

Question the Questions: Auditing Representation in Online Deliberative Processes

Soham De^{1,2*}, Lodewijk Gelauff^{3*},
Ashish Goel^{3 †}, Smitha Milli^{1 †}, Ariel Procaccia^{1,4 †}, and Alice Siu^{3 †}

¹FAIR at Meta

²University of Washington

³Stanford University

⁴Harvard University

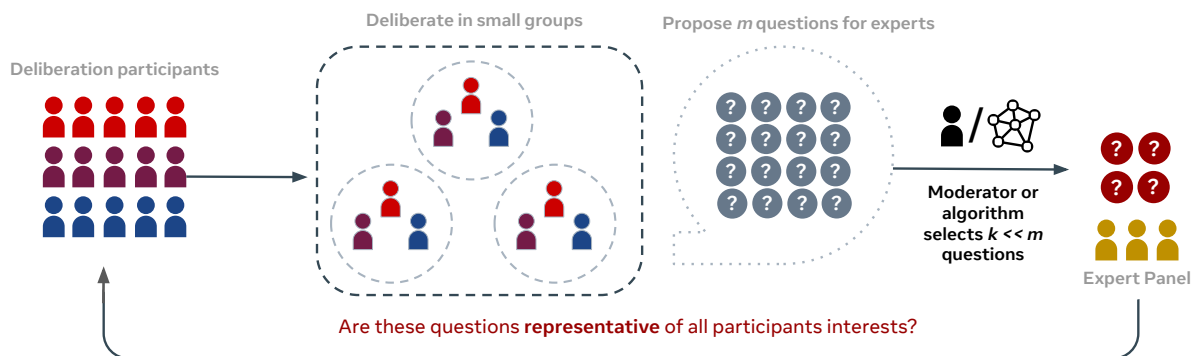
* Equal contribution

† Alphabetical order

November 2025

Abstract

A central feature of many deliberative processes, such as citizens' assemblies and deliberative polls, is the opportunity for participants to engage directly with experts. While participants are typically invited to propose questions for expert panels, only a limited number can be selected due to time constraints. This raises the challenge of how to choose a small set of questions that best represent the interests of all participants. We introduce an auditing framework for measuring the level of representation provided by a slate of questions, based on the social choice concept known as *justified representation* (JR). We present the first algorithms for auditing JR in the general utility setting, with our most efficient algorithm achieving a runtime of $O(mn \log n)$, where n is the number of participants and m is the number of proposed questions. We apply our auditing methods to historical deliberations, comparing the representativeness of (a) the actual questions posed to the expert panel (chosen by a moderator), (b) participants' questions chosen via integer linear programming, (c) summary questions generated by large language models (LLMs). Our results highlight both the promise and current limitations of LLMs in supporting deliberative processes. By integrating our methods into an online deliberation platform that has been used for over hundreds of deliberations across more than 50 countries, we make it easy for practitioners to audit and improve representation in future deliberations.



Our core contribution is an auditing method for representation, grounded in social choice theory, which we integrate into a popular online deliberation platform that has been used in >50 countries.

1 Introduction

Deliberative processes — such as citizens’ assemblies and deliberative polls [Fishkin, 1995] — involve randomly selected members of the public engaging in informed discussion in small groups about policy issues. These processes have gained popularity across the globe as a means to understand the reflective preferences of a population and guide collective decision-making [OECD, 2020]. This is especially true now that technology makes it possible to organize them largely online, significantly reducing the cost of assembling a representative set of participants.

One critical component of many deliberative designs is the opportunity for participants to interact with experts to deepen their understanding of the topic. For instance, in the deliberative polling approach, each participant (or small group) is invited to submit a question for an expert panel. However, because these events often include hundreds of participants, only a limited number of questions can ultimately be posed to the experts.

This gives rise to an important challenge: how can a small set of questions be chosen to best represent the interests and concerns of the entire participant panel? Large language models (LLMs) may offer a promising approach. In recent years, LLMs have been used to generate consensus statements in caucasus deliberations [Tessler et al., 2024], create popular summary fact-checks on Community Notes programs on social media [De et al., 2025], and summarize public opinions on collective response systems such as Pol.is [Fish et al., 2024, Jigsaw, 2025, Small et al., 2023]. Similarly, LLMs could be leveraged to synthesize commonalities in participant questions and generate a small slate of final questions. However, given the often nuanced and sensitive nature of deliberative processes, it is essential to rigorously audit whether such LLM-generated summary questions actually represent all participants.

Auditing framework for representation. We contribute an auditing framework for measuring the level of representation provided by a slate of questions. To formalize representation, we utilize a quantitative variant [Bardal et al., 2025] of a widely-used concept from social choice theory known as *justified representation* (JR) [Aziz et al., 2017]. Justified representation is an axiom that requires, informally speaking, that if k questions are selected, then any group of n/k participants (where n is the total number of participants) with shared preferences get at least one slot in the slate. We present the first algorithms for auditing JR in the general utility setting (beyond approval voting), the most efficient of which achieves an $O(mn \log n)$ runtime, where m is the number of items (participant-proposed questions) and n is the number of participants.

Auditing historical deliberations. We apply our algorithms to audit the representation of LLM-generated slates of questions in historical deliberations conducted by the Stanford Deliberative Democracy Lab. The deliberations span topics in democratic reform and generative AI. We compare the actual historical questions posed to experts (selected by a moderator) to two algorithmic approaches for question selection and generation. The first is an extractive summarization method, which selects a representative slate exclusively from participants’ original questions using an integer program that is designed to provide the optimal selection from the lens of representation (optimize the JR value). In contrast, the abstractive summarization approach synthesizes new summary questions from participant input, generating LLM-based questions that may not have been explicitly proposed.

Our findings indicate that both extractive and abstractive algorithmic methods generally yield question slates with greater representativeness than those selected by human moderators. In some cases, the abstractive approach outperforms the extractive method, while in others, the reverse holds true. This suggests that although LLM-generated summaries can sometimes enhance representativeness by synthesizing information, their benefits are not uniform. These results underscore the need for robust auditing mechanisms to determine when LLM-based methods provide added value over simply selecting from participants’ original questions — particularly since relying on participants’ own questions may offer additional benefits in terms of trust and transparency.

Integration into online deliberation platform. Finally, while we evaluate on historical data, we integrate both our auditing algorithms and question selection / generation algorithms into an online deliberation platform [Gelauff et al., 2023] that has been used to conduct hundreds of deliberations in over 50 countries, enabling practitioners to easily utilize our approach in future deliberations.

2 Related Work

AI & deliberation. The success of large language models has spurred growing interest in applying AI to deliberative contexts [Landemore, 2022, McKinney, 2024, Small et al., 2023, Korre et al., 2025]. Most relatedly, McKinney [McKinney, 2024] proposes a framework for assessing the desirability of integrating AI tools into citizens’ assemblies, focusing on two key dimensions: democratic quality and institutional capacity. In our work, we address the challenge of selecting (or generating) a set of questions for an expert panel that most effectively represents participants’ interests. Rather than the development of a specific AI tool, our primary contribution is an auditing method that can be used to assess the representativeness of different approaches to question selection, whether algorithmic or human-mediated. By applying this auditing method to different algorithmic strategies, we can identify automated solutions that achieve high levels of representativeness. This focus directly supports the democratic good of inclusiveness highlighted in McKinney’s framework. Additionally, such algorithms advance McKinney’s institutional goods of efficiency and scalability, as automating the question selection process eliminates the need for manual review, making it feasible to scale to a much larger number of participant questions.

As one approach to question selection, we evaluate the representativeness of LLM-generated summary questions. This is motivated by recent work using LLMs to synthesize the opinions of a collective. For example, Tessler et al. [Tessler et al., 2024] leverage LLMs to generate consensus statements for caucus deliberation. De et al. [De et al., 2025] build on this approach to generate summary fact-checks for Community Notes programs on social media that are preferred over human-written ones. Small et al. [Small et al., 2023] discuss the opportunities and risks of integrating LLMs into pol.is [Small et al., 2021], a collective response system [Ovadya, 2023] used to understand public opinion at scale. Google Jigsaw [Jigsaw, 2025] recently open-sourced a suite of LLM tools aimed at summarizing and identifying common ground in large-scale conversations such as those that take place on pol.is. Most relevant to our work, Fish et al. [Fish et al., 2024] develop a method that uses LLMs to generate summary slates of opinions that satisfy certain social choice guarantees of representation. In contrast, our focus is on developing an auditing algorithm for representation, enabling us to evaluate both algorithmic and human-mediated approaches and to quantify the potential improvements offered by algorithmic solutions.

Social choice theory. Our auditing framework provides a way to measure how representative a slate of questions is. To formalize representation, we draw upon the social choice concept of *justified representation* (JR) [Aziz et al., 2017]. JR is the building block that a family of subsequent axioms is based upon, e.g., PJR [Sánchez-Fernández et al., 2017], EJRP [Aziz et al., 2017], EJRP+ [Brill and Peters, 2023], and BJR [Fish et al., 2024]. While JR was originally introduced in the context of approval voting [Aziz et al., 2017], we adopt a version that accommodates general utility functions. Building on Bardal et al. [Bardal et al., 2025], we further employ a quantitative formulation of JR, which allows us to measure the degree of representativeness of a slate, rather than simply determining whether JR is satisfied. Although verifying JR in the approval voting setting is straightforward, we develop auditing algorithms tailored to the more general utility setting.

Our focus is on auditing the JR guarantees of arbitrary slates of questions *relative* to the participants’ own questions. By contrast, the work of Fish et al. [Fish et al., 2024] and Boehmer et al. [Boehmer et al., 2025] aims to generate slates that satisfy BJR over the infinite set of possible statements, but do not address the problem of auditing or verifying these guarantees. Because we seek to audit slates that may contain arbitrary questions—including those not proposed by participants, such as LLM-generated questions—we need a way to estimate participants’ utilities for these questions. To achieve this in a straightforward and interpretable manner suitable for deliberative settings, we infer a participant’s utility for a given question based on the distance between the LLM embedding of that question and the embedding of the participant’s own question. This approach bears similarity to works on proportional clustering [Chen et al., 2019, Micha and Shah, 2020, Aziz et al., 2024].

3 The Auditing Framework

Suppose there are n participants in a deliberation who propose a set $Q_p \subseteq Q$ of m potential questions for an expert panel, where Q denotes the universe of all possible questions. Due to time constraints, however, only

a small set of $k \ll m$ questions can actually be posed. The challenge, then, is to choose a set of k questions $W \subseteq Q$ that best represents the interests of the overall population. In this section, we outline our framework for auditing the degree of representation that a slate of questions offers to a population of participants. In Section 3.1, we introduce the justified representation (JR) axiom [Aziz et al., 2017] from social choice, along with the quantitative variant we employ [Bardal et al., 2025], which we use to formalize the notion of representation. In Section 3.2, we explain how we infer the utility that participants derive from arbitrary questions. In Section 3.3, we present the first known methods for auditing JR in the general utility setting, the best of which runs in time $O(mn \log n)$.

3.1 Justified representation

To formalize what it means for a set of questions to be representative, we draw on the social choice concept of *justified representation* (JR). Informally, JR is grounded in the principle of proportionality: JR requires that if k questions are asked, then any group of at least n/k participants—large enough to “deserve” one question by proportional allocation—who share similar preferences should have at least one question among the k that represents them.

Justified representation was first introduced by Aziz et al. [Aziz et al., 2017] in the context of approval voting, where each participant simply indicates a binary approval or disapproval for each of the m candidates. However, the concept extends naturally to more general utility settings, where each participant i has a utility $u_i(q) \geq 0$ for a question q or has a utility $v_i(W) \geq 0$ for a slate of questions W . In our setting, we specifically model a participant’s utility for a slate as being *unit-demand*, meaning their utility is defined by their best item on the slate: $v_i(W) = \max_{q \in W} u_i(q)$.¹ JR then requires that there can be no coalition of size at least n/k such that the coalition’s minimum utility for an alternative question is greater than the coalition’s maximum utility for the given slate. The formal definition is as follows:

Definition 1 (JR with utilities). *A slate of questions $W \subseteq Q$ satisfies JR if for every coalition S of size at least n/k , there is no alternative question $q \in Q_p$ such that $\min_{i \in S} u_i(q) > \max_{i \in S} v_i(W)$.*

Definition 1 departs slightly from the standard definition of JR by allowing the slate W to consist of arbitrary questions from the entire space Q , rather than being restricted to the discrete set Q_p of participant-proposed questions.² In our setting, we will also audit slates of LLM-generated summary questions that synthesize, but are distinct from, the participants’ original questions.

As originally proposed, JR provides only a binary notion of whether a slate is representative or not. However, we would also like to quantify the degree of representation offered by different question selection methods. Following Bardal et al. [Bardal et al., 2025], we quantify JR by identifying the smallest size group for which a JR guarantee holds. Specifically, we find the smallest size α for which α -JR is satisfied, where α -JR generalizes JR by providing guarantees to coalitions of size at least $\alpha \cdot n/k$. When $\alpha = 1$, we recover the original JR definition, guaranteeing representation only to groups of size at least n/k . When $\alpha < 1$, the guarantee extends to smaller-sized coalitions.

Definition 2 (α -JR with utilities). *A slate of questions $W \subseteq Q$ satisfies α -JR if for every coalition S of size at least $\alpha \cdot n/k$ there is no alternative question $q \in Q_p$ such that $\min_{i \in S} u_i(q) > \max_{i \in S} v_i(W)$.*

We audit the level of representation that a slate W provides by calculating the highest value of α for which it satisfies α -JR. We refer to this optimal value as the *JR value of W* : $\alpha_{\text{JR}}(W) = \inf\{\alpha : W \text{ satisfies } \alpha\text{-JR}\}$.

¹Other options are possible, e.g., additive utilities: $v_i(W) = \sum_{q \in W} u_i(q)$. However, we observed that additive utilities tend to make JR trivially satisfied in our context, since utilities are based on cosine similarity between question embeddings (Section 3.2). As a result, even with relatively small k , it is unlikely that a participant’s utility for any single question would exceed their utility for even a random slate.

²In this case, satisfying JR means that there is no sufficiently large, cohesive coalition that would prefer to deviate to a participant-proposed question in Q_p . However, it does not guarantee that there is no coalition that prefers some other question in the infinite set Q of all potential questions [Fish et al., 2024, Boehmer et al., 2025].

3.2 Utility inferences

In order to measure the JR value of an arbitrary slate $W \subseteq Q$, we must be able to infer the utility that a participant has for any potential question $q \in Q$, beyond just the participants’ own proposed questions. While complex models for predicting user utility are common in domains like recommender systems [Zhang et al., 2019], in the context of deliberation, there is usually very limited information available about participants to support such inferences. Furthermore, transparency is crucial; it must be possible to clearly explain how utility estimates are derived to both moderators and participants [McKinney, 2024].

Given these constraints, we adopt a simple and interpretable approach: we infer a participant’s utility for a question based on its similarity to their own proposed question. Specifically, each participant i is associated with their proposed question q_i , and we then measure the participant’s utility $u_i(q')$ for an alternate question q' is defined as $u_i(q') = s(q_i, q')$ where s is a similarity measure. In our implementation, we use cosine similarity between LLM-generated embeddings of the two questions.

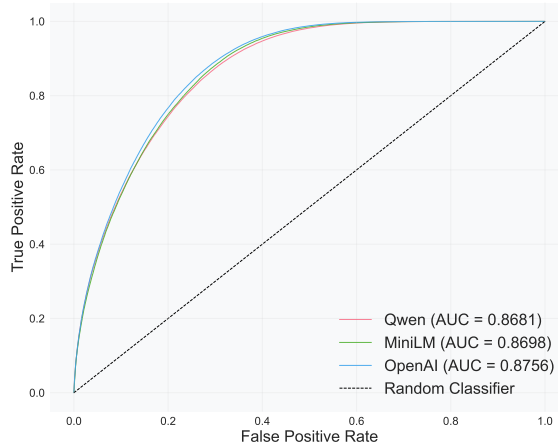


Figure 1: ROC curves comparing the binary classification accuracy of different embedding models on the Quora Question Pairs (QQP) dataset Iyer et al. [2017]. Each curve is obtained by thresholding the cosine similarity between the embeddings of paired questions.

Cosine similarity between modern sentence embeddings has been shown to correlate well with human similarity judgments [Gao et al., 2021]. To validate our approach in the question context, we evaluate how effectively cosine similarity can differentiate between duplicate and non-duplicate questions using the Quora question pairs dataset, which includes human-annotated binary labels (0 for different meanings, 1 for the same meaning) Iyer et al. [2017]. Figure 1 presents the ROC curves generated by varying the cosine similarity threshold across three different embedding models (Qwen3 0.6B Zhang et al. [2025], all-MiniLM-L6-v2 and text-embedding-3-small (OpenAI)). All embedding models achieve high AUC scores between 0.868 and 0.876. To demonstrate that our auditing approach (see Section 3.3) yields consistent outcomes across different embedding models, we apply it to slates identified as most representative by any given embedding model (using the integer program described in Section 7.1). We find that these slates yield similarly low JR-values when evaluated with other widely-used embedding models (Figure 2).

3.3 Auditing algorithms

We now introduce our algorithms for auditing the JR value of a slate. To our knowledge, these algorithms are also the first algorithms for verifying JR in the general utility setting. For binary (approval) utilities, verifying whether a slate W satisfies JR is straightforward [Aziz et al., 2017]: one simply checks whether there exists an alternative $q \in Q_p$ that is approved by at least n/k participants, all of whom disapprove every question in the slate. In contrast, for general additive utilities, the verification problem is more complex. A JR violation occurs when there exists an alternative $q \in Q_p$, a utility threshold $\tau > 0$, and a group of participants S of size

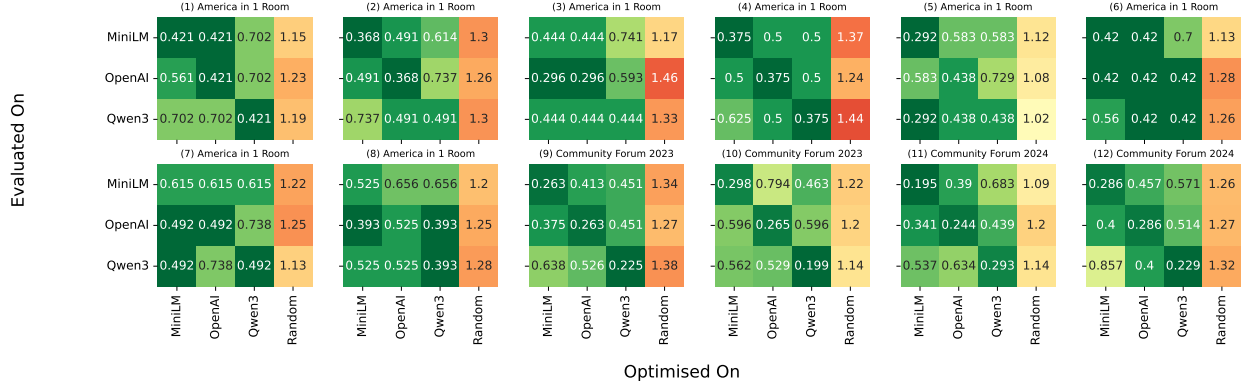


Figure 2: Cross-validating audit outcomes across embedding models. Each heatmap shows the JR-value when slates optimized using an integer program on one embedding model are evaluated using another model. Rows indicate the model used for evaluation, and columns indicate the model used for optimization (via the IP). Lower JR-values indicate greater consistency between optimization and evaluation models. Any value below 1 implies that the slate satisfies JR.

at least n/k where, for every participant $i \in S$, their utility for the alternative $u_i(q)$ is greater than τ , while their utility for the slate $v_i(W)$ is at most τ .

Naive algorithm. At first glance, it may seem necessary to check an infinite set of possible utility thresholds τ . However, it suffices to consider only the thresholds defined by the participants' utilities for the slate, i.e., the set of n thresholds: $T = \{v_i(W)\}_{i \in [n]}$. For each alternative $q \in Q_p$, we can iterate over each threshold $\tau \in T$ to identify the number of participants whose utility for the alternative q exceeds τ , while their utility for the slate W is at most τ . While doing so, we keep track of the largest size of a deviating coalition, which allows us to find the JR value of the slate. This approach yields the simple $O(mn^2)$ algorithm described in Algorithm 1.

Algorithm 1 Compute $\alpha_{JR}(W)$ in $O(mn^2)$ time

Require: Slate W , participant questions Q_p , utility function $u_i : Q \rightarrow \mathbb{R}_{\geq 0}$ for each participant $i \in [n]$

- 1: $\alpha \leftarrow 0$
- 2: **for all** $q \in Q_p$ **do**
- 3: **for** $i = 1$ to n **do**
- 4: $\tau \leftarrow v_i(W)$
- 5: $c \leftarrow |\{j \in [n] : u_j(q) > \tau \text{ and } v_j(W) \leq \tau\}|$
- 6: $\alpha \leftarrow \max(\alpha, c \cdot k/n)$
- return** α

Single-pass algorithm. We now propose an improved algorithm that achieves an $O(mn \log n)$ runtime (Algorithm 2). The previous approach (Algorithm 1) iterates over each alternative question $q \in Q_p$ and, for each possible utility threshold $\tau \in T$, determines the largest coalition of participants who would prefer q to W . This results in a higher computational cost, as each of the n thresholds is considered separately. In contrast, Algorithm 2 exploits the observation that, for each question q , the largest blocking coalition can be identified in a single pass over the sorted utility thresholds.

First, the participants are sorted in non-increasing order according to their utility for the given slate W , resulting in the ordering $\gamma_1, \dots, \gamma_n \in [n]$ where $v_{\gamma_1}(W) \geq \dots \geq v_{\gamma_n}(W)$. For each alternative question $q \in Q_p$, the participants are also sorted in non-increasing order of their utility for q , resulting in the ordering $\delta_1, \dots, \delta_n \in [n]$ where $u_{\delta_1}(q) \geq \dots \geq u_{\delta_n}(q)$. The algorithm then iterates through the sorted utilities $\{u_{\delta_i}(q)\}_{i=1}^n$, maintaining a counter c_q for the size of the largest coalition that prefers the question q to the slate W . Here, rather than checking each utility threshold $\{v_{\gamma_i}(W)\}_{i=1}^n$ separately, we iterate over them in one pass, starting with $t = 1$. Each participant's utility $u_{\delta_i}(q)$ for q is compared to the current threshold

$v_{\gamma_t}(W)$. If $u_{\delta_i}(q) > v_{\gamma_t}(W)$, then the participant is added to the coalition, and the counter is incremented. If $u_{\delta_i}(q) \leq v_{\gamma_t}(W)$, then this means that the coalition size cannot be increased without lowering the threshold. Thus, participant γ_t is blacklisted from future coalitions (and removed from the existing coalition, if applicable), and the utility threshold is decreased by incrementing t .

This process continues until all participants have been considered or all thresholds have been exhausted. The maximum coalition size c encountered during this process for any question q is recorded, and the algorithm returns $\alpha_{JR}(W) = \frac{c}{n/k}$. By leveraging sorted orderings and a single-pass approach, this algorithm achieves a significant improvement in efficiency, enabling the exhaustive verification of JR for a large number of candidate slates.

Algorithm 2 Compute $\alpha_{JR}(W)$ in $O(mn \log n)$ time

Require: Slate W , participant questions Q_p , utility function $u_i : Q \rightarrow \mathbb{R}_{\geq 0}$ for each participant $i \in [n]$

```

1:  $\gamma_1, \dots, \gamma_n \leftarrow$  Participants sorted such that  $v_{\gamma_1}(W) \geq \dots \geq v_{\gamma_n}(W)$ 
2:  $c \leftarrow 0$ 
3: for all  $q \in Q_p$  do
4:    $\delta_1, \dots, \delta_n \leftarrow$  Participants sorted s.t.  $u_{\delta_1}(q) \geq \dots \geq u_{\delta_n}(q)$ 
5:    $c_q \leftarrow 0$  ▷ Size of deviating coalition for question  $q$ 
6:    $S, B \leftarrow \emptyset, \emptyset$  ▷ Current coalition and blacklisted participants
7:    $t \leftarrow 0$ 
8:   for  $i = 1$  to  $n$  do
9:     while  $t < n$  and  $u_i(x) \leq v_{\gamma_t}(W)$  do
10:      if  $\gamma_t \in S$  then
11:         $c_q \leftarrow c_q - 1$ 
12:         $B \leftarrow B \cup \{\gamma_t\}$ 
13:         $t \leftarrow t + 1$ 
14:      if  $t \geq n$  then
15:        Break
16:      if  $i \notin B$  then
17:         $S \leftarrow S \cup \{i\}$ 
18:         $c_q \leftarrow c_q + 1$ 
19:       $c \leftarrow \max(c_q, c)$ 
return  $c \cdot k/n$ 
```

4 Empirical Evaluation

In the following sections, we first describe the platform and deliberative polls (Section 4.1) that provide the data used for our audit. Then, we introduce the relevant baselines and present the key results from applying our audit to this data (Section 4.2).

4.1 Deliberative Polls and the Online Deliberation Platform

We evaluate our audit on data from 12 sessions in 3 different *deliberative polls* [Fishkin, 1995, Luskin et al., 2002], conducted by the Stanford Deliberative Democracy Lab, that cover a wide range of topics — from the American political system to AI agents. During such a deliberative poll, participants are assigned to small groups with on average 10 participants; each group has a conversation, at the end of which they propose and rank questions for a plenary panel of experts. The plenary panel then is presented a slate of 5–15 of these questions that helps clarify the topic for the participants. Moderators may also occasionally group multiple very-similar ‘sibling’³ questions together and pose them to the panel. It is this set of selected questions that

³In human-generated slates with a set of sibling questions, we consider all possible slates that can be formed by taking one sibling question as the representative of its sibling set and report the average JR value for the Human baseline in Table 1.

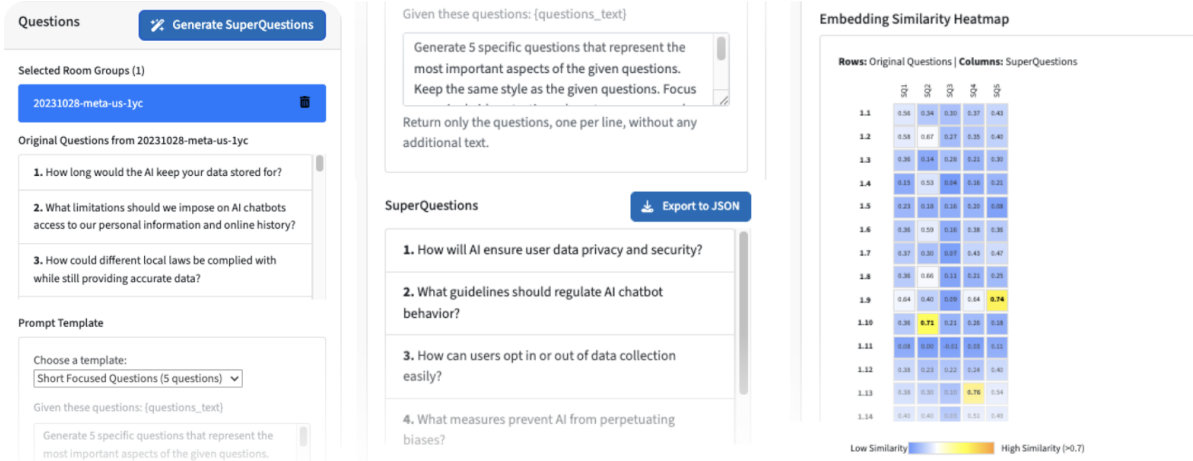


Figure 3: Screenshots illustrating our approach implemented in the online deliberation platform. The moderator can generate LLM summary questions (referred to as “SuperQuestions” in the interface) from participant-proposed questions; view which participant-proposed questions are most similar to each LLM-generated one; and export all data including, similarity scores, for representation auditing.

we use to evaluate our algorithms. In Section 4.2, we evaluate the JR value of the slates of question selected by moderators, compared to the slates of questions generated by algorithmic approaches.

We consider datasets based on sessions from 3 deliberations: “America in One Room: Democratic Reform” (June 2023), the “Meta Community Forum on AI chatbots” (two sessions from October 2023), and the “Meta Community Forum on AI agents” (two sessions from October 2024).

- The America in One Room (A1R) deliberations convened U.S. nationally representative sample of participants to discuss potential reforms to the American political system, such as changes to the electoral college and the adoption of ranked-choice voting [Fishkin and Diamond, 2023]. This results in 8 sets of questions and corresponding slates.
- The 2023 Meta Community Forum (CF’23) convened nationally representative samples of participants from Brazil, Germany, Spain and the United States to discuss policies for user interactions of AI chatbots in their respective national languages in small groups. In two sessions of small groups, the participants proposed questions, followed by an international plenary panel of experts [Chang et al., 2024]. This results in 2 sets of questions and corresponding slates.
- The 2024 Meta Community Forum (CF’24) similarly convened a nationally representative sample of participants from Brazil, India, Nigeria, Saudi Arabia, and South Africa that discussed policies and conditions for AI agents [Chang et al., 2025]. This results in 2 sets of questions and corresponding slates.

All deliberations were conducted on an online, self-moderating deliberation platform that has been shown to perform on par with human moderators across multiple dimensions [Gelauff et al., 2023]. We implemented our auditing algorithms, as well as algorithmic summarization using LLMs, within this same online deliberation platform (Figure 3). The goal of the implementation is to provide an easy way for moderators to generate and evaluate different potential slate of questions to pose to an expert panel. For each generated slate, the platform computes the JR value $\alpha_{JR}(W)$ and displays a heatmap illustrating the similarity between each selected question and those submitted by participants. This feature enables moderators, when presenting a question, to point out which participants submitted similar questions and to explicitly highlight how participants’ own contributions informed the final questions that were posed.

Panel	Human	Random	IP	LLM	LLM _{best}	N	k
A1R	0.842	1.227 $\pm$ 0.077	0.421	0.778 $\pm$ 0.032	0.421	57	8
A1R	0.798 [†]	1.258 $\pm$ 0.083	0.368	0.57 $\pm$ 0.023	0.368	57	7
A1R	1.037	1.459 $\pm$ 0.099	0.296	0.913 $\pm$ 0.052	0.444	54	8
A1R	0.875 [†]	1.241 $\pm$ 0.096	0.375	0.701 $\pm$ 0.039	0.375	56	7
A1R	1.021	1.084 $\pm$ 0.064	0.438	1.018 $\pm$ 0.044	0.583	48	7
A1R	1.68 [†]	1.281 $\pm$ 0.091	0.42	0.545 $\pm$ 0.022	0.28	50	7
A1R	0.738	1.247 $\pm$ 0.076	0.492	0.751 $\pm$ 0.041	0.369	65	8
A1R	1.049	1.246 $\pm$ 0.079	0.525	0.744 $\pm$ 0.032	0.393	61	8
CF'23	2.234 [†]	1.267 $\pm$ 0.076	0.263	1.052 $\pm$ 0.051	0.563	293	11
CF'23	0.827 [†]	1.197 $\pm$ 0.075	0.265	1.05 $\pm$ 0.043	0.562	272	9
CF'24	1.268	1.203 $\pm$ 0.067	0.244	0.948 $\pm$ 0.042	0.537	164	8
CF'24	0.8	1.267 $\pm$ 0.074	0.286	0.785 $\pm$ 0.056	0.4	175	10

Table 1: Levels of representation (α_{JR}) for different slates using embeddings from the text-embedding-3-small model by OpenAI. Baselines (Random and LLM-generated slates) reflect mean JR values over 100 runs, with 95% confidence intervals. In sessions marked with [†], human moderators posed multiple *sibling* questions.

4.2 Evaluating algorithms for question selection and generation

A representative slate of k questions that summarizes all the n questions proposed by participants in a deliberation can emerge via two distinct pathways. First, we may select a representative subset of k questions from the larger set of all proposed questions. We call the resulting slate an *extractive summarization* as it is formed by directly extracting from the set of all questions proposed by participants. Alternatively, we can generate k new questions that aim to summarize and highlight the important aspects of proposed questions but may not exist verbatim in the original set of n questions. Since these questions are formed via synthesis or abstraction, we call this slate an *abstractive summarization*.

We test the slates generated through five methods, the first three of which are extractive summarization methods, and the last two of which are abstractive.

1. *Random*. A baseline that picks from participant-proposed questions uniformly at random
2. *Human*. The historical questions posed to the expert panel, that were selected by the moderator.
3. *IP*. Following Theorem 2 in Bardal et al. [2025], we observe that it is NP-hard to determine existence of a slate satisfying α -JR for a given α . Hence, instead of a polynomial-time algorithm we implement a polynomial-size integer program, as described in Section 7.1, to determine the subset of participant-proposed questions that minimizes the JR value. In Table 1, we run our integer program using Gurobi and with all possible thresholds discretized at intervals of 0.01.
4. *LLM*. We prompt gpt-4o (see Appendix 7) with all participant-proposed questions to generate k representative questions. We sample multiple times while shuffling the order of questions with a temperature of one, resulting in 100 generated slates.
5. *LLM_{best}*. We pick the LLM-generated slate with the highest JR value, i.e., best-of- n sampling.

Table 1 presents results of running our audit on different question slates generated on data from 8 panels during the “America in One Room: Democratic Reform” (A1R) deliberation, 2 panels during the “2023 Meta Community Forum on AI Chatbots” (CF'23) deliberation and 2 panels during the “2024 Meta Community Forum on AI Agents” (CF'24) deliberation.

► **Historical performance by human moderators** These results reveal that slates selected by human moderators are, on average, more representative than a set of k randomly selected questions with a few key exceptions (such as in the first Meta'23 deliberation). However, we also note that in half of the panels on

Type	α_{JR}	Proposed Question Slates (A1R - Session 1)
IP	0.421	1. Rank choice voting seems like a good idea. Can you please discuss two pros and two cons to RCV. 2. Is there anything that prevents using RCV (Ranked Choice Voting) in party primaries? IE Federal Law, Funding Issues, Party Bylaws? 3. How can we change the system to benefit all Americans in their voting right? 4. Do you think proportional representation would help mitigate the problem of Gerrymandering? 5. Out of each of the proposals, which is considered the most cost-effective way to improve voting? 6. What is the best way for a third party (or multiple parties) to gain a foothold in the current two-party system? Which of these proposals can make that happen? 7. What does the political science research tell us about the difficulty of teaching voters how to operate in new systems, such as RCV, proportional representation, or changing to non-partisan primaries? How is Proportional Representation counted? 8. How successful is rcv in places where it has been used?
Human	0.842	1. What is the approximate cost to update voting machines to RCV? 2. what are examples of where rank choice voting has implemented and how well do people like it? 3. How do you best balance an rcv election where some rank and some dont rank and is that a fair and equal vote? 4. How can we trust the PR system to actually vote for the people? How will the people be educated on how it works? 5. In a proportional representation system at the congressional level, what would be some examples of implementation? In a multi- member district with 3 members, would all three members confer before voting as one block, or all vote with 1/3 of a vote? 6. The group in general seems highly against Gerrymandering. What can the experts tell us about the effects of any of these proposals to eliminate or reduce gerrymandering? 7. What is the best way for a third party (or multiple parties) to gain a foothold in the current two-party system? Which of these proposals can make that happen? 8. How soon would any changes to elections be implemented? Is this something we would do for the next Presidential election or are we 5-10 years away from these changes going into effect?
LLM _{best}	0.421	1. What are the most cost-effective ways to implement Ranked Choice Voting? 2. How can we address voter education challenges for new systems like RCV and proportional representation? 3. Which electoral reforms could most effectively reduce gerrymandering and partisanship? 4. How successful has RCV been in areas where it has been implemented, and what feedback has it received? 5. How would proportional representation impact party systems and mitigate political polarization? 6. What steps are needed to ensure fair enforcement of new voting systems and prevent political corruption? 7. Would implementing RCV require significant updates to voting machines, and what are the costs? 8. How can third-party and independent candidates gain traction in a two-party-dominated system?

Table 2: Questions in extractive and abstractive slates from the first panel in the A1R deliberation, as evaluated in Table 1

which we run our audit, human-moderator-selected slates do not satisfy JR, indicating a potential to improve the deliberative polling process by using more representative slates.

► **Abstractive vs. extractive slates** The slates selected by the integer program are the most representative extractive slate, which by definition must always have a $\alpha_{JR} \leq 1$. Similarly, the LLM_{best} slates are the most representative abstractive slates out of 100 LLM-generated candidates — an approximation of the most representative abstractive slate. We find that all algorithmic approaches to slate generation result in more representative slates when compared to random and human-selected slates that were historically used in the deliberative polls. Moreover, we find that abstractive slates are often able to match or surpass the representativeness of the best extractive slates, especially for sessions with fewer questions proposed. We also note that in sessions with more participants, abstractive slates are much more feasible to generate

in real time using LLMs, whereas finding the best extractive slate might be computationally intractable at interactive speeds. Relatedly, as the example in Table 2 reveals, abstractive question slates generated using LLMs typically appear more stylistically coherent and well composed compared to extractive slates, even when they achieve equivalent levels of representativeness.

Our results highlight the promise and limitations of using LLMs for generating slates in large online deliberative process. While LLM-generated abstractive slates consistently outperform existing human-driven processes, they do not uniformly surpass extractive slates selected by an integer program across all panels. This variability underscores the value of our efficient auditing algorithm, which enables a hybrid approach that leverages the strengths of both abstractive and extractive methods.

5 Conclusion

In this work, we address the challenge of selecting questions that are representative of all participants’ interests in large-scale deliberative processes. We present two complementary approaches towards solving this task — extractive (selecting from existing questions) and abstractive (generating new questions) — and develop a framework to evaluate their representativeness. To compare these approaches, we introduce an auditing framework grounded in the social choice concept of justified representation (JR). We present the first computationally efficient algorithms for auditing JR in general utility settings. Our framework uses widely available text-embedding models to infer participant utilities based on semantic similarity, enabling scalable application across diverse deliberative contexts. This semantics-based approach provides interpretable utility measurements and avoids potential biases that may arise from black-box predictive models. Our framework ensures that every participant who proposes a question during a deliberation is appropriately represented by the final slate of questions proposed to expert panelists.

Applying our methods to several historical deliberations retroactively, we demonstrate that algorithmically selected or generated questions have the potential to enhance representativeness, compared to those chosen by human moderators. This also highlights the utility of our auditing approach, which enables moderators to compare several algorithmically generated slates and evaluate them for representativeness. Finally, we integrate our framework into a widely-used online deliberation platform that has supported hundreds of deliberations in over 50 countries, making our methods readily accessible to improve representation in future deliberations.

Limitations and future work. Our work is not without limitations and we identify several promising directions for future research. First, our evaluation in Section 4 considers the JR value for LLM-generated and IP-selected questions separately; future work could explore hybrid approaches that select from both participant-proposed and LLM-generated questions. Second, we infer a participant’s utility for a question using the cosine similarity between the embeddings of that question and the participant’s proposed question. We note that our retrospective analysis of historical slates limits our ability to validate utility inferences with data from actual participants involved in the deliberation. While we evaluate our embedding-based approach on existing datasets with human similarity judgments (see Figure 1) and across different embedding models (Figure 2), future studies could directly validate these utility inferences with participants themselves during an ongoing deliberation. Finally, although our focus is on auditing the JR value, future work could explore auditing other axioms offering stronger representation guarantees such as BJR [Fish et al., 2024, Boehmer et al., 2025].

6 Acknowledgment

The authors would like to thank Harshvardhan Agarwal, Zixin Xu, and the other developers for their contributions to the Stanford Online Deliberation Platform, which was instrumental in data collection. Special thanks are extended to Harshvardhan Agarwal for implementing the algorithm. The authors also acknowledge Dhruv Gupta and Advay Ranade for their assistance with data collection for the human benchmarks. Finally, the authors are grateful to Audrey Tang for encouraging the exploration of this research direction. This work was supported in part by NSF Award #2333849.

References

- Haris Aziz, Markus Brill, Vincent Conitzer, Edith Elkind, Rupert Freeman, and Toby Walsh. Justified representation in approval-based committee voting. *Social Choice and Welfare*, 48(2):461–485, 2017.
- Haris Aziz, Barton E. Lee, Sean Morota Chu, and Jeremy Vollen. Proportionally representative clustering. In *Proceedings of the 20th Conference on Web and Internet Economics (WINE 2024)*, 2024. <https://arxiv.org/abs/2304.13917>.
- Tuva Bardal, Markus Brill, David McCune, and Jannik Peters. Proportional representation in practice: quantifying proportionality in ordinal elections. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI’25/IAAI’25/EAAI’25*. AAAI Press, 2025. ISBN 978-1-57735-897-8. doi: 10.1609/aaai.v39i13.33483. <https://doi.org/10.1609/aaai.v39i13.33483>.
- Niclas Boehmer, Sara Fish, and Ariel D. Procaccia. Generative Social Choice: The Next Generation. In *Forty-second International Conference on Machine Learning*, 2025. <https://openreview.net/forum?id=E1E6T7KH1R>.
- Markus Brill and Jannik Peters. Robust and verifiable proportionality axioms for multiwinner voting. In *Proceedings of the 24th ACM Conference on Economics and Computation, EC ’23*, page 301, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701047. doi: 10.1145/3580507.3597785. <https://doi.org/10.1145/3580507.3597785>.
- Samuel Chang, Estelle Ciesla, Michael Finch, James Fishkin, Lodewijk Gelauff, Ashish Goel, Ricky Hernandez Marquez, Shoaib Mohammed, and Alice Siu. Meta community forum: Results analysis, april 2024. Technical report, Meta and Stanford Deliberative Democracy Lab and Crowdsourced Democracy Team, Stanford University, April 2024. Corresponding author: Alice Siu, asiu@stanford.edu.
- Samuel Chang, James S. Fishkin, Ricky Hernandez Marquez, Ayushi Kadakia, Alice Siu, and Robert Taylor. Meta community forum: Results analysis, april 2025. Technical report, Meta and Stanford Deliberative Democracy Lab, April 2025.
- Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. Proportionally fair clustering. In *International Conference on Machine Learning*, pages 1032–1041. PMLR, 2019.
- Soham De, Michiel A. Bakker, Jay Baxter, and Martin Saveski. Supernotes: Driving consensus in crowdsourced fact-checking. In *Proceedings of the ACM on Web Conference 2025, WWW ’25*, page 3751–3761, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712746. doi: 10.1145/3696410.3714934. <https://doi.org/10.1145/3696410.3714934>.
- Sara Fish, Paul Gözl, David C. Parkes, Ariel D. Procaccia, Gili Rusak, Itai Shapira, and Manuel Wüthrich. Generative Social Choice. In *Proceedings of the 25th ACM Conference on Economics and Computation, EC ’24*, page 985, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400707049. doi: 10.1145/3670865.3673547. <https://doi.org/10.1145/3670865.3673547>.
- James Fishkin and Larry Diamond. Can deliberation cure our divisions about democracy? *Boston Globe*, (August 21, 2023), August 2023. <https://www.bostonglobe.com/2023/08/21/opinion/2024-elections-partisanship-democracy-common-ground/>.
- James S. Fishkin. *The Voice of the People: Public Opinion and Democracy*. Yale University Press, 1995. ISBN 9780300065565. <http://www.jstor.org/stable/j.ctt32bgmt>.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.552. <https://aclanthology.org/2021.emnlp-main.552/>.

Lodewijk Gelauff, Liubov Nikolenko, Sukolsak Sakshuwong, James Fishkin, Ashish Goel, Kamesh Munagala, and Alice Siu. Achieving parity with human moderators: A self-moderating platform for online deliberation 1. In *The Routledge handbook of collective intelligence for democracy and governance*, pages 202–221. Routledge, 2023.

Shankar Iyer, Nikhil Dandekar, and Kornél Csernai. Quora question pairs. <https://www.kaggle.com/competitions/quora-question-pairs>, 2017. Accessed: 2025-10-06.

Zhihao Jiang and Ashish Goel. Approximation algorithms for optimization problems with justified representation constraints. *Personal Communication; authors omitted for double blind review*, 2025.

Jigsaw. How one of the fastest-growing cities in Kentucky used AI to plan for the next 25 Years, Jun 2025. <https://medium.com/jigsaw/how-one-of-the-fastest-growing-cities-in-kentucky-used-ai-to-plan-for-the-next-25-years-3b70c4fd1412>

Katerina Korre, Dimitris Tsirmpas, Nikos Gkoumas, Emma Cabalé, Danai Myrtzani, Theodoros Evgeniou, Ion Androutsopoulos, and John Pavlopoulos. Evaluation and facilitation of online discussions in the llm era: A survey. *arXiv preprint arXiv:2503.01513*, 2025.

Helene Landemore. Can AI bring deliberative democracy to the masses? In *HAI Weekly Seminar; Working Paper*, pages 166–191, 2022.

Robert C. Luskin, James S. Fishkin, and Roger Jowell. Considered opinions: Deliberative polling in britain. *British Journal of Political Science*, 32(3):455–487, 2002. ISSN 00071234, 14692112. <http://www.jstor.org/stable/4092249>.

Sammy McKinney. Integrating artificial intelligence into citizens’ assemblies: Benefits, concerns and future pathways. *Journal of Deliberative Democracy*, 20(1), 2024.

Evi Micha and Nisarg Shah. Proportionally Fair Clustering Revisited. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, pages 85–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020.

OECD. *Innovative Citizen Participation and New Democratic Institutions: Catching the Deliberative Wave*. OECD Publishing, Paris, 2020. doi: 10.1787/339306da-en. <https://doi.org/10.1787/339306da-en>.

Aviv Ovadya. ‘Generative CI’ through Collective Response Systems, 2023. <https://arxiv.org/abs/2302.00672>.

Luis Sánchez-Fernández, Edith Elkind, Martin Lackner, Norberto Fernández, Jesús Fisteus, Pablo Basanta Val, and Piotr Skowron. Proportional justified representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Christopher Small, Michael Bjorkegren, Timo Erkkilä, Lynette Shaw, and Colin Megill. Polis: Scaling Deliberation by Mapping High Dimensional Opinion Spaces. *Recerca. Revista de Pensament i Anàlisi*, 26(2):1–26, 2021. doi: 0.6035/recerca.5516.

Christopher T Small, Ivan Vendrov, Esin Durmus, Hadjar Homaei, Elizabeth Barry, Julien Cornebise, Ted Suzman, Deep Ganguli, and Colin Megill. Opportunities and Risks of LLMs for Scalable Deliberation with Polis. *arXiv preprint arXiv:2306.11932*, 2023.

Michael Henry Tessler, Michiel A. Bakker, Daniel Jarrett, Hannah Sheahan, Martin J. Chadwick, Raphael Koster, Georgina Evans, Lucy Campbell-Gillingham, Tantum Collins, David C. Parkes, Matthew Botvinick, and Christopher Summerfield. Ai can help humans find common ground in democratic deliberation. *Science*, 386(6719):eadq2852, 2024. doi: 10.1126/science.adq2852. <https://www.science.org/doi/abs/10.1126/science.adq2852>.

Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Comput. Surv.*, 52(1), February 2019. ISSN 0360-0300. doi: 10.1145/3285029. <https://doi.org/10.1145/3285029>.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

7 Prompts

Below is the prompt we use for generating LLM summary questions.

```
Given these questions : \n
{questions} \n
Generate {k} specific concise questions that exhaustively summarize these given questions
as much as possible. Avoid long and generic high-level questions. Retain the same style,
specificity and length as the given questions. Return only the questions, one per line,
without any additional text.
```

7.1 Integer Program: Minimizing Largest Dissatisfied Coherent Set

Using the standard notation in social choice, we use the term “candidates” to refer to potential questions, and the term “voter” to refer to each question in the original data set. In our case, the set of candidates and voters is the same. We use similarity as the notion of utility, and assume that it is given to us (e.g. via embeddings). We will use N to refer to the number of voters, and M to the number of candidates for generality, even though these are the same for our use case.

The integer program is modeled after a similar integer program Jiang and Goel [2025] used to devise approximation algorithms for justified representation problems. We use binary integer variables to indicate whether a particular candidate is chosen. The central idea is to use auxiliary binary integer variables to keep track, for each utility level, whether a voter has at least that much utility given the set of chosen questions. This allows us to test, for each potential blocking question that is not chosen, whether that question would result in a blocking set of voters, that is, a set of voters whose minimum utility for the blocking question exceeds what all of them are currently receiving. We minimize the size of the largest blocking set, which allows us to optimize for the over-representation factor β that we are auditing for.

Given

- K : Committee size
- $u(v, c)$: Utility of voter v from candidate c (cosine similarity)

Indices used

- c : Candidates
- v : Voters
- s : Satisfaction level, ranges over all possible satisfaction levels (at most NM)
- c' : Blocking candidate

Variables

- J : Size of largest dissatisfied coherent set; integer variable
- x_c : Indicator variable, 1 if candidate c is in the committee; binary integer variable
- $y_{v,s}$: Indicator variable, 1 if voter v has utility at least s ; binary integer variable

Integer Program

$$\begin{aligned}
& \text{Minimize} && J \\
& \text{subject to} && \sum_c x_c = K \\
& && y_{v,s} \leq \sum_{c: u(v,c) \geq s} x_c \quad \forall v, s \\
& && \sum_{v: u(v,c') \geq s} (1 - y_{v,s}) \leq J \quad \forall c', s \\
& && x_c \in \{0, 1\} \quad \forall c \\
& && y_{v,s} \in \{0, 1\} \quad \forall v, s \\
& && J \in \mathbb{Z}; J \geq 0
\end{aligned}$$

This is a large integer program, with $O(N^2M)$ integer variables (since there can be up to NM distinct utility levels). To obtain a more efficient program, we simply relax the y variables to be fractional between 0 and 1, inclusive. At optimality, if a specific $y_{v,s}$ is strictly fractional, it can simply be set to 1 without violating any constraints (since x is still required to be binary). This gives us a much more efficient integer program, with only $O(M)$ integer variables:

More Efficient Integer Program

$$\begin{aligned}
& \text{Minimize} && J \\
& \text{subject to} && \sum_c x_c = K \\
& && y_{v,s} \leq \sum_{c: u(v,c) \geq s} x_c \quad \forall v, s \\
& && \sum_{v: u(v,c') \geq s} (1 - y_{v,s}) \leq J \quad \forall c', s \\
& && x_c \in \{0, 1\} \quad \forall c \\
& && 0 \leq y_{v,s} \leq 1 \quad \forall v, s \\
& && J \in \mathbb{Z}; J \geq 0
\end{aligned}$$

Note that J can also be similarly relaxed, but in practice we found that explicitly requiring J to be integral actually allows commercial solvers to solve the problem faster, since the solver can stop when the difference between an upper bound and a lower bound is less than 1.