SN Computer Science (2022) 3:444
https://doi.org/10.1007/s42979-022-01342-3

ORIGINAL RESEARCH

Promotional Predictive Marketing: User Centric Data Driven Approach

Soumi Tokdar[1] · Ananya Kanjilal[1] · Sankhayan Choudhury[2] · Swapan Bhattacharya[3]

Received: 23 August 2021 / Accepted: 21 July 2022 / Published online: 16 August 2022

Abstract

This paper proposes value addition to the classical Influence Maximization problem by introducing a quality measure to the participating nodes. The quality measure signifies the '*propensity to buy*' of a customer (node) in a promotional marketing campaign context. Two metrics, Individual Net Worth (INW) and Neighborhood Net Worth (NNW) are proposed to measure the potential of a customer(s) in buying a given product. The proposed solution, through a heuristic approach, is capable of spreading the influence to the customers with a higher propensity to buy the product. The solution is scalable and adaptable to address user requirements. All these claims are substantiated through experimental results on public datasets. We performed a comparative study with notable algorithms in this domain. The result shows that the proposed approach selects seeds of higher quality as well as maximizes the overall quality (worth) of the influenced nodes in comparison to the notable algorithms, without any adverse impact on time complexity.

Keywords Target influence maximization · Heuristic-based seed selection · Customer net worth · Promotional predictive marketing

Introduction

The objective of the Influence Maximization (IM) problem is to identify a certain number of key individuals (seed set) that spread the influence in a social network such that the expected number of influenced individuals (Influenced set) is maximized [22]. In most of the existing works [4, 7], the nodes in a community graph are considered to be homogenous with the same propensity to be influenced. Moreover, the existing solutions [4, 7] are independent of the business requirements and hence may not necessarily target the right customers. But in real life, a customer's affinity for buying a specific product is not homogeneous and that needs to be addressed within the solution. Each customer in a community is associated with various attributes like age, topic sensitivity, income group, location, etc. These attributes collectively indicate a specific person's buying propensity for an identified product.

Some existing works [4, 6, 7] use structural properties for maximizing the influence. The structural properties of a social network indicate the properties related to connectivity such as degree centrality, betweenness centrality, common neighbor, etc. These works [4, 6, 7] aim to maximize the number of influenced nodes but there is a chance of selecting some nodes with a lower propensity to buy. In the context of a real-life marketing campaign scenario, some of the customers are more attentive compared to others. As a result, the solution should attempt to maximize the influence in terms of the number of nodes (customers) with a higher chance to respond to the campaign. The solution becomes more effective if it ensures maximum dissemination to interested customers. Therefore, it is essential to consider each node's "propensity to buy" for identifying the customers with a higher chance to respond. Our work presented in this paper is aligned with some existing works as in [2, 3, 11, 14, 15, 17, 18] that focus on this direction.

✉ Soumi Tokdar
soumi.tokdar@gmail.com

Ananya Kanjilal
kananya2011@gmail.com

Sankhayan Choudhury
sccomp@caluniv.ac.in

Swapan Bhattacharya
bswapan2000@yahoo.co.in

[1] Department of CSE, BPPIMT, Kolkata, India

[2] Department of CSE, University of Calcutta, Kolkata, India

[3] Deparment of CSE, Jadavpur University, Kolkata, India

In addition, the companies or brands marketing a specific product also have varied requirements. Two different brands will use different strategies to target a specific customer group for the same product. For example, Levi's launching a new T-shirt brand may focus on customers within a particular income group and located in 'Kolkata' while Adidas may target young customers who are sports lovers rather than any particular income group or location. Thus, all the customers should not be considered homogeneously. It is therefore essential to consider the specific requirement of a brand within the proposed solution. The proposed work addresses the following crucial issues:

1. Formalize customer propensity to buy based on individual node attributes.

2. Consider user (marketing brand) requirements within the solution framework in an adaptive way.

As a first step toward achieving our objective, we introduce the concept of the "worth" of an individual as a measure of its propensity to be influenced. Worth is a measure of the goodness or a measure of the quality of a node for a given marketing requirement based on its attributes. Worth is formulated as a weighted sum of the chosen attributes. To address the above-said issues, we propose two metrics—'Individual Net Worth' (INW) and 'Neighborhood Net Worth' (NNW).

INW is a measure used to identify high-quality nodes or spot the customers having a higher chance of buying. Likewise, NNW quantifies the strength of a node in terms of the quality (total buying propensity) of its neighborhood. It is used to select seed nodes through which the maximum customers with higher propensity could be influenced. In other words, the objective is to maximize the NNW of the influenced set. At any instant, a set of nodes that are already influenced through the dissemination process forms the influence set.

In addition, we design the solution such that users (brand) can provide their inputs in selecting the acceptable value or range of values of the customer attributes for their promotion. Moreover, they can prioritize the attributes according to their choice, thereby making the solution flexible and adaptable.

Review of Related Work

In this section we broadly categorize the review works in two directions—(1) time-efficient solution and (2) context-sensitive solution.

Approaches Aiming for a Time-Efficient Solution

The authors [8] proved that influence maximization is NP-hard under traditional diffusion models. They demonstrated that the solution using the greedy algorithm is approximated within a factor of $(1 - \frac{1}{e} - \varepsilon)$. But despite giving a near-optimal solution, it lacks in time efficiency and scalability. To overcome these challenges several works on sub-modularity-based approaches, centrality-based approaches, influence path-based approaches, and community-based approaches are proposed. CELF in [7] used sub-modular properties of a graph with a lazy forward optimization technique to find a 700 times faster solution than the greedy algorithm [8]. CELF++ in [6] further optimizes the running time of CELF by almost 35–55% by computing the marginal gain of each node. The authors in [5] proposed a centrality-based evolutionary algorithm called degree-descending search evolution (DDSE) which eliminates the time-consuming simulations of the greedy algorithms significantly. The work in [10] presented an influence path-based algorithm Matrix Influence (MATI) which aims to generate a more scalable solution. MATI is based on the pre-calculation of the influence by taking advantage of the simple paths in a node's neighborhood. TOPSIS [16] is another model which worked on the selection of seeds for getting minimum overlap and maximum coverage while providing comparable spread with better runtime than other state-of-the-art methods. The degree discount algorithm provided in [4] uses a heuristic-based degree reduction technique to reduce seed selection time significantly while providing a better spread than the other degree and centrality-based algorithms.

All these works focus on the IM problem from a time efficiency perspective. However, they are not concerned about the quality of the nodes selected in the context of the problem. A larger influenced set in an optimized runtime does not necessarily guarantee nodes that will potentially respond to a promotional campaign.

Approaches Aiming for a Context-Sensitive Solution

In this section, we present a review of works that customizes the basic influence maximization approach based on context information leading to a target influence maximization problem. Context information may be location, user attributes like income, age, etc. Context information serves as a basis for the measure of the quality of the node getting influenced. Nodes that satisfy or align with the context are the "target" nodes and are desirable for the promotional marketing campaign. Some of the works presented here consider a single attribute context like "topic" as in [2, 3, 11] while some others [17, 18] consider multiple attributes for defining their target nodes.

The authors in [2, 3, 11] focus on topic-sensitive node selection considering a single attribute "topic". In [3], the authors consider each edge of the graph has a score for different topics based on user interaction on the topic. This score reflects users' interest and is used for topic-aware influence maximization to propagate the influence along the topic-inclined edges. Another work [2] aims to identify topic-based experts using large datasets collected from Twitter. They consider topics in Twitter posts and utilize user-specific features to identify seed nodes that are topical influencers i.e., experts on a topic. In [11], the authors partition the network into communities and remove non-desirable nodes based on the topic. Thereafter, they consider the degree distribution of the nodes along with the user's interests to generate an effective seed set.

Some works as in [1, 9, 12, 14, 15] consider other contexts like rate of interactions, temporal activities, location, etc. for influence maximization. The authors of [1] aim to find dense communities based on active edges i.e., edges with constant communication for propagating their influence spread. They used both temporal activity and topic to provide a topic-sensitive time-variant solution for influence propagation. In [12], the authors have used the interaction rate of each node as an attribute to prune the social graph and choose seed nodes effectively. The authors in [14] used a page rank-based model to assign costs to nodes based on their importance. In another work [15], the authors propose an efficient method of influence maximization based on location-aware queries such that the influence is maximized in the query region. The work proposed in [9] aims to maximize the influence in a signed review network. It selects the seeds in such a way that the number of influenced consumers with interest in the advertised product category will be maximum. Here, the polarity of the social relationships is considered for propagating the influence.

Some recent works as in [17, 18] consider multiple attributes associated with a node in a network for maximizing influence spread. The authors in [17] use multiple factors related to a node, such as information content, social influence, and user authority to propagate the influence among the nodes. They consider multiple node-related attributes for targeting the nodes with a higher probability to be influenced. On the other hand, the authors of [18] define an attribute category set for target nodes and use the similarity between the attributes associated with each node with this target attribute set for effective influence maximization. However, it has limited scope for tuning the target attribute set as per the choice of the customer which forms the basis of our work in this paper. The above works attempt to address the IM problem based on node attributes and interactions among them. However, there is limited scope for considering a combination of several attributes with different weights. In other words, there is no scope for user intervention to generate an adaptive solution.

In our earlier work [13], we presented a comprehensive analysis of the state-of-the-art approaches for identifying influential users in a social network. Here, the objective is to differentiate the nodes in terms of quality considering multiple attributes of the nodes. The quality implies the propensity of a customer to respond to the promotional campaign. Finally, the dissemination will be maximum within quality nodes. Moreover, the approach will be a tuneable solution by including the user/brand's requirement for a target customer.

Motivation and Scope of Work

Significant works as in [4–6, 16], target time-efficient scalable solutions but do not focus on the effectiveness of seeds in terms of the quality of the influenced set. Few attempts are there [2, 3, 9, 14, 17, 18] to address this issue through specific context but the generated solutions are rigid in terms of adapting to any combination of these attributes. In the proposed work, an adaptive approach is introduced where the user (brand/marketing agency) can tune the weights or significance levels of node attributes (customer characteristics) based on their requirements.

We aim to analyze the customer characteristics (node attributes) in a community network and generate the seed set (influencers) to maximize the total buying potential of the influenced set. A metric to formalize "Individual Net Worth" (INW) is proposed in the context of domain-specific parameters associated with each node. We also propose another metric "Net Neighborhood worth" (NNW) to denote the quality of the neighborhood of any node based on the chosen attributes. Quality indicates the propensity of a node to be influenced to buy.

We use a heuristic-based approach to generate a time-efficient solution to maximize the cumulative quality measures of the spread according to the user requirement. The whole approach is user-centric and generates a customized seed set.

The main contribution can be summarized as.

1. Proposed work selects the seed set such that the spread within the customers with a high propensity to buy is maximized. The result section shows, that the quality in terms of NNW of the influenced set is better than that of the other works compared.
2. Proposed work allows the user (brand) to provide the target attributes and their significance level to generate adaptable solutions based on their requirements.

The proposed work is validated through exhaustive experimentations and the necessary comparison with the other notable works.

Problem Formulation

Example Scenario

Let us consider a scenario where a promotional campaign for a music concert wants to target young music lovers preferably with a middle- or high-income group. The objective here is to reach the maximum targeted customers rather than reaching the maximum number of customers. The targeted customer implies a young customer with a high inclination toward music and belongs to the middle- or high-income group. The solution generated without considering these essential attributes may end up with an influenced set with only a few nodes having the chance to respond to this campaign.

For example, in Fig. 1, a social network is depicted where each customer (node) is expressed through the attributes like age, income group, and a score to indicate the music interest of that specific person. Let us assume the campaigner sets the target for this marketing campaign as 'young music lovers preferably with a middle- or high-income group' through the requirement specification. As a result, N_1, N_4, N_5, and N_8 (colored in blue) are treated as target customers and the influence maximization will be done in such a manner that the maximum number of target customers is going to be influenced.

Problem Statement

Given a graph $G(V, E)$ representing a community network, a set of normalized attributes $P=\{P_{1,} P_2 \ldots P_x\}$ associated with each node and a set of weights $W=\{w_{1,} w_2, \ldots, w_x\}$ denoting the significance level of each attribute in a specific context. The graph is considered to be undirected i.e. each node can influence all the neighboring nodes with equal probability.

Objective: Maximize (NNW(S_0)).

where $NNW(S_0)=\sum^{n}_{i=1} INW(v_i)$ *for all* $v_i \in \sigma(S_0)$.

S_0 *denotes the seed set*

$\sigma(S_0)$ *denotes the influenced set i.e., set of one hop neighbors of* S_0,

n = *Number of influenced nodes i.e., cardinality of* $\sigma(S_0)$

and $INW(v_i)=f(P_{1i}\times w_1 + P_{2i}\times w_2 + \ldots + P_{xi}\times w_x)$ $\forall v_i \in V$

Subject to constraints:

- $n(S_0)=K$ *(where* $n(S_0)$ *is the number of nodes in Seed set* S_0*)*
- $\sum^{x}_{i=1} w_i=1$ *(where* w_i *is the weight of parameter* P_i*)*
- *The value of* P_i *is normalized between 0 and 10*

Proposed Work—Quality Seed Set Selection

The proposed work can be divided into three different phases discussed in the subsections. The first phase formalizes the user requirements. In this phase, a user provides the target attributes and desired values of those attributes for identifying the target customer. In the second phase, the attributes of each node in the graph are normalized based on the provided requirements. This phase also calculates the individual net worth i.e., the INW value of each node. Finally, the seed selection phase selects the seed set based on the cumulative INW of the influenced set.

Requirement Formalization

Most of the research on influence maximization provides a solution that is independent of the requirements of a user or the brand which starts the promotional campaign. However, in the real-life scenario, it is the user or brand who usually determines who should be the target customers for effective marketing. Promoting tickets for a classical music concert

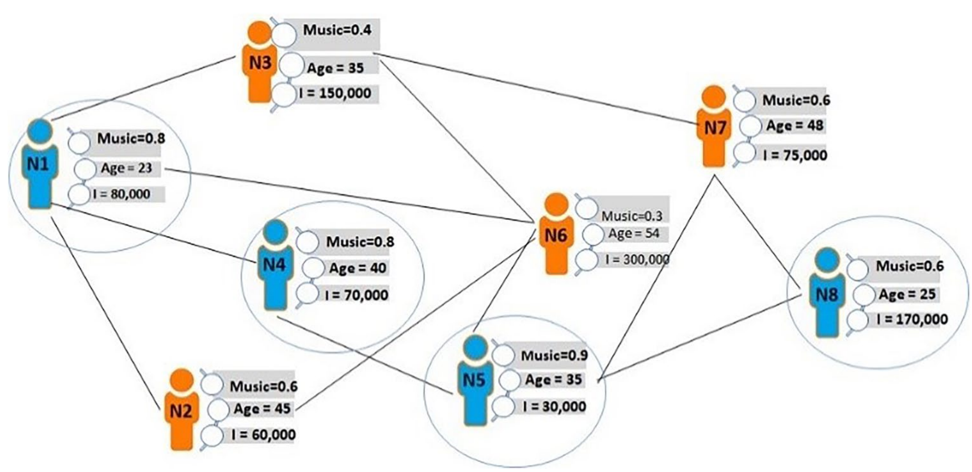

Fig. 1 Example social network with node attributes and highlighted target nodes

Table 1 Example user requirement

ReqID (R_i)	Significance level (L_i)	Attribute (A_i)	Condition (C_i)
1	2	Age	≥ 20 and ≤ 30
2	1	Music interest score	Score > 0.5
3	1	Income	$\geq 40{,}000$

and a rock music concert should be treated differently as the propensity of customers to buy tickets for both will not be the same. The target age group for people interested in rock music may be defined as lower than the target age group of classical music lovers by the user (concert organizers) thereby enabling them to tune the solution as per their requirements. This forms the basis of our motivation for this work. A requirement framework is proposed to capture the user's requirements as inspired by our earlier works in [19–21]. It is assumed that the nodes are available with their attribute values in the network. The framework allows a user to choose the attributes and the corresponding significance level for an attribute.

For example, let us consider a scenario of selling tickets for a music concert and choose the seed nodes that can influence people in the network to buy more tickets. Consider Fig. 1 for the example network. The objective is to maximize the sale of tickets. For example, the music company(user) sponsoring the event wants to target young music lovers in the age group 20–35 years with an income above Rs. 40,000. The fact of "music lover" can be adjudged by the number of interactions of the node on the topic "music". They want to give more importance to age than topic and income, the remaining two being of equal importance.

The schema of the requirement specification is given below:

$$\text{Requirement} = \{(L_1, A_1, C_1), (L_2, A_2, C_2) \ \ldots, \ (L_n, A_n, C_n)\},$$

where, L_i = Significance level: the significance level implies the weight of the corresponding attribute while computing the cumulative score for quantifying the "propensity to buy" for a customer.

A_i = Attribute: node attributes denote customer characteristics as in Fig. 1, they are age, music score, and income.

C_i = Condition: condition for selection of a node

A snapshot of sample requirement specifications for promotional ticket sales for a music concert is depicted in Table 1 for the example given in Fig. 1.

The proposed work provides a scope of user involvement. A user (brand/marketing team) can specify their formal requirements through the requirement framework as given in Table 1. The formal requirement includes the significance level of each attribute that affects the overall goodness measure. The significance level indicates the importance of the attributes set by the user (marketing brand). From the given requirement, it can be seen that the significance level of age is level 2. So, age is given more importance followed by hobbies and income at the same level. Accordingly, in the next section, we assign the weights w_i of these attributes.

Node Attribute Normalization (Calculating P_i)

In this section, we annotate the nodes with normalized values of the attributes. The range of values of the participating attributes is different from each other. For example, the attribute "age" may range from 10 to 85 whereas the attribute "income" may range in thousands. Some attributes may not have exact values but some labels like "high", "low" etc. Therefore, it is essential to bring all the attributes to a uniform scale before calculating the worth. The attributes are either scaled up or down or converted to a normalized range of 0–1. These values are normalized to avoid any unwanted bias toward a particular attribute or to prevent an attribute from influencing the result unfairly. The attributes are normalized on a scale of 0–10 based on the requirement provided by the user. The raw attribute values A_i can be normalized depending on the target condition associated with attribute A_i).

The normalized value of the Attribute attached to a node in the scale of 10 can be determined by following equation

$$P_i = 1 \times 10[\text{if condition} = \text{true as per Table1}]$$
$$= (1 - (|A_i - \text{Target}|)/\text{Range}(A_i)) \times 10\,[\text{if condition} = \text{false}]$$
$$\text{where } A_i - \text{Target} = \text{Deviation of } A_i \text{ from requirement target set by user}$$
$$\text{and Range}(A_i) = \text{Total range of values of } A_i.$$
$$= \text{Max}(A_i) - \text{Min}(A_i).$$

If the condition is true for a specific attribute of a node it implies that it satisfies the user's desired requirement and hence Pi is set to 1 which is the highest value possible.

If the condition is false, then the normalized value for the attribute A_i of a node is generated based on its deviation from the target set by the customer. The deviation is calculated as given below:

Target Condition	Node value	Deviation
> val or ≥ val	A_i	A_i -val
< val or ≤ val	A_i	A_i - val
Low_val < A_i < high_val	A_i < low_val	$\|A_i - \text{low\_val}\|$
	A_i > high_val	$\|A_i - \text{high\_val}\|$

For example, the normalized value of the attribute age of a node N_1 with age 33 can be determined as:

$$= (|A_i - \text{Target}|)/\text{Range}(A_i)$$

where Target = 30 as Ai > high_val (as per req. in Table 1) and range = Max(age) – Min(age) = 85 – 10 = 75.

Here the targeted range of values of age is given as (≥ 20 and ≤ 30). The maximum and minimum value for the attribute age is 85 years and 10 years, respectively in the graph.

Therefore, normalized age for N_1

$$= (1 - ((|33 - 30|))/(\mathrm{Max} - \mathrm{Min}))) \times 10,$$
$$= (1 - (3/(85 - 10))) \times 10 = 9.6.$$

In some cases, the condition provides an upper or lower boundary for the targeted attribute.

For example, if the condition specifying the targeted value of income is given as (≥ 40,000). The min and max value of income in the entire graph is 5000 and 100,000 then the income of node N_1 with attribute income = 45,000 can be normalized as

$$A_i = 45,000$$

Target = 40,000.

Range (A_i) = 100,000–5000 = 95,000.

Now as the condition provided is true here

$$\text{Normalized } N_1(\text{income}) = 1 \times 10 = 10$$

As per our assumption, the input graph is attributed. We considered each node as having attributes showing their percentage of interaction on a particular topic. The percentage of interaction on the targeted value of the attribute topic can be used to normalize the value of the topic field.

For example, the normalized value of the attribute topic of a node N_1 with 60% interactions on "music" can be calculated as

$$\text{Normalized } N_1(\text{topic}) = 0.6 \times 10 = 6$$

So, the normalized value of the attributes of a node N_1 {age = 33, income = 45,000, topic = 60%}

$$P_1 = \{9.6, 10, 6\}$$

Likewise, normalized attribute values P_i based on requirements are calculated for all nodes of the graph.

Weight Generation

The weight matrix carrying the significance of each attribute is generated according to the respective significance level provided in the requirement. The weights are chosen following a ratio of the relative significance level such that $\sum w_i = 1$. For determining the value of weight matrix W, we have used the following expression

$$\sum L_i \times n_i = 1$$

[where L_i = level number, n_i = Number of requirements in ith level]

For example, from the above requirement, we can understand that the significance of the attribute 'age' is higher than that of the attribute 'music interest' or 'income'. Accordingly, we can calculate the weight matrix in the ratio of 2:1:1 such that the sum of weights is 1. The calculated weight matrix comes to be

$$W = \{0.5, 0.25, 0.25\}.$$

The weights for all other node attributes that are not in requirements are set to zero i.e., $w_i = 0$ for an attribute A_i if A_i is not present in user requirements. It means that they do not have any effect on the worth of a node measured by INW. The weight matrix is uniformly used for all nodes of the entire graph.

Calculation of INW

The individual net worth (INW) of node N_1 can be calculated based on the weight matrix and the attribute values associated with the node using the formula presented in "Problem statement".

$$\mathrm{INW}(v_i) = f(P_{1i} \times w_1 + P_{2i} \times w_2 + \cdots + P_{xi} \times w_x)$$

where P_i is the normalized attribute set for node v_i.

and w_i is the generated weight matrix.

From "Node attribute normalization (calculating $\boldsymbol{P_i}$)", normalized attributes of N_1 denoted as

$$P1 = \{9.6, 10, 6\} \text{ and}$$

From "Weight generation", generated weight matrix is

$$W = \{0.5, 0.25, 0.25\}$$

Therefore, INW of node N_1 is calculated as

$$= 9.6 \times 0.5 + 10 \times 0.25 + 6 \times 0.25 = 8.8$$

Similarly, INW for all nodes of the graph can be calculated.

AWMIS Algorithm

In this section, we propose a heuristic-based Algorithm for Worth Maximization of Influenced Set (AWMIS) based on INW value to find the seed set. This can select the seed nodes keeping the cumulative net worth of the total spread into consideration. While choosing a seed node v_i, the Net neighborhood worth (NNW) value i.e., quality of the reach (one-hop neighbors) of the node is taken into consideration thus satisfying the user requirements. After each iteration, the algorithm selects the node with the highest NNW value as a seed. The INW of all the neighboring nodes of the seed is updated to 0 to reflect the fact that these nodes will not take part in the next NNW computations.

Input

i) Graph $G(V, E)$ represents an undirected community network. Any edge $e \in E$ represents a reflexive relationship between a pair of nodes $u, v \in V$. Degree of any node can be maximum d.
ii) Set of parameters $P_i = \{P_1, P_2 \ldots, P_x\}$ associated with each node $v_i \in V$, where each element of P is normalized between 0 and 10 and x = number of node attributes.
iii) Set of weights $W = \{w_1, w_2 \ldots w_x\}$ denoting significance of each node attribute.

Output

i) Seed set S_0 of cardinality K
ii) Set of Influenced nodes, $\sigma(S_0)$
iii) Cumulative INW values of the nodes in Influenced set

Procedure

Procedure-

```
Step 1   Initialize coverage σ(S0) = φ, S0=φ, CINW=0
//Initialization

Step 2 for each vertex v do
         INWv=0
         //Compute (INWv) of Each node
         for i= 1 to x do
             INWv + =Pi*Wi
         end for
       end for
Step 3 while |S0|! = K do
              for each vertex v do
                //Compute the NNW of each node
                   NNW v=INW v
                   for each neighbor u of v do
                      NNWv + =INWu

                   end for
              end for
Step 4        v= (argmax u∈(V-S0) (NNWu)
            //Select the node with maximum NNW
             S0=S0 U {v}
             CINW+= INWv
             INWv=0
Step 5       for each neighbor u of v and u ∈ (V-S0) do
                 // Reset INW of neighbors and seed
                 if INWu! =0
                        Σ(S0)= σ(S0) U {u}
                        CINW+= INWu
                        INWu=0
                 end if
             end for
       end while
```

Lemma 1 *For a given cardinality of seed; the algorithm must generate a seed such that the cumulative Individual Net Worth (INW) of the influenced nodes will be maximum.*

Proof For Graph $G(V, E)$ each node V_i has an associated individual net worth INW(V_i). The one-hop neighbors and Net Neighborhood Worth (NNW) of a node V_i is denoted respectively by $\sigma(V_i)$ and NNW(V_i). NNW(V_i) is calculated as the cumulative INW of (V_i U $\sigma(V_i)$). At each iteration, the algorithm chooses the node having maximum NNW as the seed node. Our objective is to justify that the cumulative INW of the influenced nodes (denoted by CINW) is maximum for the generated seed set of a given cardinality. CINW is the quality measure of the influenced set.

Let S_k and I_k be respectively the seed set and Influenced set after the kth iteration of the while loop. Thus $S_0 = V_0$ where V_0 is the node having the highest NNW in the graph, $I_0 = \sigma(V_0)$, and CINW = Cumulative INW of I_0.

For $k = 1 \ldots n-1$, let V_k be the node added to seed set S in the kth iteration.

$$\text{Thus, } I_k = I_{k-1} U \sigma\left(V_k\right) \text{and}$$
$$\text{CINW} = \text{Cumulative INW of } \left(I_{k-1} U\, \sigma\left(V_k\right)\right)$$
$$\text{Now let us assume at iteration } K+1 \text{ node } V_{k+1} \text{ has the maximum NNW value of NNW } \left(V_{k+1}\right). \tag{1}$$

But adding V_{k+1} as a seed does not result in a maximum CINW. So there exists another Seed V_j which maximizes the CINW. Now after adding V_j to the seed set the influenced set will be I_k U $\sigma(V_j)$, and the CINW will be Cumulative INW of (I_k U $\sigma(V_j)$). Now selecting V_j as a seed maximizes CINW so we can say

$$\text{Cumulative INW of } \left(I_k U\, \sigma\left(V_j\right)\right) > \text{Cumulative INW of} \left(I_k U \sigma\left(V_{k+1}\right)\right)$$
$$\text{Or Cumulative INW of } \left(\sigma\left(V_j\right)\right) > \text{Cumulative INW of } \left(\sigma\left(V_{k+1}\right)\right)$$
$$\left[\text{since for all nodes in } (I_k \cap \sigma(V_j) \text{ and } \left(I_k \cap \sigma\left(V_{k+1}\right)\right. \text{ the INW is reset as } 0\right]$$
$$\text{Or NNW } \left(V_j\right) > \text{ NNW } \left(V_{k+1}\right).$$

But this contradicts the earlier assumption made that V_{k+1} has the maximum NNW as given in (1). Hence, we can conclude that the assumption that adding V_{k+1} as a seed does not result in a maximum CINW is invalid. The CINW value gives a measure of the total quality of the influenced set.

Complexity Analysis of AWMIS

We analyze the complexity of the proposed algorithm (AWMIS) for seed selection in this section.

Step #	Process	Complexity
Step 1	Initialization	
Step 2	Calculation of INW of each node	O(N)
Step 3	Calculating NNW	O(N*d)
Step 4	Seed selection at each iteration	O(N)
Step 5	Updating INW	O(d)
Step 3–4-5	Selection of K seeds	O (K*(N+N*d+d))

The complexity of the proposed algorithm is O(K*N*d) [where k = number of seeds to be selected, N = number of nodes in the graph, and d is the maximum degree of a node].

Example Case Study

In this section, we explain our algorithm using an example graph having 10 nodes and work through the algorithm displaying the results of each iteration.

After each iteration, the node with the highest NNW is selected in the seed set and removed for subsequent calculations. One-hop neighbors added to the influenced set. Figure 2 depicts the example graph of 10 nodes and Table 2 details the P_i, w_i associated with each node, and the corresponding calculated INW value.

Table 2 summarizes the parameters as well as the calculated INW values associated with each node.

For seed (K = 3) selection we calculate the NNW of each node as the cumulative INW of the neighbors in 1 hop spread of each node (Table 3).

The node V_4, with maximum NNW, gets included in the seed set through step 4. The one-hop neighbors V_1, V_2, and V_7 are added to influenced set, and columns are highlighted.

Seed set $(S) = [V_4]$,

Spread $(A) = [V_1, V_2, V_7]$.

and cumulative INW = 25.4

The INW is updated to '0' for $[V_4, V_1, V_2, V_7]$.

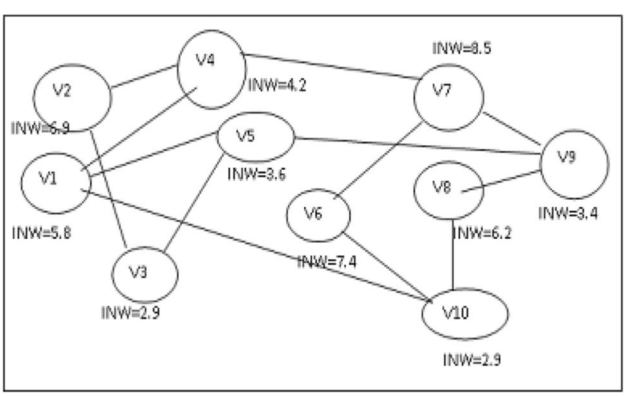

Fig. 2 Community Graph with node INW values

Table 2 INW for nodes

W_i	0.2	0.7	0.1	$INW = \sum(P_i \times W_i)$
Node	P_1	P_2	P_3	
V_1	2	7	5	5.8
V_2	1	9	4	6.9
V_3	4	2	7	2.9
V_4	2	5	3	4.2
V_5	3	4	2	3.6
V_6	5	8	8	7.4
V_7	9	9	4	8.5
V_8	4	7	5	6.2
V_9	6	3	1	3.4
V_{10}	8	1	6	2.9

Table 3 Example graph after iteration 1 showing selected seed node and influenced set

Iteration 1: Calculating the NNW of each node (Step 3)

Candidate for Seed	Set of one-hop neighbours (influenced set highlighted), seed node highlighted										NNW
	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	
V1	5.8			4.2	3.6					2.9	16.5
V2		6.9	2.9	4.2							14
V3		6.9	2.9		3.6						13.4
V4*	5.8	6.9		4.2			8.5				25.4
V5	5.8		2.9		3.6				3.4		15.7
V6						7.4	8.5			2.9	18.8
V7				4.2		7.4	8.5		3.4		23.5
V8								6.2	3.4	2.9	12.5
V9					3.6		8.5	6.2	3.4		21.7
V10	5.8					7.4		6.2		2.9	22.3

*The node with the highest NNW value is highlighted for reference

Iteration 2: In step 3 NNW of each node is recalculated.

In step 4 the node with maximum NNW is V_{10}, added to the seed set.

Seed set $(S) = [V_4, V_{10}]$,

Spread $(A) = [V_1, V_2, V_7, V_6, V_8]$.

and cumulative INW = 41.9

INW is updated to 0 for $[V_{10}, V_6, V_8]$ (Tables 4 and 5).

Iteration 2: In step 3 NNW of each node is recalculated.

In step 4 the node with maximum NNW is V_5.

Seed set $(S) = [V_4, V_{10}, V_5]$,

Spread $(A) = [V_1, V_2, V_7, V_6, V_8, V_3, V_9]$.

and cumulative INW is 51.8

INW is updated to 0 for $[V_3, V_9]$.

We can stop at this point as the number of nodes in seed set $(K) = 3$ is as per our requirement. The number of influenced nodes is 7. In this small example graph, all 10 nodes can be reached using $K = 3$ seeds. The cumulative INW of all nodes i.e., the total quality measured by CINW of selection is 51.8 which is the maximum in this case.

Table 4 Example graph after iteration 2 showing selected seed node and influenced set

Candidate for Seed	Influenced Set with one-hop neighbours (highlighted)										NNW
	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	
V1	0			0	3.6					2.9	6.5
V2		0	2.9	0							2.9
V3		0	2.9		3.6						6.5
V5	0		2.9	0	3.6				3.4		9.9
V6						7.4	0			2.9	10.3
V7				0		7.4	0		3.4		10.8
V8								6.2	3.4	2.9	12.5
V9					3.6		0	6.2	3.4		13.2
V10*	0					7.4		6.2		2.9	16.5

*The node with the highest NNW value is highlighted for reference. The highlighted columns indicate the influenced set

Table 5 Example graph after iteration 3 showing selected seed node and influenced set

Iteration 3: In step 3 NNW of each node is recalculated

	Set of one-hop neighbors										NNW
	V1	**V2**	**V3**	**V4**	**V5**	**V6**	**V7**	**V8**	**V9**	**V10**	
V1	0			0	3.6					0	3.6
V2		0	2.9	0							2.9
V3		0	2.9		3.6						6.5
V5*	0		2.9	0	3.6				3.4		9.9
V6						0	0			0	
V7				0		0	0		3.4		3.4
V8								0	3.4	0	3.4
V9					3.6		0	0	3.4		7

Simulation and Experimental Results

In this section we set some experimental set-up and evaluate the effectiveness of the proposed method in comparison with the following algorithms:

i. CELF Algorithm—Proposed in [7]
ii. Degree Discount Algorithm—Proposed in [4]
iii. ORIE Algorithm—Proposed in [12]

The objective of the proposed work differs from the compared works. The algorithms (i–iii) for comparison aim to maximize the number of nodes in the influenced set. However, the scope of these works does not ensure the quality (propensity to buy) of the influenced set. Unlike these algorithms, the scope of the proposed work considers different attributes of the nodes and aims to ensure the quality of the solution. Moreover, it provides the user scope of getting an adaptable solution based on requirements.

Data Source

We have applied our algorithm to the datasets mentioned in Table 6.

Table 6 Summary of datasets for comparative analysis

Dataset	Nodes	Edges
ego-Facebook (FB)—Social circles from Facebook [23]	4039	88,234
CA-Hept (CAH)—Collaboration network of Arxiv High Energy Physics Theory [24]	9877	25,998
Net-Hept (NH)—High Energy Physics Theory [25]	15,233	58,891

Experimental Setup

Table 6 lists the datasets used for comparison. We have considered the datasets with four attributes having values ranging from 0 to 10 associated with each node. These values are normalized values of the chosen attributes. The attributes of nodes have been simulated following a random distribution pattern. All the networks are undirected with unweighted edges.

Results and Discussion

It is to be noted that the proposed work aims to maximize the net worth of the influenced set thereby maximizing the overall buying propensity of the target customers. The proposed algorithm considers the attributes of the nodes. This is the unique contribution of our work. In contrast, the algorithms presented in [4, 7] consider the nodes of the network as homogeneous and aim to maximize the number of nodes in the influenced set while the work in [12] considers interaction rates. Considering any other feature or ensuring the quality of the influenced set is beyond their scope of work.

In the proposed work, any combination of features can be considered to generate a customized solution. Moreover, the solutions in [4, 7, 12] are independent of business requirements whereas our solution approach considers the significance level or weights of node characteristics that can be tuned by the user. We have compared the performance of our work based on the following aspects.

1. Quality-the quality of the generated influenced set measured in terms of the cumulative net worth (NNW) of influenced set.
2. Adaptability—can be measured as a percentage of change in seed set with varying user requirements. We have used different weight matrices to specify different

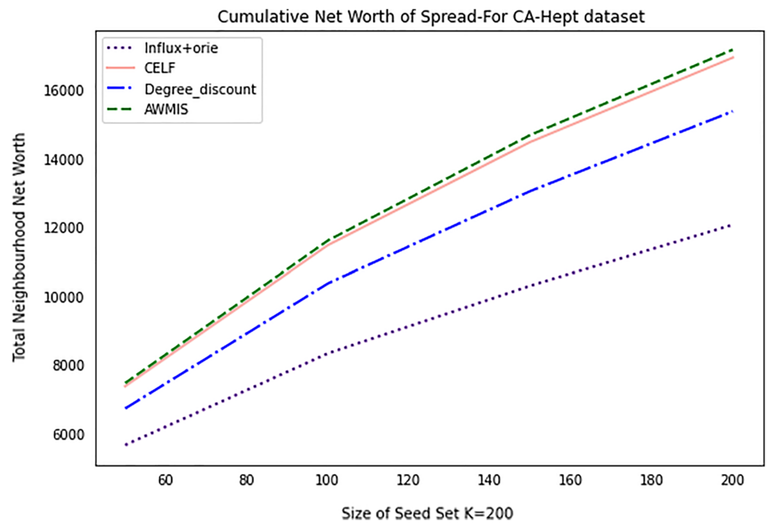

Fig. 3 Cumulative net neighborhood worth of the seed set on CAH dataset for $K=200$

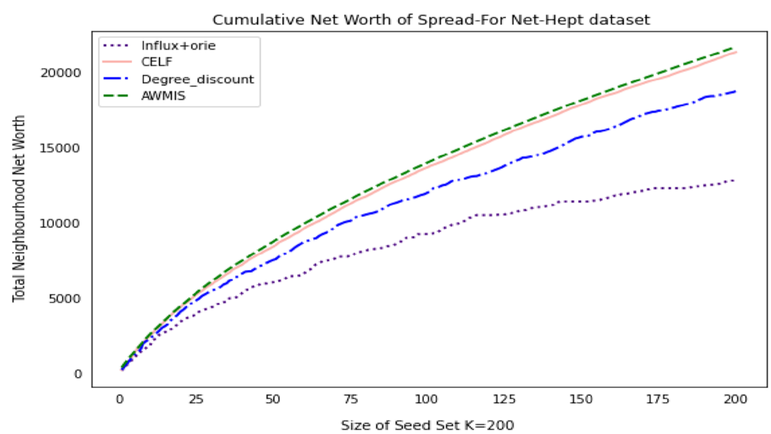

Fig. 4 Cumulative net neighborhood worth of the seed set on NH dataset for $K=200$

user requirements. The generated solution can be tuned by the user through requirements.

3. Scalability—measured in terms of comparative computation time of the approach with increasing size of both the seed nodes (K) and overall size of the graph (N).

Performance Concerning the Cumulative Net Worth of the Influenced Set

This section compares the proposed algorithm with other notable works based on the cumulative net worth or quality of the influenced set.

Figure 3, 4, and 5 shows the incremental cumulative net neighborhood worth with an increasing number of seeds. It depicts that the NNW of the seed set generated by AWMIS is higher than all the other algorithms considered. As per the aim of this work it shows that AWMIS generates the seed set keeping the quality of the neighborhood (or net worth of the influenced set) in focus.

Figure 3 and 4 compare the cumulative net neighborhood worth of the algorithms for fixed seed set size ($K=200$) on the datasets. It is to be noted that the cumulative net worth of the influenced set generated by the CELF algorithm is less than but close to AWMIS. Although it is close its computation time is much higher than AWMIS as depicted in "Performance with different sizes of seed set (K) considering N (graph size) as parameter".

Figure 5 shows the cumulative NNW is not changing after seed size = 10. The FB dataset used in Fig. 5 is a dense graph with 4039 nodes and 88,000 edges. The experiment shows a seed set of size $K=10$ is enough to reach almost all the nodes of the graph in its neighborhood. However, the cumulative net neighborhood worth for AWMIS is more than all other algorithms for all sizes of the graph (N) as well as for different values of seed.

Performance with Different Weight Values (W_i) Based on Varying User Requirements

Our approach is adaptable in the way it can accommodate user requirements in terms of assigning significance or weights to node attributes based on user requirements. In this section, we have analyzed the performance of our

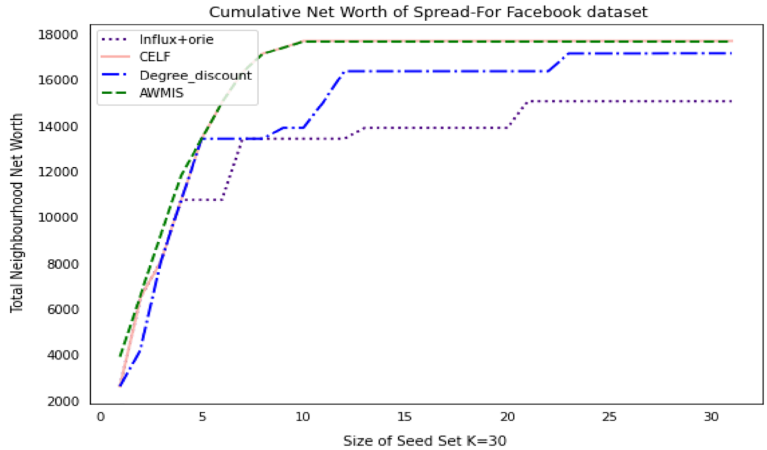

Fig. 5 Cumulative net neighborhood worth of the seed set on FB dataset for $K=30$

Table 7 Variation of seed set of AWMIS on CAH dataset for varying attribute weights with seed size (30)

Attribute weight	CELF	DD	ORIE	AWMIS
$W_1=[0, 0, 0.33, 0.67]$	Fixed seed set of 30 nodes	Fixed Seed set of 30 nodes	Fixed Seed set of 30 nodes	Benchmark
$W_2=[0, 0.5, 0.5, 0]$				20%
$W_3=[0.33, 0.66, 0, 0]$				13.33%
$W_4=[0.5, 0, 0, 0.5]$				13.33%

proposed work by varying the set of attribute weights on a dataset and also compared it with other works (given in "Simulation and Experimental Results").

For brevity, Table 7 presents a summary result depicting that the other algorithms generate a fixed seed set while in comparison there is a variation in the seed set generated by AWMIS with varying attribute weights. Appendix-1 lists the detailed information on the set of seed nodes generated for varying weight values. As an example, the seed set size considered for comparison is $K=30$.

The seed set for weight set W1 that has been used in the previous Sect. 6.3.1 is taken as a benchmark. The difference in the seed set (% of nodes that changed from the benchmark) is depicted for varying weight values establishing the fact that our approach is adaptable to user requirements.

Table 7 depicts the fact that the solution provided is adaptable and generates seed sets according to the varying requirements.

One of the main aims of the proposed work is to generate a seed set that will maximize the quality of the influenced set-in terms of cumulative net worth. Figure 6 compares the performance of AWMIS in terms of the cumulative net worth of influenced set for different weight values corresponding to different user requirements for $K=200$.

The above Fig. 6 shows that the cumulative net worth of the influenced set generated by AWMIS is higher than the other compared approaches for different requirement weights. This justifies our claim to ensure maximum quality of influenced set per user requirements. Even though the performance of CELF is close to AWMIS, it is to be noted that the seed set is unchanged with changing weight values whereas our solution is responsive to user requirements. Further, it is shown in the next section that the computation time for CELF is extremely high compared to AWMIS for all sizes of graph and seed nodes. Our solution is adaptable as well as scalable.

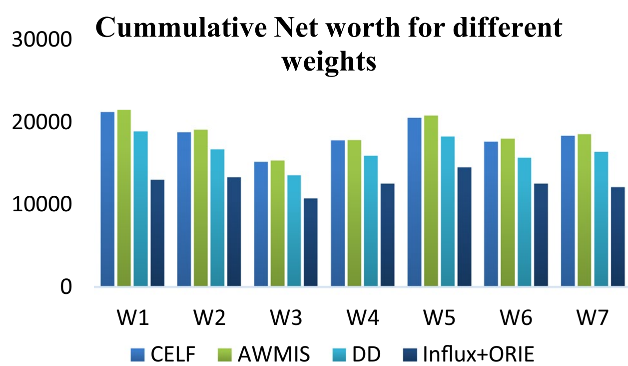

Fig. 6 Cumulative net neighborhood worth of the seed set for varying user requirements on CAH dataset for $K=200$

Performance with Different Sizes of the Graph (N) and Fixed Seed Set Size (K)—Computation Time

It can be observed from Table 8 that the absolute computation time of the proposed algorithm (AWMIS) does not vary largely with the size of the network. In terms of the relative

Table 8 Summary of results of AWMIS on different datasets for varying seed size (K)

Dataset size		$K=50$			$K=100$			$K=150$			$K=200$		
Nodes	Edges	Time	reach %	NNW	Time	reach%	NNW	Time	reach %	NNW	Time	reach %	NNW
4039	88,234	1.71	99.8	18,054	3.02	99.8	18,054	4.6	99.8	18,054	6.2	99.8	18,054
9877	25,998	1.16	15.9	7464	2.33	24.8	11,605	3.54	31.39	14,683	4.9	36.8	17,168
15,233	58,891	1.5	10.1	8734	3.02	15.6	13,985	4.52	19.79	18,117	5.9	23.3	21,657

increase in dataset sizes, we can see that the variation decreases from 46% for $K=50$ to around 25% for $K=200$. This variation is for the increase in graph size from 4000+ nodes to almost 4 times to 15,000+ nodes. This shows that the proposed algorithm is scalable in terms of network size as well as seed size. It is also seen that the computation time for CA-Hept ($N=9877$) is less than the Facebook ($N=4039$) dataset for all values of the seed set, even though the size of the graph is almost double. This may be attributed to the number of edges. FB dataset has almost 3.5 times the number of edges compared to CA-Hept. Likewise, NetHept despite being almost 4 times in size (number of nodes) takes less time due to the number of edges being less than the FB dataset.

Figures 7 and 8 depict comparative performance of algorithms for fixed seed set size (K) but varying sizes of networks from 4000+ nodes in Facebook and 15,000+ in NetHept. The computation time is measured in seconds (s).

Referring to Figs. 7 and 8, it can be observed that AWMIS performs better than the CELF algorithm and takes comparable time with respect to the degree discount & ORIE algorithms. We can also observe the computation time of ORIE is higher for the FB dataset. It depicts the fact that ORIE takes more time for datasets with a higher number of edges. In comparison to that, the computation time of the AWMIS algorithm does not vary highly with an increasing number of edges. Even though the degree discount and ORIE algorithm is providing a slightly better runtime but do not ensure the quality of the influenced set. AWMIS adapts any combination of features and generates a customized solution whereas the other State-of-the-art algorithms, used for comparison provide a fixed solution ignoring the context.

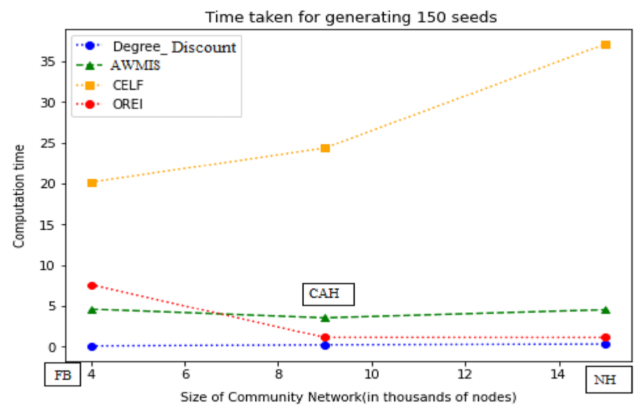

Fig. 7 Comparative performance of algorithms on different sizes of networks for $K=150$

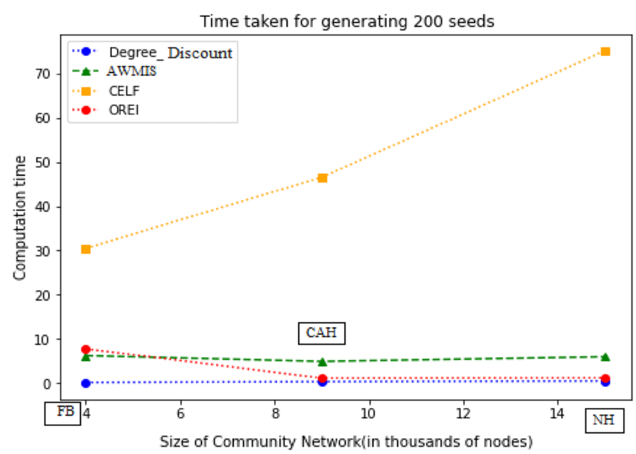

Fig. 8 Comparative performance of algorithms on different sizes of networks for $K=200$

Performance with Different Sizes of Seed Set (*K*) Considering *N* (Graph Size) as Parameter

In this section, we perform experiments to show the performance of AWMIS in terms of computation time vis-à-vis the other algorithms (CELF, DD, ORIE). Each figure (Figs. 9, 10, and 11) depicts the comparative performance of all algorithms for one particular dataset and the computation time w.r.to varying seed set size (K).

Referring to the figures it can be observed that AWMIS outperforms CELF for all values of seed set sizes $K=50$, 100, 150, 200.

It can be observed that AWMIS delivers the same performance as Degree discount (DD) and ORIE algorithms for seed set sizes $K=50$ and 100 but for higher seed set values of $K=150$, 200 it takes marginally more computation time.

Referring to Table 8, it is seen that the computation time increases almost linearly with an increase in seed set sizes. The results of the simulation establish that the proposed algorithm is comparable in time complexity with the state-of-the-art approaches while enhancing further toward a user-centric unique/customized solution to ensure the quality of the influenced set.

Conclusion

Our work focuses on selecting a quality seed set that has the potential to influence maximum targeted customers in the network having the propensity to buy. When compared with the other state-of-the-art approaches the proposed work is closer to the ground reality and represents the real-life scenario much better.

The proposed work provides a user-centric customized solution. It can be tuned to adapt any combination of attributes with their respective significance levels for providing a user-specific, adaptable and scalable solution. Rather than generating a larger spread, it focuses on generating a high-quality spread in terms of the net worth of the customer base. Unlike the compared approaches which always generated a fixed seed set irrespective of the context, the proposed approach generates different seed sets in different scenarios based on the significance of the attributes chosen by the user or brand. Moreover, this uses a data-driven approach where the attributes associated with nodes are given priority over the structural information of the nodes.

Experiments show that the proposed work selects the seed set with maximum quality measured in terms of cumulative worth and is comparable in time efficiency with the state of art influence maximization algorithms. It is adaptable to user requirements and generates different seed set and influenced set based on differing requirements and provides scalability in terms of network size as substantiated through the results.

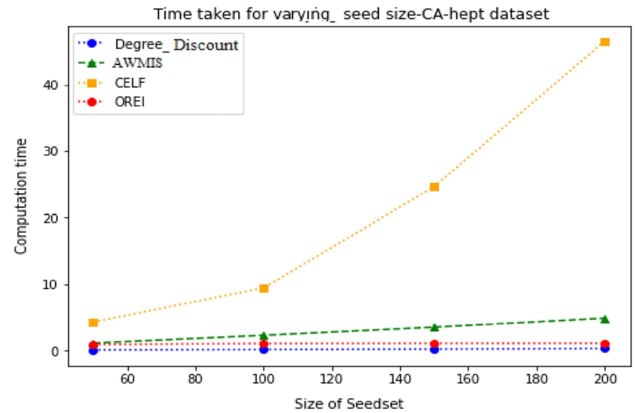

Fig. 11 Comparative performance of algorithms on CAH dataset for varying K

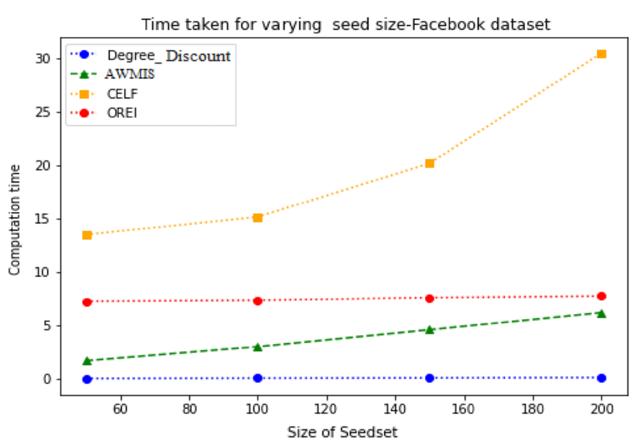

Fig. 9 Comparative performance of algorithms on FB dataset for varying K

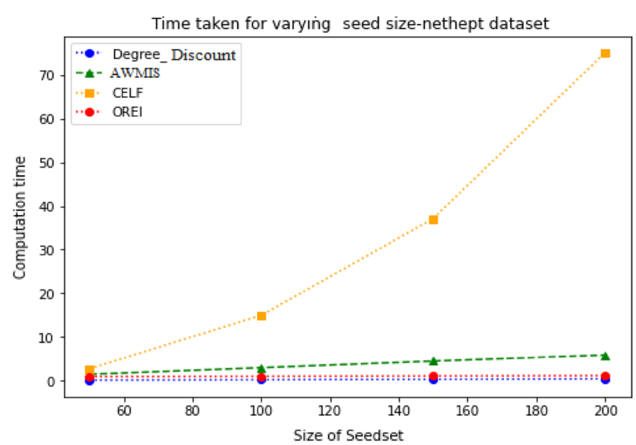

Fig. 10 Comparative performance of algorithms on NH dataset for varying K

Appendix 1: Comparison of seed node set for varying weight values

W (weights for diff attributes)	Algorithm	Seed set (Node numbers)	Diff. in seeds	Diff %
$W_1 = (0,0,0.33,0.67)$	AWMIS (proposed approach)	1441, 3423, 6142, 6948, 7859, 13648, 14017, 14726, 16164, 19615, 23282, 23420, 24059, 26122, 28950, 29928, 30160, 30744, 33512, 34787, 36103, 39085, 44262, 53601, 54465, 63113, 63697, 65168, 65553, 68111	Benchmark for comparison	
$W_2 = (0,0.5,0.5,0)$		**97,** 1441, **4436**, 6142, 6948, 7859, 13648, 14726, **17289, 17370**, 19615, 23282, 23420, 26122, 28950, 29928, 30744, 33512, 34787, 36103, 39085, 44262, 54465, **60926**, 63113, 63697, 65168, 65553, **66349**, 68111	6	20%
$W_3 = (0.33,0.67,0,0)$		1441, 3423, **4436**, 6142, 6948, **7233**, 7859, 13648, 14017, 14726, 16164, **17370**, **17793**, 19615, 23282, 23420, 26122, 28950, 30160, 30744, 33512, 36103, 39085, 44262, 53601, 54465, 63113, 63697, 65168, 68111	4	13.33%
$W_4 = (0.5,0,0, 0.5)$		1441, 3423, **4436**, 6142, 6948, 7859, 13648, 14017, 14726, 16164, **17370, 17793,** 19615, 23282, 23420, 26122, 28950, 30160, 30744, 33512, 36103, 39085, 44262, **48973**, 53601, 54465, 63113, 63697, 65168, 68111	4	13.33%
$W_1 = (0,0,0.33,0.67)$	CELF	97, 1441, 3423, 4436, 6142, 6948, 7859, 14017, 14726, 16164, 17370, 17793, 19615, 20394, 23282, 23420, 28950, 29715, 30744, 33512, 36103, 39085, 43684, 44262, 54465, 62227, 63113, 63697, 65168, 68111	No change	
W_2, W_3, W_4		Same seed set. No change due to change in W		
W_1, W_2, W_3, W_4	DD	Same seed set. No change due to change in W	No change	
W_1, W_2, W_3, W_4	ORIE	Same seed set. No change due to change in W	No change	

*Nodes in bold are new nodes selected in the seed set replacing the old ones based on varying W values

Funding This study was not funded.

Availability of Data and Material Public datasets are used. The sources of the datasets are mentioned in the dataset Sect. 6.1 in the manuscript.

Code Availability Custom code developed using python for experimental validation.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Alduaiji N, Datta A, Li J. Influence propagation model for clique-based community detection in social networks, digital object identifier. IEEE Trans Comput Soc Syst. 2018;5(2):563–75. https://doi.org/10.1109/TCSS.2018.2831694.
2. Alp ZZ, Şule Gündüz Ö. Influence factorization for identifying authorities in Twitter. Knowl Based Syst. 2019;163:944–54. https://doi.org/10.1016/j.knosys.2018.10.020.
3. Chen S, Fan J, Li G, Feng J, Tan K, Tang J. Online topic-aware influence maximization. In: Proceedings of the VLDB endowment, ACM, vol. 8, no. 6, pp. 666–677, 2015. DOI:https://doi.org/10.14778/2735703.2735706.
4. Chen W, Wang Y, Yang S. Efficient influence maximization in social networks. In: Proceedings of the 15th ACM SIGKDD, KDD'09, pp. 199–208. ACM, New York, NY. https://doi.org/10.1145/1557019.1557047.
5. Cui L, Hu H, Yu S, Yan Q, Ming Z, Wen Z, Lu N. DDSE: a novel evolutionary algorithm based on degree-descending search strategy for influence maximization in social networks. J Netw Comput Appl. 2018;103:119–30. https://doi.org/10.1016/j.jnca.2017.12.003.
6. Goyal A, Lu W, Lakshmanan LV (2011) Celf++: optimizing the greedy algorithm for influence maximization in social networks. In: Proceedings of the 20th international conference companion on World wide web, ACM, 2011, https://doi.org/10.1145/1963192.1963217.
7. Leskovec J, Krause A, Guestrin CE, Faloutsos C, VanBriesen J, Glance NS. Cost-effective outbreak detection in networks. In: KDD '07, Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, p 420–429, August 2007. https://doi.org/10.1145/1281192.
8. Kempe D, Kleinberg J, Tordas E. Maximizing the spread of influence through social network. Proc ACM SIG Knowl Discov Datamin. 2003. https://doi.org/10.1145/956750.956769.
9. Kundu G, Choudhury S. A discrete genetic learning enabled PSO for targeted positive influence maximization in consumer review networks. Innov Syst Softw Eng. 2021;17:247–59. https://doi.org/10.1007/s11334-021-00396-5.
10. Rossi M-EG, Shi B, Tziortziotis N, Malliaros FD, Giatsidis C, Vazirgiannis M. MATI: an efficient algorithm for influence maximization in social networks. PLoS One. 2018;13(11):e0206318. https://doi.org/10.1371/journal.pone.0206318.
11. Singh SS, Kumar A, Singh K, Biswas B. C2IM: community based context-aware influence maximization in social networks. Physica A Stat Mech Appl. 2019;514:796–818. https://doi.org/10.1016/j.physa.2018.09.142.
12. Sumith N, Annappa B, Bhattacharya S. A holistic approach to influence maximization. Hybrid intelligence for social networks. Springer, 2017. https://doi.org/10.1016/j.asoc.2017.12.025

13. Sumith N, Annappa B, Bhattacharya S. Influence maximization in large social networks: heuristics, models and parameters. Future Gener Comput Syst. 2018;89:777–90. https://doi.org/10.1016/j.future.2018.07.015.
14. Xu X, Zhang Y, Hu Q, Xing C. A balanced method for budgeted influence maximization. In: SEKE (2015), pp. 250–255, 2015. https://doi.org/10.18293/SEKE2015-157.
15. Guoliang Li, Shuo Chen, Jianhua Feng, Kian-lee Tan, and Wensyan Li, (2014), "Efficient location-aware influence maximization", Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD '14). ACM, NY, USA, pp. 87–98. 2014, DOI:https://doi.org/10.1145/2588555.2588561
16. Zareie A, Sheikhahmadi A, Khamforoosh K. Influence maximization in social networks based on TOPSIS. Expert Syst Appl. 2018;108(15):96–107. https://doi.org/10.1016/j.eswa.2018.05.001.
17. Li L, Liu Y, Zhou Q, Yang W, Yuan J. Targeted influence maximization under a multifactor-based information propagation model. Inf Sci. 2020;519:124–40. https://doi.org/10.1016/j.ins.2020.01.040.
18. Jiuxin C, Tao Z, Dan D, Shuai X, Ziqing Z, Ma Z, Bo L. Attribute-based influence maximization in social networks. In: Web information systems engineering—WISE 2016, vol. 10041, pp. 3–18, 2016. https://doi.org/10.1007/978-3-319-48740-3_1.
19. Chanda J, Kanjilal A, Sengupta S, Bhattacharya S. Traceability of requirements and consistency verification of UML use case, activity and class diagram: a formal approach. In: Proceedings of international conference on methods and models in computer science (ICM2CS), pp. 1–4, 2009. https://doi.org/10.1109/ICM2CS.2009.5397941.
20. Sengupta S, Kanjilal A, Bhattacharya S (2008) Requirement traceability in software development process: an empirical approach. In: 2008 The 19th IEEE/IFIP international symposium on rapid system prototyping, pp. 105–111, 2008. https://doi.org/10.1109/RSP.2008.14.
21. Kanjilal A, Sengupta S, Bhattacharya S. Analysis of complexity of requirements: a metrics based approach. In: Proceedings of the 2nd India software engineering conference (ISEC'09), pp. 131–132, 2009, https://doi.org/10.1145/1506216. 1506243.
22. Domingos P, Richardson M. Mining the network value of customers. In: Proceedings of the seventh ACM international conference on knowledge discovery and data mining (SIGKDD), pp. 57–66, 2001, https://doi.org/10.1145/502512.502525
23. Social circles: Facebook. https://snap.stanford.edu/data/ego-Facebook.html. Accessed 15 May 2021.
24. High Energy Physics—theory collaboration network, https://snap.stanford.edu/data/ca-HepTh.html. Accessed 15 May 2021.
25. High Energy Physics Theory—NetHept dataset. http://www.cs.cornell.edu/projects/kddcup/datasets.html. Accessed 10 May 2021.