Voting-based Opinion Maximization

Arkaprava Saha
NTU, Singapore
saha0003@e.ntu.edu.sg

Xiangyu Ke
ZJU, China
xiangyu.ke@zju.edu.cn

Arijit Khan
AAU, Denmark
arijitk@cs.aau.dk

Laks V.S. Lakshmanan
UBC, Canada
laks@cs.ubc.ca

Abstract*—We investigate the novel problem of *voting-based opinion maximization* in a social network: *Find a given number of seed nodes for a target campaigner, in the presence of other competing campaigns, so as to maximize a voting-based score for the target campaigner at a given time horizon.

The bulk of the influence maximization literature assumes that social network users can switch between only two discrete states, inactive and active, and the choice to switch is frozen upon one-time activation. In reality, even when having a preferred opinion, a user may not completely despise the other opinions, and the preference level may *vary over time* due to social influence. To this end, we employ models rooted in *opinion formation and diffusion*, and use several *voting-based scores* to determine a user's vote for each of the multiple campaigners *at a given time horizon*.

Our problem is NP-hard and non-submodular for various scores. We design greedy seed selection algorithms with quality guarantees for our scoring functions via sandwich approximation. To improve the *efficiency*, we develop random walk and sketch-based opinion computation, with *quality guarantees*. Empirical results validate our effectiveness, efficiency, and scalability.

***Index Terms*—social network, opinion maximization, voting**

I. Introduction

Social influence studies have attracted extensive attention in the data management research community [1], [2], [3], [4], [5], [6], [7], [8]. The classic influence maximization (IM) problem [9], [10] identifies the top-k seed users in a social network to maximize the expected number of influenced users in the network, starting from those seed nodes and following an influence diffusion model (e.g., independent cascade (IC) and linear threshold (LT) [9]). Several works also focus on competitive influence maximization [11], [12], [13], [14], [15], [16], [17], [18] which aims to find the seed set that maximizes the influence spread for a particular campaigner relative to the others or maximally blocks the diffusion of a competitor.

However, prior works on IM have two major limitations in modelling real-world opinion formation and spreading. First, they consider maximizing the expected number of users adopting a specific campaign, assuming that the reaction of each user to the campaign is *binary* (adopt or not). In reality, a user may not be completely opposed to the competing opinions, although she could have a preference for one opinion, where the degree of preference could vary among users. This scenario can be accurately modelled by allowing the opinion of a user for each campaign to be a real number in $[0, 1]$. Second, in the IC and LT models, a user's choice is frozen upon one-time activation – not permitting to switch opinions later. While this is realistic when purchasing one of the many competing products due to the user's limited budget, it is insufficient for modelling *opinion formation and manipulation over time*, e.g., in scenarios like paid movie services, elections, social issues, where a user's opinion is highly likely to change over time.

Due to the above shortcomings, we deviate from the classic influence diffusion (e.g., IC and LT models) and investigate the problem of *opinion maximization* by employing models rooted in opinion formation and diffusion, e.g., DeGroot [19] and Friedkin-Johnsen (FJ) [20], [21]. In these settings, each user in a network has a *real-valued* opinion about each campaign at *every* timestamp. Moreover, for each campaign, the *opinions of the users evolve* over discrete timestamps according to an opinion diffusion model such as DeGroot or FJ (defined in § II-A). Given a target campaign and a time horizon (a future timestamp t), our problem is to select a seed set of size k for the target campaigner, so that the target campaigner's odds of being the *winner* at the time horizon t are as high as possible.

Since opinion values are non-binary, we require more sophisticated winning criteria than the *expected influence spread* employed in classic IM [9]. Voting offers a well-understood mechanism for determining winners in an election among campaigners by considering the preferences of users ("voters") in a principled manner. We investigate voting-based scores [22], [23], [24] such as aggregated opinion values of all users about a campaigner (*cumulative*), rank of the target campaigner relative to others for all users (*plurality*), or the number of campaigners against whom the target campaigner wins in one-on-one competitions (*Copeland*). These are natural choices based on voting theory when users have non-binary opinion values towards multiple competitors. Existing works on finding the top-k seeds for opinion maximization [25], [26] *are restricted to a single campaigner and consider neither a given finite time horizon*[1]*, nor voting-based scores with multiple competing campaigners*[2]*. To the best of our knowledge, voting-based opinion maximization in the presence of multiple competing campaigns is a novel problem.*

Applications. Our problem and solutions can be effective where users vote and the winner among multiple candidates is decided based on the election outcome. Examples include the presidential election, voting in the parliament, a plebiscite

Xiangyu Ke is the corresponding author. Arijit Khan acknowledges support from the Novo Nordisk Foundation grant NNF22OC0072415. Laks V.S. Lakshmanan's research was supported by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC).

[1]In practice, the voting is held at a specific time horizon, instead of waiting for the diffusion to reach the Nash equilibrium as is done in [25].

[2]Only our cumulative score is similar to theirs due to its aggregate nature.

or a referendum (e.g., the referendum on the independence of Scotland) [27], [28], etc. We conduct a real-world case study about the ACM general election 2022 (§VIII-B). Our case study shows that the election result might have reversed after introducing only 100 optimal seed users. Our solution selects influential seeds based on (1) their common research interests with respect to the target candidate and (2) the initial preferences of the users in various research domains. Moreover, our approach smartly focuses on switching the preferences of more neutral users. These demonstrate the usefulness of our problem and the effectiveness of our solution.

Challenges and Our Contributions. With multiple competing campaigns in a network, we formulate and study a *novel problem in opinion maximization*: Find the top-k seed nodes for a target campaign that maximize a voting-based winning criterion for the target at a given time horizon (§ II-C). Our contributions are as follows.

• **Opinion Maximization and Voting Scores:** To the best of our knowledge, opinion manipulation by introducing seed nodes has not been investigated before, except, e.g., [2], [25], [26], [29], [30]. However, apart from [25], [26], prior works do not consider sophisticated DeGroot/FJ opinion models. Also, opinion maximization at a finite time horizon with multiple campaigners has not been explored even in [25], [26]. One of our novel contributions is bridging two different paradigms: **(1)** seed selection for opinion formation and diffusion till *a given finite time horizon*, and **(2)** voting-based winning criteria (e.g., plurality, Copeland) with *multiple campaigners*.

• **Sandwich Approximation:** Our problem is **NP**-hard (§ III-A) and non-submodular (§ III-B) under various winning criteria[3]. Despite these, we design bound functions for all our non-submodular scores to derive accuracy guarantees for the greedy algorithm via *sandwich approximation* [31] (§ IV).

• **Random Walks:** Computing opinion values at the time horizon via DeGroot/ FJ requires iterative matrix-vector multiplications, which is expensive. *To improve the efficiency, we propose random walk and sketching-based computations with approximation guarantees.* Random walks have been used earlier to improve the efficiency of matrix multiplication and PageRank computation [32], [33]. Our novelty is using random walks to find the k seed nodes maximizing a voting-based score by approximating the opinion values via the walks in k iterations. Also, we provide novel bounds on the number of walks required for each voting-based scoring function (§ V).

• **Sketches:** While sketches have been used in classic IM [3], [7], [34], *ours is the first work that uses sketches for opinion computation.* We adapt sketches for opinion diffusion models and voting-based scores, and derive non-trivial accuracy guarantees (§ VI). Moreover, our sketches are simpler and less memory-consuming than RR-sets-based sketches [3], [7].

Our thorough experimental evaluation and case study over five real-world social network datasets demonstrates the effectiveness, efficiency, and scalability of our solutions, over several baselines (§ VIII). Related work is discussed in § VII, while in § IX we conclude and discuss future work. Proofs of theoretical results that are omitted for brevity, as well as additional details, can be found in our extended version [35].

[3]The proofs of these results in [25] cannot be extended trivially even to our basic model of the cumulative score for any finite time horizon, warranting new techniques.

II. Preliminaries

A social network is modeled as a (directed) graph $\mathcal{G} = (V, E)$, where V is the set of n nodes (users) and $E \subseteq V \times V$ is the set of m edges (relations). We denote matrices with upper-case letters and use lower-case ones for their entries. We denote an $n \times n$ diagonal matrix by $diag(d_1, d_2, ..., d_n)$, and the $n \times n$ identity matrix by I_n. A matrix $A = (a_{ij})$ is *column-stochastic* if $a_{ij} \geq 0,\ \forall i, j$, and $\sum_{i=1}^{n} a_{ij} = 1,\ \forall j$.

Different news, campaigns, or opinions can propagate concurrently in the network, leading to competitions [11], [12], [16]. They can be information about similar products of different brands, multiple politicians campaigning for the same position, or different attitudes towards a topic, e.g., for or against gun control. We call them *candidates* and assume that there are $r > 1$ candidates: $C = \{c_1, c_2, ..., c_r\}$. All users' opinions (in the interval $[0, 1]$) on all candidates are represented by an opinion matrix $B \in [0, 1]^{r \times n}$. $B_q \in [0, 1]^{1 \times n}$ is the q^{th} row of B (denoting all users' opinions on candidate c_q), and b_{qi} is its i^{th} entry (opinion of user i on candidate c_q). The opinions evolve over discrete timestamps $\{0, 1, ..., t\}$. We denote the opinion(s) at timestamp t by, e.g., $B_q^{(t)}$ and $b_{qi}^{(t)}$.

A. Opinion Diffusion Models

Unlike the classic influence diffusion, opinion diffusion involves aggregating the peers' opinions at each timestamp [36]. We introduce a column-stochastic influence matrix [19], [37] $W \in [0, 1]^{n \times n}$, where $w_{ij} \in [0, 1]$ denotes the influence weight from user i to user j. Different candidates c_q can have different matrices W_q. Notice that *barring these weights, the graph structure and the nodes remain the same for all candidates*. The set E is the union of the edges with non-zero weights across all candidates. This setting is used in topic-aware IM [38]. We next present two widely used opinion diffusion models: DeGroot [19] and its extension FJ [20], [21].

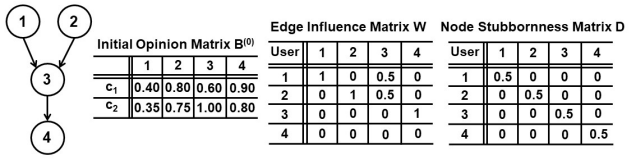
The DeGroot Model for a single candidate c_q is given by:

$$B_q^{(t)} = B_q^{(t-1)} W_q = B_q^{(t-2)} W_q^2 = ... = B_q^{(0)} W_q^t \tag{1}$$

At every timestamp, each user adopts the weighted average of her in-neighbors' opinions from the previous timestamp. Users without in-neighbors retain their initial opinions. Since W_q is column-stochastic, the opinion values remain in $[0, 1]$. We assume that the opinions about different candidates diffuse independently. In multi-campaigner and multi-feature settings, independent propagation of opinions and influences has been considered in [39], [40], [41], [42]. Note that in our case, *while the opinion propagation for multiple campaigns happens concurrently and independently, voting-based scores naturally incorporate competition among the campaigns (§ II-B).*

The Friedkin-Johnsen (FJ) Model extends the DeGroot model by introducing the notion of *stubbornness*:

$$B_q^{(t+1)} = B_q^{(t)} W_q \left(I - D_q\right) + B_q^{(0)} D_q \tag{2}$$



	1	2	3	4
c_1	0.40	0.80	0.60	0.90
c_2	0.35	0.75	1.00	0.80

User	1	2	3	4
1	1	0	0.5	0
2	0	1	0.5	0
3	0	0	0	1
4	0	0	0	0

User	1	2	3	4
1	0.5	0	0	0
2	0	0.5	0	0
3	0	0	0.5	0
4	0	0	0	0.5

Fig. 1. Running example. All users share the same influence weight and stubbornness matrices for both candidates.

$D_q = diag(d_{q1}, d_{q2}, ..., d_{qn})$ is a diagonal matrix: d_{qi} represents the stubbornness of user i on retaining her initial opinion about candidate c_q. If $d_{qi} = 1$, the user i is *fully stubborn* and sticks to her initial opinion about c_q. A *partially stubborn* user ($0 < d_{qi} < 1$) aggregates the opinions from neighbors as well as her original opinion, while *non-stubborn* users ($d_{qi} = 0$) follow the DeGroot model. Since the DeGroot model is a special case where all users are non-stubborn, all our results with the FJ model also hold for the DeGroot model.

If the opinions of all users do not change after a specific timestamp, the diffusion reaches a state of *convergence*. The conditions for the DeGroot or FJ model to reach convergence can be found in our extended version [35]. One of our novel contributions is the seed selection for opinion maximization *at any given time horizon*, which introduces non-trivial additional hardness, as discussed in § III-A and § III-B.

Example 1. *The input graph in Figure 1 consists of 4 users and 3 edges. Suppose c_1 is our target candidate and c_2 is a competing candidate. Based on the FJ model, for any $x \in \{1, 2\}$, a user's opinion about candidate c_x at any time horizon can be computed by taking the weighted average of her in-neighbors' opinions at the previous time horizon and then averaging with that of herself. Thus, users 1 and 2 will always keep their initial opinions, as they do not have any incoming edge. The opinion of user 3 at any time horizon t can be computed as $b_{x3}^{(t)} = \frac{1}{2}\left[b_{x3}^{(t-1)} + \frac{1}{2}\left(b_{x1}^{(t-1)} + b_{x2}^{(t-1)}\right)\right]$, which is the average opinion of users 1 and 2 at the previous time horizon, then averaged with that of user 3. For user 4, $b_{x4}^{(t)} = \frac{1}{2}\left[b_{x3}^{(t-1)} + b_{x4}^{(t-1)}\right]$, which is the average of the opinions of users 3 and 4 at the previous time horizon.*

B. Voting-based Scores

All campaigns start at timestamp 0 and proceed concurrently (FJ model), independently of each other. Given a time horizon t, we employ several voting-based scores [22], [23], [24] to decide the winning candidate. In particular, we compute a score $F(B^{(t)}, c)$ for each candidate c. The one with the maximum score is the winner at time t. We next define three major voting-based score functions that we study.

Cumulative Score. For a candidate c_q, the *cumulative score* is the sum of all users' opinion values about her at time t:

$$F\left(B^{(t)}, c_q\right) = \sum_{v \in V} b_{qv}^{(t)} \tag{3}$$

Plurality Score. The *plurality score* counts the number of users who prefer c_q to all other candidates at time t:

$$F\left(B^{(t)}, c_q\right) = \sum_{v \in V} \mathbb{1}\left[b_{qv}^{(t)} > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}\right] \tag{4}$$

$\mathbb{1}[\cdot]$ is an indicator that returns 1 if the condition inside is true, 0 otherwise. In practice, a user generally votes for only one politician, or has a limited budget to purchase one specific type of product. Intuitively, she selects the one with the highest opinion value in her mind – the plurality score captures this. In our extended version [35], we discuss more variants of the plurality score which suit more application scenarios. We also show that *all* our theorems and solutions can generalize to these variants with minor modifications.

Copeland Score. We define an ordering $\succ_M$ on candidates: $c_q \succ_M c_p$ (i.e., c_q wins over c_p), if more users have a higher opinion value for c_q than for c_p, compared to the other way around, at time t. The score counts how many such one-on-one competitions a candidate c_q wins:

$$F\left(B^{(t)}, c_q\right) = |\{c_p \colon c_q \succ_M c_p\}|$$
$$= \sum_{c_x \in C \setminus \{c_q\}} \mathbb{1}\left[\sum_{v \in V} \mathbb{1}\left[b_{qv}^{(t)} > b_{xv}^{(t)}\right] > \sum_{v \in V} \mathbb{1}\left[b_{qv}^{(t)} < b_{xv}^{(t)}\right]\right] \tag{5}$$

The Condorcet winner [43] is the candidate that wins all such one-on-one competitions, i.e., has the maximum possible $F(B^{(t)}, c_q)$ score, which is $r - 1$. In general, a Condorcet winner is not always guaranteed to exist [43]. However, maximizing the *Copeland score* boosts the target candidate to beat as many other candidates as possible, and to be as close to become a Condorcet winner as possible.

C. Problem Formulation

We study the novel problem of selecting k seed nodes for a target candidate that maximize one of the voting-based scores discussed in § II-B for the target candidate at a given time horizon. All our scoring functions are non-decreasing w.r.t. seed sets (§ III-B). Maximizing the score boosts the target candidate's odds of being as close as possible to winning.

For each node s in the seed set S for candidate c_q, we increase $b_{qs}^{(0)}$ and d_{qs} to 1 (i.e., node s becomes fully stubborn towards retaining the maximum opinion value about c_q). We denote the modified initial opinion row vector B_q and the stubbornness matrix D_q as $B_q[S]$ and $D_q[S]$, respectively. The problem is formulated as follows.

Problem 1 (FJ-Vote). *Given the initial opinion matrix $B^{(0)}$, a target candidate c_q, influence matrix W_q, stubbornness matrix D_q, and a time horizon t, find a set of k seed nodes $S \subset V$ that maximizes the score for c_q at timestamp t. Formally,*

$$S^* = \underset{S \subset V, |S| = k}{\arg\max} F\left(B^{(t)}[S], c_q\right) \tag{6}$$

Here $B^{(t)}[S]$ is computed from $B^{(0)}[S]$ via the FJ model (Equation 2). Note that $B^{(0)}[S]$ is obtained from the initial opinion matrix $B^{(0)}$ by updating its row vector B_q to $B_q[S]$ according to the seed set S for c_q. The function F is based on one of the three voting scores (§ II-B).

Example 2. *Suppose we aim to choose one seed user to maximize the score for c_1 (i.e., improve c_1's odds of winning against competitor c_2) at time horizon $t = 1$. The optimal seed sets are quite different for various voting-based scores. As shown in Table I, selecting user 1 as the seed leads to the maximum cumulative score; however, we still have only 2 users preferring our target candidate c_1 to c_2. Thus, the*

TABLE I
SCORES OF CANDIDATE c_1 FOR VARIOUS SEED SETS AT $t = 1$ IN FIGURE 1. ASSUMING NO SEEDS FOR c_2, THE OPINIONS OF USERS 1, 2, 3, 4 ABOUT c_2 AT $t = 1$ ARE RESP. 0.35, 0.75, 0.78, 0.90.

Seed Set	User				Score		
	1	2	3	4	Cumu.	Plu.	Cope.
$\{\}$	**0.40**	**0.80**	0.60	0.75	2.55	2	0
$\{1\}$	**1.00**	**0.80**	0.75	0.75	**3.30**	2	0
$\{2\}$	**0.40**	**1.00**	0.65	0.75	2.80	2	0
$\{3\}$	**0.40**	**0.80**	**1.00**	**0.95**	3.15	**4**	**1**
$\{4\}$	**0.40**	**0.80**	0.60	**1.00**	2.80	3	**1**
$\{1, 2\}$	**1.00**	**1.00**	**0.80**	0.75	**3.55**	3	**1**

Copeland score of c_1 remains 0. Choosing user 3 as the seed will encourage all four users to favor c_1 over c_2, which results in the highest plurality score. Meanwhile, c_1 will become the Condorcet winner (Copeland score equals 1) when user 3 or 4 is selected as the seed, since more than half the users will have higher opinion values for c_1 than for c_2.

Remarks. We assume that the opinion diffusion for multiple candidates proceeds concurrently and independently, following [39], [40], [41], [42]. **(1)** For the cumulative score, due to its aggregate nature, the top-k seeds for the target candidate can be computed independent of the others, similar to the single-campaigner setting [25], [26]. In contrast, *our other voting-based scores (plurality and Copeland) incorporate competition among the candidates via ranking-based formulations using each user's preference order.* **(2)** As long as we know the seed sets for the non-target candidates at the beginning of the diffusion (i.e., at time 0), our algorithm can compute their opinions at any time horizon, and we select the k seed nodes for the target campaign (also at time 0) so as to maximize the target's voting-based score at the time horizon, *relative to the placement of seeds for non-target candidates* at time 0. Thus, *while our analyses and techniques apply for this general case where the competing candidates have seeds, for simplicity of notation and exposition, we assume w.l.o.g. that the non-target candidates have no seeds*. **(3)** Since we find the seed set of size at most k that maximizes the score of the target candidate, winning is not always guaranteed, because even after selecting the k optimal seed nodes for the target candidate, another candidate may still have a higher score than the target. In that case, the target candidate needs more seeds to win. The following variant of our problem can mitigate this issue.

Problem 2 (FJ-Vote-Win)**.** *Given the initial opinion matrix $B^{(0)}$, a target candidate c_q, influence matrix W_q, stubbornness matrix D_q, and a time horizon t, find a set of seed nodes $S^* \subset V$ of minimum size k^* such that the score for c_q at timestamp t is the largest among all candidates. Formally,*

$$S_k^* = \underset{S \subset V, |S|=k}{\arg\max} F\left(B^{(t)}[S], c_q\right)$$
$$k^* = \min\left\{k : \left[F\left(B^{(t)}\left[S_k^*\right], c_q\right) > \max_{c_x \in C \setminus \{c_q\}} F\left(B^{(t)}\left[S_k^*\right], c_x\right)\right]\right\}$$
$$S^* = S_{k^*}^* \tag{7}$$

In § III-C, we show that a solution to Problem 1 can be extended to solve this new problem.

III. BASIC RESULTS & SOLUTION FRAMEWORK

In this section, we discuss the hardness of our problem (§ III-A) and the submodularity of our scores (§ III-B), followed by a greedy solution to our problem (§ III-C). All of these are a part of our novel contributions. A summary of these properties for all our scores is given in Table II.

TABLE II
PROPERTIES OF OUR VOTING-BASED SCORES

Score	NP-hard	Non-negative	Non-decreasing	Submodular
Cumulative	Yes	Yes	Yes	Yes
Plurality	Yes	Yes	Yes	No
Copeland	Open	Yes	Yes	No

A. Hardness

We show that the decision version of Problem 1 is **NP**-hard for the cumulative and plurality scores.

Theorem 1. *The decision version of Problem 1 is* **NP***-hard with the cumulative score.*

Proof. We prove by a reduction from the **NP**-hard VERTEX COVER problem [44]. A vertex cover in an undirected graph $G = (V, E)$ is a subset of nodes such that every edge in E is incident to at least one of them. Given G and an integer k, the decision version of the problem asks if G contains a vertex cover of size at most k.

Let $|V| = n$ and $|E| = m$. G is transformed into a directed graph $\mathcal{G} = (V, E')$, where E' contains directed edges (u, v) and (v, u) for each undirected edge $(u, v) \in E$. We create two candidates c_q (our target) and c_x. For each $y \in \{q, x\}$, we set the following: for each $i \in V$, $b_{yi}^{(0)} = 0$, $d_{yi} = 0$; and for each $(i, j) \in E'$, $w_{y;ij} = 1/deg(j)$, where $deg(v)$ denotes the degree of node v in G. This ensures that W_y is column-stochastic. The time horizon t is set to 1. This reduction takes $\mathcal{O}(m + n)$ time. We prove that a set S of at most k nodes is a vertex cover of G if and only if $F(B^{(1)}[S], c_q) \geq n$.

(1) If S is a vertex cover in G, then each node v in $\mathcal{G}$ either belongs to S or has all of its incoming neighbors in S. In the former case, $b_{qv}^{(1)}[S] = 1$ by definition. In the latter case, since W_q is column-stochastic, it follows from Eq. 2 that $b_{qv}^{(1)}[S] = 1$. This implies that $F(B^{(1)}[S], c_q) = n$. **(2)** If S is not a vertex cover in G, then there exists at least one edge $(u, v) \in E$ such that neither u nor v is in S. This implies that $b_{qv}^{(1)}[S] \leq 1 - 1/deg(v) < 1$, which means that $F(B^{(1)}[S], c_q) < n$. The theorem follows. □

Remark: While Problem 1 with the cumulative score is similar to [25], a key difference is as follows. Unlike Problem 1, [25] selects seeds to maximize the sum of the expressed opinions *at the Nash equilibrium*, instead of at *a given finite time horizon*. The proofs of NP-hardness and submodularity in [25] rely on showing that an absorbing random walk is an unbiased estimate of the true equilibrium opinion. However, we cannot use absorbing random walks to estimate opinions at *a finite time horizon*, rendering their proofs inapplicable in our case. *Our* **NP***-hardness and submodularity proofs for the cumulative score are novel contributions.*

Theorem 2. *The decision version of Problem 1 is* **NP***-hard with the plurality score.*

Proof. The reduction remains the same as in the proof of Theorem 1, except that c_x satisfies $b_{xv}^{(0)} = 1 - \delta \; \forall v \in V$, where $0 < \delta < \min_{v \in V} 1/deg(v)$; this ensures that $b_{xv}^{(1)} = 1 - \delta$. □

The computational complexity of Problem 1 with the Copeland score is open as of now. We, however, show in § III-B that the Copeland score is not submodular.

B. Submodularity

We show that the cumulative score used in Problem 1 is submodular, while the plurality and Copeland scores are not. A set function $f : 2^V \rightarrow \mathbb{R}^{\geq 0}$ over a ground set V is submodular if $f(X \cup \{i\}) - f(X) \geq f(Y \cup \{i\}) - f(Y)$, $\forall X \subset Y \subset V, i \in V \setminus Y$. The classic greedy algorithm returns a $(1-1/e)$-approximate solution for maximizing a non-negative, non-decreasing, submodular function [45]. Including a user s into the seed set S will increase her opinion value on c_q, which will in turn influence those of some other users. Thus, after the inclusion of s into S, each user's opinion value and ranking of c_q cannot decrease. Hence, *all our scoring functions are non-decreasing in seed sets for* c_q.

Submodularity of the Cumulative Score.

Theorem 3. *The opinion value of any user i about any candidate c_q is submodular w.r.t. the seed set for that candidate. Formally, $\forall X \subseteq Y \subseteq V, s \in V \setminus Y$,*

$$b_{qi}^{(t)}[X \cup \{s\}] - b_{qi}^{(t)}[X] \geq b_{qi}^{(t)}[Y \cup \{s\}] - b_{qi}^{(t)}[Y] \tag{8}$$

The cumulative score is the sum of all users' opinion values (Equation 3). As the sum of submodular functions is also submodular, the cumulative score is submodular.

Non-Submodularity of the Other Scoring Functions. We show the non-submodularity of the plurality and Copeland scores using the same running example (Figure 1 and Table I).

Example 3. *As shown in Table I, inserting node 2 into the empty seed set results in zero marginal gain for both the plurality and Copeland scores. However, inserting node 2 into seed set $\{1\}$ will make user 3 preferring the target candidate c_1 (resulting in marginal gain 1 for the plurality score) and also the number of users preferring c_1 more than the same for c_2 (resulting in marginal gain 1 for the Copeland score). Hence, submodularity is violated for both scores.*

C. Solution Overview

Since the cumulative score is non-negative, non-decreasing, and submodular, the greedy framework (Algorithm 1), which identifies the node that maximizes the marginal gain in score at each round, can provide a $(1-1/e)$-approximate solution. We show in our extended version [35] that there is a problem instance for which the well-known submodularity ratio ψ [46], [47] becomes 0 for our other non-submodular voting-based scores; thus their approximation factor $(1-e^{-\psi})$ degrades and goes to 0. However, in § IV, with the help of *Sandwich Approximation* [31], we prove that the greedy framework can still generate good approximate solutions for these scores.

Time Complexity with the Cumulative Score. To find the node that maximizes the marginal gain at each round of Algorithm 1, one can apply Eq. 2 t times (due to the input time horizon t). Since every such matrix-vector multiplication has time complexity $\mathcal{O}(m)$ using a sparse matrix package, we have k rounds (to find the top-k seed nodes), and $\mathcal{O}(n)$ candidate nodes from which a seed node is selected in each round, the final time complexity is $\mathcal{O}(ktmn)$. As the cumulative score is monotone and submodular, we also apply the CELF optimization [48]. In § V and § VI, we propose *random walk*- and *sketching*-based estimation, respectively, to further improve the efficiency, with theoretical quality guarantees.

Algorithm 1 `Greedy` Seed Selection

Require: Graph $\mathcal{G} = (V, E)$, initial opinion matrix $B^{(0)}$, influence matrix W_i and stubbornness matrix D_i for each candidate c_i, target candidate c_q, seed set size budget k, time horizon t, and a scoring function F
Ensure: Seed set S^* of size k
1: $S^* \leftarrow \emptyset$
2: **for** $j = 1$ **to** k **do**
3: $\quad u \leftarrow \arg\max_{v \in V \setminus S^*} \left[F\left(B^{(t)}[S^* \cup \{v\}], c_q\right) - F\left(B^{(t)}[S^*], c_q\right) \right]$
4: $\quad S^* \leftarrow S^* \cup \{u\}$
5: **return** S^*

Remark. **(1)** This greedy solution can be extended to solve Problem 2 about finding the smallest seed set size k^* such that the target candidate wins. Since $0 \leq k^* \leq n$ and our scoring functions are non-decreasing, we resort to a binary search for k^*, with the initial lower (resp. upper) bound as 0 (resp. n). In each iteration, we compute the optimal seed set S of size at most the value midway between the bounds. If the target wins (resp. loses) with the seed set S, the upper (resp. lower) bound is updated to the middle value and the process is repeated till the bounds converge. **(2)** Due to the hardness of our problem (§ III-A), we find an "approximately optimal" seed set (e.g., using Algorithm 1). Since such a seed set will lead to a lower voting-based score than that for the optimal solution, the final seed set size obtained could be larger than the true minimal one to achieve the winning criterion.

IV. Plurality and Copeland Scores: Sandwich Approximation

Sandwich Approximation [31] (§IV-A) is a powerful framework for providing approximation guarantees for non-submodular function maximization. Our novel contribution is to construct non-trivial upper and lower bound functions to enable sandwich approximation for our plurality (§ IV-B) and Copeland (§ IV-C) scores, as they must satisfy certain properties to admit good approximations. Furthermore, we empirically validate that the additional ratio introduced by sandwich approximation (which degrades the overall approximation) is reasonably high for our proposed bounding functions in all cases (§ IV-D). For simplicity, we rewrite $F(B^{(t)}[S], c_q)$ as $F(S)$, since the target candidate c_q is arbitrary but fixed.

A. Sandwich Approximation

For any non-submodular set function $F(S)$, $S \subseteq V$, suppose $UB(S)$ and $LB(S)$ are any set functions defined on the same ground set V, such that $LB(S) \leq F(S) \leq UB(S)$, $\forall S \subseteq V$. If we are able to compute approximate solutions for both $UB(S)$ and $LB(S)$, then we can obtain the sandwich approximation for the targeted set function $F(S)$ as follows. **(1)** Run the approximation algorithms to obtain an η-approximate solution S_U to $UB(S)$ and a τ-approximate solution S_L to $LB(S)$, where η (resp. τ) is the approximation factor afforded by the algorithm for $UB(S)$ (resp. $LB(S)$). **(2)** Find a feasible solution S_F to function $F(S)$, e.g., by

applying the standard greedy algorithm. **(3)** Report the final solution $S^{\#}$: $S^{\#} = \arg\max_{S \in \{S_U, S_L, S_F\}} F(S)$.

Theorem 4 ([31])**.** *Sandwich approximation guarantees:*

$$F\left(S^{\#}\right) \geq \max\left\{\eta \cdot \frac{F\left(S_U\right)}{UB\left(S_U\right)} \cdot F\left(S_F^*\right), \tau \cdot LB\left(S_F^*\right)\right\} \quad (9)$$

where S_F^* maximizes $F(S)$ subject to a constraint, e.g., a cardinality constraint $|S| \leq k$, or a matroid constraint.

B. Bounds on the Plurality Score

Motivated by this result, we design *non-negative, non-decreasing, submodular* lower and upper bounding functions $LB(S)$ and $UB(S)$ such that $0 \leq LB(S) \leq F(S) \leq UB(S)$, $\forall S \subseteq V$, thus enabling sandwich approximation with $\eta = \tau = 1 - 1/e$ (Eq. 9), via running Algorithm 1 on $LB(S)$, $F(S)$, and $UB(S)$, respectively. Note that ensuring the submodularity of $LB(\cdot)$ and $UB(\cdot)$ is one (not the only) way to enable sandwich approximation. We first define two useful terms.

Definition 1 (Favorable Users Set)**.** *The favorable users set, denoted by $V_q^{(t)}$, is the set of nodes (users) who would have the highest opinion value about the target candidate c_q compared to the other candidates at the time horizon t, even without introducing any seed for c_q. Formally,*

$$V_q^{(t)} = \left\{v \in V : b_{qv}^{(t)} > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}\right\} \quad (10)$$

Since the opinion of a user about c_q increases with the seed set for c_q, and the users in $V_q^{(t)}$ prefer c_q to the other candidates at the time horizon t even without any seed for c_q, they will continue doing so on the addition of seed nodes for c_q. Recall that the set of such users at the time horizon t decides c_q's plurality score. Hence, we use $V_q^{(t)}$ to construct a lower bound for the plurality score (Definition 3).

Definition 2 (Reachable Users Set)**.** *The reachable users set, denoted by $N_S^{(t)}$, is the set of nodes (users) at most t outgoing hops away from any node in a seed set S. Formally, denoting by $u \overset{h}{\rightsquigarrow} v$ the existence of a path with h edges from u to v,*

$$N_S^{(t)} = \bigcup_{s \in S} \bigcup_{h=0}^{t} \left\{v \in V : s \overset{h}{\rightsquigarrow} v\right\} \quad (11)$$

On adding seeds for c_q, along with the users in $V_q^{(t)}$, some additional users can also have higher opinions about c_q at time t, who according to the FJ model, can be at most t outgoing hops away from a seed node. Hence, $V_q^{(t)}$ and $N_S^{(t)}$ are used to construct an upper bound for the plurality score (Definition 4).

Definition 3. *The lower bounding function $LB(S)$ for the plurality score $F(S)$ is defined as the aggregated opinion value about c_q at time t for all users in the favorable users set, on the introduction of a seed set S for c_q.*

$$LB(S) = \sum_{v \in V_q^{(t)}} b_{qv}^{(t)}[S] \quad (12)$$

Definition 4. *The upper bounding function $UB(S)$ for the plurality score $F(S)$ is defined as the total number of users either in the favorable users set or in the reachable users set.*

$$UB(S) = \left|N_S^{(t)} \cup V_q^{(t)}\right| \quad (13)$$

Correctness Guarantee. We now have:

Theorem 5. *$LB(S)$ is* **(1)** *non-negative,* **(2)** *non-decreasing,* **(3)** *submodular, and* **(4)** *a lower bound for $F(S)$.*

Proof. **(1)** Since $b_{qv}^{(t)}[S] \geq 0\ \forall v \in V$, $LB(S) \geq 0$. **(2)** $LB(S)$ is the sum of $b_{qv}^{(t)}[S]\ \forall v \in V_q^{(t)}$, and each of them is non-decreasing w.r.t. the inclusion of seeds in S. **(3)** From Theorem 3, each $b_{qv}^{(t)}[S]$ is submodular, and hence so is $LB(S)$, which is the sum of such functions $\forall v \in V_q^{(t)}$. **(4)** Notice that $v \in V_q^{(t)}$ implies: $b_{qv}^{(t)}[S] \geq b_{qv}^{(t)} > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}$. Thus,

$$LB(S) = \sum_{v \in V_q^{(t)}} b_{qv}^{(t)}[S] \leq \sum_{v \in V_q^{(t)}} 1 = \sum_{v \in V_q^{(t)}} \mathbb{1}\left[b_{qv}^{(t)}[S] > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}\right]$$
$$\leq \sum_{v \in V} \mathbb{1}\left[b_{qv}^{(t)}[S] > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}\right] = F(S)$$

Lemma 1. *If a user v is not in the reachable users set, then the opinion of v about c_q does not change by virtue of the seed set. Formally, if $v \notin N_S^{(t)}$, then $b_{qv}^{(t)}[S] = b_{qv}^{(t)}$.*

Intuitively, this follows from the FJ model; the influence of the seed set diffuses by one hop in each timestamp, and hence cannot spread beyond t hops at timestamp t.

Theorem 6. *$UB(S)$ is* **(1)** *non-negative,* **(2)** *non-decreasing,* **(3)** *submodular, and* **(4)** *an upper bound for $F(S)$.*

Proof. **(1)** Since the size of any set is non-negative, $UB(S) \geq 0$. **(2)** $UB(S)$ is non-decreasing because, for any $X \subseteq Y$,

$$UB(Y) = UB(Y \cup X) = \left|N_{Y \cup X}^{(t)} \cup V_q^{(t)}\right|$$
$$= \left|N_Y^{(t)} \cup N_X^{(t)} \cup V_q^{(t)}\right| \geq \left|N_X^{(t)} \cup V_q^{(t)}\right| = UB(X)$$

(3) $UB(S)$ is submodular as for $X \subset Y \subset V$ and $s \in V \setminus Y$,

$$UB(X \cup \{s\}) - UB(X) = \left(\left|N_{X \cup \{s\}}^{(t)} \cup V_q^{(t)}\right| - \left|N_X^{(t)} \cup V_q^{(t)}\right|\right)$$
$$= \left|N_{\{s\}}^{(t)} \setminus \left(N_X^{(t)} \cup V_q^{(t)}\right)\right| \geq \left|N_{\{s\}}^{(t)} \setminus \left(N_Y^{(t)} \cup V_q^{(t)}\right)\right|$$
$$= \left(\left|N_{Y \cup \{s\}}^{(t)} \cup V_q^{(t)}\right| - \left|N_Y^{(t)} \cup V_q^{(t)}\right|\right) = UB(Y \cup \{s\}) - UB(Y)$$

(4) Suppose $v \notin N_S^{(t)}$ and $v \notin V_q^{(t)}$. Then, from Equation 10 and Lemma 1, $\exists c_x \in C \setminus \{c_q\} : b_{xv}^{(t)} \geq b_{qv}^{(t)} = b_{qv}^{(t)}[S]$. Thus, $b_{qv}^{(t)}[S] > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}$ implies $v \in N_S^{(t)} \cup V_q^{(t)}$. Hence,

$$F(S) = \sum_{v \in V} \mathbb{1}\left[b_{qv}^{(t)}[S] > \max_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)}\right]$$
$$\leq \sum_{v \in V} \mathbb{1}\left[v \in N_S^{(t)} \cup V_q^{(t)}\right] = \left|N_S^{(t)} \cup V_q^{(t)}\right| = UB(S)$$

C. Upper Bound for the Copeland Score

We construct a non-negative, non-decreasing, submodular upper bounding function for the Copeland score in a similar way as in § IV-B, under the constraint that no user has equal opinion values about any two candidates at the time horizon. Notice that this constraint does not change the definition of the Copeland score (Eq. 5) in any way; rather, whether this constraint holds or not depends on the input dataset, the seed set, and the time horizon. We enable sandwich approximation via running Algorithm 1 on $F(S)$ and $UB(S)$ only, and we get $\eta = 1 - 1/e$ in Eq. 9. As in § IV-B, ensuring the submodularity of $UB(\cdot)$ is one (not the only) way to enable sandwich approximation. The construction of a useful lower

bound and the case when a user has equal preference to two candidates at the time horizon are open for future work.

Definition 5 (Weakly Favorable Users Set)**.** *The weakly favorable users set, denoted by $U_q^{(t)}$, is the set of nodes (users) who prefer c_q to at least one other candidate at the time horizon t, even without having any seed for c_q. Formally,*

$$U_q^{(t)} = \left\{ v \in V : b_{qv}^{(t)} > \min_{c_x \in C \setminus \{c_q\}} b_{xv}^{(t)} \right\} \tag{14}$$

Since the Copeland score computes the number of one-on-one competitions won by c_q, only those users who prefer c_q to at least one other candidate, i.e., those in $U_q^{(t)}$, can contribute to this score, along with those users who could be influenced by the seed set, i.e., those in $N_S^{(t)}$. Thus, $U_q^{(t)}$ and $N_S^{(t)}$ are used to construct an upper bound as below.

Definition 6. *The upper bounding function $UB(S)$ for the Copeland score $F(S)$ is defined as the total number of users either in the weakly favorable users set or in the reachable users set, times the ratio of the number of non-target candidates to one more than half the total number of users.*

$$UB(S) = \frac{r-1}{\lfloor \frac{n}{2} \rfloor + 1} \left| N_S^{(t)} \cup U_q^{(t)} \right| \tag{15}$$

Correctness Guarantee. We show that $UB(S)$ is a non-negative, non-decreasing, submodular upper bound for $F(S)$. The proof, similar to that of Theorem 6, is given in [35].

Theorem 7. *$UB(S)$ is* **(1)** *non-negative,* **(2)** *non-decreasing,* **(3)** *submodular, and* **(4)** *an upper bound for $F(S)$.*

D. Practical Effectiveness of Our Bounds

We empirically study the ratio $\frac{F(S_U)}{UB(S_U)}$ (see [35] for details), since sandwich approximation ensures an approximation factor of at least $\frac{F(S_U)}{UB(S_U)}\left(1-\frac{1}{e}\right)$ according to Equation 9. On all our *Twitter* datasets (see § VIII for details), the ratio reaches 0.7 in 90% of the trials, and in about 60% of them, it exceeds 0.8 for both the plurality and Copeland scores. This results in an empirical approximation factor of at least $0.8(1-1/e) \approx 0.51$ in more than half of our trials. In practice, our algorithm performs much better than several baselines (§ VIII).

The greedy algorithm for finding S_U is much faster than that for computing S_F (Algorithm 1), since it does not involve any expensive opinion computation. Meanwhile, S_L is obtained via greedily maximizing the cumulative score on $V_q^{(t)}$ (Definition 3), which is also much faster, since **(1)** $\left|V_q^{(t)}\right| \ll |V|$ in practice, and **(2)** the greedy algorithm for the cumulative score is much faster than that for the other scores (§ VIII-C). Empirically, the running times for finding S_U and S_L are about 2% and 5%, respectively, of that for finding S_F.

V. Efficient Random Walk-based Estimation

The greedy framework (Algorithm 1) has time complexity $\mathcal{O}(ktmn)$ via inefficient direct matrix-vector multiplication (§ III-C). In this section, we first introduce a random walk interpretation for the opinion value of any node at any timestamp (§ V-A). Next, as our novel contribution, an *efficient* random walk-based method with a *smart truncation strategy* is designed to estimate the marginal gain (§ V-B). Finally, we establish novel *quality guarantees* of the proposed method for *all* our voting-based scores (§ V-C).

A. Random Walk Interpretation

As the influence matrix W_q is column-stochastic for any candidate c_q, the probabilities on the outgoing edges of each node add up to 1 in the reverse graph.[4] This enables the following *Direct Generation* of t-step random walks with seed set S. **(1)** Each node v in the reverse graph has a termination probability $d_{qv}[S] \in [0,1]$ that is equivalent to its stubbornness (recall that $d_{qv}[S] = 1$ if $v \in S$ and $d_{qv}[S] = d_{qv}$ otherwise), and the probabilities on its outgoing edges add up to 1. **(2)** If a random walk is at node v in the current step, it terminates at v with probability $d_{qv}[S]$. Otherwise, it proceeds to an out-neighbor of v chosen according to the edge probabilities. **(3)** From a start node u, we repeat step (2) to generate a random walk. It terminates when step (2) has been conducted t times, or the walk stops early (i.e., before reaching length t) at a node due to the termination probability. **(4)** If the random walk terminates at node v, then the node u at time t adopts the initial opinion of node v: $X_{qu}^{(t)}[S] = b_{qv}^{(0)}[S]$. We show that the expected opinion value of any node u at any time t when serving as the start node of the above reverse random walk is the same as the exact opinion value of u at time t computed by matrix-vector multiplication.[5]

Theorem 8. *For any $t \geq 0$ and seed set S, the expected value of the estimated opinion $X_{qu}^{(t)}[S]$ of any user u about any candidate c_q at timestamp t using a t-step reverse random walk by* Direct Generation *is equal to the exact opinion of u about c_q at timestamp t according to the FJ model. Formally,*

$$\mathbb{E}\left[X_{qu}^{(t)}[S]\right] = b_{qu}^{(t)}[S] \tag{16}$$

B. The Algorithmic Workflow

We estimate the opinion of every user v about any candidate c_q at time t by generating λ_v independent t-step reverse random walks starting from v. The estimated opinion of node v about candidate c_q is computed as the average of the initial opinions of the end nodes across all λ_v random walks. The seed set is generated greedily as in Algorithm 1. In Line 3, we select the best new seed based on the maximum *estimated* marginal gain instead of the maximum *actual* marginal gain. In each iteration, given the previously selected seed set S^* for c_q, we need to compute the marginal gain of including a candidate seed node w into S^*, and hence the estimated opinions with the new seed set. The *Direct Generation* approach would require the generation of new walks with the new seed set, which would be expensive. Thus, we use an alternative *Post-Generation Truncation* technique as follows: Before running Algorithm 1, we generate (only once) λ_v random walks from each node v using the same approach as in § V-A but with the empty seed set. Thereafter, for any given seed set S, the

[4]The reverse graph has the same set of nodes and edges, but with edge directions reversed. The weights on the edges, now interpreted as probabilities, remain the same.

[5]Random walks for approximating matrix-vector multiplication are employed in [33] and in PageRank [32], albeit with subtle differences from ours. While [32], [33] require a *one-time* estimation of the vector entries, we do so in an efficient way for *k iterations* of the greedy algorithm. Also, the quality guarantees required are different from [32], [33] and specific to each voting-based score (more details in [35]).

estimated opinion $Y_{qv}^{(t)}[S]$ for a given walk is the initial opinion of the end node of the walk truncated at the first occurrence of a node from S. The overall estimated opinion $\widehat{b}_{qv}^{(t)}[S]$ of v is the average of $Y_{qv}^{(t)}[S]$ across all λ_v walks from v. The above approach is clearly more efficient since it does not involve regenerating random walks for each seed set. It also does not introduce any further error, since the estimates $Y_{qv}^{(t)}[S]$ satisfy the same property as $X_{qv}^{(t)}[S]$ in Theorem 8, as shown below.

Theorem 9. *For any $t \geq 0$, any node u and any seed set S, let $Y_{qu}^{(t)}[S]$ denote the estimated opinion of u about c_q at time t by the* Post-Generation Truncation *approach, i.e., the initial opinion of the end node of the resultant random walk after initially sampling a t-step reverse random walk starting from u without any seed and then truncating the walk at the first occurrence of a node in S. Then*

$$\mathbb{E}\left[Y_{qu}^{(t)}[S]\right] = b_{qu}^{(t)}[S] \tag{17}$$

Time Complexity. For the target candidate, the generation of t-step reverse random walks starting from all nodes takes $\mathcal{O}\left(t\sum_{v\in V}\lambda_v\right)$ time. First, we analyze the time complexity of finding the top-k seed nodes for the cumulative score via random walk-based estimation. In each iteration, as detailed in [35], choosing the next seed node w scans all the random walks once and takes $\mathcal{O}\left(t\sum_{v\in V}\lambda_v\right)$ time. Next, all walks containing w are truncated at w for the subsequent iterations. This step also takes $\mathcal{O}\left(t\sum_{v\in V}\lambda_v\right)$ time. As the entire process is repeated k times (to find the top-k seed nodes), the running time of the seed selection phase is $\mathcal{O}\left(kt\sum_{v\in V}\lambda_v\right)$.

For plurality and Copeland scores, we additionally compute the exact opinion values of each user about all other candidates at time t via direct matrix-vector multiplication, taking an additional $\mathcal{O}\left((r-1)tm\right)$ time. Thus, the overall time complexity for these scores is $\mathcal{O}\left(kt\sum_{v\in V}\lambda_v + (r-1)tm\right)$. Practically, thanks to the sparseness of the matrices, the dominant term is the first one due to the seed selection phase.

C. Accuracy Guarantees

Cumulative Score. The cumulative score aggregates the opinion values of all users about a target candidate c_q. We provide a probabilistic accuracy guarantee about the estimated opinion, which follows from Hoeffding's inequality, as shown in [35].

Theorem 10. *Given $\delta, \rho > 0$, if $\lambda_v \geq \frac{1}{2\delta^2}\ln\left(\frac{2}{1-\rho}\right)$, for any node v the following holds with probability at least ρ:*

$$\left|\widehat{b}_{qv}^{(t)}[S] - b_{qv}^{(t)}[S]\right| < \delta \tag{18}$$

Plurality Score. Each user contributes a binary value (0/1) denoting whether she ranks c_q as the highest or not. Theorem 11 ensures that, with a high probability, our approach correctly estimates this contributed value.

Theorem 11. *Given a user v and a seed set S for candidate c_q, let $\gamma_v[S] = \min_{c_p\in C\setminus\{c_q\}}\left|b_{pv}^{(t)} - b_{qv}^{(t)}[S]\right|$, $\lambda_v \geq \frac{1}{2(\gamma_v[S])^2}\ln\left(\frac{2}{1-\rho}\right)$. Assume $\gamma_v[S] \neq 0$. Then, with probability at least ρ, the following holds:*

$$\mathbb{1}\left[\widehat{b}_{qv}^{(t)}[S] > \max_{c_x\in C\setminus\{c_q\}} b_{xv}^{(t)}\right] = \mathbb{1}\left[b_{qv}^{(t)}[S] > \max_{c_x\in C\setminus\{c_q\}} b_{xv}^{(t)}\right] \tag{19}$$

In each iteration of Algorithm 1, the estimation of the opinion of user v about c_q involves an average over λ_v random walks. However, the quantity $\gamma_v[S]$ in Theorem 11 depends on the seed set S for candidate c_q. For a given S, $\gamma_v[S]$ can be computed exactly via matrix-vector multiplication. But since S differs from iteration to iteration (specifically, one node is added in each iteration), a value of $\gamma_v[S]$ (and hence λ_v) that works well in one iteration may not work well in another iteration. As we generate random walks right in the beginning and reuse them for the subsequent iterations, a value of $\gamma_v[S]$ that works well in all iterations is: $\gamma_v^* = \min_{S\subseteq V\,:\,|S|\leq k}\gamma_v[S]$. However, efficiently computing the minimum over all seed sets S of size at most k is challenging. Thus, we estimate it heuristically using a greedy approach. Starting with $S = \emptyset$, we first estimate the opinion of user v about c_q by averaging over α random walks; α could, for example, be set to $\frac{1}{2\delta^2}\ln\left(\frac{2}{1-\rho}\right)$ in order to guarantee that, with probability at least ρ, each estimate differs from the true value by at most δ. Once these estimates are found, we can estimate $\gamma_v[S]$ as $\widehat{\gamma}_v[S]$. After this, we repeatedly add to S that node which minimizes the new $\widehat{\gamma}_v[S]$ computed using the newly estimated opinion values. The repetition stops once $|S| = k$ or there is no decrease in $\widehat{\gamma}_v[S]$, at which point we return $\widehat{\gamma}_v[S]$ as our estimate of γ_v^*.

Copeland Score. This score denotes the number of candidates against whom the target candidate wins in one-on-one competitions. Thus, we need the one-on-one winner to be predicted correctly (with high probability) using the estimated opinions.

Theorem 12. *Given a user v and a seed set S for candidate c_q, let $\gamma_v[S] = \min_{c_p\in C\setminus\{c_q\}}\left|b_{pv}^{(t)} - b_{qv}^{(t)}[S]\right|$. Suppose $\gamma_v[S] \neq 0$ and $\lambda_v \geq \frac{1}{2(\gamma_v[S])^2}\ln\left(\frac{1}{1-\rho}\right)$. Then the following holds with probability at least ρ for any $c_x \neq c_q$.*

$$\mathbb{1}\left[\widehat{b}_{qv}^{(t)}[S] > b_{xv}^{(t)}\right] = \mathbb{1}\left[b_{qv}^{(t)}[S] > b_{xv}^{(t)}\right] \tag{20}$$

VI. Sketch-based Estimation

Random walk-based approximation (§ V) requires the generation of reverse random walks starting from *all* nodes, which could still be expensive. In this section, we further propose a *more efficient* reverse sketching-based *approximation* technique. Notice that reverse sketching was used earlier in influence maximization (IM) [34], [7], [3]. We are the first to prove that the real-valued opinions in the FJ model can be estimated via reverse sketching and use it for opinion maximization. Moreover, our sketches (i.e., walks) are simpler and less memory consuming than the ones based on RR-sets (i.e., BFS trees), used in the classic IM.

A. The Algorithmic Workflow

We repeat the following θ times independently: Generate λ_v t-step reverse random walks starting from a node v chosen uniformly at random. We refer to the set of generated walks as the sketch set. These sketches are similar to the tree-structured sketches used in the classic IM [34], [7], [3] (see full version [35] for an intuition and formal proof). However, our sketches are walks, which are simpler and less memory consuming.

The opinions and the corresponding voting-based scores are estimated with the sketch set, as detailed in § VI-B. The greedy seed selection workflow remains the same as in Algorithm 1.

Time Complexity. The main difference between the sketching-based estimation method (§ VI) and the random walk-based estimation method (§ V) is the total number of nodes from which we need to generate random walks. Therefore, the running time of random walk generation is reduced to $\mathcal{O}\left(t\frac{\theta}{n}\sum_{v\in V}\lambda_v\right)$, and the running time of the seed selection phase is reduced to $\mathcal{O}\left(kt\frac{\theta}{n}\sum_{v\in V}\lambda_v\right)$.

For the plurality and Copeland scores, the computation of the opinion values of each user about all other candidates takes an additional $\mathcal{O}\left((r-1)tm\right)$ time. Thus, the overall time complexity is $\mathcal{O}\left(kt\frac{\theta}{n}\sum_{v\in V}\lambda_v + (r-1)tm\right)$.

B. Accuracy Guarantee for the Cumulative Score

We discuss the number of sketches (θ) required to ensure that $\widehat{F}(\widehat{B}^{(t)}[S], c_q)$ is a good estimate of $F(B^{(t)}[S], c_q)$. Let v_j denote the j^{th} sampled node, i.e., the start node of sketch j, $j \in [1, \theta]$. Denoting by $\widehat{b}^{(t)}_{qv_j}[S]$ the average of $Y^{(t)}_{qv_j}[S]$ (§ V-B) across all λ_{v_j} random walks from v_j, the estimated cumulative score is defined as:

$$\widehat{F}\left(\widehat{B}^{(t)}[S], c_q\right) = \frac{n}{\theta}\sum_{j=1}^{\theta}\widehat{b}^{(t)}_{qv_j}[S] \tag{21}$$

Let OPT be the maximum cumulative score for any size-k seed set. As shown in [35], we can derive the following.

Theorem 13. *If θ is at least*

$$\frac{2n}{OPT\cdot\epsilon^2}\left[\left(1-\frac{1}{e}\right)\sqrt{\ln\left(2n^l\right)} + \sqrt{\left(1-\frac{1}{e}\right)\left[\ln\left(2n^l\right)+\ln\binom{n}{k}\right]}\right]^2 \tag{22}$$

our algorithm returns a $(1-1/e-\epsilon)$-approximate solution S^*, with probability at least $1-n^{-l}$. More formally,

$$\Pr\left(F\left(B^{(t)}[S^*], c_q\right) \geq \left(1-\frac{1}{e}-\epsilon\right)OPT\right) \geq 1-\frac{1}{n^l} \tag{23}$$

Since the above holds for any value of λ_v, we set $\lambda_v = 1\ \forall v \in V$.[6] OPT in Equation 22 is estimated in a similar way (with similar accuracy guarantees) as *Algorithm 2* in [3].

C. Heuristic Estimation of θ for the Other Scores

While theoretical bounds on θ for the plurality and Copeland scores can be derived analogously (see the extended version [35]), we find them to be not so effective: **(1)** From the inequalities obtained in the theoretical guarantees, it is difficult to compute a closed-form expression for θ; **(2)** The sandwich approximation factor is smaller than $(1-1/e)$ (§ IV-D); coupled with the approximation via sketches, the overall approximation factor is even smaller. Instead, we use a heuristic method to compute the optimal value of θ. Note that our sketch-based method is more efficient than our random walk-based approach only when $\theta < n$. For a given dataset and score, we empirically find the smallest θ when that score converges (for some k and t). This one-time estimate of θ can be re-used on the same dataset and score, even with different number of seeds (k) and time horizon (t) as inputs, since we find such an estimate to be less sensitive to k and t. In § VIII-D, we demonstrate that the above mentioned heuristic estimation of θ produces good-quality results.

[6]Although $\lambda_v = 1$ could result in a very inaccurate estimate $\widehat{b}^{(t)}_{qv_j}$, we sample θ start nodes (not necessarily distinct) uniformly at random; thus, it is still likely that the number of walks from a particular start node is more than 1. By ensuring that θ is large enough, our overall cumulative score estimate is very accurate with a high probability.

VII. Related Work

Opinion Manipulation. [49], [50], [51] consider network modification to enable (or prevent) opinion consensus (or convergence). [52] proposes strategies for manipulating users' opinions with the voter model. Opinion maximization with the voter model is considered in [53], [54], [29]. Conformity, an opposite notion of stubbornness (used in the FJ model), measures the likelihood of a user adopting the opinions of her neighbors. Conformity-based opinion maximization has been studied in [55], [30], albeit in a *single-campaign setting*. [25], [26] study seed selection for opinion maximization in a single-campaign and without a given finite time horizon (details in [35]). To the best of our knowledge, **(a)** *we are the first to bridge two different disciplines:* **(1)** *seed selection for opinion maximization at a finite time horizon and* **(2)** *voting-based winning criteria with multiple campaigners*. Moreover, **(b)** *we are the first to design random walk and sketch-based efficient algorithms, with theoretical guarantees, for DeGroot and FJ model-based opinion maximization*.

Recall that the cumulative score, due to its aggregate nature, is independent of the other campaigns; thus it is similar to [25]. Hence, the greedy algorithm in [25], with proper modifications (e.g., adapted for a finite time horizon), would become similar to our Algorithm 1 via direct matrix-vector multiplication for the cumulative score. Regarding this score, however, we make the following *novel* contributions: **(a)** our **NP**-hardness and submodularity proofs for the cumulative score (those in [25] cannot be trivially extended to our case with any finite time horizon); **(b)** our random walk and sketch-based *efficient* algorithms, *with theoretical guarantees*, for the cumulative score (*more efficient* than the greedy algorithm in [25]).

Other Opinion Diffusion Models. Opinion diffusion has been investigated both from network science and statistical physics [56], [57] perspectives, and via discrete and continuous models. In discrete models, an individual opinion is confined to be one of several integers; examples include the voter model [58], Axelrod model [59], Sznajd model [60], majority rule models [61], [62], and social impact theory [63]. For instance, in the voter model, at each timestamp, a node chooses a random neighbor and adopts the state (i.e., preference for a certain candidate) of this neighbor. In contrast, continuous models, including DeGroot [19] (the classic model) and its extensions — FJ [20], [21], Deffuant [64], bounded confidence (BC) [65] and HK [66] models, permit opinions to be real numbers. As such, continuous models are well-suited to be integrated with voting-based winning criteria in a multi-campaign setting.

VIII. Experimental Results

We perform experiments to analyze the accuracy, efficiency, scalability, and memory usage of our methods. Our code [67] is executed on a single core, 512GB, 2.4GHz Xeon server.

TABLE III
CHARACTERISTICS OF OUR DATASETS

Name	#Nodes	#Edges	#Candidates
DBLP	63 910	2 847 120	2
Yelp	966 240	8 815 788	10
Twitter_US_Election	2 246 604	4 270 918	4
Twitter_Social_Distancing	3 244 762	4 202 083	2
Twitter_Mask	2 341 769	3 241 153	2

A. Experimental Setup

Datasets. We obtain five directed graphs from three real sources (Table III). **(1)** DBLP [68] is a well-known collaboration network. Nodes are users and edges are co-author relations. We only consider senior researchers who have published at least 50 papers. **(2)** Yelp [69] is a network of users who review businesses. Nodes are users and edges are friendships. We generate a graph based on restaurant-related records. **(3)** Twitter is a social network. Nodes are users and edges are re-tweet relationships. We generate graphs from 24M tweets (Jul. 1 to Nov. 11, 2020) related to US elections [70], and 75M tweets (Mar. 19 to Oct. 5, 2020) related to two topics ("Social distancing" and "Wear a mask") about COVID-19 [71].

Candidates. (1) DBLP. We consider the candidates for the post of President in the ACM general election 2022, i.e., Yannis E. Ioannidis and Joseph A. Konstan. **(1)** Yelp. We use the restaurant categories as candidates, e.g., American, Chinese, Italian, etc. **(2)** Twitter. The political parties (Democratic, Republican, Green, Libertarian) are the candidates in *Twitter_US_Election*. For each of the topics related to COVID-19, people may tweet for or against it. These two standpoints are the candidates in the respective Twitter COVID-19 datasets. Without loss of generality, we consider the following default target candidates for the respective datasets: "Joseph A. Konstan", "Chinese Restaurant", "Democratic Party", "For Wearing a Mask", and "For Social Distancing".

Edge Weights. Intuitively, for each category in Yelp, if user v visits a restaurant within one month of her friend u (called a common visit), we say that u influences v. Also, more common visits implies higher influence, and hence a larger edge weight. Thus, the edge (u, v) is assigned a weight of $1 - e^{-a/\mu}$ [72], where a is the number of common visits. We set $\mu = 10$ by default (details given in [35]). Similarly, we obtain edge weights (1) using the co-authorship counts for DBLP; and (2) using the number of retweets of a user pair for the Twitter datasets. Finally, we normalize the edge weights such that the incoming weights of each node add up to 1.

Initial Opinion Values. (1) DBLP. A user's initial opinion is computed as the cosine similarity between the embeddings (obtained using SpaCy [73]) of her papers to those of a candidate. **(2)** Yelp. We use the average rating of a user towards a category as the initial opinion value. **(3)** Twitter. We set the average sentiment score (computed using VADER [74]) of each user about each candidate as her initial opinion. All the initial opinion values are normalized to $[0, 1]$.

Stubbornness Values. (1) DBLP (resp. **(2)** Yelp). We set the stubbornness value of a user to 1 minus the variance of her yearly (resp. monthly) average opinions (as above), since a stubborn user is less likely to change her opinion about a candidate. **(3)** Twitter. Since most users have only 1 tweet, we assign stubborness values uniformly at random in $[0, 1]$.

TABLE IV
CASE STUDY ON THE ACM GENERAL ELECTION 2022: DOMAINS OF ACTIVITY OF THE TOP-10 SEEDS

Domain	Top-10 Seeds and their distribution across domains in which they influence the most
Data Management (DM)	Jiawei Han, Victor Leung, Philip Yu, Witold Pedrycz, Lei Zhang, Athanasios V. Vasilakos, Dusit Niyato
Human-Computer Interaction (HCI)	Yoshua Bengio, H. Vincent Poor, Lei Zhang, Dusit Niyato
Machine Learning (ML)	Yoshua Bengio, Philip Yu, Witold Pedrycz, Jiawei Han
Computer Networks (CN)	H. Vincent Poor, Dusit Niyato, Luca Benini, Victor Leung, Lei Zhang
Algorithms (AL)	Athanasios V. Vasilakos, Witold Pedrycz
Software (SW)	Luca Benini
Hardware (HW)	Luca Benini, H. Vincent Poor

Methods Compared. We find the best seed set by **(1)** Direct Matrix Multiplication (DM) via the greedy framework, coupled with CELF optimization [48]. **(2)** Random Walk Simulation (RW) and **(3)** Reverse Sketching (RS) methods are implemented for better efficiency, with accuracy guarantees. We compare them with **(4)** Independent Cascade (IC) and **(5)** Linear Threshold (LT) models-based seed selection, both coupled with IMM [3], considering only the edge weights, and assuming that a user has only one chance to accept or reject a candidate. In addition, we also compare against the **(6)** Greedy algorithm in [25] for opinion maximization, adapted for a finite time horizon, which is denoted by GED-T. Other baselines include seed selection via **(7)** PageRank score (PR) (based on the intuition that more frequently reached nodes in a random graph traversal are more likely to influence other users), **(8)** Random Walk with Restart (RWR) [25] and **(9)** Degree Centrality (DC). All baselines differ only in the seed selection methods. All of the returned seed sets are evaluated in the same multi-campaign setting with the same diffusion model and scores as in § II. We could not compare against [26] since their algorithms only work for small graphs and require more than 512GB memory on our datasets.

Parameters. (1) Seed set size (k). We vary k from 100 to 2000. In § VIII-D, k is set to 100 by default. **(2)** Time horizon (t). We vary t from 0 to 30 steps (default: 20 steps). **(3)** Random Walk Simulation. We vary ρ from 0.75 to 0.95 (default: 0.9). δ is set to 0.1. **(4)** Sketches. We vary ϵ from 0.05 to 0.3 (default: 0.1). l is set to 1 following [3].

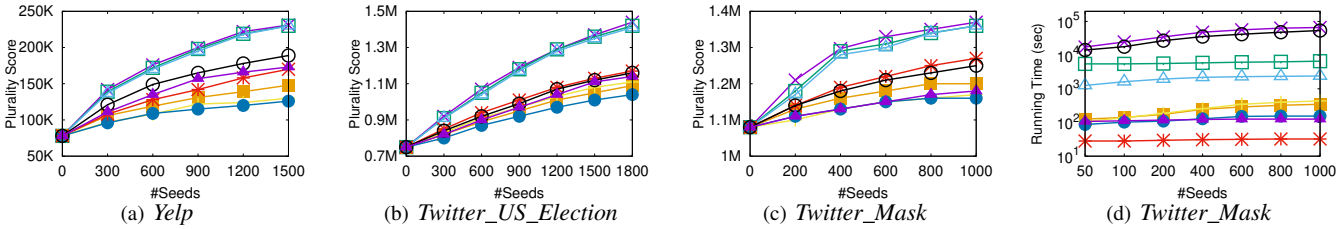
Performance Metrics. (1) Accuracy. We report the *cumulative*, *plurality*, and *Copeland* scores (§ II-B) of the seed sets returned by each method. **(2)** Efficiency. We report the running time of each method to find the best seed set.

B. Case Study: ACM General Election 2022; DBLP Dataset

We observe that after including only the top-100 seeds, the number of users favoring our target candidate *Joseph A. Konstan* will significantly increase from 13 990 (21.8%) to 46 433 (72.7%), which might have reversed the election result. We select 7 frequent domains[7] for the users who change their preferred candidates, and show the top-10 seeds and the

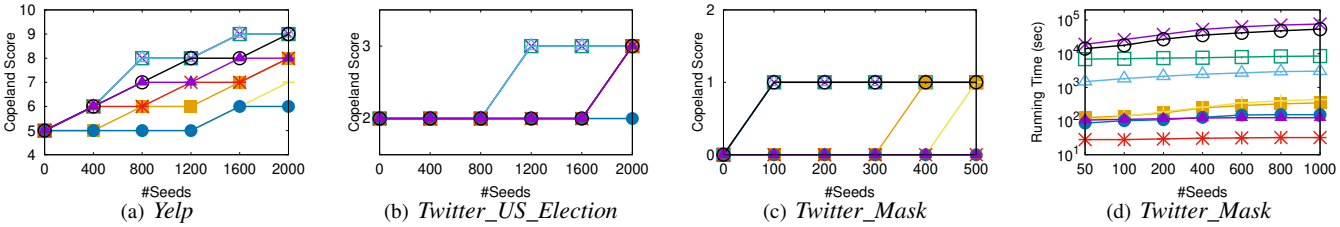
[7]We assume that a user may belong to at most 3 domains based on the frequencies of several keywords in the titles of their publications. The selected keywords for each domain can be found in our extended version [35].

Fig. 2. Legends for the methods compared in Figures 3-5



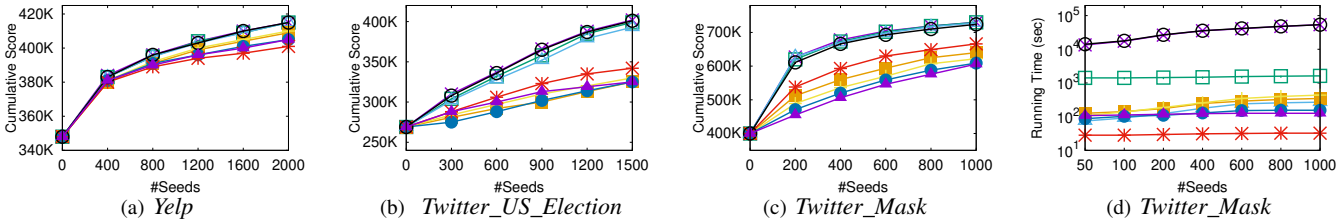
(a) *Yelp* (b) *Twitter_US_Election* (c) *Twitter_Mask* (d) *Twitter_Mask*

Fig. 3. Plurality score vs. seed set size k: (a-c) effectiveness, (d) efficiency



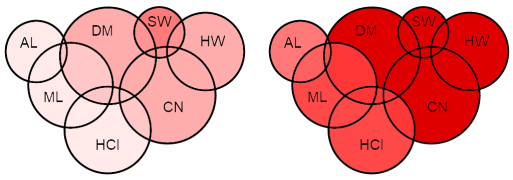
(a) *Yelp* (b) *Twitter_US_Election* (c) *Twitter_Mask* (d) *Twitter_Mask*

Fig. 4. Copeland score vs. seed set size k: (a-c) effectiveness, (d) efficiency



(a) *Yelp* (b) *Twitter_US_Election* (c) *Twitter_Mask* (d) *Twitter_Mask*

Fig. 5. Cumulative score vs. seed set size k: (a-c) effectiveness, (d) efficiency



(a) *Without Seeds* (b) *With Seeds*

Fig. 6. Case study on the ACM general election ($k = 100$, $t = 20$). The size of each circle denotes the population of users in each domain, while darker colors denote higher percentages of users who vote for the target candidate (Joseph A. Konstan).

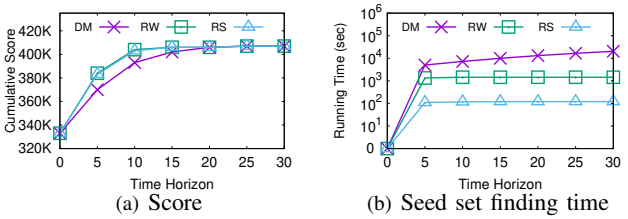
domains in which these seeds influence the most (Table IV). Figure 6 visualizes the domain overlaps and the percentage of users voting for our target candidate *Joseph A. Konstan*. Notice that a seed user may influence users from several domains. As DM is a common domain of both candidates, 7 out of the top-10 seeds are also active in the DM domain. Only 1-2 seeds are from the SW and HW domains, since (1) the users in the SW domain already favor our target candidate more based on their initial opinions (thus introducing seeds who can influence users in this domain is not that useful); (2) the HW domain does not overlap with the DM domain. The number of seeds who influence the HCI, ML, and CN domains are higher, because (1) these domains have larger populations; (2) these domains have large overlaps with DM; and (3) the users in these domains initially prefer the competitor (*Yannis E. Ioannidis*) more, thus introducing seed nodes who can influence users in these domains is more helpful. Furthermore, we investigate the average distance between the candidates and those users who change minds after introducing the seeds. 14.5% of them are closer to the target candidate, and 10.2% of them are closer to the competitors (about 2 hops away). The majority of these users (75.3%) are almost equidistant from both candidates (more than 3 hops away). This demonstrates that our solution focuses more on affecting the neutral users whose preferences are usually easier to switch.

C. Performance Analysis

Accuracy. Our proposed methods outperform the baselines in all voting-based scores (Figures 3-5 (a-c)), with the exception of our DM vs. baseline GED-T for the cumulative score. The scores increase with the number of seeds k, and the growth rates are higher when k is small. For the plurality and Copeland scores, the proposed methods outperform the baselines more significantly. For example, in *Twitter_Mask*, the best baseline DC reaches up to 70% of RW with the cumulative score, while it attains only 50% of RW with the plurality score (the actual score difference is nearly 100K users, which can lead to a significant impact in, e.g., an election's outcome). The classic IMM algorithm coupled with the IC and LT models performs poorly with voting-based scores, as does GED-T, since their seeds maximize different objective functions. Recall that GED-T is the greedy algorithm for opinion maximization [25], adapted for a finite time horizon. The cumulative score, due to its aggregate nature, is similar to opinion maximization in the single campaign setting; thus DM and GED-T perform the same for the cumulative score (*only*).

TABLE V
MINIMUM SEED SET SIZES ACHIEVED BY OUR PROPOSED METHODS FOR THE TARGET CANDIDATE TO WIN W.R.T. THE PLURALITY SCORE

Dataset	DM	RW	RS
Twitter_Mask	17	21	24
Twitter_Social_Distancing	69	71	74



(a) Score (b) Seed set finding time

Fig. 7. Cumulative score, seeds finding time vs. time horizon t; *Yelp*

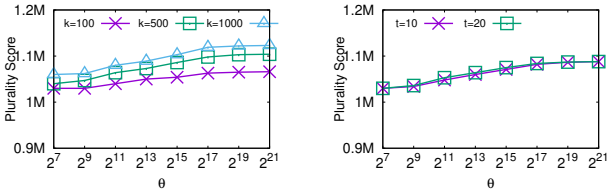
Efficiency. The running time of RW remains nearly the same for different k (Figures 3-5 (d)), while that of RS increases slightly with k. For RW, we generate a fixed number (independent of k) of random walks starting from each node (Theorem 10); while for RS, we generate walks from θ randomly sampled nodes (Theorem 13). A larger k does not necessarily increase θ as **(1)** OPT in the denominator increases with k; **(2)** $\binom{n}{k}$ in the numerator also increases with k. Moreover, the random walk generation dominates the running time of both RW and RS. The running time of DM increases linearly with k, since it applies matrix-vector multiplication in each of k iterations. The running times for the plurality and Copeland scores are higher than those of the cumulative score, but follow the same trend. Among our proposed methods, RS is the most efficient and has accuracy comparable to the others. *Therefore, we recommend* RS *as our ultimately proposed method*. Notice that RS is about two orders of magnitude faster than GED-T even for the cumulative score.

Minimum number of seeds for the target to win. As discussed in § III-C, we can adapt our methods to find the minimum number of seeds for the target to win. Table V shows these values for our three proposed methods. For a "more approximate" method, the seed sets are "less optimal", and hence the minimum number of seeds required is larger.

D. Parameter Sensitivity Analysis

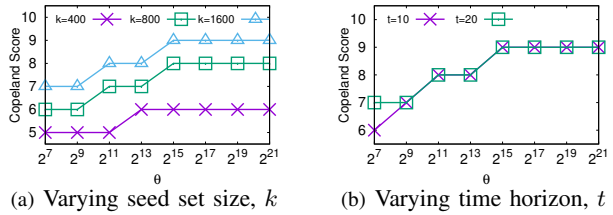
Impact of t. Figure 7 shows that the cumulative score remains nearly the same after timestamp 20 for all the proposed methods. This happens for RW and RS slightly quicker than DM. Thus, we set time horizon $t = 20$ as default in the rest of the experiments. The running time of DM is more sensitive to t than those of RW and RS, because we need to conduct exactly t rounds of matrix-vector multiplication in DM; while for RW and RS, random walks often have length less than t.

Impact of θ for the Plurality and Copeland scores. We heuristically analyze how these scores vary with θ (§ VI-C). Recall that RS is more efficient than RW only when $\theta < n$. For a specific dataset and score, we empirically find the smallest θ when that score converges (for some k and t), which is 2^{19} for *Twitter_Mask* with the plurality score (Fig. 8) and 2^{15} for *Yelp* with the Copeland score (Fig. 9). Both values are smaller than the respective n. Moreover, this estimate can be reused on the same dataset and score, even for different k and t, since it is less sensitive to k and t, as shown in Figs. 8-9.



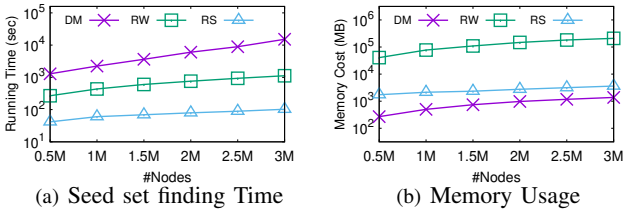
(a) Varying seed set size, k (b) Varying time horizon, t

Fig. 8. Plurality score vs. θ; *Twitter_Mask*



(a) Varying seed set size, k (b) Varying time horizon, t

Fig. 9. Copeland score vs. θ; *Yelp*



(a) Seed set finding Time (b) Memory Usage

Fig. 10. Seed set finding time and memory usage for the cumulative score vs. graph size; *Twitter_Social_Distancing*

E. Scalability and Memory Usage

We test the scalability and memory usage of our algorithms with different graph sizes. The *Twitter_Social_Distancing* graph has about 3.2M nodes; we generate six graphs by selecting 0.5M, 1M, 1.5M, 1M, 2.5M, 3M nodes uniformly at random, and apply our algorithms on the subgraphs induced by them. Figure 10(a) demonstrates that the running times of RW and RS increase almost linearly with the number of nodes (the y-axis is logarithmic), which confirms good scalability of our algorithms. The running time of DM increases polynomially – it has cubic growth with n (§ III-C).

DM consumes the least memory (Figure 10(b)) since it only stores the edge weights, initial opinions, and stubbornness values. RW and RS further store random walks (RW far more than RS). Our ultimately proposed method, RS, consumes only a few GB for the *Twitter_Social_Distancing* dataset.

IX. CONCLUSIONS

We formulated and investigated the novel problem of opinion maximization in a social network, coupled with voting-based scores. We proved that our problem is **NP**-hard and non-submodular. To solve the problem, we employed the well-known Sandwich Approximation, under which we proved that the greedy algorithm can still provide approximation guarantees to our objectives. Since exact opinion computation via iterative matrix-vector multiplications is inefficient, we proposed random walk and sketching-based opinion computations, with theoretical approximation guarantees. Experimental results validated the effectiveness and efficiency of our proposed algorithms. Considering both accuracy and efficiency results, we recommend the sketching-based approach RS as our ultimately proposed method. In the future, we shall consider more opinion diffusion models and voting scores.

REFERENCES

[1] A. Khan, Y. Ye, and L. Chen, *On Uncertain Graphs*, ser. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2018.

[2] S. Galhotra, A. Arora, and S. Roy, "Holistic influence maximization: Combining scalability and efficiency with opinion-aware models," in *Proceedings of the 2016 ACM SIGMOD International Conference on Management of Data*, 2016, p. 743–758.

[3] Y. Tang, Y. Shi, and X. Xiao, "Influence maximization in near-linear time: A martingale approach," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015, p. 1539–1554.

[4] A. Arora, S. Galhotra, and S. Ranu, "Debunking the myths of influence maximization: An in-depth benchmarking study," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, p. 651–666.

[5] J. Tang, K. Huang, X. Xiao, L. V. Lakshmanan, X. Tang, A. Sun, and A. Lim, "Efficient approximation algorithms for adaptive seed minimization," in *Proceedings of the 2019 ACM SIGMOD International Conference on Management of Data*, 2019, p. 1096–1113.

[6] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "A data-based approach to social influence maximization," *Proc. VLDB Endow.*, vol. 5, no. 1, p. 73–84, 2011.

[7] Y. Tang, X. Xiao, and Y. Shi, "Influence maximization: Near-optimal time complexity meets practical efficiency," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 2014, p. 75–86.

[8] J. Tang, X. Tang, X. Xiao, and J. Yuan, "Online processing algorithms for influence maximization," in *Proceedings of the 2018 ACM SIGMOD International Conference on Management of Data*. New York, NY, USA: Association for Computing Machinery, 2018, p. 991–1005.

[9] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, p. 137–146.

[10] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001, p. 57–66.

[11] S. Bharathi, D. Kempe, and M. Salek, "Competitive influence maximization in social networks," in *International Workshop on Web and Internet Economics*. Berlin, Heidelberg: Springer, 2007, pp. 306–311.

[12] T. Carnes, C. Nagarajan, S. M. Wild, and A. van Zuylen, "Maximizing influence in a competitive social network: A follower's perspective," in *Proceedings of the 9th International Conference on Electronic Commerce*, 2007, p. 351–360.

[13] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model," in *Proceedings of the 2012 SIAM International Conference on Data Mining*, 2012, pp. 463–474.

[14] Y. Lin and J. C. Lui, "Analyzing competitive influence maximization problems with partial information: An approximation algorithmic framework," *Performance Evaluation*, vol. 91, pp. 187–204, 2015.

[15] M. Kahr, M. Leitner, M. Ruthmair, and M. Sinnl, "Benders decomposition for competitive influence maximization in (social) networks," *Omega*, vol. 100, p. 102264, 2021.

[16] C. Budak, D. Agrawal, and A. El Abbadi, "Limiting the spread of misinformation in social networks," in *Proceedings of the 20th International Conference on World Wide Web*, 2011, p. 665–674.

[17] W. Lu, F. Bonchi, A. Goyal, and L. V. Lakshmanan, "The bang for the buck: Fair competitive viral marketing from the host perspective," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 928–936.

[18] A. Khan, B. Zehnder, and D. Kossmann, "Revenue maximization by viral marketing: A social network host's perspective," in *2016 IEEE 32nd International Conference on Data Engineering*, 2016, pp. 37–48.

[19] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.

[20] N. E. Friedkin and E. C. Johnsen, "Social influence and opinions," *Journal of Mathematical Sociology*, vol. 15, no. 3-4, pp. 193–206, 1990.

[21] N. E. Friedkin and E. C. Johnsen, "Social influence networks and opinion change," *Advances in Group Processes*, vol. 16, no. 1, pp. 1–29, 1999.

[22] E. Pacuit, "Voting methods," in *The Stanford Encyclopedia of Philosophy*. Stanford, CA, USA: Metaphysics Research Lab, Stanford University, 2019.

[23] W. Gaertner, *A primer in social choice theory*. Oxford, UK: Oxford University Press, 2006.

[24] P. C. Fishburn, "Paradoxes of voting," *The American Political Science Review*, vol. 68, no. 2, pp. 537–546, 1974.

[25] A. Gionis, E. Terzi, and P. Tsaparas, "Opinion maximization in social networks," in *Proceedings of the 2013 SIAM International Conference on Data Mining*, 2013, pp. 387–395.

[26] R. Abebe, J. Kleinberg, D. Parkes, and C. E. Tsourakakis, "Opinion dynamics with varying susceptibility to persuasion," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2018, p. 1089–1098.

[27] B. S. Frey, "Direct democracy: politico-economic lessons from swiss experience," *The American Economic Review*, vol. 84, no. 2, pp. 338–342, 1994.

[28] P. Emerson, *Decision-making in parliaments and referendums*. Springer International Publishing, 2020, pp. 3–30.

[29] Y. Li, W. Chen, Y. Wang, and Z.-L. Zhang, "Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships," in *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*, 2013, p. 657–666.

[30] H. Li, S. S. Bhowmick, A. Sun, and J. Cui, "Conformity-aware influence maximization in online social networks," *VLDB J.*, vol. 24, no. 1, pp. 117–141, 2015.

[31] W. Lu, W. Chen, and L. V. Lakshmanan, "From competition to complementarity: Comparative influence diffusion and maximization," *Proc. VLDB Endow.*, vol. 9, no. 2, p. 60–71, 2015.

[32] K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova, "Monte carlo methods in pagerank computation: When one iteration is sufficient," *SIAM Journal on Numerical Analysis*, vol. 45, no. 2, pp. 890–904, 2007.

[33] E. Cohen and D. D. Lewis, "Approximating matrix multiplication for pattern recognition tasks," *Journal of Algorithms*, vol. 30, no. 2, pp. 211–252, 1999.

[34] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier, "Maximizing social influence in nearly optimal time," in *Proceedings of the 2014 ACM-SIAM Symposium on Discrete Algorithms*, 2014, pp. 946–957.

[35] A. Saha, X. Ke, A. Khan, and L. V. Lakshmanan, "Voting-based opinion maximization," 2022. [Online]. Available: https://arxiv.org/abs/2209.06756

[36] H. Noorazar, "Recent advances in opinion propagation dynamics: A 2020 survey," *The European Physical Journal Plus*, vol. 135, no. 6, pp. 1–20, 2020.

[37] H. Z. Brooks and M. A. Porter, "A model for the influence of media on the ideology of content in online social networks," *Physical Review Research*, vol. 2, p. 023041, 2020.

[38] S. Chen, J. Fan, G. Li, J. Feng, K.-l. Tan, and J. Tang, "Online topic-aware influence maximization," *Proc. VLDB Endow.*, vol. 8, no. 6, p. 666–677, 2015.

[39] S. Tu, Ç. Aslay, and A. Gionis, "Co-exposure maximization in online social networks," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2020.

[40] J. Guo, T. Chen, and W. Wu, "A multi-feature diffusion model: rumor blocking in social networks," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 386–397, 2021.

[41] K. Garimella, A. Gionis, N. Parotsidis, and N. Tatti, "Balancing information exposure in social networks," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 2017, pp. 4663–4671.

[42] X. Ke, A. Khan, and G. Cong, "Finding seeds and relevant tags jointly: For targeted influence maximization in social networks," in *Proceedings of the 2018 ACM SIGMOD International Conference on Management of Data*, 2018, p. 1097–1111.

[43] P. C. Fishburn, "Condorcet social choice functions," *SIAM Journal on Applied Mathematics*, vol. 33, no. 3, pp. 469–489, 1977.

[44] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*. Boston, MA, USA: Springer, 1972, pp. 85–103.

[45] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions — i," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[46] A. A. Bian, J. M. Buhmann, A. Krause, and S. Tschiatschek, "Guarantees for greedy maximization of non-submodular functions with applications," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 498–507.

[47] A. Das and D. Kempe, "Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection," *Journal of Machine Learning Research*, vol. 19, pp. 3:1–3:34, 2018.

[48] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007, p. 420–429.

[49] Y. Dong, Z. Ding, L. Martínez, and F. Herrera, "Managing consensus based on leadership in opinion dynamics," *Information Sciences*, vol. 397-398, pp. 187–205, 2017.

[50] M. Pineda and G. M. Buendía, "Mass media and heterogeneous bounds of confidence in continuous opinion dynamics," *Physica A: Statistical Mechanics and its Applications*, vol. 420, pp. 73–84, 2015.

[51] D. Bauso and M. Cannon, "Consensus in opinion dynamics as a repeated game," *Automatica*, vol. 90, pp. 204–211, 2018.

[52] A. Gupta, S. Moharir, and N. Sahasrabudhe, "Influencing opinion dynamics in networks with limited interaction," arXiv:2002.00664, 2020.

[53] G. Romero Moreno, E. Manino, L. Tran-Thanh, and M. Brede, *Zealotry and influence maximization in the voter model: When to target partial zealots?* Cham: Springer, 2020.

[54] K. Rawal and A. Khan, "Maximizing contrasting opinions in signed social networks," in *2019 IEEE International Conference on Big Data*, 2019, pp. 1203–1210.

[55] A. Das, S. Gollapudi, A. Khan, and R. P. Leme, "Role of conformity in opinion dynamics in social networks," in *Proceedings of the second ACM conference on Online social networks*, 2014, pp. 25–36.

[56] W. Weidlich, "The statistical description of polarization phenomena in society," *British Journal of Mathematical and Statistical Psychology*, vol. 24, pp. 251–266, 1971.

[57] S. Galam, Y. Gefen, and Y. Shapir, "Sociophysics: a new approach of sociological collective behaviour. i. mean-behaviour description of a strike," *Journal of Mathematical Sociology*, vol. 9, pp. 1–13, 1982.

[58] R. A. Holley and T. M. Liggett, "Ergodic theorems for weakly interacting infinite systems and the voter model," *The Annals of Probability*, vol. 3, no. 4, pp. 643–663, 1975.

[59] R. Axelrod, "The dissemination of culture: A model with local convergence and global polarization," *Journal of Conflict Resolution*, vol. 41, no. 2, pp. 203–226, 1997.

[60] K. Sznajd-Weron and J. Sznajd, "Opinion evolution in closed community," *International Journal of Modern Physics C*, vol. 11, p. 1157, 2000.

[61] P. L. Krapivsky and S. Redner, "Dynamics of majority rule in two-state interacting spin systems," *Physical Review Letters*, vol. 90, p. 238701, 2003.

[62] R. Lambiotte, "Majority rule on heterogeneous networks," *Journal of Physics A: Mathematical and Theoretical*, vol. 41, no. 22, p. 224021, 2008.

[63] C. M. Bordogna and E. V. Albano, "Statistical methods applied to the study of opinion formation models: A brief overview and results of a numerical study of a model based on the social impact theory," *Journal of Physics: Condensed Matter*, vol. 19, no. 6, p. 065144, 2007.

[64] G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch, "Mixing beliefs among interacting agents," *Advanced Complex System*, vol. 3, pp. 87–98, 2000.

[65] G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch, "Mixing belief among interacting agents," *Advances in Complex Systems*, vol. 03, no. 01n04, pp. 87–98, 2000.

[66] R. Hegselmann and U. Krause, "Opinion dynamics and bounded confidence: Models, analysis and simulation," *Journal of Artificial Societies and Social Simulation*, vol. 5, no. 3, 2002.

[67] A. Saha, X. Ke, A. Khan, and L. V. Lakshmanan, "Voting-based opinion maximization: Code and data," 2022. [Online]. Available: https://github.com/ArkaSaha/Opinion-Vote

[68] The dblp team: dblp computer science bibliography, "Monthly snapshot release of july 2022," 1993. [Online]. Available: http://dblp.uni-trier.de/xml

[69] Yelp Inc, "The yelp dataset," 2004. [Online]. Available: https://www.yelp.com/dataset

[70] I. Sabuncu, "Usa nov. 2020 election 20 million tweets (with sentiment and party name labels) dataset," 2020.

[71] R. Lamsal, "Coronavirus (covid-19) tweets dataset," 2020.

[72] M. Potamias, F. Bonchi, A. Gionis, and G. Kollios, "K-nearest neighbors in uncertain graphs," *Proc. VLDB Endow.*, vol. 3, no. 1–2, p. 997–1008, 2010.

[73] Y. Vasiliev, *Natural Language Processing with Python and SpaCy: A Practical Introduction*. No Starch Press, 2020.

[74] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Proceedings of the AAAI Conference on Web and Social Media*, vol. 8, 2014, pp. 216–225.