Influence Maximization for Effective Advertisement in Social Networks: Problem, Solution, and Evaluation

Suk-Jin Hong, Yun-Yong Ko
Hanyang University
South Korea
{wjrrjrwlr, yyko}@hanyang.ac.kr

Moonjeung Joe
Geyongnam National University of Science and Technology
South Korea
joemoon@gntech.ac.kr

Sang-Wook Kim*
Hanyang University
South Korea
wook@hanyang.ac.kr

ABSTRACT

As the number of people using social network services (SNS) increases significantly, companies start to use SNS as a marketing tool. For the reason, an *advertisement agent recommendation* has been introduced, which selects and recommends advertisement agents who effectively advertize goods of a company in SNS. To address the problem of advertisement agent selection, we propose a *multi-state diffusion model*. By applying our multi-state diffusion model to existing methods for influence maximization, we could solve the advertisement agent selection problem effectively. In evaluation, we show that the advertisement agents selected by the proposed approach have higher influence spread than the advertisement agents selected by existing methods. In addition, by conducting user study, we confirm that the proposed approach is effective and thus could be used in real-world applications.

CCS CONCEPTS

• **Information systems** → *Social advertising; Social networks;*

KEYWORDS

Influence maximization, Diffusion model, Social advertising

ACM Reference format:

Suk-Jin Hong, Yun-Yong Ko, Moonjeung Joe, and Sang-Wook Kim. Influence Maximization for Effective Advertisement in Social Networks: Problem, Solution, and Evaluation. 2019. In *Proceedings of ACM SAC Conference, Limassol, Cyprus, April 8-12, 2019 (SAC'19)*, 8 pages. DOI: https://doi.org/10.1145/3297280.3297412

1 INTRODUCTION

A social network service (SNS) is an online platform which enables people to share their opinions and knowledge on the Internet. In SNS, one can create/publish her own content in her own account or share it with someone else. She can also exchange her opinions through comments or 'Like' for someone else's content. Recently, as the number of SNS users has increased significantly, companies start to use SNS as an important marketing tool. They publish advertising content (i.e., advertisement contents) related to their goods or services (i.e., advertisement items) in SNS and conduct viral marketing.

Viral marketing for advertisement contents uses the *word-of-mouth effect* that is spreading the contents gradually through the user-to-user relationships in SNS. For effective viral marketing, advertising agents should be selected carefully in such a way that the advertisement contents are spread effectively in SNS. For this reason, a service has been introduced that selects the agents able to spread effectively the advertisement contents in SNS and recommends them to companies. This service is called *advertisement agent recommendation service*.

In this paper, we aim to solve the *advertisement agent selection problem*, which is to select the users who can spread the advertisement contents as much as possible. We can model the problem by the *influence maximization problem* [3,4,7,9] and then exploit the solutions of the influence maximization problem for our problem. Here, the influence of the advertisement contents spread over SNS can be calculated through the user-to-user relationships due to the word-of-mouth effect. Also, by using the marginal gain function, we can select the advertisement agents by considering the overlapping influence spread among the advertisement agents.

To solve the influence maximization problem, a *diffusion model* is used to calculate the influence spread of users in SNS [3,4,7,9,13]. However, the existing diffusion model has two limitations in solving the advertisement agent selection problem.

1) In the existing diffusion model, a user is in a state of 'active' or 'inactive'. Therefore, it cannot explain all user states that occur in the spread process of advertisement contents over SNS. In the spread process, there could be some users who are influenced by advertisement contents but do not spread the contents. For these users, we need a different user state other than 'active' or 'inactive' in the existing diffusion model. To this end, we need a new diffusion model that can describe all user states in the spread process of advertisement contents.

2) The existing diffusion model does not provide how to calculate the *diffusion probability* for node pairs required in the spread process of advertisement contents. In previous studies, the probability is assumed to be uniformly/randomly, or simply set by considering nodes' in-degree [7,9]. These naïve methods could not reflect the spread process of advertisement contents effectively. To address this issue, we need a method to calculate the probability that reflects the spread process of advertisement contents.

*Corresponding author

SAC'19, April 8-12, 2019, Limassol, Cyprus
© 2019 Copyright held by the owner/author(s). 978-1-4503-5933-7/19/04. . . $15.00
DOI: https://doi.org/10.1145/3297280.3297412

We propose a *multi-state diffusion model* to solve the advertisement agent selection problem. In the multi-state diffusion model, a user has not only 'active' and 'inactive' states as defined in the existing diffusion model, but also an 'assimilative' state. The *assimilative state* means a state whose user has been influenced by advertisement contents but does not spread the contents to other users. In addition, we propose a method to calculate the probability for user pairs that determines a user's state by 1) the degree of a user's attention for the advertisement item (*attention score*), 2) the degree of intimacy between users (*intimacy score*), and 3) the degree of a user's tendency of sharing advertisement contents (*sharing score*). In the multi-state diffusion model, a user's influence spread on advertisement contents is calculated by applying the *path-based method* for influence maximization [3,4]. We also use the *greedy method with CELF* [10] to identify the advertisement agents set [7].

We verify the effectiveness of our proposed approach by using the collected real-world SNS data. We compare the proposed approach with 1) Single Degree Discount (SDD) [3], 2) Weighted Cascade (WC) model using a greedy algorithm [7,9], and 3) follower-based approach, simply based on the number of followers. According to the results of our extensive experiments, the influence spread of the advertisement agents selected by the proposed approach is 61%, 108%, and 61% higher than 1) SDD, 2) WC, and 3) follower-based approach. We also conduct user studies to verify that the proposed approach is very effective in a practical sense.

The paper is organized as follows. Section 2 reviews the existing studies related to our research. Section 3 proposes our approach for solving the advertisement agent selection problem. Section 4 compares our proposed approach with the existing approaches via extensive experiments. Section 5 summarizes and concludes this paper.

2 RLATED WORK

2.1 Influence Maximization

Influence maximization (IM) is a problem of finding k nodes whose highest influence spread in a social network [3,4,7,9]. In IM, a social network is defined as a graph with $G = (V, E)$ where V is a set of nodes representing users, E is a set of edges that represent the relationship between users.

To calculate a node's influence spread on a social network, it needs a model that describes how the influence spreads in the social network. *Linear threshold (LT) model* and *independent cascade (IC) model* are widely used to calculate the node's influence spread in IM. The above two models assume the following in common.

1. A node has only one of an active or inactive state.
2. Inactive nodes become active nodes as spread progresses.
3. A node that becomes active affects the activation of its inactive neighbor nodes.
4. The spread process is terminated when no more nodes are available for activation.

The LT model assumes that each node has a random threshold between [0,1]. Each node is activated when the sum of the weights received from the activated neighboring nodes is greater than its threshold. The IC model is attempted to activate the node by a given probability from the newly activated neighbor node. If there are multiple newly activated neighbor nodes, the activation attempts for a node are performed independently and sequentially. The weights and probabilities of the two models are parameters.

To find k seed nodes with the optimal solutions, the influence spread of ${}_nC_k$ seed sets must be calculated and compared. Thus, finding the global optimum of the IM is known as NP-hard [7]. Therefore, to approximate the global optimum, a SimpleGreedy algorithm has been proposed [7]. The algorithm selects one node which maximizes the influence spread in stage-by-stage. SimpleGreedy calculates *marginal gains* of each node at each stage and selects the node with the highest marginal gain as the new seed node. The marginal gain is the additional influence spread when a node is added in the existing seed node set. The influence spread of node is calculated by running Monte-Carlo (MC) simulation.

The solution of the greedy algorithm guarantees more than 63% accuracy of the optimal solution if the objective function satisfies 1) non-negativity, 2) monotonicity, and 3) submodularity. Kempe et al. [7] proved that the objective function of the IM satisfies the above conditions. For this reason, the influence spread calculated through SimpleGreedy is often considered the ground truth of IM.

If the diffusion model used in IM is modified, the influence spread for the advertisement contents can be effectively calculated. Also, advertisement agents can be selected using the greedy algorithm. Therefore, we modify the existing diffusion model according to the spread process of advertisement contents.

2.2 Existing Solutions to IM

SimpleGreedy has a micro level issue and a macro level issue in terms of performance [3,4,9,10]. The micro level issue is that calculating influence spread of node using MC-simulation is expensive. The macro level issue is that re-evaluating the influence spread of every node is expensive after selecting one seed at each stage [10].

Path-based IM (PB-IM) is a method for solving the micro level issue of IM [3,4,8]. PB-IM calculates the influence spread of a node by summing of the path weights from the node to all reachable nodes without MC-simulation. The PB-IM also reduces the amount of computation by excluding paths having weights under the predefined threshold. The PB-IM improves the performance up to three times compared with SimpleGreedy with maintaining over 99% influence spread.

The Cost-Effective Lazy Forward (CELF) algorithm has been proposed to solve the macro level issue [10]. The CELF algorithm is a method of improving the performance using the characteristics of submodularity. Submodularity means that the marginal gain of each node decreases as the size of seed node increases. If the marginal gain of the node v in stage t is smaller than the marginal gain of the node u at the stage $t+1$, the marginal gain of the node v at stage $t+1$ is always smaller than that of the node u at stage $t+1$. Therefore, node v cannot be selected as a new seed node in stage *t+1*. For this reason, it is unnecessary to calculate the marginal gain of node v in stage *t+1*. The CELF algorithm is effective in eliminating unnecessary computations. The CELF algorithm improves the performance up to 700 times with the same influence spread compared with SimlpleGreedy.

By applying the methods for solving the micro and the macro issues in SimpleGreedy, we can improve the performance in solving the advertisement agent selection problem.

2.3 Similarity Measure

The neighbor-based similarity for two objects is calculated by comparing their neighbors [5,6,11]. SimRank is a method to measure the similarity between two objects [6] by taking the average of similarities between all possible pairs of their neighbors.

However, the SimRank has the following limitations.

1) If two objects have the same neighbors, the similarity decreases as the number of the neighbors increases. 2) If the most similar neighbor pair is excluded, the similarity between the objects does not decrease but rather increases [6,11]. MatchSim [11] solves the limitations by using the pairs selected by maximum matching instead of using the all possible pairs of neighbors.

Word embedding is a method to map words into n-dimensional vectors where n is a parameter. It can be used to calculate the similarity between words using the cosine similarity of word vectors. The word embedding is hypothesized that the words having similar meanings occur together in contents [2]. For this reason, the word vectors with similar meanings are closely mapped in the embedding space. Using this property, the similarity between words is calculated in the embedding space. Word2Vec is a neural network model for learning word vectors [12]. Word2Vec trains word vectors which are located closely to each other in the same sentence.

Using the neighbor-based similarity measure and the word embedding, the similarity of two objects can be expressed by similarities between words representing two objects. In this paper, we calculate the similarity between a user and an advertisement item by combining the neighbor-based similarity measure and the word embedding.

3 Proposed Approach

In this section, we mathematically formulate the advertisement agent selection problem and propose a multi-state diffusion model to address the problem. We also describe the spread process of the advertisement contents in the model.

3.1 Problem Definition

The advertisement agent selection problem is to find the advertisement agent set $S_{i_n}^*$ in the SNS graph $G = (V, E)$ where i_n is an element of the advertisement item set $I = \{i_1, i_2, i_3, \cdots, i_n, \cdots\}$, V is a set of nodes representing users, and E is a set of edges that represent the relationships between users. The advertisement agent selection problem is thus expressed by equation (1) below.

$$S_{i_n}^* = argmax_{S\subset V,|S|=k}\ \sigma(S, i_n) \quad (1)$$

In equation (1), S denotes a subset of V, $\sigma(S, i_n)$ is the influence spread of the advertisement agent set S for the advertisement item i_n, and $S_{i_n}^*$ means a set of k seed users with the maximizing influence spread on the advertisement contents i_n. Actually, $S_{i_n}^*$ is the solution to this problem.

3.2 Multi-State Diffusion Model

We propose a multi-state diffusion model for the advertisement agent selection problem. Though the IC model and the LT model were proposed for IM, they have the following two problems to apply the advertisement agent selection problem.

1) Since the user's state is either ‘active’ or ‘inactive’ in the existing diffusion model, the two states cannot represent a user who is influenced by the advertisement contents but does not spread the advertisement contents to other users. If the existing diffusion model is applied to the spread process of the advertisement content, the user influenced by the advertisement content always spreads the advertisement contents to their neighbors. Each user in SNS has her own tendency and decides to spread the advertisement contents to their neighbors according to her tendency. Therefore, we need to define a new state for such a user who is influenced by but does not spread the advertisement contents.

2) In the existing diffusion model, the method to calculate probabilities for user pairs in the spread process was not discussed. In the model, the probability is given as an equal value, random values, or $1/\mathrm{d}_v$ where d_v is the in-degree of user v [7,9]. From the point of view in the advertisement agent selection problem, this method does not reflect users’ attention for an item, relationships between users, and users’ tendencies in SNS. Therefore, we propose a method to calculate the probability considering the real spread process of advertisement contents in SNS.

For the advertisement agent selection problem, we define the user states as active, inactive, and assimilative. The *assimilative* user indicates the user who is influenced by the advertisement contents but does not spread the advertisement contents to her neighbors. We also introduce the following measures to calculate the diffusion probability for user pairs in the spread process of advertisement contents; 1) *attention score*: the degree of a user’s attention for the advertisement item, 2) *intimacy score*: the degree of intimacy between users, and 3) *share score*: the degree of a user’s tendency of sharing contents with others.

3.2.1 User States

The advertisement contents in SNS can be spread along the relationships between users. Some users who read the contents are influenced by the advertisement. Some users who are influenced share the advertisement contents in their own account. The shared advertisement contents can be spread again over their neighbors. As this process is repeated, the advertisement contents are spread over SNS. In this process, there are three kinds of users according to their characteristics. We define them as *active*, *assimilative*, and *inactive*.

1) An *active* user is influenced from the advertisement contents and then shares the advertisement content. We define an active user to have influences and spread effects.

2) An *assimilative* user is influenced from the advertisement content but does not share the advertisement contents. We define an assimilative user to have influences and doesn't have spread effects.
3) An *Inactive* user is not influenced from the advertisement contents and thus does not share them with others. We define the inactive user not to have influences and spread effects.

Table 1 shows the user state of the multi-state diffusion model compared with that of the existing IC and LT models.

Table 1: User states in multi-state diffusion model, IC model, and LT model

Characteristics	Multi-state diffusion model			IC/LT models	
	Active	Assimilative	Inactive	Active	Inactive
Influences	O	O	X	O	X
Spread Effects	O	X	X	O	X

In Table 1, the active user in the IC and LT models has influences and spread effects while the inactive user does not. Therefore, the user who has influences and doesn't have spread effects cannot be represented in IC and LT models. On the other hand, we define an *assimilative* user as a user who has influences and does not have spread effects. In the multi-state diffusion model, it is possible to distinguish assimilative users from active and inactive users.

3.2.2 Attention Score: $AS(v, i_n)$

The attention score $AS(v, i_n)$ is a probabilistic factor representing the degree of a user v's attention for advertisement item i_n. If user v mentions i_n a lot in her contents, the user v is likely to have attention (i.e., interest) in i_n. The attention score $AS(v, i_n)$ is calculated by the neighbor-based similarity between $i_n{}'$ s keywords and the words frequently used by user *v* [6,11].

The user $v's$ frequently used words are denoted by $W_v^* = (W_v, F_{W_v})$ where $W_v = \{a_1, a_2, \cdots, a_m\}$ is a word set and $F_{W_v} = \{f_{a_1}, f_{a_2}, \cdots, f_{a_m}\}$ is a frequency set for W_v. The keywords for i_n are denoted by $W_{i_n} = \{b_1, b_2, \cdots, b_{m'}\}$. $|W_v|$ is a parameter. We assume that the keywords W_{i_n} and $|W_{i_n}|$ are given by advertisers.

To calculate $AS(v, i_n)$, we represent a weighted bipartite graph $G_{v,i_n} = (W_v^*, W_{i_n}, E)$, where E indicates a weighted edge set $E = \{ f_a \cdot \mathrm{sim}(a,b) \mid a \in W_v,\ b \in W_{i_n},\ f_a \in F_{W_v}\}$. In E, $\mathrm{sim}(a,b)$ is a cosine similarity between a's word vector and b's word vector. In graph G_{v,i_n}, we give a weight to an edge between words *a* and *b* by $f_a \cdot \mathrm{sim}(a,b)$ other than $\mathrm{sim}(a,b)$. This is for reflecting f_a, which is a frequency of a, in similarity computation. Then, we find maximum matching on graph G_{v,i_n}. The reason for this is to resolve the limitations of using all possible pairs occurring in SimRank [6,11]. Finally, we calculate $AS(v, i_n)$ by summing the word pairs' edge weights in maximum matching. The equation of $AS(v, i_n)$ can be formulated as follows.

$$AS(v, i_n) = \frac{\sum_{(a,b)\in m_{W_v^*, W_{i_n}}} f_a \cdot sim(a,b)}{c},\ f_a \in F_{W_v} \tag{2}$$

In equation (2), $m_{W_v^*, W_{i_n}}$ is a maximum matching set on G_{v,i_n}, and c is a normalization factor to make $AS(v, i_n)$ a probabilistic value between [0,1]. As $AS(v, i_n)$ gets higher, user *v* is more likely to be influenced from advertisement content i_n.

3.2.3 Intimacy Score: $IS(v, u)$

The intimacy score $IS(v, u)$ is a probabilistic factor representing the degree of intimacy from user v to user u. If user v leaves a lot of actions like comments and 'Like' in user u's contents, user v could be thought to have intimate relationship with user u. Thus, we compute the intimacy score $IS(v, u)$ by normalizing the number of actions that user *v* gave to user *u*.

The number of actions given by user *v* to user *u* is denoted by $action_{v,u}$ and the maximum number of actions for two users in SNS is $\max_{a,b\in V}(action_{a,b})$. To convert $IS(v, u)$ to a probability, we normalize $action_{v,u}$ to [0,1] using $\max_{a,b\in V}(action_{a,b})$. To prevent the probability of a user pair having no action from becoming zero, we add *1* to both of $action_{v,u}$ and $\max_{a,b\in V}(action_{a,b})$. The equation of $IS(v, u)$ becomes as follows.

$$IS(v, u) = \frac{action_{v,u} + 1}{\max_{a,b\in V}(action_{a,b}) + 1} \tag{3}$$

We assume that as $IS(v, u)$ gets higher, user *v* will be more likely to be influenced from user *u*'s advertisement content.

3.2.4 Share Score: $SS(v)$

The share score $SS(v)$ is a probabilistic factor representing the degree of user v's tendency of contents sharing. If user v shared other users' contents many times in the past, she is expected to share other advertisement contents in the future. Therefore, share score $SS(v)$ is calculated by normalizing the number of user *v*'s sharing contents in her account.

The number of user *v*'s sharing contents is denoted by $share_v$ and the maximum number of user's sharing contents in SNS is $\max_{a\in V}(share_a)$. To prevent the probability of a user having no sharing contents from becoming zero, we add *1* to both of $share_v$ and $\max_{a\in V}(share_a)$. The equation of $SS(v)$ now becomes as follows.

$$SS(v) = \frac{share_v + 1}{\max_{a\in V}(share_a) + 1} \tag{4}$$

Thus, as $SS(v)$ gets higher, user *v* will be more likely to share any advertisement contents in the future.

3.3 Spread Process of Advertisement Contents

The spread process of advertisement contents i_n in the multi-state diffusion model is shown in Fig. 1. *u* and *v* are users, and the directed edge between users means a directed spread of the contents. The spread direction is the opposite of the follower relationship. If user *u* is activated in step *t-1*, she influences her followers $v_1, v_2,$ and v_3 in step *t*. If users $v_1, v_2,$ and v_3 have high values for $AS(v., i_n)$ and $IS(v., u)$, they are likely to be influenced. In Fig. 1, users v_1 and v_3 are influenced and become assimilative, but v_2 is not influenced and thus become inactive.

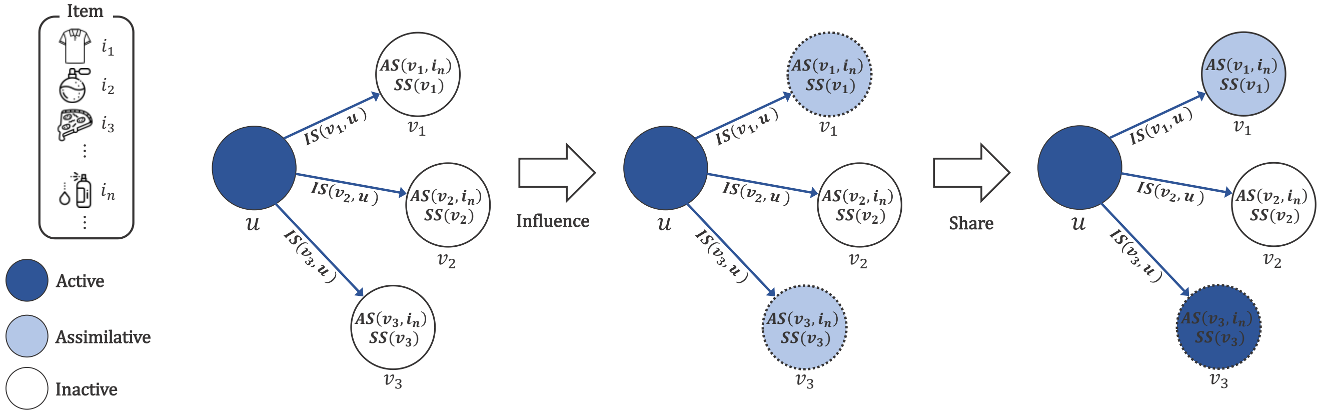

Figure 1: Process of multi-state diffusion model.

Then, assimilative users v_1 and v_3 share the advertisement contents, depending their tendency. If users v_1 and v_3 have high values for $SS(v.)$, they are likely to be activated. In Fig. 1, user v_1 does not share the advertisement content and thus remains assimilative, and user v_3 shares the advertisement content and thus becomes active. Active user v_3 has spread effect now on their followers in the next step *t + 1*.

The probability that user *v* will be in each state, due to user *u* who activated in step *t*, at step *t + 1*, is defined as follows.

$$P(v = active) = AS(v, i_n) \cdot IS(v, u) \cdot SS(v) \tag{5}$$
$$P(v = assimilative) = AS(v, i_n) \cdot IS(v, u) \cdot (1 - SS(v)) \tag{6}$$
$$P(v = inactive) = 1 - AS(v, i_n) \cdot IS(v, u) \tag{7}$$

Algorithm 1 shows the spread process of the multi-state diffusion model. $AS(\cdot, i_n)$, $IS(\cdot,\cdot)$, $SS(\cdot)$, advertisement item i_n, and initial set of users A_0 are given. In the existing diffusion model, users in A_0 start with active states. However, in the multi-state diffusion model (line 2-7), users in A_0 start to be activated by using $AS(\cdot, i_n)$. This is because advertisers (i.e., companies) evaluate the advertisement agents by their attention for advertisement item i_n. The user *u* activated in the previous stage tries to influence their followers *v* (line 12). The state of user *v* is determined randomly according to the calculated probabilities and random number (line 13-23). For the users who are activated in this step, they have spread effect on their followers in the next step (lines 11 and 18). Finally, the multi-state diffusion model returns the influence spread of A_0, which is the sum of the sizes of an active user set and an assimilative user set (line 28).

We solve the advertisement agent selection problem by applying the solutions for IM to the multi-state diffusion model. For calculating the influence spread of a user efficiently, the path-based IM method is employed [3,4]. Based on the user's influence spread calculated, the advertisement agent is selected one by one by considering the marginal gain [7]. In this process, we use the CELF algorithm to reduce computational redundancy without any loss of influence spread [10].

Algorithm 1 Multi-State Diffusion Model

```
Input: network $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, initial set of users $\boldsymbol{A_0}$, advertisement item
       $\boldsymbol{i_n}$, attention score $\boldsymbol{AS}(\cdot, \boldsymbol{i_n})$, intimacy score $\boldsymbol{IS}(\cdot,\cdot)$, share score
       $\boldsymbol{SS}(\cdot)$
Output: number of users affected by $\boldsymbol{i_n}$ advertisement contents
1:  $i = 0, B = \emptyset, C = \emptyset$ // i: step
2:  for all $u \in A_0$ do
3:    rand = generate a random number in [0,1];
4:    if rand < $ITR_{u,i_n}$ then
5:      $A_0 = A_0 \cup \{u\}$
6:    end if
7:  end for
8:  while $A_i \neq \emptyset$ do
9:    $i = i+1$;
10:   $A_i = \emptyset$;
11:   for all $u \in A_{i-1}$ do
12:     for all $v \in O(u)$ s.t. $v \notin \cup_{j=0}^{i} A_j$ and $v \notin B$ and $v \notin C$ do
13:       $P(v = active) = AS(v, i_n) \cdot IS(v, u) \cdot SS(v)$
14:       $P(v = assimilative) = AS(v, i_n) \cdot IS(v, u) \cdot (1 - SS(v))$
15:       $P(v = inactive) = 1 - AS(v, i_n) \cdot IS(v, u)$
16:       rand = generate a random number in [0,1];
17:       if $p(v = active) >= rand$ then
18:         $A_i = A_i \cup \{v\}$;            // v is active user
19:       else if $p(v = active) < rand$ and $p(v = active) +$
          $P(v = assimilative) >=$ rand then
20:         $B = B \cup \{v\}$;            // v is assimilative user
21:       else
22:         $C = C \cup \{v\}$;            // v is inactive user
23:       end if
24:     end for
25:   end for
26: end while
27: $A = \cup_{j=0}^{i} A_j$;
28: Return $|A| + |B|$;
```

4 Evaluation

In this section, we evaluate the effectiveness of the proposed approach through extensive experiments. We design our experiments to answer the following key questions.

1. Parameter tuning
   a. What is the best set of Word2Vec parameters for calculating the similarity between words used in attention score $AS(v, i_n)$?
   b. What is the best pruning threshold for efficient path-based influence spread evaluation?
2. Proposed approach
   a. How much effective are attention, intimacy, and share scores in selecting an advertisement agent in terms of influence spread?
   b. How much accurate are the advertisement agents selected by the proposed approach than those selected by the existing approach in terms of influence spread?
3. User study
   a. How do real SNS users rate the advertisement agents selected by our proposed and existing approaches?

4.1 Experimental Setup

4.1.1 Dataset

We randomly sampled 30 users subscribed in ReviewShare[1], an advertisement agent recommendation service. Then, we collected the contents and actions of those users and their followers in the Naver Blog[2]. Naver Blog is the largest blog service in Korea provided by Naver Corp. Because most blog contents are written in Korean called *Hangul*, some optimization for Hangul was needed. Table 2 shows the statistics of the Naver Blog dataset.

Table 2: Statistics of Naver blog dataset

# of users (nodes)	# of followings (edges)	Maximum in-degree	Maximum out-degree
2,336,953	11,648,733	1,272	152,417

4.1.2 Experiment set

We performed two experiments *A* and *B* according to the keywords in an advertisement item and words used frequently by a user. In Experiment *A,* advertisement item i_n is '유모차(stroller)' and keywords for i_n $W_{i_n} = \{b_1, b_2, b_3\}$ are '육아(infant care)', '유모차(stroller)', and '아기(baby)'. In Experiment *A*, the number of words used frequently by user v, *m,* was fixed at 10 in $W_v = \{a_1, a_2, \cdots, a_m\}$. In Experiment *B*, advertisement item i_n is '라면(ramen)' and keywords for i_n $W_{i_n} = \{b_1, b_2\}$ are '국물(soup)' and '라면(ramen)'. In Experiment *B*, the number of words used frequently by user v, m, was fixed at 5. We selected advertisements agent among 30 sampled users. This is because the users who subscribed could be candidates of advertisement agents in ReviewShare.

4.2 Parameter Tuning

4.2.1 Best Word2Vec parameters

In this experiment, we try to find the best Word2Vec training parameters for mapping a Hangul word to its word vector [12]. Because the contents in Naver Blog are mostly written in Hangul, it is necessary to find the Word2Vec training parameters best for Hangul. Using the trained word vectors, the attention scores are calculated.

Towards this end, we used Wikipedia[3] data as the training data set and used WS353-r and WS353-s as test data sets, which are typically used for semantic similarity and relatedness in the word embedding [1]. We translated the two data sets into Korean for this purpose. We evaluated the accuracy by calculating the Pearson correlation between the similarity scores specified and the cosine similarities of their corresponding word vectors

Table 3 shows the results of parameter tuning. For test data sets WS353-r and WS353-s, the best Word2Vec parameters with the largest Pearson correlation are window of 7, minimum count of 5, and dimensionality of 200. In another experiment, we observe that our result is better than that from the word vectors trained by Fasttext[4]. The Pearson correlations of word vectors by Fasttext were 0.582 (WS353-r) and 0.618 (WS353-s).

Table 3: Evaluation of word vectors with different parameters

Parameter			Evaluation	
Window	Minimum count	Dimension	WS353-r	WS353-s
2	3	50	0.570	0.655
3	4	100	0.587	0.660
7	5	200	**0.611**	**0.661**
5	7	300	0.593	0.641

4.2.2 Best pruning threshold

In PB-IM in the proposed approach, it is known as P-hard to calculate the influence spread for all possible paths [14]. To reduce the calculation, PB-IM eliminates the paths with lower weights than the pre-defined pruning threshold α. In this experiment, we try to find the best pruning threshold α that reduces the running time while maintaining accurate influence spread. We measured the influence spread and the running time with α = $10^{-1}, 10^{-2}, \cdots,$ and 10^{-7}. The number of users was set as 5.

Fig. 2 shows the influence spread and the running time according to the pruning threshold α. The left is the result for Experiment *A* while the right is that for Experiment *B* in Fig. 2. The *x*-axis represents the running time (sec) and the *y*-axis represents the influence spread of the users. When the pruning threshold α is decreased from 10^{-6} to 10^{-7}, the influence spread increases by 5% while the running time increases by 458% in Experiment *A* and also influence spread increases by 4% while the running time increases by 543% in Experiment *B*. Therefore, we used 10^{-6} as the pruning threshold in the proposed approach in the following experiments.

[1] https://reviewshare.io
[2] https://section.blg.naver.com
[3] https://ko.wikipedia.org
[4] https://fasttext.cc/docs/en/crawl-vectors.html

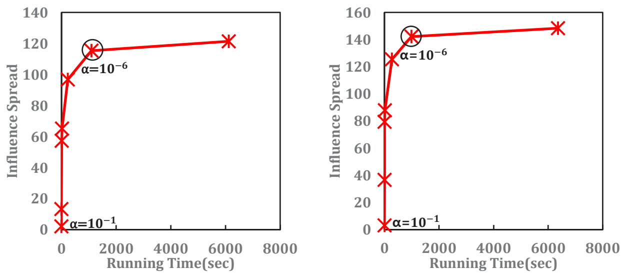

Figure 2: Influence spread and running time with pruning threshold α.

Table 4: Pruning threshold α for each comparison method

Experiment	IS	AS	AI
Threshold α	1/10,000	1/1,280	1/12,500

4.3 Proposed Approach

4.3.1 Effectiveness of attention, intimacy, share scores in term of influence spread

In this experiment, we aimed to verify that the attention score, the intimacy score, and the share score affect the influence spread. We compared the methods that exclude each element in the proposed approach as follows: the proposed approach with all elements (AIS), the proposed approach without the attention score (IS), the proposed approach without the intimacy score (AS), and the proposed approach without the share score (AI).

Since the four methods have different probability distributions, it is necessary to find the pruning threshold for each comparison method. As a result, we found the best pruning threshold for IS, AS, and AI in the same way as Section 4.2.2. Table 4 shows the best pruning threshold α for each method.

We selected the advertisement agents using the pruning threshold α for each comparison method. Finally, we measured the influence spread of the agents obtained by each method via a multi-state diffusion model. Although the attention score was different in experiment sets *A* and *B,* we obtained the same best pruning threshold in them.

Fig. 3 shows the influence spread of each method according to the number of selected users. The left is the result for Experiment *A* while the right is the result for Experiment *B*. The *x*-axis represents the number of selected agents and the *y*-axis represents the influence spread by them. When the number of selected advertisement agents is 5 in Experiment *A*, the influence spread of IS, AS, and AI is 8%, 17%, and 4% lower than AIS, respectively. In Experiment *B*, the influence spread of IS, AS, and AI is 39%, 39%, and 2% lower than AIS, respectively. This result shows that the attention, intimacy, and share score are all meaningful factors in selecting advertisement agents.

4.3.2 Accuracy of advertisement agents

In this experiment, we tried to compare the influence spread for the proposed approach and the existing methods. Since there is no ground truth for advertisement agent selection, we measured the influence spread for the advertisement agents selected by each method by using our multi-state diffusion model.

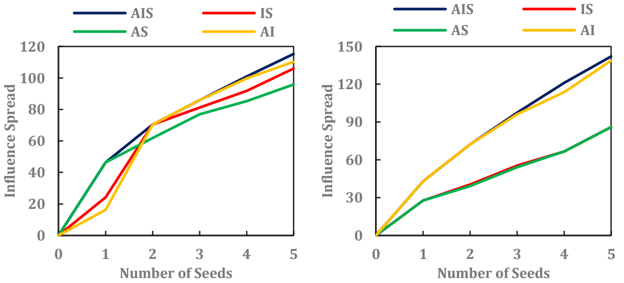

Figure 3: Influence spread of advertisement agents selected by each method.

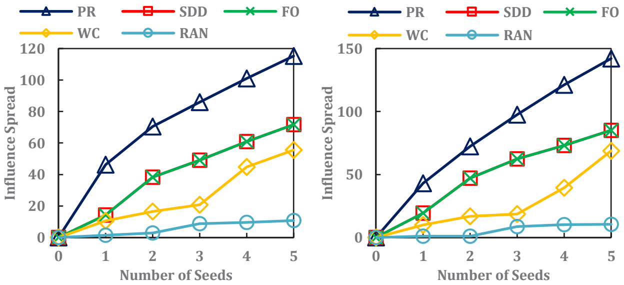

Figure 4: Comparison of influence spread of advertisement agent selected by each method.

The methods for comparisons are as follows: the proposed approach (PR), the weighted cascade model using a greedy algorithm (WC) [7,9], the singe degree discount (SDD) [3], the follower-based method (FO), and a random selection (RAN). WC, a variation of the IC model, assigns the probability by $1/d_v$ where d_v is the number of in-degree of user *v*. We also found the pruning threshold α for WC in the same way as in Section 4.2.2, and it is 1/2,048. SDD selects users having the highest degree. Once a seed user is selected, it decreases the degree of its neighbors by 1. FO, a method actually used in ReviewShare, selects seed users in the order of their degrees. RAN, a baseline method, it selects seed user randomly in each stage. We measured the influence spread by using the multi-state diffusion model while increasing the number of agents selected by each method.

Fig. 4 shows the influence spread of advertisement agents selected by each method. When the number of selected advertisement agents is 5 in Experiment *A*, the influence spread of PR is 61%, 61%, 108%, and 970% higher than SDD, FO, WC, and RAN, respectively[5]. In Experiment *B*, the influence spread of PR is 66%, 66%, 206%, and 1,253% higher than SDD, FO, WC, and RAN, respectively.

4.4 User Study

In this experiment, we conducted user study by human beings to evaluate the quality of advertisement agents selected by each method. Because there is no ground truth in our advertisement agent recommendation situation, we aimed to evaluate each set of advertisement agents selected by each method through the user

[5] For each item, once all advertisement agents are sorted in the order by our IM algorithm, the time for retrieving a set of *k* advertisement agents from the list appears to be around 10 milliseconds.

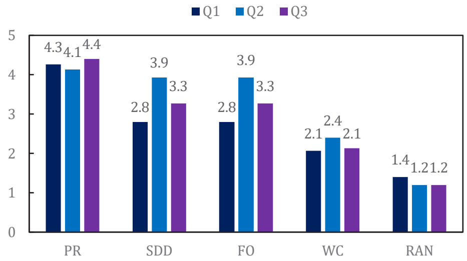

Figure 5: Quality of advertisement agents selected by each method.

study. The methods for comparisons are the same as in Section 4.3.2, and Experiment *A* is the subject of our user study here.

In the user study, we asked 15 evaluators who had experience on using SNS to evaluate five advertisement agents chosen by each method. Then, we provided the evaluators with information on those agents. The information for the advertisement agents is the number of followers, the crawled contents, and the comments and 'Like' from her followers left; the information about the item is the name and its description. We asked the evaluators to give 1~5 point to each set of agents for our questions according to the guidelines: guidelines and questions are as follows.

Guideline: Evaluate each set of advertisement agents from the advertiser's perspective.

Q1: Do you think a given set of advertisement agents is selected effectively for the target item (in terms of attention)?

Q2: Do you think a given set of advertisement agents is selected effectively for the target item (in terms of the influence spread)?

Q3: Do you think a given set of advertisement agents is selected effectively for the target item (in terms of overall advertisement)?

Fig. 5 shows the evaluation results for the questions. The *x*-axis represents each method and the *y*-axis represents the point averaged over 15 evaluators. The set of advertisement agents selected by the proposed approach showed the scores in Q1, Q2 and Q3 significantly higher than those of SDD, FO, WC and RAN. This result indicates that the advertisement agents selected by our multi-state diffusion model are nice and reasonable from a human point of view, which subsequently shows the practicality of our approach.

5 Conclusions

In this paper, we proposed a multi-state diffusion model for the advertisement agent selection problem. In advertisement agent selection, the existing LT and IC models have a limitation to represent user states in the spread process of advertisement contents. Also, the existing diffusion models do not provide how to calculate the diffusion probability for user pairs in the spread process of advertisement contents. We addressed all these issues based on the multi-state diffusion model and the efficiency issue by employing the state-of-the-art IM methods.

We evaluated the proposed approach by using real-world SNS data via extensive experiments. The results showed that the advertisement agents selected by the proposed approach have influence spread much higher than those of the existing methods. Furthermore, by conducting user study, we confirmed that the proposed approach is quite effective from the human beings' perspective, and thus could be employed in real-world applications.

ACKNOWLEDGMENTS

This work (Grants No. C0511678) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2017 and also supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) (NRF-2017M3C4A7069440) and the National Research Foundation of Korea (NRF) grant (No. NRF2017R1A2B3004581). Sang-Wook Kim is the corresponding author.

REFERENCES

[1] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa. 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-Based Approaches. In *Proceeding of the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (NAACL'09). Association for Computational Linguistics, Stroudsburg, PA, 19-27.

[2] Y. Bengio, R. Dicharme, P. Vincent, and C. jauvin. 2003. A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, Vol. 3, 1137-1155.

[3] W. Chen, Y. Wang, and S. Yang. 2009. Efficient Influence Maximization in Social Networks. In *Proceeding of the 15th International Conference on Knowledge Discovery and Data Mining* (SIGKDD'09). ACM, New York, NY, 199–208.

[4] W. Chen, Y. Yuan, and L. Zhang. 2010. Scalable Influence Maximization in Social Networks under the Linear Threshold Model, In *Proceeding of the 26th International Conference on Data Mining* (ICDM'10). IEEE, 88–97.

[5] M. R. Hamedani, S.-W. Kim, 2017. JacSim: An accurate and efficient link-based similarity measure in graphs. *Information Sciences*, Vol. 414, Elsevier, 203-224.

[6] G. Jeh, and J. Widom. 2002. SimRank: A Measure of Structural-context Similarity, In *Proceeding of the 13th International Conference on Knowledge Discovery and Data Mining* (SIGKDD'07). ACM, New York, NY, 538-543.

[7] D. Kempe, J. Kleinberg, and É. Tardos. 2003. Maximizing the Spread of Influence through a Social Network. In *Proceeding of the 9th International Conference on Knowledge Discovery and Data Mining* (SIGKDD'03). ACM, New York, NY, 137-146.

[8] Y.-Y. Ko, D.-K. Chae, and S.-W. Kim. 2016. Accurate Path-based Methods for Influence Maximization in Social Networks. In *Proceeding of the 25th International Conference on World Wide Web* (WWW'16). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 59-60.

[9] Y.-Y. Ko, K.-J. Cho, and S.-W. Kim. 2018. Efficient and Effective Influence Maximization in Social Networks: A Hybrid-Approach. *Information Sciences*, Vol. 465, Elsevier, 144-161.

[10] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, and J. VanBriesen, and N. Glance. 2007. Cost-Effective Outbreak Detection in Networks, In *Proceeding of the 13th International Conference on Knowledge Discovery and Data Mining* (SIGKDD'07). ACM, New York, NY, 420–429.

[11] Z. Lin, M.R. Lyu, and I. King. 2009. MatchSim: A Novel Neighbor-Based Similarity Measure with Maximum Neighborhood Matching. In *Proceeding of the 18th International Conference on Information and Knowledge Management* (CIKM'09), ACM, New York, NY, 1613-1616.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781* (2013).

[13] S. Peng, A. Yang, L. Cao, S. Yu, and D. Xie. 2017. Social Influence Modeling using Information Theory in Mobile Social Networks. *Information Sciences*, Vol. 379, Elsevier, 146-159.

[14] L.G. Valiant. 1979. The Complexity of Enumeration and Reliability Problems. *SIAM Journal on Computing*, Vol. 8(3), Society for Industrial and Applied Mathematics, 410–421.