International Journal of Data Science and Analytics
https://doi.org/10.1007/s41060-018-0121-2

REGULAR PAPER

Easy-Mention: a model-driven mention recommendation heuristic to boost your tweet popularity

Soumajit Pramanik[1] · Mohit Sharma[1] · Maximilien Danisch[2] · Qinna Wang[2] · Jean-Loup Guillaume[3] · Bivas Mitra[1]

Received: 24 August 2017 / Accepted: 29 March 2018

Abstract
This paper investigates the role of mentions on tweet propagation. We propose a novel tweet propagation model SIR_{MF} based on a multiplex network framework which allows to analyze the effects of mentioning on final retweet count. The basic bricks of this model are supported by a comprehensive study of multiple real datasets, and simulations of the model show a nice agreement with the empirically observed tweet popularity. Studies and experiments also reveal that follower count, retweet rate and profile similarity are important factors for gaining tweet popularity and allow to better understand the impact of the mention strategies on the retweet count. Interestingly, we experimentally identify a critical retweet rate regulating the role of mention on the tweet popularity. Finally, our data-driven simulations demonstrate that the proposed mention recommendation heuristic *Easy-Mention* outperforms the benchmark *Whom-To-Mention* algorithm.

Keywords Mention recommendation · Multiplex networks · Information diffusion

1 Introduction

In recent times, Twitter has become one of the most influential microblogging systems for spreading and sharing breaking news, personal updates and spontaneous ideas [19]. However, it is observed that the popularity of tweets and hashtags follows a skewed distribution in any unbiased collection of tweets: only a small set of the tweets (or hashtags) are heavily popular [29]. In Twitter, propagation of a tweet or hashtag from one user to another occurs mainly via two activities: 'retweeting' and 'mentioning' [26]. In case of retweet, information is simply relayed to all the followers of the retweeting user. However, mention utility allows to spread an information far beyond the neighborhood and improve its visibility by making it available to the appropriate set of users. Moreover, as mentions get listed in a separate tab, they gain higher attention than regular posts. Admittedly, mention utility has a potential to play a significant role in the cascading behavior of tweets and hashtags. For instance, in our dataset, we observe that the probability that a mentioned user retweets a post is on average 32% higher than the one of a follower. Hence, investigating the role of mention utility behind popularizing a tweet is an interesting research question.

The problem of popularizing a tweet has two opposite facets. On one hand, it is important to realize that artificially boosting popularity may immediately lead to spamming behavior [14]. Moreover, public mentions and direct message features have been exploited a lot by spammers for spamming hyperlinks and irrelevant content. Automatic mentioning

This paper is an extension version of the DSAA'2016 paper "On the Role of Mentions on Tweet Virality" [40].

✉ Soumajit Pramanik
soumajit.pramanik@gmail.com;
soumajit.pramanik@cse.iitkgp.ernet.in

Mohit Sharma
mohit.sharma@cse.iitkgp.ernet.in

Maximilien Danisch
maximilien.danisch@gmail.com

Qinna Wang
qinna.wang@gmail.com

Jean-Loup Guillaume
jean-loup.guillaume@univ-lr.fr

Bivas Mitra
bivas@cse.iitkgp.ernet.in

[1] Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, India

[2] Sorbonne Universités, UPMC Univ Paris 06, CNRS, LIP6 UMR 7606, 4 Place Jussieu, 75005 Paris, France

[3] L3I, University of La Rochelle, La Rochelle, France

Published online: 27 April 2018

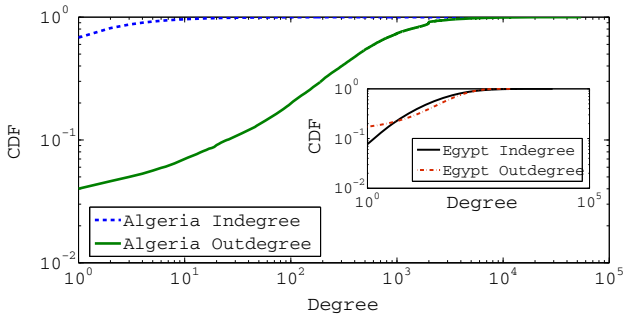

Fig. 1 Indegree (follower count) and outdegree (followee count) distributions for 'Algeria' and 'Egypt' datasets

through bots may further compound the problem and surely lead to annoyance. Hence, any attempt toward popularizing a tweet should be ready to deal with the possible mistreatment by the spammers. On the other hand, follower distribution (see Fig. 1) exhibits the fact that most of the normal Twitter users only have a low to moderate number of followers [29]. Hence, any useful information, produced by a normal and trustworthy user, reaches only to a small population.

Several studies have been carried out for understanding the dynamics behind popularity of tweets; details are discussed in Sect. 2.2 [7,17,27,36,39,41,44]. Considering Twitter as an open arena for 'viral marketing,' researchers developed a deck of influence models [9,11] to identify the influential nodes in a network. Subsequently, considering *mentioned user* as an information broker, the influence models have been further explored by Kempe et al. [27]; Saito et al. [41]; Gomez-Rodriguez et al. [17]. Notably, success of the aforesaid models in spreading information depends on the propensity to retweet by the mentioned users. However, predicting retweeting probability of the mentioned user is not trivial; rather, it depends on multiple latent factors including information content of the post and profile of the tweeting user, which are not considered while computing the influence. This motivates the community to develop mention recommendation algorithms for identifying suitable users to mention. Nevertheless, there are several challenges pertaining to this task; first, there is a huge number of active users in microblogging services. This means that the recommendation target space is extremely large. Second, personalization is an inherent requirement of this task. Since users have their specific preferences, the set of recommended users should largely vary across users, even if their microblog posts have the exactly same content. Third, due to the limitation of character length in microblog posts, there is brevity in information and inadequacy in context and structure. However, several recommendation systems have been proposed which aimed to deal with this aforementioned challenges. For instance, Wang et al. [45] proposed *Whom-To-Mention* heuristic that uses features (such as user interest match, content dependent user relationship and user influence) and trains a machine learned ranking function to extract the best users to mention. Similar recommendation heuristics can be found in [18,30,34,43,50].

Notably the aforementioned state-of-the-art endeavors suffer from several limitations. In [45], the relevance function remains unchanged for different tweet messages, leading to same recommended ranked list for different tweets. Moreover, most of these heuristics rely on a large set of features to be calculated on a large population which is infeasible in real time; hence, those approaches cannot be used to design an online mention recommendation system. More importantly, all these works fail to shed light on the interplay between the factors involved in the propagation of the tweets. For instance, it is not clear how exactly the mentioned user can make the tweet popular; does mentioning somebody in a tweet of her interest really helps in gaining popularity; how does the users' activity (say retweet) rate influence the choice of the mention strategy? In order to address these questions, a simple model to mimic the tweet cascading process is necessary. This model can guide one to identify the role of individual factors on the tweet propagation and lead to the development of a simple recommendation system which may recommend users to mention. This paper takes an important step toward this direction.

In this paper, we dissect the impact of *mentioning* on tweet popularity. Section 2 summarizes the relevant literature. In Sect. 3, we introduce the datasets and describe the multiplex representation of the tweet propagation process [20]. Next, we perform a comprehensive data study to motivate the importance of mention utility on the popularity of a tweet. This study enables us to identify the important features of the mentioned users contributing to tweet popularity (Sect. 4). Motivated by the experiments, in Sect. 5, we propose a framework SIR_{MF} to model the flow of tweets. We introduce a parametric mention strategy to model the mentioning behavior of users. Simulations of SIR_{MF} model with suitable parameters show a nice agreement with the empirical tweet popularity observed in the dataset (Sect. 5). Moreover, the proposed model highlights the elegance of smart mention strategy, pointing to the potential of a mention recommendation heuristic to boost tweet popularity (Sect. 6). The role of different model parameters on the retweet count reveals the presence of a phase transition in cascade formation (Sect. 7). The detailed analysis of the SIR_{MF} model shows that the following three factors play major roles in maximizing the retweet count—(a) retweet rate of the mentioned user, (b) content similarity between the posted tweet and the profile of the mentioned user (c) follower count of the mentioned user. Finally, taking cues from this model, we develop *Easy-Mention*, a simple mention recommendation heuristic which computes a score to rank the potential users to be mentioned in a tweet to boost its popularity (Sect. 8).

We perform rigorous experiments to show that *Easy-Mention* heuristic outperforms the state-of- the-art *Whom-To-Mention* baseline algorithm [45] (Sect. 9).

2 Related works

The state-of-the-art literature in this area can be summarized in three different segments—(a) modeling information diffusion via retweets, (b) various attempts to analyze and boost popularity of tweets and (c) recent endeavors incorporating mentions in tweets. The detail is as follows.

2.1 Information propagation in Twitter

Diffusion on social network classically involves the following two propagation models—independent cascade [16] and linear threshold [21]. Independent cascade model associates a fixed spreading probability per graph edge and allows each node to attempt infecting another node only once. On the other hand, the linear threshold model associates a threshold with each node; a node gets infected if the number of infected neighbors exceeds that threshold. Further studies [13,15] have generalized these models. In continuation, Kwak et al [28] treated retweet trees as communication channels of information diffusion and analyzed the tweets of top trending topics, whereas Lerman and Ghosh [31] studied the distribution of retweet cascades in Twitter. Side by side, popular epidemic like models such as SIS (Susceptible-Infected-Susceptible), SIR (Susceptible-Infected-Recovered) are also explored to model information contagion in Twitter [1,24,35]. This type of models allows individuals to have the flexibility of cyclically changing their dynamical states based on whether they are exposed to the information, have actively participated in the spreading process or are immune to it.

2.2 Analyzing and boosting tweet popularity

With the advent of text mining, research in the domain of information propagation started progressing in two clearly distinguishable tracks. On one hand, several studies have been carried out for understanding the dynamics behind the popularity of tweets. For example, Suh et al. [42] used a generalized linear model to understand what features influence the chance of a tweet being retweeted by anyone. In similar line, in Petrovic et al. [39] and Malhotra et al. [36], researchers investigated the role of content and contextual features of tweets and identified factors that are significantly associated with retweet rate and tweet popularity. Additionally, few [3,48] tried to analyze the problem at individual level and predict the existence of a retweet between a particular pair of users. On the other hand, several studies have been made on influence models [2,11] and different recommendation systems have been proposed. For example, Uysal and Croft [44] proposed methods to recommend useful tweets that users are really interested in and more likely to retweet: given a tweet, they rank users based on retweet probability. Importantly, Cha et al. [10] revealed that follower count is not necessarily the best metric to measure the influence. Subsequently, considering the influential users in Twitter as potential information brokers, researchers proposed models to identify them and maximize the information propagation [9,11]. Notably, all the aforementioned models consider retweets as the only mode of tweet propagation.

2.3 Mentioning activities in Twitter

Mentioning is mainly considered as a medium of attracting the attention of influential people regarding a tweet so that the popularity of the corresponding content increases. However, mentioning one influential user in a tweet does not ensure that she reposts it. This later part depends on several factors including information content of the post and profile of the tweeting user. Standard influence models in general do not capture this type of properties while computing the influence of individual users. This motivates the community for the development of mention recommendation algorithms to identify the suitable users to mention. For instance, [45] formulated the task as a learning-to-rank problem and proposed *Whom-To-Mention* heuristic that uses features (such as user interest match, content- dependent user relationship and user influence) and trains a machine learned ranking function to extract the best users to mention. Similar recommendation heuristics can be found in [43,50]. However, instead of being limited to maximizing the spread of a microblog, Gong et al. [18] proposed a novel topical translation-based method to predict the users whom the authors try to mention. Most of these recommendation heuristics are based on empirical observations and lack comprehensive perception of the role of mentions on tweet diffusion.

3 Dataset and representation

In this section, we introduce the datasets and describe the way we represent the flow of information via follow and mention links using a multiplex network framework.

3.1 Dataset

We collect the tweets posted during two particular real-life events—(a) Arab-Spring Movement 2011 and (b) World Cup Football 2014. In both events, Twitter was used extensively to propagate news and opinions; however, the domains, locations and time spans of these two events are very different;

hence, tweet propagation processes in both events are completely independent. We may therefore assume that observed behaviors and results may hold more generally in Twitter.

(a) *Arab-spring dataset* We collected the following two publicly available datasets [23] connected to these events—(i) 'Algeria' dataset is a collection of around 60K tweets (tweet IDs) and 20K users who posted them during the 'Algeria movement.' (ii) 'Egypt' dataset is a collection of around 0.2 million tweets (tweet IDs) posted during 'Egypt uprising' by around 60K users. In both the datasets, we crawled the corresponding tweet contents, user profiles and follower network using the Twitter API.[1] Twitter provides the 'GET statuses/show/:id' functionality for crawling the tweet content and the corresponding author details against each tweet ID. Similarly, the 'GET followers/id' functionality allows us to obtain the list (user IDs) of followers for a specific user (5000 follower IDs per request). Each API probe returns the required response within a constant time, and one can issue at most 900 such tweet content requests and 15 such follower requests, respectively, within every 15 minutes interval.[2]

(b) *World Cup Football dataset* This dataset[3] consists of all tweets (2.8 million tweets) which were posted during the soccer World Cup 2014 and contain official team hashtags (#BRA, #CRO, etc.) or match hashtags (#BRACRO, #MEXCMR, etc.).

3.2 Multiplex network representation

For a given hashtag '#h,' the multiplex network representation contains two layers: the bottom one represents tweet propagation via follow links, and the top one via mention links (Fig. 2). More precisely, all users who tweet '#h' appear as a node in the bottom (follow) layer. A directed link connects user 'A' to 'B' if 'A' (re)tweets '#h' before 'B' further retweets and 'B' is a follower of 'A.' In the top (mention) layer, a directed link connects 'C' to 'D' if 'C' tweets '#h' before 'D' further retweets and 'C' *mentions* 'D' in her post ('D' may or may not be a follower of 'C'). One user is free to appear in both the layers.

A closer look reveals that both the layers are essentially collections of directed acyclic graphs (DAGs).We denote the root of each DAG as an initiator since they are responsible for initiating the spreading process. We can identify two classes of initiators, the 'true initiators' and the 'dummy initiators.' A *true initiator* of '#h' is a user who is the root in a follow or a mention DAG but never appears as non-root member

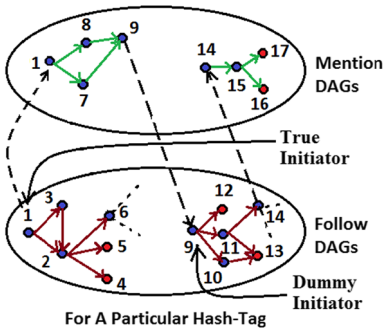

Fig. 2 Example of a mention–follow multiplex network

of any DAG. These users have actually started the spreading process (for '#h') as a result of some external influences. A *dummy initiator* is a user who is the root of a follow DAG but a non-root member of a mention DAG. Basically, a dummy initiator gets the information from someone else via mention and subsequently initiates the spreading process to its followers.

4 Analysis of mentioning activities

In this section, we first establish the importance of mention utility on the spread of tweets. Subsequently, we perform few data study experiments which enable us to identify the key features of the mentioned users and provide a general guideline for choosing the right users to mention for boosting tweet popularity.

4.1 Importance of mention

Given this multiplex network representation, we measure the impact of the mentioned users on the popularity of hashtags. Let us define the popularity of a hashtag as the number of (re)tweets it receives. We select few popular hashtags for which we estimate the popularity reduction by dropping mentions. In this estimation, first we find the dummy initiators (set D) for a hashtag '#h' and all the users (set S) who only belong to the DAGs rooted by dummy initiators. Obviously, the retweet activity of the $S \cup D$ users is dependent on the mention layer. If hashtag '#h' is tweeted by total n users, then mention dependency of '#h' can be measured as $\frac{(|S \cup D|)}{n}$. Looking at the most popular hashtags (tweets) in Fig. 3a, b, we observe that such hashtags (tweets) are heavily mention dependent.

4.2 Properties of mentioned users

Next we turn our attention to the node-level properties of the mentioned users. This may provide us some guideline to select proper users to mention for boosting tweet popularity.

[1] http://apps.twitter.com/.

[2] http://developer.twitter.com/en/docs/basics/rate-limits.

[3] We received it from the 'linkfluence' company (http://linkfluence.com/en/).

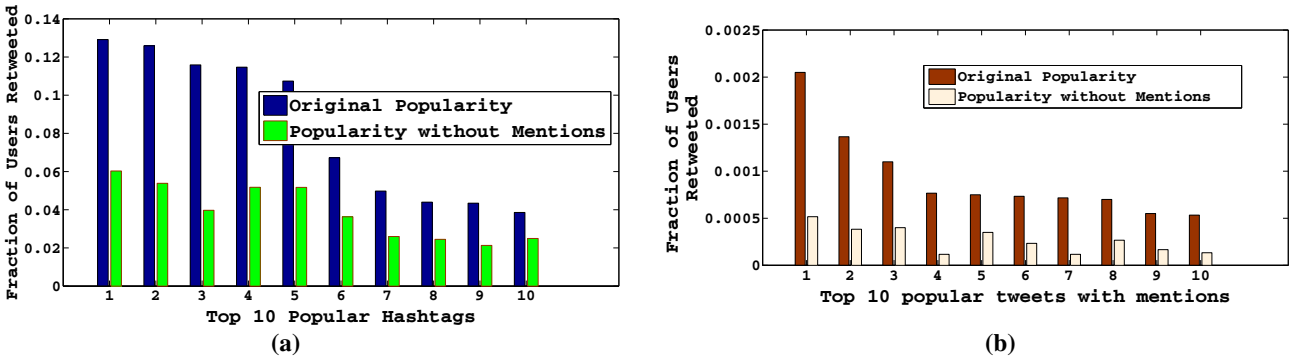

Fig. 3 Mention dependency for tweets and hashtags in 'Algeria' and 'Egypt' datasets. **a** Popularities (number of times posted) of top 10 popular hashtags in 'Algeria' dataset with and without Mentions. **b** Popularities (retweet counts) of top 10 popular tweets (containing mentions) in 'Egypt' dataset with and without Mentions

(a) *Impact of popularity and retweet activity* In order to confirm whether people like to mention popular users, we plot the probabilities of mentioning users with different follower counts (see Fig. 4a). The plot clearly depicts that a significant fraction of users mention popular people. On the other hand, Fig. 4b shows that the probability of getting a retweet from a mentioned user reduces sharply if her follower count is over 1000 (celebrities are choosy in retweeting).This clearly demonstrates that two opposite forces play roles in tweet propagation through mentions; highly popular users are less likely to retweet, but they provide high exposure when they retweet. In order to measure the combined effect of user popularity and retweet rate, we introduce *visibility*, which is the product of follower count and retweet probability of a mentioned user, and plot the expected visibility distribution in Fig. 4c. The peak of the curve demonstrates the existence of a balance between popularity and retweet rate, while mentioning some user.

(b) *Impact of content similarity* Content similarity between the profile of the mentioned user and the posted tweet is another factor which determines the propensity of retweeting. We compute the expectation that the mentioned user retweets in the 'World Cup' dataset (see Fig. 5), (a) if the tweet contains at least one hashtag that she has already posted (expected probability to retweet 0.029) and (b) if the tweet does not contain any hashtag which she has already posted (expected probability to retweet 0.017). Hence if the mentioned user has already posted the hashtag, her probability to retweet becomes almost twice. Moreover, Fig. 5 also reveals that this fact is independent of the follower count of the mentioned user.

(c) *Impact of the number of mentions* Mentioning the correct number of users is important to gain a high number of retweets. In the 'Egypt' dataset, we observe that 23.9% of all tweets in our dataset contain mentions. Out of them, 80.5% of the tweets contain only one mention, 14.7% contain two, 3.2% contain three, and the remaining 1.6% contain more than three. We also observe similar statistics in the 'Algeria' and 'World Cup' datasets. Figure 6 highlights the fact that mentioning few (say 2–3) intended users is always beneficial in gaining retweets; mentioning too many people makes the tweet content short and probably less interesting. Confirm-

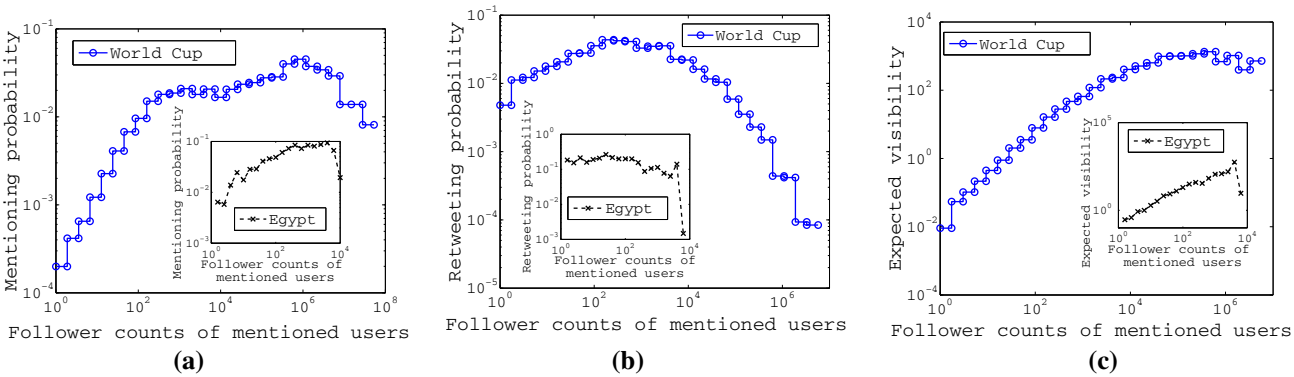

Fig. 4 Users' tendency and reasons to mention popular users in 'World Cup' and 'Egypt' datasets. **a** Probability of mentioning popular users. **b** Retweeting probability of mentioned users. **c** Expected visibility

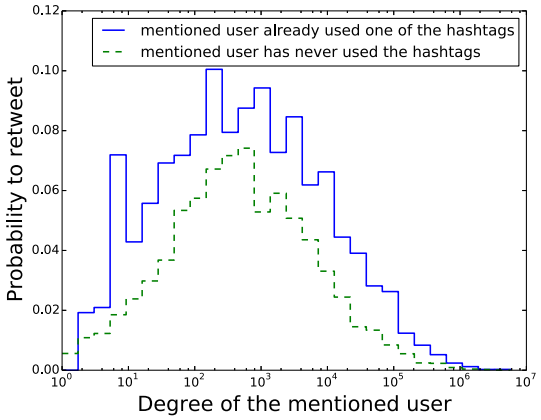

Fig. 5 Probability of retweeting for a mentioned user based on content similarity in 'World Cup' dataset

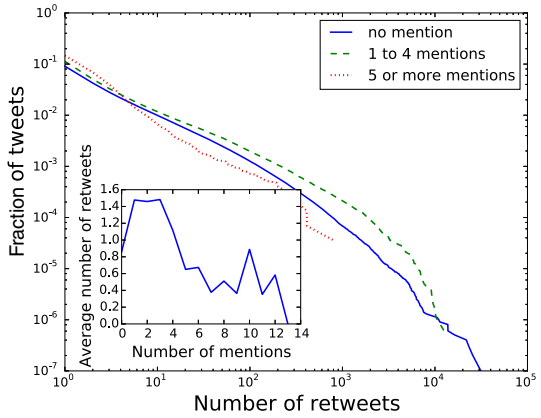

Fig. 6 The distribution of retweet counts of tweets containing different numbers of mentions in 'World Cup' dataset. The inset shows how the average retweet count changes with number of mentions in the tweets

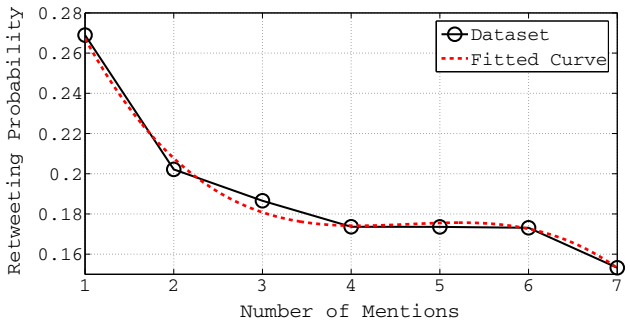

Fig. 7 Dependency of retweet rate from mention on number of mentions in 'World Cup' dataset; fitting a polynomial curve

ing the same, in Fig. 7, we observe that a mentioned user's propensity of retweeting a tweet reduces sharply if it contains higher number of mentions.

5 Simulation model

In this section, we propose a model to demonstrate the tweet propagation dynamics in an epidemiological framework [38]. The objective of the model is to closely analyze the role of mentioning on tweet popularity. We introduce a parametric mention strategy to replicate real mentioning behavior. Finally, we illustrate the simulation setup and validate the model in light of empirical dataset.

5.1 Model description

Information propagation via follow and mention links in Twitter can be conceptualized as a diffusion process. This type of dynamics can be classically modeled using epidemiological models (such as SI, SIR, SIS and SIRS), Galton–Watson (branching) process, influence models (independent cascade model, linear threshold model), etc. [16,21,22,25,32,33,37,47,49]. Inspired by the same, we propose a SIR-based epidemic model SIR_{MF} to mimic the propagation of tweets on the mention–follow multiplex network (Fig. 2). Initially, all the nodes are in the susceptible state. A node v gets infected by a tweet T if it retweets T in the next timestamp. A user once infected gets recovered instantaneously in the next timestamp. We assume there is only one information (post) propagating in the system and any node can tweet/retweet it only once. The simulation stops when no more users can be infected. The size of the cascade, depicted by the infected population, denotes the popularity of the tweet. Table 1 illustrates the mapping between tweet propagation and the epidemic model.

In this framework, the infection of a node v for a tweet T is governed by three factors: (a) v has to be exposed to T, (b) v has to show interest in T and (c) v must have a certain retweet rate to retweet T; even being exposed to an interesting tweet, v may not retweet it. Precisely, (a) a node v may get exposed to tweet T by a node u in two different ways; (i) via follower links: if u posts T and v is a follower of u, (ii) via mention links: if v is not a follower of u but u mentions v while posting T. This forms the structure of the multiplex network (see Fig. 2). (b) The interest of v in tweet T depends on whether it has been exposed through mention or follow link. We model user interests (normalized between [0,1]) with two Poisson distributions with mean μ_1 and μ_2, respectively, for the posts received through mention and follow links. Since mentions are more visible than normal posts, we keep $\mu_1 \geq \mu_2$. (c) The (retweet) activity rate κ_v (normalized between [0,1]) of node v is modeled by a power law distribution with exponent κ [31].

Table 1 Mapping the terminologies and parameters of epidemic propagation and tweet propagation

Epidemic propagation	Tweet propagation
Susceptible	Users yet to post any tweet or retweet
Getting Infected	Tweeting/retweeting a post
Infected Individual	User who tweets/retweets a post
Model parameters	
κ	Exponent of power law distribution representing user activity
Infection probability (via mention) $\alpha_u = (\kappa_u \times \mu_{1_u} \times P_\lambda)$	Probability that v has been mentioned in post T and v retweets T in the next timestep
Infection probability (via follow) $\beta_u = (\kappa_u \times \mu_{2_u})$	Probability that v receives the post from followee and retweets the post in the next timestep
λ	Average number of users mentioned in each tweet T

5.1.1 Modeling the retweet rates

In SIR_{MF} model, we introduce the following two retweet (infection) rates (i) retweeting probability of the mentioned user (α) and (ii) retweeting probability of the normal followers (β). For a node v, we denote the retweeting rate (probability of infection) through (i) mention links as α_v and (ii) through follow links as β_v. The retweet probabilities are functions of user interests (μ_1 and μ_2) and user activity rates (κ_v). Hence, in Fig. 2 nodes get infected in the follow layer with average probability $\beta_v = g(\kappa_v, \mu_{2_v})$. The function g can simply be the product of all the factors. On the other hand, the retweet probability via mention (α_v) is dependent on μ_1 and κ_v, along with the number of mentions present in the tweet (denoted as λ in average). In order to model the influence of λ on α_v, we fit (using Vandermonde matrix)[4] the curve shown in Fig. 7 as a third-degree polynomial of λ, $P_\lambda = p_1\lambda^3 + p_2\lambda^2 + p_3\lambda + p_4$ where the coefficients p_1, p_2, p_3 and p_4 are estimated as -0.0020, 0.0286, -0.1309 and 0.3716, respectively. Subsequently in Fig. 2, nodes get infected in the mention layer with average probability $\alpha_v = g(\kappa_v, \mu_{1_v}, P_\lambda)$. The model parameters are summarized in Table 1.

5.1.2 Mention strategies

In SIR_{MF}, we model mention strategies following which a user u can be chosen for mentioning in a tweet. We introduce a generic *'Parametric'* mention strategy[5] where the user u is chosen preferentially to her $(f_u^{\theta_1} \times \alpha_u^{\theta_2})$ score where f_u and α_u are the follower count and retweet rate of u, respectively, and $\theta_1, \theta_2 \in [0, 1]$ are tunable parameters. Notably, the extent of preference to each factor can be regulated with parameters θ_1 and θ_2.

5.2 Simulation setup and metrics

Next, we develop a simulation setup illustrating the underlying follower network, fixing the model parameters and specifying the evaluation metrics.

5.2.1 Parameter setting

In order to simulate the SIR_{MF} model, we fix λ as the average number of mentions in the empirical dataset and $\kappa = -2.5$ considering the fact that these parameters do not change frequently over time [31]. We vary μ_1 and μ_2 to regulate the probabilities α (avg. of α_vs) and β (avg. of β_vs), respectively. Each simulation result presented in the paper is an average of 500 simulations.

5.2.2 Follower networks

A follower network is a dynamic communication medium which facilitates tweet propagation. In order to simulate the proposed SIR_{MF} model, we implement the following two types of follower networks;

1. *Empirical network* We implement two real follower networks from 'Algeria' and 'Egypt' datasets. In the 'Algeria' network, we have 21,141 users and 19,802,923 directed follow links (avg. indegree 1118.1 and avg. outdegree 772.1). The largest strongly connected component of the network contains 71% of all users. Similarly, in the 'Egypt' network, there are 59,776 users, 5,521,949 follow links (avg. indegree 116.5 and avg. outdegree 92.4) and its largest strongly connected component consists of 74% of all users. The indegree and outdegree distributions of both networks are shown in Fig. 1.

[4] http://in.mathworks.com/help/matlab/ref/polyfit.html.

[5] There exist multiple possible mention practices in reality; for instance, people mention other users in a tweet depending on their relevance with that post, depicting personal relationship with them, targeting them for trolling/cyber bullying, etc. However, the focus of *'Parametric'* mention strategy is concentrated and limited to the retweet count gained by a post.

2. *Synthetic network* We generate scale-free networks synthetically to model the follower networks. Scale-free or power law network is a popular topology to model the real social networks [4,5,12]. We generate power law degree distributions ($p_k \sim k^{-\gamma}$) with the exponent γ varying as 1.3, 1.8 and 2.3. To be able to observe the effect of γ on the tweet propagation dynamics, we fix the total number of nodes as 16,384 and the total edge count around 98,000 in all three networks so that the average degree $\langle k \rangle$ of these networks get fixed around 6. It is not very trivial to generate scale-free networks with same average degree but different exponents. Here, we use the generalized Barabási-Albert's method [4] for generating these scale-free networks. In this method, at each step a node enters the network with a constant outdegree and gets attached to existing nodes with probabilities proportional to $k + k_0$ where k is the indegree of an existing node and k_0 is a constant. By varying k_0, we vary the exponent of the obtained scale-free network.

5.2.3 Evaluation metrics

We introduce the following four metrics to quantify the role of mentions in tweet propagation dynamics. These set of metrics will be further applied for evaluating the performance of different mention recommendation algorithms in Sect. 9:

(a) *Retweet count with mentions* (R_U) is the average number of times tweets containing mentions are retweeted. In simulations, we have a single tweet in the system and that tweet contains mentions (as $\lambda > 0$); therefore, R_U is simply the infected population in the network.
(b) *Retweet count without mentions* (N_U) is the average number of times tweets without mentions are retweeted.
(c) *Retweet fraction by mentioned users* (F_M) is the average fraction of all the retweets (of the posts containing mentions) done by the mentioned users. In simulations, this gives the fraction of retweeting users who has received the tweet via mention links and retweeted it.
(d) *Fraction of mentioned users retweeted* (F_C) is the fraction of mentioned users who retweeted the post.

Note that N_U is not relevant for simulations since we only simulate tweets with mentions. Similarly, F_C is not an observable metric in simulations; this simply depicts our model parameter α. However, both metrics will play an important role to evaluate the performance of different mention recommendation algorithms in Sect. 9.

5.3 Model validation

We validate the SIR_{MF} model with respect to the retweet counts (R_U) of the tweets containing mentions in the empirical datasets. We implement the 'Parametric' mention strategy and simulate the model for each tweet (with a positive α) on the follower network obtained from the datasets. In order to execute the model, we estimate suitable μ_1 and μ_2 to keep the respective average infection probabilities α and β close to the real data. Moreover, we simulate each tweet diffusion by starting with the same set of initiators and keeping the same number of mentions (λ) as in the empirical data. We estimate the model parameters θ_1 and θ_2 using maximum likelihood estimation [8] such that the total infected population R_U exhibits best agreement with total retweet count, observed in the empirical dataset. In Fig. 8, we observe a nice agreement between the infected population of SIR_{MF} model and the real retweet count R_U estimated for both the 'Algeria' and 'Egypt' datasets. Interestingly, for most of the tweets we estimate both $\theta_1 \approx 0$ and $\theta_2 \approx 0$ from the empirical data.

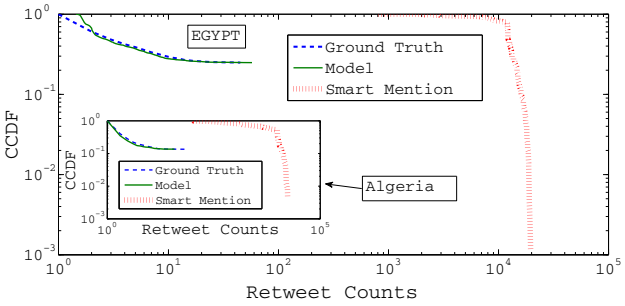

Fig. 8 Matching ground truth tweet popularities with the simulation model with parametric mention strategy (with same α, β, λ and initiator as in the dataset) and comparing with 'smart' mention strategy for 'Algeria' and 'Egypt' datasets

6 Exploring the impact of mention strategy

In this section, we dissect the *'Parametric'* mention strategy and evaluate the performance in terms of retweet count R_U. This result demonstrates the fact that there is ample scope to boost the retweet count by choosing the users to be mentioned, smartly.

6.1 Introducing smart and random mentioning

We start with the 'parametric' mention strategy and regulate the parameters θ_1 and θ_2 which can maximize the retweet count R_U. In order to do so, we vary both θ_1 and θ_2 from 0 to 1 for different α and β values and measure the corresponding R_U. In Fig. 9, we show that for all cases the strategy with $\theta_1 = \theta_2 = 1$ consistently performs the best in terms of R_U. We designate this strategy as *smart mention* where user u is chosen preferentially to her $f_u \times \alpha_u$ score. Evidently, the main objective of the 'smart' strategy is to maximize the expected number of users exposed to that tweet. Side

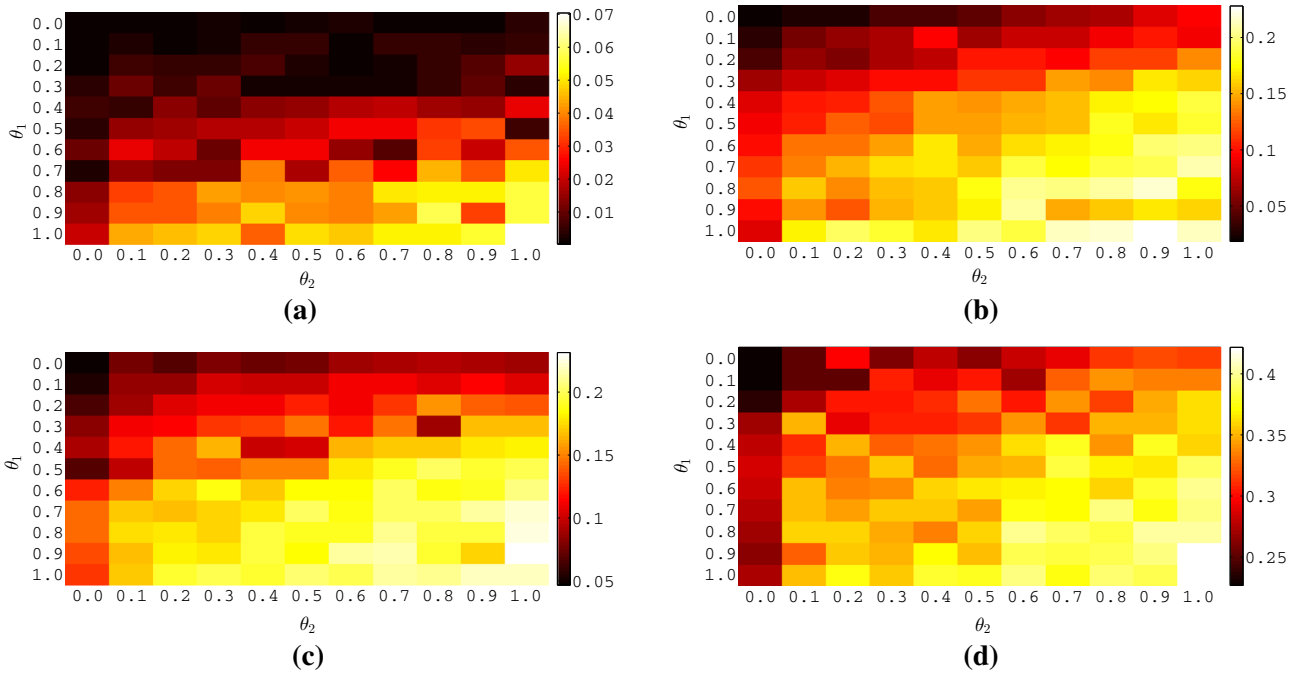

Fig. 9 Heatmaps showing R_U for different θ_1 and θ_2 values (parametric mentioning) for varying α (**a**, **b**) and β (**c**, **d**) in 'Algeria' dataset. **a** Varying θ_1 and θ_2 with $\alpha = 0.1$, $\beta = 0.01$. **b** Varying θ_1 and θ_2 with $\alpha = 0.3$, $\beta = 0.01$. **c** Varying θ_1 and θ_2 with $\alpha = 0.2$, $\beta = 0.02$. **d** Varying θ_1 and θ_2 with $\alpha = 0.2$, $\beta = 0.05$

by side, we introduce *random mention* strategy as baseline ($\theta_1 = \theta_2 = 0$) where the user u is chosen uniformly at random from the set of all susceptible users.

6.2 Benefit of smart mentioning

Next, we demonstrate the performance of smart mention strategy on the Algeria follower network. Figure 10 shows that smart mention proves beneficial especially in the low activity environment (low β). However, increase in β reduces the gap of R_U between the two mention strategies.[6] This is because as β increases, mention strategies become less important as most of the users start to get infected via only follow links. Similarly, for synthetic scale-free follower networks, the smart mention strategy outperforms the random mention strategy (see Fig. 13b). Interestingly, the gap between the R_Us corresponding to these two strategies becomes more significant with a lower power law exponent γ. This is due to the presence of hub-like nodes (high degree) in low γ scale-free networks. Unlike random mention, smart mention intelligently targets these hub nodes for mentioning, which helps it to spread the tweet to a larger population, and thereby significantly improving the retweet count R_U.

[6] In real dataset, β is observed in the range of [0.002–0.01].

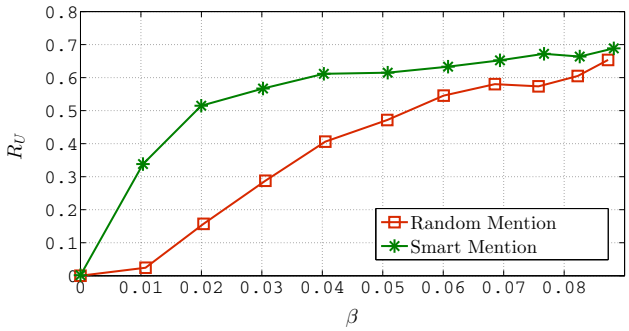

Fig. 10 Smart mentioning versus random mentioning for $\alpha = 0.4$ w.r.t. R_U in 'Algeria' dataset

7 Exploring the impact of model parameters

In this section, we investigate the role of different model parameters such as retweet rates α and β and number of mentions λ on the retweet count. We restrict ourselves to random mention strategy only, since smart mention exhibits the similar kind of observations.

7.1 Impact of retweet rates α and β

The simulation of the model points to the presence of critical retweet rates for the formation of cascade. First we consider the Algeria follower network to execute the simulation. Fig-

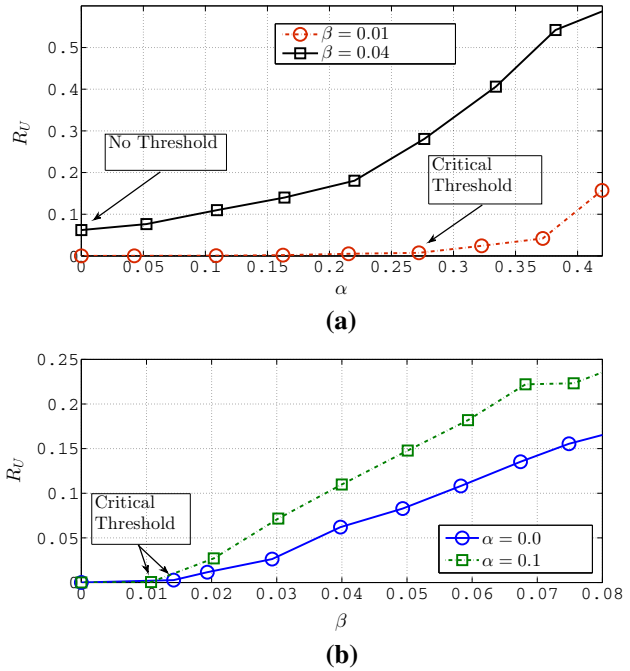

Fig. 11 Effect of varying α and β on R_U (random mentioning) in 'Algeria' dataset. **a** Effect of varying α on R_U. **b** Effect of varying β on R_U

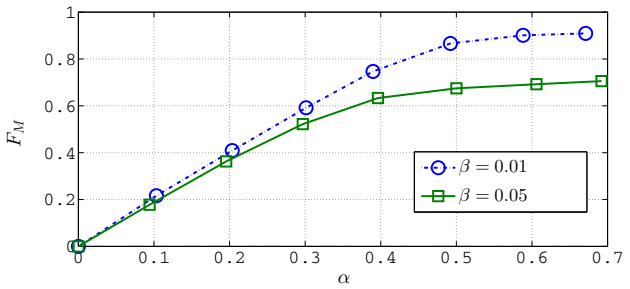

Fig. 12 Effect of varying α on F_M (random mentioning) in 'Algeria' dataset

ure 11a shows that under a critical value of α, the tweet does not gain much retweet. Once it exceeds that threshold, the total retweet count increases almost linearly with α. However, the critical threshold value of α decreases with increasing β. Similar effects can be seen if we keep α constant and vary β in X-axis (see Fig. 11b), i.e., after a threshold value of β, R_U increases sharply and that critical β threshold value lowers if α is higher. In Fig. 12, we observe that for same α, retweet fraction by the mentioned users (F_M) is lower for higher β values. This is intuitive because if β is high, more people retweet due to follow links which in turn lowers the fraction F_M. We note that F_M increases almost linearly with α up to a point and then converges.

Similarly, in Fig. 13a, we show the impact of β on R_U for synthetic scale-free topologies. Here also we observe the existence of a critical β beyond which R_U increases sharply. It is observed that in case of random mentioning, for the same α, β combination, a higher retweet count (R_U) can be achieved for the topology with a high power law exponent γ. Clearly, a scale-free topology with higher exponent γ implies higher uniformity of node degrees where it is less essential to choose the nodes for mentioning intelligently in comparison with skewed degree distributions (obtained for lower γ values). Hence, mentioning users randomly works relatively better for scale-free networks with higher γ. For the same argument, the critical threshold corresponding to β is found to be lower for higher γ values. Similar effect can be observed while varying α (not shown here).

7.2 Impact of number of mentions λ

Next, we investigate the role of λ by simulating the model on the Algeria follower network. Figure 14a shows that similar to critical α and β, there also exists a critical value of λ beyond which the total retweet count increases sharply with λ. However, as α is inversely proportional to λ, if λ crosses beyond a threshold, the drop in α sharply decreases R_U. Notably, due to this dependency between α and λ, in this plot (and other plots with λ in X-axis), we use different μ_1 values in the legend instead of α values. Similarly, in Fig. 14b, we observe that R_U increases with β for all the λ values; notably moderate λ (= 5) achieves higher R_U than the extreme cases (λ = 1, λ = 9).

Similarly, Fig. 15 confirms the existence of a critical λ for synthetic (scale-free) follower networks, beyond which R_U increases sharply. We observe an upper threshold of λ beyond which R_U decreases sharply (due to the inverse relationship between α and λ). As explained before, for random mentioning, with the same α, β and λ combination, R_U for low exponent γ is much lower than the same for higher γ.

8 Easy-Mention: recommendation heuristic

In this section, we propose *Easy-Mention*, a mention recommendation heuristic which is easily deployable in online systems. The design of *Easy-Mention* is mostly driven by the insights obtained from the model proposed in Sect. 5. Precisely, we leverage on the benefit observed in smart mention strategy and the role of regulating model parameters to develop *Easy-Mention* heuristic.

8.1 Development of Easy-Mention

The objective of the *Easy-Mention* heuristic is to recommend one user, while she posts a tweet, the best set of candidates to mention in order to boost the retweet count of that tweet. Hence, the input of the heuristic is the submitted tweet and

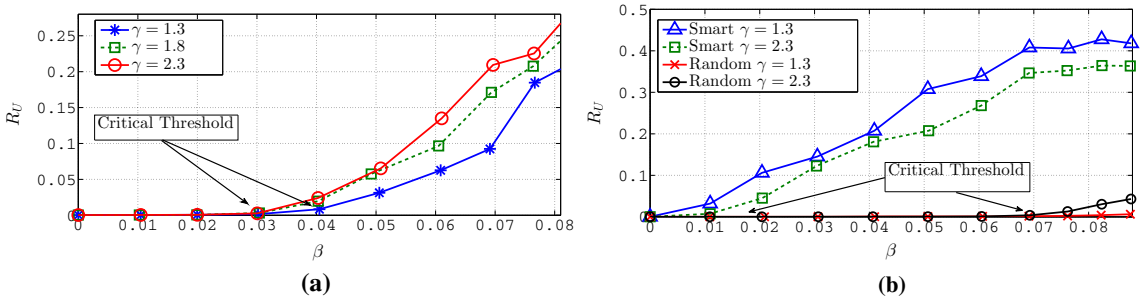

Fig. 13 Impact of β on R_U for different scale-free topologies and different mention strategies. **a** R_U vs β for different scale-free networks with $\alpha = 0.4$ (random mentioning). **b** Comparing R_U for smart mention versus random mention with varying β for different scale-free networks keeping $\alpha = 0.3$.

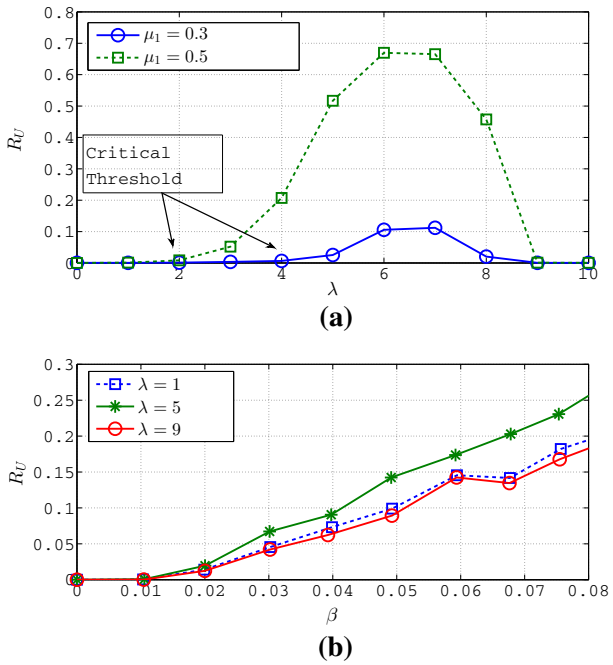

Fig. 14 Impact of λ on R_U for different α and β values in 'Algeria' dataset. **a** R_U vs λ for different μ_1 values keeping $\beta = 0.01$ fixed (random mentioning). **b** R_U versus β for different λ values keeping $\mu_1 = 0.1$ fixed (random mentioning)

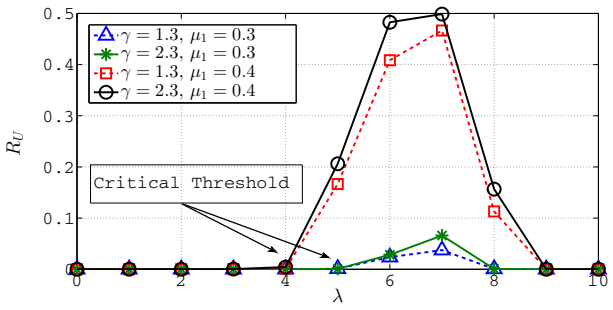

Fig. 15 Impact of λ on R_U for different scale- free topologies, keeping μ_1 and β fixed at 0.4 and 0.04 (random mentioning)

the output is a ranked list of users to be mentioned. The three major stages of this recommendation are the following.

8.1.1 Detect spammers

The first stage of *Easy-Mention* is to protect the application from malicious users. It is expected that any mention recommendation system has a high potential to be exploited by spammers for spreading their spam tweets. We implement a spammer detection algorithm (inspired from [6]) at the first stage, to refrain spammers from using our service.[7] If this stage detects one user as a potential spammer, *Easy-Mention* terminates immediately. In this spammer detection algorithm, we crawl her recent tweets and focus on the following two class of features.

(a) *Content attributes* Content attributes are features of the tweet text posted by the users, which capture specific properties related to the way people write tweets. Studies show that in general spammers post tweets with higher number of hyperlinks, mentions and hashtags compared to non-spammers [6]). We analyze the tweet content characteristics based on the maximum, minimum, average and median of the features shown in Table 2. In total, we consider 39 attributes related to the content of tweets for spammer classification.
(b) *Behavioral attributes* Behavioral attributes capture specific features connected to user behavior in terms of the posting frequency, social interactions and influence on the Twitter network. Admittedly, spammers have a lower followers to followees ratio than non-spammers and they generally possess recent accounts since Twitter continuously suspends potential spammers [6]). We consider 23 different features connected to user's behavioral attributes as summarized in Table 2.

We evaluate the performance of this spammer detection algorithm on the 'Algeria' and 'Egypt' dataset; however, the major challenge is the ground truth labeling of spammers

[7] The details of the spammer detection methodology is out of the scope of this paper.

Table 2 Examples of content and behavioral attributes used for spammer detection

Content attributes	Behavioral attributes
#Words per tweet	#Followees
#Characters per tweet	#Followers
#URLs per tweet	#Followers/#followees
#Hashtags per tweet	#Tweets
#Users mentioned per tweet	Age of account
#Retweets per tweet	#Times mentioned

and non-spammers. We train our model on a labeled spammer dataset available in [6].[8] The model classifies 537 out of 21,141 users in 'Algeria' dataset and 27 out of 59,776 users in 'Egypt' dataset as spammers. During validation, this is comforting for us to notice that 10% of the detected accounts have already been suspended by Twitter. For the remaining 90% of the accounts detected as spammers, we perform a human survey with 3 volunteers and they labeled 89% of them as true spammers unanimously by manually going through their profiles. Their justification and rationale are summarized in the inset of Fig. 16. In summary, stage I efficiently performs the spammer detection in our datasets.

8.1.2 Identify the candidate users

In stage II, we narrow down the search space for ranking and recommending the users to be mentioned. We identify the keywords in the submitted tweet (hashtags and proper nouns) and search for the followers and followees, who recently posted them. This is a quick way to collect a reasonable set of active users who are interested in that post. In general, we find that if users are mentioned within one hop neighborhood (happens in 50% of cases), they have higher probabilities of retweeting (see Fig. 16). Moreover, selecting the candidates from the one hop neighbors may significantly reduce the spamming threat for *Easy-Mention*. Hence, we denote the set of users in one hop neighborhood of the person posting the tweet, as the candidate set C_U of users to mention.

8.1.3 Calculating a score for each candidate user

In stage III, *Easy-Mention* assigns a quality score to each candidate user $u \in C_U$. This score basically signifies the expected gain in popularity of tweet T, if u is mentioned in T. The data study and SIR_{MF} model show that the following factors may regulate the quality score (i) follower count (f_P): this is motivated from the smart mention strategy described in Sect. 6.1; (ii) retweet rate (f_R): this factor captures the general retweet rate of an user. This is motivated from the

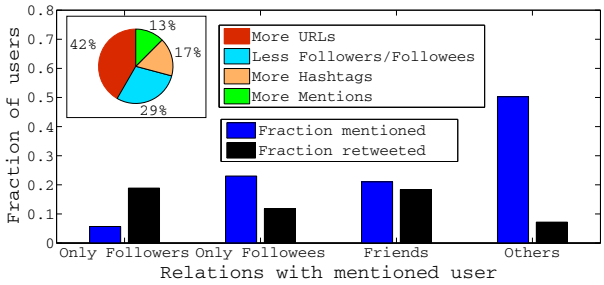

Fig. 16 Probabilities of mentioning users with different relations (reciprocal followers are denoted as 'Friends' here) and their probabilities of retweeting in 'Egypt' dataset. Inset shows the annotators' major reasons of labeling users as spammers for 'Egypt' dataset

retweet rate β of the SIR_{MF} model; and (iii) the content similarity (f_I) between the posted tweet T and the profile of the mentioned user u: a mentioned user with higher content similarity has higher propensity to retweet. This essentially captures the notion of α in SIR_{MF} model.

Finally, the score $S(u, T)$ is computed for each candidate user u related to a submitted tweet T. In order to estimate the score $S(u, T)$, we simply use the regression models to suitably combine the key features $f_P(u)$, $f_R(u)$ and $f_I(u, T)$ ($f_P(u)$ is u's normalized follower count, $f_R(u)$ is her normalized retweet rate, and $f_I(u, T)$ is the similarity between the profile of u and the tweet T) to optimize 'Relevance' introduced in [45]. Relevance of a user-tweet pair (say, u and T) is calculated as the sum of the follower counts of the (re)tweeting users in the cascade subtree (of tweet T) rooted by u. In other terms, relevance for a user-tweet pair measures the visibility brought by the user u to the tweet T.

To represent the profile of u in real time, we use the term vector T_u^V created from the words (after stemming and stopwords removal) in u's past (re)tweets.[9] In the same way, we create another term vector T_T^V for the submitted tweet T and finally calculate $f_I(u, T)$ as the cosine similarity between these two term vectors (T_u^V and T_T^V). The score $S(u, T)$ assigned to each user $u \in C_U$ prepares the ranked list of candidate users who maximize the expected visibility. The user posting the tweet is free to choose any number of users (within 140 character limit) from the ranked list for mentioning.[10]

8.2 Time complexity

Next, we compute the time complexity of the proposed *Easy-Mention* recommendation heuristic. As defined earlier, C_U

[8] http://homepages.dcc.ufmg.br/~fabrico/spammerscollection.html.

[9] In the evaluation experiments, we compose the profile of a user from all her (re)tweets in the dataset.

[10] However, as observed in Sect. 4.2 (see Fig. 6) and Sect. 7 (see Fig. 14a, b), it is not recommended to mention more than 4–5 users in a tweet.

is the set of candidate users to be mentioned by the user u posting the tweet. In this calculation, we assume that (i) the number of one hop neighbors (followers and followees) of the user u ($|C_U|$) is limited by n_f and (ii) the maximum number of tweets posted by each user is limited by n_t.

There are three components in the complexity calculation, corresponding to the major steps of the heuristic.

1. Computing candidate users' popularity: The popularity of each candidate user $v \in C_U$ is estimated by her follower count, which can be retrieved using Twitter API in $O(1)$ time (as described in Sect. 3.1). Hence, the popularities of all candidate users can be calculated in $O(n_f)$ time.
2. Computing candidate users' activity rate: Computing any user's number of retweets per day takes $O(n_t)$ time, and this has to be repeated for each follower and followee taking $O(n_t \times n_f)$ time.
3. Computing candidate users' profile similarity: Computing the similarity between the candidate post and the past tweets of a given user can be done in $O(n_t)$ time (since length of a tweet is limited by 140 characters). Again, this has to be repeated for each candidate user, taking $O(n_t \times n_f)$ time.

Finally, we combine the three components to compute the score and sort the candidate users in $O(n_f \times \log(n_f))$ time. Hence, the overall running time complexity of *Easy-Mention* becomes $O(n_t \times n_f + n_f \times \log(n_f))$.

9 Evaluation of Easy-Mention

In this section, we show the effectiveness of *Easy-Mention* by comparing it with the benchmark algorithms. We begin with explaining the experimental setup and subsequently evaluate the performance based on obtained retweet counts and run time.

9.1 Experimental setup

In order to evaluate the performance of *Easy-Mention*, first (a) we implement a standard retweet model which simulates the propagation of the tweets via *retweet* activity. Next (b) on top of the retweet model, we implement the mention recommendation algorithms to evaluate the performance of *Easy-Mention*.

9.1.1 Retweet model

We choose a well-accepted retweet model by Vespignani et al. [46]. It basically deals with competing memes in social networks and employs a parsimonious agent based model to study whether such a competition may affect the popularity of different memes. Since this is just a retweet model and does not handle the mention dynamics separately, we adapt it to include the mention utility in the following way. First we construct a tweet corpus D_T from each of the 'Algeria' and 'Egypt' datasets such that only 50% of tweets contain mentions. In order to post a new tweet or retweet, one user is chosen preferentially based on her retweet rate. If she chooses to post a new tweet, one tweet is selected randomly from D_T and she tweets the post with the same number of mentions (including zero) as in the original tweet. The specific users to be mentioned in that tweet are regulated by the specific 'mention recommendation' algorithm. The other possibility is that she opts to retweet an already received post. For each user u, we maintain a 'screen window' and a 'mention window' where tweets received via follow links (retweet from the followees of u) and tweets received via mention links (tweets where u has been mentioned) are stored, respectively. If the selected user u chooses to retweet, one of these two windows is chosen based on its similarity with the profile of u (computed as cosine similarity of term vectors), and then, the most similar post (with respect to u's profile) in that window is retweeted. However, there is a fair possibility of not retweeting any post, if the context similarity is below a threshold. The value of the threshold is fixed externally depending on the tweet environment.

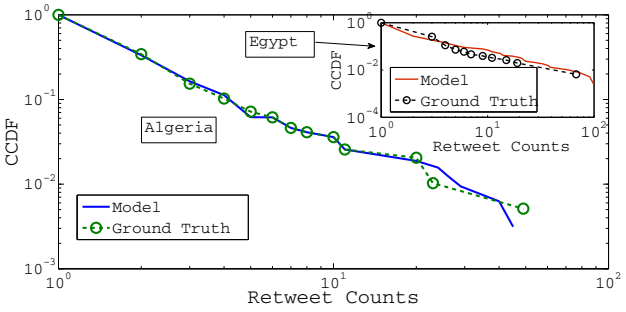

Fig. 17 Comparison of the tweet popularity distribution from the 'Algeria' dataset and the model. The inset shows the same for the 'Egypt' dataset

In order to validate, we simulate this retweet model on 'Algeria' and 'Egypt' datasets with posts containing no mentions. It is comforting for us to observe that the result explains the heterogeneity in the tweet popularity distribution with reasonable accuracy (see Fig. 17). Now we are ready to use this retweet model to evaluate the performance of different mention recommendation heuristics.

9.1.2 Competing algorithms

On top of this retweet model, we apply the proposed mention recommendation heuristic *Easy-Mention* and compare

its performance with the baseline algorithms *Whom-To-Mention* [45] and *Random-Mention*. The outline of the baseline algorithms is given below.

(a) *Whom-To-Mention* We consider *Whom-To-Mention* [45] as the state-of-the-art mention recommendation algorithm for comparison. In this algorithm, whenever a user u wishes to mention somebody in her tweet T, all the users in Twitter are considered as a potential user to mention. In order to rank these potential users, three types of features are extracted—(i) interest match between the post and users' recent tweets; (ii) Social tie; and (iii) user influence. Finally, an SVR (support vector regression)-based system is used to rank these users, taking into account the average depth of the retweet cascades created by them.
(b) *Random-Mention* This is a baseline algorithm where the recommended users to be mentioned are chosen randomly from the set of users in the dataset. This baseline is inspired from the random mention strategy introduced in Sect. 6.1.

9.2 Performance evaluation

Finally, we perform the evaluation experiments on the 'Algeria' and 'Egypt' datasets (tweets and follower network); the evaluation metrics are already introduced in Sect. 5. In this experiment, while posting a tweet T, we remove the original mentions from the tweet T and replace each mention by the username selected by the specific mention recommendation algorithm. To ensure fairness, we keep the same number of mentioned users in each tweet as in the original tweet. Once the users to be mentioned are identified, we simulate the retweet model.

9.2.1 Retweet count comparison

Figure 18 clearly illustrates the fact that *Easy-Mention* outperforms the other competing algorithms in achieving tweets with higher retweet counts. Delving deep, in Table 3 we enumerate the observed evaluation metrics for different mentioning algorithms. Table 3 uncovers the rationale behind the superiority of the *Easy-Mention*. It can be clearly observed that *Easy-Mention* is able to mention those users who not only frequently retweet that post (high F_C) but also are popular enough to give the tweet high visibility (the average follower count of users recommended by *Easy-Mention* is 158.4, whereas the same for *Whom-To-Mention* and *Random-Mention* are 93.1 and 28.2, respectively). This in turn helps *Easy-Mention* to achieve more retweets for the posts with mentions (R_U) than posts without mentions (N_U). Moreover, Fig. 19 points to the fact that the mentioned users in case of *Easy-Mention* retweet more frequently compared to the competing algorithms; this directly contributes to the cascade size. In summary, all the aforesaid properties help *Easy-Mention* to popularize tweets effectively by creating more cascades and of larger ones.

9.2.2 Run time comparison

We claim that *Easy-Mention* is optimum in terms of execution time. We establish this fact by performing comparative

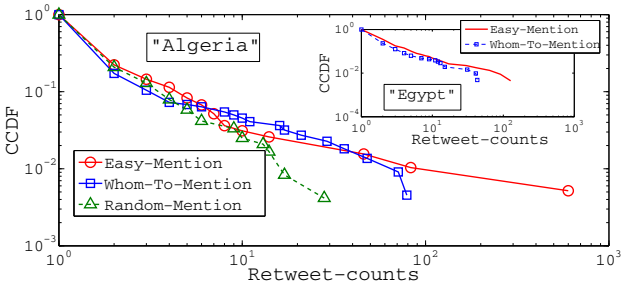

Fig. 18 CCDF of retweet counts of tweets using different mention strategies for 'Algeria' and 'Egypt' datasets

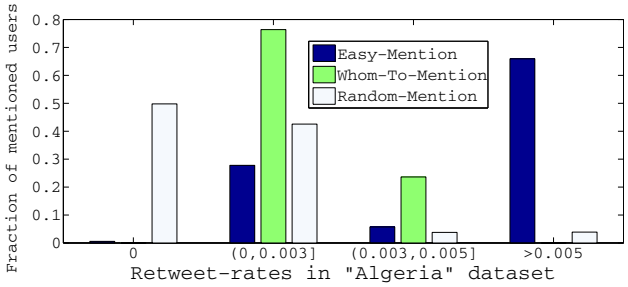

Fig. 19 Comparison of retweet rates (in 'Algeria' dataset) of users mentioned by competing recommendation algorithms

Table 3 Metric values for different mentioning strategies applied on 'Algeria' and 'Egypt' datasets. Importantly, the metric values corresponding to *Easy-Mention* are statistically higher than *Whom-To-Mention*

Dataset	Algorithms	R_U	F_M	R_U-N_U	F_C
'Algeria'	*Easy-Mention*	**2.52**	**0.136**	**0.69**	**0.087**
	Whom-To-Mention [45]	1.77	0.012	−1.31	0.024
'Egypt'	*Easy-Mention*	**2.32**	**0.588**	**1.22**	**0.195**
	Whom-To-Mention [45]	1.38	0.307	−0.19	0.029

t test confirms the statistical significance with p value < 0.05
Bold value indicates the maximum of each column (metric) for each dataset

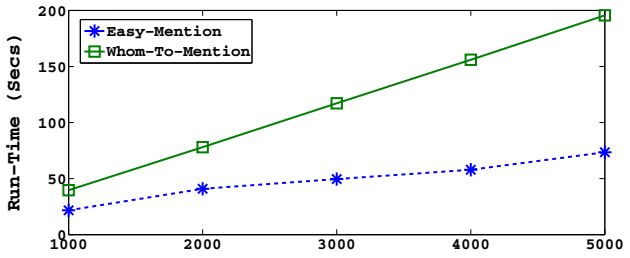

Fig. 20 Run time comparison of *Easy-Mention* and *Whom-To-Mention* recommendation algorithms with respect to number of candidate users considered

experiments on a computing system with configuration 64 GB RAM, 2 × Intel(R) Xeon(R) CPU X5690 @3.47GHz - (6 core processor). Figure 20 demonstrates the actual running time of *Easy-Mention* and *Whom-To-Mention* heuristics with respect to the number of users in the dataset. Since *Whom-To-Mention* algorithm performs costly operations (such as calculating average coverage) for each user in the dataset, the execution time sharply increases with number of users. However, in case of *Easy-Mention*, the operations (feature computation) on each user considered are quite lightweight, so even if we increase the number of candidate users, it scales slowly compared to *Whom-To-Mention*.

9.2.3 Realistic evaluation of *Easy-Mention*

In order to perform a controlled realistic evaluation of *Easy-Mention*, we conduct the following experiment for 1.5 months. We develop an application which extracts the recently posted tweets containing any of the 20 commonly used keywords (chosen by us) such as 'soccer,' 'news,' and 'storm.' We remove the actual mentions from the tweet and replace them with the users recommended by either (a) *Easy-Mention* heuristic or (b) Random-Mention algorithm. Finally, we post these modified tweets via a dummy Twitter account. Overall, we have posted 2394 such modified tweets from this dummy account, 50% of which containing mentions recommended by *Easy-Mention* and 50% containing mentions recommended by Random-Mention. At the end of the experiment, the finally obtained total retweet count for these two mention strategies have been observed as 21 and 11, respectively. This clearly indicates that in spite of receiving the posts from an unknown (dummy) account, the users mentioned by *Easy-Mention* heuristic retweet with higher propensity in comparison with Random-Mention.

10 Conclusion

In this paper, we offer an in-depth study on explaining the role of mentions on tweet virality. We have identified that a significant fraction (sometimes even up to 50–60%) of retweets might disappear if people stop using mentions (see Fig. 3a, b). In order to have a detailed understanding, we have proposed a SIR-based epidemic model, SIR_{MF} to mimic the propagation of tweets on the mention–follow multiplex framework. We have introduced a 'smart' mentioning strategy which aims to mention the users who can potentially increase the visibility of a tweet by manifold and validated it across a wide variety of parameters. Exploiting the insights obtained from the motivational studies and modeling experiments, we have extracted the following three key parameters controlling the effectiveness of mentioning: follower count, retweet rate and content similarity and proposed *Easy-Mention* recommendation heuristics. We have shown that our proposed approach outperforms the state-of-the-art *Whom-To-Mention* algorithm [45] in the yardstick of performance.

Nevertheless, the state-of-the-art 'Influence maximization' algorithms ([11,27]) may open up new possibilities for further improvement of the *Easy-Mention* heuristics.

Acknowledgements We thank the anonymous reviewer for providing insightful comments and suggestions to improve the quality of our manuscript. This work has been partially supported by the SAP Labs India Doctoral Fellowship program, DST - CNRS funded Indo - French collaborative project 'Evolving Communities and Information Spreading' and French National Research Agency contract CODDDE ANR-13-CORD-0017-01.

Compliance with ethical standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Abdullah, S., Wu, X.: An epidemic model for news spreading on twitter. In: 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, pp. 163–169 (2011)
2. Bakshy, E., Hofman, J.M., Mason, W.A., Watts, D.J.: Everyone's an influencer: quantifying influence on twitter. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, pp. 65–74 (2011a)
3. Bakshy, E., Hofman, J.M., Mason, W.A., Watts, D.J.: Identifying influencers on twitter. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM) (2011b)
4. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
5. Barabási, A.L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., Vicsek, T.: Evolution of the social network of scientific collaborations. Physica A Stat. Mech. Appl. **311**(3), 590–614 (2002)
6. Benevenuto, F., Magno, G., Rodrigues, T., Almeida, V.: Detecting spammers on twitter. In: Collaboration, Electronic Messaging, Anti-abuse and Spam Conference (CEAS), vol. 6, p. 12 (2010)
7. Bhowmick, A.K., Gueuning, M., Delvenne, J.C., Lambiotte, R., Mitra. B.: Temporal pattern of (re)tweets reveal cascade migra-

tion. In: 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (2017)
8. Bock, R.D., Aitkin, M.: Marginal maximum likelihood estimation of item parameters: application of an EM algorithm. Psychometrika **46**(4), 443–459 (1981)
9. Borge-Holthoefer, J., Rivero, A., Moreno, Y.: Locating privileged spreaders on an online social network. Phys. Rev. E **85**(6), 066,123 (2012)
10. Cha, M., Haddadi, H., Benevenuto, F., Gummadi, P.K.: Measuring user influence in twitter: the million follower fallacy. In: Proceedings of the 4th International AAAI Conference on Weblogs and Social Media (ICWSM), vol. 10, p. 30 (2010)
11. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 199–208 (2009)
12. Dezső, Z., Barabási, A.L.: Halting viruses in scale-free networks. Phys. Rev. E **65**(5), 055,103 (2002)
13. Dickens, L., Molloy, I., Lobo, J., Cheng, P.C., Russo, A.: Learning stochastic models of information flow. In: Proceedings of the 28th IEEE International Conference on Data Engineering (ICDE), pp. 570–581 (2012)
14. Freitas, C.A., Benevenuto, F., Ghosh, S., Veloso, A.: Reverse engineering socialbot infiltration strategies in twitter. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, (ASONAM), pp. 25–32 (2015)
15. Galuba, W., Aberer, K., Chakraborty, D., Despotovic, Z., Kellerer, W.: Outtweeting the twitterrs—predicting information cascades in microblogs. In: Proceedings of the 3rd Conference on Online Social Networks (WOSN), pp. 3–11 (2010)
16. Goldenberg, J., Libai, B., Muller, E.: Talk of the network: a complex systems look at the underlying process of word-of-mouth. Mark. Lett. **12**(3), 211–223 (2001)
17. Gomez-Rodriguez, M., Balduzzi, D., Schölkopf, B.: Uncovering the temporal dynamics of diffusion networks. In: Proceedings of the 28th International Conference on Machine Learning (ICML), pp. 561–568 (2011)
18. Gong, Y., Zhang, Q., Sun, X., Huang, X.: Who will you "@"? In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM), pp. 533–542 (2015)
19. González-Bailón, S., Borge-Holthoefer, J., Rivero, A., Moreno, Y.: The dynamics of protest recruitment through an online network. Sci. Rep. **1**, 197 (2011)
20. Granell, C., Gómez, S.: Dynamical interplay between awareness and epidemic spreading in multiplex networks. Phys. Rev. Lett. **111**(12), 128,701 (2013)
21. Granovetter, M.: Threshold models of collective behavior. Am. J. Sociol. **83**(6), 1420–1443 (1978)
22. Hethcote, H.W.: The mathematics of infectious diseases. SIAM Rev. **42**(4), 599–653 (2000)
23. Howard, P.N., Duffy, A., Freelon, D., Hussain, M.M., Mari, W., Mazaid, M.: Opening closed regimes: what was the role of social media during the Arab spring? Available at SSRN 2595096 (2011)
24. Jin, F., Dougherty, E., Saraf, P., Cao, Y., Ramakrishnan, N.: Epidemiological modeling of news and rumors on twitter. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis, ACM, SNAKDD '13, pp. 8:1–8:9 (2013)
25. Jin, Y., Wang, W., Xiao, S.: An sirs model with a nonlinear incidence rate. Chaos Solitons Fractals **34**(5), 1482–1497 (2007)
26. Kato, S., Koide, A., Fushimi, T., Saito, K., Motoda, H.: Network analysis of three twitter functions: favorite, follow and mention. In: Richards, D., Kang, B. (eds.) Knowledge Management and Acquisition for Intelligent Systems. Lecture Notes in Computer Science, vol. 7457, pp. 298–312. Springer, Berlin (2012)
27. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 137–146 (2003)
28. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: Proceedings of the 19th ACM International Conference on World Wide Web (WWW), pp. 591–600 (2010)
29. Kywe, S.M., Hoang, T.A., Lim, E.P., Zhu, F.: On recommending hashtags in twitter networks. In: Aberer, K., Flache, A., Jager, W., Liu, L., Tang, J., Guéret, C. (eds.) Social Informatics, vol. 7710, pp. 337–350. Springer, Berlin (2012)
30. Lee, K., Mahmud, J., Chen, J., Zhou, M., Nichols, J.: Who will retweet this? detecting strangers from twitter to retweet information. In: Proceedings of the 19th ACM International Conference on Intelligent User Interfaces (IUI), pp. 247–256 (2014)
31. Lerman, K., Ghosh, R.: Information contagion: an empirical study of the spread of news on digg and twitter social networks. In: Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM), vol. 10, pp. 90–97 (2010)
32. Leskovec, J., McGlohon, M., Faloutsos, C., Glance, N., Hurst. M.: Patterns of cascading behavior in large blog graphs. In: Proceedings of the 2007 SIAM International Conference on Data Mining, SIAM, pp. 551–556 (2007)
33. Li, M., Wang, X., Gao, K., Zhang, S.: A survey on information diffusion in online social networks: models and methods. Information **8**(4), 118 (2017)
34. Li, Q., Song, D., Liao, L., Liu, L.: Personalized mention probabilistic ranking—recommendation on mention behavior of heterogeneous social network. In: Web-Age Information Management: WAIM 2015 International Workshops: HENA, HRSUNE pp. 41–52 (2015)
35. Li, Y., Feng, Z., Wang, H., Kong, S., Feng, L.: Retweet P: Modeling and predicting tweets spread using an extended susceptible-infected-susceptible epidemic model. In: Proceedings of the 18th International Conference on Database Systems for Advanced Applications (DASFAA), pp 454–457. Springer, Berlin (2013)
36. Malhotra, A., Malhotra, C.K., See, A.: How to get your messages retweeted. MIT Sloan Manag. Rev. **53**(2), 61–66 (2012)
37. McCluskey, C.C.: Complete global stability for an sir epidemic model with delaydistributed or discrete. Nonlinear Anal. Real World Appl. **11**(1), 55–59 (2010)
38. Newman, M.E.: Spread of epidemic disease on networks. Phys. Rev. E **66**(1), 016,128 (2002)
39. Petrovic, S., Osborne, M., Lavrenko, V.: Rt to win! predicting message propagation in twitter. In: Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM), pp. 586–589 (2011)
40. Pramanik, S., Wang, Q., Danisch, M., Bandi, S., Kumar, A., Guillaume, J.L., Mitra, B.: On the role of mentions on tweet virality. In: Proceedings of the 4th IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 204–213 (2016)
41. Saito, K., Nakano, R., Kimura, M.: Prediction of information diffusion probabilities for independent cascade model. In: Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES), Part III, pp. 67–75 (2008)
42. Suh, B., Hong, L., Pirolli, P., Chi, E.H.: Want to be retweeted? large scale analytics on factors impacting retweet in twitter network. In: 2010 IEEE Second International Conference on Social computing (SocialCom), pp. 177–184 (2010)
43. Tang, L., Ni, Z., Xiong, H., Zhu, H.: Locating targets through mention in twitter. World Wide Web **18**, 1–31 (2014)
44. Uysal, I., Croft, W.B.: User oriented tweet ranking: a filtering approach to microblogs. In: Proceedings of the 20th ACM Inter-

national Conference on Information and Knowledge Management (CIKM), pp. 2261–2264 (2011)
45. Wang, B., Wang, C., Bu, J., Chen, C., Zhang, W.V., Cai, D., He, X.: Whom to mention: expand the diffusion of tweets by @ recommendation on micro-blogging systems. In: Proceedings of the 22nd International Conference on World Wide Web (WWW), pp. 1331–1340 (2013)
46. Weng, L., Flammini, A., Vespignani, A., Menczer, F.: Competition among memes in a world with limited attention. Sci. Rep. **2**, 335 (2012)
47. Yagan, O., Qian, D., Zhang, J., Cochran, D.: Conjoining speeds up information diffusion in overlaying social–physical networks. IEEE J. Select. Areas Commun. **31**(6), 1038–1048 (2013)
48. Zaman, T.R., Herbrich, R., Van Gael, J., Stern, D.: Predicting information spreading in twitter. In: Workshop on Computational Social Science and the Wisdom of Crowds. Neural Information Processing Systems (NIPS) vol. 104, pp. 17 (2010)
49. Zhao, Q., Erdogdu, M.A., He, H.Y., Rajaraman, A., Leskovec, J: Seismic: A self-exciting point process model for predicting tweet popularity. In: Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1513–1522 (2015)
50. Zhou, G., Yu, L., Zhang, C.X., Liu, C., Zhang, Z.K., Zhang, J.: A novel approach for generating personalized mention list on microblogging system. In: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), pp. 1368–1374 (2015)