Politics and Governance
2026 • Volume 14 • Article 11256
https://doi.org/10.17645/pag.11256

cogitatio

ARTICLE

Open Access Journal

Estimating the Recommendation Certainty in Candidate-Based Voting Advice Applications

Fynn Bachmann, Daan van der Weijden, Cristina Sarasua, and Abraham Bernstein

Department of Informatics, University of Zurich, Switzerland

Correspondence: Fynn Bachmann (fynn.bachmann@uzh.ch)

Submitted: 31 August 2025 **Accepted:** 12 November 2025 **Published:** 21 January 2026

Issue: This article is part of the issue "Voting Advice Applications: Methodological Innovations, Behavioural Effects, and Research Perspectives" edited by Diego Garzia (University of Lausanne / Bologna University), Stefan Marschall (Heinrich Heine University Düsseldorf), Mathias Wessel Tromborg (Aarhus University), and Andreas Albertsen (Aarhus University), fully open access at https://doi.org/10.17645/pag.i485

Abstract

Voting advice applications typically require users to answer questionnaires before receiving party or candidate recommendations. As users answer more questions, the recommendations naturally become more accurate. However, when users do not complete the questionnaire, the certainty of these recommendations is unknown. In this work, we develop and present a measure to quantify this certainty by introducing an algorithm that estimates the candidate recommendation accuracy—the overlap between early and final recommendations—after each question. Through simulations based on existing voter data, we find that our algorithm is more accurate than heuristic estimates. Additionally, it can identify *stable recommendations*—candidates who are likely to be among the final recommendations—with fewer false positives. Furthermore, we conduct a user experiment investigating different ways of communicating recommendation certainty to users. Our results show that users answer more questions when they see a preview of stable recommendations, but quit the questionnaire earlier when we display an artificially high candidate recommendation accuracy estimate. Moreover, we find that users appreciate the interface's simplicity over its accuracy. We conclude that displaying personalized stable recommendations can spark curiosity towards voting advice applications while providing a robust estimate of recommendation certainty for users who submit incomplete questionnaires.

Keywords

human–computer interaction; personalized interfaces; recommendation quality; recommender systems; statistical modelling; voting advice applications

1. Introduction

Voting advice applications (VAAs), lately described as "democratic recommender systems" (Berdoz et al., 2025), contribute to political education in multi-party democracies by recommending candidates or parties to voters (Garzia & Marschall, 2019; Tromborg & Albertsen, 2023). During elections, voters complete specifically designed questionnaires to identify their closest matches among a set of political actors (Louwerse & Rosema, 2014). In this way, VAAs inform citizens about pre-election party positions (Schwarz et al., 2010), which has been shown to positively affect turnout, vote choice, and issue knowledge (Garzia et al., 2017; Ladner et al., 2012; Munzert & Ramirez-Ruiz, 2021).

While the design of VAAs has been extensively studied (Bruinsma, 2020; Buryakov et al., 2024), there has been limited work published on users' primary concerns: the *quality* of their recommendations. Existing work on recommendation quality mainly focuses on the algorithmic perspective, i.e., on the robustness of the matching function (Berdoz et al., 2025) or the impact of scales (Rosema & Louwerse, 2016). However, it remains unclear how accurate the recommendations are if users submit incomplete questionnaires. This is the case for approximately 76% of Smartvote users, the most popular VAA in Switzerland, where 34% of users answer less than 30 out of 75 questions. Bachmann et al. (2024) approached this topic by introducing an adaptive question selection mechanism to increase the accuracy for early dropouts. Still, they did not provide a method to estimate the certainty of early recommendations. The question of whether and how such an estimate influences users and what measure of certainty they would appreciate most seems to be a gap in the literature, especially in candidate-based VAAs, where the choice is significantly more complex.

The contributions of this article are two-fold: First, we define the quality of early recommendations as the overlap with the final recommendations and develop an algorithm that estimates this metric before all answers are known by predicting users' remaining responses. Through simulations based on existing voter data obtained from Smartvote (Politools, 2023), we evaluate the precision of the resulting *recommendation certainty*. Second, we design and experimentally test three interfaces that communicate the certainty forecast to users. Inspired by Law et al. (2016), some of the interfaces are designed to trigger curiosity by displaying a preview of the most likely candidate recommendations. To evaluate these interfaces, we conduct a controlled experiment where users can finish the questionnaire when they feel they have answered enough questions. Together, these two contributions address the following research questions:

> RQ1: How precisely can our algorithm estimate the recommendation certainty?
>
> RQ2: How does the display of the recommendation certainty affect user behavior in candidate-based VAAs?
>
> RQ3: Which interface works best to communicate the recommendation certainty comprehensively to users?

The results of the experiments demonstrate that our method can predict individual progressions of the recommendation certainty, capturing sudden changes in the response pattern of each user. Moreover, the preview of stable recommendations increased the number of questions that users answered before exiting the questionnaire, therefore enhancing engagement. Conversely, users dropped out earlier when the

displayed certainty was artificially high, indicating that users consider recommendation certainty when deciding whether to answer another question. These findings motivated us to develop *semantic progress indicators* in sequential questionnaires that estimate the level of completion for each user individually, rather than the typical static progress bar. User feedback collected in the post-survey revealed that users understand simpler interfaces better than more accurate ones—creating a trade-off for VAA developers when designing semantic progress indicators for VAAs in practice.

2. Related Work

VAAs are widely used in multi-party democracies (Garzia & Marschall, 2019; Germann & Mendez, 2016). In more than 10 European countries, at least 10% of the population regularly engage with such online tools before elections. The German Wahl-O-Mat has by far the largest audience, with almost 26 million uses in 2025. Controlling for the population size, Norway has the highest share of VAA users, with over 60% of voters consulting a VAA before the election in 2021. Underlining their importance, it has been shown across many studies that VAAs impact reported election turnout (Germann & Gemenis, 2019; Munzert & Ramirez-Ruiz, 2021).

2.1. User Studies in VAA Research

Several randomized field experiments were conducted to measure the effects of VAAs on affective polarization, political knowledge, or vote intention (Garzia et al., 2017; Pianzola et al., 2019). However, this recruiting approach often suffers from self-selection bias (Ladner et al., 2012). To the best of our knowledge, few controlled user experiments were implemented. Bruinsma (2020) investigated the effect of visualizations in VAAs and found that many users had difficulties understanding them, regardless of their political interest or graphical knowledge. Kamoen and Liebrecht (2022) ran controlled user experiments to evaluate their large language model-enhanced conversational agent VAA. Users who interacted with the conversational agent VAA reported higher political knowledge and answered more factual knowledge questions correctly. We contribute to the body of work by conducting a controlled user experiment that investigates the effect of visualizing recommendation certainty to VAA users (Kay et al., 2016).

2.2. VAAs as Democratic Recommender Systems

The effect of the matching algorithm in VAAs has been intensively explored. It was often found that the distance function to compute matches drastically affects recommendations (Berdoz et al., 2025; Louwerse & Rosema, 2014). While such distance functions are most frequently used, a different approach is to learn recommendations based on response patterns of other voters (Romero Moreno et al., 2022). Social VAAs integrate a network approach into the matching algorithm by learning preferences based on other users' recommendations (Agathokleous & Tsapatsoulis, 2016; Katakis et al., 2014). This connects VAA research to collaborative filtering (Elahi et al., 2016; Rubens et al., 2015) and neighborhood-based methods of recommender systems (Lü et al., 2012). Such systems are compelling in e-commerce, where they predict final preference profiles as responses trickle in. We use a statistical method from collaborative filtering to estimate the recommendation certainty in VAAs.

2.3. Statistical Models for Political Data

Political scientists have developed several statistical models that visualize candidates or voters in a low-dimensional space (Clinton et al., 2004; Poole & Rosenthal, 1985). These models take features (such as roll-call data, survey responses, or social media interactions) and identify the ideal points of both the political actors and the features. If an actor is embedded close to a feature, they are more likely to agree with it. However, missing values are a considerable problem for most of these models. Often, mean imputation is used for simplicity, while item-response theory frameworks yield more nuanced results, especially for political data (Bachmann et al., 2024). Iterative algorithms, such as iterative principal component analysis (Grung & Manne, 1998), multiple imputation by chained equations (Buuren, 2012), or variational autoencoders (McCoy et al., 2018), do not yield a single maximum-likelihood imputation. Instead, they treat missing entries as latent variables and iteratively update or sample them using approximate posterior distributions. In our case, we adopt an iterative procedure that samples personalized completions of partially filled questionnaires based on users' inferred positions in the political space.

3. Methods

We propose a novel algorithm to estimate the recommendation certainty throughout a candidate-based VAA questionnaire. The core of this method is a statistical model that can predict users' missing answers based on their responses to the initial questions. This is necessary to gauge users' final recommendations. In this section, we explain the details of the statistical model and describe the estimation algorithm in more detail.

3.1. Statistical Model

In the trade-off between expressiveness and interpretability, we chose a two-dimensional latent-space model where the dimensions are learned from the distribution of the training data, rather than being predefined by human experts. Specifically, we follow the approach of Potthoff (2018) and Bachmann et al. (2025), where the statistical model is a combination of principal component analysis (PCA) and logistic regression (LR). Based on PCA projection of the candidates' answers into the latent space, an LR is learned for each question. This training procedure results in two ideal-point parameters per candidate (x_1, x_2) and three LR parameters per question $(\beta_1, \beta_2, \alpha)$. The likelihood of answering "yes" to question q (given a position x in the latent space) is defined as:

$$\hat{y}_q(x) := P\left(y_q = 1 \mid x\right) = \sigma(x^T \beta_q + \alpha_q)$$

Where σ is the sigmoid function:

$$\sigma(t) = \frac{1}{1 + e^{-t}}$$

And $x^T \beta$ is the scalar product of the candidates' position and question parameters. To embed new users into the latent space, we can then use their given answers to find the optimal position $\tilde{x}$. This optimal position (maximum likelihood estimate) minimizes the distances of the real given answers y_Q and the models' predictions $\hat{y}_{q \in Q}(x)$ such that:

$$f\left(x; y_Q\right) := \prod_{q \in Q} \left(1 - \left|\hat{y}_q(x) - y_q\right|\right)$$

Is maximal at $f(\tilde{x})$. Normalizing this function $f(x; y_Q)$ with respect to x then leads to the posterior distribution:

$$g(x; y_Q) := P(x \mid y_Q) = \frac{f(x; y_Q)\ \mathcal{N}(x; \mu, \Sigma)}{\int f(x; y_Q)\ \mathcal{N}(x; \mu, \Sigma)\ dx}$$

Where we include a Gaussian prior $\mathcal{N}(x; \mu, \Sigma)$ with mean μ and covariance matrix Σ to avoid the user positions moving to infinity. Note that if the set of answers is incomplete, the missing values are ignored for the embedding. This ensures more nuanced embedding for sparse data.

3.1.1. Discretization

By embedding users into the latent space, we can predict all remaining questions for a user by combining their position and the questions' likelihoods as previously described. However, minimizing the posterior distribution to obtain the maximum likelihood estimate is not analytically solvable. We, therefore, discretize the space into a grid of 200×200 points. We choose this grid size because it is sufficiently expressive and reasonably small, allowing for efficient computation. Figure 9A in the Supplementary File shows, as an example, the distribution of candidates' positions in the latent space and visualizes the likelihood function.

3.1.2. Sampling

The posterior distribution $g(x; y_Q)$ indicates the latent ideology of the response pattern y_Q (see Figure 9 in the Supplementary File). However, given the Bayesian approach, each position has uncertainty. To predict missing values with variations (Buuren, 2012), we can draw samples of the posterior distribution instead of taking the maximum likelihood predictions defined by $\hat{y}(\tilde{x})$. Then, we predict the remaining answers for each of the sampled positions to obtain many possible completions.

3.2. Recommendation Certainty

In candidate-based VAAs, users receive as recommendations the set of k candidates with the smallest distance to their response profile y_q (k-nearest neighbors). This distance is computed using the L_1 norm (or "Manhattan" distance) of the user's and candidates' answers:

$$D_M(c, y_Q) = \sum_{q \in Q} \left| y_q - c_q \right|$$

If the number of answered questions is less than the total number of questions ($Q < 75$), the remaining candidates' answers are ignored. We refer to the k candidates that minimize this distance to the response pattern y_Q of the user as *early recommendations* R_Q. However, through the predictions of the statistical model, it is possible to impute the remaining answers of the user. If these predictions are included in the distance computation, we refer to them as *predicted recommendations* R_P. After completing all questions, users receive their *final recommendations* R_F. Note that $R_Q = R_P = R_F$ if $Q = 75$. Figure 1 provides an overview of these three recommendation types.

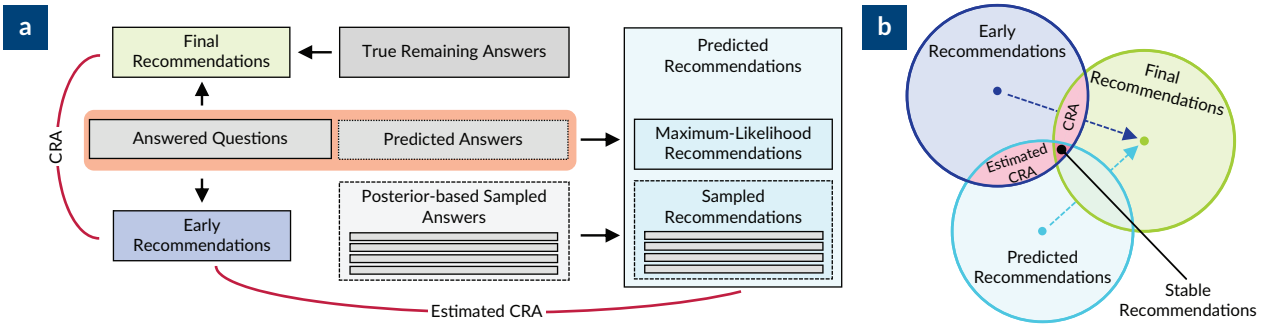

Figure 1. Schematic sketch of the recommendation types and the candidate recommendation accuracy (CRA) estimation algorithm. Notes: (a) The CRA is computed as the overlap of the *final* and *early recommendations*; it can be estimated using the *predicted recommendations*, which are informed by sampling the remaining answers; (b) At the beginning of the questionnaire, all types of recommendations are distinct sets without overlap; throughout the questionnaire, the early and predicted recommendations move towards the final recommendations; if the overlap of early and final recommendations increases (CRA), the overlap of early and predicted recommendations increases as well (estimated CRA); the overlap of all three types is called *stable recommendations*.

3.2.1. Candidate Recommendation Accuracy

The candidate recommendation accuracy (CRA) quantifies the quality of early recommendations R_Q as their overlap with the final recommendations R_F. More generally, the CRA of any set of candidates R is given by its overlap with R_F:

$$\omega\left(R; R_F\right) = \frac{\left|R \cap R_F\right|}{k}$$

where $k = |R_F|$ is the number of candidates per recommendation. Here, the final recommendations serve as "ground truth," i.e., the optimal set of recommendations possible within the framework of a questionnaire-based VAA—even though they might not globally be the best set of candidates for a user.

3.2.2. Estimation Algorithms

Calculating the CRA is only possible *a posteriori*, i.e., when all questions are answered, and the final recommendations are known. To estimate the certainty of early recommendations *during* the questionnaire, we use the predicted recommendations as a proxy for the final recommendations. As shown in Figure 1a, we calculate the overlap between the early and predicted recommendations to estimate the CRA. This does not imply that the predicted recommendations are necessarily correct. As Figure 1b indicates, the early and predicted recommendations approach the final recommendations from different directions. Even if both are equally false, their overlap estimates *how false* both are.

We evaluate two approaches to predict the remaining answers for CRA estimation: In the first approach, we use the maximum likelihood predictions. We call this approach One-Shot. The estimated CRA is then given by $\omega(R_Q; R_P)$. In the second approach, we use sampling to generate variations in the predictions (Posterior). Here, the CRA estimate is given by the average overlap:

$$\frac{1}{M}\sum_{m}^{M}\omega\left(R_m; R_Q\right)$$

Where R_m is the set of recommended candidates obtained from the sample m, and ω is the CRA defined above. This estimate thus captures how similar the candidate recommendations are across different plausible completions of the user's answers. We compare both estimation algorithms to the baseline of user-agnostic estimations, such as a linearly increasing estimate (Linear) and aggregated CRA values from the Smartvote data (Historic). The Linear estimate corresponds to a static progress bar increasing from 0% to 100% in even steps, while the Historic estimate adjusts for the average CRA progression of existing user profiles (see Figure 2b). We chose these baselines as they are straightforward alternatives to displaying the recommendation certainty to users.

3.2.3. Stable Recommendations

Lastly, we define *stable recommendations* as those candidates in the early recommendations that will also appear in the final recommendations. They can be used in the preview of recommended candidates, where it is essential to display only a few false positives. We compare two approaches to select stable recommendations and evaluate them by the number of true and false positives: In the first approach, we select the overlap of the early and predicted recommendations. This selection of size $|R_P \cap R_Q|$ is initially empty and then grows as the early and predicted recommendations converge to the final ones. The advantage of these *estimated recommendations* is their fast computation with just one maximum-likelihood prediction. In the second approach, we use sampling to add variance to the predicted recommendations. Then, we compute the frequency with which each candidate appears in the *sampled recommendations* R_M. If this frequency exceeds a certain threshold, we select the candidate. A shortcoming of this approach is that the threshold must be learned from previous user interactions. However, it is convenient that the candidates' frequencies converge to the sampled CRA estimate when sufficiently increasing the sample size.

4. Evaluating the CRA Estimation Algorithms

In the first study, we simulate the progression of recommendation certainty for existing voters in the Smartvote data and evaluate the estimation algorithms based on their accuracy. In this section, we describe the data used, the framework, and the simulation results.

4.1. Data

The Smartvote data were obtained during the Swiss National Elections in 2023 (Politools, 2023). They include the answers of candidates and voters from Zurich—the largest canton in Switzerland in terms of population—to 75 political questions, recorded on a Likert scale with options ranging from 4 to 7. We convert these ordinal responses into a continuous scale from 0 (*complete disagreement*) to 1 (*complete agreement*), evenly spacing the remaining options across this range. In addition, the data include personal information about the candidates (e.g., party affiliation, mottos, budget, age). In total, we use the data of 1,029 Zurich-based candidates, where we focus on candidates from the eight major parties (represented in the Federal Assembly) and categorize candidates from smaller parties under "Others." Detailed information on these parties can be found in Table 3 in the Supplementary File.

Additionally, to accurately evaluate the precision of the recommendation certainty, we generate a representative sample from the voters in the Smartvote data. Our sampling procedure involves iterating through each combination of gender, age group, and political position, selecting the appropriate number of voters to obtain 1,122 representative individuals. This approach ensures that our voter data accurately reflect the demographic and political landscape of the canton of Zurich and are comparable to the sample in our user experiments. We consulted demographic data from Urbistat (2025) and election results from the Swiss Federal Statistical Office (2023). Figure 16 in the Supplementary File shows the distribution of voters in the representative sample.

4.2. Simulation Framework

We use the representative voter sample from the Smartvote data to identify the best-performing estimation algorithm through a simulation. As described in Section 3.2.1, the final recommendations are based on all voters' answers as their k-nearest neighbors in the candidate's data. We consider that $k = 36$ as this is the number of candidates per recommendation in the canton of Zurich (for comparison, the results for $k = 10$ are provided in the Supplementary File in Table 4 and Figures 10 to 14). For each user, we iterate through the questions and compute the early and predicted recommendations after each answer. To predict the recommendations, we compare the different approaches explained in Section 3.2.2. We then compare the true CRA, the estimated CRA, and the set of stable recommendations (see Section 3.2.3). For the sampled recommendations, we consider thresholds $t \in \{0.1, 0.2, \ldots, 0.9\}$. Together, this simulation addresses RQ1, exploring whether our algorithm can accurately forecast the recommendation certainty.

4.3. Simulation Results

The simulation was implemented in Python and run on a Mac with an M1 processor. Using the *sklearn* k-nearest-neighbor tree, all algorithms computed the recommendation certainty in less than 90 ms per estimate. The number of samples did not drastically impact this duration. For example, it took 18.1 ms to calculate the recommendation stability based on the maximum-likelihood estimate, while it only took 82.9 ms to do the same with 1,000 Posterior samples. The fastest methods were the user-agnostic estimates with less than 1 ms look-up time. This puts all times well below the threshold of 0.1 s for seeming instantaneous reactions (Card et al., 1991).

4.3.1. Error of the Estimated CRA

We tested four different approaches to estimate the CRA throughout the questionnaire. As seen in Figure 2a, the average error depends on the number of questions answered. All estimation algorithms start with an error of 4% which increases as users answer more questions. The best-performing approach, 1,000 Posterior samples, then reaches a plateau after seven questions with an error of 8.7%. For fewer samples, the error is slightly higher. In contrast, the maximum-likelihood approach, One-Shot, has a maximum error of 10.7% after seven questions. Still, this result is better than the baseline approaches, Linear and Historic, which have an average error of 13.8% and 11.5% respectively, at this stage.

Table 1 quantifies how accurately the algorithms estimate the true CRA averaged across all users and questions. The worst-performing approach, Linear, has an average error of 10.22%, while the

best-performing algorithm, 1,000 Posterior samples, is on average 6.28% off. Here, the number of samples significantly affects the performance. We find that more samples have smaller errors, fluctuate less, and require longer computation times. While the decrease of the error becomes very small for sample sizes larger than 100, the paired t-test still shows that they are significant ($p < 0.001$). Of all deterministic algorithms, the maximum likelihood estimate reaches the best results with an average error of 7.27%, which is significantly better than the baselines of Linear and Historic ($p < 0.001$).

Table 1. Fluctuation, error, and duration for different estimation algorithms.

Algorithm	Method	Fluctuation (%)	Error (%)	Duration (ms)	Recommendation
CRA	Ground truth	3.35 ± 0.02			Only a-posteriori
Historic	User-agnostic	1.25%	9.33 ± 0.12	< 1 ms	Requires user data
Linear	User-agnostic	1.33%	10.22 ± 0.14	< 1 ms	Linear progress bar
One-Shot	Maximum-likelihood	3.90 ± 0.02	7.27 ± 0.08	25.1 ± 0.3	Includes estimated recs.
Posterior	1 sample	7.44 ± 0.04	7.95 ± 0.07	25.9 ± 1.1	Includes sampled recs.
Posterior	10 samples	3.63 ± 0.02	6.47 ± 0.07	26.6 ± 0.7	Includes sampled recs.
Posterior	100 samples	2.88 ± 0.01	6.30 ± 0.07	31.6 ± 1.0	Includes sampled recs.
Posterior	1,000 samples	2.78 ± 0.01	6.28 ± 0.07	82.9 ± 4.1	Includes sampled recs.

Notes: The fluctuation is computed as the sum of absolute successive differences; a higher value means the estimations are less smooth; the error gives the mean absolute difference between the estimated CRA and the true CRA; the duration includes computing the estimated CRA and stable recommendations; all values are means and their standard errors.

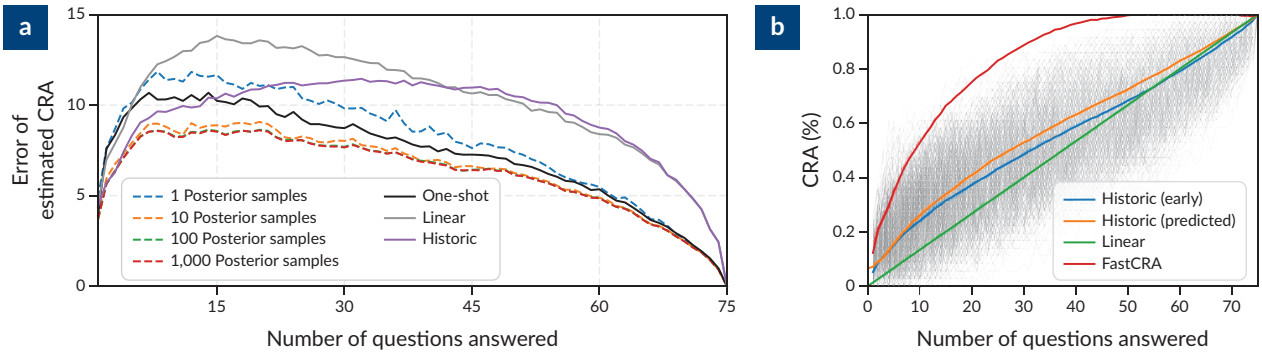

Figure 2. Mean absolute error of the estimation algorithms compared to the true CRA. Notes: (a) For all estimation algorithms, the error increases from 4% to around 7–12% after seven questions; the user-agnostic predictions, Linear and Historic, perform worse than Posterior and One-Shot; (b) Each voter has an individual progression of the true CRA (gray lines); the Historic estimates correspond to the average CRA across all voters.

4.3.2. Fluctuation of the Estimated CRA

Defined as the sum of absolute differences between the current and the previous estimated CRA, a high fluctuation indicates that the estimates strongly vary from question to question. However, as shown in Table 1, the true CRA also fluctuates with a mean value of 3.35%. A similar fluctuation is achieved by One-Shot with a value of 3.90%. Sampling reduces the fluctuation to 2.88% and 2.78% (for 100 and 1,000 samples, respectively). This difference is statistically significant with a p-value of $p < 0.001$ in a paired t-test.

The lowest fluctuation is achieved by user-agnostic estimates. Linear estimates achieve 1.33%, while the Historic achieves 1.25% (note that the initial estimate here is larger than 0%).

4.3.3. Precision of the Stable Recommendations

In Figure 3, we inspect the empirical relation between the Early and Predicted recommendations. Initially, these two sets are distinct from each other (and from the final recommendations). As users answer more questions, all sets increasingly overlap until they are finally identical. Indicated by the blue and orange lines, both Early and Predicted converge to the final recommendations at a similar pace. The number of stable recommendations (those candidates who are part of all three sets) monotonically increases from 0 to 36, as evidenced by the pink line. This indicates that even after 15 questions, 50% of the Estimated recommendations are stable. Already after 35 questions, 75% of them are true positives. We therefore find that the predictions uncover different but similarly accurate recommendations compared to the early ones.

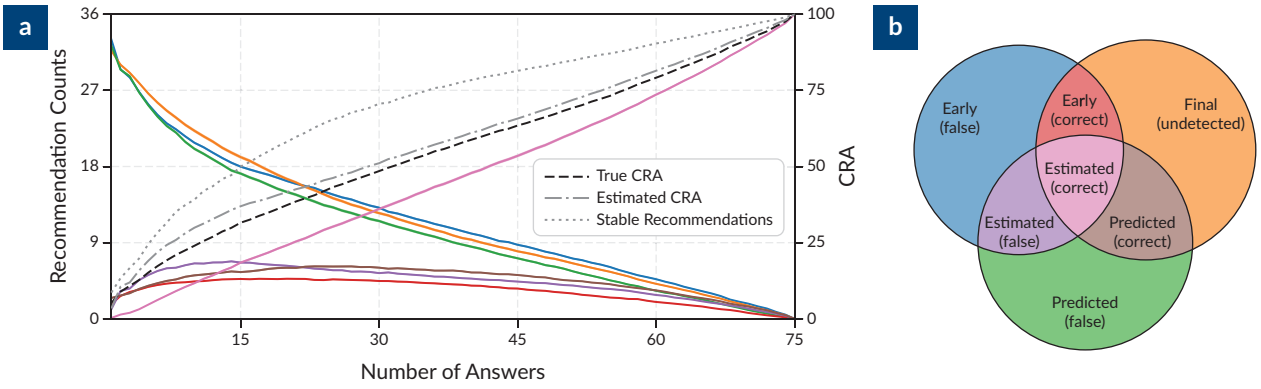

Figure 3. Venn diagram of the early, predicted, and final recommendations. Notes: (a) The blue, green, and orange lines show the number of candidates that are only in one of the three sets of recommendations; they decrease from 36 (when all sets are distinct) to 0 (when all sets are identical); the pink line shows the intersection of all three sets; the dotted line shows the fraction of stable recommendations in the estimated recommendations; (b) In the corresponding Venn diagram, the true positives (correct) and false positives (false) are colored according to the lines in the graph.

4.3.4. Optimal Thresholds for Sampled Recommendations

Figure 4 shows the number of true and false positives for each of the three approaches. While Early recommendations contain most true positives, they start overconfidently with around 35 false positives. In contrast, Estimated recommendations initially only select around three true and seven false positives. Sampled recommendations further reduce the number of false positives, such that, especially at the beginning of the questionnaire, there are very few false positives. Here, we identified a sample size of $N = 100$ and a threshold of $t = 0.7$ to be optimal (see Table 9 in the Supplementary File). As shown in Figure 4b, during the first five questions, the difference between true and false positives is minimal and has a low variance across voters. The bar at the bottom of the figure shows that the improvement of Sampled is statistically significant for the first 25 questions ($p < 0.001$). Then, the effect disappears, and the performance of both Sampled and Estimated equalizes.

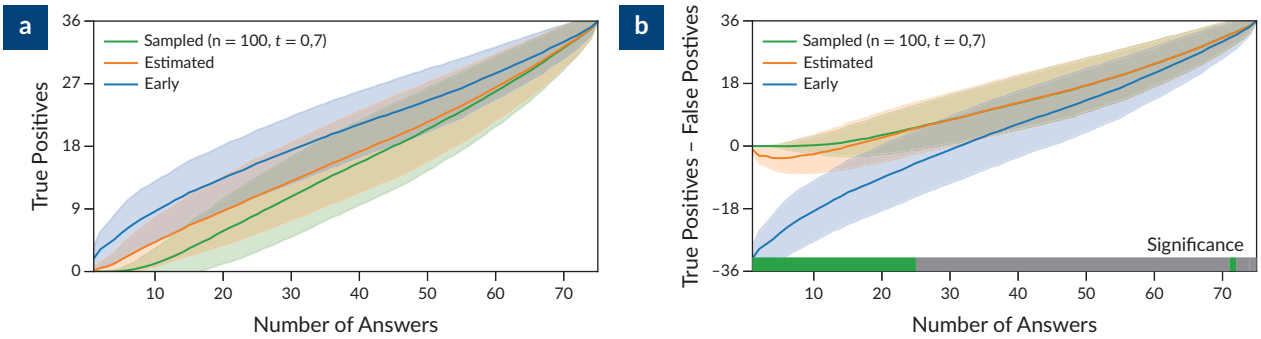

Figure 4. The precision of the stable recommendations for three different selection algorithms. Notes: (a) The y-axis shows the number of true positives per question across all voters (mean and standard deviation); the Early recommendations constantly have a higher score than the Sampled and Estimated ones; (b) The y-axis shows the difference between true and false positives per question across voters (mean and standard deviation); the Early recommendations start with many false positives; the horizontal bar shows where the difference between the Estimated (orange) and Sampled (green) recommendations is statistically significant.

5. Evaluating the Display of Recommendation Certainty

To evaluate the effects of displaying the estimated CRA and stable recommendations to users, we conduct a controlled user experiment on a self-developed VAA platform. Here, we implement the best-performing estimation algorithms and three different user interfaces within the platform, communicating the recommendation certainty to the experiment participants. This section presents the interfaces, the experimental setup, and the results of the user experiment.

5.1. Platform Design

To run the experiment, we developed a web-based platform specifically designed to run controlled experiments on user behavior in VAAs. It consists of two main components: the survey page and the candidate recommendation page. On the survey page, users react sequentially to the Smartvote questions. These reactions are on a 5-point Likert scale, including a neutral option. On the candidate recommendation page, users see the top matches with their party affiliation, similarity score, and position in the political map (see Figure 15 in the Supplementary File). Users can select which matches seem relevant to them. For simplicity, the number of candidates on the recommendation page is always limited to a relatively small set of $k = 10$ candidates. This way, users who are unfamiliar with VAAs are not overwhelmed by a large selection. A live online demo of the platform is accessible here: https://aqvaa.ifi.uzh.ch/survey/3xqPQz

We extend the survey page of the VAA platform with three semantic progress indicators: The first interface displays the estimated CRA with a progress bar below each question (see Figure 5). In contrast to a standard progress bar, this estimated CRA adapts to the users' answers and may increase or decrease after each question. The second and third interfaces display a set of stable recommendations below each question (see Figure 6). Depending on the estimation algorithm (Estimated and Sampled), the interface is slightly different. As shown in Figure 6a, the Sampled recommendations are shown together with their probability given by percentages. Only for the candidates whose frequency is above the threshold t are party and name displayed in a blue box. For candidates with a lower frequency, a question mark is displayed in a yellow box.

This is designed to spark curiosity while keeping the number of false positives minimal. As shown in Figure 6b, the third interface merely shows the Estimated recommendations in blue boxes without indicating their stability and corresponding percentage. Both these interfaces show a grey box for the fields that cannot yet be filled with stable recommendations.

Note that for all interfaces, we add a textual explanation that reads: "As you answer more questions, your recommendations become more stable." This should encourage users to answer more questions, increasing the recommendation certainty. In addition, we disable the standard progress bar that typically shows the number of answered questions at the top. This way, we avoid people feeling urged to complete all 75 questions.

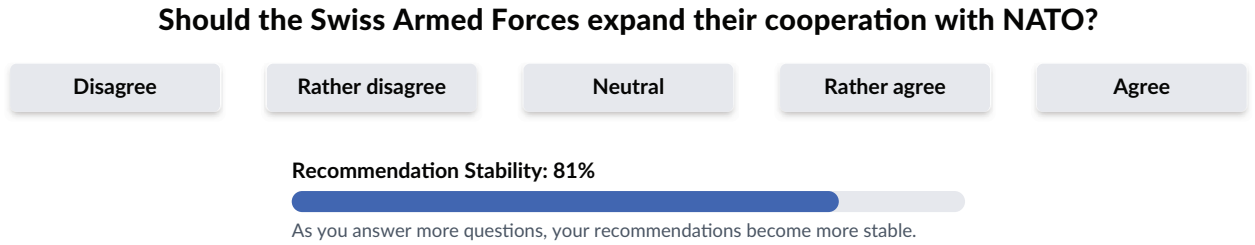

Figure 5. Interface of the estimated CRA in the VAA platform. Notes: At each question, the estimated CRA is displayed in the progress bar; depending on the response pattern, the certainty can increase or decrease with each answer; the question and response buttons are the same for each different interface.

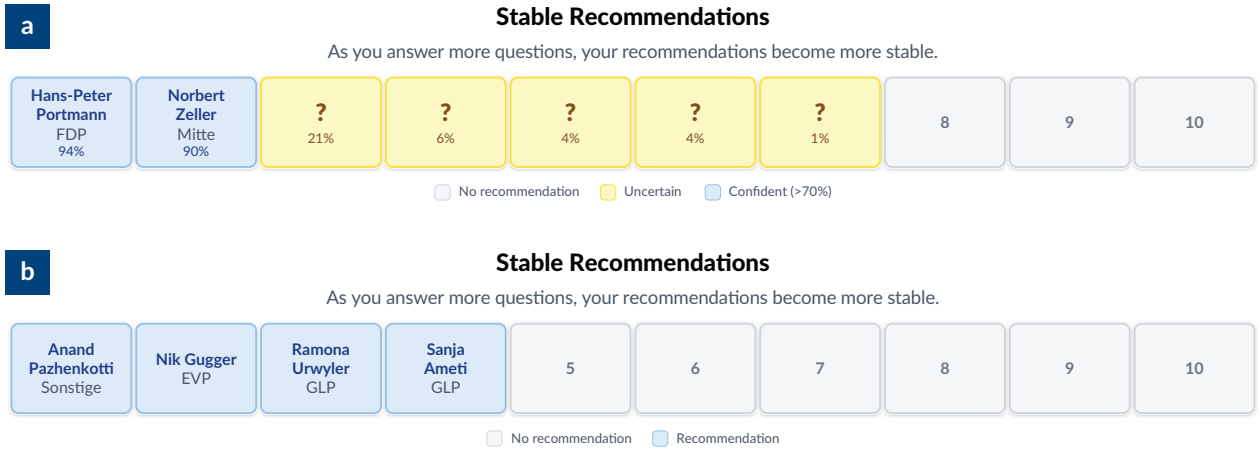

Figure 6. Interfaces of the stable recommendations in the VAA platform. Notes: (a) For the sampled recommendations, the algorithm computes the probability of each candidate being among the final recommendations; the name and party are displayed if the percentage is above a certain threshold; if it is below the threshold, just the probability is displayed; for the remaining recommendations, a gray box is shown; (b) For the estimated recommendations, only the overlap of early and predicted recommendations is highlighted in blue.

5.2. Experimental Setup

The following user experiment was reviewed and approved by the human subjects committee of our faculty. As shown in Figure 7, users start with a pre-survey for screening purposes. These questions, listed in Table 5 in the Supplementary File, collect demographic variables. Then, users are forwarded to our platform, where they answer the VAA questionnaire with the condition-specific interface. After answering a minimum of

15 questions, a button “Show recommendations” appears. By clicking this button, users can, from then on, exit the questionnaire to view their candidate recommendation page. A pop-up explains to the users that they can do so whenever they feel that they have answered enough questions. On the recommendation page, users then rate their recommendations for the first time (Q_1). After that, they must answer at least another five VAA questions before they can click “Show recommendations” for a second time (Q_2). This time, proceeding to the recommendation page terminates the questionnaire. Again, we explicitly explain this procedure with a pop-up. After submitting the second rating, users are forwarded to a post-survey, which collects information about their perception of the interface. The post-survey questions are listed in Tables 6 and 7 in the Supplementary File.

Users are randomly assigned to one of five conditions: With the first and second conditions, we address RQ2, exploring whether the displayed CRA estimate influences the point of dropout (Q_2). Both these conditions share the same interface with the progress bar displaying the recommendation certainty as shown in Figure 5. In one condition, however, users see the true estimated CRA (TrueCRA) while in the other, they see an artificially fast-growing estimated CRA (FastCRA). This way, we can evaluate the influence of the displayed value. With the third and fourth conditions, we address RQ3, investigating whether users perceive simpler interfaces as easier to understand than more accurate ones. Here, we display the stable recommendations to the users with the interfaces shown in Figure 6. In one condition, users see the estimated recommendations without percentages (Estimated), while in the other, users see the sampled recommendations with displayed probabilities for each candidate (Sampled). We set the threshold for these probabilities to $t = 0.7$ as preliminary results showed that this threshold maximizes the difference between true and false positives. Lastly, in the control group, users do not see any information about the recommendation certainty (Control).

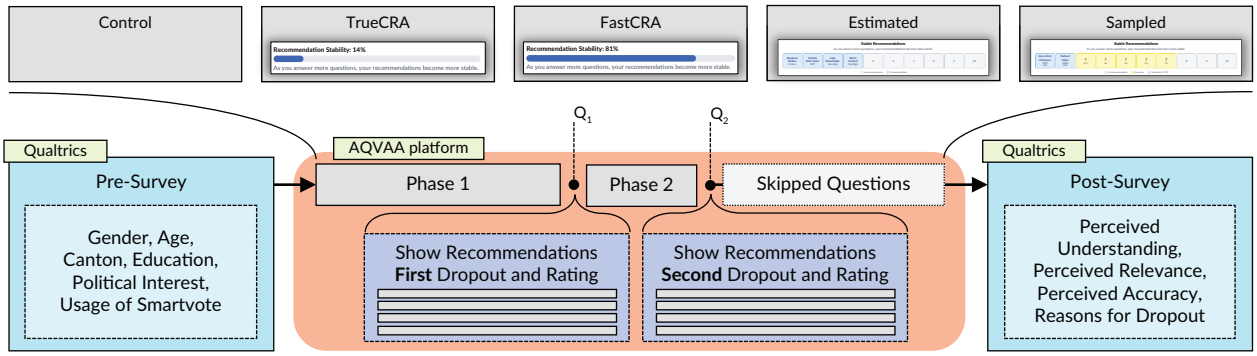

Figure 7. Flowchart of the user experiment. Notes: Participants start with a pre-survey on Qualtrics, where they are screened for demographic variables; then, they are assigned to one of 5 groups—Control, TrueCRA, FastCRA, Estimated, or Sampled—and forwarded to the VAA; on the VAA, users answer a minimum of 15 questions; then they can freely decide to see their recommendations (Q_1); after rating the recommendations, users proceed to answer more questions; after a minimum of 5 additional questions, they can again freely terminate the questionnaire to see their updated recommendations (Q_2); after rating them, users proceed to the post-survey, evaluating the interface and providing reasons to skip the remaining questions.

5.3. Results of the User Experiment

We recruited 130 participants from the Canton of Zurich through the market research company Bilendi. All participants (53 men and 77 women) completed the survey between the 24th and 30th of July 2025. The median survey duration was 14:12 minutes, where participants from the Estimated and Sampled condition took significantly longer than the control group ($p = 0.020$). We excluded from the analysis 27 users for one of the following reasons: either they did not understand the instructions (they stated that they completed all 75 questions to get the full financial incentive or were not aware that early dropout was possible), or they had an unreasonable duration of the survey (faster than 5 min or slower than 120 min). Despite random assignment to the experiment conditions, users in one condition (Estimated) report a significantly higher political interest than the Control group ($p = 0.011$). In our statistical analysis, we therefore control for political interest using an OLS model and an ANCOVA with the corresponding pairwise Tukey's HSD test.

5.3.1. Answered Questions Before Dropping Out

On average, Smartvote users answer 57 questions before proceeding to their candidate recommendations. In the Control group of our experiment, users answered on average 56 questions before dropping out. Across conditions, this number differs between participants: 88 users (67.7%) filled out all 75 questions, while 6 (4.6%) quit the questionnaire directly after the minimum of 20 questions. The rest was evenly distributed in between (see Figure 17 in the Supplementary File). We investigate whether the displayed recommendation certainty influenced participants' decision to finish the questionnaire. As Table 2 shows, Estimated has a significantly later dropout than the Control group ($p = 0.005$). Furthermore, TrueCRA, Estimated, and Sampled have a significantly later dropout than FastCRA ($p < 0.001$).

5.3.2. Perceived Influence of the Display

Contrary to the significantly different number of answered questions, users did not perceive that having the certainty displayed influenced their decision to click "Show recommendations." As shown in Table 2, most users reported that the certainty display had a neutral influence on them (with a slight negative tendency ranging from −0.7 to −0.2). However, the displayed recommendation certainty at dropout was lower for FastCRA than for the other three conditions. These differences, however, are not statistically significant.

Participants selected different reasons in the post-survey to explain their dropping out (a list of the pre-defined options is given in Table 7 in the Supplementary File). Users who quit early were mainly "satisfied with the certainty" (25%) or "looked at the certainty" (25%). Others saw "no incentive to continue" (25%) or gave other reasons (25%). In the open-ended feedback, one user wrote that "in hindsight, [they] would have done the full questionnaire." Those users who completed the whole questionnaire mainly did so "out of curiosity" (41%). Some wanted to have the recommendations "as precise as possible" (28%) and "looked at the certainty" (10%). In the open-ended feedback, one user reported that they "did not want to be influenced by any information." Another wrote: "The more information is available, the more precise (and therefore relevant) are the recommendations."

Table 2. User evaluation of the interfaces in the VAA platform.

Condition	Dropout	Final CRA	Displayed	Understanding	Influence	Relevance	Accuracy	Usage
Control (21)	*56 ± 5*	*75 ± 7*						0.8 ± 0.4
FastCRA (17)	*49 ± 5*	*58 ± 8*	77 ± 6	*0.6 ± 0.5*	−0.4 ± 0.5	0.8 ± 0.4	0.6 ± 0.4	1.2 ± 0.4
TrueCRA (22)	**63 ± 4**	**86 ± 5**	86 ± 5	0.9 ± 0.4	−0.2 ± 0.4	0.7 ± 0.4	0.2 ± 0.4	0.6 ± 0.4
Estimated (21)	**68 ± 3**	**90 ± 4**		**1.5 ± 0.3**	−0.7 ± 0.4	0.9 ± 0.4	0.5 ± 0.3	0.9 ± 0.4
Sampled (20)	**65 ± 4**	**86 ± 6**		*0.3 ± 0.4*	−0.5 ± 0.4	1.3 ± 0.2	0.6 ± 0.3	1.1 ± 0.3

Notes: The table shows means and their standard errors; the dropout refers to the total number of answered questions; only in FastCRA did the displayed CRA estimate differ from the estimated CRA; the remaining columns are results from the post-survey; values in bold indicate a significantly higher result than that of another condition, and values in italic a significantly lower result than that of another condition.

5.3.3. Understanding of the User Interface

Users reported understanding the Estimated interface significantly more than Sampled ($p < 0.001$) and FastCRA ($p = 0.018$). Perceived relevance and accuracy of the interfaces were not significantly different. However, there is a tendency that Sampled were found to be more relevant than all other conditions, including Estimated. Users in the FastCRA reported the highest likelihood of recommending the website to a friend or family member. Their evaluation was closely followed by Sampled.

Users in the Control group did not see the interfaces during the questionnaire. In the post-survey, we showed them screenshots of all interfaces and asked users to rate them hypothetically. Users reported that they understood the interface with the certainty bar better than the interface with stable recommendations; they also found it more interesting and more accurate. However, the number of participants in the control group (21) did not allow for statistically significant results in a paired t-test ($p = 0.201$, $p = 0.108$, and $p = 0.367$).

5.3.4. General Thoughts of Participants

The users in our experiment were not typical VAA users; 54 (52%) had never consulted a VAA before. As one user put it: "It would be great if such a tool were publicly available during cantonal elections or similar." Yet, users' overall experience of our VAA platform was positive. As shown in Table 8 in the Supplementary File, this was explicitly expressed by 25 users in the open-ended feedback. For example, one user wrote that they "found it interesting to get recommendations based on [their] answers." Twelve users mentioned that they had learnt something using the VAA. One noted that they were "surprised, as a not very politically interested person, what kind of candidates were recommended." Seven users commented on the displayed recommendation certainty. Among those, four users critiqued the explanations of the certainty display. One said that while they "found the questionnaire good, the certainty display was slightly confusing at first." They thought that the certainty bar displayed the system's evaluation of their responses. Three other users highlighted the interface positively. One wrote: "Since there was no (normal) progress bar, I focused on the certainty bar and was happy."

6. Discussion

Overall, we found that users in our experiment appreciated the interfaces similarly and reported a high likelihood of recommending the platform. However, there are some differences across users and conditions that warrant further attention. We first address the results of the simulation, focusing on the accuracy of the algorithms and their variance across user types. We then address the results of the user experiment, focusing on the discrepancy between users' perception and their observed behavior on the platform.

6.1. Posterior-Informed Samples Predict Stable Recommendations

Our simulation results provided clear evidence that the recommendation certainty can be accurately estimated, addressing RQ1. The posterior-based samples yielded estimates that were, with an error of 7.95%, significantly more accurate than those of the heuristic baseline. Moreover, we found that the number of samples significantly affected the error and the fluctuation of the estimated CRA. In the trade-off between duration and performance, we therefore argue that the number of samples be set to $M = 100$ with a threshold of $t = 0.7$, which seems to be a point of diminishing returns.

The precision of the algorithms, however, was not noticed by all participants. Across conditions, users reported a similar perceived accuracy—even in FastCRA, which displayed an artificially high CRA estimate (see Figure 2b). Our interpretation is that users might have a vague view of "good" recommendations and therefore do not perceive false positives or overconfident CRA estimates as inaccurate.

6.2. Precision Differs Across User Types

The estimated CRA, as a measure of recommendation certainty, depends on users' individual response patterns. For example, Figure 8 shows the progression of the true and estimated certainty for user 3613 throughout the questionnaire. Despite sudden fluctuations (e.g., after five questions), the estimated CRA accurately follows the trend of the true CRA. However, there might be a larger error for other users (see

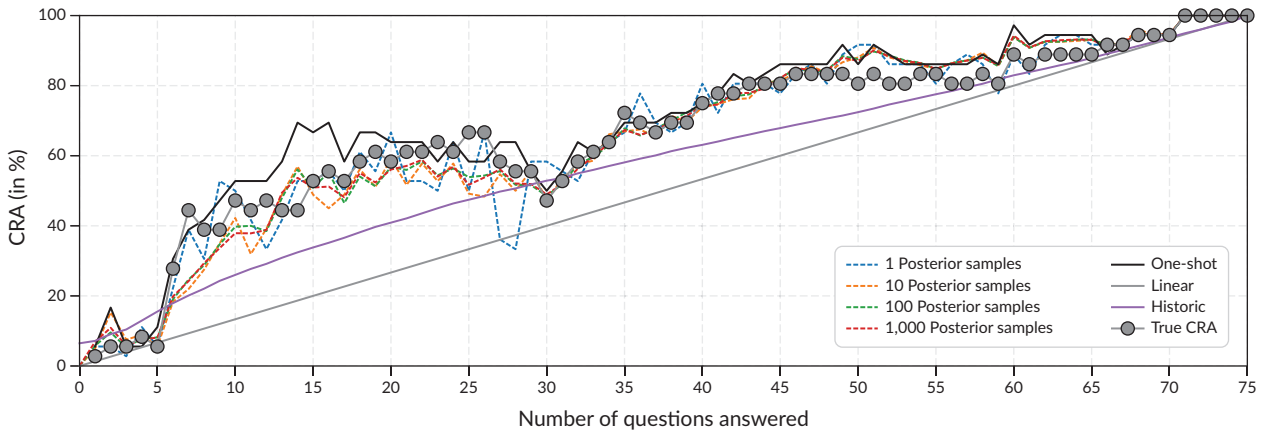

Figure 8. Progression of the true CRA compared to different estimation algorithms. Notes: The gray dots show the true CRA of user 3613; after questions 6 and 7, the CRA suddenly jumps to 42%; then, the increase in CRA is moderate until it drops again after 30 questions; for the remaining questions, the CRA increases almost monotonically until it reaches 100% after all questions; the colored lines indicate the estimated CRA for the different algorithms.

Figure 18 in the Supplementary File). We identified three possible explanations for this behavior: First, these users tend to answer the questionnaire with an atypical response pattern; second, they often answer the questions with relatively neutral stances; third, the number of candidates in their proximity is very high. If a user, for example, holds political views that few candidates share, these candidates will quickly emerge as the corresponding recommendations. In contrast, if many similar candidates could equally well be recommended, sampling the remaining answers affects the choice among these candidates, thereby reducing the certainty of the recommendations. This, however, is not due to the quality of the method, but is instead an artifact of the political landscape itself, which lacks a uniform density of candidates.

6.3. Curiosity Keeps Users in the Questionnaire

The user experiment demonstrated a significant effect of the displayed recommendation certainty on user behaviour, addressing RQ2. Even though users reported that the display did not influence the number of answered questions, they still dropped out significantly earlier in FastCRA than in TrueCRA. In addition to this result, we also found that users in Estimated and Sampled spent more time on the survey and answered more questions than users in the conditions with progress bars. We take this as an indication that stable recommendations sparked users' curiosity. Displaying intermediate recommendations may be a cognitive anchor for users to reflect more deeply on their responses. It remains to be investigated, however, whether such a display would influence their answers in a specific direction through some form of confirmation bias or reassurance. This concept is discussed in the literature as performative prediction and calls for future research in the context of political questionnaires (Mendler-Dünner et al., 2022; Perdomo et al., 2020).

6.4. Users Prefer Simple Over Accurate Displays

We furthermore found that users perceived their understanding of each interface differently, as inquired by RQ3. The display of Estimated recommendations, which just shows candidates' names and parties, was perceived as most understandable. The display of Sampled recommendations, which also shows individual candidates' certainties as percentages, was perceived to be the least understandable. As the percentages are the main difference between the interfaces, this result indicates that such additional information was difficult for users to interpret. However, while the Estimated interface is simpler, it is also less accurate, since the simulation results revealed more false positives in that condition. Interestingly, users across all conditions rated their interest in using the platform in a similar way, indicating that their understanding of the interface did not impact their experience.

6.5. Generalizability Beyond Candidate-Based VAAs

Lastly, we address the generalizability of the study. While we focused our analysis on candidate-based VAAs, the method can also be applied to party-based VAAs—or any application that recommends a target variable based on interactions in a sequential questionnaire. However, we identify two adjustments for party-based VAAs: First, the CRA needs to be changed to another metric, such as Spearman's rank correlation, as there are no "sets" of candidates but ranked lists with recommended parties or single recommendations. Moreover, the smaller size of the training data might limit the quality of the model's predictions. This can, however, be addressed by using historic voter-question interactions or by generating training data with large language models (Bachmann et al., 2025).

7. Limitations

Our study has several limitations. We divide them into limitations of the simulation framework and limitations of the user experiment.

7.1. Simulation Framework

The most apparent limitation of our method is that we measure the certainty as the overlap of early and final recommendations, which might not reflect nuances of the recommendation quality. First, this metric does not include the ranking of candidates, which might be important to some users. Second, we ultimately do not know what a "good recommendation" is, i.e., it might be that the final recommendations are not perceived as more relevant than early ones. However, we believe this approach is the best approximation of the ground truth, while acknowledging that users might perceive it differently.

We used a simplified simulation of the VAA to evaluate the CRA estimation algorithms. For example, our analysis focused on a specific size of recommendation set ($k = 36$ and $k = 10$). However, in some cantons, different numbers of candidates are recommended. While the overall trend of the results is the same for $k = 10$, we note that the CRA estimates are less accurate and fluctuate more for smaller k. Moreover, our choice of statistical model might not be optimal. While the combination of PCA and LR yields promising results, it entirely ignores the fact that users often prioritize certain questions (weighing) and that the ordinal responses in the original data do not necessarily map to the continuous predictions of the model. Lastly, a general limitation of our method is its dependency on training data. These data are specific to each election and questionnaire. Even if training data naturally arise in candidate-based VAA through the responses of the eligible candidates, the model's performance could be affected if there is a different response pattern in the candidates' and users' answers. This could decrease the predictive accuracy for voters.

7.2. User Experiment

There are also limitations to our user experiment. First, we designed the interfaces focusing more on technical accuracy than user friendliness. While we tried to provide an accessible platform, some participants got stuck on the candidate recommendation page and did not know how to proceed to the post-survey. Moreover, the participants were recruited through a market research company and received a financial incentive to complete the questionnaire, rather than for personal interest. We therefore had to exclude a substantial number of participants from the analysis, resulting in a relatively modest sample size, which is not representative of typical Smartvote users (who are younger and more liberal) and does not represent Zurich's political and demographic population (our participants are older and slightly more conservative). However, we argue that this exploratory research is valuable because it does not survey the typical Smartvote user base, who already use the tool.

A different experimental setup—such as a 2×2 study design—would have improved the evaluation of the interfaces. Instead of only showing an artificially fast-growing CRA in the certainty bar interface (TrueCRA vs. FastCRA), we could have also displayed an artificially large number of stable recommendations in the other interfaces. Such a setup would have allowed the analysis of cross-effects of both treatments. For a more complete understanding of the results, it would have been beneficial to collect more data on *why* people decided to drop out and see their recommendations. Instead, we collected reasons through a multiple-choice

post-survey. Lastly, we did not measure users' *actual* understanding of the interfaces; we just asked for their *perceived* understanding. Despite these limitations, the results we do have provide exploratory insights into the implementation of semantic progress indicators in VAAs and can be used to inform future research.

8. Conclusion

In this study, we explored the integration of a semantic progress indicator into a candidate-based VAA. We developed a method to estimate the certainty of early recommendations and implemented this algorithm with three different interfaces in our VAA. Through a controlled user experiment, we found that this additional information can spark curiosity and lead to prolonged user engagement.

The core idea of the study—using different types of predictions to estimate their accuracy—applies to many fields outside VAAs. While previous research has shown that the predicted recommendations are slightly more accurate than the early recommendations (Bachmann et al., 2024), the unexpected finding was that their overlap indicates the quality of both. In other words, the predicted recommendations uncover different candidates from the early recommendations if and only if both sets are far from the final set. This geometric phenomenon makes the topic interesting from a mathematical perspective.

From a political perspective, it was interesting that users cared more about the interface in which the estimate was displayed than about its accuracy. This finding suggests that, for interface design, simplicity is often preferable, even at the expense of technical precision. As a promising trade-off, the preview of estimated recommendations emerged—an interface that was well understood, while having satisfactory properties regarding true and false positives.

Thinking ahead, this research on estimating the CRA not only informs the feature design in candidate-based VAAs but can also help to personalize adaptive strategies through the model's predictions of users' individual progression of the CRA. The different types of CRA curves (smooth, surprising, and unpredictable) could be formalized into a typology of user profiles and contribute to political research on citizens' voting patterns in contrast to those of political elites.

Acknowledgments

First, we would like to thank Politools for providing the Smartvote data and offering invaluable advice on the study design. Moreover, we would like to thank all students who helped implement the VAA and conduct the experiments. Lastly, we thank the reviewers, editors, and everyone who provided feedback on the early drafts of this work, thereby helping to improve it.

Funding

We gratefully acknowledge the support of the Swiss National Science Foundation (SNSF) under grant ID CRSII5-205975, which provided the primary funding for our research. Publication of this article in open access was made possible through the institutional membership agreement between the University of Zurich and Cogitatio Press.

Conflict of Interests

The authors declare no conflict of interests.

Data Availability

All data and code necessary to reproduce the results are publicly available in this GitHub repository: https://github.com/fsvbach/recommendations-pag-paper. Please contact the authors directly with potential questions.

LLMs Disclosure

Generative AI was used to improve and document our research code during the analysis, and to revise elements of the text for grammar and clarity.

Supplementary Material

Supplementary material for this article is available online in the format provided by the authors (unedited).

References

Agathokleous, M., & Tsapatsoulis, N. (2016). Applying hidden Markov models to voting advice applications. *EPJ Data Science*, *5*(1), Article 34. https://doi.org/10.1140/epjds/s13688-016-0095-z

Bachmann, F., Sarasua, C., & Bernstein, A. (2024). Fast and adaptive questionnaires for voting advice applications. In A. Bifet, T. Krilavičius, I. Miliou, & S. Nowaczyk (Eds.), *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track* (pp. 365–380). Springer Nature. https://doi.org/10.1007/978-3-031-70381-2_23

Bachmann, F., van Der Weijden, D., Heitz, L., Sarasua, C., & Bernstein, A. (2025). Adaptive political surveys and GPT-4: Tackling the cold start problem with simulated user interactions. *PLoS One*, *20*(5), Article e0322690. https://doi.org/10.1371/journal.pone.0322690

Berdoz, F., Brunner, D., Vonlanthen, Y., & Wattenhofer, R. (2025). Recommender systems for democracy: Toward adversarial robustness in voting advice applications. In James Kwok (Eds.), *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence* (pp. 9564–9572). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2025/1063

Bruinsma, B. (2020). Evaluating visualisations in voting advice applications. *Statistics, Politics and Policy*, *11*(1), 1–21. https://doi.org/10.1515/spp-2019-0009

Buryakov, D., Kovacs, M., Serdült, U., & Kryssanov, V. (2024). Enhancing the design of voting advice applications with BERT language model. *Frontiers in Artificial Intelligence*, *7*, Article 1343214. https://doi.org/10.3389/frai.2024.1343214

Buuren, S. V. (2012). *Flexible imputation of missing data*. CRC Press.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The information visualizer, an information workspace. In S. P. Robertson, G. M. Olson, & J. S. Olson (Eds.), *CHI '91: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 181–186). Association for Computing Machinery. https://doi.org/10.1145/108844.108874

Clinton, J., Jackman, S., & Rivers, D. (2004). The statistical analysis of roll call data. *American Political Science Review*, *98*(2), 355–370. https://doi.org/10.1017/S0003055404001194

Elahi, M., Ricci, F., & Rubens, N. (2016). A survey of active learning in collaborative filtering recommender systems. *Computer Science Review*, *20*, 29–50. https://doi.org/10.1016/j.cosrev.2016.05.002

Garzia, D., & Marschall, S. (2019). Voting advice applications. In *Oxford research encyclopedia of politics*. https://doi.org/10.1093/acrefore/9780190228637.013.620

Garzia, D., Trechsel, A. H., & De Angelis, A. (2017). Voting advice applications and electoral participation: A multi-method study. *Political Communication*, *34*(3), 424–443. https://doi.org/10.1080/10584609.2016.1267053

Germann, M., & Gemenis, K. (2019). Getting out the vote with voting advice applications. *Political Communication*, *36*(1), 149–170. https://doi.org/10.1080/10584609.2018.1526237

Germann, M., & Mendez, F. (2016). Dynamic scale validation reloaded: Assessing the psychometric properties of latent measures of ideology in VAA spatial maps. *Quality & Quantity*, *50*(3), 981–1007. https://doi.org/10.1007/s11135-015-0186-0

Grung, B., & Manne, R. (1998). Missing values in principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, *42*(1/2), 125–139. https://doi.org/10.1016/s0169-7439(98)00031-8

Kamoen, N., & Liebrecht, C. (2022). I need a CAVAA: How conversational agent voting advice applications (CAVAAs) affect users' political knowledge and tool experience. *Frontiers in Artificial Intelligence*, *5*, Article 835505. https://doi.org/10.3389/frai.2022.835505

Katakis, I., Tsapatsoulis, N., Mendez, F., Triga, V., & Djouvas, C. (2014). Social voting advice applications—Definitions, challenges, datasets and evaluation. *IEEE Transactions on Cybernetics*, *44*(7), 1039–1052. https://doi.org/10.1109/TCYB.2013.2279019

Kay, M., Kola, T., Hullman, J. R., & Munson, S. A. (2016). When (ish) is my bus? User-centered visualizations of uncertainty in everyday, mobile predictive systems. In J. Kaye, A. Druin, C. Lampe, D. Morris, & J. P. Hourcade (Eds.), *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 5092–5103). Association for Computing Machinery. https://doi.org/10.1145/2858036.2858558

Ladner, A., Fivaz, J., & Pianzola, J. (2012). Voting advice applications and party choice: Evidence from SmartVote users in Switzerland. *International Journal of Electronic Governance*, *5*(3/4), 367–387.

Law, E., Yin, M., Goh, J., Chen, K., Terry, M., & Gajos, K. Z. (2016). Curiosity killed the cat, but makes crowdwork better. In J. Kaye, A. Druin, C. Lampe, D. Morris, & J. P. Hourcade (Eds.), *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 4098–4110). Association for Computing Machinery. https://doi.org/10.1145/2858036.2858144

Louwerse, T., & Rosema, M. (2014). The design effects of voting advice applications: Comparing methods of calculating matches. *Acta Politica*, *49*(3), 286–312. https://doi.org/10.1057/ap.2013.30

Lü, L., Medo, M., Yeung, C. H., Zhang, Y.-C., Zhang, Z.-K., & Zhou, T. (2012). Recommender systems. *Physics Reports*, *519*(1), 1–49. https://doi.org/10.1016/j.physrep.2012.02.006

McCoy, J. T., Kroon, S., & Auret, L. (2018). Variational autoencoders for missing data imputation with application to a simulated milling circuit. *IFAC-PapersOnLine*, *51*(21), 141–146. https://doi.org/10.1016/j.ifacol.2018.09.406

Mendler-Dünner, C., Ding, F., & Wang, Y. (2022). Anticipating performativity by predicting from predictions. *Advances in Neural Information Processing Systems*, *35*, 31171–31185. https://dl.acm.org/doi/10.5555/3600270.3602530

Munzert, S., & Ramirez-Ruiz, S. (2021). Meta-analysis of the effects of voting advice applications. *Political Communication*, *38*(6), 691–706. https://doi.org/10.1080/10584609.2020.1843572

Perdomo, J. C., Zrnic, T., Mendler-Dünner, C., & Hardt, M. (2020). Performative prediction. In H. Daumé & A. Singh (Eds.), *ICML '20: Proceedings of the 37th International Conference on Machine Learning* (Vol. 119, pp. 7599–7609). PMLR. https://proceedings.mlr.press/v119/perdomo20a.html

Pianzola, J., Trechsel, A. H., Vassil, K., Schwerdt, G., & Alvarez, R. M. (2019). The impact of personalized information on vote intention: Evidence from a randomized field experiment. *The Journal of Politics*, *81*(3), 833–847. https://doi.org/10.1086/702946

Politools. (2023). *Daten zu den Nationalrats- und Ständeratswahlen 2023 der Online-Wahlhilfe Smartvote* [Data set]. Smartvote. https://www.smartvote.ch

Poole, K. T., & Rosenthal, H. (1985). A spatial model for legislative roll call analysis. *American Journal of Political Science*, *29*(2), 357–384. https://doi.org/10.2307/2111172

Potthoff, R. (2018). Estimating ideal points from roll-call data: Explore principal components analysis, especially for more than one dimension? *Social Sciences*, *7*(2), Article 12. https://doi.org/10.3390/socsci7010012

Romero Moreno, G., Padilla, J., & Chueca, E. (2022). Learning VAA: A new method for matching users to parties in voting advice applications. *Journal of Elections, Public Opinion and Parties*, *32*(2), 339–357. https://doi.org/10.1080/17457289.2020.1760282

Rosema, M., & Louwerse, T. (2016). Response scales in voting advice applications: Do different designs produce different outcomes? *Policy & Internet*, *8*(4), 431–456. https://doi.org/10.1002/poi3.139

Rubens, N., Elahi, M., Sugiyama, M., & Kaplan, D. (2015). Active learning in recommender systems. In F. Ricci, L. Rokach, & B. Shapira (Eds.), *Recommender systems handbook* (pp. 809–846). Springer. https://doi.org/10.1007/978-1-4899-7637-6_24

Schwarz, D., Schädel, L., & Ladner, A. (2010). Pre-election positions and voting behaviour in parliament: Consistency among Swiss MPs. *Swiss Political Science Review*, *16*(3), 533–564. https://doi.org/10.1002/j.1662-6370.2010.tb00440.x

Swiss Federal Statistical Office. (2023). *Federal elections*. https://www.wahlen.admin.ch/en/zh

Tromborg, M. W., & Albertsen, A. (2023). Candidates, voters, and voting advice applications. *European Political Science Review*, *15*(4), 582–599. https://doi.org/10.1017/S1755773923000103

Urbistat. (2025). *Age classes by gender region Zurich. Maps, analysis and statistics about the resident population*. https://ugeo.urbistat.com/AdminStat/en/ch/demografia/eta/zurich/1/2

About the Authors

Fynn Bachmann is a PhD candidate in computer science at the University of Zurich. Previously, he obtained a master's degree in machine learning from the University of Tübingen and a bachelor's degree in physics from the University of Heidelberg. More information can be found on his personal website: https://fynnbachmann.com

Daan van der Weijden is a PhD candidate in computer science at the University of Zurich. He obtained a master's degree in artificial intelligence and a double bachelor's degree in artificial intelligence and linguistics, both from the University of Utrecht. More information can be found on his personal website: https://daanvdweijden.com

Cristina Sarasua is a senior researcher in computer science at the University of Zurich. Her research lies at the intersection of human-centered computing, data-driven technology, and civic participation. Prior to joining the University of Zurich, Cristina worked at the University of Koblenz-Landau, where she received her PhD in computer science.

Abraham Bernstein is a full professor of informatics at the University of Zurich. His current research focuses on various aspects of artificial intelligence and human-computer interaction with practical applications in media, medicine, and political science. His work is based on both social science and technical foundations. Mr. Bernstein is also a founding director of the University of Zurich's initiative for the digital transformation (called DSI) involving more than 1,400 researchers from all disciplines. For more information, see his research group's webpage at https://ddis.ch