Topical Influence Modeling via Topic-Level Interests and Interactions on Social Curation Services

Daehoon Kim[1], Jae-Gil Lee[1] *, Byung Suk Lee[2]
[1]Department of Knowledge Service Engineering, KAIST, Korea
[2]Department of Computer Science, University of Vermont, U.S.A.
Email: {daehoonkim, jaegil}@kaist.ac.kr, bslee@uvm.edu

***Abstract*—*Social curation services* are emerging social media platforms that enable users to curate their contents *according to the topic* and express their interests *at the topic level* by following curated collections of other users' contents rather than the users themselves. The topic-level information revealed through this new feature far exceeds what existing methods solicit from the traditional social networking services, to greatly enhance the quality of *topic-sensitive* influence modeling. In this paper, we propose a novel model called the *topical influence with social curation* (*TISC*) to find *influential users* from social curation services. This model, formulated by the continuous conditional random field, fully takes advantage of the explicitly available topic-level information reflected in both contents and interactions. In order to validate its merits, we comprehensively compare TISC with state-of-the-art models using two real-world data sets collected from Pinterest and Scoop.it. The results show that TISC achieves higher accuracy by up to around 80% and finds more convincing results in case studies than the other models. Moreover, we develop a distributed learning algorithm on Spark and demonstrate its excellent scalability on a cluster of 48 cores.**

I. Introduction

Content curation is the process of collecting, organizing, and displaying information relevant to a particular topic or area of interest [1]. *Social curation*, then, is defined to be collaborative sharing of Web contents in support of content curation [2]. Thus, a social curation service combines social media features, such as liking, following, and commenting, with content curation features. A large number of social curation services have been launched recently, including Digg, Reddit, Delicious, Pinterest, We Heart It, Storify, and Scoop.it, to name just a few. They have been applied to filter out uninteresting contents from the Web or social media, thereby mitigating the problem of *information overload*. Pinterest[1] is one of the most popular services among them [3], becoming the 3rd popular social media site in 2014.[2]

Recent studies on social network analysis—especially, expert finding and influence maximization—have recognized the importance of quantifying social influence *separately for each topic* [4]–[12]. The interactions (e.g., following and friendship) in social networks are the important indicators of social influence. Technical challenges arise because there is no way of declaring the reason for an interaction in traditional social networking services. For example, in Twitter, a following from a user *A* to a user *B* means that the user *B* influenced the user *A* but does *not* say which aspect of the user *B* played a major role in this following. The models for *topic-sensitive* influence basically attempt to infer the main reason for each interaction from the topic perspective.

A widely-accepted assumption in this line of research is that interactions are made between the users having interests in similar topics [5]–[10]. We call it the *common-interest assumption*. This assumption works well for several scenarios. For example, in a coauthorship network, an author tends to collaborate with other authors working on the same topic. However, interactions could occur even if there is no common interest [11]. For example, a user may be willing to follow a celebrity on Twitter only because the celebrity is popular. Precisely inferring the reason for an interaction is inherently challenging and yet to be figured out, because it cannot be modeled by a simple assumption and is dependent on each user's individual characteristics.

A. Social Curation Services

In social curation services, the users are eager to express their interests and opinions on contents as well as interactions. First, the users collect and organize Web contents (e.g., photos, news articles, posts, and bookmarks) such that those sharing a common topic are displayed within the same collection. Such a collection is called a *board* in Pinterest.[3] For example, one can store (or "pin") the photos relevant to "jewelry" into a board. Then, the users can show their *topic-level* interests by referring to a particular board. For example, suppose that a user *C* has two boards on "jewelry" and "dress," respectively. Another user *D* can follow the "jewelry" board rather than the user *C* to express the reason for this following—the curated content about jewelry.

The emergence of social curation services calls for a new paradigm of *topic-sensitive* influence analysis. The existing studies are less important for social curation services because the topic distribution of a user's content as well as the strength of a relationship between users per topic, which these studies attempt to extract, are explicitly available in social curation services. The curated information is very *reliable* since it is created by the users themselves—not by inference. Furthermore, it contains *fine-grained* curation of the contents and interactions, as multiple boards are allowed to exist for the same topic. Therefore, our novel model should fully take advantage of this rich and reliable information to find topic-sensitive influence.

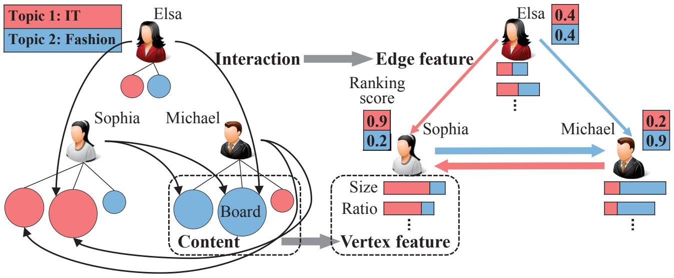
∗ Jae-Gil Lee is the corresponding author.

[1] https://www.pinterest.com/

[2] http://www.pewinternet.org/2015/01/09/social-media-update-2014/

[3] We prefer to use the terminology used in Pinterest which is the most popular social curation service.

978-1-5090-2020-1/16/$31.00 © 2016 IEEE



(a) Social curation service. (b) The CCRF.

Fig. 1: The key concept of the TISC model.

B. Contributions

In this paper, we propose a novel model of *topical influence with social curation*, which we call the *TISC* model, to find *influential users* from social curation services. The TISC model formulates our problem of finding influential users as a *global ranking* problem [13]. As opposed to local ranking that considers only a single object, global ranking considers the entire set of objects as well as the relationships between them. It is self-evident that global ranking fits better with our problem than local ranking since influence should be determined on the basis of not only the contents of users but also the interactions between users. Thus, given a topic and the *entire* network of a social curation service, we execute a *learning to rank* method to calculate the ranking scores that indicate how much each user in the network is influential in that topic.

We propose to employ the *continuous conditional random field (CCRF)* [14] for conducting the learning task. The CCRF computes a conditional probability distribution over the ranking scores of objects (users) conditioned on the objects (users). It is a perfect match for our problem since it allows us to use both the content information of objects and the relation information between objects [15]. Our model is based on supervised learning, consisting of the *learning* and *inference* tasks; the former decides the parameters of the model using a training data set, and the latter predicts the scores of the users according to the trained model. The learning task is processed by maximum likelihood estimation [16].

The TISC model satisfies two requirements necessary for social curation services. First, TISC fully takes advantage of the topic-level interests reflected in both contents and interactions. The state-of-the-art models have limitations especially in harnessing contents and use them just for distilling the topic distributions of users [6], [8]. In the motivating example by Tang et al. [6], their model calculates the *probability* on the topics "data mining" and "database." Thus, a user who wrote 100 articles on both topics cannot be differentiated from another user who wrote 10 articles on both topics. Second, TISC is free from the common-interest assumption, since it uses real topic-level interactions. As far as we know, there is no previous work satisfying both of these requirements.

Figure 1 shows the key concept of the TISC model. In Figure 1(a), topic-level contents are manifested in the boards explicitly created by users, and topic-level interactions are manifested in the followings of other users' boards. Here, the color and diameter of a circle indicate the topic and size of the board. In Figure 1(b), the content information is modeled as the *vertex* features of the CCRF, and the interaction information is modeled as the *edge* features of the CCRF. Here, the size of a bar and the width of an arrow represent the values of the features. Using these two types of features, the ranking scores are inferred for each user and each topic.

The source code of TISC is available at https://github.com/jaegil/Topical-Influence. In addition to the development of the **TISC model**, the contributions of this paper include:

1. **Empirical study**: To emphasize the need for the TISC model, we perform an in-depth analysis on the two real-world data sets from Pinterest and Scoop.it, showing that the common-interest assumption does *not* universally hold.
2. **Hierarchical influence graph**: To incorporate all the information—both content and interaction—into the TISC model, we propose a *hierarchical influence graph (HIG)* that represents a social curation service.
3. **Evaluation**: To demonstrate the merits of the TISC model, we conduct extensive evaluation using the two real-world data sets. Our model is shown to achieve higher accuracy than four other popular models by up to about 80%.

C. Outline

The rest of this paper is organized as follows. Section II provides background knowledge for our work. Section III empirically refutes the common-interest assumption. Section IV proposes the TISC model. Section V presents the learning and inference algorithms implementing the TISC model. Section VI presents the results of evaluation. Section VII summarizes the state-of-the-art related work. Finally, Section VIII concludes this study.

II. Preliminary

A. Social Curation Services

We define a *social curation service* to be a social media service that supports the *five* features below. These features are commonly found in many popular services, including Pinterest, We Heart It, and Scoop.it. (The other features relevant to generic social media services are not described here.) Figure 2 shows the main page of a user on Pinterest for an illustration of the five features. In addition, Table I summarizes the notation used throughout this paper.

1. A *user* $u \in \mathbb{U}$ creates a set of boards $\mathbb{B}_u$ (see **A** and **B**).
2. A *board* $b_u \in \mathbb{B}_u$ is a set of items $\mathbb{I}_{b_u}$ on a common topic (see **C**).
3. An *item* $i_{b_u} \in \mathbb{I}_{b_u}$ is a basic unit of curated contents.
4. A *topic* $t \in \mathbb{T}$ is attached to each board, explaining the theme of the board.
5. A user u can follow / unfollow either a user v or a board of v $(b_v \in \mathbb{B}_v)$. Following a user is regarded as following *all* of the user's boards. This topic-level interaction, which allows us to follow a board, is the main ingredient for breaking the common-interest assumption (see **D** and **E**).

B. Continuous Conditional Random Field (CCRF)

The *conditional random field (CRF)* [17] is a statistical modeling method, which is often used for relational learning (or structured learning). Relational learning deals with the cases in which statistical dependency exists between objects and each object has a rich set of features that can

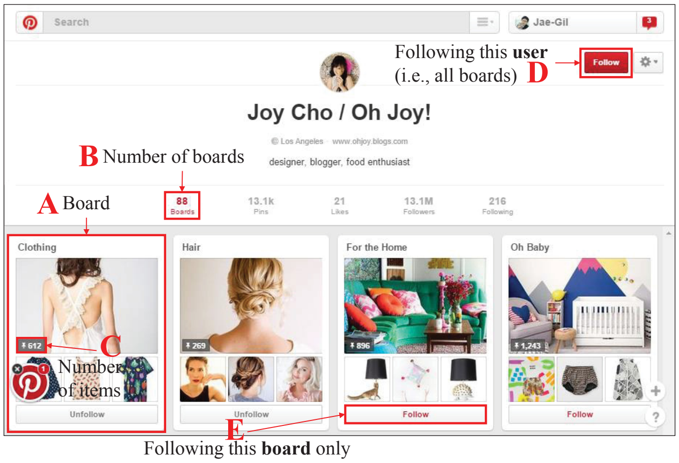


Fig. 2: A main page of a user on Pinterest.

TABLE I: Summary of notations used in this paper.

Notation	Description
$\mathbb{U}$	the set of users in a social curation service
u, v, u_i	a specific user (i.e., $u, v, u_i \in \mathbb{U}$)
$\mathbb{B}_u$	the set of boards that a user $u \in \mathbb{U}$ created
b_u, b_v, b_{u_i}	a specific board of the corresponding user
$\mathbb{I}_{b_u}$	the set of items stored in a board $b_u \in \mathbb{B}_u$
$\mathbb{F}_{b_u}$	the set of users who follow a board $b_u \in \mathbb{B}_u$
$\mathbb{FU}_u$	the set of users that a user $u \in \mathbb{U}$ follows
$\mathbb{FB}_u$	the set of all boards that a user $u \in \mathbb{U}$ follows
$\mathbb{FB}_{u,v}$	the set of a user v's boards that a user u follows
$\mathbb{T}$	the set of topics
$T(b_u)$	the topic attached to a board b_u

aid classification or ranking [15]. In this paper, the curated contents as in Figure 2 provide much information about the degree of influence, and the interactions of *following* define the relationships (between users) that can improve ranking. A *graphical model* is a natural formalism for exploiting such a dependency among objects because it can easily express the conditional dependency between random variables. Thus, a CRF as a graphical model associates a conditional distribution $P(\boldsymbol{y}|\boldsymbol{X})$ with a graphical structure, where $\boldsymbol{y}$ is the set of variables we want to predict and $\boldsymbol{X}$ is the set of observed variables [15]. However, only discrete values can be assigned to $\boldsymbol{y}$ in the CRF, while discrete values are insufficient to precisely represent ranking scores. Therefore, *continuous* conditional random field (CCRF) [14] has been proposed as an extension of the CRF to allow continuous values of $\boldsymbol{y}$.

Figure 3 depicts a graphical model of the CCRF. It is an *undirected* graph. A gray vertex ($x_{i,k}$) represents the k-th *input variable (feature)* of the i-th object, and a white vertex (y_i) represents the *output variable* of the i-th object. A solid edge between two white vertices represents the dependency between output variables, and a dotted edge between a white vertex and a gray vertex represents the dependency between an input feature and an output variable.

The CCRF is formulated by the density function in Eq. (1), where $\boldsymbol{X}(= \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}\})$ is a set of input feature vectors and $\boldsymbol{y}(= \{y_1, y_2, \ldots, y_n\})$ is a set of output variables. The denominator is a normalization constant which makes the probability distribution valid. In Eq. (2), $f_k(k = 1, 2, \ldots, K_1)$ models the k-th feature, depicted by the dotted lines in Figure 3, and $g_k(k = 1, 2, \ldots, K_2)$ models the k-th relationship, depicted by the solid lines. f_k is called a *vertex feature function*, and g_k is called an *edge feature function*. These two types of feature functions for our problem will be elaborated in Section IV.

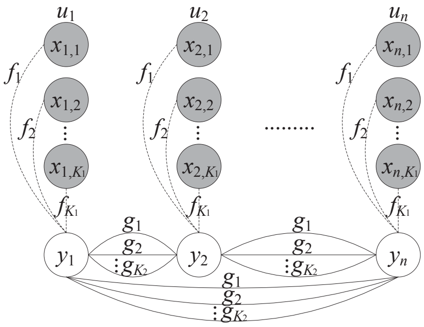


Fig. 3: A graphical representation of the CCRF [18].

$$P(\boldsymbol{y}|\boldsymbol{X}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{\exp(\Psi)}{\int_{-\infty}^{\infty} \exp(\Psi) d\boldsymbol{y}}, \text{ where} \tag{1}$$

$$\Psi = \sum_i \sum_{k=1}^{K_1} \alpha_k f_k(y_i, \boldsymbol{X}) + \sum_{i,j} \sum_{k=1}^{K_2} \beta_k g_k(y_i, y_j, \boldsymbol{X}) \tag{2}$$

The model parameters $\boldsymbol{\alpha}(= \{\alpha_1, \alpha_2, \ldots, \alpha_{K1}\})$ and $\boldsymbol{\beta}(= \{\beta_1, \beta_2, \ldots, \beta_{K2}\})$ need to be estimated through learning and provided for inference. α_k and β_k represent the importance of the corresponding dependency in vertex features and edge features, respectively. The inference task selects $\boldsymbol{y}$ that maximizes $P(\boldsymbol{y}|\boldsymbol{X})$ using the estimated values of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The learning and inference tasks for our problem will be elaborated in Section V.

III. Empirical Study on Interactions

A. Overview of the Data Sets

Two real-world data sets were used. One was crawled from Pinterest (https://www.pinterest.com) during the period of June to August in 2015 using the Pinterest API, and the other was crawled from Scoop.it (http://www.scoop.it) during the period of July to September in 2013 using the Scoop.it API. The size of the raw data before preprocessing reached 13.0 Gbytes in Pinterest and 7.5 Gbytes in Scoop.it. For Pinterest, owing to a huge number of users and items, we conducted sampling when expanding the set of users and items to collect. First, the number of followers to sample was determined using Eq. (3) as Lim et al. [19] did. Here, n_{fol} denotes the number of followers of a user, and $\langle n_{fol} \rangle$ denotes its average for all users. Eq. (3) is meant for sampling only the *hub* users with a large number of followers. Second, 10% of items were randomly sampled from each board. We did not retrieve the detailed contents of items such as photos and news texts. The statistics of the two data sets are summarized in Table II.

$$n_{sample} = \min(n_{fol}, 0.1\langle n_{fol} \rangle + \ln n_{fol}) \tag{3}$$

Please recall that social curation services allow us to express topic-level interests in boards. Figure 4 shows the proportion of the curated items belonging to each topic in our Pinterest data set. Popular topics include "design," "home decoration," "women's fashion," and so on.

TABLE II: Statistics of the two data sets.

	Pinterest	Scoop.it
# All Vertices	4,405,821	10,065,778
# Users	8,624	9,325
# Boards	212,805	25,761
# Items	4,184,392	10,030,692
# All Edges	4,892,902	10,338,607
# Following Edges	495,705	282,154
# User←Board Edges	212,805	25,761
# Board←Item Edges	4,184,392	10,030,692

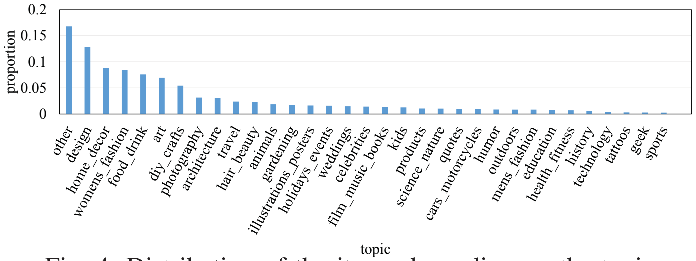
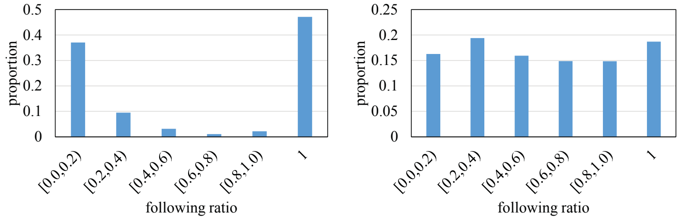


Fig. 4: Distribution of the items depending on the topics.



(a) Pinterest. (b) Scoop.it.

Fig. 5: Proportion of following the boards of a followee.

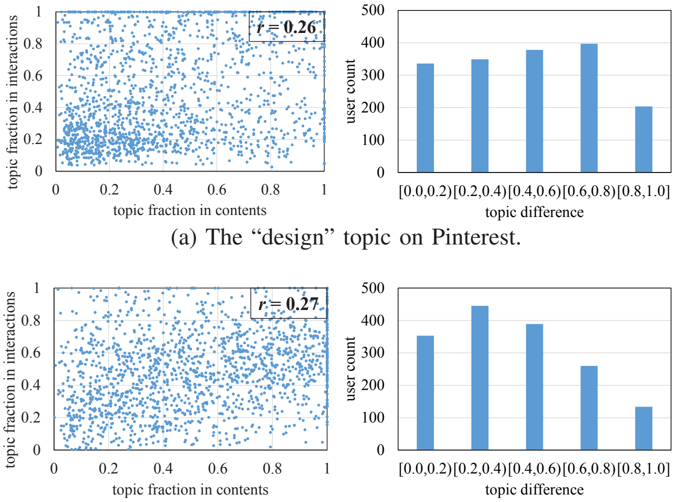
Figure 5 shows the interaction behaviors of the users on the two services. The x axis, denoted by *following ratio*, represents the ratio of the number of the boards followed to that of all the boards of a followee in each follower-followee relationship; the y axis represents the proportion of each bin. In Figure 5(a) for Pinterest, most users follow either all boards or very few boards. A little high proportion of *following ratio* = 1 is due to the interface for following the entire set of boards at once in Pinterest. However, since there is no such interface in Scoop.it, the users are evenly distributed to the entire range in Figure 5(b). Overall, in the two services, about 50%∼80% of the users *selectively* follow the boards, thereby expressing their topic-level interests explicitly.

B. Refutation of the Common-Interest Assumption

We now discuss the common-interest assumption based on the empirical study whose results are shown in Figure 6. The *topic fraction* in the scatter plots is defined as the fraction of the boards on a given topic per user, and these fractions are measured for the boards that the user has created (x-axis) and those that the user has followed (y-axis), respectively. The *topic difference* in the histograms is defined by Eq. (4), where $frac\_x$ and $frac\_y$ indicate the values of the x and y axes of the scatter plots, respectively, for a user.

$$topic\_diff = \frac{|frac\_x - frac\_y|}{\max(frac\_x, frac\_y)} \qquad (4)$$

Figure 6 shows the results for the "design" topic on Pinterest and the "education" topic on Scoop.it. If the common-interest assumption were true, the points in the scatter plots would be concentrated on the diagonal. However, the points tend to spread out on the plane. The Pearson correlation coefficients are 0.26 and 0.27, respectively. Also, in these histograms, 50∼59% of the users have very distinct (≥ 0.4) behaviors in creating and following the boards. Therefore, we assert that the common-interest assumption is *not* valid (at least in our real-world data sets). One possible explanation is that people may follow other people to complement their insufficiency of knowledge [20]. The benefits of not relying on the assumption will be discussed in Section VI.



(a) The "design" topic on Pinterest.

(b) The "education" topic on Scoop.it.

Fig. 6: Some results to refute the common-interest assumption.

IV. TISC: Topical Influence Model

A. Data Representation

A hierarchical influence graph (HIG) is a set of directed rooted trees, as illustrated in Figure 7. It represents (i) the curated contents and (ii) the topic-level interactions between users. Since the users tend to organize contents into a hierarchy, it is natural to model each user's contents using a directed rooted tree. Here, the root indicates a user, the intermediate vertices indicate the boards, and the leaf vertices indicate the items. The *level* of a vertex is defined as one larger than the number of links from the vertex to the root. The vertices located at the same level represent the same type of objects in social curation services. Then, these trees are interconnected to represent the interactions between users. We formally define the HIG in Definition 1.

Definition *1:* A *hierarchical influence graph (HIG)* is $\mathbb{G} = (\mathbb{V}, \mathbb{E}, \mathbb{A}, \boldsymbol{w})$, where $\mathbb{V}$, $\mathbb{E}$, $\mathbb{A}$, and $\boldsymbol{w}$ are defined as follows.

- $\mathbb{V} = V_{L_1} \cup V_{L_2} \cup \ldots \cup V_{L_H}$: V_{L_i} is the set of vertices located at the level L_i. V_{L_1} represents the users; V_{L_2} represents the largest boards, V_{L_3} represents the next largest boards, and so on; V_{L_H} represents the items.
- $\mathbb{E} = (E_{L_{2,1}} \cup E_{L_{3,2}} \cup \ldots \cup E_{L_{H,H-1}}) \cup E_{int}$:
  - ○ $E_{L_{i,i-1}}$ $(2 \leq i \leq H)$ is the set of directed edges from $v_{L_i} \in V_{L_i}$ to $v_{L_{i-1}} \in V_{L_{i-1}}$. Each edge represents the membership between the vertices in consecutive levels. For example, with $H = 3$, $E_{L_{2,1}}$ connects boards to the user who owns them, and $E_{L_{3,2}}$ connects items to the board which contains them. Thus, each edge $E_{L_{i,i-1}}$

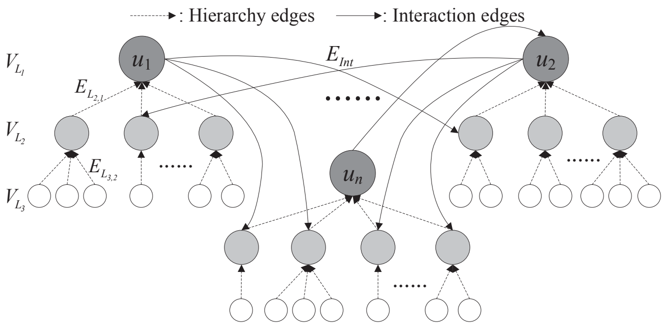


Fig. 7: A hierarchical influence graph (HIG) where $H = 3$.

represents a many-to-one relationship. These edges are collectively called *hierarchy edges*.

- ○ E_{int} is the set of directed edges from $v_{L_1} \in V_{L_1}$ to $v_{L_i} \in V_{L_i}\,(1 \le i \le H)$ such that v_{L_1} and v_{L_i} are not in the same hierarchy. Each edge represents the user's interests in other users or their contents. Note that these interests can be expressed in objects at any level. Particularly, if an edge is connected to V_{L_2}, it is said to be a *topic-level* interest. These edges are collectively called *interaction edges*.

- $\mathbb{A} = \boldsymbol{A}_{L_1} \cup \boldsymbol{A}_{L_2} \cup \ldots \cup \boldsymbol{A}_{L_H}$: $\boldsymbol{a}_{L_i} \in \boldsymbol{A}_{L_i}$ is a vector of the attributes of the corresponding vertex $v_{L_i} \in V_{L_i}$. The specific attributes are dependent on the level of a vertex. For instance, the attributes of vertices in V_{L_2} include the *topic* of a board.
- $\boldsymbol{w} = \{w_{v_i,v_j} \,|\, (v_i, v_j) \in \mathbb{E}\}$: $\boldsymbol{w}$ is the set of the weights $(\mathbb{V} \times \mathbb{V} \rightarrow [0, 1])$ of edges which are either hierarchy or interaction edges.

Example *1:* In Figure 7, three vertices in V_{L_1} are shown, each of which represents a user. Multiple vertices in V_{L_2}, each of which represents a board, are linked to one of the users through the hierarchy edges (dotted lines). Similarly, multiple vertices in V_{L_3} are linked to one of the boards through the hierarchy edges. The interaction edges (solid lines) connect a user to either another user (e.g., $u_n \rightarrow u_2$) or another user's board (e.g., $u_1 \rightarrow u_n$'s boards). $\mathbb{A}$ and $\boldsymbol{w}$ are omitted here. □

The notations in Table I are interpreted using the HIG structure, as in Eq. (5). Suppose that $v_u, v_v \in V_{L_1}$ correspond to specific users we want to consider and $v_{b_u} \in V_{L_2}$ corresponds to a specific board.

$$\begin{aligned}
\mathbb{U} &= V_{L_1} \\
\mathbb{B}_u &= \{v_{L_2} \,|\, (v_{L_2}, v_u) \in E_{L_{2,1}} \wedge v_{L_2} \in V_{L_2}\} \\
\mathbb{I}_{b_u} &= \{v_{L_3} \,|\, (v_{L_3}, v_{b_u}) \in E_{L_{3,2}} \wedge v_{L_3} \in V_{L_3}\} \\
\mathbb{F}_{b_u} &= \{v_{L_1} \,|\, (v_{L_1}, v_{b_u}) \in E_{Int} \wedge v_{L_1} \in V_{L_1}\} \\
\mathbb{FB}_u &= \{v_{L_2} \,|\, (v_u, v_{L_2}) \in E_{Int} \wedge v_{L_2} \in V_{L_2}\} \\
\mathbb{FU}_u &= \{v_{L_1} \,|\, (v_{L_2}, v_{L_1}) \in E_{L_{2,1}} \wedge \\
&\qquad (v_{L_2} \in \mathbb{FB}_u \wedge v_{L_1} \in V_{L_1})\} \\
\mathbb{FB}_{u,v} &= \{v_{L_2} \,|\, (v_{L_2}, v_v) \in E_{L_{2,1}} \wedge v_{L_2} \in \mathbb{FB}_u\}
\end{aligned} \tag{5}$$

In subsequent sections, we relax our definition of the HIG structure without loss of generality. First, the boards exist in only a single level since no commercial service supports more than one level. Thus, the total number of levels is 3 (i.e., $H = 3$). Second, interaction edges to leaf vertices are not included since each of them carries too narrow interest. Third, for ease of analysis, an interaction edge to the root is converted to the interaction edges to its all children. Thus, interaction edges exist always from V_{L_1} to V_{L_2}.

Algorithm 1 Topical Influence Ranking

```
INPUT: Training data set $\{\mathbb{G}_{tr}, \boldsymbol{y}_{tr}\}$, Unseen data set $\mathbb{G}$, One
    or more topics $\mathbb{T}_q \subseteq \mathbb{T}$
OUTPUT: Ranking score vector $\boldsymbol{r}$
 1: $\boldsymbol{R} \leftarrow \emptyset$; /* Set of ranking score vectors */
 2: for each $t \in \mathbb{T}_q$ do
 3:     /* I. LEARNING PHASE (Section V-A) */
 4:     $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \leftarrow$ Learning($\{\mathbb{G}_{tr}, \boldsymbol{y}_{tr}\}, t$);
 5:     /* II. INFERENCE PHASE (Section V-B) */
 6:     $\boldsymbol{y}^* \leftarrow$ Inference($\mathbb{G}, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$);
 7:     $\boldsymbol{R} \leftarrow \boldsymbol{R} \cup \{(\boldsymbol{y}^*, t)\}$;
 8: end for
 9: /* Optional Merge Phase (Section V-C) */
10: $\boldsymbol{r} \leftarrow$ Combining($\boldsymbol{R}, \mathbb{G}, \mathbb{T}_q$); /* $\boldsymbol{r} \leftarrow \boldsymbol{y}^*$ if $|\mathbb{T}_q| = 1$ */
11: return $\boldsymbol{r}$
```

B. Problem Definition

As mentioned in Section I-B, we formulate our problem of finding *influential users* as global ranking. The ranking scores are obtained according to Definition 2. The order in the ranking scores does matter according to Definition 3.

Definition *2:* Given a HIG $\mathbb{G}$ and a topic $t \in \mathbb{T}$, the *topical influence* of a user $u_i \in \mathbb{U}$ is defined to be the value of y_i that satisfies Eq. (6) according to the CCRF of Eqs. (1) and (2).

$$F(\boldsymbol{X}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, P(\boldsymbol{y}|\boldsymbol{X}; \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{6}$$

Definition *3:* A user $u_i \in \mathbb{U}$ is said to be more *influential* in a topic $t \in \mathbb{T}$ than a user $u_j \in \mathbb{U}$ if $y_i > y_j$.

Algorithm 1 shows the overall procedure for our TISC model. The algorithm receives a training data set $\mathbb{G}_{tr}$ as well as an unseen data set $\mathbb{G}$ from which we want to measure the influence. Both data sets are represented as HIG's. The true output values $\boldsymbol{y}_{tr}$ are known for the users in the training data set, whereas they are not in the unseen data set. For a specific topic, our algorithm learns the optimal values of the model parameters using the training data set (Lines 3 and 4) and infers the output values by applying the trained model to the unseen data set (Lines 5 and 6). If multiple topics need to be considered, the algorithm combines the scores obtained for each topic into the overall score (Lines 9 and 10).

C. Input Features for Model Definition

We need to instantiate Eqs. (1) and (2) for learning and inference. There are two types of feature functions for the CCRF: a vertex feature function $f_k(y_i, \boldsymbol{X})$ in Eq. (7) and an edge feature function $g_k(y_i, y_j, \boldsymbol{X})$ in Eq. (8).

$$f_k(y_i, \boldsymbol{X}) \;=\; -(y_i - x_{i,k})^2 \tag{7}$$

$$g_k(y_i, y_j, \boldsymbol{X}) \;=\; -S^{(k)}_{i,j}(y_i - y_j) \tag{8}$$

In Eq. (7), for each user u_i, the difference between the ranking score y_i and each feature value $x_{i,k}$ should be as small as possible to maximize our objective function in Eq. (1). By our definition of $x_{i,k}$ that will be introduced later, the higher $x_{i,k}$'s are, the more influential a user u_i tends to be. Thus, a difference between the two values induces a penalty by converting it to a negative number. For this purpose, the square of the difference is used to make it always positive. $x_{i,k}$ is called a *vertex feature*.

In Eq. (8), for each interaction from a user u_i to a user u_j, it is preferable that u_j's ranking score y_j is higher than u_i's ranking score y_i. This design choice is due to the *status theory* [21]. In this line of theory, a positive directed link means that the creator of the link views the recipient as having higher status. Thus, unless y_j is larger than y_i, this pair of users receives a penalty corresponding to that difference. Additionally, the difference of ranking scores is weighted by $S_{i,j}^{(k)}$, which is called an *edge feature*.

In order to extract these vertex and edge features, since a HIG contains the contents about all topics, we need to select the contents and interactions *related to a topic* in which we are interested. Let $\mathbb{B}'_{u,t}$, $\mathbb{FB}'_{u,t}$, and $\mathbb{FB}'_{u,v,t}$ denote the boards to which a topic $t \in \mathbb{T}$ is attached, as in Eq. (9), where $u, v \in \mathbb{U}$. That is, they are the subsets of $\mathbb{B}_u$, $\mathbb{FB}_u$, and $\mathbb{FB}_{u,v}$, respectively, that contain only the boards related to the topic t. In addition, $\mathbb{FU}'_{u,t}$ denotes the set of users who have at least one board *on the topic* t that a user u follows. Then, the vertex and edge features are defined using those symbols.

$$\begin{aligned}
\mathbb{B}'_{u,t} &= \{b_u \,|\, T(b_u) = t \wedge b_u \in \mathbb{B}_u\} \\
\mathbb{FB}'_{u,t} &= \{b_v \,|\, T(b_v) = t \wedge b_v \in \mathbb{FB}_u\} \\
\mathbb{FB}'_{u,v,t} &= \{b_v \,|\, T(b_v) = t \wedge b_v \in \mathbb{FB}_{u,v}\} \\
\mathbb{FU}'_{u,t} &= \{v \,|\, \mathbb{FB}'_{u,v,t} \neq \emptyset \wedge v \in \mathbb{FU}_u\}
\end{aligned} \tag{9}$$

Another important benefit in addition to explicitly attached topics in social curation services is the fine-grained curation of the contents. Accordingly, we can precisely quantify the degree of a user's interest and influence in a specific topic by measuring the sizes of the boards that the user has created and followed on that topic. We refer to the number of items collected in a board as its size, as in Definition 4.

Definition *4:* The *aggregate size* of a set of boards $\mathbb{B}$ is defined by the total number of the items contained in $\mathbb{B}$, which is denoted as $size(\mathbb{B})$.

We now explain the vertex and edge features used in this paper. Note that the TISC is flexible enough to adopt more features if such information is available. That is, the applicable features are *not* limited to those explained in this section.

1) Vertex Features: The main goal of the vertex features is to relate the *contents* of a user to his/her influence on a given topic. Five features are defined in this section. For each feature, the z-score is actually used to suppress the difference in scales.

1. The first feature $x_{i,1}$ in Eq. (10) indicates the aggregated size of the boards on the topic. The higher $x_{i,1}$ is, the more items on the topic the user u_i have collected. Large-size boards show the user's own high interests and are likely to attract many other users.

$$\begin{aligned}
x_{i,1} = z_{x'_{i,1}}, \text{ where } x'_{i,1} &= \sum_{b'_{u_i,t} \in \mathbb{B}'_{u_i,t}} |\mathbb{I}_{b'_{u_i,t}}| \\
&= size(\mathbb{B}'_{u_i,t})
\end{aligned} \tag{10}$$

2. The second feature $x_{i,2}$ in Eq. (11) indicates the total number of the followers of the boards on the topic. The higher $x_{i,2}$ is, the more boards of u_i on the topic the users follow. A large number of followers demonstrate the user's high popularity to other users.

$$x_{i,2} = z_{x'_{i,2}}, \text{ where } x'_{i,2} = \sum_{b'_{u_i,t} \in \mathbb{B}'_{u_i,t}} |\mathbb{F}_{b'_{u_i,t}}| \tag{11}$$

3. The third feature $x_{i,3}$ in Eq. (12) is the product of the previous two features, which is to consider the size of curated contents and the number of followers together.

$$x_{i,3} = z_{x'_{i,3}}, \text{ where } x'_{i,3} = x'_{i,1} \cdot x'_{i,2} \tag{12}$$

4. The fourth feature $x_{i,4}$ in Eq. (13) indicates the ratio of the aggregate size of the user u_i's boards on the topic to the aggregate size of all of u_i's boards. That is, $x_{i,4}$ is the *relative* value of $x_{i,1}$.

$$\begin{aligned}
x_{i,4} = z_{x'_{i,4}}, \text{ where } x'_{i,4} &= x'_{i,1} \Big/ \sum_{b_{u_i} \in \mathbb{B}_{u_i}} |\mathbb{I}_{b_{u_i}}| \\
&= x'_{i,1} / size(\mathbb{B}_{u_i})
\end{aligned} \tag{13}$$

5. The fifth feature $x_{i,5}$ in Eq. (14) indicates the ratio of the number of the followers of the user u_i's boards on the topic to the number of the followers of all of u_i's boards. That is, $x_{i,5}$ is the *relative* value of $x_{i,2}$.

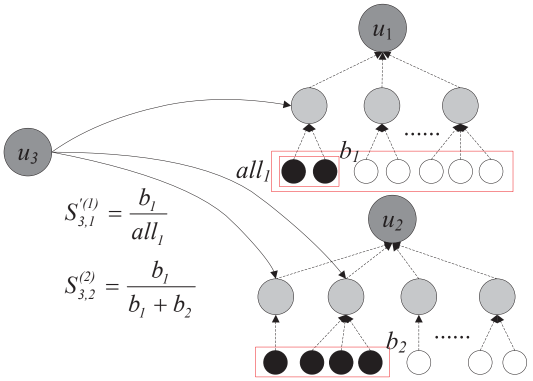
$$x_{i,5} = z_{x'_{i,5}}, \text{ where } x'_{i,5} = x'_{i,2} \Big/ \sum_{b_{u_i} \in \mathbb{B}_{u_i}} |\mathbb{F}_{b_{u_i}}| \tag{14}$$

2) Edge Features: The main goal of the edge features is to relate the *interactions* of a user to his/her influence on a given topic. Two features are defined in this section, as depicted in Figure 8.

1. The first feature $S_{i,j}^{(1)}$ in Eq. (15) indicates the normalized fraction of the boards of a user u_j that a user u_i follows on a topic t among all the boards of the user u_j. Simply speaking, the fraction ($S_{i,j}^{\prime(1)}$) represents what proportion of u_j's contents u_i likes. For example, $S_{3,1}^{\text{'}(1)}$ is illustrated in Figure 8. These fractions are normalized to $S_{i,j}^{(1)}$ such that those originating from u_i are summed to unity.

$$\begin{aligned}
S_{i,j}^{(1)} &= \frac{1}{\sum_{u_k \in \mathbb{FU}'_{u_i,t}} S_{i,k}^{\prime(1)}} \cdot S_{i,j}^{\prime(1)}, \text{ where} \\
S_{i,j}^{\prime(1)} &= \underbrace{\sum_{b'_{u_j,t} \in \mathbb{FB}'_{u_i,u_j,t}} |\mathbb{I}_{b'_{u_j,t}}|}_{b_1 \text{ in Figure 8}} \Big/ \underbrace{\sum_{b_{u_j} \in \mathbb{B}_{u_j}} |\mathbb{I}_{b_{u_j}}|}_{all_1 \text{ in Figure 8}} \\
&= size(\mathbb{FB}'_{u_i,u_j,t}) / size(\mathbb{B}_{u_j})
\end{aligned} \tag{15}$$

2. The second feature $S_{i,j}^{(2)}$ in Eq. (16) indicates the fraction of the boards of a user u_j that a user u_i follows on a topic t among all the boards that a user u_i follows on the same topic. Simply speaking, it represents what portion of the contents u_i likes is from u_j's contents, among all contents u_i likes. For example, $S_{3,2}^{(2)}$ is illustrated in Figure 8. Those fractions originating from u_i are naturally summed to unity.



u_1
u_3
b_1
all_1
u_2
$S'^{(1)}_{3,1} = \frac{b_1}{all_1}$
$S^{(2)}_{3,2} = \frac{b_1}{b_1 + b_2}$
b_2
......
......

Fig. 8: Description of the two edge features.

$$S^{(2)}_{i,j} = \underbrace{\sum_{b'_{u_j,t} \in \mathbb{FB}'_{u_i,u_j,t}} |\mathbb{I}_{b'_{u_j,t}}|}_{b_1 \text{ in Figure 8}} \Bigg/ \underbrace{\sum_{u_k \in \mathbb{FU}'_{u_i,t}} \left(\sum_{b'_{u_k,t} \in \mathbb{FB}'_{u_i,u_k,t}} |\mathbb{I}_{b'_{u_k,t}}| \right)}_{b_1 + b_2 \text{ in Figure 8}}$$
$$= size(\mathbb{FB}'_{u_i,u_j,t}) \Bigg/ \sum_{u_k \in \mathbb{FU}'_{u_i,t}} size(\mathbb{FB}'_{u_i,u_k,t}) \tag{16}$$

D. Summary

Putting Eqs. (10)∼(16) all together, the TISC model maximizes Eq. (17), where $K_1 = 5$ and $K_2 = 2$.

$$P(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\alpha},\boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{X})} \exp(\Psi), \text{ where}$$
$$Z(\boldsymbol{X}) = \int_{-\infty}^{\infty} \exp(\Psi) d\boldsymbol{y} \quad \text{and}$$
$$\Psi = \sum_i \sum_{k=1}^{K_1} -\alpha_k (y_i - x_{i,k})^2 + \sum_{i,j} \sum_{k=1}^{K_2} -\beta_k S^k_{i,j}(y_i - y_j) \tag{17}$$

V. TISC Implementations

This section discusses the implementation of the TISC model regarding the learning and inference procedures. In the interest of space, some detailed steps of algebraic derivations are omitted.

We first obtain Eq. (18), which is a detailed form of Eq. (17), by substituting Ψ into $Z(\boldsymbol{X})$. Here, n is the number of users for the given topic, $a = \boldsymbol{\alpha}^T \boldsymbol{e}$ where $\boldsymbol{e}$ is a vector space basis, $\boldsymbol{b} = 2\boldsymbol{X}\boldsymbol{\alpha} + \sum_{k=1}^{K_2} \beta_k (\boldsymbol{D}^k_r - \boldsymbol{D}^k_c)\boldsymbol{e}$ where $\boldsymbol{D}^k_r$ and $\boldsymbol{D}^k_c$ are the diagonal matrices with $D^k_{r\,i,i} = \sum_j S^k_{j,i}$ and $D^k_{c\,i,i} = \sum_j S^k_{i,j}$, respectively, and $c = \sum_i \sum_{k=1}^{K1} \alpha_k x^2_{i,k}$.

$$Z(\boldsymbol{X}) = (2a)^{-\frac{n}{2}} (2\pi)^{\frac{n}{2}} \exp(\frac{1}{4a} \boldsymbol{b}^T \boldsymbol{b} - c) \tag{18}$$

A. Model Learning for a Single Topic

The values of the parameters $\boldsymbol{\alpha}(= \{\alpha_1, \alpha_2, \ldots, \alpha_{K1}\})$ and $\boldsymbol{\beta}(= \{\beta_1, \beta_2, \ldots, \beta_{K2}\})$ of Eq. (17) are estimated on a training data set $\{\boldsymbol{X}, \boldsymbol{y}\}$, where $\boldsymbol{X}(= \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}\})$ is a set of input feature vectors of n users and $\boldsymbol{y}(= \{y_1, y_2, \ldots, y_n\})$ is a set of true output values of the n users. $\boldsymbol{X}$ is a feature *matrix*, where $x_{i,k}$ represents the k-th feature of the user u_i. A ranking score $y_i (i = 1, 2, \ldots, n)$ can be a real number.

As mentioned earlier, we employ the maximum likelihood estimation (MLE) to find the optimal values of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ that maximize Eq. (17) through Eqs. (19) and (20). $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$ are the maximum likelihood estimators of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ respectively. Specifically, we calculate the conditional log likelihood with respect to the CCRF, as in Eq. (20).

$$(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \underset{\boldsymbol{\alpha},\boldsymbol{\beta}}{\operatorname{argmax}} \mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}), \text{ where} \tag{19}$$
$$\mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \ln P(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\alpha},\boldsymbol{\beta}) \tag{20}$$

Then, we adopt stochastic gradient ascent (SGA) [16] to maximize the log likelihood. Since it finds a *local* maximum, we prove that our log-likelihood function is concave in Theorem 1 to guarantee a *global* maximum.

Theorem *1:* $\mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ in Eq. (20) is a concave function.

Proof: By substituting $Z(\boldsymbol{X})$ from Eq. (18) into Eq. (17) and in turn $P(\boldsymbol{y}|\boldsymbol{X};\boldsymbol{\alpha},\boldsymbol{\beta})$ into Eq. (20), we obtain $\mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{n}{2}\ln(2a) - \frac{n}{2}\ln(2\pi) - \frac{1}{4a}\boldsymbol{b}^T\boldsymbol{b} + c + \sum_i \sum_{k=1}^{K1} -\alpha_k (y_i - x_{i,k})^2 + \sum_{i,j} \sum_{k=1}^{K2} -\beta_k S^k_{i,j}(y_i - y_j)$. We only need to check the two terms $\frac{n}{2}\ln(2a)$ and $-\frac{1}{4a}\boldsymbol{b}^T\boldsymbol{b}$, because the other terms are affine with $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. First, since $\ln(\cdot)$ is a concave function and $a(= \boldsymbol{\alpha}^T \boldsymbol{e})$ is affine with $\boldsymbol{\alpha}$, $\ln(2a)$ is concave on $\boldsymbol{\alpha}$. Second, $\frac{1}{a}\boldsymbol{b}^T\boldsymbol{b}(= \frac{b_1^2}{a} + \ldots + \frac{b_n^2}{a})$ is convex on $(a, \boldsymbol{b})$, because it is a sum of convex terms, and in turn $(a, \boldsymbol{b})$ is affine with $(\boldsymbol{\alpha}, \boldsymbol{\beta})$; hence, $\frac{1}{a}\boldsymbol{b}^T\boldsymbol{b}$ is convex on $(\boldsymbol{\alpha}, \boldsymbol{\beta})$. Consequently, $-\frac{1}{4a}\boldsymbol{b}^T\boldsymbol{b}$ is concave on $(\boldsymbol{\alpha}, \boldsymbol{\beta})$. In summary, since both terms are concave and the rest of the terms together are affine, $\mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is concave. □

It is necessary that $\alpha_k > 0 \, (k = 1, 2, \ldots, K_1)$ in order to make the integral of $\exp(\Psi)$ valid, i.e., for the use of the Gaussian integral. Gradient ascent cannot be directly applied to such a constrained optimization problem ($\alpha_k > 0$). Thus, we maximize $\mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ with respect to $\ln \alpha_k$ instead of α_k as in other work using the CCRF [14], [18].

Algorithm 2 shows the procedures of this MLE using stochastic gradient ascent. The values of the two parameters $\ln \alpha_k$ and β_k are updated by their gradients in each iteration of the algorithm (Lines 4∼11), and it repeats the same procedure until the parameter values reach a convergence point at which the relative size of the updates measured by the Euclidean norm is below a convergence threshold δ (Line 12). Note that in our experiment the data have been crawled *randomly* and thus there is no need to shuffle the training data randomly in the algorithm.[4] The random shuffling was omitted in other work [14], [18] as well. v is called a *learning rate* and determines the size of each step (Lines 6 and 10). The values of δ and v are usually determined empirically.

Eqs. (21) and (22) show how the gradients $\nabla_{\ln \alpha_k}$ and ∇_{β_k} in Lines 5 and 9 of Algorithm 2 are calculated, respectively.

$$\nabla_{\ln \alpha_k} = \frac{\partial \mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \ln \alpha_k} = \alpha_k \big(\sum_i -(y_i - x_{i,k})^2 - \frac{\partial \ln Z(\boldsymbol{X})}{\partial \alpha_k} \big) \tag{21}$$
$$\nabla_{\beta_k} = \frac{\partial \mathcal{L}(\boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \beta_k} = \sum_{i,j} S^k_{i,j}(y_j - y_i) - \frac{\partial \ln Z(\boldsymbol{X})}{\partial \beta_k} \tag{22}$$

[4] If the training data were given in some meaningful order, then it could bias the gradient and lead to poor convergence. Thus, stochastic gradient ascent would shuffle the data randomly.

Algorithm 2 Learning (with MLE using SGA)

```
INPUT: Training data set $\{\mathbb{G}_{tr}, \boldsymbol{y}_{tr}\}$, Topic $t$
OUTPUT: Optimal parameter values $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$
 1: Derive $\boldsymbol{X}$ and $\{\boldsymbol{D}_r^k, \boldsymbol{D}_c^k\}$ for $t$ from $\mathbb{G}_{tr}$;
 2: Choose initial values for parameters $\ln \alpha_k$ and $\beta_k$;
 3: repeat
 4:     for each $k \in \{1, \ldots, K_1\}$ do
 5:         Compute the gradient $\nabla_{\ln \alpha_k}$ using Eq. (21);
 6:         $\ln \alpha_k \leftarrow \ln \alpha_k + v\nabla_{\ln \alpha_k}$; /* an updated $\alpha_k$ */
 7:     end for
 8:     for each $k \in \{1, \ldots, K_2\}$ do
 9:         Compute the gradient $\nabla_{\beta_k}$ using Eq. (22);
10:         $\beta_k \leftarrow \beta_k + v\nabla_{\beta_k}$; /* an updated $\beta_k$ */
11:     end for
12: until $\frac{||(\nabla_{\ln \boldsymbol{\alpha}}, \nabla_{\boldsymbol{\beta}})||}{||(\ln \boldsymbol{\alpha}, \boldsymbol{\beta})||} < \delta$ /* convergence condition */
13: $\boldsymbol{\alpha}^* \leftarrow \{\alpha_1, \ldots, \alpha_{K1}\}$, $\boldsymbol{\beta}^* \leftarrow \{\beta_1, \ldots, \beta_{K2}\}$;
14: return $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$;
```

Here, the partial derivatives $\frac{\partial \ln Z(\boldsymbol{X})}{\partial \alpha_k}$ and $\frac{\partial \ln Z(\boldsymbol{X})}{\partial \beta_k}$ are calculated through Eqs. (23) and (24), respectively. $\boldsymbol{X}_{\cdot,k}$ in Eq. (23) denotes the k-th column of the matrix $\boldsymbol{X}$.

$$\frac{\partial \ln Z(\boldsymbol{X})}{\partial \alpha_k} = -\frac{n}{2a} - \frac{1}{4a^2}\boldsymbol{b}^T\boldsymbol{b} + \frac{1}{2a}\boldsymbol{b}^T\boldsymbol{X}_{\cdot,k} - \sum_i x_{i,k}^2 \quad (23)$$

$$\frac{\partial \ln Z(\boldsymbol{X})}{\partial \beta_k} = \frac{1}{2a}\boldsymbol{b}^T(\boldsymbol{D}_r^k - \boldsymbol{D}_c^k)\boldsymbol{e} \quad (24)$$

Theorem *2:* The running time complexity of TISC learning (Algorithm 2) is $O(n)$ where n is the number of users.

Proof: First, in Line 5, computing each gradient $\nabla_{\ln \alpha_k}$ using Eqs. (21) and (23) takes $O(n)$. Normally Eq. (23) would take $O(n^2)$ because of $\boldsymbol{b}$ that requires a multiplication of an $n \times n$ matrix $(\boldsymbol{D}_r^k - \boldsymbol{D}_c^k)$ and an n-vector $\boldsymbol{e}$, but since $\boldsymbol{D}_r^k$ and $\boldsymbol{D}_c^k$ are diagonal matrices in this case, the running time is reduced to $O(n)$ for computing $\sum_{i=1}^{n}(D_{r\,i,i}^k - D_{c\,i,i}^k)e_i$. Thus, computing all $\nabla_{\ln \alpha_k}$ for $k = 1, 2, \ldots, K_1$ takes $O(nK_1)$. Also, in Line 9, computing each gradient ∇_{β_k} using Eqs. (22) and (24) takes $O(n)$ as well. $\sum_{i,j} S_{i,j}^k(y_j - y_i)$ in Eq. (22) takes approximately $O(n)$ since $\langle d \rangle \ll n$ where $\langle d \rangle$ is the average degree. Thus, computing all ∇_{β_k} for $k = 1, 2, ..., K_2$ takes $O(nK_2)$. Second, Lines 6 and 10 take $O(K_1 + K_2)$ altogether. Hence, if τ iterations are executed, the entire algorithm as a whole takes $O(\tau n(K_1 + K_2))$, which can be simplified to $O(n)$ since $K_1 + K_2 \ll n$. □

B. Influence Score Calculation for a Single Topic

We calculate the value of $\boldsymbol{y}$ that maximizes Eq. (17) in an unseen data set using $\boldsymbol{\alpha}^*$ and $\boldsymbol{\beta}^*$ estimated in Algorithm 2. The value of $\boldsymbol{y}$ for which the derivative of Eq. (17) with respect to $\boldsymbol{y}$ is equal to zero corresponds to a maximum $\boldsymbol{y}^*$. In this way, the ranking scores are calculated using Eq. (25).

$$\begin{aligned} \boldsymbol{y}^* &= \operatorname*{argmax}_{\boldsymbol{y}} P(\boldsymbol{y}|\boldsymbol{X}; \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \\ &= \frac{1}{\boldsymbol{\alpha}^{*T}\boldsymbol{e}}(2\boldsymbol{X}\boldsymbol{\alpha}^* + \sum_{k=1}^{K_2}\beta_k(\boldsymbol{D}_r^k - \boldsymbol{D}_c^k)\boldsymbol{e}) \end{aligned} \quad (25)$$

Theorem *3:* The running time complexity of TISC inference is $O(n)$ where n is the number of users.

Proof: The first term $2\boldsymbol{X}\boldsymbol{\alpha}^*$ in Eq. (25) is a multiplication of an $n \times K_1$ matrix and a K_1-vector, and therefore takes $O(nK_1)$. The second term $\sum_{k=1}^{K_2}\beta_k(\boldsymbol{D}_r^k - \boldsymbol{D}_c^k)\boldsymbol{e}$ in Eq. (25) takes $O(nK_2)$ because each summand $\beta_k(\boldsymbol{D}_r^k - \boldsymbol{D}_c^k)\boldsymbol{e}$, a multiplication of an $n \times n$ matrix and an n-vector, can be done in $O(n)$ since $\boldsymbol{D}_r^k$ and $\boldsymbol{D}_c^k$ are diagonal matrices (as in Theorem 2). Hence, the total time complexity is $O(n(K_1 + K_2))$, which is simplified to $O(n)$ since $K_1 + K_2 \ll n$. □

C. Influence Score Combining for Multiple Topics

If one wants to find influential users with consideration of *multiple* topics $\mathbb{T}_q \subseteq \mathbb{T}$, our model needs to combine the ranking scores obtained separately for each topic $t \in \mathbb{T}_q$, as in Eq. (26), where the combined score r_i of a user u_i is a weighted sum of those individual scores. Let $y_{i,t}$ be the ranking score of the user u_i on a specific topic t and $w_{i,t}$ be his/her weight on t. This weight can be considered as the probability for the user u_i to curate an item under a board about a given topic t among all boards about $\mathbb{T}_q$.

$$r_i = \sum_{t \in \mathbb{T}_q} w_{i,t} \cdot y_{i,t}, \quad \text{where } w_{i,t} = \frac{size(\mathbb{B}'_{u_i,t})}{\sum_{t' \in \mathbb{T}_q} size(\mathbb{B}'_{u_i,t'})} \quad (26)$$

D. Distributed Learning with Spark

In order to support large-scale networks prevalent in these days of big data, we attempt to speed up the learning phase rather than the inference phase. The former is much more time-consuming than the latter despite the same time complexity, since the former involves *iterations*. More specifically, we parallelize the computation of gradient ascent in Algorithm 2 (Lines 5 and 9), which is the dominant cost step.

We choose Spark [22] as a distributed computing framework to implement *distributed* learning. Spark allows us to construct a *resilient distributed dataset (RDD)*, which is a read-only collection of objects maintained in memory of multiple machines. Spark fits our purpose perfectly since it is designed to optimize iterative and interactive computation.

Algorithm 3 presents a parallel version of Algorithm 2 on Spark (in the Python form). First, a training data set is partitioned and allocated to individual worker nodes (Line 1). Then, in the map stage, each worker node computes the gradients $\nabla_{\ln_{\alpha_k}}^m$ and then $\nabla_{\beta_k}^m$ from the allocated partition m, and in the reduce stage, the gradients computed from different partitions are aggregated to output $\nabla_{\ln_{\alpha_k}}$ and ∇_{β_k} into a vector $gradient(= \{\nabla_{\ln \alpha_1}, \ldots, \nabla_{ln\alpha_{K1}}, \nabla_{\beta_1}, \ldots, \nabla_{\beta_{K2}}\})$ (Line 5). The functions *gradient* and *add* are responsible for computing the gradients and summing them up.

While running Algorithm 3 on Spark, each worker node caches an allocated partition into the RDD. In addition, Spark keeps the same partitioning of the data set and allocates a partition to the same worker node throughout the iterations. Thus, iterative computation in this algorithm is really fast since the data set is *not* loaded repeatedly once it is cached.

VI. Evaluation

In this section, we conduct extensive experiments to evaluate the effectiveness and efficiency of our TISC model. Section VI-A explains the setting for our experiments, Section VI-B

Algorithm 3 Distributed Learning on Spark (in Python)

```
points = spark.textFile(...).MAPPARTITIONS(...);
/* Initialize a parameter vector $w$ */
$w$ = numpy.random.ranf(size = $K_1 + K_2$);
for $i$ in range(ITERATIONS) do
    $gradient$ = points.MAP(lambda $m$: gradient($m$, $w$)).REDUCE(add);
    $w$ += $v \times gradient$;
    if $||gradient||/||w|| < \delta$ then
        break;
    end if
end for
```

TABLE III: Properties of the methods compared.

	Content Use	Interaction Use	No CI Assumption	Learning Method
TSPR	×	◯	◯	Unsupervised
TwitterRank	△	◯	×	Unsupervised
TAP-TPRI	△	◯	×	Supervised
Regression	◯	△	◯	Supervised
TISC	◯	◯	◯	Supervised

presents the results for accuracy, Section VI-C presents a few cases that show the benefits of our model, and Section VI-D presents the results for scalability.

A. Experiment Setting

1) Methods: We compared our TISC model with the state-of-the-art methods of finding topic-sensitive influential users from social networks [4], [6], [8]. In addition, *linear regression* was adopted to predict the ranking scores using our vertex features. However, existing work for maximizing topic-sensitive influence (e.g., [7], [10], [12]) has a goal different from this line of research and thus is not included for comparison. Overall, the five methods below were compared with one another.

- Topic-Sensitive PageRank (TSPR) [4]
- TwitterRank [8]
- Topical Affinity Propagation for PageRank with Topic-Based Influence (TAP-TPRI) [6]
- Linear Regression (denoted as *Regression*)
- **Topical Influence with Social Curation** (**TISC**): our proposed method

Table III summarizes the properties of the five methods. Our TISC model is free from the common-interest assumption as well as fully uses both contents and interactions, whereas the other alternatives do not. (i) TwitterRank and TAP-TPRI use the contents only for extracting topic distribution, but not for directly calculating the influence of a user. (ii) Regression simply uses the number of followers, but not individual following relationships. (iii) TwitterRank and TAP-TPRI are based on the common-interest assumption.

The parameters needed in these methods were set to be typical or default values. In TISC, the learning rate v was set to be 10^{-7}, and the convergence threshold δ was set to be $5 \cdot 10^{-4}$. In TSPR, TwitterRank, and TAP-TPRI whch are based on PageRank, the damping factor α, the probability that a person will continue traversal, was set to be 0.85, and the convergence threshold δ was set to be 10^{-6}, which are the default values of the NetworkX[5] package.

All methods except TAP-TPRI were implemented using Python 2.7 with the numpy, scipy, scikit-learn, and NetworkX packages. For TAP-TPRI, we used the source code in C++ provided by the authors.

2) Data Sets: We used the two real-world data sets introduced in Section III-A. Each data set was divided into the *training* data set and the *unseen* data set. The former was a vertex-induced subgraph that consisted of 40% of the users, and the latter was a vertex-induced subgraph that consisted of the rest of the users.

As for the true output values $\boldsymbol{y}_{tr}$ in the ***training*** data set, we used the scores of boards in Scoop.it, which are calculated by the service provider. It is known that the scores consider keywords, update frequencies, amounts of sharing, and so on. Then, a user was assigned the weighted sum of the scores of the boards on the topic, where the weight is the size of a board. However, owing to lack of such scores in Pinterest, we calculated them for Pinterest in a similar manner. The resulting score is the sum of the normalized values of the numbers of relevant keywords, amounts of sharing (repin), numbers of likes, numbers of followers, and so on. In reality, true influence scores are available for only a subset of users, and this is the reason why inference needs to be done.

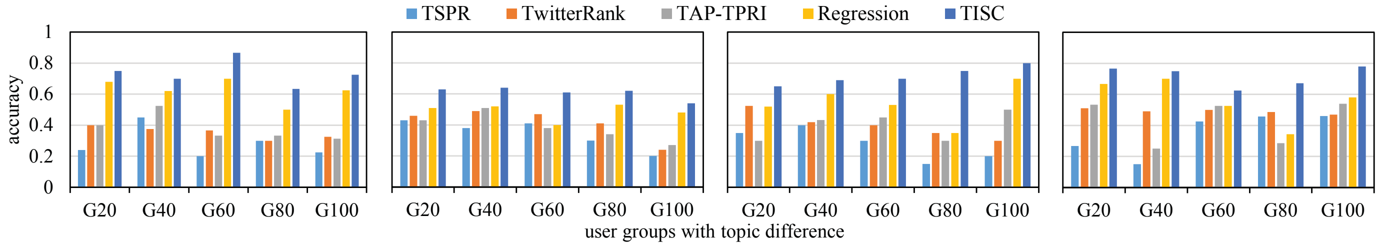
In Section VI-B, the users in the ***unseen*** data set were split into five groups according to their *topic difference* of Eq. (4). These users were sorted in the ascending order of the topic difference. Then, the users in the 20th percentile were assigned to the group "G20," those in the 20th to 40th percentile were assigned to the group "G40," and so on. Thus, the group "G100" had the *highest* topic difference. This split is intended to verify the *negative* impact of the common-interest assumption on accuracy.

In Section VI-D, in order to provide huge data sets for scalability test, we duplicated the ***entire*** original data sets in Table II by 200, 400, 600, 800, and 1,000 times, which are labled as "nx". The graph of the original data set was one giant connected component, and this component was duplicated with no connection to existing components. For instance, the largest one (i.e., "1000x") of Pinterest contained about 8.6 million users in total.

3) Configuration: All experiments except in Section VI-D were conducted on a Linux server equipped with two Xeon E5-2640 processors (2.60 GHz, 8 cores in each) and 48 Gbytes of main memory. The server ran on Ubuntu 14.04.2 LTS. For the scalability test in Section VI-D, we used twelve Microsoft Azure A3 instances located in East Asia. Each A3 instance had four cores, 7 GBytes of main memory, and 285 GBytes of hard disk. On this cluster of machines, we ran Spark 1.3.1, where the amount of main memory per executor was 1 GBytes and the storage level was "MEMORY_ONLY." One instance was dedicated to the head node of Spark, and the others were used as worker nodes. A given training data set was split into 32 partitions for parallel processing in Spark.

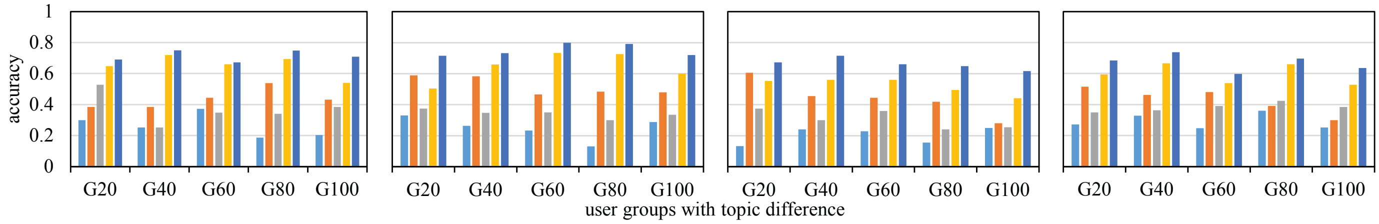
4) Fairness: For a fair comparison between our method and the other methods, we provided them with rich information in social curation services as much as possible. For TSPR, the teleport vector was precisely calculated using the number of items belonging to a given topic in each user. For TwitterRank and TAP-TPRI, the topic distribution of a user was precisely

[5] https://networkx.github.io/



(a) Design. (b) Home Decoration. (c) Women's Fashion. (d) Food & Drink.

Fig. 9: Results of ranking accuracy for the Pinterest data set.



(a) Education. (b) Art. (c) Media. (d) Audio & Video.

Fig. 10: Results of ranking accuracy for the Scoop.it data set.

Algorithm 4 Accuracy Evaluation

```
INPUT: User group, Method, Unseen data set $\mathbb{G}$
OUTPUT: Accuracy value
$E_{test} \leftarrow$ random sample of follower-followee relationships from the user group; /* typically, $|E_{test}| = 10$ */
for each $(u_i, u_j) \in E_{test}$ do
    $U_{irr} \leftarrow$ random sample of users whom $u_i$ does not follow; /* typically, $|U_{irr}| = 10$ */
    Remove all interaction edges from $u_i$ to $u_j$ in $\mathbb{G}$;
    Apply a given method to calculate $\boldsymbol{y}$;
    $accuracy \leftarrow accuracy + Q(u_j, U_{irr})$; /* Eq. (27) */
end for
return $(accuracy/|E_{test}|)$; /* the average */
```

calculated using the number of items belonging to each topic, without having to use LDA [23]. In addition, since the numbers of followers were considered to calculate the true output values, we dropped the relevant vertex features, i.e., $x_{i,2}$, $x_{i,3}$, and $x_{i,5}$, in TISC to favor the other methods.

5) Accuracy Metric: We adopted an approach suggested by Weng et al. [8] to compare the accuracy of the methods, which is explained in Algorithm 4. The existence of an interaction edge from u_i to u_j means that u_i has been already influenced by u_j. Thus, even though we remove the interaction, the ranking score of u_j had better be higher than those of the users with no previous interaction from u_i. Eq. (27) counts the number of *irrelevant* users whose ranking scores are lower than that of the *prospective* followee. The higher the value of Eq. (27) is, the more accurate a method is.

$$Q(u_f, U_{irr}) = \frac{|\{u_k | u_k \in U_{irr}, score(u_k) < score(u_f)\}|}{|U_{irr}|} \tag{27}$$

B. Accuracy Result

Figures 9 and 10 show the accuracy of the five methods on Pinterest and Scoop.it respectively. The four most popular topics were selected from each service. The x-axis indicates the user groups, and the y-axis indicates the accuracy measured using Algorithm 4. Overall, TISC outperformed other methods significantly. TISC improved accuracy by up to 62∼80%, 39∼62%, 50∼66%, and 21∼53% compared with TSPR, TwitterRank, TAP-TPRI, and Regression, respectively, in Figure 9 and by up to 60∼83%, 41∼54%, 50∼66%, and 16∼29% in Figure 10. These results indeed demonstrate the benefits of TISC, primarily from taking advantage of both contents and interactions and secondarily from being free of the common-interest assumption.

In terms of comparing the accuracy across the user groups, TISC was rather insensitive to the topic difference. In contrast, the two methods that rely on the common-interest assumption, TwitterRank and TAR-TPRI (see Table III), were affected. TwitterRank, which is heavily dependent on the assumption, showed pretty strong tendency that the average accuracy for the four topics decreased as the topic difference increased, i.e., when going from G20 to G100, from 0.47 down to 0.44 to 0.43 to 0.39 and to 0.33 in Figure 9 and from 0.52 down to 0.47 to 0.46 to 0.46 and to 0.37 in Figure 10. TAP-TPRI, which is also dependent on the assumption, showed weaker tendency, i.e., from 0.42 at G20 down to 0.31 at G80 in Figure 9 and from 0.41 at G20 down to 0.34 at G100 in Figure 10. Overall, we confirm that this assumption is one of the main reasons for incorrect prediction.

C. Case Study Result

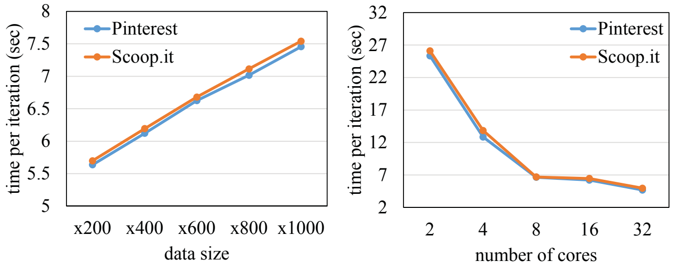
Table IV summarizes the top-5 influencers found from the unseen data set about each of the four topics on Pinterest. They are real Pinterest user names. We manually examined each of them to categorize them into three cases as below.

- *Content-type errors* (in red): the users who created less than 500 items *on a given topic*
- *Interaction-type errors* (in blue): the users who received less than 100 followings *on a given topic*
- *Correct answers* (in **bold**): none of the above

In short, the result showed that TISC did not yield any errors whereas the other methods yielded quite many errors.

TABLE IV: Top-5 influencers on Pinterest (in **bold**: correct answers, in red: content-type errors, in blue: interaction-type errors).

Topic Method	Design	Home Decoration	Women's Fashion	Food & Drink
TSPR	**designmilk**, erofili, **myan_duong**, designsponge, pennyweight	**designmilk**, designsponge, pennyweight, **ohjoy**, **myan_duong**	**designmilk**, designsponge, pennyweight, **myan_duong**, amandajanejones	designsponge, designmilk, ohjoy, pennyweight, amandajanejones
TwitterRank	**designmilk**, **myan_duong**, **codeplusform**, **aliciacarvalho**, **packagingdiva**	**designmilk**, **myan_duong**, designcrush, mollymadfis, **ohjoy**	**designmilk**, **myan_duong**, imptwitch, **codeplusform**, longinaphillips	designmilk, **myan_duong**, designsponge, imptwitch, **simplyDesigning**
TAP-TPRI	kinfolkmag, erofili, thedesignfiles, **tempspaz**, kneelandco	**designmilk**, 20x200, pennyweight, kayolgr, amandajanejones	pennyweight, **codeplusform**, missmossblog, **myan_duong**, **designmilk**	designmilk, designsponge, bianca_cash, wideeyedlegless, sugarAndCloth
Regression	designseedslove, **designmilk**, **packagingdiva**, **codeplusform**, **itoyoshi**	**designmilk**, **ohjoy**, **carlaaston**, **codeplusform**, **myan_duong**	**codeplusform**, ohjoy, **myan_duong**, **designmilk**, kaleb_willis	**packagingdiva**, **buzzfeedfood**, ohjoy, **jchongdesign**, designcrush
TISC	**packagingdiva, designmilk, thinkmule, itoyoshi, vestidadeflores**	**designmilk**, **ohjoy**, **myan_duong**, **4piccolina**, **hgdesignideas**	**levato, codeplusform, myan_duong, 4urenjoyment, designmilk**	**acurioustaste, buzzfeedfood, addapinch, codeplusform, acuriouswork**



(a) Varying the data size. (b) Varying the number of cores.

Fig. 11: Results of performance and scalability (12 machines).

It turns out that these errors are caused by the shortcomings of the other methods. First, regarding the content-type errors, TSPR does not distinguish the reasons for interactions, thus possibly assigning high scores to irrelevant users. TwitterRank considers the number of *all* items belonging to a user, not the items specific to a topic and, therefore, even though the total number of items is large, the fraction of the relevant items is small in many erroneous users. TAP-TPRI does not consider the *absolute* numbers of items, so many erroneous users have a small number of items. Second, the interaction-type errors are mainly due to the common-interest assumption, since the reasons for interactions are incorrectly inferred by the assumption. In addition, Regression does not exploit individual following relationships, unlike TISC.

D. Performance and Scalability Result

Figure 11 presents learning performance accelerated by Spark for huge duplicated data sets. The preprocessing time spent for transforming a HIG into the features is not included here. Thus, the results for the two social curation services overlap because of similar numbers of sampled users. Figure 11(a) shows the elapsed time per iteration when running Algorithm 3 with eight cores as the data size increases. The result showed linear scalability of TISC, thereby confirming Theorem 2. The elapsed time increased by only 1.3 times when the data size increased by 5 times. Figure 11(b) shows the elapsed time per iteration against the "600x" data sets as the number of cores increases. Since TISC is fully parallelizable, the elapsed time decreases with more cores, as long as there is no resource idle. The rate of decrease started slowing down after 8 cores were used. In addition, this parallel version improved performance by up to 11 times compared with a non-parallel version. Overall, we confirm that our method is capable of processing large-scale networks, thanks to its high scalability and parallelizability.

VII. Related Work

A. Topic-Sensitive Influence Analysis

In an earlier work by Haveliwala [4], the author proposed a *topic-sensitive PageRank* algorithm for crawled web pages. It enhances the conventional PageRank algorithm by preparing a set of PageRank vectors respectively biased toward different topics and uses them to calculate page importance scores specific to queries.

Topic-sensitive influence modeling is a recent trend gaining momentum particularly in social network analysis. Nallapati and Cohen [5] proposed a single framework called *Link-PLSA-LDA*, which addresses both topic discovery (based on PLSA [24]) and topic-specific influence modeling (based on Link-LDA [25]) from online blogs (available from Nielson Buzzmetrics). Tang et al. [6] proposed a *topical affinity propagation (TAP)* model to analyze social influence in a large network by the topic. Once learned in a certain network structure, the model can be used to perform topic-level influence propagation. Liu et al. [7] proposed a topic-level influence mining from *heterogeneous* networks, which contain different types of vertices such as users and documents, comprised of Twitter, Digg, and Cora. Their approach combines link information and textual content to mine direct topic-level influence between vertices and extends it to indirect chain of influence via a topic-level influence propagation model. Weng et al. [8] proposed *TwitterRank*, a topic-sensitive user influence ranking approach in view of Twitter users with reciprocal following relationships attributed to serious common topical interests. Pal and Counts [9] addressed the problem of identifying users considered authorities on a given topic in microblogs (tweets). Their approach uses probabilistic clustering (as opposed to network analysis) in a feature space defined on users' tweet activities and then ranks users within a selected target cluster. Barbier et al. [10] proposed a topic-aware influence propagation model extended from the well-known independent cascade and linear threshold models and also proposed a model focusing on how authoritative and interested a user is in a topic in order to reduce the model complexity. Chen et al. [12] proposed a topic-aware influence maximization algorithm for finding a set of seed users that maximizes topic-aware influence spread (i.e., the number of users influenced). The proven premise is that topic-awareness increases the influence spread.

None of these research done by others can take advantage of the social curation services characterized by the topic-level interests expressed by users who create their own boards and/or follow specific boards of other users instead of the users themselves. Refer to Table III, for example.

B. *Social Curation Service Analysis*

Several analyses of the social curation services appeared in recent years, mostly by Pinterest. Gilbert at el. [3] made a distinction of their service from other social networking services, particularly Twitter. Some of the reported findings include that "being female means repins, but fewer followers" and that they are distinct in terms of "use, look, want, and need." A more comprehensive study on the gender-specific user behaviors was reported by Ottoni et al. [26]. Geng et al. [27] introduced their deep learning technique for user profiling, specifically to profile models and related image features together, thereby enabling *content-based social media* technologies.

VIII. Conclusion

In this paper, we proposed the *TISC* model to find influential users from social curation services such as Pinterest and Scoop.it. These social curation services provide us with a lot of opportunities for topic-sensitive influence analysis, since the topic-level interests in both contents and interactions are actively expressed by the users, thus eliminating the need for the common-interest assumption. Our model was designed to fully take advantage of the rich and reliable information. We conducted extensive experiments to demonstrate the benefits of our model using two real-world data sets. TISC significantly outperformed other methods in terms of the accuracy of finding prospective followees, and the quality of the top-5 influencers was observed to be higher in TISC than in other methods. Furthermore, TISC was highly scalable and parallelizable, as proven by its Spark implementation. We expect that the trend of allowing users to express their interests and intentions in detail will continue, being witnessed by the growth of Pinterest and the ubiquity of hashtags on many services. Therefore, we believe that the importance of this work will continue to increase with this trend on emerging social media platforms.

Acknowledgment

This research was partly supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2015R1A1A1A05001475) and the MOLIT (The Ministry of Land, Infrastructure and Transport), Korea, under the national spatial information research program supervised by the KAIA (Korea Agency for Infrastructure Technology Advancement) (15NSIP-B081011-02).

References

[1] Wikipedia, "Content curation," https://en.wikipedia.org/wiki/Content_curation, accessed: 2015-10-19.

[2] WhatIs.com, "What is social curation?" http://whatis.techtarget.com/definition/social-curation, accessed: 2015-10-19.

[3] E. Gilbert, S. Bakhshi, S. Chang, and L. Terveen, "I need to try this!: A statistical overview of Pinterest," in *Proc. 2013 ACM SIGCHI Conf. on Human Factors in Computing Systems*, 2013, pp. 2427–2436.

[4] T. H. Haveliwala, "Topic-sensitive PageRank," in *Proc. 11th Int'l Conf. on World Wide Web*, 2002, pp. 517–526.

[5] R. Nallapati and W. W. Cohen, "Link-PLSA-LDA: A new unsupervised model for topics and influence of blogs," in *Proc. 2nd Int'l AAAI Conf. on Weblogs and Social Media*, 2008.

[6] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proc. 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, 2009, pp. 807–816.

[7] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang, "Mining topic-level influence in heterogeneous networks," in *Proc. 19th ACM Int'l Conf. on Information and Knowledge Management*, 2010, pp. 199–208.

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

[9] A. Pal and S. Counts, "Identifying topical authorities in microblogs," in *Proc. 4th ACM Int'l Conf. on Web Search and Data Mining*, 2011, pp. 45–54.

[10] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware social influence propagation models," in *Proc. 12th IEEE Int'l Conf. on Data Mining*, 2012, pp. 81–90.

[11] B. Bi, Y. Tian, Y. Sismanis, A. Balmin, and J. Cho, "Scalable topic-specific influence analysis on microblogs," in *Proc. 7th ACM Int'l Conf. on Web Search and Data Mining*, 2014, pp. 513–522.

[12] S. Chen, J. Fan, G. Li, J. Feng, K.-L. Tan, and J. Tang, "Online topic-aware influence maximization," *Proc. of the VLDB Endowment*, vol. 8, no. 6, pp. 666–677, 2015.

[13] H. Li, *Learning to Rank for Information Retrieval and Natural Language Processing*, 2nd ed. Morgan & Claypool Publishers, 2014.

[14] T. Qin, T.-Y. Liu, X.-D. Zhang, D.-S. Wang, and H. Li, "Global ranking using continuous conditional random fields," in *Advances in Neural Information Processing Systems 21 / Proc. 22 Annual Conf. on Neural Information Processing Systems*, 2008, pp. 1281–1288.

[15] C. Sutton and A. McCallum, *Introduction to Statistical Relational Learning*. MIT Press, 2006, ch. 4 An Introduction to Conditional Random Fields for Relational Learning.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[17] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 8th Int'l Conf. on Machine Learning*, 2001, pp. 282–289.

[18] T. Baltrušaitis, N. Banda, and P. Robinson, "Dimensional affect recognition using continuous conditional random fields," in *Proc. 10th IEEE Int'l Conf. on Automatic Face & Gesture Recognition*, 2013, pp. 1–8.

[19] S. Lim, S. Ryu, S. Kwon, K. Jung, and J.-G. Lee, "LinkSCAN*: Overlapping community detection using the link-space transformation," in *Proc. 30th IEEE Int'l Conf. on Data Engineering*, 2014, pp. 292–303.

[20] H. Yli-Renko, E. Autio, and H. J. Sapienza, "Social capital, knowledge acquisition, and knowledge exploitation in young technology-based firms," *Strategic Management Journal*, vol. 22, no. 6-7, pp. 587–613, 2001.

[21] J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg, "Signed networks in social media," in *Proc. 28th Int'l Conf. on Human Factors in Computing Systems*, 2010, pp. 1361–1370.

[22] H. Karau, A. Konwinski, P. Wendell, and M. Zaharia, *Learning Spark*. O'Reilly Media, 2015.

[23] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[24] T. Hoffman, "Probabilistic latent semantic analysis," in *Proc. 15th Conf. on Uncertainty in Artificial Intelligence*, 1999, pp. 289–296.

[25] E. Erosheva, S. Fienberg, and J. Lafferty, "Mixed membership models of scientific publications," *Proc. National Academy of Sciences*, vol. 101, no. suppl 1, pp. 5220–5227, 2004.

[26] R. Ottoni, J. P. Pesce, D. B. Las Casas, G. Franciscani Jr, W. Meira Jr, P. Kumaraguru, and V. Almeida, "Ladies first: Analyzing gender roles and behaviors in Pinterest," in *Proc. 7th Int'l AAAI Conf. on Weblogs and Social Media*, 2013.

[27] X. Geng, H. Zhang, Z. Song, Y. Yang, H. Luan, and T.-S. Chua, "One of a kind: User profiling by social curation," in *Proc. 22nd ACM Int'l Conf. on Multimedia*, 2014, pp. 567–576.