Longitudinal Modeling of Social Media with Hawkes Process based on Users and Networks

P. K. Srijith
Computer Science and Engineering
IIT Hyderabad, India
srijith@iith.ac.in

Michal Lukasik
Kalina Bontcheva
Department of Computer Science
University of Sheffield, United Kingdom
m.lukasik@sheffield.ac.uk
k.bontcheva@sheffield.ac.uk

Trevor Cohn
Computing and Information Systems
University of Melbourne, Australia
t.cohn@unimelb.edu.au

***Abstract*—Online social networks provide a platform for sharing information at an unprecedented scale. Users generate information which propagates across the network resulting in information cascades. In this paper, we study the evolution of information cascades in Twitter using a point process model of user activity. We develop several Hawkes process models considering various properties including conversational structure, users' connections and general features of users including the textual information, and show how they are helpful in modeling the social network activity. We consider low-rank embeddings of users and user features, and learn the features helpful in identifying the influence and susceptibility of users. Evaluation on Twitter data sets associated with civil unrest shows that incorporating richer properties improves the performance in predicting future activity of users and memes.**

I. Introduction

Online social networks play a major role in dissemination of information. They provide main stream media with an opportunity to obtain real time information about events happening around the world. Understanding the evolution of events in social networks is important for a variety of reasons. For instance, it can help authorities prevent the spread of misinformation or rumors that are starting to disseminate through social networks.

In this paper, we consider problems of predicting which memes become popular in the future and which users will be most active. Tweets exhibit a highly complex temporal behaviour which is not easy to model. For instance, Figure 1 shows the evolution of different memes associated with the Ferguson unrest in 2014. Statistical models based on point processes have been proposed for modeling the temporal dynamics of Twitter [1]. In Twitter, a tweet can elicit further tweets, thus leading to a cascade. Such dynamics can be captured using self-exciting point processes such as Hawkes process (HP). Multivariate Hawkes process [2] models influence from other users in the network in generation of new tweets. Another stream of research on popularity prediction [3], [4], [5], [6] considered features such as tweet content, network and temporal descriptors, and applied simple regression models to predict tweet popularity. However, none of these works considered features in a point process framework which we argue provides a much better model of tweet occurrences. In this paper, we propose a model which elegantly combines social media features in a Hawkes process framework, resulting in models with excellent predictive performance.

We develop HP models which leverage various user features and network properties in Twitter. First, we develop a model over the conversation structure in Twitter, which we use to limit infectivity to individuals who react to other's posts. This results in a computationally efficient and scalable HP model. A second model considers the local network connectivity between the users to bias the learned user influence. Our third innovation is to incorporate rich user features in Twitter, which includes the profile information, individual's number of tweets, retweets and textual information in their tweets such as mentions, hashtags etc. We use these characteristics to parameterize user influences, based on a matrix decomposition technique inspired by the deep learning literature. This allows us to determine factors driving influence and susceptibility. We show that these additional information sources yield accuracy gains when applied to several collections of publicly available rumour datasets from Twitter [7]. The software implementing the HP models will be released upon publication.

The novelty of this paper lies in developing new Hawkes Process models: 1) considering conversation structure information, 2) considering network information, 3) integrating user features including textual information, and 4) learning user features determining influence and susceptibility.

II. Related Work

Information cascades in social networks are predominantly modeled using regression or classification models over extracted features. The features used include tweet content, network and temporal features [8], [3], [9], [4]. The problem of predicting the popularity of tweets is solved as a classification problem considering content, network and user information in [4]. In addition to user features, [5] also uses features determining flow of cascade and page rank to train a learning algorithm which can predict number of retweets. It is found in [6] that retweet dynamics are affected by content features such as hashtags and urls and contextual features such as the number of followers.

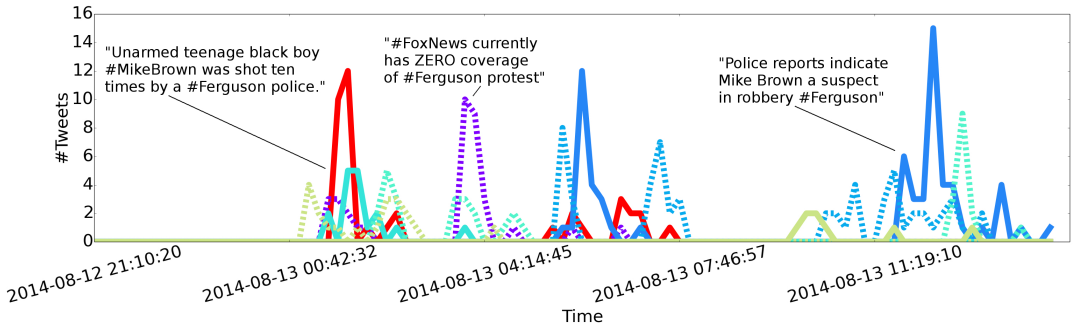

Fig. 1. Temporal profile of memes in the Ferguson data set.

There also exist approaches based on point processes to model the Twitter dynamics. However, most of such approaches are developed to infer the underlying influence network [10], [11], [12], [13], [2], [14], [15]. Seismic [1] is a recent model based on the self exciting point processes developed to predict the final number of reshares of a post. A networked Weibull regression model was proposed in [16] to predict the size of the cascades by considering behavioral dynamics. A Hawkes process approach which consider the circadian nature of users is developed in [17] to model the retweet dynamics of posts in Twitter. Our approach differs from the related work in that it incorporates user, content and network features into the Hawkes process framework to model Twitter dynamics. Thus, it conjoins two related approaches (point process based and feature based) into a unified framework to model Twitter dynamics.

III. Problem Formulation

We consider a set of N tweets, $D = \{d_1, \ldots, d_N\}$, belonging to M memes. Let R denote the total number of users in the Twitter data set. Each tweet contains information about their time of occurrence, meme topic and the user who generated the tweet. Thus, a tweet can be represented as a tuple $d_n = (t_n, m_n, i_n)$, where t_n is the time of tweet occurrence, m_n is the meme id and i_n is the id of a user who generated the tweet. Given the history of past occurrences of tweets, our goal is to predict future occurrences of tweets, future tweets from a particular user (i.e. user activity) and future tweets about a particular a meme (i.e. meme virality), considering conversation information (U), connection information ($\mathcal{F}$) and user features ($\boldsymbol{x}$) in a point process framework. Table I summarizes the notations.

TABLE I
Table of Symbols

Symbol	Description
(t_n, m_n, i_n)	User i_n, meme m_n and time t_n of n^{th} tweet
U	Conversation information
$\mathcal{F}$	Connection information
$\boldsymbol{x}_i$	Features of user i
$\lambda_{i,m}$	Intensity of user i for meme j
$\boldsymbol{\mu}$	User base intensities
$\boldsymbol{\gamma}$	Meme base intensities
$\boldsymbol{\alpha}$	Influence matrix of users
$\kappa(t)$	Exponential decaying function
N	Total number of tweets
R	Total number of users
M	Total number of memes

IV. Model

The longitudinal data arising in Twitter can be modeled using a point process over a continuous time period such as a Poisson process [18], [19]. It is characterized by rate parameter or intensity $\lambda > 0$ which provides the expected number of tweets happening in unit time. A homogeneous Poisson process (HPP) assumes the intensity to be constant (with respect to time), which makes them ill suited to modeling Twitter dynamics, which is much more irregular. Inhomogeneous Poisson process (IPP) [20] can model tweets occurring at a variable rate by considering the intensity to be a function of time, *i.e.* $\lambda(t)$.

Due to the social nature of Twitter, users tend to influence one another – a process which occurs alongside exposure to past tweets. A multi-variate Hawkes process defines an underlying non-negative intensity function which considers the influence of tweets from other users. The intensity function associated with a multivariate Hawkes process takes the following form:

$$\lambda_{i_n,m_n}(t) = \mu_{i_n}\gamma_{m_n} + \sum_{\ell=1}^{n-1} \mathbb{I}(m_\ell = m_n)\alpha_{i_\ell,i_n}\kappa(t-t_\ell) \quad (1)$$

where the first term represents the constant base intensity in which the tweets are generated by user i_n for meme m_n. Here, μ_{i_n} is the base intensity with which user i_n generates tweets and γ_{m_n} is the base intensity with which tweets are generated for meme m_n. The second term represents the influence from the tweets that happen prior to time of interest. The influence from each tweet decays over time and is modeled using an exponential decay function $\kappa(t-t_\ell) = \omega \exp(-\omega(t-t_\ell))$. The asymmetric matrix $\boldsymbol{\alpha}$ of size $R \times R$ encodes the latent degrees of influence between the pairs of users generating the tweets.

A. Considering Conversational Structure

The intensity function defined in (1) for a Hawkes process assumes all previous tweets influence the current tweet. However, many of the new tweets are generated in response

to some particular tweet in a conversation thread, and this conversation structure is observed in Twitter. Therefore, we modify the intensity function to take into account the tweet thread structure. In addition, the proposed model avoids the computational complexity of summing over all previous tweets for calculating the HP intensity function.

Adding conversational information leads to a Hawkes process (*HPconversation*) with intensity function

$$\begin{aligned}&\lambda_{i_n,m_n}(t) = \mu_{i_n}\gamma_{m_n}U_{n,n} \\ &+\textstyle\sum_{\ell=1}^{n-1}\mathbb{I}(m_\ell = m_n)U_{\ell,n}\alpha_{i_\ell,i_n}\kappa(t-t_\ell).\end{aligned} \tag{2}$$

where U represents an $N \times N$ binary matrix with $U_{\ell,n}$ containing information whether the tweet n was influenced by a previous tweet $\ell < n$ ($U_{\ell,n} = 1$) or is spontaneous ($U_{n,n} = 1$) [1]. Here, only one of the terms in (2) is active for a tweet: spontaneous tweets have only the base intensity (first term), and tweets replying to another tweet have only one of the second intensity term. The intensity function in (2) can also be written in a product form as

$$\begin{aligned}\lambda_{i_n,m_n}(t_n) &= \left[\mu_{i_n}\gamma_{m_n}\right]^{U_{n,n}} \\ &\times \textstyle\prod_{\ell=1}^{n-1}\left[\alpha_{i_\ell,i_n}\kappa(t_n-t_\ell)\right]^{\mathbb{I}(m_\ell=m_n)U_{\ell,n}},\end{aligned} \tag{3}$$

where only one term in the product is active for any tweet.

The parameters associated with the intensity function are learnt by maximizing the likelihood over observations. For a Hawkes process, the complete likelihood is given by

$$\prod_{n=1}^{N}\lambda_{i_n,m_n}(t_n)\times\exp\left(-\sum_{i=1}^{R}\sum_{m=1}^{M}\int_0^T\lambda_{i,m}(s)ds\right) \tag{4}$$

where T represents an upper bound on the considered time period. The parameters in the Hawkes process model are estimated by maximizing the log-likelihood.

$$\begin{aligned}&\arg\max_{\boldsymbol{\mu},\boldsymbol{\gamma},\boldsymbol{\alpha},\omega}\sum_{n=1}^{N}U_{n,n}\log(\mu_{i_n}\gamma_{m_n}) \\ &+\sum_{n=1}^{N}\sum_{\ell=1}^{n-1}\mathbb{I}(m_\ell=m_n)U_{\ell,n}\log(\alpha_{i_\ell,i_n}\kappa(t_n-t_\ell)) \\ &-T\sum_{i=1}^{R}\sum_{m=1}^{M}\mu_i\gamma_m-\sum_{i=1}^{R}\sum_{\ell=1}^{N}\alpha_{i_\ell,i}K(T-t_\ell)\,,\end{aligned} \tag{5}$$

where $K(T-t_\ell) = 1-\exp(-\omega(T-t_\ell))$ arises from the integration of $\kappa(t-t_\ell)$. A coordinate descent approach is followed to solve the optimization problem where each parameter is computed keeping all others fixed. The parameters $\boldsymbol{\mu}, \boldsymbol{\gamma}$ and $\boldsymbol{\alpha}$ have closed form updates, while the parameter ω is learned using gradient optimisation.

We also learn a variant of *HPconversation*, *HPregularization*, which performs ℓ_2 regularization of the influence matrix. It results in an additional regularization term, $-\sum_{i,j}\alpha_{i,j}^2$, in the log-likelihood which provides better generalization ability to the proposed HP model. In this case, $\boldsymbol{\alpha}$ is obtained in a closed form using the formula to find roots of a quadratic form.

[1]In the case of Twitter, this information is readily available from the 'in_reply_to_status_id' field of the JSON representation of tweets.

B. Network Information

Twitter also provides information about connectivity among the users. The user connectivity in Twitter is asymmetric in nature, *i.e.* user i may follow user j while user j may not follow user i. We can obtain 'who follows whom' information from Twitter and this can be very useful in modeling the way users influence each other. If user i follows user j, there is a higher chance that the tweets of user i are influenced by user j than by those published by an unconnected user k. We provide an approach, *HPconnection*, which considers this prior social connection information in a Hawkes process learning framework. *HPconnection* performs a selective ℓ_2 regularization of the $\boldsymbol{\alpha}$ matrix entries corresponding to the disconnected users. This leads to disconnected users in the network to have low values in $\boldsymbol{\alpha}$ matrix and hence low influence. The $\boldsymbol{\alpha}$ matrix is estimated by maximizing log-likelihood term and an additional regularization term, $-\sum_{i,j\notin\mathcal{F}}\alpha_{i,j}^2$, where $\mathcal{F}$ is the pairs of connected users in the follow graph. The regularized values of $\boldsymbol{\alpha}$ is obtained by solving for roots in a quadratic form.

C. Influence decomposition

An issue with the *HPconversation* model above is that the number of parameters in α is quadratic in number of users R. This results in over-fitting when modeling typical social media datasets which have large R and sparsely occurring tweets by each individual. We consider a HP model, *HPdecomposition*, to address this problem using a low-rank decomposition of the influence matrix. The advantage of the model is that it can learn the influence matrix with a lower number of parameters, and helps to avoid over-fitting to sparse data.

We assume the influence matrix $\boldsymbol{\alpha}$ can be decomposed into a product of two lower dimensional matrices $\boldsymbol{\alpha} = \mathbf{I}\boldsymbol{S}^\top$, with both matrices of rank $K \ll R$. We call $\mathbf{I}$ the influence embedding matrix and $\boldsymbol{S}$ the susceptibility embedding matrix, which give a lower dimensional latent representation of user features governing influence and susceptibility, respectively. Since each element of $\boldsymbol{\alpha}$ must be positive, we enforce all elements of $\mathbf{I}$ and $\boldsymbol{S}$ to be positive. This results in a Hawkes process (*HPdecomposition*) with intensity function of the following form:

$$\begin{aligned}&\lambda_{i_n,m_n}(t) = \mu_{i_n}\gamma_{m_n}U_{n,n}+ \\ &\textstyle\sum_{\ell=1}^{n-1}\mathbb{I}(m_\ell=m_n)U_{\ell,n}[\mathbf{I}\cdot\boldsymbol{S}^\top]_{i_\ell,i_n}\kappa(t-t_\ell).\end{aligned}$$

The latent factors are learnt by maximizing the log-likelihood of the data under this model. We use the *LBFGS-B* gradient based optimization method to learn $\mathbf{I}$ and $\boldsymbol{S}$, with parameters bounded to ensure non-negativity.

D. Incorporating User Features

We propose a modification of the *HPdecomposition* which considers features specific to social media users. They can play an important role in determining user impact on Twitter [21]. We consider user features listed in the first column of Table II in modeling the underlying influence network ($\boldsymbol{\alpha}$). This

TABLE II
USER FEATURES DETERMINING INFECTIVITY (INF.) AND SUSCEPTIBILITY (SUSC.), WITH LEARNED WEIGHTS SHOWN FOR THE FERGUSON AND OTTAWA TRAINED RANK 1 MODELS (HIGHEST MAGNITUDE IN BOLD.) FEATURES DESCRIBED IN [21].

	Ferguson		Ottawa	
Feature	susc.	inf.	susc.	inf.
# favourites	-0.59	1.51	0.08	0.03
# historical tweets	0.85	**1.52**	0.02	0.01
# unique mentions	4.40	-0.49	0.94	-0.67
geolocation	-0.34	-0.01	-0.11	0.82
hashtag-token ratio	0.12	0.15	0.02	0.06
links ratio	**-7.55**	-0.21	**-2.26**	**3.61**
profile background	-0.99	-0.03	-0.22	1.01
profile image	-0.43	0.20	-0.06	0.08
prop. hashtags	-3.72	0.54	-0.31	0.91
prop. reply tweets	-2.15	0.68	-0.48	2.76
prop. user-mentions	-4.90	0.39	-0.79	1.97

includes profile information as well as linguistic infomation extracted from historical tweet of users such as proportion of hashtags, mentions etc.

Let $\boldsymbol{x}_i$ represent the L dimensional features associated with the i^{th} user. We find an embedding of the user features in a lower dimensional space of size K. We assume that users have different embeddings representing their infectivity and susceptibility. The infectivity and the susceptibility embeddings are obtained by linearly transforming the user features through matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ (of sizes $L \times K$) respectively. The influence matrix $\boldsymbol{\alpha}$ is obtained using a non-linear transformation of these embeddings, with a sigmoid activation function which ensures non-negativity of $\boldsymbol{\alpha}$. The intensity function associated with a Hawkes process considering user features (*HPfeatures*) is:

$$\lambda_{i_n,m_n}(t) = \mu_{i_n}\gamma_{m_n}U_{n,n} + \sum_{\ell=1}^{n-1}\mathbb{I}(m_\ell = m_n)U_{\ell,n}\sigma\left([\boldsymbol{x}_{i_l}\boldsymbol{A}]\cdot[\boldsymbol{x}_{i_n}\boldsymbol{B}]^\top\right)\kappa(t-t_\ell)$$

where $\sigma(z) = \frac{1}{1+\exp(-z)}$ is the logistic sigmoid. Since we use the sigmoid to obtain a non-negative representation of the influence matrix, we do not have to restrict the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ to be non-negative. This helps to better model the correlations across the features in determining influence and susceptibility of the users. The parameters, including the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, are learnt by maximizing the log-likelihood.

The links ratio mentioned in Table II depends on the number of followers (ϕ_{in}), the number of followees (ϕ_{out}) and the number of times a user has been listed by others (ϕ_{lis}). It is calculated as $\log\left(\frac{(\phi_{lis}+1)(\phi_{in}+1)^2}{\phi_{out}+1}\right)$ [21].

V. PREDICTION ALGORITHM

We use the intensity function, with parameters learnt by maximizing the likelihood, to make predictions about future tweet occurrences in the test interval. The predictions are done by sampling points from the learnt intensity function using Ogata's modified thinning algorithm (see Algorithm 1) [22]. The algorithm samples a point from a Homogeneous Poisson Process (HPP) with a constant intensity β which is an upper bound to the conditional intensity $\lambda(t)$ in the interval of interest. For a HP with an exponentially decaying function, β can be easily obtained which is the intensity value at the preceding event. The point generated by the HPP is accepted as a point from a Hawkes process with probability $\frac{\lambda(t)}{\beta}$. Future points are generated by repeating the whole process with the intensity conditioned on the accepted point. During prediction, we consider the contribution from all the previous tweets in calculating the intensity function due to the lack of conversation structure information for test data points.

Algorithm 1: Ogata's thinning algorithm for sampling points from a Hawkes process

1: **Input:** conditional intensity function $\lambda(t)$, past tweet occurrence times t_1, t_2, t_n, time T
2: Initialize $t = t_n$, $S = \{\}$.
3: **while** $t \leq T$ **do**
4: Compute $\beta = \lambda(t)$.
5: Generate candidate next arrival time from $HPP(\beta)$, $s \sim \exp(\beta)$
6: Generate random number $U \sim Unif([0,1])$
7: **if** $(t + s > T)$ OR $(U > \frac{\lambda(t+s)}{\beta})$ **then**
8: Set $t = t + s$
9: **else**
10: Set $t = t + s$, $S = S \cup t$, Update Intensity $\lambda(t)$
11: **end if**
12: **end while**
13: **Return:** S

VI. EXPERIMENTS

A. Baselines

The proposed Hawkes process models are compared against the HP baseline, and previous state of the art approaches: HPmixed [2] and Seismic [1].

HPbase: The HP model considering a constant influence matrix, where all the values are set to 1. It helps us to check if learning the influence matrix helps in improving predictive performance within the HP framework.

HPmixed: The HP model [2] which could perform both temporal dynamics modeling and meme tracking. It infers the conversational structure instead of using the conversation information available in the data.

Seismic: The point process model developed to predict the resharing popularity of Twitter posts [1]. It takes into account the network information during prediction. However, it does not predict the exact times at which future posts will occur.

B. Evaluation

The models are trained on tweets up until a given point in time and then the learnt models are used to predict tweet occurence times in the future. Predictive performance of the models is evaluated alongside three different dimensions.

TABLE III
DATASET CHARACTERISTICS. DURATION SHOWN IN MINUTES.

Data	# Memes	# Users	# tweets	Duration
Synthetic	5	5	2739	16.8
Ferguson	31	1481	2190	8145
Ottawa	39	4096	6134	4140
Sydney	2	12607	24166	5704

1) User centric: ability to predict future tweet times of a user. 2) Meme centric: ability to predict future tweet times on a meme. 3) Joint: ability to predict overall future tweet times irrespective of memes and users. Predictions along each of these dimensions are obtained by considering the intensity function marginalized over memes or users or both memes and users.

We order the tweets based on their timestamps. The parameters of the model are learnt from the initial few percent of tweets and the rest are used to evaluate the predictive performance of the models. In these experiments, predictive performance is measured using aligned root mean squared error (ARMSE), which aligns the arrival times and calculates the root mean squared error (RMSE) between the two subsequences of predicted and actual tweet arrival times [18]. For the user centric and meme centric evaluations, ARMSE scores are calculated separately for each user and meme respectively and the average ARMSE is presented along standard deviation. We also consider the final count of predictions made by various models and compute mean absolute error (MAE) with respect to actual count of tweets.

We conduct experiments on both synthetic and real world datasets. The purpose of the synthetic experiments is to observe the model behaviour in controlled settings. We use three real world datasets [7] consisting of rumour memes from three major events[2]:

Ferguson relates to the unrest that took place in Ferguson, USA in August, 2014. The data set consists of tweets manually labeled by journalists as belonging to 31 different memes.

Ottawa relates to shootings at the parliament building in Ottawa during October 2014. It is structured similarly to the Ferguson dataset.

Sydney relates to the hostage taking in a Sydney cafe in December 2014, and is considerably larger than the other two datasets.

The properties of the datasets are shown in Table III. Note the sparsity of the data, with a very high ratio of users to tweets. Consequently, these datasets pose significant modeling challenges.

C. Synthetic Data Experiments

The synthetic dataset is generated by fixing the number of users, memes and the parameters associated with the intensity function. The number of users and memes in the synthetic data are both set to 5.[3] We assume a sparse influence matrix, with users 1 and 2 influencing user 3 and user 5 influencing user 1, and all other $\boldsymbol{\alpha}$ values set to zero.[4] We conduct experiments to verify the ability of the HP models to learn the influence matrix. The times of tweets are simulated using Ogata's thinning algorithm.

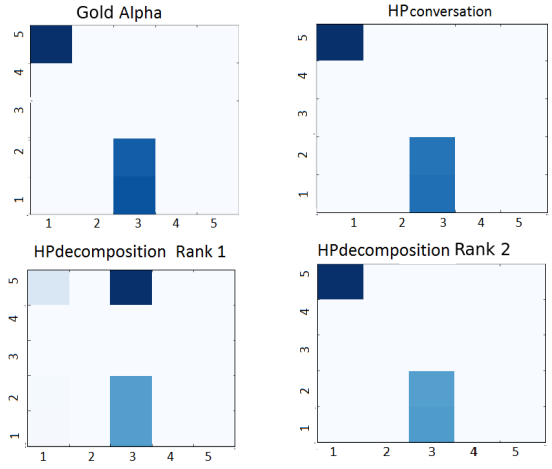

Fig. 2. Ground truth and learnt $\boldsymbol{\alpha}$ matrices learnt from 50% of synthetic dataset.

Figure 2 provides the heat map associated with the influence matrix learnt by the different HP models. The influence matrices are learnt by training on 50% of synthetic data. We can observe that the $\boldsymbol{\alpha}$ matrix learnt using *HPconversation* is close to the gold standard $\boldsymbol{\alpha}$ which generated the observations. Here, *HPdecomposition* with rank 2 learns a better $\boldsymbol{\alpha}$ matrix than *HPdecomposition* with rank 1 which is not sufficiently expressive.

We study the predictive performance of *HPconversation*, *HPdecomposition* and *HPregularization* by varying the fraction of training data on the synthetic data. Since *HPconnection* requires a social network and *HPfeatures* needs user features, we could not evaluate them on the synthetic data. In Figure 3, we provide the results on synthetic data for user, meme and joint evaluations using ARMSE score. As the fraction of training data increases, the ARMSE score decreases since the models can fit the parameters well. For smaller training fraction, *HPregularization* performed best with respect to all the evaluation dimensions. By regularizing the $\boldsymbol{\alpha}$ matrix, it avoids the possible over-fitting that arise due to limited data size. This is validated by the fact that *HPconversation* and *HPdecomposition* start achieving performance close to *HPregularization* when more training data become available. We could also observe that all the proposed approaches achieved a performance better than *HPbase* and the previous approach

[2]Available at http://tinyurl.com/z7kkso2

[3]The parameters of the intensity function used to generate the synthetic data are: $\boldsymbol{\mu} = [0.13, 0.13, 0.09, 0.14, 0.003]$ and $\boldsymbol{\gamma} = [0.97, 0.68, 0.92, 0.96, 0.64]$. These values were generated randomly from $\mathcal{U}([0, 1])$. The exponential decay parameter ω is fixed to 1.

[4]$\alpha_{1,3} = 0.45$, $\alpha_{2,3} = 0.42$ and $\alpha_{5,1} = 0.52$; see Gold Alpha in Figure 2.

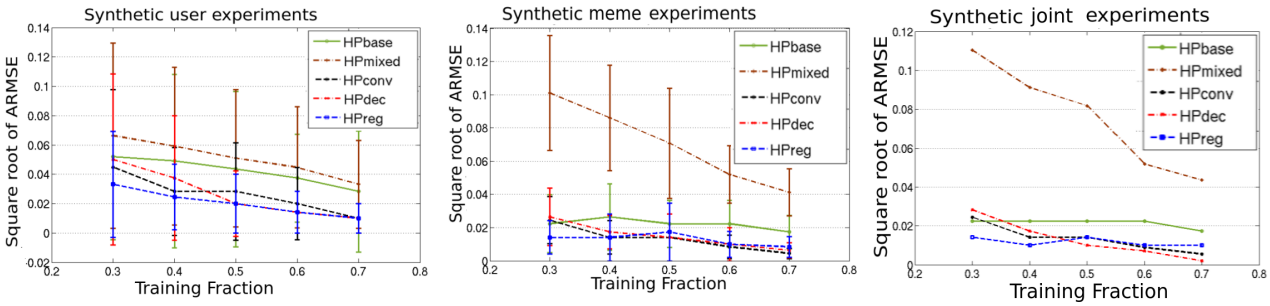

Fig. 3. Results on the Synthetic dataset based on the ARMSE of the occurrence times of predicted and actual tweets. We use square root of ARMSE to more clearly depict the variation in performance of all the different approaches.

TABLE IV
RESULTS ON THE SYNTHETIC DATASET, REPORTING MEAN ABSOLUTE ERROR ON THE COUNT OF PREDICTED AND ACTUAL TWEETS.

	30% training			50% training			70% training		
Approach	User	Meme	Joint	User	Meme	Joint	User	Meme	Joint
HPbase	159±50	1457±84	423	130±50	**988±68**	558	119±60	**550±44**	552
HPmixed	360±190	1881±10	1748	254±136	1346±8	1275	158±83	810±1	763
Seismic	7590±4239	9586±84	7931	9714±164	9714±61	8573	9834±97	9834±32	9178
HPconversation	119±212	1432±89	574	76±114	1030±61	289	42±51	638±36	137
HPdecomposition(1)	145±213	**1416±91**	605	74±117	1046±66	**180**	43±56	630±30	**72**
HPregularization	**86±105**	1592±63	**266**	**55±54**	1151±30	238	**37±36**	680±27	164

HPmixed. Note that *Seismic* does not predict exact times and hence does not yield to the ARMSE evaluation.

We compare the performance of the approaches with respect to mean absolute error (MAE) on the final count of predictions in Table IV. Again *HPregularization* gives best performance with 30% training data. *HPconversation* and *HPdecomposition* start to catch up with *HPregularization* with increase in fraction of synthetic data available for training. *HPdecomposition* did not improve over *HPconversation* in the synthetic dataset due to the smaller number of users. The proposed HP approaches performed better than *HPbase*, *HPmixed* and *Seismic* with respect to user evaluation. Learning influence matrix seems to positively affect the predictions for individual users but has a negative effect on meme predictions. In the synthetic data set, *Seismic* made some occasional high predictions. To avoid unduly penalizing the method, the MAE measure is based on an upper bound of 10000 (which only affects *Seismic*).

D. Experiments on Real-world Datasets

Turning to our real-world datasets, we compare the accuracy of our proposed models on predicting future tweets.

a) Ferguson Shootings: First, we conduct experiments to study the predictive performance of *HPdecomposition* on varying the rank of the factors. Figure 4 plots how ARMSE measure varies with respect to the ranks on various evaluation measures when trained using 50% of the Ferguson data. We observe that lower rank factors give better predictive performance than higher rank factors. Here, lower rank decomposition improves performance as it results in learning

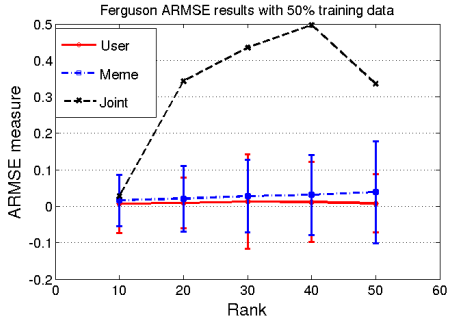

Fig. 4. Variation in predictive performance of *HPdecomposition* with respect to rank on the Ferguson dataset.

a smaller number of parameters from the data and prevents over-fitting.

Table V compares the ARMSE performance of the HP models on various fractions of training data. In all the fractions, the proposed models outperformed *HPP*, *HPbase* and *HPmixed* for user, meme and joint evaluations. Among the models trained with 30% Ferguson data, *HPdecomposition* and *HPregularization* performed better than other methods. These models could achieve better generalization performance by avoiding over-fitting through regularization and low rank decomposition. *HPconnection* is also useful in this scenario as they perform selective regularization to evade over-fitting. These models are particularly useful with sparse data and with a large number of users. As the fraction of training data increases, *HPconnection* starts improving its performance. It is found to be the best performing method with 70% of training data. *HPfeatures* gives a good performance with respect to user

TABLE V
RESULTS ON THE FERGUSON DATASET, REPORTING ARMSE ON OCCURRENCE TIMES OF PREDICTED AND ACTUAL TWEETS.

	30% training			50% training			70% training		
Approach	User	Meme	Joint	User	Meme	Joint	User	Meme	Joint
HPbase	100±441	199±428	1752	0.032±0.206	0.098±0.253	0.93	0.013±0.080	0.08±0.14	0.34
HPmixed	27±228	310±563	1715	0.004±0.050	0.184±0.439	0.33	0.003±0.033	0.04±0.08	0.15
HPconversation	20±177	259±388	978	0.009±0.050	0.027±0.098	0.44	0.005±0.042	0.01±0.04	0.06
HPdecomposition(1)	9±135	**101±232**	596	0.008±0.092	0.026±0.109	**0.29**	0.004±0.043	**0.01±0.03**	0.04
HPregularization	**9±90**	154±259	882	**0.001±0.001**	0.030±0.150	0.82	0.002±0.032	0.04±0.17	0.33
HPconnection	20±169	142±286	**102**	0.001±0.021	**0.023±0.080**	0.34	**0.001±0.016**	0.01±0.04	**0.01**
HPfeatures	15±151	165±413	322	0.001±0.024	0.056±0.220	0.73	0.002±0.032	0.04±0.08	0.10

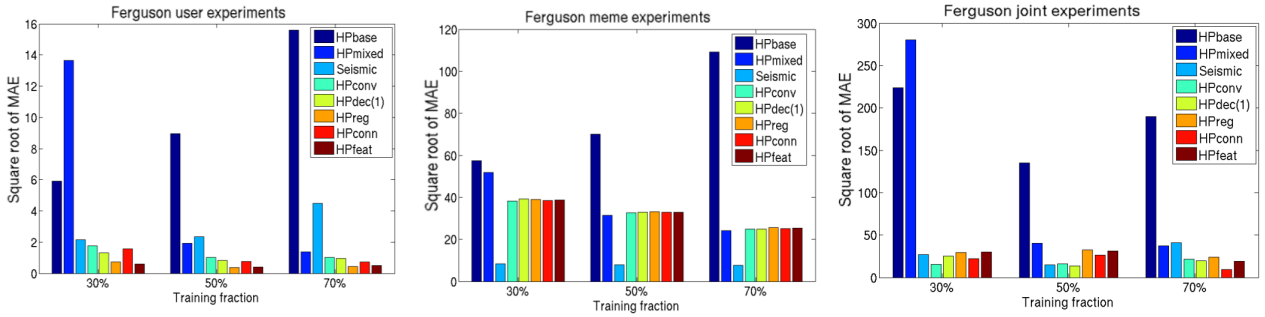

Fig. 5. Results on the Ferguson dataset using MAE on the count of predicted and actual tweets. We use square root of MAE to more clearly depict the variation in performance of all the different approaches.

evaluation on all the training fractions. The ARMSE measure of the models when trained using 30% of the Ferguson data is much higher than that with 50%. This indicates that the models need to see at least 50% of the data to predict reasonably well.

Figure 5 compares the predictive performance of the HP models in terms of MAE on various fractions of training data. *HPfeatures* which consider user features is giving a better performance than the rest with respect to user evaluation. With respect to meme evaluation, *Seismic* outperformed all other approaches. *Seismic* which is specifically developed to predict meme popularity could better predict the tweet count. The HP models tend to improve performance with respect to user evaluation. This is reasonable since the proposed HP models considered factors specific to users such as connection information, user features etc.

We learn the features influencing user infectivity and susceptibility from the one dimensional matrices (weight vectors) that transform user features to a latent dimension. Table II lists the features and the corresponding weight vector values. We find that features such as number of historical tweets, reply tweets and favourites have a larger impact on infectivity than other features. The susceptibility of a user is mainly affected by links ratio and user mentions.

b) Ottawa Shootings: Table VI shows comparison of the MAE performance of the HP models when trained on 30% data. The proposed HP models give better performance than the previous models for user evaluation. Here, *HPconnection* and *HPdecomposition* prove useful in modeling user activity. Incorporating regularization and connection information helped to improve the performance of *HPconversation* on joint

TABLE VI
RESULTS ON OTTAWA USING MAE.

Approach	User	Meme	Joint
HPbase	209±523	7197±5132	81912
HPmixed	253±1419	16208±22083	344457
Seismic	109±1030	**1431±206**	**1749**
HPconversation	20±40	2963±1244	24725
HPdecomposition (1)	9±24	3564±921	8670
HPregularization	13±35	3748±678	5511
HPconnection	**6±23**	3214±1192	4954
HPfeatures (1)	12±30	3005±1181	30957

evaluation while it slightly degraded performance on meme evaluation. Among the various HP models, *HPconnection* gives best performance on joint evaluation. *Seismic* gives the overall best performance on meme and joint evaluation according MAE score.

Next, Table II (right) shows the user features determining susceptibility and infectivity. We find that the order of user features affecting susceptibility is the same as in the Ferguson dataset, with links ratio being the most important. In the Ottawa data set, the links ratio also has an impact on infectivity. In addition to links ratio, the proportion of reply tweets affects infectivity while user mentions affect susceptibility.

c) Sydney Siege: Table VII reports the predictive performance of the proposed approaches *w.r.t.* ARMSE score and trained using 30% of data. We observe that *HPdecomposition* gives best performance on user evaluation and *HPconnection* gives best performance on meme and joint evaluation criteria. As observed before, *HPconnection* uses global connection

TABLE VII
RESULTS ON SYDNEY SIEGE USING ARMSE.

Approach	User	Meme	Joint
HPconversation	16±136	82±21	97
HPdecomposition (1)	**3±76**	727±111	599
HPregularization	16±111	435±424	441
HPconnection	15±139	**36±4**	**49**
HPfeatures (1)	13±119	80±14	96

TABLE VIII
RANK CORRELATION ON MEME POPULARITY.

Approach	Synthetic	Ferguson	Ottawa
HPbase	0.9	0.74	0.42
HPmixed	0.8	0.75	0.51
Seismic	NA	0.57	0.17
HPconversation	1.0	0.73	0.55
HPdecomposition(1)	1.0	0.69	**0.59**
HPregularization	0.8	0.65	0.46
HPconnection	NA	0.66	0.51
HPfeatures (1)	NA	**0.76**	0.58

information to obtain a better predictive performance in determining overall activity in the network. *HPfeatures* also gives a good performance on all the three evaluation dimensions.

E. Predicting Meme Virality

We rank the memes according to the predicted counts by various approaches and compare them with the actual ranking, using the Spearman rank correlation coefficient. In Table VIII, we provide rank correlation for various approaches after training on 30% of data (higher value indicates better correlation with the actual ranking). We observe that on the synthetic dataset, all approaches do well in predicting popular memes. The number of memes is small in synthetic and thus the task is not quite as challenging. Here, *Seismic* predicts the same upper bound for all the memes and hence Spearman rank cannot be computed. On Ferguson, there is not much difference in the scores of different approaches. *HPfeatures* which consider user specific features to improve predictive performance on individual users, are also good in predicting meme popularity in both Ferguson and Ottawa. Though *Seismic* was found to be good in predicting the count of tweets associated with a meme, they are not found to be good in ranking memes according to popularity. Sydney Siege data contain only 2 memes, hence Spearman rank correlation does not provide any useful insights on meme popularity ranking.

VII. CONCLUSION

We proposed an effective approach to integrate the social media and social network features into a point process framework. Considering the conversation structure (*HPconversation*) provides an advantage in terms of computation and is useful in learning user influences. Over-fitting is avoided by decomposing the influence matrix using a matrix factorisation technique or by regularization, both of which improved user activity modeling. The decomposition approach was also shown to be useful in modeling the spread of memes with very few tweets. Considering the social network information for influence learning was found to be more suitable for predicting the overall activity in the network, and the diffusion of larger sized memes. Incorporating features in learning the influence provided insights into which features are most important for infectivity and susceptibility, with *links ratio* proving particularly important. Overall, considering social media features in a point process framework improve the ability to model behavioral dynamics of users and memes.

REFERENCES

[1] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "Seismic: A self-exciting point process model for predicting tweet popularity," in *KDD*, 2015, pp. 1513–1522.

[2] S.-H. Yang and H. Zha, "Mixture of mutually exciting processes for viral diffusion." in *ICML*, vol. 28, 2013, pp. 1–9.

[3] J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec, "Can cascades be predicted?" in *WWW*, 2014, pp. 925–936.

[4] L. Hong, O. Dan, and B. D. Davison, "Predicting popular messages in twitter," in *WWW*, 2011, pp. 57–58.

[5] A. Kupavskii, L. Ostroumova, A. Umnov, S. Usachev, P. Serdyukov, G. Gusev, and A. Kustarev, "Prediction of retweet cascade size over time," in *CIKM*, 2012, pp. 2335–2338.

[6] B. Suh, L. Hong, P. Pirolli, and E. H. Chi, "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network," in *International Conference on Social Computing*, 2010, pp. 177–184.

[7] A. Zubiaga, M. Liakata, R. Procter, G. Wong Sak Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PLoS ONE*, vol. 11, pp. 1–29, 2016.

[8] D. Agarwal, B.-C. Chen, and P. Elango, "Spatio-temporal models for estimating click-through rate," in *WWW*, 2009, pp. 21–30.

[9] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: Quantifying influence on twitter," in *WSDM*, 2011, pp. 65–74.

[10] N. Du, L. Song, M. Yuan, and A. J. Smola, "Learning networks of heterogeneous influence," in *NIPS*, 2012, pp. 2780–2788.

[11] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," in *ICML*, 2011, pp. 561–568.

[12] M. Gomez Rodriguez, J. Leskovec, and B. Schölkopf, "Structure and dynamics of information pathways in online media," in *WSDM*, 2013, pp. 23–32.

[13] M. Gomez Rodriguez, J. Leskovec, and B. Schölkopf, "Modeling information propagation with survival theory," in *ICML*, 2013, pp. 666–674.

[14] M. Farajtabar, Y. Wang, M. Gomez-Rodriguez, S. Li, H. Zha, and L. Song, "COEVOLVE: A joint point process model for information diffusion and network co-evolution," in *NIPS*, 2015, pp. 1954–1962.

[15] H. Xu, M. Farajtabar, and H. Zha, "Learning granger causality for hawkes processes," in *ICML*, 2016, pp. 1717–1726.

[16] L. Yu, P. Cui, F. Wang, C. Song, and S. Yang, "From micro to macro: Uncovering and predicting information cascading process with behavioral dynamics," in *ICDM*, 2015, pp. 559–568.

[17] R. Kobayashi and R. Lambiotte, "Tideh: Time-dependent hawkes process for predicting retweet dynamics," in *ICWSM*, 2016, pp. 191–200.

[18] M. Lukasik, P. K. Srijith, T. Cohn, and K. Bontcheva, "Modeling tweet arrival times using log-gaussian cox processes," in *EMNLP*, 2015, pp. 250–255.

[19] R. D. Perera, S. Anand, K. Subbalakshmi, and R. Chandramouli, "Twitter analytics: Architecture, tools and analysis," in *MILCOM*, 2010, pp. 2186–2191.

[20] S. H. Lee, M. M. Crawford, and J. R. Wilson, "Modeling and simulation of a nonhomogeneous poisson process having cyclic behavior," *Communications in Statistics Simulation, 20(2):777–809*, 1991.

[21] V. Lampos, N. Aletras, D. Preotiuc-Pietro, and T. Cohn, "Predicting and characterising user impact on twitter," in *EACL*, 2014, pp. 405–413.

[22] Y. Ogata, "On lewis' simulation method for point processes." *IEEE Transactions on Information Theory*, vol. 27, no. 1, pp. 23–30, 1981.