Fairness and Robustness in Answering Preference Queries

Senjuti Basu Roy
New Jersey Institute of Technology
senjutib@njit.edu

Abstract

Given a large number of users preferences as inputs over a large number of items, preference queries leverage different preference aggregation methods to aggregate individual preferences in a systematic manner and come up with a single output (top-k ordered or unordered/a complete order) that is most representative. The preference aggregation methods are widely adopted from the social choice theory, some of which are rank based (single-round vs. multi-round), while others are non-rank based. These queries are prevalent in high fidelity applications, including search, ranking and recommendation, hiring and admission, and electoral voting systems. This article outlines algorithmic challenges and directions in designing an optimization guided computational framework that allows to change the original aggregated output (either ordered or unordered top-k or a complete order) to satisfy different criteria related to fairness and robustness, considering different preference elicitation models (ways users provide their input preferences) and aggregation methods (ways the individual preference get aggregated).

1 Introduction

The need to aggregate a large number of individual preferences in a systematic manner is ubiquitous. Users can provide preferences in many ways - as likes/dislikes, ordinal preferences, or ranked order (full or partial). The social choice theory [17] offers a plethora of aggregation methods to aggregate individual preferences and come up with a single output. These outputs may be a single rank that is most representative of all users preferences, or sometimes a smaller number of k items (top-k) that are ordered or presented as a set. While designed for electoral voting systems primarily, the applicability of answering queries is prevalent in many high fidelity applications, such as, ranking and listing web search results, recommending movies/songs, selecting a handful of candidates for domains where resource is scarce (such as hiring and admission), to name a few. It is not a stretch to consider a setting in which thousands of items (notationally n) have received preferences from hundreds of thousands (or even millions) of users (notationally m) and the goal is to produce a single output (notationally σ) that is most representative.

The computational implications of different preference aggregation methods are well studied. What is not so well understood is how hard it is to change the original produced output, which may be necessary for many compelling reasons. Satisfying additional criteria, such as, promoting fairness (e.g., ensuring presence of individuals with certain socio-demographic properties), or Understanding robustness, i.e., figuring out the minimum amount of change of the inputs that would result in a different outcome than the original output. This latter aspect provides understanding on how manipulable the proposed aggregation methods are which

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

are certainly important for aggregation methods that are heavily used in electoral systems, but are applicable in other scenarios as well (e.g., in figuring out the robustness of a rating system of products). To the best of our knowledge, a systematic study is needed to investigate these aspects in conjunction with different preference elicitation models, requiring different preference aggregation methods. That, in nutshell, is the focus of this article.

We discuss these challenges considering four interspersed dimensions, as described below.

Preference Elicitation Models. The article simultaneously considers a vast range of preference elicitation processes that we broadly categorize as rank based and non rank based. In rank based processes, the users can provide a fully ranked order over all items, a partial order, or a coarser preference (like item a ranked higher than item b, etc). In non rank based preferences, users can provide only likes, both likes and dislikes, or even an ordinal preference (likes item a as "excellent", b as "good", etc). The choice of these preference elicitation methods is dictated by the different applications. Rank based ones are suitable in hiring/admission/electoral system, while non rank based ones are more relevant in obtaining user feedback from search results, user satisfaction survey, product reviews, etc.

Preference Aggregation Methods. Then the preference aggregation methods that are most commensurate to the underlying preference elicitation process and underlying application are studied. For example, when user preferences are given as ranked order, depending on the underlying application, we will aggregate them using existing single-round rank based methods (e.g., Kemeny, Spearman's footrule, or Borda), or multi-round based methods (STV, IRV). The former aggregation methods are suitable in hiring decision, whereas, the latter ones are gaining popularity in voting systems. On the other hand, when users provide non rank based preferences, we will show how Jaccard similarity or Hamming distances are suitable to aggregate them and come up with the final output.

Produced Output Form. From the application point of view, the produced output may require an order over all n items (hiring/admission), or a small number k of n items as outputs. In case of top-k items requirement, the returned k-items may need to be ordered for certain applications (top-k web pages returned by the search engine), or in some cases it is fine to return them as a set (selecting a set of representatives or body to form certain committee).

Change Original Output. The importance of quantifying the minimum effort needed to change the original output is evident for several reasons, such as promoting fairness and robustness. Robustness is heavily used in electoral system to produce margin, that investigates how to bound the amount of change of the original outcome in case x% of the inputs are destroyed/deleted/modified. We discuss them further in details below.

2 Overarching Research Goals

The overarching goal is to design optimization guided computational framework containing principled models and scalable solutions that allows to change the original aggregated output (either ordered or unordered top-k or a complete order) to satisfy different criteria, considering different preference elicitation models and aggregation functions to promote: **a. Fairness** from the standpoint of the protected attributes[27] of the items/candidates (e.g., race, gender, ethnicity), where the candidates are selected by aggregating elicited preferences of the members (panelists, voters, search committee). We shall investigate existing group fairness criteria in the context of preference aggregation [27, 29], as well as adapt fairness criteria studied in the context of resource allocation or social choice theory. **b. Robustness**, namely, understanding how easy or hard it is to change the original outcome of different preference aggregation models given a budgeted preference substitution requirement. For instance, if the total number of preference updates is budgeted to be $\leq x$, is it possible to change the original outcome? We are interested in exploring these viewpoints for multiple preference elicitation models and output forms. What is also important to notice is that a given preference elicitation may be suitable to multiple aggregation methods and may require to satisfy more than one produced output form. These gives rise to many combinations of the

problem.

The rest of the article is organized as follows: In Section 4, we study how to Satisfy output constraints in single round rank-based preference aggregation methods. We study this considering ranking, which is a commonly used method to prioritize desirable outcomes among a set of items/candidates and is an essential step in many high impact applications. Here the members elicit a complete or partial preference order over the candidates and the goal is to produce an aggregated ranked order over all candidates or produce top-k results that minimize disagreements among individual preferences. We will also include preference substitution in single round rank-based preference aggregation methods to satisfy complex top-k constraints, where the requirement is defined over a set R of protected attributes.

In Section 5, we study how to satisfy output constraints in multi round rank-based preference aggregation methods, popularly known as ranked choice voting or (RCV) [12]. Two popular representatives of these models are IRV (Instant run-off voting) [12] that selects one item/candidate as the winner, and STV (single transferable vote) [10, 13] that generalizes IRV and selects a set of k-items/candidates as winners. It is known that RCV represents majority rules and improves result diversity. Unlike single round preference aggregation models, RCV minimizes the effect of strategic voting as users can provide their "true preference" for the candidates they support, not just provide preference against the items/candidates they oppose most. It is also shown in recent works, how RCV promotes anonymity and anti-plurality [13], compared to single round based algorithms.

In Section 6, we will study how to satisfy output constraints in non rank based preference aggregation methods. Here we investigate preference aggregation methods that do not require users inputs to be ranked order. A simple case in this context is a Boolean model, where each user describes their preference over n items as a Boolean vector of 0 and 1. When users provide only their "likes" on the items, the aggregation function such as Jaccard Similarity or Overlap similarity [24] may be appropriate to find top-k items that have exhibited maximum similarity over the users preferences. On the other hand, when the users provide both "likes" and "dislikes", the aggregation function may intend to produce a Boolean vector that minimizes the Hamming Distance between the input preferences and the produced output. Generalization of the Boolean preference elicitation models is also discussed.

Comparison with Existing Work. This contribution builds on our work recent works on fairness [20, 28], and prior works on preference aggregation [2, 3, 8], studying robustness [24]. We acknowledge that the existing popular group based fairness definition, such as, statistical parity [15] is somewhat similar to one of our proposed fairness notion. However, the best adapted version of top-k statistical parity studied in a recent paper [21] does not account for proportionate representation in every position of the top-k, limiting its applicability. Studying computational challenges related to computing the margin of victory has been a focus of recent research [4, 6, 10] in the context of electoral voting and related applications. But none of these existing works study the general version of the problem, which is, how to promote additional simple/complex constraints/criteria in the output, which is our primary focus. Other than these prior works, which are much narrow in scope, we are unaware of any computational work that systematically studies different preference elicitation models, multiple output changing criteria, and preference aggregation combining these two.

3 Formalism

There are 4 types of inputs that our proposed framework takes: (a) a set N of n items, where each item has a set $\mathcal{A}$ of discrete attributes. Each attribute $a \in \mathcal{A}$ has ℓ_a different values. (b) a set of m users, where the i-th user $u(i)$ provides her preference as σ_i. The users' preferences could be rank based, partial or full order, or non rank based. (c) a distance function $\mathcal{F}$ (defined formally below) that measure the "distance" between a set of m input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ and an output σ with the required output form. The exact distance function depends on the underlying preference elicitation model and the required output form which may be either a complete ranking of the items or a subset of k items, either ranked or not. (d) a set $\mathcal{C}$ of output criteria/constraints. Some variants of our problem also include as input a budgetary constraint B.

Definition 3.1: Distance function $\mathcal{F}$. Given m input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ and an output σ with the required output form, the function $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m)$ is the distance of σ from the input preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$. In some cases the function $\mathcal{F}(\cdot)$ is an aggregation of a distance function between a single input preference and the output. Examples for such an aggregation are the sum of the pairwise distances and the maximum distance to any of the input preferences. In other cases $\mathcal{F}(\cdot)$ measures the minimum modification of the input preferences that would result in the preference aggregation method outputting the output σ.

Definition 3.2: Output criteria/constraints. For an attribute $a \in \mathcal{A}$, let $c(p_a)$ denote the cardinality constraints of items with value p_a (p_a is one of the ℓ_a possible values of attribute a). Given to the framework is a set C of such cardinality constraints for each attribute value p_a, for every $a \in A$, $A \subset \mathcal{A}$. There are two explicit cases that we consider.

- **The output σ is ordered and consists of $k \leq n$ items.** In this case the cardinality constraints are defined for every $\kappa \in [1..k]$ items, and for every such $\kappa \in [1..k]$, the κ top ranked items of output σ have to satisfy these cardinality constraints.
- **The output σ is an unordered set of k items.** In this case the cardinality constraints are defined for k items and the items in the output set σ have to satisfy these cardinality constraints.

Definition 3.3: A budgetary constraint. A budgetary constraint B is an upper bound on the distance of the output from the input preferences. The budgetary constraint implies that $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m) \leq B$.

Definition 3.4: Preference Aggregation Considering Constraints. We intend to study different types of problem definitions that require different algorithmic treatments. Given either complete or partial preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ over the items in N, a preference aggregation method, a distance function $\mathcal{F}(\cdot)$, and a set of output criteria $\mathcal{C}$.

- **(Constrained optimization).** Produce an output σ with the required form that minimizes $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m)$ and satisfies $\mathcal{C}$.
- **(Optimization under budgetary constraints).** Produce an output σ with the required form that optimizes $\mathcal{C}$, while satisfying $\mathcal{F}(\sigma, \sigma_1, \sigma_2, \ldots, \sigma_m) \leq B$. (The objective function for optimizing $\mathcal{C}$ varies.)
- **(Bi-criteria optimization).** Given parameters α and β produce an output σ with the required form that satisfies both $\mathcal{F}(\sigma, \sigma_1, \ldots, \sigma_m) \leq \alpha$ and $\mathcal{G}(\mathcal{C}) \leq \beta$, where $\mathcal{G}$ is the objective function for optimizing $\mathcal{C}$.

3.1 Specifying Output Criteria

We discuss orthogonal reasons where the original outputs coming out of the preference aggregation methods need to be "massaged" further. What unifies them is that these criteria are defined over one or more attributes of the items. Depending on how many attributes are involved in the definition and their relationship thereof gives rise to additional challenges.

3.1.1 Fair Preference Aggregation

We will study fairness in the context of group based protected attributes of the candidates. Output criteria/constraints for fairness (refer to Definition 3.2) are expressed over one or more protected attributes. Their protected attributes could be expressed over gender, ethnicity, race, or the state the candidates are living in.

Formally speaking, each item/candidate $v \in N$ has one or more protected attributes. When $\ell_a = 2$, it is a binary protected attribute; when $\ell_a \geq 2$ it is a multi-valued protected attribute. As an example, race is (usually) a multi-valued protected attribute, and gender is sometimes a binary protected attribute.

p-fairness. p-fairness has been studied in the context of resource allocation satisfying temporal fairness or proportionate progress [7, 25]. It was introduced in the classical Chairman Assignment Problem [5, 25] that studies how to select a chairman of an union every year from a set of n states such that that at any time the accumulated number of chairmen from each state is proportional to its weight.

In the context of ranking, suppose that each of the n ranked items has a protected attribute $a(\cdot)$ that can take any of ℓ_a different values. For $p_a \in [1..\ell_a]$, let $c'(p_a)$ denote the fraction of items with protected attribute value p_a, that is, $c'(p_a) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{a(i)=p_a}$. The goal is to ensure that $c'(p_a)$ fraction (rounded either up or down) of every top κ items have protected attribute value p_a.

3.1.2 Robust Preference Aggregation

Output criteria/constraints for robustness on the other hand investigates the flip questions: Given either complete or partial preferences $\sigma_1, \sigma_2, \ldots, \sigma_m$ over n items, let σ be the output obtained by the preference aggregation method. Given a budget B, how to make B or less changes in the original preferences, such that the outcome is different from σ? This question is related to finding the margin in electoral systems and quantifies how manipulable the underlying aggregation method is. We study this problem under different manipulation models – addition only, deletion only, or substitution (addition + deletion).

4 Single Round Rank based Preference Aggregation

We outline two separate lines of algorithmic problems: (1) incorporating output criteria (e.g., p-fairness) in single round rank-based preference aggregation methods, and (2) satisfying complex constraints in single round rank-based preference aggregation methods.

4.1 Incorporating output criteria in rank aggregation

The input to the classical rank aggregation problem consists of m complete order of preferences over the n items/candidates. Traditionally, producing the final ranking involves aggregating potentially conflicting preferences from multiple individuals, and is known as the rank aggregation problem [1, 16, 26]. Our goal is to minimally change the aggregated output to enable fairness. We will study p-fairness [7, 25] that ensures proportionate representation of every group based on a protected attribute in every position of the aggregated ranked order. The classical problem in this context is known as the Chairman Assignment Problem [5, 25] which studies how to select a chairman of a union every year from a set of r states such that that at any time the accumulated number of chairmen from each state is proportional to its weight. p-fairness generalizes other notions of fairness [19] that were considered in prior work, including the existing popular group based fairness definition statistical parity [15].

4.1.1 Research Directions

Consider rankings of the items in a set V. Each such ranking can be viewed as a permutation. We will use the terms ranking and permutation interchangeably.

Kendall-Tau and Kemeny distances. Given two rankings $\sigma, \eta : V \to [1..n]$, the Kendall-Tau distance between the two rankings is the sum of pairwise disagreements between σ and η (bubble-sort distance)

$$\mathcal{K}(\sigma, \eta) = \sum_{\{u,v\} \subseteq V} \mathbb{1}_{(\sigma(v)-\sigma(u))(\eta(v)-\eta(u))<0}.$$

For a set of rankings $\{\eta_1, \eta_2, \ldots, \eta_m\}$ the Kemeny distance of the ranking σ to this set as

$$\kappa(\sigma, \eta_1, \eta_2, \ldots, \eta_m) = \sum_{i=1}^{m} \mathcal{K}(\sigma, \eta_i).$$

Spearman's footrule distance. Given two rankings $\sigma, \eta : V \to [1..n]$, the Spearman's footrule distance between the two rankings is the sum of the absolute values (ℓ_1 distance) of the differences between rankings σ and η.

$$\mathcal{S}(\sigma, \eta) = \sum_{u \in V} |(\sigma(u) - \eta(u)|$$

For a set of rankings $\{\eta_1, \eta_2, \ldots, \eta_m\}$ the Spearman's footrule distance of the ranking σ to this set is the sum of the pairwise distances.

Rank aggregation. The aggregated ranking of a set of m rankings $\{\rho_1, \rho_2, \ldots, \rho_m\}$ for a given distance function is a ranking that minimizes the distance to this set.

p-fairness for a ranking. For a permutation $\sigma, k \in [1..n], p \in [1..\ell]$, let $P(\sigma, k, p)$ denote the number of elements with protected attribute value p among the k top ranked elements in σ. A ranking σ is proportionate fair or p-fair if

$$\forall k \in [1..n] \, \forall p \in [1..\ell] : \; P(\sigma, k, p) \in \{\lfloor f(p) \cdot k \rfloor, \lceil f(p) \cdot k \rceil\}.$$

We formalized two optimization problems, individual p-fairness or **IPF** and the rank aggregation problem subject to proportionate fairness (**RAPF**) considering binary ($\ell = 2$) and multi-valued ($\ell > 2$) protected attributes. These problems and associated algorithmic results could be found in [28].

4.1.2 Open Problems

We plan to investigate the following open problems.

p-fairest aggregate ranking (PFAR). The PFAR problem is defined as follows. Given a set of m rankings choose the "p-fairest" ranking among all rankings that minimize the Kemeny distance to this set. We need to define "p-fairest" ranking or a distance measure to a p-fair ranking. We propose the following distance measure (using the notations defined above). For an integer $d \geq 0$, a ranking σ is at distance d from a p-fair ranking if

$$\forall k \in [1..n] \, \forall p \in [1..\ell] : \; P(\sigma, k, p) \in \{\lfloor f(p) \cdot k \rfloor - d, \lceil f(p) \cdot k \rceil + d\}.$$

We observe that PFAR is also NP-Hard as directly follows from the fact that unconstrained rank aggregation is NP-hard when $m \geq 4$ [1]. For some fixed $\alpha > 1$, We would like to find an algorithm that finds the p-fairest ranking among all rankings whose Kemeny distance from the set of input rankings is at most α times the minimum such distance.

Bi-criteria p-fair rank aggregation (BPFRA). The most general problem that we plan to consider in this context is the bi-criteria optimization problem, that is, for a given pair $(\alpha > 1, \beta > 1)$ and a set of m rankings find a ranking whose Kemeny distance to the set of rankings is at most α times the Kemeny distance of the aggregated rank from the set and its distance from a p-fair ranking is at most β, if such a ranking exists.

p-fair rank aggregation with affirmative action. We plan to consider a variant of p-fair rank aggregation that involves "affirmative action". This will be modeled by varying the proportion of the values of the protected attribute in the p-fair aggregated rank. For example, consider a binary protected attribute with values A and B each needs to appear the same number of times. Suppose that our goal is to promote the items with attribute value A. In this case we can vary the proportion of A making it higher in the top ranked elements and lower in the lower ranked elements so that overall items with attribute value A will appear the same number of times as items with attribute value B.

The complexity of individual p-fair ranking (IPF). We plan to further investigate the IPF problem for multi valued protected attributes as it is open whether it can be solved accurately in polynomial time. We conjecture that this problem is NP-Hard. We also plan to look for improved approximation algorithms for this problem.

Better approximation of rank aggregation subject to p-fairness (RAPF). We plan to develop more sophisticated RAPF algorithms with better approximation ratios, and to improve the computational aspects of the RAPF

problem. This problem can be formulated as an Integer Programming (IP) problem. We plan to consider various IP formulations as well as various rounding techniques to accelerate the computation.

Robust rank aggregation. It is known that rank aggregation under Kemeny distance is NP-hard. We will explore other aggregation methods, such as Spearman's footrule and Borda, and study how manipulable these rank aggregation methods are – that is, if only $x\%$ of the preferences are allowed to be changed, how easy it is to change the outcome.

4.2 Complex Constraints

Our goal is to optimize preference substitution to satisfy complex top-k fairness constraints, where the fairness requirement is defined over a set R of protected attributes. One of the objectives we will consider is minimizing the number of single ballot (ranking) substitutions that guarantee fairness in the top-k results. In a preliminary work we defined the problem of finding the smallest number of single ballot substitutions to promote a set of k candidates that satisfy fairness requirements defined over a set R of protected attributes to the top-k.

4.2.1 Research Directions

We assume that there are ℓ protected attributes , denoted $A_1, \ldots, A_\ell$. For $i \in [1..\ell]$, attribute A_i has ℓ_i possible values, denoted $A[i, j]$, for $j \in [1..\ell_i]$. Each candidate is associated with a specific value from each attribute. In addition, we are given target quantities $a[i, j]$, for $i \in [1..\ell]$, and $j \in [1..\ell_i]$, with property that all row marginals sum to k. Namely, for every $i \in [1..\ell]$, $\sum_{j=1}^{\ell_i} a[i, j] = k$. A fair outcome should satisfy the fairness condition that for $i \in [1..\ell]$, and $j \in [1..\ell_i]$, exactly $a[i, j]$ candidates whose A_i attribute value is $A[i, j]$ are elected.

We note that one way to approach this problem is by converting the multiple protected attributes to a single multi-valued protected attribute by computing joint distribution over the attributes assuming their independence. For example, instead of considering two binary valued attributes A_1 and A_2 we consider a single attribute with 4 possible values and the requirement that the value $i * j$ should appear $a[1, i] \cdot a[2, j]/k$ times, for $i, j \in \{1, 2\}$. The shortcomings of this approach are two-fold: First, this approach may yield that the problem is infeasible while there is still a solution without assuming independence. A solution that assumes independence may be inferior (require more substitutions) than a solution that does not assume independence.

In [20], we showed that the problem of finding the smallest number of single ballot substitutions (original preference) to promote a set of k candidates that satisfy proportionate representation over a single protected attribute is computationally easy for any domain size of the protected attribute. On the other hand the same problem becomes computationally hard if we increase the number of protected attributes. When there are two different protected attributes involved in outlining the fairness requirement, we proved that the decision version of that problem is (weakly) NP-hard, For three (or more) protected attribute, even the question whether there exists a set of top-k that satisfies the complex fairness constraint is strongly NP-Hard by a reduction from 3 Dimensional Matching. On the positive side for the case of two protected attributes we designed an efficient algorithm that obtains a 2 approximation factor and runs in $O(n^2 \ell \log m)$ time, where ℓ is the number of possible attribute values. We also designed an exact algorithm with running time n^c, where c is the size of the Cartesian product of all the attribute domains.

4.2.2 Open Problems

There propose two possible ways to extend these problems.

Improved approximation ratio in the case of 2 protected attributes. Since the problem of minimizing the number of single ballot substitutions in the case of 2 attributes is currently proven to be weakly NP-Hard, it may admit a PTAS (Polynomial Time Approximation Scheme). We plan to investigate the existence of a better approximation algorithm. Alternatively, we will try to improve the hardness result and show that this problem is strongly NP-Hard or Max-SNP Complete.

Relaxed solutions in the case of 3 or more protected attributes. Clearly, the hardness result of even checking the existence of a solution in case of 3 or more attributes precludes the existence of any approximation algorithm

for this case. We plan to design an algorithm that will generate a relaxed set of items/candidates. The relaxation may be in two dimensions: (i) the generated set will be a top-k set of candidates but the fairness requirements will not be fully satisfied for all protected attributes. (ii) the generated set will have size larger than k but it will satisfy the (lower bounds of the) fairness constraints for top k. Clearly, the larger the generated set is the easier the problem. We will find the smallest such extended set that guarantees the fairness constraints imposed by all protected attributes.

5 Multi Round Rank based Preference Aggregation

We study algorithmic challenges to satisfy output constraints in multi-round rank based preference aggregation methods, popularly known as ranked choice voting or (RCV) [12].

5.1 Research Directions

We start by describing the STV (single transferable vote) method [10, 13] that generalizes the IRV method, and selects a set of k items/candidates as the winners. STV is gaining popularity as an electoral system. It is used to elect candidates to the Australian Senate, in all elections in Malta, in most elections in the Republic of Ireland, and in Cambridge, MA. There are also plans to use STV in other USA localities. As mentioned in Section 3 this method of preference aggregation is also applicable in other settings.

The input to an STV preference aggregation method consists of m either complete or partial rankings of the items/candidates. Suppose that the total number of items/candidates is n out of which k items need to be elected. The preference aggregation process requires a predefined quota. In most cases this quota is Droop quota [22] defined as $\left\lfloor \frac{n}{k+1} \right\rfloor + 1$. The aggregation is done in rounds. In each round every item/candidate is associated a tally. Initially, the tally of every item is the number of rankings in which it is ranked highest. A round starts by considering the items whose tally is at least the quota. These items are elected in non-increasing order of their tally, as long as k items/candidates have not been elected (which always holds for Droop quota). When an item is elected their "surplus" (the number by which their tally exceeds the quota) is distributed to the next preferred item in their ranking (that has not been eliminated yet). The exact way this "surplus" is allocated varies. In a most cases, this allocation is done either fractionally or by a random selection of the surplus rankings out of all the rankings in which the elected item is top ranked. This is repeated as long as there are items whose tally is at least the quota (and k items/candidates have not been elected). Then, if less than k items/candidates are elected, the item/candidate with the smallest tally is eliminated from all the rankings, and the tallies are updated based on the new rankings. If the number of items/candidates remaining (not elected and not yet eliminated) equals the number of items/candidates left to be elected, these candidates are elected and the STV process terminates, otherwise the process repeats.

There is evidence that IRV and thus also STV preference aggregation methods are computationally hard to manipulate. It is NP-Hard to decide whether an IRV method can be manipulated even by adding one complete ranking [6]. On the positive side, [9, 11, 23] suggested branch and bound algorithms that use Integer Programming to compute the Margin of Victory (MOV) in IRV.

Approximating the number of ranking substitutions in multi round methods. We plan to develop approximation algorithms with proven performance for IRV and STV. The first step is to design such an algorithm for the simplest case which is approximating the minimum number of ranking substitutions required to change the outcome of an IRV preference aggregation method when every user is limited to input only two top items. From there we hope to be able to generalize to the IRV problem with no restriction on the ranking size, and eventually to the more general STV.

Improved computational frameworks for minimizing number of ranking substitutions in multi round methods. As mentioned above most of the existing computational frameworks are based on branch and bound algorithms. We plan to investigate other methods and possibly alternative formulation of the respective Integer Programming model that may result in more efficient computational frameworks.

Heuristic algorithms for minimizing the number of ranking substitutions in multi round methods. Another way to tackle the complex computational problem of minimizing number of ranking substitutions in multi round methods is designing heuristics for this task, analyzing and benchmarking these heuristics. One approach for designing such a heuristic for the problem of minimizing the number of ranking substitutions in STV to guarantee an elected set of k items with a given requirement on their protected attribute is by first identifying the desired elected set and then computing the number of substitutions required to achieve this set. One way of fixing the desired set is as follows. Run the STV process, and whenever the number of the currently elected items/candidates with a given value of their protected attribute reaches its bound, eliminate all the items/candidates with this value of their protected attribute. A naive implementation of this rule may not even guarantee a feasible solution and thus we also need to add the option of reintroducing items/candidates that were already eliminated. Analyzing such an algorithm is a challenge.

6 Non-rank based Preference Aggregation

Our goal is to study preference models that allow users to elicit their choice not as a ranked order. When the input preferences are not ranked, the output produces a set of k items that best reflects the users preferences. Akin to the previous two sections, our goal is to investigate which preference aggregation methods are suitable for such elicitation models, how to handle output constraints, and understand their computational implications. We identify the following research directions.

6.1 Research Directions

We begin by considering simple Boolean preference elicitation models, as"only likes", "likes and dislikes" , or "only dislikes". Indeed, such preference elicitation models are realistic in a wide variety of applications, such as providing preferences over products, news articles, movies, songs, social media posts, to name a few.

The simplest form of preference elicitation comes in the following form - each user $u(i)$ provides σ_i as preference, which is a Boolean vector of 1's and 0's over the set of n items, and the underlying application *only objective* is to find a set of k-items that are "most liked" by all the users. We propose to use Jaccard similarity or overlap similarity [24] for measuring similarity (inverse of distance) between two vectors in such cases. Given two vectors σ_i, σ_j their overlap similarity $S_{\sigma_i,\sigma_j} = \sum_{\forall \ell \in [n]} [\sigma_{i_\ell} \wedge \sigma_{j_\ell}]$, the number of positive bits that are shared between σ_i, σ_j. When the users provide both "likes" and "dislikes" and both have to be accounted for, we will use Hamming Distance which measures the minimum number of substitutions required to change σ_i to σ_j.

We have explored two alternative preference aggregation methods [3, 24] in the past that serve as the basis of this study.

- **Aggregated Voting.** Produce σ, such that $\mathcal{F}(\sigma, \sigma_1) + \mathcal{F}(\sigma, \sigma_2) + \ldots \mathcal{F}(\sigma, \sigma_m)$ is minimized.
- **Least Misery.** Produce σ, such that $\mathbf{Maximum}\{\mathcal{F}(\sigma, \sigma_1), \mathcal{F}(\sigma, \sigma_2), \ldots \mathcal{F}(\sigma, \sigma_m)\}$ is minimized.

The goal is to produce σ, which is also a vector of length n with exactly k number of 1's and remaining $0's$ that minimizes the Inverse of overlap similarity/Hamming Distance, denoted $\mathcal{F}(\cdot, \cdot)$, between σ and $\{\sigma_1, \sigma_2, \ldots\ldots, \sigma_m\}$.

We realize that the overlap similarity function is monotone, as when a new item is considered in the mix, the overlap similarity can never decrease (or inverse overlap similarity can never increase). This is likely to make preference aggregation computationally tractable and give rise to polynomial time solution to produce optimal σ. Under Hamming distance, however, finding σ considering either of the preference aggregation models is likely to be NP-hard, as a known NP-Complete problem Median String Problem could be reduced to a variant of this problem [14].

Satisfying Output Constraints. The output constraints in this case are defined on the top-k items/candidates and involve one or more protected attributes. When the output criteria is simple (designed on a single attribute), the Preference Aggregation Problems Considering Constraints defined in Section 3 for aggregated voting under Overlap Similarity is likely to give rise to computationally tractable problem for all three variants - Constrained

optimization, Optimization under budgetary constraints, and Bi-criteria optimization. On the other hand, these problems are likely to be computationally harder for least misery under Overlap Similarity. We will study how to exploit the monotonicity property of overlap similarity to see if it is possible to design greedy algorithms with provable approximation factors. Under Hamming Distance, irrespective of the underlying aggregation method, the Preference Aggregation Problems Considering Constraints are likely to be NP-hard, since the Preference Aggregation under Hamming Distance itself is NP-hard. We intend to study the possibility of designing approximation algorithms as well as efficient heuristics for these problems.

6.2 Open Problems

The applicability of the ordinal preference model is explored as one of the open problems - an ordinal value g is defined on an s-point performance scale, that is totally ordered $g_1 \prec g_2 \prec \ldots \prec g_s$. Given m input ordinal preferences and an output criteria, goal is to produce σ (an ordered list of n items/ top-k ordered/unordered set) that aggregates the preferences and satisfies the criteria. The input is studied as ordered sorting problem in decision aid literature [18]. Concretely speaking, each user's preference σ_i corresponds to assignment of each item into a pre-defined ordered categories, such as excellent, good, average, poor and the aggregation problem intends to find the best set of k-items σ. When studied under output constraints, the general challenge is to minimally change the original outcome so as to satisfy the constraints.

Preference Aggregation Methods. One key challenge in ordinal preference elicitation model is to identify the appropriate aggregation method and/or distance functions. Per our initial investigation, we realize that an ordinal preference elicitation could be expressed as a set of pairwise comparisons. As an example, if user $u(i)$ rates i_1 as excellent, i_2 as good, and i_3 as fair, this gives rise to the following 3 pairwise comparisons: $i_1 \prec i_2, i_2 \prec i_3, i_1 \prec i_3$. Given two preferences σ_i, σ_j, one can compute Kendall-Tau distance between these two to quantify the number of inversions or distance between them. Given m input preferences $\sigma_1, \sigma_2, \ldots \ldots, \sigma_m$, when the output is to produce an ordered outcome, the preference aggregation problem intends to produce a ranking σ that optimizes (minimizes) the Kemeny Distance [28] (sum of Kendall-Tau distance) between σ and $\{\sigma_1, \sigma_2, \ldots, \sigma_m\}$.

Additionally, we will study partial net score [18] of an item i ($PNS(i)$) that is proposed as an indicator of computing the overall "worth" of an item in decision aid literature. Based on the aforementioned pairwise representation, $PNS(i)$ can be expressed as $PNS(i) = \sum_{j \in [n] \setminus \{i\}} \left(|u^{[i \prec j]}| - |u^{[j \prec i]}| \right)$. Basically, $PNS(i)$ is the number of times item i is preferred over any other item j by any user (represented as $u^{[i \prec j]}$) minus the number of times these other items are preferred over i by any user (represented as $u^{[j \prec i]}$). By computing partial net score of each item one can design the outcome σ easily and efficiently. If σ needs to be ordered then the items will be ordered in decreasing order of partial net score; when the goal is to produce a top-k set of items, this will contain the items with the top-k highest partial net score.

Satisfying Output Constraints. We will study how to satisfy output constraints that are suitable to ordinal preference models. We will study both simple and complex output constraints, defined over single and multiple attributes, respectively. For the preference aggregation problem under output constraints, this is equivalent to producing a σ that minimizes the partial net score or Kemeny Distance between σ and input preferences, while satisfying the output constraints. When studied as an optimization problem under budgetary constraints B (B is the upper bound of partial net score or Kemeny Distance), the goal will be to produce σ, such that partial net score or Kemeny Distance is at most B and $\mathcal{C}$ is optimized. We anticipate most of these problems to be NP-hard. We will study how to design efficient approximation algorithms with provable guarantees, as well as effective heuristic algorithms.

7 Conclusion

The article lays a scientific foundation for systematically changing the outcome of a variety of preference aggregation methods to satisfy additional criteria related to fairness and robustness. The article studies single-round rank based, multi-round rank based, and non rank based preference aggregation methods that are suitable to different preference elicitation models and investigates how to minimally modify them to promote fairness. It identifies underlying computational and algorithmic challenges, proposes research directions, and formalizes several open problems.

8 Acknowledgment

The work is supported by the NSF CAREER Award #1942913, IIS #2007935, IIS #1814595, PPoSS:Planning #2118458, and by the Office of Naval Research Grants No: #N000141812838, #N000142112966.

References

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: ranking and clustering. Journal of the ACM (JACM), 55(5):1–27, 2008.

[2] S. Amer-Yahia, B. Omidvar-Tehrani, S. Basu, and N. Shabib. Group recommendation with temporal affinities. In International Conference on Extending Database Technology (EDBT), 2015.

[3] S. Amer-Yahia, S. B. Roy, A. Chawlat, G. Das, and C. Yu. Group recommendation: Semantics and efficiency. Proceedings of the VLDB Endowment, 2(1):754–765, 2009.

[4] M. Ayadi, N. B. Amor, J. Lang, and D. Peters. Single transferable vote: Incomplete knowledge and communication issues. In 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS 19), pages 1288–1296, 2019.

[5] P. C. Baayen and Z. Hedrlin. On the existence of well distributed sequences in compact spaces. Stichting Mathematisch Centrum. Zuivere Wiskunde, 1964.

[6] J. J. Bartholdi III and J. B. Orlin. Single transferable vote resists strategic voting. Social Choice and Welfare, 8(4):341–354, 1991.

[7] S. K. Baruah, N. K. Cohen, C. G. Plaxton, and D. A. Varvel. Proportionate progress: A notion of fairness in resource allocation. Algorithmica, 15(6):600–625, 1996.

[8] S. Basu Roy, L. V. Lakshmanan, and R. Liu. From group recommendations to group formation. In Proceedings of the 2015 ACM SIGMOD international conference on management of data, pages 1603–1616, 2015.

[9] M. Blom, P. Stuckey, and V. Teague. Toward computing the margin of victory in single transferable vote elections. INFORMS Journal on Computing, 31:636—653, 05 2019.

[10] M. Blom, P. J. Stuckey, and V. J. Teague. Toward computing the margin of victory in single transferable vote elections. INFORMS Journal on Computing, 31(4):636–653, 2019.

[11] M. Blom, V. Teague, P. J. Stuckey, and R. Tidhar. Efficient computation of exact irv margins. In Proceedings of the Twenty-Second European Conference on Artificial Intelligence, ECAI'16, pages 480—-488. IOS Press, 2016.

[12] D. Cary. Estimating the margin of victory for instant-runoff voting. EVT/WOTE, 11, 2011.

[13] A. Chakraborty, G. K. Patro, N. Ganguly, K. P. Gummadi, and P. Loiseau. Equality of voice: Towards fair representation in crowdsourced top-k recommendations. In Proceedings of the Conference on Fairness, Accountability, and Transparency, pages 129–138, 2019.

[14] C. de la Higuera and F. Casacuberta. Topology of strings: Median string is np-complete. Theoretical computer science, 230(1-2):39–48, 2000.

[15] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In Proceedings of the 3rd innovations in theoretical computer science conference, pages 214–226, 2012.

[16] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In Proceedings of the 10th international conference on World Wide Web, pages 613–622, 2001.

[17] A. M. Feldman and R. Serrano. Welfare economics and social choice theory. Springer Science & Business Media, 2006.

[18] J. Figueira, S. Greco, M. Ehrogott, P. Meyer, and M. Roubens. Choice, ranking and sorting in fuzzy multiple criteria decision aid. Multiple criteria decision analysis: State of the art surveys, pages 471–503, 2005.

[19] M. M. Islam, M. Asadi, and S. Basu Roy. Equitable top-k results for long tail data. Proceedings of the ACM on Management of Data, 1(4):1–24, 2023.

[20] M. M. Islam, D. Wei, B. Schieber, and S. B. Roy. Satisfying complex top-k fairness constraints by preference substitutions. Proceedings of the VLDB Endowment, 16(2):317–329, 2022.

[21] C. Kuhlman and E. Rundensteiner. Rank aggregation algorithms for fair consensus. Proceedings of the VLDB Endowment, 13(12):2706–2719, 2020.

[22] J. Lundell and I. Hill. Notes on the droop quota. Voting matters, 24:3–6, 2007.

[23] T. R. Magrino, R. L. Rivest, and E. Shen. Computing the margin of victory in IRV elections. In 2011 Electronic Voting Technology Workshop/Workshop on Trustworthy Elections (EVT/WOTE 11), San Francisco, CA, 2011. USENIX Association.

[24] S. B. Roy, S. Thirumuruganathan, S. Amer-Yahia, G. Das, and C. Yu. Exploiting group recommendation functions for flexible preferences. In 2014 IEEE 30th international conference on data engineering, pages 412–423. IEEE, 2014.

[25] R. Tijdeman. The chairman assignment problem. Discrete Mathematics, 32(3):323–330, 1980.

[26] A. Van Zuylen and D. P. Williamson. Deterministic algorithms for rank aggregation and other ranking and clustering problems. In International Workshop on Approximation and Online Algorithms, pages 260–273. Springer, 2007.

[27] S. Verma and J. Rubin. Fairness definitions explained. In 2018 ieee/acm international workshop on software fairness (fairware), pages 1–7. IEEE, 2018.

[28] D. Wei, M. M. Islam, B. Schieber, and S. Basu Roy. Rank aggregation with proportionate fairness. In Proceedings of the 2022 International Conference on Management of Data, pages 262–275, 2022.

[29] M. Zehlike, K. Yang, and J. Stoyanovich. Fairness in ranking: A survey. arXiv preprint arXiv:2103.14000, 2021.