq L-1$$

Similar to SVM [Weston, 2014], such a hard-margin approach often may lead to an infeasible solution. Therefore, we introduce slack variables,

$$\zeta_{\boldsymbol{H},\boldsymbol{H}'}^i = \max(0, 1 - y_{\boldsymbol{H},\boldsymbol{H}'}^i \int_{iT_s}^{(i+1)T_s} [\lambda_{\boldsymbol{H}}(t) - \lambda_{\boldsymbol{H}'}(t)] dt)$$

with

$$y_{\boldsymbol{H},\boldsymbol{H}'}^i = \text{sign}(N_{\boldsymbol{H}}[iT_s, (i+1)T_s) - N_{\boldsymbol{H}'}[iT_s, (i+1)T_s))$$

and cast the problem as

$$\max_{\boldsymbol{\lambda}_{\mathbb{H},0}, \boldsymbol{B}} \log[L(\boldsymbol{\lambda}_{\mathbb{H},0}, \boldsymbol{B} | \epsilon, \boldsymbol{\omega}, \omega)] - C \sum_{i=0}^{L-1} \sum_{\boldsymbol{H}, \boldsymbol{H}' \in \mathbb{H}} \zeta_{\boldsymbol{H},\boldsymbol{H}'}^i$$
$$\text{with, } y_{\boldsymbol{H},\boldsymbol{H}'}^i \int_{iT_s}^{(i+1)T_s} \Big(\lambda_{\boldsymbol{H}}(t) - \lambda_{\boldsymbol{H}'}(t)\Big) dt \geq 1 - \zeta_{\boldsymbol{H},\boldsymbol{H}'}^i \tag{10}$$
$$\forall \boldsymbol{H}, \boldsymbol{H}' \in \mathbb{H} \text{ and } 0 \leq i \leq L-1$$

Note that the above problem is convex and thus can be solved efficiently. We call this framework, ***Large-Margin** self-exciting **Point Process*** (LMPP), since it incorporates the variations in ranking by increasing the popularity-margins of competing hashtags while maximizing the corresponding log-likelihood.

2.6 Popularity Forecasting

Our goal here is to develop efficient methods that leverage our model to forecast a hashtag's popularity at a given time t. In the context of our model, we aim to compute $N^*_{\boldsymbol{H}}(t) = \mathbb{E}_{\mathcal{H}_{\boldsymbol{H}}(t)}[N_{\boldsymbol{H}}(t)]$, the expected value of total retweet counts of all tweets for a given hashtag $\boldsymbol{H}$.

$$N^*_{\boldsymbol{H}}(t) = \mathbb{E}_{\mathcal{H}_{\boldsymbol{H}}(t)}[N_{\boldsymbol{H}}(t)] = \mathbb{E}_{\mathcal{H}_{\boldsymbol{H}}(t)}\Big[\int_0^t \tilde{\lambda}_{\boldsymbol{H}}(t)dt\Big]$$

Theorem 1 *The expected popularity of a hashtag $\boldsymbol{H}$ at time t is given by,*

$$\mathbb{E}_{\mathcal{H}_{\boldsymbol{H}}(t)}\Big[\int_0^t \tilde{\lambda}_{\boldsymbol{H}}(\tau)d\tau\Big] = \frac{1}{\epsilon}\boldsymbol{\lambda}_{\boldsymbol{H},0}(1-e^{-\epsilon t})$$
$$+ \boldsymbol{\lambda}_{\boldsymbol{H},0}\sum_{j=1}^{M}\int_0^t \Big[\sum_{k=0}^{\infty}\frac{(b^j_{\boldsymbol{H}}t^{(2-\alpha)})^k}{\Gamma((2-\alpha)k)}e^{-\omega_j t}\Big] \star e^{-\epsilon t}dt \quad (11)$$

where $b^j_{\boldsymbol{H}} = \frac{\omega_j^{(1-\alpha)}\beta^j_{\boldsymbol{H}}(\alpha-1)\pi}{\sin((\alpha-1)\pi)}$.

Proof From Eq. (7), we have

$$\tilde{\lambda}_{\boldsymbol{H}}(t) = \lambda_{\boldsymbol{H},0}e^{-\epsilon t}$$
$$+ c\sum_{j=1}^{M}\beta^j_{\boldsymbol{H}}\int_1^\infty\int_0^t e^{-(\omega_j+\frac{\omega}{k})(t-\theta)}k^{-\alpha}dN_{\boldsymbol{H}}(\theta)dk \quad (12)$$

A trite calculation reduces Eq. (12) to,

$$\tilde{\lambda}_{\boldsymbol{H}}(t) = \lambda_{\boldsymbol{H},0}e^{-\epsilon t} +$$
$$\sum_{j=1}^{M}a^j_{\boldsymbol{H}}\int_0^t e^{-\omega_j(t-\theta)}(\omega(t-\theta))^{1-\alpha}dN_{\boldsymbol{H}}(\theta) \quad (13)$$

where $a^j_{\boldsymbol{H}} = c\beta^j_{\boldsymbol{H}}\Gamma(\alpha-1) = \beta^j_{\boldsymbol{H}}\Gamma(\alpha)$.
Since $\mathbb{E}_{dN_{\boldsymbol{H}}(\theta)\sim\{0,1\}}[dN_{\boldsymbol{H}}(\theta)|\mathcal{H}_{\boldsymbol{H}}(\theta)] = \lambda_{\boldsymbol{H}}(\theta)d\theta$, by taking the Laplace transform of Eq (13) and then applying inverse Laplace transform, we obtain

$$\tilde{\lambda}_{\boldsymbol{H}}(t) = \lambda_{\boldsymbol{H},0}e^{-\epsilon t} + \lambda_{\boldsymbol{H},0}\Big[\sum_{k=0}^{\infty}\sum_{j=1}^{M}\frac{(b^j_{\boldsymbol{H}}t^{(2-\alpha)})^k}{\Gamma((2-\alpha)k)}e^{-\omega_j t}\Big] \star e^{-\epsilon t}$$

On integrating the above, we obtain Eq. (11). ■

3 Experimental Evaluation

In this section, we first describe the datasets used, the evaluation protocol, a short description of baseline paradigms and then provide a detailed comparative performance analysis of our proposal and the baselines.

3.1 Datasets

To implement our proposal, we collected seven datasets associated with a diverse set of real events. The events are chosen in such a way that they provide significant number of messages. So, we focus on popular events from entertainment, sports, e-commerce and disaster. We used Twitter search API to collect all the tweets (corresponding to a 2-3 weeks period around the event date) of the following events/topics, also summarized in Table 1. (i) *The Academy Awards 2016* (Oscars), (ii) *MTV Awards 2016* (MTV), (iii) *Earthquake in Nepal 2015* (Nepal-Earthquake), (iv) *Democratic Primaries for US Presidential Election 2016* (Dem-Primary), (v) *Big Billion Day sale of e-commerce site Flipkart 2014* (BBD), (vi) *Copa America Football Tournament 2016* (Copa), and (vii) *T20 Cricket World Cup 2016* (T20WC). We crawled $\sim$2 million tweets for each dataset. Thereafter, from each dataset, we carefully select the hashtags in such a way that, (i) there are a significant number of concurrent hashtags with large tweet count, (ii) the hashtags have variations in terms of the number of constituent tweet chains and (iii) the hashtags show notable deviations in the popularity ranking list across time. As an aggregated measure of such deviations, we define rank-diversity of a hashtag as the fraction of times its rank has changed. If out of total I time-windows, a hashtag $\boldsymbol{H} \in \mathbb{H}$ has changed her rank k times, then rank-diversity($\boldsymbol{H}$)=k/I. The mean rank-diversity of a dataset is thereby obtained by averaging the rank-diversity of all hashtags in the corresponding dataset (shown in Table 1).

3.2 Evaluation Protocol

Training and testing: Given a stream of temporal data $\mathcal{H}_{\boldsymbol{H}}$ for hashtags $\boldsymbol{H} \in \mathbb{H}$, we first split it into training and test set where training comprises of the first 80% of the total number of messages ($|\mathcal{H}|$). We use this 80% messages as input to train our model for estimating the parameters. Here, to construct the constraints (Eq 10), we divide the training-time in several ten-hour intervals and compare the popularity of the competing hashtags in each of them. The estimated model is thereafter used to predict the popularity dynamics of the hashtags in the test set. In order to determine the predictive prowess of LMPP (and the baselines), we follow two different evaluation approaches.

(i) Forecasting hashtag popularity: In this approach, the predictor aims to forecast the hashtag popularity by computing the expected number of message-posts in the test set.

(ii) Rank prediction of competing hashtags: As mentioned earlier, hashtag competition usually exhibits variations in the popularity rankings of the corresponding hashtags. Here, we attempt to retrieve the popularity rankings of competing hashtags in the test-set, which in turn indicates how efficiently an algorithm captures the phenomenon of hashtag competition.

3.3 Evaluation Metrics

Mean Absolute Percentage Error (MAPE): It captures the mean deviation between the observed and the predicted popularity for a hahstag up to time t. It is defined by the formula

$$\text{MAPE}(\boldsymbol{H}) = \frac{1}{M_{\boldsymbol{H}}}\sum_{i=0}^{M_{\boldsymbol{H}}-1}\left|\frac{\hat{N}_{\boldsymbol{H}}(t_i) - N_{\boldsymbol{H}}(t_i)}{N_{\boldsymbol{H}}(t_i)}\right|.$$

Here $\hat{N}_{\boldsymbol{H}}(t)$ and $N_{\boldsymbol{H}}(t)$ are the estimated and actual number of retweets for a hashtag $\boldsymbol{H}$ respectively, at time t. $M_{\boldsymbol{H}}$ denotes the total number messages of hashtag $\boldsymbol{H}$ in the test-set. For a given dataset, we report MAPE as the average of all MAPE($\boldsymbol{H}$) over the hashtags $\boldsymbol{H}$ of that dataset.

Spearman's Rank Correlation Coefficient (SRCC): The Spearman's Rank Correlation Coefficient between predicted rank-list $\hat{R}_{\mathbb{H}}$, and actual rank-list $R_{\mathbb{H}}$ of a hashtag set $\mathbb{H}$ can be defined as,

$$\rho(\hat{R}_{\mathbb{H}}, R_{\mathbb{H}}) = \frac{\mathbf{Cov}(\hat{R}_{\mathbb{H}}, R_{\mathbb{H}})}{\sqrt{\mathbf{Var}(\hat{R}_{\mathbb{H}})\mathbf{Var}(R_{\mathbb{H}})}}.$$

Datasets	MAPE(%) LMPP	HTR	RPP	Hawkes	SEISMIC	SpikeM	SRCC LMPP	HTR	RPP	Hawkes	SEISMIC	SpikeM
Oscars	**18.90** (1.7%)	21.78	24.30	*19.23*	24.79	27.11	**0.85**	0.68	*0.80*	0.52	0.10	0.74
MTV-Awards	**05.14** (23.7%)	06.76	15.37	*13.57*	19.09	24.45	**0.87**	0.86	0.81	0.75	0.70	0.82
Nepal-Earthquake	**07.50** (12.2%)	08.54	22.28	15.42	13.73	17.95	**0.91**	0.87	0.63	0.32	0.63	0.75
Dem-Primary	**08.33** (20.7%)	10.50	*11.33*	11.62	26.09	19.12	**0.86**	0.68	*0.80*	0.51	0.10	0.73
BBD	**15.40** (3.4%)	17.94	19.09	*15.94*	18.03	20.89	**0.95**	0.79	0.90	*0.91*	0.43	0.79
Copa	**17.67** (0.4%)	20.07	*17.75*	19.18	23.32	22.44	**0.91**	0.42	0.75	*0.88*	0.42	0.64
T20WC	**10.25** (13.9%)	11.90	*13.10*	15.08	25.55	41.74	**0.87**	0.58	*0.83*	0.31	0.56	0.47

Table 2: MAPE (%) and SRCC of proposed and baseline algorithms on all datasets with 20% held-out set. The cells with light orange (blue) color indicates the best (second best) predictor. Numbers in the bracket denote percentage improvement over the nearest baseline. Numbers in the italics indicate the best performer among the four state-of-the-art baselines.

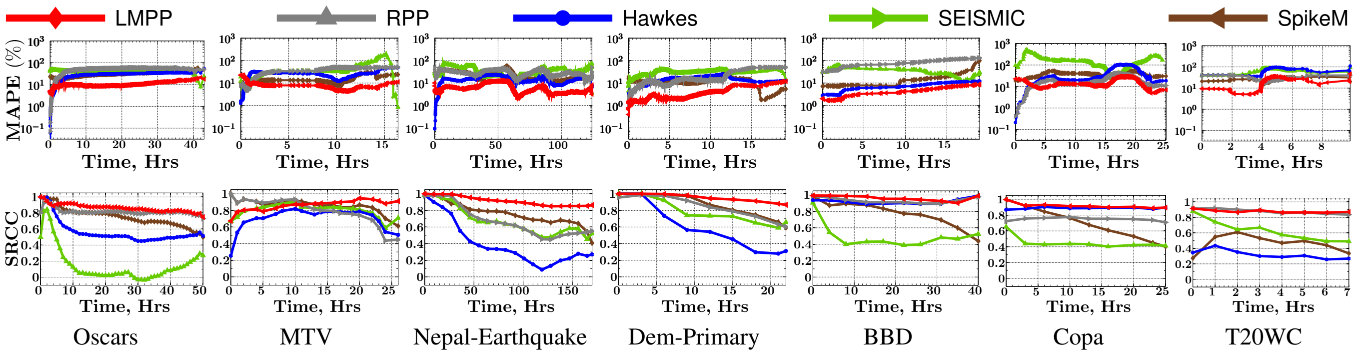

Figure 1: Variation of popularity forecasting performance with time using a 20% held-out set for each real-world dataset. Performance is measured in terms of MAPE (%)(top row) on popularity and SRCC (bottom row) between predicted and actual popularity rankings.

Here **Cov**(.) defines covariance of the two variables, and **Var**(.) denotes the variance.

Avg. Recall (AvRe) and Avg. Precision (AvPr) (for jump detection): If in two consecutive time-intervals there is a sudden change in the rank of a hashtag by more than half the total no. of competing hashtags, we call it a *jump*. Let the set of hashtags be $\mathbb{H}$ and the rank of a hashtag $\boldsymbol{H} \in \mathbb{H}$ be $\text{rank}_{\boldsymbol{H},[t_i,t_{i+1})}$ at time-interval $[t_i, t_{i+1})$. Then, if $|\text{rank}_{\boldsymbol{H},[t_i,t_{i+1})} - \text{rank}_{\boldsymbol{H},[t_{i-1},t_i)}| \geq |\mathbb{H}|/2$, it is considered a jump. Recall measures the fraction of cases an algorithm rightly identifies a jump, while precision measures the fraction of time a real jump has occurred when the algorithm predicts a jump. Incidentally a jump is also referred to as *'Flash in the pan'* (as described in [Yang and Leskovec, 2011]).

3.4 Baselines

We compare LMPP with five strong baselines: (i) Reinforced Point Process (RPP) [Hua-Wei *et al.*, 2014], (ii) Simple Hawkes process [Bao *et al.*, 2015], (iii) SEISMIC [Zhao *et al.*, 2015], (iv) SpikeM [Matsubara *et al.*, 2015; Yang and Leskovec, 2011], and (v) HTR (**H**ashtag **T**weet **R**einforcement model). The baselines represent a diverse class of existing temporal models. For example, RPP considers intrinsic attractiveness followed by a decay in popularity, Hawkes and SEISMIC represent the self-exciting processes, whereas SpikeM is a temporal pattern based approach. RPP, Hawkes and SEISMIC were primarily proposed for single tweet popularity prediction, which we extended for hashtags by aggregating over the popularity of the corresponding tweets. Finally, HTR only considers hashtag-tweet reinforcement, but not the inter-hashtag competition. Comparing HTR with LMPP helps us to understand the role of hashtag competition in hashtag propagation.

3.5 Performance Comparison (MAPE)

Table 2 presents a comparative sketch in terms of MAPE on the 20% held-out set. We observe that for all datasets, LMPP performs best by achieving lowest MAPE compared to all other baselines. *The performance of SpikeM and SEISMIC are consistently poor* in most of the cases. This is because, SpikeM emphasizes on modeling realistic patterns from the temporal data (e.g., periodicity). However, the temporal patterns in training-set often do not match with that in the test-set. The performance of SEISMIC is better than SpikeM in most of the datasets. In contrast to SpikeM, SEISMIC does not rely on a fixed set of temporal patterns present in the data. Rather, a point-process based formulation helps it to properly capture the stochastic dynamics of the (re)tweet posts.

We find the performance of RPP and Hawkes to be better than SEISMIC and SpikeM. RPP attempts to capture intrinsic attractiveness of a tweet, and the aging of different posts. There are mainly two distinctive features that help Hawkes to obtain a significant performance-boost. First, Hawkes models the temporal effect of each and every post rather than their simple collective effect, which is very well suited to capture the bursty nature of the tweets. Second, the underlying learning problem is convex for this approach. As a result, one can accurately estimate the parameters, making it a robustly identifiable learning model. However, both RPP and Hawkes are designed to model the popularity dynamics of a single tweet. So, they do not consider the reinforcement of a hashtag and tweet chains. Moreover, ignoring inter-hashtag competition further limits their predictive power.

We observe that HTR performs better than RPP and Hawkes in four datasets. Since HTR only captures the hashtag-tweet reinforcement, it can reasonably record the effect of inherent attractiveness of the hashtags on the tweets.

	Avg. Precision						Avg. Recall					
Datasets	**LMPP**	**HTR**	**RPP**	**Hawkes**	**SEISMIC**	**SpikeM**	**LMPP**	**HTR**	**RPP**	**Hawkes**	**SEISMIC**	**SpikeM**
Oscars	**0.74**	0.54	0.32	*0.38*	0.31	0.33	**0.75**	0.45	0.32	*0.37*	0.33	0.34
MTV-Awards	**0.31**	0.30	0.30	0.31	0.31	0.30	**0.33**	0.32	0.33	*0.33*	0.30	0.32
Nepal-Earthquake	**0.61**	0.60	*0.37*	0.28	0.40	0.44	**0.70**	0.54	0.37	0.33	0.52	*0.57*
Dem-Primary	**0.69**	0.56	*0.48*	0.30	0.45	0.34	**0.72**	0.49	0.48	0.29	*0.57*	0.36
BBD	**0.66**	0.48	0.32	*0.55*	0.31	0.43	**0.67**	0.48	0.32	*0.64*	0.28	0.40
Copa	**0.72**	0.29	0.42	*0.60*	0.29	0.34	**0.59**	0.33	0.42	*0.53*	0.33	0.42
T20WC	**1.0**	0.32	0.64	0.10	0.29	*0.65*	**0.67**	0.32	*0.64*	0.10	0.21	0.54

Table 3: Average precision and recall in jump detection for all algorithms. Orange (blue) indicates best (second best) predictor. Numbers in italics indicate the best predictor from four baselines.

However, it does not incorporate hashtag competitions which in-turn constraints its performance.

The importance of hashtag competition is clearly reflected by the best performance of LMPP which properly blends both hashtag-tweet reinforcement and hashtag competition to accurately model the popularity dynamics. In particular, for four datasets viz. MTV-Awards, Nepal-Earthquake, Dem-Primary and T20WC, the performance boost of LMPP is substantial compared to its immediate competitor. This is because, for each of these datasets, the rank diversity (Table 1, last column) is very high. As a result, LMPP exploits this signal to train the model better than others, which results in a significant performance boost.

Analysis of forecasting capabilities with time: To have a better understanding of the prediction performance of all the models w.r.t. time, we further probed the timestamps in the 20% held-out test set and computed MAPE for each sample point. Figure 1 (top row) shows the change of predictive-performance of popularity with time (MAPE of sample points aggregated over time). We observe that the forecasting performance of LMPP is better than the other methods. As expected, as time progresses, the performance of all the approaches degrades - this rate is relatively much lower in LMPP than all other methods.

3.6 Performance Comparison (SRCC)

To evaluate the ability of the algorithms to detect variations in the popularity ranking of the competing hashtags, we divide the test time into several intervals of same duration. At each of them, we obtain a ranked list of the participating hashtags, based on the predicted popularity using LMPP as well as the baselines. On the other hand, from the original trace, we derive the ground-truth of these ranked lists. To measure how closely a predicted order matches the actual one, we compute Spearman's rank correlation coefficient (SRCC) between them. Table 2 and Figure 1 (bottom row) dissect a comparative analysis in terms of SRCC for all the proposals. We observe that LMPP performs significantly better than the baselines across all datasets. It is interesting to observe that, in this case, HTR outperforms the baselines only for two datasets. That is, here, HTR fares quite poorly as compared to its performance in terms of MAPE. Since HTR neglects the effect of hashtag competition, despite a strong forecasting power, it cannot accurately predict the relative variation of popularity among the competing hashtags. It is important to note that the performance of other baselines are poor and inconsistent across all the datasets. Although the baselines are specifically designed for popularity prediction, they are insensitive to ranking of hashtag popularity. That said, better MAPE does not guarantee that the relative popularity order will be maintained. This is because, MAPE counts absolute error, but does not reflect a relative popularity variation.

3.7 Detection of Sudden Changes in Ranking

Besides evaluating the accuracy of prediction of general rank order, the efficacy of a system can be measured in terms of its success in 'catching' the instances where a particular hashtag suddenly becomes popular (or suddenly loses popularity). We extracted such instances from the datasets (around 9% hashtags in the test-set), and checked how accurately the competing algorithms identify those instances. Table 3 reports the average precision and recall of jump detection for algorithms across all the datasets. In both precision and recall, LMPP significantly outperforms the other algorithms. Considering the hardness of the problem, we believe the performance of LMPP (both precision and recall are around 70%) is excellent by itself. The only exception being the *MTV-Awards* dataset, where the reason for the less impressive performance is that the absolute numbers of tweet-chains produced by different ranked hashtags are very close. Note that although in terms of SRCC, the gains of LMPP were modest over baselines, LMPP can correctly predict the abrupt changes much better. The problem with other baselines is not only their inferior performance but also their inconsistency, which results in no clear second ranker across datasets.

4 Conclusion

In this paper, we propose LMPP, a novel point process driven framework that unifies several realistic factors to model hashtag popularity such as hashtag-tweet reinforcement and inter-hashtag competitions. Such a unified approach does not only efficiently estimate hashtag popularity for which it is designed, but also gives an accurately prediction of the relative ranking of concurrent and competing hashtags. Extensive experiments over seven real-world datasets show that LMPP significantly outperforms state-of-the-art baselines in popularity prediction, additionally offering the accurate prediction of ranking and sudden change in popularity of the hashtags.

Acknowledgments

A. De and A. Chakraborty were supported by Google India PhD Fellowship. A. Chakraborty was also supported by Prime Minister's Fellowship Scheme for Doctoral Research.

References

[Bao *et al.*, 2015] Peng Bao, Hua-Wei Shen, Xiaolong Jin, and Xue-Qi Cheng. Modeling and predicting popularity dynamics of microblogs using self-excited hawkes processes. In *WWW*, 2015.

[Bi and Cho, 2016] Bin Bi and Junghoo Cho. Modeling a retweet network via an adaptive bayesian approach. In *WWW*, 2016.

[Bourigault *et al.*, 2014] Simon Bourigault, Cedric Lagnier, Sylvain Lamprier, Ludovic Denoyer, and Patrick Gallinari. Learning social network embeddings for predicting information diffusion. In *WSDM*, 2014.

[Chakraborty *et al.*, 2016] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. Stop clickbait: Detecting and preventing clickbaits in online news media. In *ASONAM*, 2016.

[Cheng *et al.*, 2016] Justin Cheng, Lada A Adamic, Jon Kleinberg, and Jure Leskovec. Do cascades recur? In *WWW*, 2016.

[De *et al.*, 2013] Abir De, Niloy Ganguly, and Soumen Chakrabarti. Discriminative link prediction using local links, node features and community structure. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1009–1018. IEEE, 2013.

[De *et al.*, 2014] Abir De, Sourangshu Bhattacharya, Parantapa Bhattacharya, Niloy Ganguly, and Soumen Chakrabarti. Learning a linear influence model from transient opinion dynamics. In *ACM CIKM*, 2014.

[De *et al.*, 2016a] Abir De, Sourangshu Bhattacharya, Sourav Sarkar, Niloy Ganguly, and Soumen Chakrabarti. Discriminative link prediction using local, community, and global signals. *TKDE*, 2016.

[De *et al.*, 2016b] Abir De, Isabel Valera, Niloy Ganguly, Sourangshu Bhattacharya, and Manuel Gomez Rodriguez. Learning and forecasting opinion dynamics in social networks. In *NIPS*, 2016.

[Farajtabar *et al.*, 2014] Mehrdad Farajtabar, Nan Du, Manuel Gomez-Rodriguez, Isabel Valera, Le Song, and Hongyuan Zha. Shaping social activity by incentivizing users. In *NIPS*, 2014.

[Ferraz Costa *et al.*, 2015] Alceu Ferraz Costa, Yuto Yamaguchi, Agma Juci Machado Traina, Caetano Traina Jr, and Christos Faloutsos. Rsc: Mining and modeling temporal activity in social media. In *KDD*, 2015.

[Gao *et al.*, 2016] Jinhua Gao, Huawei Shen, Shenghua Liu, and Xueqi Cheng. Modeling and predicting retweeting dynamics via a mixture process. In *WWW*, 2016.

[Gomez-Rodriguez *et al.*, 2011] Manuel Gomez-Rodriguez, David Balduzzi, and Bernhard Schölkopf. Uncovering the Temporal Dynamics of Diffusion Networks. In *ICML*, 2011.

[Hua-Wei *et al.*, 2014] Shen Hua-Wei, Dashun Wang, Chaoming Song, and Albert Barabasi. Modeling and predicting popularity dynamics via reinforced poisson processes. In *AAAI*, 2014.

[Iwata *et al.*, 2013] Tomoharu Iwata, Amar Shah, and Zoubin Ghahramani. Discovering latent influence in online social activities via shared cascade poisson processes. In *KDD*, 2013.

[Kobayashi and Lambiotte, 2016] Ryota Kobayashi and Renaud Lambiotte. Tideh: Time-dependent hawkes process for predicting retweet dynamics. In *AAAI*, 2016.

[Kupavskii *et al.*, 2012] Andrey Kupavskii, Liudmila Ostroumova, Alexey Umnov, Svyatoslav Usachev, Pavel Serdyukov, Gleb Gusev, and Andrey Kustarev. Prediction of retweet cascade size over time. In *CIKM*, 2012.

[Mathioudakis and Koudas, 2010] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *ACM SIGMOD*, 2010.

[Matsubara *et al.*, 2015] Yasuko Matsubara, Yasushi Sakurai, B. Aditya Prakash, Lei Li, and Christos Faloutsos. Rise and fall patterns of information diffusion: model and implications. In *WWW*, 2015.

[Myers and Leskovec, 2012] Seth A Myers and Jure Leskovec. Clash of the contagions: Cooperation and competition in information diffusion. In *ICDM*, 2012.

[Rosenfeld *et al.*, 2016] Nir Rosenfeld, Mor Nitzan, and Amir Globerson. Discriminative learning of infection models. In *WSDM*, 2016.

[Samanta *et al.*, 2017] Bidisha Samanta, Abir De, and Niloy Ganguly. Strm: A sister tweet reinforcement process for modeling hashtag popularity. In *IEEE INFOCOM*, 2017.

[Shi *et al.*, 2016] Bichen Shi, Georgiana Ifrim, and Neil Hurley. Learning-to-rank for real-time high-precision hashtag recommendation for streaming news. In *WWW*, 2016.

[Shuai and Jun, 2015] Gao Shuai and Chen Jun, Ma Zhumin. Modeling and predicting retweeting dynamics on microblogging platforms. In *WSDM*, 2015.

[Sikdar *et al.*, 2016] Sandipan Sikdar, Anshit Chaudhary, Shraman Kumar, Niloy Ganguly, Abhijnan Chakraborty, Gaurav Kumar, Abhijeet Patil, and Animesh Mukherjee. Identifying and characterizing sleeping beauties on youtube. In *CSCW Companion*, 2016.

[Valera and Gomez-Rodriguez, 2015] Isabel Valera and Manuel Gomez-Rodriguez. Modeling adoption and usage of competing products. In *ICDM*, 2015.

[Weston, 2014] Jason Weston. Support vector machine. 2014.

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

[Zhao *et al.*, 2015] Qingyuan Zhao, Murat A. Erdogdu, Hera Y. He, Anand Rajaraman, and Jure Leskovec. Seismic: A self-exciting point process model for predicting tweet popularity. In *KDD*, 2015.