9-2016

Microblogging content propagation modeling using topic-specific behavioral factors

Tuan Anh HOANG
Singapore Management University, tahoang.2011@phdis.smu.edu.sg

Ee-peng LIM
Singapore Management University, eplim@smu.edu.sg

Published in IEEE Transactions on Knowledge and Data Engineering, September 2016, 28 (9), pp. 2407-2422 .
http://doi.org/10.1109/TKDE.2016.2562628

Microblogging Content Propagation Modeling Using Topic-Specific Behavioral Factors

Tuan-Anh Hoang and Ee-Peng Lim

Abstract—When a microblogging user adopts some content propagated to her, we can attribute that to three behavioral factors, namely, *topic virality*, *user virality*, and *user susceptibility*. Topic virality measures the degree to which a topic attracts propagations by users. User virality and susceptibility refer to the ability of a user to propagate content to other users, and the propensity of a user adopting content propagated to her, respectively. In this paper, we study the problem of mining these behavioral factors specific to topics from microblogging content propagation data. We first construct a three dimensional tensor for representing the propagation instances. We then propose a tensor factorization framework to simultaneously derive the three sets of behavioral factors. Based on this framework, we develop a numerical factorization model and another probabilistic factorization variant. We also develop an efficient algorithm for the models' parameters learning. Our experiments on a large Twitter dataset and synthetic datasets show that the proposed models can effectively mine the topic-specific behavioral factors of users and tweet topics. We further demonstrate that the proposed models consistently outperforms the other state-of-the-art content based models in retweet prediction over time.

Index Terms—Content propagation, virality, susceptibility, user behavior, microblogging

1 Introduction

1.1 Motivation

Content propagates among microblogging users through their follow links, from followees to followers. The former are the *senders*, and the latter are known as the *receivers*. A receiver may adopt the content exposed to her based on a number of factors, namely the: (a) virality of the sender [1], [2], [3], (b) susceptibility of the receiver [4], [5], (c) virality of the content topic [6], [7], and (d) strength of relationships between sender and receiver [8]. User virality refers to the ability of a user in getting others to propagate her content, while user susceptibility refers to the tendency of a user to adopt her followees' content. Topic virality refers to the tendency of a topic in getting propagated. Since microblogging has been shown rather an information source than a social networking service [9], we assume in this paper that most relationships among users in a microblogging site are casual and identical in strength. We therefore focus on modeling the user and content factors that drive content propagation without considering the pairwise relationships among users.

The modeling of the virality and susceptibility factors has many important applications. In advertisement and marketing, companies may hire viral users to propagate positive content about their products, or to attach the advertisement with viral content so as to maximize their reach [10]. Similarly, politicians may leverage on viral users to disseminate their messages widely or to conduct campaigning [11], [12]. Also, one may detect events by tracking those mentioned by non-susceptible users [13], and detect rumors based on susceptible users' interactions with the content [14], [15].

There has been a number of research works measuring these virality and susceptibility factors from observed propagation data, e.g., [13], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27]. These works however suffer from two major shortcomings: (i) they do not consider the inter-relationship among the three factors, and/or (ii) they do not model the factors at topic level.

Inter-relationship among user virality, user susceptibility and content virality. Prior empirical research have suggested there are inter-dependencies among the three factors [4], [5], [26], [28], [29]. Hence, the measurement of a user's susceptibility requires the virality of topics of tweets propagated to her and the virality of users propagating the tweets. The same can be said about the measurement of user virality and topic virality. Existing models however measure the three behavioral factors *separately*. That is, they measure a user's virality by aggregating propagations on her content without considering the virality of content and susceptibility of the receivers (e.g., [17], [18], [19], [22]). Again, similar remarks are applicable to existing works that measure users' susceptibility and topics' virality (e.g., [20], [24], [27]). Such simplistic approaches may lead to less accurate modeling results.

Consider the example scenario of propagation in Twitter shown in Fig. 1(a). In this example, content are tweets ($\mathbf{t}_1, \ldots \mathbf{t}_{13}$), and the tweets are propagated from the authors (u_1, u_2, and u_2) to their followers (v_1, v_2, and v_2) when the followers retweet (forward). Without considering the followers' susceptibility one may conclude that u_3 is more viral in propagating tweets than u_1 since the former gets more retweets (i.e., 7) than the latter (i.e., 6). However, v_3 is observed to be much more susceptible than v_1 and v_2 since v_3 retweets all the followees' tweets. The same is not observed on v_1 and v_2. Moreover, u_3 receives retweets mostly by v_3 while all u_1's tweets are retweeted by all the

• *The authors are with the School of Information Systems, Singapore Management University, 80 Stamford Road, Singapore 178902. E-mail: {tahoang.2011, eplim}@smu.edu.sg.*

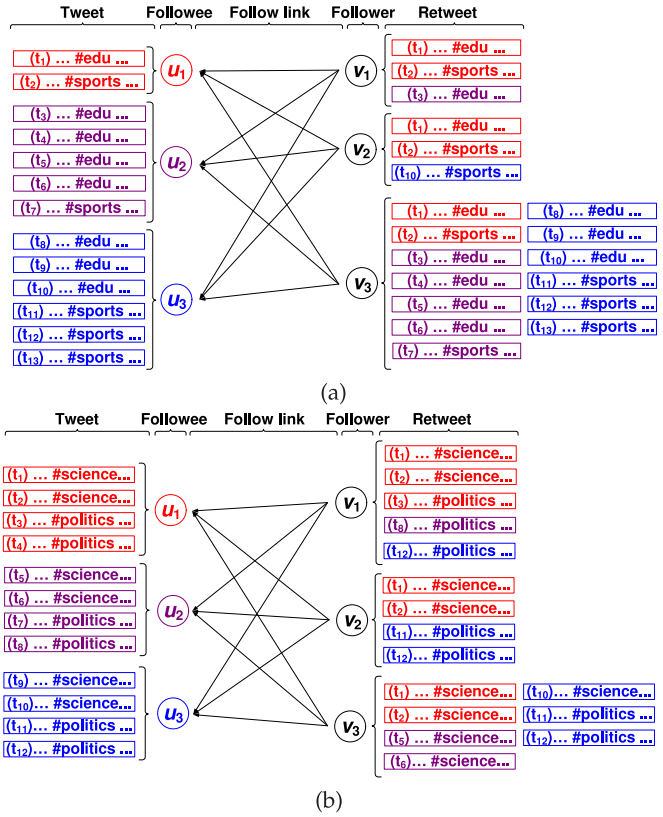


Fig. 1. Example scenarios of retweeting in microblogging.

followers. Hence, by considering that v_3 is a susceptible user, we conclude that u_1 is more viral than u_3.

Similarly, without considering the users' virality and susceptibility, one may conclude that *edu* topic is more viral than *sports* topic since the former attracts more retweets than the latter (11 and 8 respectively). However, most of *edu*'s retweets are due to u_1 who is viral, and v_3 who is susceptible. *sports* in contrast attracts retweets from all the users. Hence, a more reasonable conclusion is that *sports* is more viral than *edu*. In this research, we will incorporate these interdependencies among the factors in our proposed models.

Topic-specific virality and susceptibility. Past studies have shown that some topics are viral, e.g., political and entertainment events [11], and disasters [30], while there are also many other non-viral topics [5], [31]. Previous works have also suggested that both user virality and user susceptibility are topic specific: users are viral/susceptible on some topics but not viral/susceptible on other topics [2], [4], [5], [6], [22], [28]. Existing models however ignore content topics (e.g., [13], [16], [19], [20], [21], [22], [24], [25], [26], [27]). Such topic-independent approach may also lead to inaccurate modeling results.

Consider another example scenario of propagation in Twitter shown in Fig. 1(b). Here, again, content are tweets, and they are propagated through retweets. A topic-independent model would conclude that (a) u_1 is more viral than u_3 since the former gets more retweets (i.e., 7) than the latter (i.e., 6), and (b) v_3 is more susceptible than v_1 since the former retweets more than the latter (7 and 5 respectively). However, on *politics* topic, (i) u_3 receives retweets from all the followers, and v_1 retweets all the followees' tweets; while (ii) u_1's tweets are only retweeted by v_1, and v_3 retweets only u_3's tweets. Hence, we may conclude that, for *politics* topic u_3 is more viral than u_1 and v_1 is more susceptible than v_3.

1.2 Research Objective

Our research objective is to jointly model the above three topic-specific behavioral factors, i.e., *topic virality*, and *user topic-specific virality* and *susceptibility* from observed propagation data. Defined at the topic level, these factors can be used to perform prediction of content propagation more effectively.

To meet the objectives, we have to address a few challenges. First, both content propagation and user-content exposure instances are required for modeling the behavioral factors. However, in microblogging, we could only observe the adoption and propagation of content by users, but not their exposure to content. Second, microblogging content are known to be very noisy and their topics are not clear. For example, [32], [33] report that as many as 75 percent of tweets do not carry meaningful topics. Third, each content propagation instance is jointly determined by all the three topic-specific behavioral factors. How to separate the effects of each factor from the other two as we measure them therefore another challenge. This scenario is analogous to the computation of hubs and authorities from a set of links between web pages [34], and users' influence and passivity in social networks [21], except that we now have to consider three (not two) factors simultaneously. Last, there is no ground-truth information for the virality and susceptibility factors. The modeling results therefore cannot be evaluated using the conventional ground-truth-based evaluation metrics.

In this paper, we address the first challenge by inferring user-content exposure based on the chronological order in microblogging users' timeline and their following network. To address the second challenge, we devise a multi-step heuristic method for removing noise and identifying topics of the content, coupling with the state-of-the-art topic model for microblogging content. For the third challenge, we construct a *propagation tensor* representing senders—content—receivers relationship, and propose a factorization framework on this tensor to simultaneously derive the three topic-specific behavioral factors. We develop two factorization models base on the framework so as to learn the behavioral factors effectively. Last, to evaluate the proposed models, we examine the performance of our models in propagation prediction tasks, comparing them with the state-of-the-art baselines. We also use synthetically generated datasets to evaluate the models and the learning algorithm.

It is important to note that the problem addressed in this paper is related but not the same as modeling and maximizing information propagation. These research focus on (a) deriving the propagation rate (e.g., [35], [36]), (b) mining the interactions between user network and the diffusion process [37], [38], [39], and (c) maximizing the number of users adopting item subject to some constraint(s) (e.g., [40]). In contrast, our work focuses on deriving user and item behavioral factors from the observed propagations. Also, our work is related to but not the same as works on mining more fine-grained factors underlying user virality and susceptibility and content virality. Empirical studies have shown that there are such factors, e.g., user and network characteristics [3], [41], and emotional and linguistic

characteristics of items [7], [42]. Calibrating these fine-grained factors is however beyond the scope of this work.

1.3 Contribution

Our main contributions in this work consist of the following.

- We propose a tensor factorization framework, called **V2S** framework, to model an observed content propagation dataset using three behavioral factors, i.e., topic virality, topic-specific user virality, and topic-specific user susceptibility. Within this framework, we develop two factorization methods: *Numerical Factorization Method* and *Probabilistic Factorization Method* to simultaneously measure topics' virality as well as topic-specific users' virality and susceptibility.
- We convert the above constrained factorization problem into a unconstrained optimization which can be solved effectively using gradient descent methods.
- We apply the **V2S**—based factorization models to predict retweets in a large Twitter dataset and show that the models outperform state-of-the-art methods.
- We also conduct extensive experiments on synthetic datasets to verify the effectiveness of our approach in learning the three behavioral factors.

1.4 Paper Outline

The rest of the paper is organized as follows. We cover the related works in Section 2. Section 3 provides justifications that the behavioral factors should be modeled at the topic level. We describe our $V2S$ framework and the factorization models in Section 4. We evaluate the application of the learnt topics' and users' behavioral factors in retweet prediction tasks using large real datasets, and give empirical analysis of the factors in Section 5. We then verify the effectiveness of our approach in learning the behavioral factors through extensive experiments on synthetic datasets in Section 6. Finally, we conclude the paper in Section 7.

2 Related Works

In this section, we review prior works on analyzing content virality, user virality and susceptibility in online social networks that are closely related to ours. Also, we review works on retweet analysis since retweet is the most common action that generates content propagation in microblogging sites.

2.1 Virality and Susceptibility Analysis

User virality and susceptibility. In many works, a user's virality is simply measured by *FanOut*, i.e., the average number of friends the user diffuses item(s) to [16], [23]. Other existing works borrow user influence as a proxy for user virality, e.g., [19], [21], [22], [26]. On the other hand, prior works has measured a user's susceptibility by *FanIn*, i.e., the number or fraction of items the user adopts once she is exposed to them [26].

Early studies of user influence and susceptibility in online social networks focus on examining the existence of these factors and distinguishing them from other related factors. For example, Crandall et al. [43] and Shi et al. [44] showed that users in online communities influence each others through their interactions. Anagnostopoulos et al. [45], Aral et al. [26], and Fond et al. [46] proposed different randomization tests for distinguishing user influence and/or susceptibility from homophily effects.

The subsequent works proposed different methods for identifying influential and susceptible users in online social networks. Cha et al. [19] found that the top influential Twitter users as measured by the number of followers, pagerank score, and the number of retweets are quite different. Weng et al. [17] proposed a topic-sensitive pagerank algorithm for ranking influential users in Twitter. Using the same approach, Romero et al. [21] and Achananuparp et al. [13] proposed HITS-based algorithms for ranking, respectively, influential and passive, and originating and promoting Twitter users. There are also works on modeling user influence. For example, Goyal et al. [47] and Liu et al. [18] proposed different methods for learning users' pairwise influence from their social links, behavior traces, and generated content. Cui et al. [22] proposed a factorization method for modeling item-specific user influence.

Content virality. In some previous works, content virality has been simply measured by *popularity*, and *viral coefficient*. Popularity can be defined differently including the number of users adopting the item [27], [48], the number of *views*, *likes*, *comments*, and *shares* [49], and the number of *downloads* and *citations* [50]. Viral coefficient is defined by the average number of new adopters generated by each existing adopter [51]. For microblogging data, the viral coefficient of a tweet is the same as the retweet count of the tweet. Previous works on analyzing item virality include (a) empirical works on examining effects of different factors on item virality; and (b) works on predicting item virality.

In the first category, Romero et al. [52] showed that there is a strong correlation between a hashtag's virality and its associated content. Berger et al. [7] showed that content evoking higher-arousal positive or negative emotions is more viral. Bakshy et al. [41] and Weng et al. [27] examined the effect of the social network structures on item virality. They found that most of the content are more viral within communities of highly clustered network, while a few others are more viral across many communities through weak ties connecting the communities. Guerini et al. [50] examined effects of linguistic factors showing that certain psycholinguistic style and readability have effects on popularity of scientific papers .

In the second category, Szabo et al. [20], Li et al. [53], Shen et al. [54], and Zhao et al. [55] proposed different models for predicting long-term popularity of an item based on its early adopting patterns. Shamma et al. [49] proposed a classification method for predicting if a video goes viral in the near future based on its viewing and sharing patterns. Similarly, Bandari et al. [24] proposed both regression and classification methods for predicting popularity of a news article based on features that are derived from the article's content. Recently, Cheng et al. [56] proposed a classification method for predicting the relative growth of the number of users sharing an image based on the image's content, its early adopters, and the network among them.

In summary, the works mentioned above use simple virality and susceptibility measures that only consider some but not all the three factors in a common framework, hence neglecting the inter-relationship among the factors. Our

work here overcomes this shortcoming by showing how the behavioral factors can be jointly derived from the data traces of user—information content interactions in the propagation process. The works in [25] and [57] are the most close to ours. However, the former does not measure the factors specific to topics, while the latter does not model virality of topics. Aggregating the factors across topics oversimplifies the problem and would results in less than optimal models. Also, without topics' virality, dealing with future content items requires more side information which is not always available. Our work, on the other hand, aims to model topic-specific virality and susceptibility factors, which would can be easily used to predict propagation of future content.

2.2 Retweet Analysis

Existing works on retweet analysis include: (a) empirical works on studying the effects of different factors on tweets' retweetability, and (b) works on modeling retweet actions. In the former category, researchers have examined the correlation between retweetability with authority features [3], social and emotional features [4], [5], [29], [58], and content and linguistic features [6], [42], [59]. Most of works in the latter category formulated the retweet modeling problem as a tweet recommendation task in which retweets are considered as positive user feedback [60], [61], [62], [63]. While these works are reported to achieve high performance, they suffer from a few shortcomings. First, they use features that require a large dataset covering user activities over a long time period (e.g., users' tweets, retweets, and interactions) or even no longer available system features of Twitter (e.g., the retweet traces of tweets). In contrast, our models only requires the retweet data and considers new topic and user factors. Second, they can only perform *in-matrix* recommendation: only tweets in the training dataset can be recommended to users. Hence, they cannot be applied to predict retweets for the future tweets like our models.

Lastly, existing works on retweet modeling are based on the similarity between the tweets' topics and topical preference of users. Taking a different approach, our methods are based on the similarity between tweets' topics and topics' virality as well as topic-specific virality and susceptibility of users. Like in [63], the virality and susceptibility factors can be further combined with the existing models to derive more effective tweet recommendation methods. This extension is however beyond the scope of this paper.

3 Empirical Findings about Content Propagation

In this section, we conduct an empirical analysis of content propagation on a large dataset collected from Twitter. The methodology used to derive content propagation behavior and topics will be presented. The study will show that virality and susceptibility contributing to content propagation should be modeled at topic level.

In microblogging, retweet is the most common form of content propagation. We therefore use retweet to define propagation in the remaining part of this section. That is, *each original tweet m is considered as a content item*, and *we say user v is exposed to m if (a) v follows m's author, and (b) v receives and reads m*. Lastly, *m is said to be propagated from its author u to v if (i) v follows u and (ii) v retweets m*. We do not consider in this work the subsequent retweets of m by v's followers and by followers of the followers, since: (1) only less than 5 percent of retweets are subsequent retweets [9], and (2), as aforementioned, Twitter no longer provides subsequent retweets' trace.

TABLE 1
Statistics of the Dataset

Time window	#Users	#Tweets	#Retweeted tweets	#Retweets
0	268,676	9,612,207	396,010	1,312,037
1	269,163	9,555,811	391,980	1,309,824
2	268,386	9,362,051	377,298	1,274,902
3	267,898	9,247,465	371,962	1,257,921
4	251,940	7,646,186	284,368	791,901
5	250,559	7,651,155	289,344	802,166
6	252,139	7,941,359	312,342	873,631
7	266,093	9,561,264	414,620	1,419,549
8	265,698	9,363,371	406,117	1,401,437
9	263,262	9,169,674	393,072	1,379,512

3.1 Dataset

Our dataset is a large corpus of tweets collected just before the 2012 US presidential election. To construct this corpus, we first manually selected a set of 56 *seed users*. These are highly-followed and politically-oriented Twitter users, including major US politicians, e.g., Barack Obama, Mitt Romney, and Newt Gingrich; well known political bloggers, e.g., America Blog, Red State, and Daily Kos; and political sections of US news media, e.g., CNN Politics, and Huffington Post Politics. The set of users was then expanded by adding all users following at least three seed users so as to get more politics savvy users. Lastly, we crawled the following network among those users and all their tweets posted during the first two weeks of October 2012. This period includes many events related to the 2012 US presidential election, e.g., the national conventions of both democratic and republican parties, and the debates between presidential candidates. This dataset thus contains both network and content propagation for a large set of Twitter users actively participating US politics during a politically active period. We therefore expect tweets in this dataset to be well read, and highly retweeted.

In Twitter, topics of tweet content change rapidly and so do the user behaviors [9], [64]. We therefore conduct our analysis in a series of sliding time windows derived from the crawled dataset, each within a short duration of time, to examine topics and user behaviors in each window. More precisely, as the crawled dataset spans over 14 days, we divide it into 10 sliding windows: each window spans five days, and the sliding step is one day. This choice of window size is based on the findings of Yang et al. [65] that most of Twitter content have lifespan of around 5 days. Table 1 shows the statistics about the data in each time window. Roughly, in each time window, about 4 percent of tweets are retweeted and each of such tweets generates around 3.5 retweets, leading to around 14 percent of all the tweets are retweets. These numbers are significantly higher than those

reported in previous works (e.g., [6]). This confirms that our dataset actually contains tweets that are highly retweeted.

3.2 Methodology

Both content propagation and content topics are usually not observable when the microblogging data are crawled. We have therefore devise the methodological steps to infer them as described below.

Determining user-tweet exposure. In Twitter, the latest tweets posted by a user's followees always appear at the top of her timeline. Hence, many tweets may have been missed by the user who does not monitor the timeline closely, and such tweets would never be retweeted. As Twitter API does not reveals the tweets seen by users, we define a time window in which the received tweets will be read. We know that every retweet by a user v comes with a corresponding tweet m that v must have read. We first count the number of other tweets v receives within the duration from the time v receives m to the time v retweets m. Based on this count we estimate N_r the number of tweets a user may read on her timeline whenever she performs a retweet. We found that N_r follows a long tail distribution. For more than 90 percent of the times, N_r is not larger than 200. We therefore determine that a user v receives and actually reads through the tweet m, i.e., v is exposed to m, if and only if m is among last 200 tweets posted by v's followees up to the time v makes a retweet. Otherwise, v is considered not exposed to the tweet m.

Topic discovery. We applied TwitterLDA model [31] to automatically identify the topics of every original tweet. This step is conducted for every time window, independently from each others.

We first remove all retweets and non-informative tweets, e.g., tweets generated by third party applications like Foursqure or Instagram.[1] We then remove from remaining tweets all stop words, slang words,[2] and non-English phrases. Next, we iteratively filter away words, tweets, and users such that: each word must appear in at least 3 remaining tweets, each tweet contains at least 3 remaining words, and each user has at least 20 remaining tweets. These minimum thresholds are designed to ensure that for each user, tweet, and word, we have enough observations to learn the latent topics accurately.

Fig. 2(a) shows the likelihood of the TwitterLDA model in the first time window with respect to the number of topics K varying from 10 to 100. As expected, larger K gives larger likelihood. The quantum of improvement decreases as K increases. Considering both time and space overheads, we set $K = 80$ for the first time window. The number of topics in each of the remaining windows is determined similarly.

Based on the learnt topics and topic distributions of users, we compute the topic distribution of every remaining tweet m with author u as follows:

$$D_{m,k} \propto \theta_{u,k} \cdot \prod_{w \in m} \phi_{k,w}, \quad (1)$$

where $D_{m,k}$ and $\theta_{u,k}$ is the probability of topic k of tweet m and user u respectively; and $\phi_{k,w}$ is probability of word w given topic k.

1. https://foursquare.com/; http://instagram.com/
2. http://en.wikipedia.org/wiki/Slang

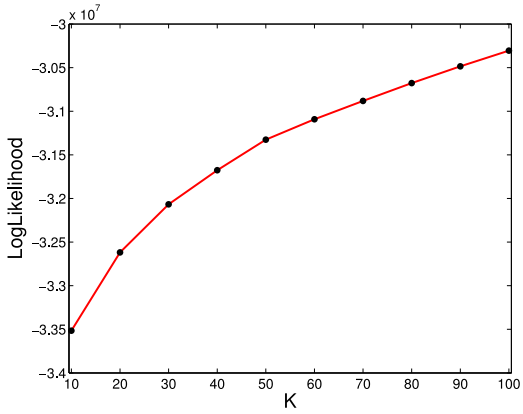


Fig. 2. Likelihood of the TwitterLDA model in the first time window

Due to the filtering steps above, many tweets are filtered away, and there is only 15 percent of tweets are topically modeled by the TwitterLDA model. We therefore expanded the set of modeled tweets as follows. First, we include in the set all the tweets of filtered away users that contain at least 3 remaining words. Then, we compute the topic distribution of each of these tweets using their (remaining) words and the learnt topics, assuming the tweet's author u (who is filtered away) has uniform topic distribution (i.e., $\theta_{u,k} = 1/K$).

Moreover, as each tweet is a short document, we are not interested in tweets that cover many topics. Instead, we only consider tweets having some dominating topics. To do this, we filter away tweets whose sum of top K_{dom} topic probabilities is less than 0.95. Then, the remaining tweets, we *normalize* their topic distributions such that: (1) sum of K_{dom} highest topic probabilities equals to 1, and (2) all other topics have probability 0. In this study, we set $K_{dom} = 3$. This number is reasonable given that there are some suggestions of assigning only one topic per tweet [31], [33].

Finally, for each time window, we obtained 25 percent tweets with topic distributions. This agrees with the previous findings that only about 25 percent of all tweets are topical tweets [32].

3.3 Empirical Findings

We now present a set of findings about how different topics get propagated (retweeted). In particular, we aim to answer the following questions: (a) Do all topics get equally retweeted? (b) Does a user get relatively same amount of retweets for every topic? and (c) Does a user performs relatively same amount of retweets for every topic?

The common notations used in this paper are shown in Table 2. Like in topic modeling, we conducted the following analysis for every time window independently from the others. Therefore, we exclude the index of time window in the notations for the simplicity in presentation.

3.3.1 Topics of Tweets and Retweets at Network Level

To compare the likelihood of getting retweeted across topics, in each time window and for each topic k, we derive the relative popularities of topic k among the set of all original tweets and the bag of retweets in the time window. The former is called *generating popularity* of the topic k, denoted by G_k, and

TABLE 2
Notations Used to Describe Topic and Behavioral Factor Analysis in One Time Window

$\mathcal{M}$	Set of all content items
$\mathcal{M}_u$	Set of content items user u generated
$\mathcal{M}_v^e$	Set of content items of user v exposed to/ adopted
$\mathcal{M}_v^a$	Set of content items of user v adopted due to propagation
p_m	Number of time content m is propagated successfully
$D_{m,k}$	Probability of topic k in content item m's topic distribution
G_k/ P_k	Generating popularity/ propagating popularity of topic k
$G_{u,k}$	Sender-specific generating popularity of user u for topic k
$P_{u,k}$	Sender-specific propagating popularity of user u for topic k
$E_{v,k}$	Receiver-specific exposing popularity of user v for topic k
$A_{v,k}$	Receiver-specific adopting popularity of user v for topic k
(u, v, m)	A propagation observation
δ_{uvm}	Indicator of (u, v, m) observation: = 1 if v adopts m, 0 otherwise
$\mathcal{O}$	Set of all propagation observations
T_k/ T	Virality of topic k/ Topic virality vector
$V_{u,k}$/ $S_{v,k}$	Virality/ susceptibility of user u/ v for topic k
V_u/ S_v	Topic-specific virality/ susceptibility vector of user u/ v
$\mathcal{V}_k$/ $\mathcal{S}_k$	Set of targeting users for virality/ susceptibility for topic k
$\mathcal{V}$/ $\mathcal{S}$	$\bigcup_k \mathcal{V}_k$/ $\bigcup_k \mathcal{S}_k$

the later is called *propagating popularity*, denoted by P_k. The two popularities are defined based as follows:

$$G_k = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} D_{m,k}, \tag{2}$$

$$P_k = \frac{1}{\sum_{m \in \mathcal{M}} p_m} \sum_{m \in \mathcal{M}} [p_m \cdot D_{m,k}], \tag{3}$$

where, in each time window, $\mathcal{M}$ is the set of all content items, and p_m is number of time m is propagated successfully. Since we use tweets and retweets to define content and propagation respectively, $\mathcal{M}$ is the set of original tweets while p_m is number of m's retweets.

To examine the difference between the two popularities of topics, we use their Pearson rank correlation coefficient $PRCC$, defined as below:

$$PRCC = \frac{\sum_{k=1}^{K} (r_G(k) - \bar{r})(r_P(k) - \bar{r})}{\sqrt{\sum_{k=1}^{K} (r_G(k) - \bar{r})^2} \sqrt{\sum_{k=1}^{K} (r_P(k) - \bar{r})^2}}, \tag{4}$$

where $r_G(k)$ is the rank of generating popularity of topic k (i.e., the rank of G_k in $G_1, \ldots, G_K$), $r_P(k)$ is the rank of propagating popularity of topic k (i.e., the rank of P_k in $P_1, \ldots, P_K$), and $\bar{r}$ is the mean rank: $\bar{r} = (K+1)/2$. Certainly, $PRCC \in [-1, 1]$. $PRCC$ is close to 1 (respectively -1) if the two popularities are strongly correlated (respectively invert correlated), and $PRCC$ is close to 0 if the two popularities are not correlated.

Fig. 3 shows the Pearson rank correlation coefficient between the two popularities across the time windows. The figure clearly shows that (a) the relative popularity of a topic in the bag of retweets is similar but not the same as the topic's popularity in the set of original tweets; and (b) this observation is consistent across the time windows. This implies that different topics have different likelihood of being retweeted.

3.3.2 Topics of Tweets and Retweets at Individual Level

On the author side. In each time window, to compare the likelihood of user u getting retweeted for different topics, we compare the relative popularities of each topic k in the set of tweets posted by u, and in the bag-of-retweets that u got. The former is called *sender-specific generating popularity* of u for topic k, while the latter one is called *sender-specific propagating popularity* of u for topic k. The two popularities are denoted by $G_{u,k}$ and $P_{u,k}$ respectively, and are defined below:

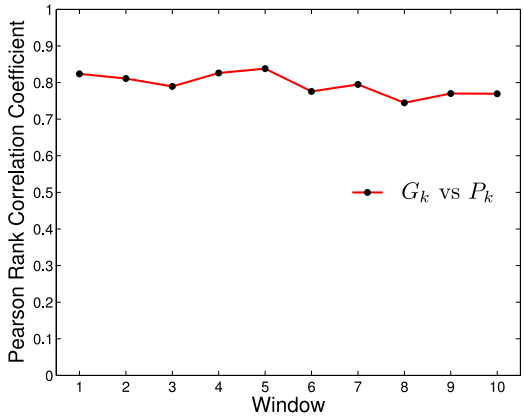


Fig. 3. Correlation between topics' popularities at network level

$$G_{u,k} = \frac{1}{|\mathcal{M}_u|} \sum_{m \in \mathcal{M}_u} D_{m,k}, \tag{5}$$

$$P_{u,k} = \frac{1}{\sum_{m \in \mathcal{M}_u} p_m} \sum_{m \in \mathcal{M}_u} [p_m \cdot D_{m,k}], \tag{6}$$

where $\mathcal{M}_u$ is the set of content items generated by u. In this section, $\mathcal{M}_u$ consists of all u's original tweets.

Similarly, we compute Pearson rank correlation coefficients between $G_{u,k}$ and $P_{u,k}$ for each user u, and between $P_{u_1,k}$ and $P_{u_2,k}$ for each pair of different users u_1 and u_2. Figs. 4(a) and (b) show the means and standard deviations of the coefficients across the time windows. The figures clearly show that, for each user, the relative popularities of topics in her bag-of-retweets are different from that popularities in her tweets, and are also different from the popularities in the bag-of-retweets of other users. This implies that **(1)** the same user has different likelihoods of getting retweeted for different topics, and **(2)** the same topic has different likelihoods of being retweeted when the topic is mentioned in the tweets generated by different users.

On the receiver side. Similarly, in each time window, to compare the likelihood of retweeting by user v for different topics, we compute the relative popularities of each topic k in the set of tweets v (exposed to and in the set of tweets v retweeted. The former popularity is called *receiver-specific exposing popularity* of user v for the topic k, and the latter is called *receiver-specific adopting popularity* of user v for topic k. The two popularities are denoted by $E_{v,k}$ and $A_{v,k}$ respectively, and are defined below:

$$E_{v,k} = \frac{1}{|\mathcal{M}_v^e|} \sum_{m \in \mathcal{M}_v^e} D_{m,k}, \tag{7}$$

$$A_{v,k} = \frac{1}{|\mathcal{M}_v^a|} \sum_{m \in \mathcal{M}_v^p} D_{m,k}, \tag{8}$$

where $\mathcal{M}_v^e$ and $\mathcal{M}_v^a$ are the set of content items v has exposed to and the set of content items v has adopted due to propagation, respectively. In this section, $\mathcal{M}_v^e$ is consist of

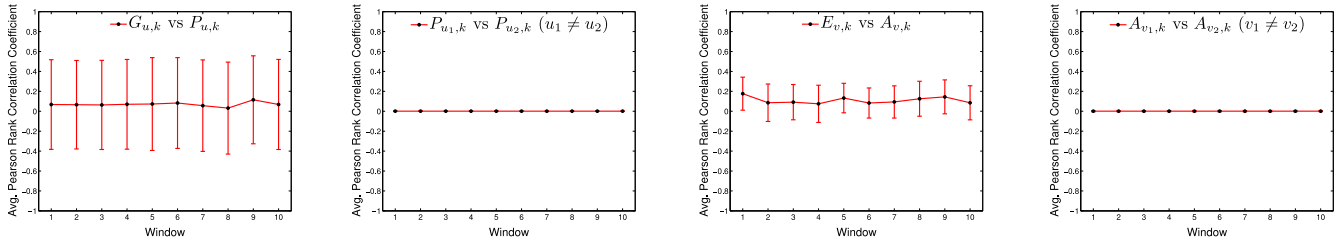


Fig. 4. Correlation between topics' popularities at individual level: (a, b) on the author side and (c, d) on the receiver side

original tweets v has received and read, while $\mathcal{M}_v^p$ is the set of retweets by v.

Again, we compute Pearson rank correlation coefficients between $E_{v,k}$ and $A_{v,k}$ for each user v, and between $A_{v_1,k}$ and $A_{v_2,k}$ for each pair of different users v_1 and v_2. Figs. 4(c) and (d) show the means and standard deviations of the coefficients across the time windows. Again, the figure clearly shows that, for each user, the relative popularities of topics in the set of tweets she retweeted are different from that popularities in the set of tweets she received and read, and are also different from the popularities in the set of tweets that other users retweeted. This implies that **(1)** the same user shows different likelihoods of performing retweet for different topics, and **(2)** the same topic has different likelihoods of being retweeted when the topic is mentioned in tweets received by different users.

4 Content Propagation Modeling Using Topic-Specific Behavioral Factors

In this section, we define the topic-specific behavioral factors and present our proposed framework that incorporates all the factors to generate microblogging content propagation data. We also present two models that implement the proposed framework, and describe an algorithm for the models' parameters learning.

4.1 Topic-Specific Diffusion Behavioral Factors

We now define the following three users' and content's behavioral factors.

Topic virality: This refers to the ability of a topic to attract propagation. Every topic k is associated to a virality score $I_k \in [0,1]$ indicating how viral the topic is, i.e. how likely a content about the topic will get propagated.

Topic-specific user virality: This refers to the ability of a user to get her content propagated for a specific topic. We assign to every user u a topic-specific user virality vector $V_u = (V_{u,1}, \ldots, V_{u,K})$ where $V_{u,k} \in [0,1]$ for $\forall k = 1, \ldots, K$. For topic k, $V_{u,k}$ denotes how viral user u is for the topic, i.e., how likely u gets propagations for her content with topic k.

Topic-specific user susceptibility: This refers to the tendency of a user to adopt content propagated to her for a specific topic. Each user v is associated with a topic-specific user susceptibility vector $S_v = (S_{v,1}, \ldots, S_{v,K})$ where $S_{v,k} \in [0,1]$ for $\forall k = 1, \ldots, K$, and $S_{v,k}$ indicates how susceptible user v is to topic k, i.e., how likely v adopts a content about the topic k after being exposed to the content.

Note that not all users generate content with a given topic, or have the chances to be exposed to content with the topic from their followees. We therefore may not be able to measure virality and susceptibility for every user-topic pair due to the lack of observation data. Instead, we identify, for each topic k, the subset of users $\mathcal{V}_k$ generating content about the topic, and the subset of users $\mathcal{S}_k$ being exposed to the topic's content. We then measure virality and susceptibility specific to topic k for users in $\mathcal{V}_k$ and in $\mathcal{S}_k$ respectively. We use V to denote the set of all $V(u)$ vectors with $u \in \mathcal{V} = \bigcup_{k=1}^{K} \mathcal{V}_k$, and use S to denote the set of all $S(v)$ vectors with $v \in \mathcal{S} = \cup_{k=1}^{K} \mathcal{S}_k$. Similarly, we use I to denote the vector $(I_1, \ldots, I_K)$ of virality scores of all K topics.

4.2 The V2S Framework

Our **V2S** framework represents each content propagation observation by a tuple (u, v, m) where m is a content item generated by user u, and exposed to user v. We use a binary variable δ_{uvm} to denote whether v adopts m ($\delta_{uvm} = 1$) or otherwise ($\delta_{uvm} = 0$). We call a propagation observation *positive* or *negative* when $\delta_{uvm} = 1$ and 0 respectively. In $V2S$ framework, δ_{uvm} depends on topic-specific virality of u, topic-specific susceptibility of v, and the topics' virality as follows.

Consider a propagation observation (u, v, m), we assume that the likelihood that v adopts m is determined by: *(a)* m's topic distribution $D_m = (D_{m,1}, \ldots, D_{m,K})$; **(b)** u's topic-specific user virality V_u; *(c)* topic virality I; and *(d)* v's topic-specific user susceptibility S_v. Under this assumption, we estimate δ_{uvm} using the dot product of D_m, V_u, I, and S_v. That is,

$$l(\delta_{uvm}) \propto f(\textstyle\sum_{k=1}^{K} [D_{m,k} \cdot V_{u,k} \cdot I_k \cdot S_{v,k}]), \tag{9}$$

where $f : [0,1] \longrightarrow \mathcal{R}^+$ is a non-negative monotonic function; and $l(\delta_{uvm})$ is either (i) an approximation of δ_{uvm}, or (ii) the likelihood of δ_{uvm}, depending on the context. Different forms of the l and f functions give rise to different implementations of the $V2S$ framework.

In $V2S$ framework, the topics' virality and the users' topic-specific virality and susceptibility can be learnt through solving the following minimization problem.

$$(I^*, V^*, S^*) = arg.min_{I,V,S}\ L(I, V, S), \tag{10}$$

subject to

$$I_k, V_{u,k}, S_{v,k} \in [0,1] \text{ for } \forall u \in \mathcal{V}_k, \forall v \in \mathcal{S}_k, \forall k = 1, \ldots, K, \tag{11}$$

where and L is the regularized sum-of-loss:

$$\begin{aligned} L(I, V, S) = {} & \textstyle\sum_{(u,v,m) \in \mathcal{O}} R_{l,f}(u, v, m) \\ & + \alpha \cdot r_1(I, V, S) + \beta \cdot r_2(I, V, S), \end{aligned} \tag{12}$$

where $\mathcal{O}$ is the set of all content propagation observations, and $R_{l,f}(u,v,m)$ is the loss in estimating δ_{uvm} with respect to the actual form of l and f. The two regularization terms r_1 and r_2 are defined as follows:

$$\begin{aligned} r_1(I,V,S) = &\textstyle\sum_{u\in\mathcal{V}}\left[||V_u - P_{u,\cdot}\cdot\sum_{k=1}^{K}V_{u,k}||^2 + ||V_u||^2\right] \\ &+\textstyle\sum_{v\in\mathcal{S}}\left[||S_v - A_{v,\cdot}\cdot\sum_{k=1}^{K}S_{v,k}||^2 + ||S_v||^2\right] \end{aligned} \tag{13}$$

$$r_2(I,V,S) = ||I - P\cdot\textstyle\sum_{k=1}^{K}I_k||^2 + ||I||^2 \tag{14}$$

In Equations 13 and 14, $P=(P_1,\ldots,P_K)$ in which P_k is defined in Equation 3. $P_{u,\cdot}$ and $A_{v,\cdot}$ are similarly formed from $P_{u,k}$s and $A_{v,k}$s which are defined in Equations 6 and 8 respectively. The term $||V_u - P_{u,\cdot}\cdot\sum_{k=1}^{K}V_{u,k}||^2$ is the distance between V_u and $P_{u,\cdot}$ after weighting the latter by sum of all components of the former. This term ensures that V_u follows a distribution that is close to $P_{u,\cdot}$ as we do expect that users should be more viral for topics which they are more likely to get propagated. Similarly, the terms $\sum_{v\in\mathcal{S}}||S_v - A_{v,\cdot}\cdot\sum_{k=1}^{K}S_{v,k}||^2$ and $||I - P\cdot\sum_{k=1}^{K}I_k||^2$ ensure that S_v and I follow distributions that are respectively close to $A_{v,\cdot}$ and P. Lastly, the regularization terms $||I||^2$ and $\sum_{u\in\mathcal{V}}||V_u||^2$, and $\sum_{v\in\mathcal{S}}||S_v||^2$ are to avoid overfitting.

4.3 Factorization Models

We now describe two factorization models built based on the $V2S$ framework.

4.3.1 Numerical Factorization Model

In this model, we consider $l(\delta_{uvm})$ as an approximation of δ_{uvm}, and f is the identity function. That is,

$$\delta_{uvm} \approx \textstyle\sum_{k=1}^{K}[D_{m,k}\cdot V_{u,k}\cdot I_k\cdot S_{v,k}]. \tag{15}$$

Given the approximation in Equation 15, the loss function $R_{l,f}(u,v,m)$ is then the squared loss, defined as follows:

$$R_{l,f}(u,v,m) = (\delta_{uvm} - \textstyle\sum_{k=1}^{K}[D_{m,k}\cdot V_{u,k}\cdot I_k\cdot S_{v,k}])^2. \tag{16}$$

4.3.2 Probabilistic Factorization Model

In this model, we consider $l(\delta_{uvm})$ as the likelihood of δ_{uvm}, and f is a probability distribution. Since $\delta_{uvm}\in\{0,1\}$, we choose f to be the Bernoulli distribution with mean $\mu(u,v,m) = \sum_{k=1}^{K}[D_{m,k}\cdot V_{u,k}\cdot I_k\cdot S_{v,k}]$. That is,

$$l(\delta_{uvm}) = \mu(u,v,m)^{\delta_{uvm}}\cdot(1-\mu(u,v,m))^{(1-\delta_{uvm})}. \tag{17}$$

The loss function $R_{l,f}(u,v,m)$ is now the negative log-likelihood of δ_{uvm}, defined as follows.

$$\begin{aligned} R_{l,f}(u,v,m) = &-\delta_{uvm}\cdot\ln(\mu(u,v,m)) \\ &-(1-\delta_{uvm})\cdot\ln(1-\mu(u,v,m)) \end{aligned} \tag{18}$$

4.4 Model Learning

Learning algorithm. With respect to the loss defined in Equations 16 or 18, the objective function $L(I,V,S)$ defined in Problem 10 is not a convex function of (I,V,S) but a convex function of I, V, and S respectively. Hence, the problem can be solved efficiently by *alternating gradient descent* methods [66]. However, due to the conditions in Equation 11, we cannot apply the methods directly as they require variables unconstrained. To deal with these conditions, we first tried the projected gradient descent method [67]. However, this method only returns locally optimal solutions for the alternating optimization problems (i.e., minimizing $L(I,V,S)$ with respect to I, V, or S), and hence results in poor solutions for Problem 10. Hence, we make use of the following variable transformation to transform the constrained variables into unconstrained ones.

$$x = h(z)\text{or}z = h^{-1}(x)\text{ for }\forall x\in[0,1], \tag{19}$$

where $h:\mathcal{R}\longrightarrow[0,1]$ is the sigmoid function:[3]

$$h(z) = \frac{1}{2}\cdot\frac{exp(z)-exp(-z)}{exp(z)+exp(-z)} + \frac{1}{2}. \tag{20}$$

Now, denote $\mathcal{Z}^t = (h^{-1}(I_k),\ldots,h^{-1}(I_K))$, $\mathcal{Z}^s(u) = (h^{-1}(V_{u,1}),\ldots,h^{-1}(V_{u,K}))$, and $\mathcal{Z}^r(v) = (h^{-1}(S_{v,1}),\ldots,h^{-1}(S_{v,K}))$, then $\mathcal{Z}^t, \mathcal{Z}^s(u), \mathcal{Z}^r(v)\in\mathcal{R}^K$. In other words, Problem 10 becomes a unconstrained optimization problem with respect to $\mathcal{Z}^t$, $\mathcal{Z}^s(u)$s, $\mathcal{Z}^r(v)$s, and now can be solved using alternating gradient descent based methods. The main idea here is to (*A1*) perform gradient descent steps by $\mathcal{Z}^s$ directions while keeping $\mathcal{Z}^r$ and $\mathcal{Z}^t$ unchanged, followed by (*A2*) performing gradient descent steps by $\mathcal{Z}^r$ directions while keeping $\mathcal{Z}^s$ and $\mathcal{Z}^t$ unchanged, and lastly (*A3*) perform gradient descent steps by $\mathcal{Z}^t$ directions while keeping $\mathcal{Z}^s$ and $\mathcal{Z}^r$ unchanged. This process repeats until we reach a predefined maximum number of iterations or when the values converge. In nth step of (*A1*) (respectively (*A2*) and (*A3*)), $\mathcal{Z}^s$ (respectively $\mathcal{Z}^r$ and $\mathcal{Z}^t$) is updated according to Equation 21 (respectively Equations 22 and 23). In these equations, $\lambda^s_{(n)}$, $\lambda^r_{(n)}$, and $\lambda^t_{(n)}$ are learning rate that are determined using the line search method [67].

$$\mathcal{Z}^s_{(n+1)} \leftarrow \mathcal{Z}^s_{(n)} - \lambda^s_{(n)}\frac{\partial L}{\partial\mathcal{Z}^s}(\mathcal{Z}^s_{(n)},\mathcal{Z}^r,\mathcal{Z}^t), \tag{21}$$

$$\mathcal{Z}^r_{(n+1)} \leftarrow \mathcal{Z}^r_{(n)} - \lambda^r_{(n)}\frac{\partial L}{\partial\mathcal{Z}^r}(\mathcal{Z}^s,\mathcal{Z}^r_{(n)},\mathcal{Z}^t), \tag{22}$$

$$\mathcal{Z}^t_{(n+1)} \leftarrow \mathcal{Z}^t_{(n)} - \lambda^t_{(n)}\frac{\partial L}{\partial\mathcal{Z}^t}(\mathcal{Z}^s,\mathcal{Z}^r,\mathcal{Z}^t_{(n)}). \tag{23}$$

Complexity. The main computational cost in each gradient descent iteration of the above learning procedure is in evaluating the objective function $L(I,V,S)$. From Equations 12, 16, and 18, we know that this cost includes (1) cost of computing the loss in estimating all propagation observations, and (2) cost of computing the regularization terms. The former is $O(K_{dom}\cdot|\mathcal{O}|)$ since we normalized topic distribution of tweets so that each tweet has at most K_{dom} topics, and the latter is $O(K\cdot(2+|\mathcal{V}|+|\mathcal{S}|))$. Hence, the cost of evaluating $L(I,V,S)$ is linear to the number of propagation observations $|\mathcal{O}|$, the number of topics K, and the number of users $|\mathcal{V}|+|\mathcal{S}|$. The number of iterations is data dependent, and we often observe the convergence after tens of alternating iterations, each with tens of gradient descent iterations. Our method is therefore scalable to large datasets.

3. $h(z)=\frac{1}{1+exp(-y)}$, with $y=2z$, is another form of sigmoid function.

Parallel implementation. We present here an implementation of the above learning algorithm that allows us to quickly evaluate the regularized sum-of-loss $L(I, V, S)$ and its gradients by parallel computing. We first rewrite the loss function as follows.

$$L(I,V,S) = \alpha \cdot r_1(I,V,S) + \beta \cdot r_2(I,V,S) + \sum_{u \in \mathbf{V}} \Big(\sum_{(u,v,m) \in \mathcal{O}_u} \sum R_{l,f}(u,v,m) \Big), \quad (24)$$

where $\mathcal{O}_u$ is the set of all propagation observations wherein u is the sender, i.e., $\mathcal{O}_u = \{(u,v,m) : (u,v,m) \in \mathcal{O}\}$.

As suggested by Equation 24, to evaluate $L(I, V, S)$, we can use multiple child processes, each corresponding to a sender u, to compute $\sum_{(u,v,m) \in \mathcal{O}_u} R_{l,f}(u,v,m)$ simultaneously. We then use a master process to compute $\alpha \cdot r_1(I,V,S) + \beta \cdot r_2(I,V,S)$ and aggregate results returned by the child processes.

Similarly, the computation of gradient of $L(I, V, S)$ by a direction is independent from those of all other directions (regardless of the variable transformation as in Equation 20). Hence, the gradient of $L(I, V, S)$ by $\mathcal{Z}^t$, $\mathcal{Z}^s(u)$, and $\mathcal{Z}^r(v)$ directions can also be computed simultaneously using multiple child processes, each corresponding to a direction $h^{-1}(I_k)$, $h^{-1}(V_{u,k})$, or $h^{-1}(S_{v,k})$.

In our implementation, in evaluating $L(I, V, S)$, we build a process pool, and submit a process for computing $\sum_{(u,v,m) \in \mathcal{O}_u} R_{l,f}(u,v,m)$ to the pool for each sender u. At any time, a fixed number **P** of the pool's processes are running. In the ideal case, we can reduce the running time of $L(I, V, S)$ to **P** times. Similarly, we use process pool to reduce the running time in computing the gradients and updating the variables.

5 Experiments on a Real Dataset

In this section, we evaluate and compare our proposed methods with some baseline methods in future propagation prediction task. Again, we use the Twitter dataset described in Section 3.1.

To deal with the dynamic of topics and the propagation factors, our dataset is divided into 10 consecutive sliding time windows, each spans five days. Since we want to examine different models in predicting propagation for the future content, we conduct the same experiments for the time windows independently. This also allows us to examine the consistency of the predictive power of models across time. Like in Section 3, we use original tweets as content, and retweets as content propagation. That is, for each time window, we *train the models using data from first 4 days of the window*, and use the models to *predict retweets for tweets posted in the last day of the window*.

5.1 Data Preprocessing

Topic discovery. For each time window, we first apply TwitterLDA model on the set of all tweets posted in the first four days of the window. We use the same pre- and post-processing steps as in Section 3.2 for learning topics of the tweets. We then use the learnt topic model to infer topics of the tweets posted in the last day of the window.

For clarity, for each time window, we call the tweet m a *training tweet* if (i) m is posted in the first 4 days of the window, and (ii) m is topically modeled. Similarly, we call the tweet m' a *test tweet* if (i) m' is posted in the last day of the window, and (ii) m' is topically modeled.

Training and test sets. We first apply the same steps presented in Section 3.2 to determine user-tweet exposure and identify all propagation observations. We then construct the training and test sets of every time window as follows.

As mentioned in Section 4.1, for each time window and each topic k, we only can measure user virality specific to topic k for a subset of users $\mathcal{V}_k$ tweeting about the topic, and measure user susceptibility specific to topic k for a subset of users $\mathcal{S}_k$ who are exposed to tweets about the topic. We therefore have to determine $\mathcal{V}_k$ and $\mathcal{S}_k$ for every topic k. To do this, we first set $\mathcal{V}_k$ and $\mathcal{S}_k$ to be the set of all users in our dataset. Then, to ensure that we have sufficient observations for each user and each topic, we iteratively: (**a**) remove from $\mathcal{V}_k$ users who have less than 5 training tweets about the topic k that are read by users in $\mathcal{S}_k$; and (**b**), remove from $\mathcal{S}_k$ users who either have no retweet on the training tweets posted by users in $\mathcal{V}_k$, or read less than five training tweets about the topic k that are posted by users in $\mathcal{V}_k$. The training set of the time window then includes all retweet observations (u,v,m) wherein $u \in \mathcal{V}$, $v \in \mathcal{S}$, and m is a training tweet posted by u. Lastly, the test set of the time window includes all retweet observations (u,v,m') wherein $u \in \mathcal{V}$, $v \in \mathcal{S}$, and m' is a test tweet posted by u.

Table 3 shows the statistics of the final dataset, called **ExpDB** dataset, which has much fewer users than the original dataset due to the different filtering criteria. Nevertheless we still have a large number of retweet observations. The table also shows that (i) the training and test sets have similar positive observation rates across the time windows, and (ii) in all the time windows, *ExpDB* is highly imbalanced with less than $1\,\mathrm{percent}$ positive observations. This makes the prediction task much more difficult.

5.2 Prediction Tasks & Evaluation Metrics

We examine the performance of different methods in the following retweet prediction tasks.

Global retweet prediction. In this task, we aim to predict positive retweet observations among all the observations in the test set, regardless of the users in the observations.

For this task, for each retweet prediction method, we generate a ranking of observations in the test set based on the likelihood of retweet returned by the method. We then construct a Precision-Recall (PR) curve from the test set and the ranking, and measure the area under the PR curve (*AUPRC*). Methods with the higher *AUPRC* are better. We choose this metric since it has been shown to be more suitable for highly imbalanced datasets like ours than other metrics (e.g., precision and recall at some given cutoff, or ROC curve) [68].

Personalized retweet prediction. In this task, given a receiver v, we aim to predict tweets that v retweets among all the tweets in the test set that v receives.

In this task, for each retweet prediction method and for each receiver v, we generate a ranking of v's observations in

TABLE 3
Statistics of the Experimental Dataset (*ExpDB*)

Time window	$\|\mathcal{V}\|$	Avg. $\|\mathcal{V}_k\|$	$\|\mathcal{S}\|$	Avg. $\|\mathcal{S}_k\|$	#observation in training set		#observation in test set	
					all	positive	all	positive
1	6,795	664.95	26,295	8,475.65	8,647,038	75,161	1,643,727	11,382
2	6,786	677.85	26,280	9,188.06	8,985,206	76,127	1,044,329	7,050
3	6,063	607.79	24,391	8,001.73	7,717,675	67,261	921,216	6,525
4	5,823	557.54	23,072	7,010.48	7,022,667	62,576	1,215,506	8,617
5	4,107	397.25	10,701	3,624.50	3,300,547	25,143	1,022,287	6,961
6	3,596	361.89	8,990	3,361.96	2,687,635	20,722	880,724	6,004
7	4,372	444.04	11,396	4,342.80	3,719,318	28,099	1,152,191	8,129
8	4,579	487.23	12,763	5,357.58	4,631,836	33,262	2,406,220	17,618
9	6,752	703.26	28,625	9,522.31	10,208,491	90,075	1,086,309	7,806
10	6,540	648.53	27,029	8,786.13	8,980,865	80,957	1,130,862	8,751

the test set based on the likelihood of retweet returned by the method. We then construct a PR curve from the v's test observations and the ranking. Last, we compute the average area under all receivers' PR curves (*Avg.AUCPR*). Methods with the higher *Avg.AUPRC* are better.

5.3 V2S-Based Models & Parameter Settings

We evaluate both two models presented in Section 4.3, i.e., **V2S**-based numerical factorization model and **V2S**-based probabilistic factorization model. We denote the former by $\mathbf{V2S}_F$, and the latter by $\mathbf{V2S}_B$.

In learning the models by alternating gradient descent, we found that the converged measure values could be obtained within 50 alternating iterations, each iteration includes 20 gradient descent steps. The control parameters α, and β are also set through empirical evaluation on a large set of tuples of values. That is, we try all possible combinations of $\alpha, \beta \in \{i \times 10^{-j} | i = 1, \cdots 9; j = 0, \cdots 5\}$ and compare their performance on the first window sub-dataset. We found that parameter set $\alpha = 10^{-4}$ and $\beta = 1$ gives the best performance. This parameter setting is reasonable as $V_{u,k}$ and $S_{v,k}$ affect only a subset of retweet observations where u and v are involved respectively. In contrast, we have much fewer variables I_k that affect a much larger set of retweet observations (where the tweets are about topic k). Hence, I should be regularized with a larger weight than that of V and S. We therefore use these parameter settings for all our experiments.

5.4 Comparison With Baselines

We first compare our proposed $V2S$-based methods with the following baselines for diffusion behavioral factors.

5.4.1 Baselines

We choose *FanOut* and *FanIn* as baseline for user virality and susceptibility respectively. In our context, the topic-specific FanOut $f^o_{u,k}$ of sender u for topic k is defined as the ratio between u's propagating popularity and her generating popularity for topic k (defined in Equations 6 and 5 respectively),

$$f^o_{u,k} = \begin{cases} P_{u,k}/G_{u,k} & \text{if } G_{u,k} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, the topic-specific FanIn $f^i_{v,k}$ of receiver v for topic k is defined as the ratio between v's adopting popularity and her exposing popularity for topic k (defined in Equations 8 and 7 respectively),

$$f^i_{v,k} = \begin{cases} A_{v,k}/E_{v,k} & \text{if } E_{v,k} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Lastly, we use the following baselines for virality of topic k.

- *Global popularity* G_k as defined in Equation 2
- *Propagation popularity* P_k as defined in Equation 3
- *Viral coefficient* vc_k defined as the average number of times an original tweet about topic k is propagated (retweeted). That is,

$$vc_k = \frac{1}{|\{m \in \mathcal{M} : D_{m,k} > 0\}|} \sum_{m \in \mathcal{M}} [p_m \cdot D_{m,k}].$$

As above baselines measure only a single user/topic factor, we combine them to the following retweet prediction methods using three factors together.

- *FanOut & global popularity & fanIn*: The likelihood $l_g(u, v, m)$ that $\delta_{uvm} = 1$ is defined as follows:

$$l_g(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot f^o_{u,k} \cdot G_k \cdot f^i_{v,k}].$$

- *FanOut & propagation popularity & fanIn*: The likelihood $l_p(u, v, m)$ that $\delta_{uvm} = 1$ is defined as follows:

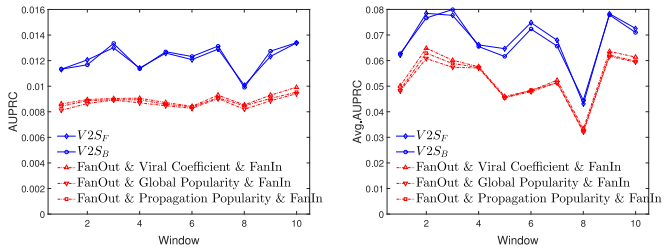
$$l_p(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot f^o_{u,k} \cdot P_k \cdot f^i_{v,k}].$$

- *FanOut & viral coefficient & fanIn*: The likelihood $l_{vc}(u, v, m)$ that $\delta_{uvm} = 1$ is defined as follows:

$$l_{vc}(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot f^o_{u,k} \cdot vc_k \cdot f^i_{v,k}].$$

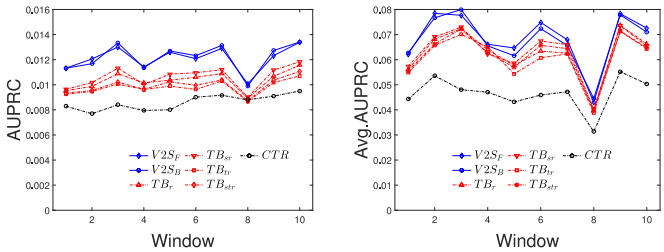
5.4.2 Performance Comparison

Fig. 5(a) shows the performance of **V2S**-based models and other baseline models in global retweet prediction task, while Fig. 5(b) shows the models' performance in personalized retweet prediction task. The figures clearly show that (i) the two **V2S**-based models have similar results while the three baselines models have similar results, and (b), across time windows, the **V2S**-based models consistently outperform the baseline models significantly.



(a) (b)

Fig. 5. Performance of different models for diffusion behavioral factors in (a) global retweet prediction task, and (b) personalized retweet prediction task.



(a) (b)

Fig. 6. Performance of different retweet recommendation models in (a) global retweet prediction task, and (b) personalized retweet prediction task.

5.5 Comparison with Content-Based Baselines for Retweet Prediction

5.5.1 Baseline Models

In this section, we compare **V2S**-based methods with methods specially designed for retweet prediction which can be viewed as a kind of recommendation task.

As aforementioned in Section 2.2, existing retweet prediction methods are for prediction on existing tweets, and hence are not applicable in our tasks—prediction for future tweets. We therefore compare our proposed **V2S**-based methods with the following content-based baseline models for the retweet prediction tasks.

$\mathbf{TB_r}$ model: The likelihood that $\delta_{uvm} = 1$ depends on topics of m, and topics where v is more likely to adopt due to propagation (retweet),

$$TB_r(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot A_{v,k}].$$

$\mathbf{TB_{sr}}$ model: The likelihood that $\delta_{uvm} = 1$ depends on topics of m, topics where u is more likely to get propagated (retweeted), and topics where v is more likely to retweet.

$$TB_{sr}(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot P_{u,k} \cdot A_{v,k}].$$

$\mathbf{TB_{tr}}$ model: The likelihood of $\delta_{uvm} = 1$ depends topics of m, topics that are more likely to be retweeted by all users, and topics where v is more likely to retweet.

$$TB_{tr}(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot P_k \cdot A_{v,k}].$$

$\mathbf{TB_{str}}$ model: The likelihood that $\delta_{uvm} = 1$ depends topics of m, topics where u is more likely to get retweeted, topics that are more likely to be retweeted by all users, and topics where v is more likely to retweet.

$$TB_{str}(u, v, m) = \sum_{k=1}^{K} [D_{m,k} \cdot P_{u,k} \cdot P_k \cdot A_{v,k}].$$

Collaborative topic regression (CTR) model [69]: This model combines collaborative filtering data with content-based features to perform recommendation tasks. Similar to our proposed methods, *CTR* is solely based on hidden user and content characteristics, and therefore is a suitable baseline. In applying *CTR*, we set the number of topics to the same as that of TwitterLDA model (see Section 3.2).

5.5.2 Performance Comparison

Fig. 6(a) shows the performance of $V2S$-based methods and other content-based baseline methods in global retweet prediction task, while Fig. 5(b) shows the models' performance in personalized retweet prediction task. Among the baseline methods, TB_r and TB_{sr} outperform the others in both tasks. This suggests that user specific retweetable topics give a stronger retweet prediction than globally retweetable topics. The fact *CTR* performs worse can be explained by *CTR* suffering from noise as the model infers tweet topics and user preference simultaneously, while other methods does not since we employ the topic normalization step (see Section 3.2). Again, the figures clearly show that, across time windows, the **V2S**-based methods consistently outperform the content-based baseline models significantly.

5.6 Case Studies

We present here case studies to illustrate how the $V2S$-based methods work differently than the baselines.

Viral topic example. Table 4 shows the profiles of topics having significantly different scores by different topic virality models. For each topic, the topic's label is manually assigned based on its representative words, and further insights from its top tweets. A topic's top words are the words having the highest probabilities given the topic, and the topic's top tweets are the tweets having the lowest perplexities given the topic. Also, for each topic k, we select a set of tweets with the normalized probability of topic k (see Section 3.2) is at least $\theta = 0.5$, and call them the *on-topic tweets* of topic k. The table shows that topic 2 (*Romney at the 1st presidential debate*[4]), topic 12 (*Obama at the 1st presidential debate*[4]), and topic 71 (*Unemployment rate*) are more popular and have more retweets than topic 41 (*"Big bird" icon in 2012 election campaigns*). However, the three formers have significantly higher proportions of retweets by their top 1 percent retweeted senders/retweeting receivers than those of the latter. This suggests that topics 2, 12, and 71's retweets are mostly due to their top viral senders and/or top susceptible receivers. Hence, it is reasonable that topic 41 is assigned much higher virality scores by $\mathbf{V2S}_F$ and $\mathbf{V2S}_B$ models.

Viral user example. Similarly, Table 5 shows the profiles of two users having most number of retweets for topic 2 (*Romney at the 1st presidential debate*[4]). The user *rolandsmartin* has more retweets for topic 2 than the user *mmfa*. However, on the topic, *rolandsmartin* has lower retweeting rate. Also, the table shows that *rolandsmartin*'s proportion of retweets by top 10 percent of her retweeting receivers is significantly higher than that of *mmfa*. This suggests that *rolandsmartin*'s retweeting users are more susceptible at topic 2 than those of *mmfa*. It is therefore reasonable that *mmfa* is assigned much higher virality scores by $\mathbf{V2S}_F$ and $\mathbf{V2S}_B$ models.

TABLE 4
Profile of Example Viral Topics at Time Window 4

Topic Id	Topic Label	#On-topic tweets	#On-topic retweet observations	#On-topic positive observations (rate)	Proportion of retweets		Global popularity	Propagation popularity	Viral coefficient	Virality	
					of top 1% retweeted senders	by top 1% retweeting receivers				by $\mathbf{V2S}_F$	by $\mathbf{V2S}_B$
2	Romney at the 1st presidential debate[4]	15,769	244,604	2,144 (0.9%)	19.4%	5.2%	**0.08**	**0.10**	0.05	0.18	0.10
12	Obama at the 1st presidential debate[4]	14,061	223,985	2,086 (0.9%)	17.4%	5.0%	0.07	**0.10**	0.04	0.86	0.27
71	Unemployment rate	6,427	183,004	2,211 (1.2%)	23.4%	4.9%	0.03	0.09	**0.14**	0.79	0.26
41	"Big bird" icon in 2012 presidential election campaigns	4,428	55,725	573 (1.0%)	14.3%	2.6%	0.01	0.03	0.07	**0.99**	**0.87**

TABLE 5
Profile of Example Viral Users at Topic 2 (*Romney at the 1st Presidential Debate*[4]) of Time Window 4

user	#On-topic tweets	#On-topic retweet observations	#On-topic positive observations (rate)	Proportion of retweets by top 10% retweeting receivers	FanOut	Virality	
						by $\mathbf{V2S}_F$	by $\mathbf{V2S}_B$
rolandsmartin	50	5,618	57 (1.0%)	26.3%	**1.3**	0.24	0.34
mmfa	9	2,198	47 (2.1%)	14.9%	1.1	**0.65**	**0.96**

TABLE 6
Profile of Example Susceptible Users at Topic 2 (*Romney at the 1st Presidential Debate*[4]) of Time Window 4

user	#On-topic received tweets	#On-topic retweet observations	#On-topic positive retweet observations	Proportion of retweets of the top retweeted sender	FanIn	Susceptibility	
						by $\mathbf{V2S}_F$	by $\mathbf{V2S}_B$
susieq68old	22	179	10 (3.57%)	50.0%	2.13	0.54	0.76
treecia73	16	104	8 (7.69%)	25.0%	1.23	**0.68**	**0.99**

Susceptible user example. Last, Table 6 shows the profiles of two users retweet the most for topic 2 (*Romney at the 1st presidential debate*[4]). The user *susie68old* retweets more for the topic than the user *treecia73*. However, *susie68old* has lower retweeting rate for the topic. Also, on topic 2, the table shows that *susie68old*'s proportion of retweets by her top retweeted senders is significantly higher than that of *treecia73*. This suggests that *susie68old*'s retweets are mostly due to a viral sender. $\mathbf{V2S}_F$ and $\mathbf{V2S}_B$ models therefore reasonably assign higher susceptibility scores to *treecia73*.

6 Experiments on Synthetic Datasets

Since real datasets do not have ground-truth information on the virality and susceptibility factors, it is impossible to evaluate the accuracy and efficacy of the proposed models in recovering the factors using the datasets. In this section, address this by conducting experiments on synthetically generated datasets. The data generating process is designed to follow the findings of previous empirical works and does not follow our proposed model closely. This is to ensure that we obtain good datasets for fair evaluations of the models.

6.1 Synthetic Data Generation

Generating the user network. We generate a follow network of N users whose in- and out-degrees are at least d_min and have power law distributions with exponent α as follows. We first sample a degree sequence of N nodes from the power law distribution. We then sample links for the nodes using the *expected degree model* [70] with the generated degree sequence. Last, for each node having less than d_{min} incoming links, we sample more incoming links for the node using the same probabilities as in the previous step until it gets d_{min} incoming links. Similarly, we sample more outgoing links for the nodes until each has at least d_{min} outgoing links.

Generating the tweets. Given the number of topics K and the number of topics dominating each tweet $K_{dom} < K$, we generate the set of tweets for each user as follows. First, we sample a topic distribution for each user so that the distribution is totally skewed to 10 percent of the K topics. This skewness is to make each user's tweets focus on only some topics and hence, for each topic the user tweets about, we have enough number of retweet observations to learn her virality for the topic. Then, the number of tweets of each user is uniformly drawn from the range $[n_{min}^{tweet}, n_{max}^{tweet}]$. To generate topic distribution for a tweet of user u, we sample the tweet's main topic from u' topic distribution. We then assign a probability of 0.9 for this main topic. Lastly, we also randomly choose other $K_{dom} - 1$ other (dominating) topics of the tweet, and randomly assign probabilities for these chosen topics so that the probabilities sum up to 0.1.

Generating the ground-truth scores. We randomly choose a small number of topics, let say $K_{viral} = 10\%$ of K, to be viral topics. These topics have virality scores randomly

4. https://en.wikipedia.org/wiki/United_States_presidential_election_debates,_2012

uniformly drawn from $[1-\epsilon, 1)$ for a small value of the so called *score width* ϵ, while the remaining topics have scores uniformly drawn from $[0, \epsilon)$. For each topic, we randomly choose a small number of users having at least one tweet about the topic, let say $N_{viral} = 2\%$ of N, to be viral users at the topic. For each user u and each topic k, if u is viral at k, the virality score of u at k is uniformly drawn from $[1-\epsilon, 1)$. Otherwise, the score is uniformly drawn from $[0, \epsilon)$. Similarly, for each topic, we also choose a small number of users receiving at least one tweet about the topic, let say $N_{susceptible} = 10\%$ of N, to be susceptible users at the topic. For each user v and each topic k, if v is susceptible at k, the susceptibility score of v at k is uniformly drawn from $[1-\epsilon, 1)$. Otherwise, the score is uniformly drawn from $[0, \epsilon)$.

Generating the retweet observations. Now that we have generated the following network, the set of tweets by each users, it is straight forward to determine which users receive which tweets of other users: v receives tweets from u if v follows u. For simplicity, we assume that v reads all the tweets she receives. Hence, we define as retweet observations all the tuples of (u, m, v) where: (i) user v follows user u, and (ii) m is a tweet of user u. A retweet observation (u, m, v), is assigned to be a positive observation (i.e., v retweets m) with the probability $prob(u, m, v)$ computed as follows:

$$prob(u, m, v) = \sum_{k=1}^{K} g_D(m, k) \frac{g_V(u, k) + g_I(k) + g_S(v, k)}{3},$$

wherein, $g_D(m, k)$ is the probability of topic k in tweet m that is generated in the previous step. Similarly, $g_V(u, k)$, $g_I(k)$, and $g_S(v, k)$ are ground-truth virality of user u for topic k, virality of topic k, and susceptibility of user v for topic k as generated previously.

6.2 Performance Comparisons

We now evaluate our proposed **V2S**-based methods and other baselines in recovering ground-truth topic-specific virality and susceptibility using the synthetic datasets. Similar to experiments in Section 5.4, we use *FanOut* and *FanIn* as baselines for user virality and susceptibility respectively, and use *Tweet popularity*, *Retweet popularity*, and *Viral coefficient* as baselines for topic virality.

We generated synthetic datasets with different number of users N, number of topics K, and score width ϵ parameter settings, while fixing $\alpha = 2.5$, $d^i_{min} = d^o_{min} = 3$, $n^{tweet}_{min} = 10$, $n^{tweet}_{min} = 100$, $K_{dom} = 3$, $K_{viral} = 10\%$ of K, $N_{viral} = 2\%$ of N, and $N_{susceptible} = 10\%$ of N. For each dataset instance and each model, we rank topics by their virality scores produced by the model and select the top scored $10\,\text{percent}$ topics as the predicted viral topics and denote the set by $\mathcal{T}_p$. The precision@10% of the model for topic virality is then defined by $\frac{|\mathcal{T}_p \cap \mathcal{T}_g|}{|\mathcal{T}_g|}$ where $\mathcal{T}_g$ is the set of viral topics in the ground truth. For each topic k, and for each user virality model, the model's precision@2% of topic-specific user virality for topic k is similarly defined, and its precision@2% across topics is computed by averaging the precision from all topics. Lastly, for each user susceptibility model, we compute the model's precision@10% across topics in the similar way.

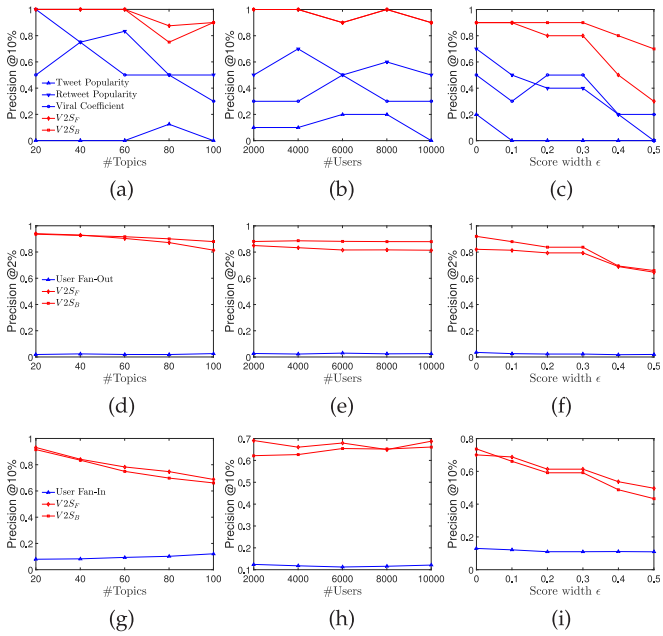


Fig. 7. Performance of different models in experiments with synthetic datasets.

Figs. 7(a), (d), and (g) show the precision@10% of topic virality models, precision@2% of user virality models, and precision@10% of user susceptibility models as we varies K from 10 to 100, keeping $N = 10,000$ and $\epsilon = 0.1$. The figures show that the **V2S**-based models significantly outperform other models. All models demonstrate decreasing precision as K increases. They however still outperform the random selection significantly.

Similarly, Figs. 7(b), (e), and (h) show the precision@10% of topic virality models, precision@2% of user virality models, and precision@10% of user susceptibility models as we varies N from 1,000 to 10,000, keeping $K = 100$ and $\epsilon = 0.1$. Figs. 7(c), (f), and (i) show the precisions as we varies ϵ from 0.1 to 0.5, keeping $K = 100$ and $N = 10,000$. Again, all the models demonstrate decreasing precision as N and ϵ increases though still outperform the random selection

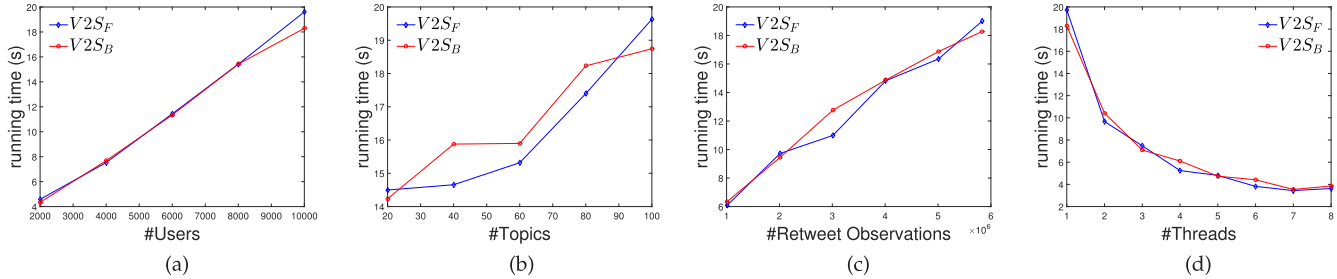


Fig. 8. Running time of the **V2S**-based models in different settings of the number of (a) users, (b) topics, (c) retweet observation, and (d) parallel threads.

significantly; and the **V2S**-based models significantly outperform other models.

6.3 Scalability

We theoretically analyse the complexity of our learning algorithm for **V2S**-based models and describe a parallel implementation in Sections 4.4. We now empirically examine the running time of the algorithm and the efficacy of the implementation.

Running time. Fig. 8(a) shows the running time of **V2S**-based models in one alternating iteration as we varies N from 1,000 to 10,000, keeping $K = 100$. Similarly, Fig. 8(b) shows the running time as we varies K from 20 to 100, keeping $N = 10,000$, and Fig. 8(c) shows the running time as we varies number of retweet observations $|\mathbf{O}|$ from 1 million to 10 millions, keeping $K = 100$ and $N = 10,000$. In all these three cases, we keep $\epsilon = 0.1$. The figures clearly show that the running time of **V2S**-based models are linear to the number of users, the number of topics, and the number of retweet observations. This verifies the learning algorithm's theoretical complexity, and shows its scalability.

Efficacy of the parallel implementation. Fig. 8(d) shows the running time of **V2S**-based models in one alternating iteration as we varies the number of parallel processes from 1 to 8, keeping number of retweet observations $|\mathcal{O}| = 10$ millions, $K = 100$, and $N = 10,000$. The figure shows that the larger the number of parallel processes used **P** results in less running time, and the amount of improvement decreases as **P** increases. This shows the efficacy of our parallel implementation. The fact that the running time even increases slightly when **P** is increased to 8 is expected due to the additional time for managing the process pool.

7 Conclusion

In this paper, we study user and content factors underlying content propagation in microblogging. Motivated by an empirical studying showing that different topics have different likelihood of getting propagated at both network and individual levels, we propose to model the factors to topic level. We develop **V2S**, a tensor factorization based framework and its associated models, to learn topic-specific user virality and susceptibility, and topic virality from content propagation data. Our experiments on a large Twitter dataset shows that the proposed **V2S**-based models outperform baseline models significantly in propagation prediction. Our experiments on synthetic databases also show that our proposed models outperform all the other baseline methods in learning the topic-specific factors.

In the future, we want to relax the assumption on the tie identical strength by incorporating heterogeneous pair-wise influence among users in modeling the propagation. We would also like to incorporate more fine-grained factors affecting the propagation. These factors include users' positions in the network, linguistic features in content, and emotion factors of users.

Acknowledgements

This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

[1] 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.

[2] S. A. Macskassy and M. Michelson, "Why do people retweet? anti-homophily wins the day!" in *Proc. 5th Int. AAAI Conf. Weblogs Social Media*, 2011, pp. 209–216.

[3] Z. Liu, L. Liu, and H. Li, "Determinants of information retweeting in microblogging," *Internet Res.*, vol. 22, pp. 443–466, 2012.

[4] S. Stieglitz and L. Dang-Xuan, "Political communication and influence through microblogging–an empirical analysis of sentiment in twitter messages and retweet behavior," in *Proc. 45th Hawaii Int. Conf. Syst. Sci.*, 2012, pp. 3500–3509.

[5] T.-A. Hoang, W. W. Cohen, E.-P. Lim, D. Pierce, and D. P. Redlawsk, "Politics, sharing and emotion in microblogs," in *Int. Conf. Adv. Soc. Netw. Anal. Mining*, 2013, pp. 282–289.

[6] B. Suh, L. Hong, P. Pirolli, and E. H. Chi, "Want to be retweeted? large scale analytics on factors impacting retweet in twitter network," in *Proc. IEEE 2nd Int. Conf. Social Comput.*, 2010, pp. 177–184.

[7] J. A. Berger and K. L. Milkman, "What makes online content viral?" *J. Marketing Res.*, vol. 49, pp. 192–205, 2012.

[8] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proc. 21st Int. Conf. World Wide Web*, 2012, pp. 519–528.

[9] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proc. 21st Int. Conf. World Wide Web*, 2010, pp. 591–600.

[10] B. J. Jansen, M. Zhang, K. Sobel, and A. Chowdury, "Twitter power: Tweets as electronic word of mouth," *J. Amer. Soc. Inform. Sci. Technol.*, vol. 60, pp. 2169–2188, 2009.

[11] Z. Zhou, R. Bandari, J. Kong, H. Qian, and V. Roychowdhury, "Information resonance on twitter: watching iran," in *Proc. 1st Workshop Social Media Anal.*, 2010, pp. 123–131.

[12] J. H. Parmelee and S. L. Bichard, *Politics and the Twitter Revolution: How Tweets Influence the Relationship Between Political Leaders and the Public*. Lexington Books, Lanham, MD, 2011.

[13] P. Achananuparp, E.-P. Lim, J. Jiang, and T.-A. Hoang, "Who is retweeting the tweeters? modeling, originating, and promoting behaviors in the twitter network," *ACM Trans. Manage. Inform. Syst.*, vol. 3, 2012, Art. no. 13.

[14] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on Twitter," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 675–684.

[15] J. Ratkiewicz, M. Conover, M. Meiss, B. Gonalves, A. Flammini, and F. Menczer, "Detecting and tracking political abuse in social media," in *Proc. 5th Int. AAAI Conf. Weblogs Social Media*, 2011, pp. 297–304.

[16] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proc. 13th Int. Conf. World Wide Web*, 2004, pp. 491–501.

[17] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, 2010, pp. 261–270.

[18] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang, "Mining topic-level influence in heterogeneous networks," in *Proc. 19th ACM Conf. Inf. Knowl. Manage.*, 2010, pp. 199–208.

[19] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in Twitter: The million follower fallacy," in *Proc. Int. AAAI Conf. Weblogs Social Media*, 2010, pp. 10–17.

[20] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Commun. ACM*, vol. 53, pp. 80–88, Aug. 2010.

[21] D. Romero, W. Galuba, S. Asur, and B. Huberman, "Influence and passivity in social media," in *Proc. 20th Int. Conf. Companion World Wide Web*, 2011, pp. 113–114.

[22] P. Cui, F. Wang, S. Liu, M. Ou, S. Yang, and L. Sun, "Who should share what?: item-level social influence prediction for users and posts ranking," in *Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval*, 2011, pp. 185–194.

[23] J. L. Iribarren and E. Moro, "Affinity paths and information diffusion in social networks," *Social netw.*, vol. 33, pp. 134–142, 2011.

[24] R. Bandari, S. Asur, and B. A. Huberman, "The pulse of news in social media: Forecasting popularity," in *Proc. Int. AAAI Conf. Weblogs Social Media*, 2012, pp. 26–33.

[25] T.-A. Hoang and E.-P. Lim, "Virality and susceptibility in information diffusions," in *Proc. Int. AAAI Conf. Weblogs Social Media*, 2012, pp. 146–153.

[26] S. Aral and D. Walker, "Identifying influential and susceptible members of social networks," *Science*, vol. 337, pp. 337–341, 2012.

[27] L. Weng, F. Menczer, and Y.-Y. Ahn, "Virality prediction and community structure in social networks," *Sci. Rep.*, vol. 3, 2013.

[28] S. Wu, J. M. Hofman, W. A. Mason, and D. J. Watts, "Who says what to whom on Twitter," in *Proc. Int. Conf. World Wide Web*, 2011, pp. 705–714.

[29] M. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, and F. Menczer, "Political polarization on Twitter," in *Proc. Int. AAAI Conf. Weblogs Social Media*, 2011, pp. 89–96.

[30] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: what Twitter may contribute to situational awareness," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2010, pp. 1079–1088.

[31] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing Twitter and traditional media using topic models," in *Proc. 33rd Eur. Conf. Advances Inform. Retrieval*, 2011, pp. 338–349.

[32] R. Balasubramanyan and A. Kolcz, "w00t! feeling great today! chatter in Twitter: Identification and prevalence," in *Proc. IEEE/ACM Int. Conf. Advances Social Netw. Anal. Mining*, 2013, pp. 306–310.

[33] S.-H. Yang, A. Kolcz, A. Schlaikjer, and P. Gupta, "Large-scale high-precision topic modeling on Twitter," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 1907–1916.

[34] J. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. ACM*, vol. 46, pp. 604–632, 1999.

[35] S. A. Myers, C. Zhu, and J. Leskovec, "Information diffusion and external influence in networks," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 33–41.

[36] M. Gomez-rodriguez, J. Leskovec, et al., "Modeling information propagation with survival theory," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 666–674.

[37] S.-H. Yang and H. Zha, "Mixture of mutually exciting processes for viral diffusion," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1–9.

[38] N. Du, L. Song, M. Gomez-Rodriguez, and H. Zha, "Scalable influence estimation in continuous-time diffusion networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3147–3155.

[39] M. Farajtabar, Y. Wang, M. Rodriguez, S. Li, H. Zha, and L. Song, "Coevolve: A joint point process model for information diffusion and network co-evolution," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1945–1953.

[40] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2003, pp. 137–146.

[41] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proc. Int. Conf. World Wide Web*, 2012, pp. 519–528.

[42] C. Tan, L. Lee, and B. Pang, "The effect of wording on message propagation: Topic-and author-controlled natural experiments on Twitter," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2014, pp. 175–185.

[43] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri, "Feedback effects between similarity and social influence in online communities," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 160–168.

[44] X. Shi, J. Zhu, R. Cai, and L. Zhang, "User grouping behavior in online forums," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 777–786.

[45] A. Anagnostopoulos, R. Kumar, and M. Mahdian, "Influence and correlation in social networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 7–15.

[46] T. La Fond and J. Neville, "Randomization tests for distinguishing social influence and homophily effects," in *Proc. Int. Conf. World Wide Web*, 2010, pp. 601–610.

[47] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2010, pp. 241–250.

[48] F. Bonchi, C. Castillo, and D. Ienco, "Meme ranking to maximize posts virality in microblogging platforms," *J. Intell. Inform. Syst.*, vol. 40, pp. 211–239, 2013.

[49] D. A. Shamma, J. Yew, L. Kennedy, and E. F. Churchill, "Viral actions: Predicting video view counts using synchronous sharing behaviors," in *Int. AAAI Conf. Weblogs Social Media*, pp. 618–621, 2011.

[50] M. Guerini, A. Pepe, and B. Lepri, "Do linguistic style and readability of scientific abstracts affect their virality?" in *Proc. 6th Int. AAAI Conf. Weblogs Social Media*, 2012, pp. 475–478.

[51] S. Jurvetson, "From the ground floor: What exactly is viral marketing?" *Red Herring Commun.*, pp. 110–111, May 2000.

[52] D. M. Romero, B. Meeder, and J. Kleinberg, "Differences in the mechanics of information diffusion across topics: Idioms, political hashtags, and complex contagion on twitter," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 695–704.

[53] H. Li, X. Ma, F. Wang, J. Liu, and K. Xu, "On popularity prediction of videos shared in online social networks," in *Proc. 22nd ACM Int. Conf. Inform. Knowl. Manag.*, 2013, pp. 169–178.

[54] H. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 291–297.

[55] 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. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1513–1522.

[56] 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.

[57] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware social influence propagation models," *Knowl. Inform. Syst.*, vol. 37, no. 3, pp. 555–584, 2013.

[58] L. K. Hansen, A. Arvidsson, F. A. Nielsen, E. Colleoni, and M. Etter, "Good friends, bad news—Affect and virality in twitter," in *Proc. 6th Int. Conf. Future Inform. Technol.*, 2011, pp. 34–43.

[59] J. Yang and S. Counts, "Predicting the speed, scale, and range of information diffusion in twitter," in *Proc. 4th Int. Conf. Weblogs Social Media*, 2010, pp. 355–358

[60] Z. Yang, J. Guo, K. Cai, J. Tang, J. Li, L. Zhang, and Z. Su, "Understanding retweeting behaviors in social networks," in *Proc. 19th ACM Int. Conf. Inform. Knowl. Manag.*, 2010, pp. 1633–1636.

[61] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu, "Collaborative personalized tweet recommendation," in *Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval*, 2012, pp. 661–670.

[62] R. Yan, M. Lapata, and X. Li, "Tweet recommendation with graph co-ranking," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguistics*, 2012, pp. 516–525.

[63] Y. Pan, F. Cong, K. Chen, and Y. Yu, "Diffusion-aware personalized social update recommendation," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 69–76.

[64] J. Lin and G. Mishne, "A study of "churn" in tweets and real-time search queries," in *Proc. 6th Int. AAAI Conf. Weblogs Social Media*, 2012, pp. 503–506.

[65] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 177–186.

[66] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[67] C. T. Kelley, *Iterative Methods for Optimization*, Philadelphia, PA, USA: SIAM, 1999.

[68] J. Davis, M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 233–240.

[69] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 448–456.

[70] F. Chung and L. Lu, "The average distances in random graphs with given expected degrees," *Proc. Nat. Acad. Sci.*, vol. 99, pp. 15879–15882, 2002.