Incomplete Gamma Integrals for Deep Cascade Prediction Using Content, Network, and Exogenous Signals

Subhabrata Dutta, Shravika Mittal, Dipankar Das, *Member, IEEE*,
Soumen Chakrabarti, and Tanmoy Chakraborty, *Senior Member, IEEE*

Abstract—The behavior of information cascades (such as retweets) has been modeled extensively. While point process-based generative models have long been in use for estimating cascade growths, deep learning has greatly enhanced the integration of diverse features and signals. We observe two significant temporal signals in cascade data that have not been reported or exploited to our knowledge. First, the popularity of the cascade root is known to influence cascade size strongly; but we find that the effect falls off rapidly with time. Second, we find a measurable positive correlation between the novelty of the root content (with respect to a streaming external corpus) and the relative size of the resulting cascade. Responding to these observations, we propose `GammaCas`, a new cascade growth model as a parametric function of time, which combines deep influence signals from content (e.g., tweet text), network features (e.g., followers of the root user), and exogenous event sources (e.g., online news). Specifically, our model processes these signals through a customized recurrent network, whose states then provide the parameters of the cascade rate function, which is integrated over time to predict the cascade size. The network parameters are trained end-to-end using observed cascades. `GammaCas` outperforms seven recent and diverse baselines significantly on a large-scale dataset of retweet cascades coupled with time-aligned online news — it beats the best baseline with 18.98% increase in terms of Kendall's τ correlation and a reduction of 19.2 in Mean Absolute Percentage Error. Extensive ablation and case studies unearth interesting insights regarding retweet cascade dynamics.

Index Terms—Cascade prediction, social network, exogenous signals, Twitter

◆

1 INTRODUCTION

SHARING and re-sharing are common ways in which content spreads in social networks. A *root user* posts some content (such as a photo or an article) and then *friends* or *followers* of that user share it with their friends, and so on, resulting in a *cascade*. In such a cascade tree, information flows from the root to the leaves. In case of Twitter, resharing is called *retweeting*. The size, duration, and intensity of a reshare cascade are important indicators of user engagement at various levels: within the topic, the community, or the social media platform at large. Modeling user engagement is useful in political discourse mining, market trend analysis, and user-persona detection.

Predicting the progression of a cascade, given early observations at its onset, is known to be a challenging problem [1], [2], [3], [4]. Early approaches [5], [6] relied on three types of features (network structure, root content, and initial observations along time) for modeling the growth of reply trees. Self-exciting point processes [7], [8] were also employed as generative models. Recently, exogenous influence has been incorporated [9], [10]. Neural methods, particularly graph embedding-based techniques, are quickly becoming popular [11], [12].

Despite these advances in feature engineering and modeling approaches, existing methods fail to generalize across data sets, because the importance and interdependence of different features vary sharply over different platforms. Pure point-process based models, however simple and explainable, do not adequately exploit important signals of cascade growth (e.g, content-based features). They often rely on the numerical growth of the cascade over the observed time to predict future behavior. Previous studies [13] as well as our experiments suggest that the predictions of such models are often adversely affected by noise in the observed cascade. Prior neural models often heavily depend on the graph structure of the cascade growth. In most platforms, however, only the cascade participants are observable and not the exact cascade formation path (i.e., if a retweeter is a common follower of two previous retweeters, it is ambiguous to decide which one of them is the predecessor in the cascade graph). Moreover, most of these approaches do not model cascade growth as an explicit function of the prediction horizon. They need to be trained separately for predicting on different prediction horizons.

- *Subhabrata Dutta, Shravika Mittal, and Dipankar Das are with Department CSE, Jadavpur University, Kolkata, West Bengal 700032, India. E-mail: {subha0009, dipankar.dipnil2005}@gmail.com, shravika16093@iiitd.ac.in.*
- *Soumen Chakrabarti is with the Department of CSE, IIT Bombay, Mumbai, Maharashtra 400076, India. E-mail: soumen@cse.iitb.ac.in.*
- *Tanmoy Chakraborty is with the Department of CSE, IIIT Delhi, New Delhi, Delhi 110020, India. E-mail: tanmoy@iiitd.ac.in.*

Manuscript received 12 June 2021; revised 25 March 2022; accepted 2 May 2022. Date of publication 19 August 2022; date of current version 1 May 2023.
(Corresponding author: Tanmoy Chakraborty.)
Recommended for acceptance by S.S. Bhowmick.
Digital Object Identifier no. 10.1109/TKDE.2022.3174206

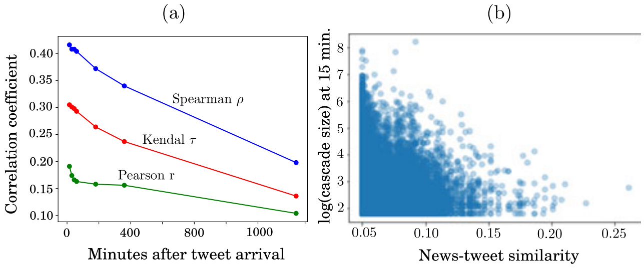


Fig. 1. (a) Correlation between root user's follower count and cascade size at different time after the arrival of the tweet. All three correlation coefficients indicate a decreasing influence of the root follower as the cascade grows further in time. (b) The "novelty premium": tweets that are not mere repetition of current news enjoy greater cascade rates. Average unigram and bigram similarity between a tweet and the news articles published within 12 hrs. before its arrival is plotted against the log of cascade growth (starting 15 mins.). The later value signifies the virality of the tweet among its first responders. We observe a weakly negative correlation (-0.09 Spearman's ρ) but with p-value $< 10^{-5}$.

Our point of departure is the recognition of certain delicate temporal dynamics that existing cascade prediction methods seem unable to exploit, despite their rapidly increasing sophistication. As an example, Fig. 1a shows that, although the root user's popularity (follower count) is initially strongly predictive of cascade growth rate, the effect is not stationary, but rapidly fades with time. As another example, Fig. 1b shows a scatter of cascade sizes (logarithmic) achieved in the first 15 minutes against the content similarity between the root tweet and a body of news articles published shortly before and after the root tweet. It hints at a certain "novelty premium" — text that is not mere repetition of current news enjoys greater cascade rates. However, such influences interact with each other in a complex, non-linear fashion to govern the cascade growth. For example, let us consider a root user positioned within a clique of the social network that is rarely connected with the rest of the network. Tweets from such a user may enjoy a high exposure at the beginning; however, their popularity will decay sharply once the rest of the clique is fully exposed to the content. However, the novelty of the content can break this 'clique barrier' due to the underlying recommendation algorithm of the platform. Since in most real platforms, neither the full network information nor the recommendation algorithm is known apriori, a predictor is required to estimate these dynamics from partial observations.

Guided by observations like the ones narrated above, we present **`GammaCas`**, a novel deep model for cascade prediction. We directly model the gradient of cascade growth as a trainable neural function of content, network, and exogenous features. Specifically, we monitor network (popularity) features evolving through time, and feed (continuous forms of) these features into a novel LSTM [14] variant, whose *hidden states are then mapped to parameters that dictate the gradient of cascade growth*. Textual and exogenous features modulate how LSTM states influence the temporal process parameters.

The gradient of cascade growth is then integrated over the past to predict the size of the cascade at a given time beyond the observation horizon. Inspired by many natural growth processes [15], [16], we model cascade trajectory as an incomplete gamma function by integrating its temporal derivative numerically. This allows us, during training, to back-propagate prediction errors and train all model weights end to end.

We report on extensive experiments using 342,111 resharing cascades from Twitter, temporally aligned with 206,180 news articles published online on 5,138 news sources. We compare `GammaCas` against several recent competitive approaches: a basic Hawkes process, SEISMIC [17], TiDeH, a time-dependent Hawkes Process [7], NeuralPointProcess [18], CasPred [2], DeepHawkes [12], DeepCas [11] and ChatterNet [10]. `GammaCas` achieves lower mean absolute percentage prediction error compared to these prior systems. It is more stable and robust to variations in prediction horizons, compared to some prior systems. Another benefit of `GammaCas`'s transparent network design is that, by correlating observable features against the parameters involved in the time integration, we get additional insights into the factors that govern cascade dynamics.

Summarizing, our major contributions are as follows:

- We propose `GammaCas`, a novel framework for reshare cascade prediction which incorporates content, network and exogenous signals over observable cascade progress to learn parametric representation of cascade growth at a future time. `GammaCas` achieves a Kendall's τ correlation of **0.63** (**25.06** Mean Absolute Percentage Error) between predicted and actual size of the cascade at 24 hours, after only 6 hours of early observation.
- We collect and contribute a large-scale dataset of recent retweet cascades with a temporally aligned stream of online news articles.
- We compare `GammaCas` with several recent baselines for cascade size prediction developed upon generative, feature-driven, and neural network-based approaches. While `GammaCas` outperforms each of these baselines by a significant margin, we also investigate the behaviors of these baseline models on our dataset.
- We perform in-depth ablation and case study using `GammaCas` to investigate into the different signals influencing its parameter estimation. We present insights from these experiments which may be of independent interest.

Organization. The rest of the paper is organized as follows:

- We review the relevant literature on cascade and popularity prediction in Section 2, focusing on point-process and neural methods that incorporate different influence signals.
- `GammaCas` is presented in Section 3 with detailed descriptions of its various functional components.
- In Section 4 we describe the dataset preparation, training protocols of `GammaCas`, baseline methods and ablation variants of `GammaCas`.
- We present experimental results in Section 5.
- We conclude with important observations and possible future direction in Section 6.

Reproducibility. To encourage reproducible research, we present detailed hyper-parameter configurations in the Supplementary material, Section 2, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2022.3174206.

Moreover, we supplement our submission with dataset and source code of `GammaCas`, available at: https://github.com/LCS2-IIITD/GammaCas.

2 RELATED WORK

Prior works in the field of information cascade modeling can be broadly distinguished into two categories: *Macro* cascade modeling focused on the overall growth and structural properties of a cascade (e.g., retweet count prediction) [2], [7] and *Micro* cascade modeling which investigates the behavior and dynamics of individual agents participating in the cascade (e.g., retweeter prediction) [19], [20], [21]. Our work specifically aligns with the macro category.

Feature-Driven Cascade Modeling. Among the earliest of works, Cheng et al. [2] studied the structural and temporal properties of resharing cascades and came up with a feature-driven strategy to devise a classification problem: after observing a cascade reaching a size k, what is its probability of reaching size nk? Bakshy et al. [22] attempted to identify potential influencers in a feature-driven approach to predict information cascades. To explore richer feature set of cascade dynamics, Krishnan et al. [23] conceptualized cascades as information flow along forests as opposed to the usual tree structure. Most of the feature-driven approaches have revolved around temporal features [2], [24], structural and network features [22], [25], user features [22], [26] and content features [27]. Kong et al. [28] modeled the evolution of different stages of cascade growth based on various such factors. While feature-based approaches have produced seminal insights regarding the dynamics of cascade growth, they require heavily curated manual feature engineering that are strongly platform-dependent.

Generative Models for Cascade Prediction. An alternative emerging approach that has seen significant success involves generative models that perceive cascades as temporal event arrival sequences, generates random arrival sequences conditioned on certain parameters, and finally maximizes a chosen likelihood function between the observed and generated sequences [7], [17], [29]. Crane and Sornette [30] described the view dynamics of YouTube as an epidemic modeled by a self-exciting Hawkes Process. Multiple studies reported using Hawkes Process or its modified variations to predict retweet cascade size [7], [8], [17], [31], [32]. In a cross-platform setting, Rizoiu et al. [8] used a Hawkes process to model popularity growth of content in one platform controlled by endorsement provided in other platforms. Mishra et al. [13] combined feature-driven approach with Hawkes process for popularity prediction. Rizoiu et al. [33] proposed a hybrid of epidemic and self-excitation models to analyse diffusion cascades. Although not often applied to cascade modeling, recent advances have been used to model more complex dynamics of temporal point processes using neural networks [18], [34]. Li et al. [35] extended a Hawkes process to model conformity hidden in diffusion activity. Other than point-process models, a few others explored epidemic models [27], [36], [37], Bass model [38], [39], Survival Analysis [40], [41], Jump Processes [9], etc. Despite their explainable behavior and zero need for heavy feature engineering, generative models are susceptible to adverse influences from outliers [13] and found less powerful at making precise predictions [12].

Neural Network Based Methods. Recently, neural models have facilitated more powerful representations of two major components of cascade predictions: recurrent neural architectures can learn the complex temporal dynamics of early observation without constrained approximations [42], [43] and graph learning methods render the integration of complex structural properties to be seamless [11]. In their proposed model DeepCas, Li et al. [11] sought to learn the structural properties of observed retweet cascade using random walk embeddings of the cascade graph and aggregated the dynamics using gated recurrent units with attention. DeepHawkes was proposed by Mishra et al. [13] to translate the explainable behavior of Hawkes Process into the representational superiority of neural networks to predict retweet and citation cascades. Cao et al. [44] modeled the interplay between the social network and influence network for popularity prediction by coupling two graph neural networks. Dutta et al. [10] proposed ChatterNet to model the growth of reply cascades in the absence of explicit knowledge about a social or information network like Reddit. Their model integrates exogenous and endogenous influence to learn textual representations of content using time-evolving convolution kernels, and aggregates the observed cascade growth using LSTMs. One implementation challenge regarding most of these models is their lack of flexibility to migrate to different observation/prediction horizons without retraining. Moreover, in most of the cases, the superior representation power of neural network-based models is overshadowed by the lack of explainability and the inability to produce actionable insights from the learned representations.

Exogenous Influence Over Cascade Growth. While the works mentioned above mostly focus on the driving factors of cascade growth implicit to the cascade and the platform, signals exogenous to the platform can also heavily influence the virality and popularity of content [10], [45]. Prior works seeking to identify the influence of exogenous event arrivals have explored point processes with self and external excitation to model observed event sequences [46]. De et al. [47] attempted to demarcate opinion diffusion in Twitter under the influence of exogenous influence from endogenous ones. Broxton et al. [48] investigated the influence of external information sources on virality of online video content. Cascade predictions based on cross domain influences are specialized scenarios of modeling and exploiting signals external to a platform, i.e., predicting YouTube view cascades from Twitter cascades [49]. Dutta et al. [10] employed a similar strategy to incorporate exogenous signals; with the target domain being Reddit, their source domain of external influence was free-flowing new-streams on online news portals.

Given this vast prior development in modeling cascade dynamics, our proposed `GammaCas` model seeks to deliver a generalizable, flexible model for cascade growth prediction, similar to the generative family while incorporating the powerful representation capability of neural methods in an end-to-end fashion to capture the temporal, network-based, content-based and exogenous influences on the cascade growth.

Differences Between ChatterNet [10] and `GammaCas`*.* Among the discussed models for cascade and popularity predictions, ChatterNet seeks to use a set of influence signals similar to ours. It predicts the future chatter intensity under a submission on Reddit, defined as the number of comments posted under that submission. However, there are some key differences: (i) Owing to the closed definition of Reddit's communities (i.e., subreddits), the original design of ChatterNet is able to characterize *endogenous influences* in terms of contemporary submissions posted within that subreddit. This is not at all possible for a Twitter-like open platform. Instead, `GammaCas` uses the social network information (i.e., follower count of users) to model the endogenous influence. ChatterNet is not developed to handle such information because Reddit does not provide any. (ii) Being a purely deep learning based model like DeepCas [11], ChatterNet does not learn the prediction function as explicitly dependent on the prediction horizon. Therefore, a new training setup is needed for each different prediction horizon. `GammaCas` overcomes this lack of flexibility by learning a parametric estimation of retweet arrival intensity and then performing numerical integration of the said intensity function over the prediction horizon. This novel hybrid of deep feature learning with numerical function approximation empowers `GammaCas` with the flexibility that, once trained, it may predict for arbitrary prediction horizons.

3 PROPOSED MODEL

In this section, we describe `GammaCas` in detail. It has many modules which may appear complex, but we will justify their utility through ablation in Section 4.

3.1 Preliminaries and Problem Definition

Let $\mathcal{G} = \{\mathcal{U}, \mathcal{E}\}$ be a directed graph representing the social network of Twitter, where $\mathcal{U}$ is the set of vertices representing the users and $e_{ij} \in \mathcal{E}$ if u_j *follows* u_i for any $u_i, u_j \in \mathcal{U}$. Therefore, the follower count of any given user u_i translates to the out-degree of the corresponding node in $\mathcal{G}$.

Given a tweet τ posted by a user u at time t_0, its *retweet cascade at time* $t > t_0$ can be defined as *an ordered sequence of retweet arrival timestamps along with the corresponding retweeter*, $\mathcal{R}^\tau_t = \{(t_i, u_i) | t_i > t_j \text{ for } i > j, t_i \leq t\}$. The exogenous event signals within any time frame $[t, t + \Delta t]$ are substantiated as the sequence of news articles $N(t, t + \Delta t) := \{(n_j, t_j) | t \leq t_j < t + \Delta t\}$, where n_j is an article published at t_j. Without loss of generality, we idealize exogenous influence on users to be captured by this news article stream. Other possible stimuli can be integrated into our model similarly.

For any given cascade $\mathcal{R}^\tau_t$, we define the *early observation window* to be $(t_0, t_0 + \Delta_{obs}]$. A model would estimate the future growth of the cascade upon observing the dynamics within this observation period. We also define a *prediction horizon* $[t_0, t_0 + \Delta_p], \Delta_p > \Delta_{obs}$, so that our problem translates to predicting $|\mathcal{R}^\tau_{t_0+\Delta_p}|$ upon observing $\mathcal{R}^\tau_{\Delta_{obs}}$, τ, and $N(t_0 - \Delta_{obs}, t_0 + \Delta_{obs})$. Henceforth, for the sake of brevity, we will consider $t_0 = 0$ in general.

Notation. Table 1 summarizes important notations and denotations. While describing `GammaCas`, we use bold lower cased symbols to denote vector inputs and outputs, and bold upper cased symbols to denote sequences of vectors as well as the trainable parameters of `GammaCas`.

TABLE 1
Denotation of Important Notations Used

Notation	Denotation
$\mathcal{R}^\tau_t$	Retweet cascade of tweet τ through time t
Δ_{obs}	Intial observation window of cascade
Δ_p	Prediction horizon for future cascade
Δ_o	Binning size of the observation window
M	Number of bins in observation window
$N(t_1, t_2)$	News articles published within $[t_1, t_2)$
C^r_m	Total retweets within mth observation bin
C^f_m	Total followers within mth observation bin

3.2 Parametric Estimation of Cascade Growth

As As Zhao et al. [17] suggested, a cascade can be either in a *supercritical* stage (rate of cascade growth is increasing) or in a *subcritical* stage (rate of cascade growth is decreasing) at different points of time, depending on multiple factors like the relevance of the content expressed by the piece of tweet, outdegree of the nodes participated in the cascade by that time, inter-arrival time of retweets, etc. Extending discrete-valued $\mathcal{R}^\tau_t$ to a continuous, real-valued map of time, we can redefine these two stages as $\frac{d^2|\mathcal{R}^\tau_t|}{dt^2} \geq 0$ (supercritical) or $\frac{d^2|\mathcal{R}^\tau_t|}{dt^2} < 0$ (subcritical). Such a rate of growth can be modeled as a simple product of two functions of time,

$$\frac{d|\mathcal{R}^\tau_t|}{dt} = \Psi_1(t)\Psi_2(t), \tag{1}$$

constrained with the following conditions: i) $\Psi_1(t), \Psi_2(t) > 0$, ii) $\frac{d\Psi_1}{dt} > 0, \frac{d\Psi_2}{dt} < 0$ and iii) $\lim_{t\to+\infty} \Psi_1(t)\Psi_2(t) = 0$. The first condition ensures a monotonous growth of the cascade, while the second and third conditions ensure a possible initial supercritical growth followed by a mandatory subcritical growth.

Simple choices for such functions would be a polynomial Ψ_1 and an exponentially decaying Ψ_2. Concretely, we can approximate Eq. (1) in a parametric form as follows:

$$\frac{d|\mathcal{R}^\tau_t|}{dt} = At^\gamma e^{-\lambda t}, \tag{2}$$

where A, γ, and λ are arbitrary constants that govern the 'shape' of the temporal growth pattern of the cascade. γ controls how steeply the cascade will grow in the supercritical stage while λ dictates the onset of decay in popularity. A acts as an overall modulation parameter. One can draw analogies between the roles of these three parameters and the influence signals described in Section 1. A higher follower count of the root user signifies that the content is exposed to a large number of users from the very beginning, which translates to a higher value of A and γ. On the other hand, if the retweeters have a diminishing follower count compared to the root, a steeper decay (i.e., higher λ) is expected. Similar analogies can be drawn with the degree of news-tweet similarity.

The choice of such a function restrains $\frac{d|\mathcal{R}^\tau_t|}{dt}$ to a single "hill"-shaped curve corresponding to a single supercritical and single subcritical phase, whereas real cascades may have multiple consecutive super- and subcritical phases. The growth rate of such cascades can be easily approximated as

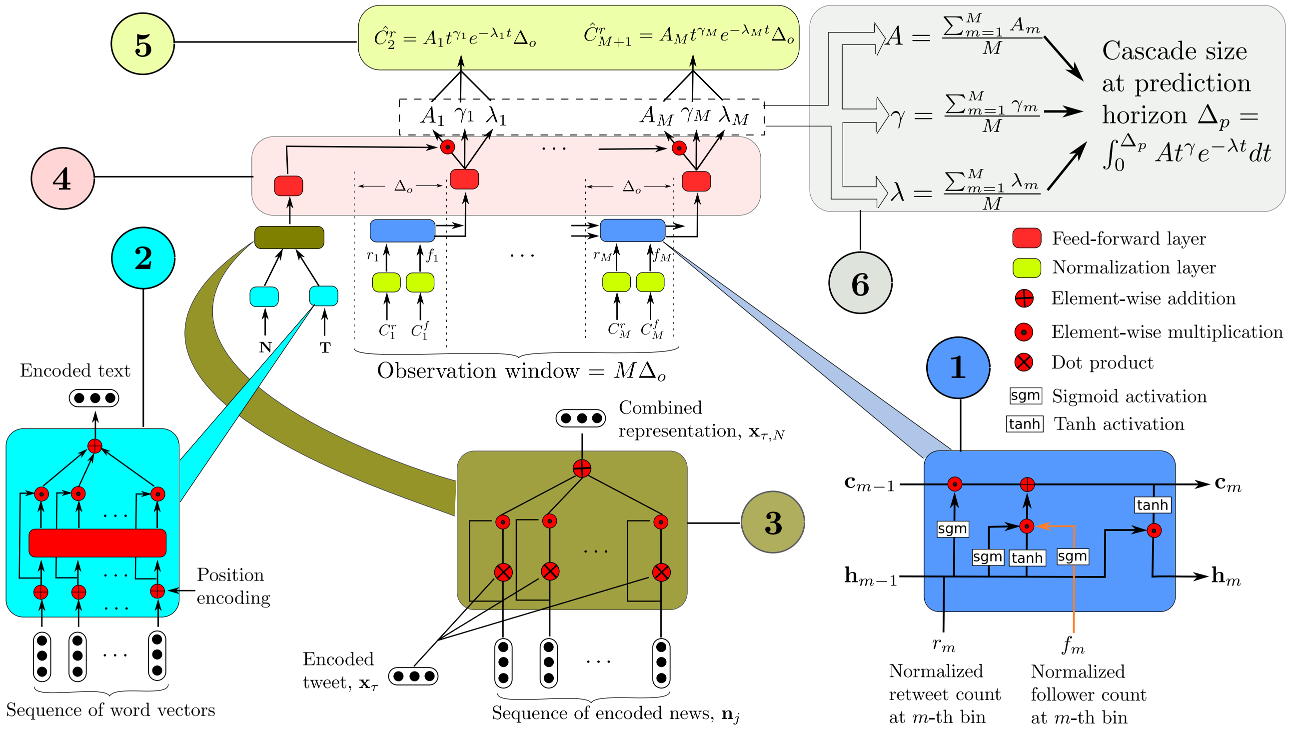


Fig. 2. Design of `GammaCas` explained with its different modules. Retweet count and aggregate follower count at each observation bin (Δ_o) is normalized and fed to (1) the modified LSTM layer (Section 3.3). Textual content from tweet and news are processed in (2) the text processing module (Section 3.4) which performs word-wise attention and aggregation to generate a single vector per piece of text. Encoded tweet and sequence of news articles are then combined into a single representation in (3) a scaled dot-product attention layer (Section 3.5). Hidden state output from (1) at each bin and the news-tweet combined representation from (3) are then used in (4) the parameter estimation module to compute the parameters A_m, γ_m, and λ_m for each bin m (Section 3.6). In (5) the autoregressive module, the mth set of parameters is used to predict the retweet arrival at $(m+1)$th bin and the average-pooled parameters are used in (6) where the future cascade size at prediction horizon Δ_p is computed (Section 3.7).

$$\frac{d|\mathcal{R}_t^\tau|}{dt} = \sum_k A_k(t-\phi_k)^{\gamma_k} e^{-\lambda_k(t-\phi_k)}, \tag{3}$$

where A_k, λ_k, γ_k correspond to the growth parameters of the kth "hill" in the cascade growth and ϕ_k correspond to its starting time.

However, we are interested in the size of the cascade after a finite amount of time T, which is given by

$$\begin{aligned}\mathcal{R}_T^\tau &= \int_0^T \sum_k A_k(t-\phi_k)^{\gamma_k} e^{-\lambda_k(t-\phi_k)}dt\\ &= \sum_k A_k \int_0^T (t-\phi_k)_k^{\gamma} e^{-\lambda_k(t-\phi_k)}dt\\ &= \sum_k \frac{A_k}{\lambda_k^{(\gamma+1)}}\left(-\Gamma(\gamma_k+1, \lambda(t-\phi_k))\right)\Big|_{t=0}^{t=T}\\ &= \sum_k \frac{A_k}{\lambda_k^{(1+\gamma_k)}}\left(\Gamma(\gamma_k+1, -\lambda_k\phi_k) - \Gamma(\gamma_k+1, \lambda_k(T-\phi_k))\right), \end{aligned}\tag{4}$$

where $\Gamma(s,z)$ are *incomplete Gamma functions*. Since any generalized incomplete Gamma function can be represented as a finite sum of modified Bessel functions of the first kind [50], the above form is equivalent to a single Gamma function with suitably chosen values of the arbitrary constants. Therefore, we choose to model cascade growth as a parametric function

$$|\mathcal{R}_{\Delta_p}^\tau| = \int_0^{\Delta_p} At^\gamma e^{-\lambda t}, \tag{5}$$

where the parameters, in turn, are estimated as (neural) functions of $\mathcal{R}_{\Delta_{obs}}^\tau$, τ, and $N(t_0, t_0+\Delta_{obs})$.

3.3 Capturing Temporal Dynamics of Retweet Arrival

At any time t, the rate of cascade growth $\frac{d|\mathcal{R}_t^\tau|}{dt}$ directly depends on the retweets arriving within interval $(t, t+\Delta)$. The exact number of retweets (we denote it as $C_{t,t+\Delta}^r$) arrived within this interval directly adds to the size of the cascade. Moreover, each of the new retweets expands the number of potential future retweeters (i.e., susceptible nodes) by the out-degree of the current retweeter.

To capture this temporal dynamics within the early observation phase, we quantize the observation window into M consecutive, equal-sized bins of size Δ_o (i.e., $\Delta_{obs} = M\Delta_o$), where M is an application-driven hyperparameter. We denote the total number of retweets arrived within the mth bin as C_m^r, where $m \in [M]$. We aggregate the additional number of susceptible nodes created within the mth bin as $C_m^f = \sum_j outdegree(u_j),\ \forall (t_j, u_j) \in \mathcal{R}_{m\Delta_o}^\tau \setminus \mathcal{R}_{(m-1)\Delta_o}^\tau$. Furthermore, as shown in Fig. 2 (module 5), we apply trainable normalization on the integer elements of C_m^r and C_m^f to avoid gradient saturation in the subsequent layers of our framework. The resulting values are denotes as r_m and f_m, respectively.

The sequences $\{r_m\}_{m=1}^M$ and $\{f_m\}_{m=1}^M$ represent the temporal dynamics of cascade growth within the observation window, and a simple choice of architecture to model it would be from the Recurrent Neural Network (RNN) family. While LSTMs have been successfully applied to model temporal dependencies over long sequences, we modify information flow along the LSTM gates according to the intuitive knowledge of the retweet arrival dynamics. As shown in Fig. 2 (1), the modified LSTM cell in our architecture instantiates the following six operations:

$$\mathbf{x}_g = \sigma(\mathbf{W}_g[r_m : \mathbf{h}_{m-1}] + \mathbf{B}_g) \tag{6}$$

$$\mathbf{x}_{in} = \sigma(\mathbf{W}_{in}[r_m : \mathbf{h}_{m-1}] + \mathbf{B}_{in}) \tag{7}$$

$$\mathbf{x}_c = \tanh(\mathbf{W}_c[r_m : \mathbf{h}_{m-1}] + \mathbf{B}_c) \tag{8}$$

$$\mathbf{x}_f = \sigma(\mathbf{W}_f f_m + \mathbf{B}_f) \tag{9}$$

$$\mathbf{c}_m = \mathbf{c}_{m-1} \odot \mathbf{x}_g + \mathbf{x}_{in} \odot \mathbf{x}_c \odot \mathbf{x}_f \tag{10}$$

$$\mathbf{h}_m = \mathbf{h}_{m-1} \odot \tanh(\mathbf{W}_h \mathbf{c}_m + \mathbf{B}_h), \tag{11}$$

where [:] denotes concatenation; $\odot$ denotes the Hadamard product; σ signifies the sigmoid non-linearity; $\mathbf{c}_m$ and $\mathbf{h}_m$ correspond to the cell and hidden state of the LSTM after the mth timestep (observation bin) respectively; $\mathbf{W}_g$, $\mathbf{W}_{in}$, $\mathbf{W}_c$, $\mathbf{W}_f$, $\mathbf{W}_h$ are the learnable weight matrices, and $\mathbf{B}_g$, $\mathbf{B}_{in}$, $\mathbf{B}_c$, $\mathbf{B}_f$, $\mathbf{B}_f$ are the learnable bias matrices.

Equations (6), (7), (8) and (11) correspond to the operations performed by the original LSTM cell. However, Equation (9) generates a modulation signal $\mathbf{x}_f$ from the out-degree of the participating user nodes at that step to control the contribution of their retweets. Equation (10) takes this modulation into account to update the cell state for the current step. Moreover, this modification decreases the size of the parameter space compared to the original LSTM. Assuming the state size to be s, the four weight matrices of the original LSTM cell would incur a total of $12 \times s$ number of weight and bias parameters, while the modified one uses $10 \times s$ parameters due to split inputs.

3.4 Processing Textual Content

We take every piece of text (tweet or news) as a sequence of words and compute a single vector representation of the text relevant to the downstream task, as shown in Fig. 2 (module 2).

We use a trainable embedding layer to map each word w_i to a d-dimensional vector $\mathbf{v}_i \in \mathbb{R}^d$, converting a piece of text into a sequence of vectors $\mathbf{V}$. Typical content-sharing platforms like Twitter incur heavy traffic, with millions of textual pieces arriving each second. To speed up the processing, we intend to maintain parallel operations on $\mathbf{V}$. Consequently, we do not use any sequential architecture involving variants of RNN to encode the representation. Instead, we compute a *positional encoding* vector [51] $\mathbf{p}_i \in \mathbb{R}^d$ as

$$p_i^{(j)} = \begin{cases} \sin(\omega_i) \text{ if } j \text{ is even} \\ \cos(\omega_i) \text{ otherwise} \end{cases},$$

where $i, j \in \mathbb{N}$, $\omega_k = L^{-\frac{2k}{d}}$, L is the maximum length of the input text sequence in the corpus, and $p_i^{(j)}$ denotes the jth element of the vector $\mathbf{p}_i$. The embedded sequence of words, $\mathbf{V}$ is then transformed to a position encoded sequence $\mathbf{V}' = \{\mathbf{v}'_i | \mathbf{v}'_i = \mathbf{v}_i + \mathbf{p}_i\}$.

Next, for every token position, we compute an attention weight α_i using a feed-forward layer followed by a softmax activation:

$$\alpha_i = \frac{e^{s_i}}{\sum_i e^{s_i}}, \tag{12}$$

where $s_i = \mathbf{W}_a \mathbf{v}_i + \mathbf{B}_a$, $\mathbf{W}_a$ and $\mathbf{B}_a$ are learnable weight and bias matrices, respectively. We compute the final representation of the text as weighted aggregation of $\mathbf{V}'$ as $\sum_i \alpha_i \mathbf{v}'_i$. Intuitively, Equation (12) generates a word-wise attention weight sequence, which modulates the contribution of each word in the final representation of the text.

We also experimented with more complex text encoding methods like the Transformer encoder, Bi-LSTM encoder, and BERT. These models incurred higher training/inference cost in terms of memory and time with no significant improvement over our proposed method. As Dutta et al. [10] suggested, popularity of a content in social media is majorly governed by simpler textual features like topic, polarity, etc. which can be easily captured by simpler models, and sophisticated NLP methods tend to be overkill. Furthermore, the per-word weights, α_i, computed by this proposed approach further serve to explain the effects of the textual content of the tweet on the growth of the resulting retweet cascade.

3.5 News-Tweet Attention as Exogenous Influence

For a given tweet τ and a sequence of news N, the text processing module outputs a single vector $\mathbf{x}_\tau$ and a sequence of vectors $\{\mathbf{n}_j\}$, respectively. As exogenous influence on cascade growth varies for tweets expressing different topics, we amalgamate the two signals to compute the final influence, as shown in Fig. 2 (3).

We compute an attention weight between the tweet representation $\mathbf{x}_\tau$ and a news representation $\mathbf{n}_j$ as

$$\beta_{\tau,j} = \mathrm{softmax}_j\left(\frac{\mathbf{x}_\tau^\top \mathbf{n}_j}{\sqrt{d}}\right). \tag{13}$$

The scaling component $d^{-0.5}$ reduces the chance of $\mathrm{softmax}(\cdot)$ reaching saturation. Similar to the text processing module, the final representation of the exogenous influenced tweet text is computed as $\mathbf{x}_{\tau,N} = \sum_j \beta_{\tau,j} \mathbf{n}_j$.

3.6 Computing Cascade Growth Parameters

The cascade growth parameters A, γ, and λ (see Equation (5)) are computed from the textual representation $\mathbf{x}_{\tau,N}$ and the observed cascade dynamics encoded by the modified LSTM, $\mathbf{h}_m$ (see Equation (11)). We hypothesize that while the growth and decay parameters, γ and λ, can be estimated from observing the retweet arrivals exclusively, the scaling parameter A is dependent on the tweet text and the exogenous influence.

We map $\mathbf{h}_i$ to three separate non-negative scalars, A'_m, γ_m and λ_m, using three parallel feed-forward layers as follows:

$$A'_m = \mathrm{relu}(\mathbf{W}_A \mathbf{h}_m + \mathbf{B}_A) \tag{14}$$

$$\gamma_m = \mathrm{relu}(\mathbf{W}_\gamma \mathbf{h}_m + \mathbf{B}_\gamma) \tag{15}$$

$$\lambda_m = \mathrm{softplus}(\mathbf{W}_\lambda \mathbf{h}_m + \mathbf{B}_\lambda). \tag{16}$$

We choose these activations experimentally. While $\mathrm{relu}(\cdot)$ is the most straightforward activation function to ensure non-negative output, `GammaCas` suffers from the zero-gradient problem of ReLU while computing λ_i, so we use softplus.

Next, we compute a modulation parameter emerging from the tweet and the exogenous signals as another non-negative scalar value and scale A'_i as follows:

$$A_m = A'_m \text{relu}(\mathbf{W}_\mu \mathbf{x}_{\tau,N} + \mathbf{B}_\mu), \tag{17}$$

where W_μ and B_μ are learnable parameters of a feed-forward layer.

3.7 Final Prediction

From Equations (14), (15), and (16), we estimate the cascade growth parameters for each observation bin. We apply average-pooling from these three sequences to get the cascade size parameters A, γ, and λ. For a given prediction horizon Δ_p, the predicted size of the cascade can then be found by solving the integration in Equation (5). We use 4th order Runge-Kutta method with fixed number of steps to solve this integration numerically and predict the cascade size at Δ_p as Y_{Δ_p}.

Learning to estimate the aggregate parameters of cascade growth at some prediction horizon is the primary task which `GammaCas` is designed for. However, within the observation window, a fine-grained prediction modeling of retweet arrival is supposed to help the model learn more robustly. We use a joint learning strategy in an autoregressive setting. At the mth observation bin, we have already estimated the parameters A_m, γ_m, and λ_m. From these, we predict the aggregate retweet arrival at the $(m+1)$th bin as $\hat{C}^r_{m+1} = A_m t^{\gamma_m} e^{-\lambda_m t}\Delta_o$. The gradient from the loss can be back-propagated through the quadrature [52].

Loss/Cost Function. We use two different loss functions to train the model in the joint learning setting. As future cascade size varies largely, we use the *Mean Absolute Percentage Error* between the predicted and actual cascade size at a prediction horizon Δ_p, as suggested by Dutta et al. [10]. For the autoregressive task of predicting retweet arrival in the next observation window, we use Mean Squared Error loss. The final loss function therefore becomes

$$J = \frac{\left||\mathcal{R}^\tau_{\Delta_p}| - Y_{\Delta_p}\right|}{|\mathcal{R}^\tau_{\Delta_p}|} + \zeta \sum_{m=1}^{M} (C^r_{m+1} - \hat{C}^r_{m+1})^2 / M, \tag{18}$$

where $\zeta < 1$ is a hyperparameter to set the relative importance of the autoregressive gradient.

4 EXPERIMENTAL SETUP

In this section, we present the dataset used in the experiments, the baselines and ablation variants of `GammaCas` considered for the comparison. Implementation details of `GammaCas` is provided in the supplementary material, available online.

4.1 Dataset

As collecting retweet information and parallel news articles for existing datasets often result in lots of missing information, we proceed with curating a dataset of our own. Overall, we use a total of 239,478 and 102,633 retweet cascades, respectively, for training and testing purposes. To encode exogenous signal, we use a total of 206,180 news articles published online within the same time period as the cascades. Additional details of data collection is provided in the supplementary material, available online.

Furthermore, we use two existing datasets to investigate the generalization of `GammaCas`: (i) *ActiveRT 2017* [32] contains 30,535,891 retweet cascades originated from tweets published in the year 2017 that mention YouTube videos; (ii) *Sina-Weibo* [12] contains a total of 119,313 cascades originated from posts in Sina-Weibo platform in June 1, 2016.

4.2 Baseline Methods

To compare the performance of `GammaCas`, we implement a diverse set of baselines from generative, feature-based, and neural network-based families of frameworks.

4.2.1 Generative Baselines

We implement the following three self-excitation process-based models:

Hawkes. We implement a univariate Hawkes Process-based model with the exponential kernel, optimized using maximum log-likelihood estimation to provide a basic generative baseline for future cascade size prediction on our data.

SEISMIC, proposed by Zhao et al. [17], uses a self-exciting point process for retweet cascade prediction combined with the exposure provided by a user's follower base.

TiDeH, a time-dependent Hawkes Process [7], looks at how a cascade evolves with time considering the network structure and aging of information.

DMHP, a dual-mixture Hawkes Process model proposed by Kong et al. [32].

4.2.2 Feature-Driven Baseline

Following the work of Cheng et al. [2], we implement *CasPred* to predict whether a given cascade will reach a particular size, exploiting rich, hand-crafted temporal and textual features of the cascade. We implement two versions of the model as our baseline – *CasPred (org)* which uses a subset of the original features used, applicable to our setting, and *CasPred (add)* which uses additional features proposed by Dutta et al. [10].

4.2.3 Neural Network Baselines

We consider the following three recent neural architectures as baselines:

NNPP or Neural Network Point Process [18] is an RNN-based method for generalized modeling of temporal point processes.

DeepHawkes [12] is an end-to-end deep learning framework that combines the predictive power of models based on neural network architectures and interpretability of cascades provided by the Hawkes Process.

DeepCas [11] is a neural network model for predicting cascade growth. It learns a representation of cascade networks by sampling node sequences through random walks processes, thereby leveraging the structural information of the network.

ChatterNet [10] is a neural network model to predict social chatter intensity leveraging on exogenous and endogenous influence combination. To apply it in our setting, we remove the endogenous influence module, resulting in a single LSTM layer integrating exogenous signals from news. Moreover, we incorporate aggregated follower count at each observation bin (similar to `GammaCas`) in addition to retweet arrival.

4.3 Ablation Variants

We seek to investigate the contributions of different components of `GammaCas` in the overall performance by ablation. We explore the following three ablation variants:

`GammaCas` -text. We take away the contribution of exogenous influence in this variation by removing the scaled dot-product attention between news and tweet. In this variation, the modulation parameter μ in Section 3.6 is computed by applying the feed-forward layer transformation on the tweet text representation X_τ only.

`GammaCas` -CO. In this variation, contributions from the tweet content as well as the exogenous influence are ablated; retweet growth parameters are estimated from the cascade growth dynamics in the observation window alone, using the modified LSTM layer.

`GammaCas` -LSTM. To investigate the gain in modeling capacity enforced by the modifications we applied on LSTM gates in Section 3.3, we replace it with the original LSTM layer with rest of the components unchanged.

5 RESULTS AND DISCUSSION

The growth of a retweet cascade is a stochastic process that is hard to predict, as random events may shift the growth dynamics of a cascade even after a sufficient observation window. It is important for a model to decide which tweets possess the potential to generate a larger cascade compared to another even when the predicted sizes may not be in range with the actual cascade sizes in a future time. For this reason, we compare `GammaCas`, its variants, and all the baselines with three evaluation metrics — *Mean Absolute Percentage Error (MAPE)* to estimate the difference in predicted and actual sizes; *Kendall's τ* and *Spearman's ρ correlation* between the predicted and actual set of cascade sizes to estimate the models' ability to rank tweets according to their potential to generate cascades. As CasPred predicts whether a cascade will reach a certain size range instead of predicting the actual size, we compute *step-wise Kendall's τ correlation* [10] between the predicted range and the actual range.

5.1 Overall Performance

In Table 2, we present the performance of `GammaCas`, its ablation variants, and baselines to predict cascade size at 24 hrs. prediction horizon upon 6 hrs. observation window.

5.1.1 Comparison Among Baselines

All the purely generative models (SEISMIC, TiDeH, and Hawkes) yield high MAPE (i.e., poor performance) across all prediction horizons. After investigating the actual predictions made by these three models, we find that these models often overestimate the future cascade size by a large margin (often to an order of 10^3–10^4). Though excluding such cases results in a performance comparable to `GammaCas`, the fraction of such overestimating instances is high enough ($>20\%$) to cause performance instability. *Among the generative baselines, in terms of correlation coefficients, SEISMIC emerges as the best performing generative baseline, while TiDeH stands as best in terms of MAPE.*

All the three neural network-based baselines perform closely with respect to all the evaluation metrics, with DeepCas emerging as the best performing one. ChatterNet suffers from the tailoring we had to introduce for the sake of making it applicable to retweet cascade prediction in a different problem setting altogether. Neural network-based model of temporal point processes is able to model cascade growth better compared to simple generative models. However, NNPP does not take any other features except the retweet-arrival statistics. This explains its limitation compared to DeepCas. *In terms of consistent performance on variable-sized cascades and MAPE, we consider DeepCas to be the best performing baseline altogether.*

TABLE 2
Comparison With the Baselines and the Variants of `GammaCas`

Model	τ	ρ	MAPE (%) ↓	Step-τ	t/s (ms.)
Hawkes	0.202	0.277	110.25	0.231	196.72
SEISMIC	0.532	0.572	138.86	0.522	67.80
TiDeH	0.306	0.403	77.90	0.370	14.59
DMHP	0.492	0.599	44.26	0.503	8.14
NNPP	0.344	0.427	79.12	0.379	6.23
DeepHawkes	0.315	0.411	71.57	0.326	11.23
DeepCas	0.350	0.476	60.69	0.419	9.14
ChatterNet	0.342	0.455	63.69	0.404	8.77
CasPred (org)	-	-	-	0.231	**0.01**
CasPred (add)	-	-	-	0.300	0.02
`GammaCas-LSTM`	0.597	0.769	35.78	0.688	5.54
`GammaCas-CO`	0.625	0.784	24.16	0.741	1.08
`GammaCas-text`	0.627	0.789	**24.01**	0.742	2.19
`GammaCas`	**0.633**	**0.793**	25.06	**0.744**	5.40

(↓ : lower value is better). CasPred versions do not predict the actual size of future cascades; hence metrics other than step-τ are unapplicable for these two baselines. SEISMIC and TiDeH emerge as the best generative baselines in terms of correlation and MAPE, respectively. `GammaCas` outperforms the rest of the neural network baselines in both metrics. t/s signifies average inference time per sample.

5.1.2 Comparing `GammaCas` With Baselines

From the lowermost block of Table 2, it is evident that `GammaCas` and all its ablation variants perform better than all the baselines by a substantial margin in terms of correlation and absolute error (**18.98%** increase in Kendall's τ from SEISMIC and **35.63** absolute reduction in MAPE compared to DeepCas). In Fig. 3, we plot how the performance of four highly-ranked competing models, namely SEISMIC, TiDeH, DeepCas, and `GammaCas`, are influenced by the actual size of the cascade at 24 hours prediction horizon. The overshooting problem of SEISMIC and TiDeH is evident from these plots as well.

All the ablation variants perform closely to `GammaCas`; the common signal present in all these models is the temporal dynamics of retweet arrival within the observation window. One may trivially decide this to be the most important signal for modeling cascade growth dynamics. However, we can observe significant improvement of correlation measures once we introduce the exogenous influence-modulated signals. Interestingly, the overall MAPE error decreases slightly with some ablated variants. We investigate the influence of tweet content and exogenous signals later in Section 5.4 while diagnosing `GammaCas` predictions.

The design choice we made to introduce extra gating mechanism to LSTM cell to model retweet arrival dynamics evidently brings performance gain. As seen in Table 2,

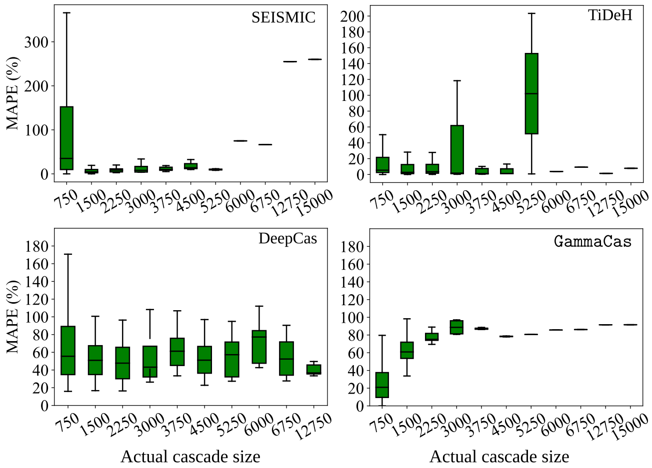


Fig. 3. Variation in performance (MAPE) over different cascade sizes for SEISMIC, TiDeH, DeepCas, and `GammaCas`. We plot the mean, max, min and standard deviations of absolute percentage error at different bins of cascade sizes.

TABLE 3
Comparison of `GammaCas` With Baselines on ActiveRT and Sina-Weibo Datasets

Model	ActiveRT			Sina Weibo		
	τ	ρ	MAPE (%)	τ	ρ	MAPE (%)
Hawkes	0.122	0.253	98.24	0.135	0.264	100.27
SEISMIC	0.341	0.433	101.87	0.358	0.472	103.14
TiDeH	0.293	0.382	78.11	0.305	0.384	74.33
DMHP[1]	0.414	0.562	51.33	-	-	-
DeepHawkes	0.356	0.471	67.82	0.373	0.499	64.26
DeepCas	0.361	0.488	66.49	0.368	0.487	63.88
ChatterNet	0.381	0.479	67.43	0.366	0.492	63.59
`GammaCas`	**0.581**	**0.627**	**34.28**	**0.593**	**0.614**	**33.27**

Due to the absence of exogenous information in ActiveRT and any textual information in Sina-Weibo, `GammaCas` *corresponds to the respective ablated variants.*

`GammaCas`-LSTM (with all signals included) is outperformed `GammaCas` as well as rest of the ablation variants.

We also investigate the latency of prediction for all the models in Table 2. Generative models usually take longer to predict per sample as they use the observation window to estimate the parameters using a likelihood measure. As CasPred solely depends on a manually engineered feature set and does not need any temporal processing (thereby reducing the number of operations), it emerges as the fastest inferring model. Among the rest, `GammaCas` is an order of magnitude faster than the models which show comparable accuracy. Ablated variants with no news-tweet attention or textual features are faster than full `GammaCas`.

As shown in Table 3, `GammaCas` generalizes well to other datasets in the absence of exogenous or content-based signals.

5.2 Variation With Observation Window

As past studies suggested [2], a sufficient amount of early observation is necessary to estimate the future size of a cascade. `GammaCas` offers the flexibility of using different observation windows due to its temporal processing of the input along with an incremental estimation of the growth parameters. In Fig. 4, we show the variation of performance of `GammaCas` for multiple observation windows. Evidently, a larger observation window helps predict the future cascade size with better accuracy. However, even with a shorter observation window (4 hours), `GammaCas` outperforms all the baseline models in terms of correlation and absolute percentage error.

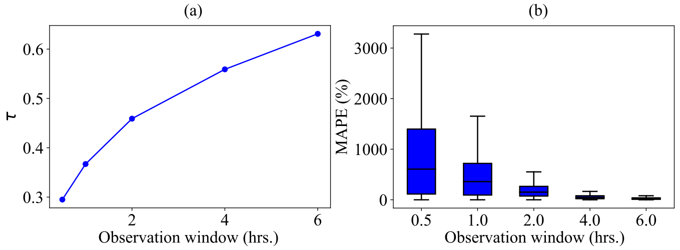


Fig. 4. Variation in performance of `GammaCas` to predict future cascade size at 24 hours prediction horizon with observation window sizes (Δ_{obs}) 30 min., 1 h, 2 hours, 4 hours, and 6 hours. In (a), we show the correlation in terms of Kendall's τ between predicted and actual cascade sizes. In (b), we plot the maximum, minimum and mean values of sample-wise absolute percentage errors along with standard deviation.

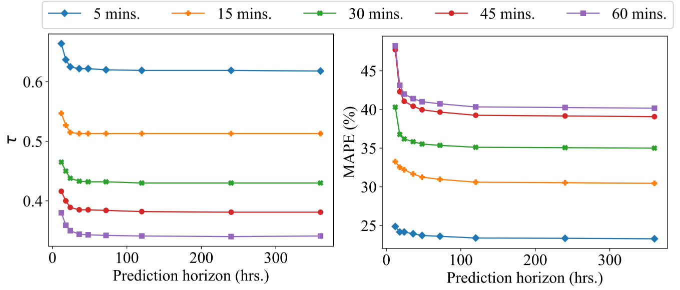


Fig. 5. Variation in performance of `GammaCas` at different prediction horizons (Δ_p) for different widths of observation bins (Δ_o). We evaluate this performance in terms of Kendall's τ and MAPE. With coarser binning (larger Δ_o), the performance drops significantly.

Splitting the cascade dynamics within the observation window into successive bins of retweet arrival and aggregate follower counts serves as a uniform discretization of the irregular arrival processes. Intuitively, a smaller temporal bin width would result in a more accurate approximation of time, leading to superior performance. This is also evident in Fig. 5, where we plot τ (left) and MAPE (right) of `GammaCas` for predicting cascade sizes at different prediction horizons when using different bin widths ($5, 15, 30, 45$ and 60 mins.). While with narrower bins, the performance drop from near to distant prediction horizons is steep, it effectively flattens with the higher error rate in longer bins. However, narrow bins result in a longer sequence of input, resulting in longer recurrence relations to be captured and higher training/testing cost.

5.3 Variation With Prediction Horizons

The quality of fit for the estimated parameters of a monotone function of time is judged by how they fit at different future horizons. We vary the prediction horizon and observe the evaluations for `GammaCas`, its ablation variants, and the best-performing baseline, DeepCas. As shown in Fig. 6, `GammaCas` and its ablation variants produce a more stable performance over different horizons, compared to DeepCas. While in terms of correlation, `GammaCas` shows an initial performance drop as the prediction horizon increases, we can see an almost consistent MAPE over all

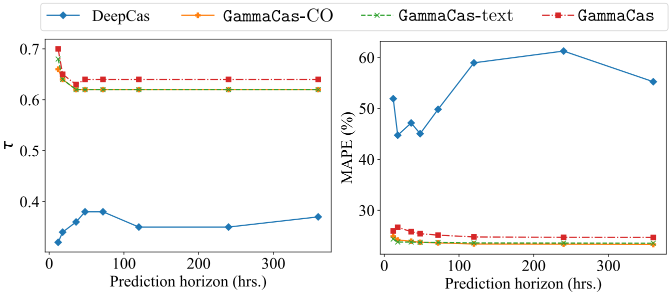


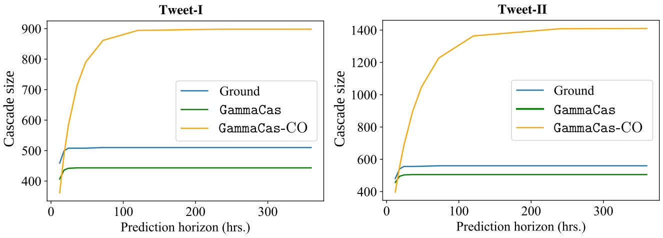
Fig. 6. Variation of performance of `GammaCas`, its ablation variants, and the best performing baseline DeepCas, on different prediction horizons.

the horizons. Moreover, models like DeepCas need to be trained and tested for each prediction horizon separately, while `GammaCas` offers a flexible prediction setting much similar to its generative counterparts, adding significance to the judgment of parameter utility.

5.4 Diagnostic Experiments on `GammaCas`

In Section 3, we provided intuitive justifications of our design decisions. To look for the potential presence of more profound connections between different influencing signals and the cascade growth parameters that `GammaCas` attempts to model, we look into individual predictions as well as the overall distribution of parameters.

In Fig. 7, we present two example tweets, actual sizes of the cascades they generate, and the predicted sizes by `GammaCas` and `GammaCas`-CO over different prediction horizons. While *Tweet-I* was from a popular social media influencer addressing their fan-base (no exogenous influence), *Tweet-II* was regarding a teacher passing abusive remarks towards students in the context of COVID-19 (triggered by exogenous event). In both cases, `GammaCas`-CO, in the absence of content-based signals, underestimates A and to fit the observed retweet arrivals, underestimates the decay parameter λ as well. This leads to overshooting the actual cascade size by a large margin. The low value of λ also sets a longer supercritical phase of the cascades. On the other hand, `GammaCas` estimates a much higher value of A with larger λ decay, providing a better approximation of the future cascade size.



GammaCas-CO: $A = 0.040$, $\lambda = 0.086$, $\gamma = 0.715$
GammaCas: $A = 63.77$, $\lambda = 0.308$, $\gamma = 0.725$

GammaCas-CO: $A = 0.041$, $\lambda = 0.087$, $\gamma = 0.715$
GammaCas: $A = 66.01$, $\lambda = 0.295$, $\gamma = 0.740$

Tweet-I: I will be doing a major giveaway soon. Keep your notifications ON. I am going to send a surprise to many of you

Tweet-II: Why I homeschool my kids Who vets these lunatics allowed to teach? No one.

Fig. 7. Predicted and actual cascade sizes for two tweets by `GammaCas` and `GammaCas`-CO. Underlined words in the tweets are those attaining higher attention weights. In both the cases, `GammaCas`-CO meets a very low value of λ entangled with a low value of A, which leads to overshooting the cascade size.

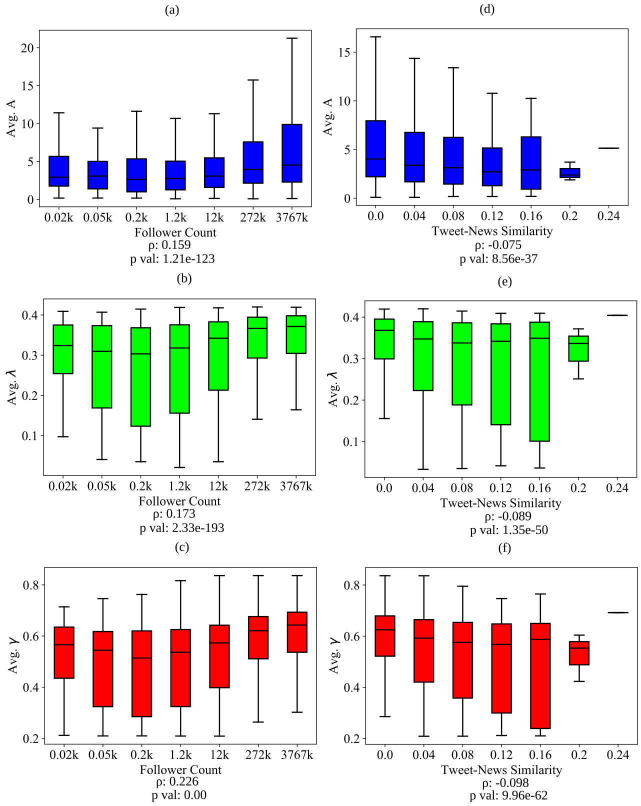


Fig. 8. Variations of A, λ, and γ estimated by `GammaCas` with follower count of the root user and news-tweet similarity. We plot the max, min, mean and standard deviation of the parameters for tweets at different bins of follower count/news-tweet similarity. We also show the correlations between each pair of variables in terms of Spearman's ρ and the corresponding p-value.

We extract the attention values α_i (see Equation (12)) for each token (other than stopwords) of the tweets. In Fig. 7, we mark the words receiving significant attention. It is evident that certain topic-signaling and positive/negative sentiment words put a higher contribution constituting the signals deciding cascade growth.

To investigate the effects of follower count of the root users and exogenous influence on the cascade growth parameters estimated by `GammaCas`, we plot one-to-one mappings between them in Fig. 8. We compute the correlations between each pair of variables to find out their statistical significance. Evidently, the follower count of the root user holds a strong influence on all of the three parameters (subplots (a), (b), and (c) in Fig. 8). However, the growth parameter γ is the most positively correlated one. Intuitively, one can translate this as high follower count ensures an influential user with a high degree of organic reach; when such a user tweets something, the rate of growth at the supercritical stage is likely to be higher compared to some less influential user. Alternatively, if the root user of the cascade reaches a large number of users directly, the subsequent levels are likely to have a lower value of average out-degree and thereby, decreasing the rate of subsequent cascade growth. This points to the high value of the decay parameter λ as well. Lastly, users become influential with historical activity, i.e., the degree of diffusion of contents posted by them are usually high, pointing towards a possible positive reinforcement of A in the future cascades they cause.

As opposed to the follower count, similarity of a tweet with news articles published in the past 6 hrs. shows a weakly negative (yet statistically significant) correlation with all three of the parameters. In this case, the effect is strongest in the case of both γ and λ, pointing towards a slow growth as well as decay when the similarity is high, and vice versa. This weakly negative correlation is consistent with our findings shown in Fig. 1b in Section 1, where we observed a similar weakly negative impact of similarity between a tweet and past news on the cascade size. Again, a plausible intuition behind this might be that the potential of a tweet be the genesis of a large cascade is facilitated if it brings new, hitherto unknown information.

6 CONCLUSION AND FUTURE WORK

We presented `GammaCas`, a new deep cascade prediction architecture that combines content, network, and exogenous signals into a transparent, parameterized time integral. Prediction loss can be back-propagated to the feature-processing networks. We prepared a large-scale dataset of retweet cascades and time-aligned news texts, and provided insightful findings on the dynamics of cascade growth. `GammaCas` provides a better and more robust cascade size prediction compared to recent competitive baselines with added flexibility of prediction horizons. Investigations on parametric functions and feature representations learned by `GammaCas` provide a meaningful interpretation of relations between cascade dynamics and various input features.

Since `GammaCas` uses only the out-degree of cascade participants, one does not need to provide the complete network information. However, the exact time-stamp of cascade participation is still required. In case of Twitter-like networks, such information can be acquired easily. This may not be the case for cascades appearing over platforms with more loose structures (i.e., Reddit, Web, etc.). Similar to the existing cascade models, `GammaCas` depends on the observed dynamics and might fail when drastic events affect the cascade growth beyond the observation window.

As a future extension, one may intend to introduce multimodal signals introduced by richer metadata of the tweet (images, memes, videos, etc.). Information cascades formed from a tweet are not limited to simple retweet trees as well. For example, link to an existing tweet may be posted as standalone tweets. When such a tweet gets retweeted, this practically forms an extended information cascade of the original tweet. These complex dynamics makes the cascade modeling problem intrinsically challenging. Modeling such dynamics using the various signals we used is likely to provide further insights.

ACKNOWLEDGMENTS

T. Chakraborty would like to acknowledge the support of the Ramanujan Fellowship, and ihub-Anubhuti-iiitd Foundation set up under the NM-ICPS scheme of the Department of Science and Technology, India.

REFERENCES

[1] M. J. Salganik, P. S. Dodds, and D. J. Watts, "Experimental study of inequality and unpredictability in an artificial cultural market," *Science*, vol. 311, no. 5762, pp. 854–856, 2006.

[2] J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec, "Can cascades be predicted?," in *Proc. 23rd Int. Conf. World Wide Web*, 2014, pp. 925–936.

[3] D. J. Watts, *Everything is Obvious: How Common Sense Fails Us*. New York, NY, USA: Random House LLC, 2012.

[4] J. M. Hofman, A. Sharma, and D. J. Watts, "Prediction and explanation in social systems," *Science*, vol. 355, no. 6324, pp. 486–488, 2017.

[5] K. Wang, M. Bansal, and J.-M. Frahm, "Retweet wars: Tweet popularity prediction via dynamic multimodal regression," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2018, pp. 1842–1851.

[6] Z. Zhao et al., "Attentional image retweet modeling via multi-faceted ranking network learning," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 3184–3190.

[7] R. Kobayashi and R. Lambiotte, "TiDeH: Time-dependent hawkes process for predicting retweet dynamics," in *Proc. 10th Int. AAAI Conf. Web Soc. Media*, 2016, pp. 191–200.

[8] M.-A. Rizoiu, L. Xie, S. Sanner, M. Cebrian, H. Yu, and P. Van Hentenryck, "Expecting to be HIP: Hawkes intensity processes for social media popularity," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 735–744.

[9] J. Jia and A. R. Benson, "Neural jump stochastic differential equations," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 9847–9858.

[10] S. Dutta, S. Masud, S. Chakrabarti, and T. Chakraborty, "Deep exogenous and endogenous influence combination for social chatter intensity prediction," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 1999–2008.

[11] C. Li, J. Ma, X. Guo, and Q. Mei, "DeepCas: An end-to-end predictor of information cascades," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 577–586.

[12] Q. Cao, H. Shen, K. Cen, W. Ouyang, and X. Cheng, "DeepHawkes: Bridging the gap between prediction and understanding of information cascades," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2017, pp. 1149–1158.

[13] S. Mishra, M.-A. Rizoiu, and L. Xie, "Feature driven and point process approaches for popularity prediction," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2016, pp. 1069–1078.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] J. Milbrandt and M. Yau, "A multimoment bulk microphysics parameterization. Part I: Analysis of the role of the spectral shape parameter," *J. Atmospheric Sci.*, vol. 62, no. 9, pp. 3051–3064, 2005.

[16] A. Seifert and K. D. Beheng, "A two-moment cloud microphysics parameterization for mixed-phase clouds. Part 1: Model description," *Meteorol. Atmospheric Phys.*, vol. 92, no. 1/2, pp. 45–66, 2006.

[17] 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 *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1513–1522.

[18] T. Omi, N. Ueda, and K. Aihara, "Fully neural network based model for general temporal point processes," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 2122–2132.

[19] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 561–568.

[20] J. Wang, V. W. Zheng, Z. Liu, and K. C. Chang, "Topological recurrent neural network for diffusion prediction," in *Proc. IEEE Int. Conf. Data Mining*, 2017, pp. 475–484.

[21] M. R. Islam, S. Muthiah, B. Adhikari, B. A. Prakash, and N. Ramakrishnan, "DeepDiffuse: Predicting the 'who' and 'when' in cascades," in *Proc. IEEE Int. Conf. Data Mining*, 2018, pp. 1055–1060.

[22] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: Quantifying influence on Twitter," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 65–74.

[23] S. Krishnan, P. Butler, R. Tandon, J. Leskovec, and N. Ramakrishnan, "Seeing the forest for the trees: New approaches to forecasting cascades," in *Proc. 8th ACM Conf. Web Sci.*, 2016, pp. 249–258.

[24] S. Petrovic, M. Osborne, and V. Lavrenko, "RT to win! Predicting message propagation in Twitter," in *Proc. 5th Int. AAAI Conf. Web Soc. Media*, 2011, pp. 586–589.

[25] L. Weng, F. Menczer, and Y. Ahn, "Predicting successful memes using network and community structure," in *Proc. Int. AAAI Conf. Web Soc. Media*, 2014, pp. 535–544.

[26] T. Zaman et al., "A Bayesian approach for predicting the popularity of tweets," *Ann. Appl. Statist.*, vol. 8, no. 3, pp. 1583–1611, 2014.

[27] A. Kupavskii et al., "Prediction of retweet cascade size over time," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2012, pp. 2335–2338.

[28] Q. Kong, W. Mao, G. Chen, and D. Zeng, "Exploring trends and patterns of popularity stage evolution in social media," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 10, pp. 3817–3827, Oct. 2020.

[29] M. Lukasik, T. Cohn, and K. Bontcheva, "Point process modelling of rumour dynamics in social media," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, 2015, pp. 518–523.

[30] R. Crane and D. Sornette, "Robust dynamic classes revealed by measuring the response function of a social system," *Proc. Nat. Acad. Sci. USA*, vol. 105, no. 41, pp. 15 649–15 653, 2008.

[31] P. Bao, H.-W. Shen, X. Jin, and X.-Q. Cheng, "Modeling and predicting popularity dynamics of microblogs using self-excited hawkes processes," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 9–10.

[32] Q. Kong, M.-A. Rizoiu, and L. Xie, "Describing and predicting online items with reshare cascades via dual mixture self-exciting processes," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2020, pp. 645–654.

[33] M.-A. Rizoiu, S. Mishra, Q. Kong, M. Carman, and L. Xie, "SIR-hawkes: Linking epidemic models and hawkes processes to model diffusions in finite populations," in *Proc. Int. Conf. World Wide Web*, 2018, pp. 419–428.

[34] H. Mei and J. Eisner, "The neural hawkes process: A neurally self-modulating multivariate point process," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6754–6764.

[35] H. Li, H. Li, and S. S. Bhowmick, "CHASSIS: Conformity meets online information diffusion," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2020, pp. 1829–1840.

[36] Q. Wang, Z. Lin, Y. Jin, S. Cheng, and T. Yang, "ESIS: Emotion-based spreader–ignorant–stifler model for information diffusion," *Knowl.-Based Syst.*, vol. 81, pp. 46–55, 2015.

[37] Q. Kong, M.-A. Rizoiu, and L. Xie, "Modeling information cascades with self-exciting processes via generalized epidemic models," in *Proc. 13th ACM Int. Conf. Web Search Data Mining*, 2020, pp. 286–294.

[38] Y. Yan, Z. Tan, X. Gao, S. Tang, and G. Chen, "STH-bass: A spatial-temporal heterogeneous bass model to predict single-tweet popularity," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, 2016, pp. 18–32.

[39] X. Gao, Z. Zheng, Q. Chu, S. Tang, G. Chen, and Q. Deng, "Popularity prediction for single tweet based on heterogeneous bass model," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 5, pp. 2165–2178, May 2021.

[40] J. G. Lee, S. Moon, and K. Salamatian, "An approach to model and predict the popularity of online contents with explanatory factors," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, 2010, pp. 623–630.

[41] X. Gao, X. Jia, C. Yang, and G. Chen, "Using survival theory in early pattern detection for viral cascades," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 5, pp. 2497–2511, May 2022.

[42] Y. Wang, H. Shen, S. Liu, J. Gao, and X. Cheng, "Cascade dynamics modeling with attention-based recurrent neural network," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 2985–2991.

[43] C. Yang, M. Sun, H. Liu, S. Han, Z. Liu, and H. Luan, "Neural diffusion model for microscopic cascade prediction," 2018, *arXiv:1812.08933*.

[44] Q. Cao, H. Shen, J. Gao, B. Wei, and X. Cheng, "Popularity prediction on social platforms with coupled graph neural networks," in *Proc. 13th ACM Int. Conf. Web Search Data Mining*, 2020, pp. 70–78.

[45] S. A. Myers, C. Zhu, and J. Leskovec, "Information diffusion and external influence in networks," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2012, pp. 33–41.

[46] S. Koyama and S. Shinomoto, "Statistical physics of discovering exogenous and endogenous factors in a chain of events," *Phys. Rev. Res.*, vol. 2, no. 4, p. 043358, 2020.

[47] A. De, S. Bhattacharya, and N. Ganguly, "Demarcating endogenous and exogenous opinion diffusion process on social networks," in *Proc. World Wide Web Conf.*, 2018, pp. 549–558.

[48] T. Broxton, Y. Interian, J. Vaver, and M. Wattenhofer, "Catching a viral video," *J. Intell. Inf. Syst.*, vol. 40, no. 2, pp. 241–259, 2013.

[49] S. D. Roy, T. Mei, W. Zeng, and S. Li, "Towards cross-domain learning for social video popularity prediction," *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1255–1267, Oct. 2013.

[50] E. Veling, "The generalized incomplete gamma function as sum over modified bessel functions of the first kind," *J. Comput. Appl. Math.*, vol. 235, no. 14, pp. 4107–4116, 2011.

[51] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[52] A. Wehenkel and G. Louppe, "Unconstrained monotonic neural networks," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 1545–1555.

Subhabrata Dutta is currently working toward the PhD degree with the Department of Computer Science and Engineering, Jadavpur University. His research interests include social computing, natural language processing, and machine learning.

Shravika Mittal received the undergraduate degree in computer science and engineering from IIIT-Delhi. She is a software development engineer with Adobe, India. Her research interests include social network analysis, network science, and natural language processing. She received the Chancellor's gold medal, Dean's list for Innovation in Research and Development as an undergraduate student.

Dipankar Das (Member, IEEE) is an assistant professor with the Department of Computer Science and Engineering, Jadavpur University and Visveswaraya Young Faculty, Ministry of Electronics and Information Technology (MeitY), Government of India. He is currently leading four research projects of DRDO, SERB, DST, UGC, Govt. of India. His research interests are in the area of natural language processing, emotion and sentiment analysis, information extraction, machine learning, deep learning, social network analysis and so on. He is also an active member of the ACL, HUMAINE groups.

Soumen Chakrabarti received the PhD degree from the University of California, Berkeley and worked with IBM Almaden, CMU and Google in the past. He is currently a professor of computer science with IIT Bombay. He works on knowledge graphs, question answering, and social networks. He has published extensively in WWW, SIGKDD, ACL, EMNLP, IJCAI, AAAI, VLDB, SIGIR, ICDE and other conferences. He won the Best Paper Award with WWW 1999. He was coauthor on the best student paper with ECML 2008. His work on keyword search in databases got the 10-year influential paper award with ICDE 2012. He received the Bhatnagar Prize, in 2014 and the Jagadis Bose Fellowship, in 2019.

Tanmoy Chakraborty (Senior Member, IEEE) is an assistant professor of computer science and a Ramanujan fellow with IIIT-Delhi, India, where he leads a research group, Laboratory for Computational Social Systems (LCS2). His primary research interests include social computing and natural language processing. He has received several awards/fellowships including Faculty Awards from Google, IBM and LinkedIn; Early Career Research Award, DAAD Faculty Fellowship. He is a member of ACM. More details at http://faculty.iiitd.ac.in/~tanmoy/.