mathematics

MDPI

Article

Influence Maximization Based on Backward Reasoning in Online Social Networks

Lin Zhang and Kan Li *

School of Computer Science, Beijing Institute of Technology, Beijing 100081, China; bitcszhl@bit.edu.cn

* Correspondence: likan@bit.edu.cn

Abstract: Along with the rapid development of information technology, online social networks have become more and more popular, which has greatly changed the way of information diffusion. Influence maximization is one of the hot research issues in online social network analysis. It refers to mining the most influential top-*K* nodes from an online social network to maximize the final propagation of influence in the network. The existing studies have shown that the greedy algorithms can obtain a highly accurate result, but its calculation is time-consuming. Although heuristic algorithms can improve efficiency, it is at the expense of accuracy. To balance the contradiction between calculation accuracy and efficiency, we propose a new framework based on backward reasoning called Influence Maximization Based on Backward Reasoning. This new framework uses the maximum influence area in the network to reversely infer the most likely seed nodes, which is based on maximum likelihood estimation. The scheme we adopted demonstrates four strengths. First, it achieves a balance between the accuracy of the result and efficiency. Second, it defines the influence cardinality of the node based on the information diffusion process and the network topology structure, which guarantees the accuracy of the algorithm. Third, the calculation method based on message-passing greatly reduces the computational complexity. More importantly, we applied the proposed framework to different types of real online social network datasets and conducted a series of experiments with different specifications and settings to verify the advantages of the algorithm. The results of the experiments are very promising.

Keywords: influence maximization; backward reasoning; influence cardinality; online social networks

Citation: Zhang, L.; Li, K. Influence Maximization Based on Backward Reasoning in Online Social Networks. *Mathematics* **2021**, *9*, 3189. https://doi.org/10.3390/math9243189

Academic Editor: Manuel Alberto M. Ferreira

Received: 6 November 2021
Accepted: 9 December 2021
Published: 10 December 2021

1. Introduction

With the development of the Internet, online social networks have become more and more popular, and have penetrated all aspects of human lives. We can share what we have seen and heard through online social networks, and we can also obtain the latest news and marketing information from there. Online social networks have become a new carrier for the spread of information [1]. The information transfer in online social networks is faster and more convenient, it is easier to create a popular trend, which brings new opportunities and challenges to marketing, and also attracts great attention of researchers.

Word-of-mouth marketing in the new situation can be abstractly described as an influence maximization problem, which is one of the key issues in social network analysis [2]. It refers to mining the most influential top-*K* nodes from an online social network to maximize the final spread of influence in the network. Kempe et al. first formulated the problem of influence maximization as a discrete optimization problem, which is a milestone in the research of influence maximization [3]. They also proved that when the diffusion model is independent cascade and linear threshold models, the problem of influence maximization is an NP-hard.

As one of the important issues of online social network analysis, previous research on this issue is mainly divided into two categories: research based on greedy algorithms and heuristic algorithms. Greedy algorithms usually require Monte Carlo simulation of the propagation process, and an accurate Monte Carlo simulation often needs to be repeated many times. Although this method guarantees the accuracy of the results, it is very time-consuming. For the heuristic algorithms, they can use heuristic information to shorten the running time, but this will cause a loss of accuracy and stability. In response to this problem, researchers are constantly exploring new methods to balance the contradiction between calculation accuracy and efficiency.

In this paper, we propose a solution using backward reasoning, which is based on maximum likelihood estimation. The basic idea is to improve the efficiency of the algorithm by avoiding the Monte Carlo simulations while ensuring the accuracy of the result. We define the influence cardinality of nodes and establish rules by using backward reasoning to evaluate the probability of each node as a seed node to spread the information to the entire network. Based on these foundations, the set of seed nodes can be constructed according to given strategies. The influence cardinality considers the process of information diffusion, which provides a guarantee for the accuracy of the algorithm. In addition, to reduce the time complexity of calculating the influence cardinality, we propose an algorithm based on message-passing, which greatly reduces the computational complexity of our framework.

Summing up, the highlights of this paper are four-fold. First of all, to solve the problem to influence maximization, we propose a new framework, Influence Maximization based on Backward Reasoning, (IMBR), which achieves a balance between the accuracy of the result and the efficiency. Second, we define the influence cardinality of the node based on the information diffusion process and the network topology structure. Third, the calculation method based on message-passing reduces the number of repeated traversals of the entire network. Finally. A series of experiments with different specifications and settings were carried out on real online social network datasets to examine the advantages of the framework, which proves to be quite promising.

The rest of the paper is organized as follows. In the next section, a brief survey of the related work about influence maximization is given. The framework proposed in this paper is presented in Section 3. The experiments on real world networks are performed in Section 4. Finally, conclusions and future works are given in Section 5.

2. Related Work

The problem of influence maximization was first proposed by Domingos and Richardson [4]. Based on this research, Kempe et al. [3] first defined the problem as a discrete optimization problem. According to their research, the definition of the optimization problem for maximizing influence is to mine K seed nodes that can maximize the spread of influence on an online social network based on a given diffusion model. Besides, they also proved that the influence function satisfies the sub-module properties and the monotonicity under the most commonly used independent cascade and linear threshold models. Finally, they proposed a greedy climbing algorithm [5], which can guarantee the approximate optimal of $1 - \frac{1}{e} - \varepsilon$. Later, Sviridenko [6] extends the greedy framework using a non-uniform cost function for seed selection. In order to improve calculation efficiency, Leskovec et al. [7] proposed the Cost Effective Lazy Forward (CELF) schema, Goyal et al. [8] proposed CELF++, an extension of CELF. CELF and CELF++ use the sub-mode attribute of the influence function to avoid calculating the marginal gain of all non-seed nodes when selecting a seed node every time. Estevez et al. [9] presented a Set Covering Greedy (SCG) algorithm which was also the improvement of the greedy algorithm. When selecting the seed nodes, SCG discards the overlapping part with the neighbor nodes of the seed nodes. To improve the efficiency of the solution, Chen et al. [10] proposed a new algorithm called New Greedy-IC. This algorithm removes those edges that cannot successfully propagate the information in the iterative process. After these, Zhou et al. [11] used the influence function to find the upper limit of the marginal benefit of the node influence diffusion. Based on this, the Upper Bound based Lazy Forward (UBLF) algorithm was proposed. UBLF can reduce the number of repetitions of Monte Carlo simulation to improve computational efficiency by using boundaries. Unfortunately, at the beginning of selecting seed nodes, the effect of reducing the number of Monte Carlo simulations is significant. As more and more seed nodes are selected, the effect continues to decay. Lu et al. [12] proposed a cascade discount (CD) algorithm, which is based on the greedy algorithm, to solve the problem of influence maximization in the independent cascade model. Moreover, there are some other improved methods based on greedy algorithms, Ge et al. [13] extended the confidence interval estimator and proposed an end-to-end method for influence maximization, called the influence maximization based on learning automata (IMLA). Tian et al. [14] proposed a hybrid potential-influence greedy algorithm (HPG) based on the linear threshold model. The HPG algorithm uses the "influence accumulation" property of the diffusion model to heuristically select half of the seed nodes with the largest potential influence, and then greedily select the other half seed nodes. Wang et al. [15] proposed a potential-based greedy selection strategy to solve this problem, and so on [16].

Although the greedy algorithm can guarantee the best approximation, the solution process takes a huge amount of time. This is because in the process of selecting seed nodes, it is necessary to repeatedly calculate the marginal revenue of all non-seed nodes through Monte Carlo simulations. As we all know, Monte Carlo simulation is very time-consuming. Even if the number of repetitions is reduced, when the network scale becomes larger, the computational complexity is still considerable. Therefore, some researchers began to consider using heuristics to improve efficiency. Chen et al. [10] discussed the relationship between influence diffusion and the degree of nodes and proposed the Degree-Discount heuristic algorithm. This algorithm can significantly reduce the calculation time, but at the expense of some accuracy. Wang et al. [17] proposed a community-based greedy algorithm (CGA) to mine top-K influential seed nodes. CGA considers the community structure in the process of influence diffusion. Furthermore, Khomanmi et al. [18] considered the spread of influence in the local community and proposed a fast and scalable algorithm named the community finding influential node (CFIN). The CFIN algorithm includes two main parts: local community spreading and seed selection. After, Kundu et al. [19] defined the diffusion degree of the node, which is a centrality measure used to indicate the influence of the node on other nodes. They used this measure to mine the influential nodes. Kim et al. [20] proposed a scalable influence approximation algorithm based on the independent cascade model, which is called the Independent Path Algorithm (IPA). IPA treated the independent influence path as an influence evaluation unit to effectively calculate the influence of the node. There are some other heuristics based on influence path like shortest path (SPIM) [21], SIMPATH [22], and LDAG [23]. Besides, Singh et al. [24] designed a new method called Influence Maximization Algorithm Based on Ant Colony Optimization (ACO-IM) to solve the problem of influence maximization in social networks. ACO-IM defined the rules for individuals to exchange and update information in the problem of influence maximization. Jung et al. [25] combined influence ranking (IR) and influence estimation (IE) to propose a novel algorithm, IRIE, to solve the problem of influence maximization. The algorithm can be used for the independent cascade model and its extended models. Tang et al. [26] proposed an algorithm, called TIM. In theory, TIM can return a $(1 - \frac{1}{e} - \varepsilon)$ approximate solution. In practice, TIM adopts a novel heuristic method, which can significantly improve efficiency without affecting its asymptotic performance. To support massive graphs, some other different algorithms have also been proposed. Cohen et al. [27] proposed the greedy sketch-based impact maximization (SKIM) algorithm. The core of SKIM is to construct the summary structure of each node, that is, the combined reachability sketch. SKIM is a greedy algorithm in the sketch space. It has high scalability and can be easily extended to graphs with billions of edges. To improve the efficiency of calculation, Nguyen et al. [28] were inspired by the stop and stare strategy and proposed the Stop-and-Stare Algorithm

(SSA), which is based on sampling technology. And they also proved that SSA is the first approximation algorithm that uses the (asymptotically) minimum number of samples. In addition, other heuristic-based algorithms are MIA [29], CIM [30], CoIM [31], and others [32,33]. These heuristic methods use the heuristic information in the influence diffusion process to improve the efficiency of seed node selection but at the expense of accuracy.

In addition to these works, it is important to note that some variant problems based on the influence maximization problem have recently appeared. Gong et al. [34] considered the robustness assumption in the influence maximization problem and proposed the robust influence maximization problem. This problem focuses on the uncertainty factors affecting the information diffusion model and the algorithms. Seo et al. [35] changed the search target from individuals to communities and mined communities containing influential individuals in the network. They used a multi-dimensional vector to represent influence, where each dimension represents a kind of influence factor. Based on this, they proposed a search method across multiple impact criteria. Considering the dynamic behavior of online social networks, Singh et al. [36] studied the influence maximization problem in an online social network that evolves over time. In this framework, the first step is to predict the change of the network and then mine the seed nodes in the predicted network. Li et al. [37] paid attention to the problem of maximizing the positive influence in the signed social network. The formal definition of this problem is to select the top-K to send nodes with the largest positive influence in a given signed social network. They proposed a greedy algorithm for maximizing net positive influence based on game theory. Wu et al. [38] combined the problem of influence maximization with the diffusion of multi-source information and proposed the problem of multiple influence maximization (MIM), in which multiple pieces of information can be spread independently in the network. The goal of the multiple influence maximization problem is to maximize the overall cumulative influence spread of different information under the constraints of the seed budget k. Xie et al. [39] studied the problem of influence maximization in a competitive environment and analyzed the effect of inactive nodes and community homogeneity on information diffusion. Besides, to improve the effectiveness of seed nodes, some works have extended the classic IM problem by adding contextual features, such as topic, time, location [40–42].

Many improvements to the problem of influence maximization have been proposed, however, the efficiency of these methods is still unsatisfactory. Most of the previous work can be divided into two categories based on greedy algorithms and heuristic algorithms. The greedy-based algorithm uses a large number of Monte Carlo simulation processes to get high accuracy, but the computation is time-consuming. The heuristic algorithm is highly efficient, but it is easy to fall into the local optimal solution. Therefore, it is a particularly critical issue to design a more efficient and more accurate scheme for evaluating the influence of nodes, which can get a higher accuracy result and avoid a large number of Monte Carlo simulations.

3. Methodology

In this section, we first give a formal definition of the influence maximization problem and describe in detail the new framework proposed in this paper.

3.1. Preliminaries

In this part, we first give a formal definition of the online social networks, then define the problem of influence maximization based on the above, and finally introduce the information diffusion model used in this paper.

Definition 1. *An Online Social network with N users can be represented by a graph $G(V, E)$, Set of nodes V represents users, where $|V| = N$ and the set of edges E refer to the relationship between individuals. Influence can spread along the edges in the social network.*

Definition 2. *The definition of the influence maximization is to select K seed nodes for a given online social network $G(V, E)$ and an influence diffusion model M, which simulates how information spreads in the social network, to maximize the number of influenced nodes in G. Suppose S denotes the set of seed nodes and $R(S)$ be the number of nodes influenced by the selected seed nodes based on the diffusion model M. The formal representation is as follows:*

$$S^* = arg \max_{S \subset V, |S| = K} R(S). \tag{1}$$

The influence takes information as the carrier. For the information diffusion process, we focus on the spread of influence of the node when the process is over. Therefore, we use an Epidemic model, which is the most common model used to study information propagation as the information diffusion model. The susceptible infection (SI) model is one of the Epidemic models. In the SI model, in the beginning, the state of the seed nodes is set to the infected state, and the state of all other nodes is susceptible. A node changes from susceptible state to infected state when accepting information, and it can affect the susceptible nodes in its neighbor nodes according to probability p in the future. Once a node becomes infected, it will always maintain this state. This process ends when no more new nodes are infected [43].

3.2. Proposed Method

The influence maximization framework based on backward reasoning put forth by this paper consists of two parts. First, the first stage evaluates the ability of each node as a seed node to affect the entire network. In the second stage, seed nodes are selected according to the evaluation results and strategies.

3.2.1. Evaluate the Influence of Node Based on Maximum Likelihood Estimation

Suppose that a piece of information is sent out by a seed node v in an online social network G, and then the information continues to spread far away according to the given information diffusion model. When the process is over, we can find that N nodes are infected. Since the information spreads along the edges in the graph, so these infected nodes form the influence area of node v, which is a connected subgraph of G. The goal of maximizing influence is to select the node with the greatest influence as the seed nodes. Using backward reasoning, the greater the influence of the node, the larger the influence area. After removing the isolated nodes in the online social network G, the remaining nodes construct the largest connectivity subgraph G_M, which is also the largest influence area. In an online social network, the influence area of the most influential node should be the largest, assuming that the influence area of the best seed node is G_M. To find out the most influential node, we evaluate the probability of each node v as a seed node leading to the largest influence area G_M. The greater the probability, the higher the possibility that node v would affect the social network, that is, the greater the influence. To make this estimation, we assume that the seed node propagates information in G based on the above SI model and the nodes in G_M have a uniform prior probability. Based on this reasoning, a maximum likelihood estimator is constructed, which is defined as follows:

$$\hat{v} = arg \max_{v \subset G_M} P(G_M | v) \tag{2}$$

where $P(G_M|v)$ represents the probability of using node v as the seed node to simulate the information diffusion process according to the SI model, and the final influence area is G_M. Therefore, to select the seed nodes, we need to evaluate $P(G_M|v)$ for all $v \in G_M$.

However, we found that evaluation of $P(G_M|v)$ may not be computationally tractable. To solve this problem, we consider it from another perspective and transform the problem into a permutations problem, which is subject to the structure of G_M. As shown in Figure 1, if the seed node is the node v_1, it can only affect node v_3. After node v_3 is activated, the information may be transmitted to nodes v_2, v_4, v_5. According to this, the infected order $\{v_1, v_3, v_2, v_4, v_5\}$ and $\{v_1, v_3, v_5, v_2, v_4\}$ are permitted permutations. However, owing to the limitations of the network structure, the order of infection $\{v_1, v_5, v_2, v_4, v_3\}$ is impossible. Therefore, we need to find all these permitted permutations, where node v is the beginning node, and their corresponding probabilities to evaluate $P(G_M|v)$.

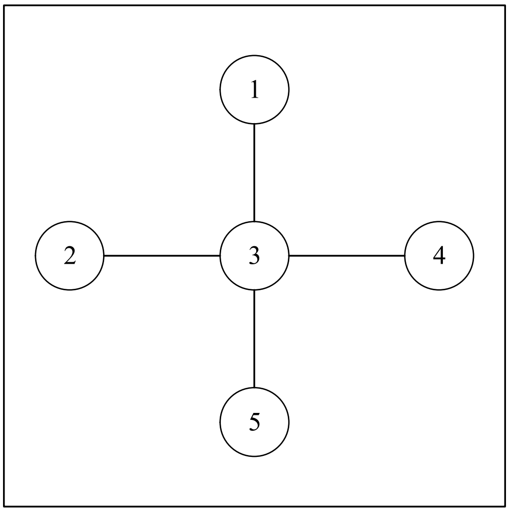


Figure 1. An online social network with 5 individuals.

Definition 3. *For a given online social network $G(V, E)$, we define $I(v, G)$ as the influence cardinality of node v. $I(v, G)$ is the total number of different permitted permutations of the nodes on G, which start from node $v \in G$ and are subject to the structure of graph G.*

When the problem is transformed, it becomes feasible to evaluate the $P(G_M|v)$. Assuming that $\Omega(v, G_M)$ is the set of all permitted permutations, in which source is v and influence area is G_M. σ represents a permitted permutation, which is also a spread path. In order to compute $P(\sigma|v)$ for every $\sigma \in \Omega(v, G_M)$, let $\sigma = \{v_1 = v, v_2, \cdots, v_N\}$, then

$$P(\sigma|v) = \prod_{k=2}^{N} P(k^{th} \text{ infected node} = v_k|G_{k-1}(\sigma), v) \tag{3}$$

where $G_k(\sigma)$ represents the subgraph of G_M. There are k nodes in the $G_k(\sigma)$, $\{v_1 = v, v_2, \cdots, v_k\}$ for $1 \leq k \leq N$.

Given the seed node v and $G_{k-1}(\sigma)$, the next node to be infected can be any of the uninfected neighbors of the infected nodes in $G_{k-1}(\sigma)$. Since the infection probability of all individuals is independent and identically distributed in the SI model, each of these uninfected nodes has an equal probability of becoming the next infected node. Assuming $G_{k-1}(\sigma)$ has $n_{k-1}(\sigma)$ uninfected neighboring nodes, we can simplify formula 3 to

$$P(\sigma|v) = \prod_{k=2}^{N} \frac{1}{n_{k-1}(\sigma)}. \tag{4}$$

Observing the above formula, we find that the problem of calculating $P(\sigma|v)$ has become to computing the size of the diffusion boundary $n_{k-1}(\sigma)$ for $2 \leq k \leq N$. In general, computing this boundary is complicated, we first assume that G_M is a regular tree, and then gradually change G_M from a tree to a graph, which is also helpful for understanding. In this way, suppose the k^{th} node added to $G_{k-1}(\sigma)$ is $v_k(\sigma)$, and the degree is $d_k(\sigma)$. Based on this, we can obtain

$$n_k(\sigma) = d_1(\sigma) + \sum_{i=2}^{k}(d_i(\sigma) - 2). \tag{5}$$

Next, it is easy to find

$$P(\sigma|v) = \prod_{k=2}^{N} \frac{1}{d_1(\sigma) + \sum_{i=2}^{k}(d_i(\sigma) - 2)}. \tag{6}$$

According to formula 6, if G_M is a d regular tree, every node has the same degree d in G_M, so for any node v and permitted permutation σ.

$$P(\sigma|v) = \prod_{k=1}^{N-1} \frac{1}{dk - 2(k-1)} \equiv p(d, N). \tag{7}$$

Therefore, the maximum likelihood estimator becomes

$$\begin{aligned} \hat{v} &\in arg \max_{v \in G_M} P(G_M|v) \\ &= arg \max_{v \in G_M} \sum_{\sigma \in \Omega(v, G_M)} P(\sigma|v) \\ &= arg \max_{v \in G_M} I(v, G_M) p(d, N) \\ &= arg \max_{v \in G_M} I(v, G_M) \end{aligned} \tag{8}$$

We can find that if G_M is a regular tree, the influence of node v can be evaluated simply by evaluating $I(v, G_M)$ for all $v \in G_M$. This is because the structure of the regular tree is special, and the probabilities of all permitted permutations are the same. When G_M is a general tree, it becomes computationally intractable since different permitted permutations have different probabilities. Due to the exponential number of terms involved, constructing a maximum likelihood estimator for a general tree could be computationally quite expensive. Therefore, we adopted a heuristic method to resolve the heterogeneity of degrees.

If the G_M is a general tree, assuming that information is transmitted between nodes in a breadth-first search (BFS) way, which is the fastest way of information diffusion.

In order to calculate the probability of BFS permitted permutations, the BFS permitted permutation with node v as the seed node is denoted as σ_v^*. Then in a general tree, the maximum likelihood estimation becomes as follows after derivation

$$\hat{v} \in arg \max_{v \in G_M} P(\sigma_v^*|v) I(v, G_M) \tag{9}$$

Next, if G_M is a general graph, calculating the probabilities of all possible permitted permutations in a given network is computationally prohibitive. Therefore, we also adopt a heuristic approach. As we all know, when a piece of information is sent out by a node, its spread path is a spanning tree of the graph. Therefore, suppose that if the node $v \in G_M$ is a seed node, after the information is sent from v, it will spread along a breadth first search tree rooted at v. This breadth first search tree, which is the fastest spread fashion of information, is denoted as $T_{bfs}(v)$. Therefore, we effectively obtain the following maximum likelihood estimator for a general graph:

$$\hat{v} \in arg \max_{v \in G_M} P(\sigma_v^*|v) I\Big(v, T_{bfs}(v)\Big) \tag{10}$$

We find that the node influence cardinality $I(v, G_M)$ plays a crucial role in evaluating whether the node v is suitable as a seed node. To calculate efficiently, we convert the graph into the corresponding tree structure in the process. So, in the following, we describe in detail how to efficiently compute $I(v, G_M)$ when G_M is a tree.

Let T_u^v denote the subtree rooted at node u, with node v as the seed node in a tree graph G_M. $|T_u^v|$ represents the number of nodes in T_u^v. To explain this definition clearly, an example is given in Figure 2. Figure 2 shows the spread path with node v_1 as the seed node. The subtree on the left has 3 nodes. It takes node v_2 as the root node and node v_1 as the seed node, so $|T_2^1| = 3$. Similarly, we can obtain $|T_7^1| = 1$.

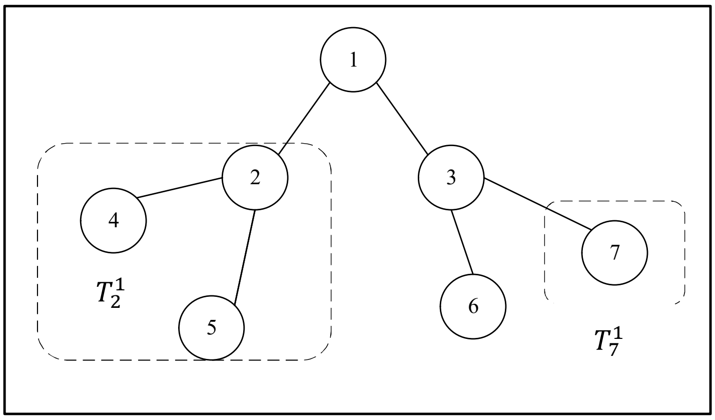


Figure 2. Illustration the definition of subtree variable T_u^v.

In a tree graph G_M with N nodes, to calculate $I(v, G_M)$, we need to find the permissible permutations of N nodes in the G_M. For a given permutation, there are N slots and the first of that must be the seed node v. Therefore, we only need to find the number of distinct ways to fill the remaining $N-1$ slots. These permutations are constrained by the structure of the graph, that is, a node u must be located before all nodes in its subtree T_u^v. According to this constraint, the number of permissible permutations of all nodes in T_u^v is $I(u, T_u^v)$. Based on this, the following relationship can be obtained.

$$I(v, G_M) = (N-1)! \prod_{u \in child(v)} \frac{I(u, T_u^v)}{|T_u^v|!} \tag{11}$$

where $child(v)$ denotes the set of all children nodes in the subtree rooted at v in the graph G_M.

We extend this recursion to the next depth level in G_M, and we will get

$$\begin{aligned} I(v, G_M) &= (N-1)! \prod_{u \in child(v)} \frac{I(u,T_u^v)}{|T_u^v|!} \\ &= (N-1)! \prod_{u \in child(v)} \frac{(T_u^v - 1)!}{|T_u^v|!} \prod_{w \in child(u)} \frac{I(w,T_w^v)}{|T_w^v|!} \\ &= (N-1)! \prod_{u \in child(v)} \frac{1}{|T_u^v|} \prod_{w \in child(u)} \frac{I(w,T_w^v)}{|T_w^v|!} \end{aligned} \tag{12}$$

A leaf node l only has one node and one permitted permutation, so $I\left(l, T_l^v\right) = 1$. Repeat this recursion until the leaf layer of the tree graph is reached, then the number of permitted permutations rooted at the seed node v in a given graph G_M can be calculated by the following formula.

$$I(v, G_M) = N! \prod_{u \in G_M} \frac{1}{|T_u^v|} \tag{13}$$

To evaluate the influence of nodes, we need to compute the influence cardinality of every node in G_M. The computational complexity of calculating the size of subtrees T_u^v for all u and v in a G_M with N nodes is $O\left(N^2\right)$. When the network scale becomes larger, the computational efficiency will be greatly reduced. To improve efficiency, we make a profound study of the relation between neighbor nodes. When node v and node u are two neighbor nodes in G_M, all their subtrees have the same size except for the subtrees rooted at u and v. Based on this discovery, a special relationship exists between two subtrees, as follows:

$$|T_u^v| = N - |T_v^u| \tag{14}$$

According to this relationship, we can calculate the influence cardinality of any two neighbor nodes.

$$I(u, G_M) = I(v, G_M)\frac{T_u^v}{N - T_u^v} \tag{15}$$

This special relationship is the key to optimizing the algorithm for calculating the influence cardinality of each node in G_M. Suppose that the node v is selected as the seed node in G_M. In order to obtain the size of all its subtrees T_u^v to calculate its influence cardinality $I(v, G_M)$, every other node u needs to pass two messages up to its parent node. The number of nodes in $u's$ subtree is the first message, which is represented by $t^{up}_{u\to parent(u)}$. The other message includes the cumulative product of the size of the subtree of every node in $u's$ subtree, expressed as $p^{up}_{u\to parent(u)}$. Then the parent node can get the size of its own subtree by adding the $t^{up}_{u\to parent(u)}$ messages together. Using these messages, the parent node can also get its cumulative subtree product by multiplying the $p^{up}_{u\to parent(u)}$ messages together. Starting from the leaf nodes, recursively pass the message upwards until the seed node receives the message. According to formula (13), the seed node multiplies the cumulative subtree products of its child nodes to obtain its influence cardinality $I(v, G_M)$. After that, we use formula (15) to evaluate the influence cardinality for the children of v. Each node passes its influence cardinality to its child nodes, which we call $r^{down}_{u\to w}$ for $w \in child(u)$. The complete process is as the following Algorithm 1. After optimization, the computational complexity of the algorithm that calculates the influence cardinality of all nodes is reduced from $O(N^2)$ to $O(N)$.

Algorithm 1 Calculate the influence cardinality based on message-passing

```
Input: An online social network $G(V, E)$
Output: Influence Cardinality of nodes in $G$
$G_{max} = f(G)$ //Find out the largest connectivity subgraph $G_{max}$.
$G_M = f(G_{max})$ //Choose a node $v$ as the seed node and change $G_{max}$ to BFS tree graph $G_M$ with $v$ as root.
for $u$ in $G_M$ do
    if $u$ is a leaf then
        $t^{up}_{u\to parent(u)} = 1$
        $p^{up}_{u\to parent(u)} = 1$
    else
        if $u$ is root $v$ then
            $\forall\ w \in child(v):\ r^{down}_{v\to w} = \frac{N!}{N\prod_{j\in child(v)} p^{up}_{j\to v}}$
        else
            $t^{up}_{u\to parent(u)} = \sum_{w\in child(u)} t^{up}_{w\to u} + 1$
            $p^{up}_{u\to parent(u)} = t^{up}_{u\to parent(u)} \prod_{w\in child(u)} p^{up}_{w\to u}$
            $\forall\ w \in child(u):\ r^{down}_{v\to w} = r^{down}_{parent(u)\to u}\frac{t^{up}_{u\to parent(u)}}{N-t^{up}_{u\to parent(u)}}$
        end if
    end if
end for
```

3.2.2. Select Seed Nodes Based on Strategy

In this part, we construct the set of seed nodes according to different strategies. In the previous, we evaluated the probability that each node will affect the entire network when it is used as a seed node in G_M. To maximize the spread of the seed set, we use a greedy strategy to select seed nodes. This strategy aims to select K most influential nodes from the candidate seed nodes. For each round, the node with the largest influence is selected to join the seed set. Algorithm 2 describes the complete process of the strategy, as follows

Algorithm 2 Construction of seed set based on greedy strategy

```
Input: The BFS tree subgraph $G_M$ of an online social network $G(V,E)$, the size of seed nodes set $K$
Output: the set of seed nodes $S$
1: while $|S| < K$ do
2:     $s_a = arg \max_{v \subset G_M - S} P(G_M \big| v)$
3:     $\mathrm{S} = \mathrm{S} \cup \ s_a$
4: end while
```

4. Experiments and Discussion

In this section, a series of experiments with different specifications are carried out for highlighting the performance and efficiency of the proposed scheme. This section is divided into two parts. The datasets used in the experiments and the baseline algorithms are introduced in the first part, and the remaining part is the analysis of the experimental results.

4.1. Experiment Setup

We compare the performance of five different algorithms on four real social network datasets. These four datasets are crawled from real online social networks, and the compared algorithms include classic algorithms and the latest algorithms. To avoid randomness in the process of selecting seed nodes, each algorithm was run 50 times independently.

4.1.1. Dataset Description

To verify the performance of our framework, we conduct experiments on four different online social network datasets. These datasets are detailed in Table 1. The second column represents the name of the online social network dataset. The number of edges and nodes of the dataset are shown in the third and fourth columns. It can be found from the table that we use different scale datasets to make the experiment more convincing. The average degree of nodes in the online social network is listed in the fifth column. The last two columns of the table respectively show the average clustering coefficient and diameter of the network.

Table 1. The description of datasets.

No.	Name	Edges	Nodes	Average Degree	Average Clustering Coefficient	Diameter
1	Facebook	88,234	4039	43.7	0.61	8
2	ca-HepTH	25,998	9877	5.3	0.47	17
3	Epinions	508,837	75,879	6.7	0.14	14
4	Twitter	1,768,149	81,306	43.5	0.57	7

4.1.2. Baseline Algorithms

To demonstrate the superior performance of the framework IMBR, our algorithm is compared with several existing algorithms. The algorithms to be compared are summarized below.

Degree-Discount [10]: This is a heuristic algorithm based on node degree, which selects the nodes with the largest out-degree as the seed nodes. Besides, it considers the influence of neighbor nodes. If its neighbor is selected as the seed node, the out-degree of the node will decrease by one.

Upper Bound based Lazy Forward (UBLF) [11]: This is a greedy algorithm, it uses the upper bound based lazy forward to mine the top-K influential nodes in an online social network. UBLF has the same accuracy and less running time than other greedy algorithms such as CELF and CELF++.

Influence Maximization Algorithm Based on Ant Colony Optimization (ACO-IM) [24]: This algorithm uses ant colony optimization to maximize the influence of the set of seed nodes. According to the fitness function, the best solution is selected in the iterative process and added to the seed set.

Community-based Greedy Algorithm (CGA) [17]: This algorithm first detects the structure of the community, and then exploits a dynamic programming algorithm to find influential nodes in these communities.

Sketch-based Impact Maximization Algorithm (SKIM) [27]: This is a greedy algorithm in the sketch space. The key of the algorithm is per-node summary structures.

Stop-and-Stare Algorithm (SSA) [28]: This is a sampling-based framework for the influence maximization problem, which is faster than other sampling algorithms.

4.1.3. Evaluation Measure

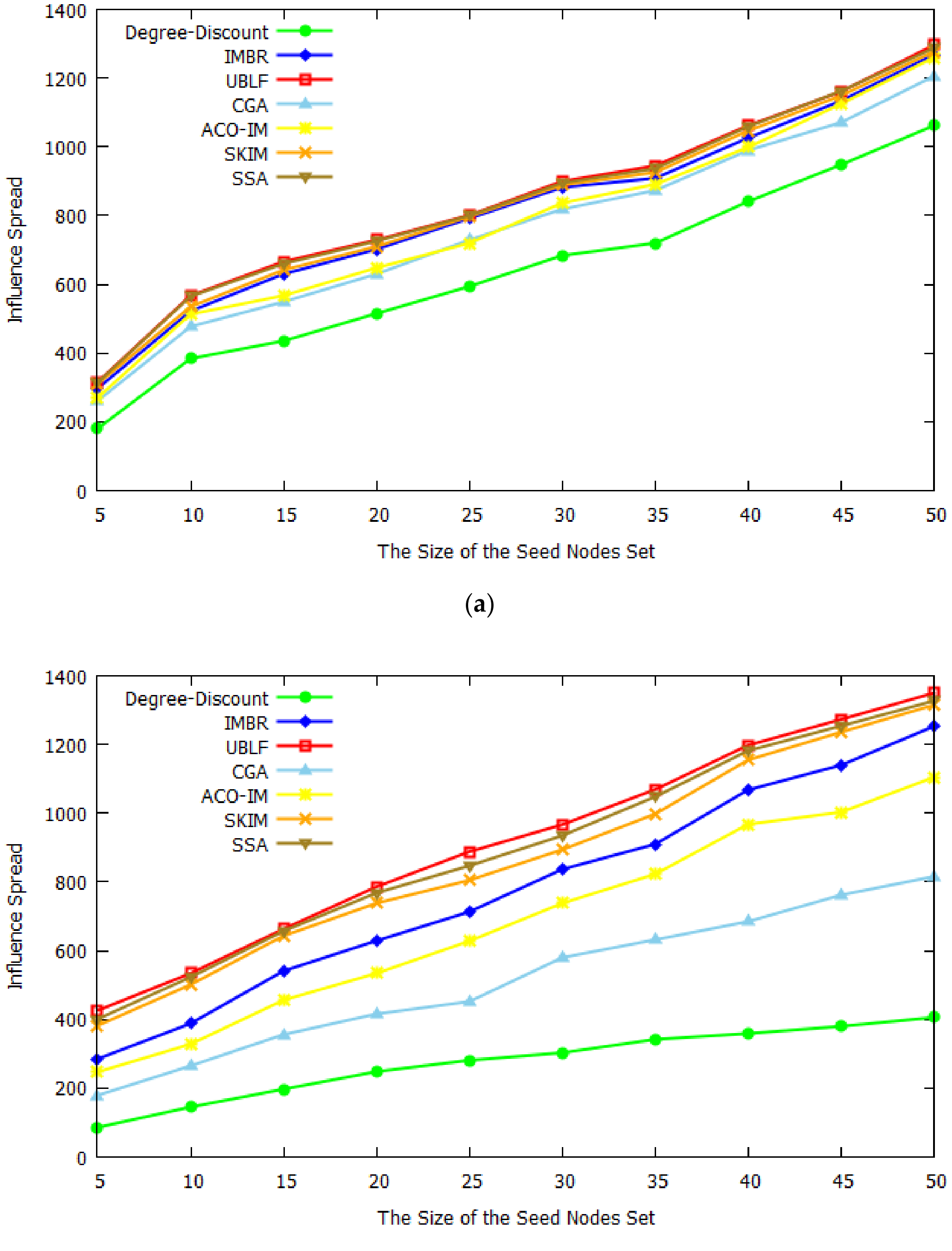
According to the previous analysis, the goal of influence maximization is to mine k seed nodes with the greatest influence to maximize the final influence range. In order to measure the performance of the algorithm, we applied each algorithm to different datasets and get k seed nodes respectively. Then, the seed nodes are used as the information source, and the process of information spreading on the corresponding network is simulated based on the SI model. When the information spreading process is over, the number of affected nodes is used as the influence spread of the seed nodes. We use the influence spread of the seed nodes as a measure of the effectiveness of the algorithm.

In the experimental part, the probability of an individual changing from the susceptible state to the infected state in the SI model is set to $p = 0.08$, and each propagation process goes through 1000 iterations. The range of k ranges from 5 to 50, with 5 as an interval.

4.2. Result and Discussion

To show the performance of our IMBR algorithm, we applied IMBR to different datasets and compared it with six rival benchmark algorithms from different aspects. The first measure to be evaluated is the influence spread of the seed nodes selected by different algorithms, which is the most important in the problem of influence maximization. We use the seed nodes obtained in the experiment to perform 1000 Monte Carlo simulations and take the average of the simulation results as the final influence spread of the set of seed nodes. Applying IMBR and other algorithms to different real online social network datasets, the results are shown in Figure 3.

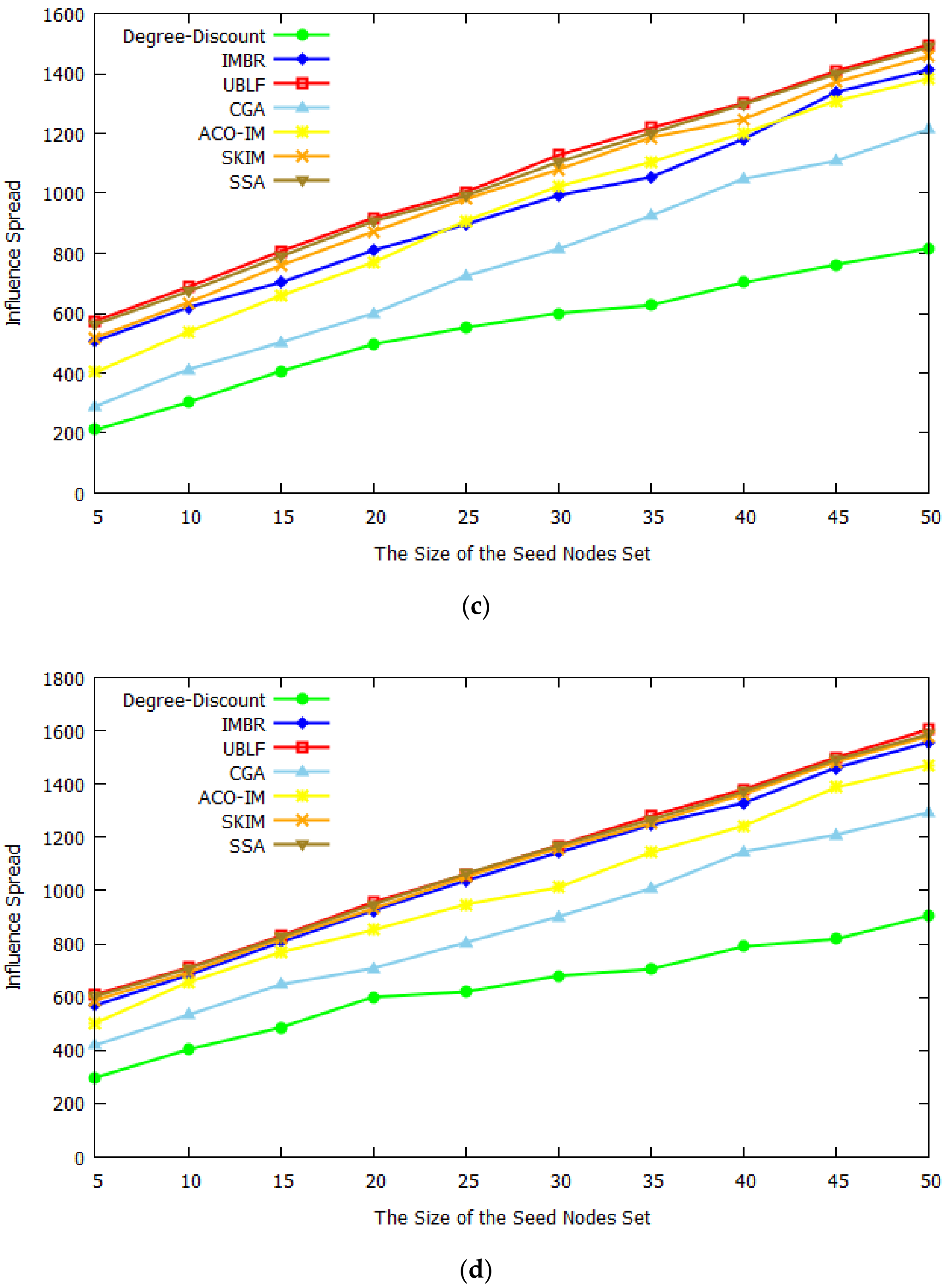
Analyzing these figures carefully, we can find that as the size of the seed nodes set K increases, the influence spread becomes larger in all datasets. Greedy algorithm UBLF performed best, and the results of SSA and SKIM were close to that of UBLF. At the same time. Another important result is that our algorithm IMBR has achieved similar results to SKIM, which provides a piece of evidence for the accuracy of our algorithm. The idea of the maximum likelihood estimator is to evaluate the influence of nodes in the diffusion process based on the entire network structure. It considers the entire process of information diffusion, which leads to higher accuracy of the results. In addition, the seed nodes set selected by Degree-Discount has a poor result in different situations. This is because it considers less node information in the process of selecting, which improves efficiency at the expense of accuracy. Moreover, the stability of Degree-Discount is not good. As the size of the datasets increases, the gap between Degree-Discount and other algorithms gradually becomes larger. The performance of the ACO-IM algorithm is second only to our algorithm. The seed nodes selected by the CGA algorithm are not good, which is due to the CGA algorithm mines influential seed nodes in different communities while ignoring information dissemination between communities. It is worth noting that the gap on the ca-HepTH and Epinions datasets between other algorithms and the UBLF, which is based on the greedy algorithm are larger than that on the Facebook and Twitter datasets. Through the analysis of the datasets, we found that owing to the sparse network structure of the datasets, the average node degree in the ca-HepTH and Epinions datasets is only 5.3 and 6.7, respectively, while the average node degree in the Twitter and Facebook datasets is 43.7 and 43.5, respectively.



(**a**)

(**b**)

Figure 3. *Cont.*



(**c**)

(**d**)

Figure 3. Comparison of the influence spread of IMBR with other algorithms on different datasets. (**a**) Comparison of the influence spread of IMBR with six rival algorithms on the Facebook dataset. (**b**) Comparison of the influence spread of IMBR with six rival algorithms on the ca-HepTH dataset. (**c**) Comparison of the influence spread of IMBR with six rival algorithms on the Epinions dataset. (**d**) Comparison of the influence spread of IMBR with six rival algorithms on the Twitter dataset.

Next, we analyze the impact of network type and network properties on the results of our algorithm. From the above analysis, it is found that the greedy algorithm UBLF has the highest accuracy. To facilitate the analysis, we normalize the accuracy of the algorithm according to the following formula:

$$Acc_{IMBR} = \frac{ISP_{IMBR}}{ISP_{UBLF}}. \quad (16)$$

where Acc_{IMBR} represents the normalized accuracy of our algorithm IMBR, and ISP_{IMBR} denotes the influence spread of the seed node selected by the IMBR algorithm. Similarly, ISP_{UBLF} represents the influence spread of the seed nodes selected by the UBLF algorithm.

The following table shows the normalized accuracy of our algorithm under different data sets. The second to tenth columns of Table 2 indicates the normalized accuracy of the IMBR algorithm when k takes different values. The last column of Table 2 represents the average of the normalized accuracy under different datasets.

Table 2. Normalized accuracy of IMBR algorithm on different datasets.

	5	**10**	**15**	**20**	**25**	**30**	**35**	**40**	**45**	**50**	**Average**
Facebook	0.94	0.92	0.95	0.96	0.99	0.98	0.96	0.97	0.98	0.98	0.96
ca-HepTH	0.67	0.54	0.82	0.8	0.8	0.87	0.85	0.89	0.9	0.93	0.81
Epinions	0.88	0.9	0.87	0.88	0.89	0.88	0.87	0.91	0.95	0.94	0.9
Twitter	0.93	0.96	0.97	0.97	0.98	0.98	0.97	0.96	0.97	0.97	0.97

With further analysis of the experimental results, it is found that the IMBR algorithm has the highest accuracy on the Twitter dataset and the worst performance on the ca-HepTH dataset; when k is larger, the accuracy of the IMBR algorithm is higher. Moreover, to explore the relationship between the results and network properties, a comparative analysis of Tables 1 and 2 shows that the smaller the diameter of the network, the higher the accuracy of the IMBR algorithm. This is because the smaller the network diameter, the shorter the path for each individual to receive information. The IMBR algorithm evaluates the influence of nodes based on the spread path. The shorter the spread path, the smaller the evaluation error.

In the last part of the experiment, we compared the running time between our algorithm and the other six rival algorithms on different online social network datasets. We fixed the size of seed nodes to 50 and compared the cost time of different algorithms. The result is shown in Figure 4.

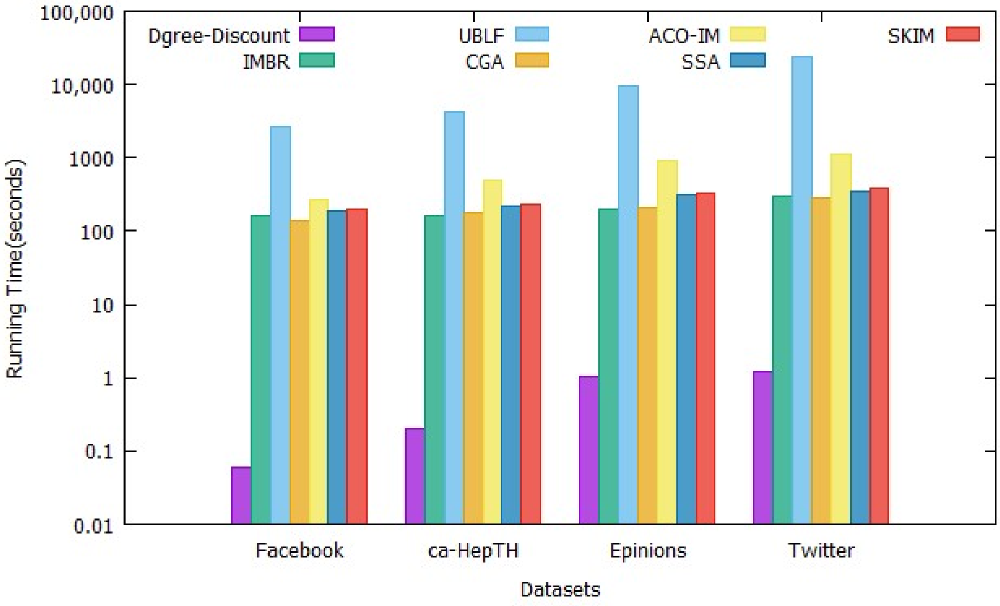


Figure 4. Comparison of running time between IMBR and other six rival algorithms.

The result in Figure 4 reveals that the running time of the IMBR proposed in this paper is similar to the running time of CGA. UBLF has the longest running time, followed by ACO-IM, and Degree-Discount has the shortest running time. Degree-Discount algorithm saves time at the expense of accuracy. In addition, it can be found that the running time of our algorithm IMBR is faster than SSA and SKIM. Our algorithm IMBR avoids Monte Carlo simulations, and the calculation method based on message-passing greatly reduces the computational complexity. Compared with the greedy algorithm UBLF, IMBR significantly improves computational efficiency.

Based on the experimental results, we conclude that our proposed scheme based on the maximum likelihood estimator has found an optimal solution in the trade-off between time and accuracy. In terms of running time and influence spread, IMBR is not the best among all algorithms. However, compared with the UBLF based on the greedy algorithm, IMBR achieves a higher running speed. Compared with the Degree-Discount algorithm, which is the fastest algorithm, IMBR maintains a higher accuracy. Compared with the CGA and ACO-IM algorithms, IMBR has a great accuracy improvement. Compared with SKIM and SSA, IMBR has improved efficiency while maintaining accuracy close to SKIM. Considering the computational efficiency and accuracy, IMBR strikes a better balance between efficiency and accuracy.

5. Conclusions

With the rapid development of the Internet, online social networks have greatly changed human lives. This change has attracted many researchers to devote themselves to studying online social networks. Influence maximization is one of the hot research issues in this field. To solve this problem, many algorithms have been proposed. These works are mainly divided into two categories: algorithms based on greed and algorithms based on heuristic information. Greedy algorithms have high accuracy, but poor efficiency. The reason is that the Monte-Carlo simulations used by greedy algorithms are time-consuming. The heuristic algorithms are efficient, but the accuracy is not high. To find a better balance between efficiency and accuracy, we continued to study this issue in depth.

In this paper, we propose a framework of influence maximization based on backward reasoning in online social networks. Using this new framework, it is possible to estimate the ability of each node as a seed node to affect the entire network, which can measure the influence of nodes. The seed nodes can be selected based on this estimation using a greedy strategy. A series of experiments were conducted on real online social network datasets to explore the advantages of our framework. The results show that our method significantly shortens the running time under the premise of ensuring the accuracy of the results. In other words, the proposed framework achieves a good balance between efficiency and accuracy.

In future work, we intend to conduct in-depth research based on this work. One potential extension is based on this work to study the problem of maximizing marketing revenue under budget constraints. Another potential direction is to delve into the problem of influence maximization when the entire network structure cannot be obtained.

Author Contributions: Conceptualization, K.L.; Project administration, K.L.; Resources, K.L.; Supervision, K.L.; Formal analysis, L.Z.; Investigation, L.Z.; Methodology, L.Z.; Software, L.Z.; Writing—original draft, L.Z.; Writing—review and editing, L.Z., K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Beijing Natural Science Foundation, China (No. L181010, 4172054) and National Key R & D Program of China (No. 2016YFB0801100).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The four online social network datasets used in the experiment can be downloaded at http://snap.stanford.edu/data/, accessed on 12 April 2021.

Conflicts of Interest: The authors declare that there is no conflict of interest. The funders have no role in the research process and the writing of the manuscript.

References

1. Wang, F.; Wang, H.; Xu, K. Diffusive logistic model towards predicting information diffusion in online social networks. In Proceedings of the 32nd International Conference on Distributed Computing Systems Workshops, Macau, China, 18–21 June 2012; pp. 133–139.
2. Li, K.; Zhang, L.; Huang, H. Social Influence Analysis: Models, Methods, and Evaluation. *Engineering* **2018**, *4*, 40–46. [CrossRef]
3. Kempe, D.; Kleinberg, J.; Tardos, E. Maximizing the spread of influence through a social network. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–27 August 2003; pp. 137–146.
4. Domingos, P.; Richardson, M. Mining the network value of customers. In Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 26–29 August 2001; ACM: New York, NY, USA, 2001; pp. 57–66.
5. Kempe, D.; Kleinberg, J.; Tardos, E. Influential nodes in a diffusion model for social networks. In Proceedings of the 32nd International Conference on Automata, Languages and Programming, Lisbon, Portugal, 11–15 July 2005; pp. 1127–1138.
6. Sviridenko, M. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.* **2004**, *32*, 41–43. [CrossRef]
7. Leskovec, J.; Krause, A.; Guestrin, C.; Faloutsos, C.; VanBriesen, J.; Glance, N. Cost-effective outbreak detection in networks. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD2007, San Jose, CA, USA, 12–15 August 2007; Association for Computing Machinery: New York, NY, USA, 2007; pp. 420–429.
8. Goyal, A.; Lu, W.; Lakshmanan, L.V.S. CELF++: Optimizing the greedy algorithm for influence maximization in social networks. In Proceedings of the 20th International Conference Companion on World Wide Web 2011, Hyderabad, India, 28 March–1 April 2011; ACM: New York, NY, USA, 2011; pp. 47–48.
9. Estevez, P.; Vera, P.; Saito, K. Selecting the Most Influential Nodes in Social Networks. In Proceedings of the International Joint Conference on Neural Networks, Orlando, FL, USA, 12–17 August 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 2397–2402.
10. Chen, W.; Wang, Y.; Yang, S. Efficient influence maximization in social networks. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; ACM: New York, NY, USA, 2009; pp. 199–208.
11. Zhou, C.; Zhang, P.; Zang, W.; Guo, L. On the upper bounds of spread for greedy algorithms in social network influence maximization. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 2770–2783. [CrossRef]
12. Lu, F.; Zhang, W.; Shao, L.; Jiang, X.; Xu, P.; Jin, H. Scalable influence maximization under independent Cascade model. *J. Netw. Comput. Appl.* **2017**, *86*, 15–23. [CrossRef]
13. Ge, H.; Huang, J.; Di, C.; Li, J.; Li, S. Learning automata based approach for influence maximization problem on social networks. In Proceedings of the 2017 IEEE Second International Conference on Data Science in Cyberspace (DSC), Shenzhen, China, 26–29 June 2017; pp. 108–117.
14. Tian, J.; Wang, Y.; Feng, X. A new hybrid algorithm for influence maximization in social networks. *Chin. J. Comp.* **2011**, *34*, 1956–1965. [CrossRef]
15. Wang, Y.; Feng, X. A potential-based node selection strategy for influence maximization in a social network. In Proceedings of the Advanced Data Mining and Applications, 5th International Conference, ADMA 2009, Beijing, China, 17–19 August 2009; pp. 350–361.
16. Ohsaka, N.; Akiba, T.; Yoshida, Y.; Kawarabayashi, K. Fast and accurate influence maximization on large networks with pruned monte-carlo simulations. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2014, Québec City, QC, Canada, 27–31 July 2014; AAAI Press: Menlo Park, CA, USA; pp. 138–144.
17. Wang, Y.; Cong, G.; Song, G.; Xie, K. Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; ACM: New York, NY, USA, 2010; pp. 1039–1048.
18. Khomami, M.M.D.; Rezvanian, A.; Meybodi, M.R.; Bagheri, A. CFIN: A community-based algorithm for finding influential nodes in complex social networks. *J. Supercomput.* **2020**, *77*, 2207–2236. [CrossRef]
19. Kundu, S.; Murthy, C.; Pal, S. A New Centrality Measure for Influence Maximization in Social Networks. In Proceedings of the Pattern Recognition & Machine Intelligence-International Conference, Moscow, Russia, 27 June–1 July 2011; pp. 242–247.
20. Kim, J.; Kim, S.; Yu, H. Scalable and Parallelizable Processing of Influence Maximization for Large-Scale Social Networks. In Proceedings of the Twenty-Ninth International Conference on Data Engineering, Brisbane, Australia, 8–12 April 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 266–277.
21. Kimura, M.; Saito, K. *Tractable Models for Information Diffusion in Social Networks*; PKDD 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 259–271.

22. Goyal, A.; Lu, W.; Lakshmanan, L. Simpath: An efficient algorithm for influence maximization under the linear threshold model. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining, ICDM, Vancouver, BC, Canada, 11–14 December 2011; pp. 211–220.
23. Chen, W.; Yuan, Y.; Zhang, L. Scalable influence maximization in social networks under the linear threshold model. In Proceedings of the 2010 IEEE 10th International Conference on Data Mining, ICDM, Sydney, Australia, 13–17 December 2010; pp. 88–97.
24. Singh, S.; Singh, K.; Kumar, A.; Biswas, B. ACO-IM: Maximizing influence in social networks using ant Colony optimization. *Soft Comput.* **2020**, *24*, 10181–10203. [CrossRef]
25. Jung, K.; Heo, W.; Chen, W. IRIE: Scalable and robust influence maximization in social networks. In Proceedings of the 2012 IEEE 12th International Conference on Data Mining, ICDM, Brussels, Belgium, 10–13 December 2012; pp. 918–923.
26. Tang, Y.; Xiao, X.; Shi, Y. Influence maximization: Near-optimal time complexity meets practical efficiency. In Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, Snowbird, UT, USA, 22–27 June 2014; pp. 75–86.
27. Cohen, E.; Delling, D.; Pajor, T.; Werneck, R.F. Sketch-based Influence Maximization and Computation: Scaling up with Guarantees. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (CIKM 2014), New York, NY, USA, 3–7 November 2014; pp. 629–638.
28. Nguyen, H.T.; Thai, M.T.; Dinh, T.N. Stop-and-Stare: Optimal Sampling Algorithms for Viral Marketing in Billion-scale Networks. In Proceedings of the 2016 International Conference on Management of Data (SIGMOD 2016), New York, NY, USA, 26 June–1 July 2016; pp. 695–710.
29. Chen, W.; Wang, C.; Wang, Y. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1029–1038.
30. Chen, Y.; Zhu, W.; Peng, W.; Lee, W.; Lee, S. Cim: Community-based influence maximization in social networks. *ACM Trans. Intell. Syst. Technol.* **2014**, *5*, 45–75. [CrossRef]
31. Singh, S.; Singh, K.; Kumar, A.; Biswas, B. Coim: Community-Based Influence Maximization in Social Networks. In Proceedings of the ICAICR: International Conference on Advanced Informatics for Computing Research, Shimla, India, 14–15 July 2018; Springer: Singapore, 2018; pp. 440–453.
32. Narayanam, R.; Narahari, Y. A shapley valuebased approach to discover influential nodes in social networks. *IEEE Trans. Autom. Sci. Eng.* **2011**, *8*, 130–147. [CrossRef]
33. Borgs, C.; Brautbar, M.; Chayes, J.; Lucier, B. Maximizing social influence in nearly optimal time. In Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, Portland, OR, USA, 5–7 January 2014; pp. 946–957.
34. Gong, Y.; Liu, S.; Bai, Y. Efficient parallel computing on the game theory-aware robust influence maximization problem. *Knowl.-Based Syst.* **2021**, *220*, 106942. [CrossRef]
35. Seo, J.H.; Kim, M.H. Finding influential communities in networks with multiple influence types. *Inf. Sci.* **2021**, *548*, 254–274. [CrossRef]
36. Singh, A.K.; Kailasam, L. Link Prediction-Based Influence Maximization in Online Social Networks. *Neurocomputing* **2021**, *453*, 151–163. [CrossRef]
37. Li, D.; Wang, Y.J.; Li, M.H.; Sun, X.; Pan, J.C.; Ma, J. Net Positive Influence Maximization in Signed Social Networks. *Intell. Fuzzy Syst.* **2021**, *41*, 3821–3832. [CrossRef]
38. Wu, G.; Gao, X.; Yan, G.; Chen, G. Parallel Greedy Algorithmto Multiple Influence Maximization in Social Network. *ACM Trans. Knowl. Discov. Data* **2021**, *15*, 1–21.
39. Xie, X.; Li, J.; Sheng, Y.; Wang, W.; Yang, W. Competitive influence maximization considering inactive nodes and community homophily. *Knowl.-Based Syst.* **2021**, *233*, 107497. [CrossRef]
40. Singh, S.; Kumar, A.; Singh, K.; Biswas, B. C2IM: Community based context-aware influence maximization in social networks. *Phys. A Stat. Mech. Its Appl.* **2019**, *514*, 796–818. [CrossRef]
41. Min, H.Y.; Cao, J.X.; Yuan, T.F.; Liu, B. Topic based time-sensitive influence maximization in online social networks. *World Wide Web* **2020**, *23*, 1831–1859. [CrossRef]
42. Guo, J.X.; Wu, W.L. Adaptive Influence Maximization: If Influential Node Unwilling to Be the Seed. *ACM Trans. Knowl. Discov. Data* **2021**, *15*, 1–23. [CrossRef]
43. Daley, D.J.; Kendall, D.G. Epidemics and Rumours. *Nat. Cell Biol.* **1964**, *204*, 1118. [CrossRef] [PubMed]