Finding the Signal in the Spam: Jointly Learning Rewards and Worker Reliability from Pairwise Comparisons

Kaustubh Shivshankar Shejole[1,*] **Tanish Agarwal**[1,*] **Arpit Agarwal**[1] **Avishek Ghosh**[1]

[*]Equal contribution
[1]IIT Bombay, Mumbai, India – 400076

Abstract

The problem of learning from pairwise comparisons has been widely studied across many domains such as recommendation systems, social choice, and more recently, fine-tuning large language models. In this problem, the goal is to learn item rewards based on pairwise comparisons between them. In many scenarios, these comparisons are elicited from crowdworkers using platforms such as Amazon Mechanical Turk, Scale AI, etc. However, crowdworkers are often unreliable due to limited domain knowledge or revenue-maximizing (spamming) behavior. In this work, our goal is to understand whether worker reliability (competency) can be learned jointly with item rewards. To this end, we adopt the Boltzmann-rational model for pairwise comparisons, which extends the Bradley–Terry–Luce model by incorporating worker competencies. We derive an EM-based algorithm for learning under this model by introducing Polya-Gamma latent variables to transform the logistic likelihood into a conditionally Gaussian form, enabling tractable optimization and leading to a simplified Q function in the E-step of the algorithm. This technique allows us to reduce our formulation to a matrix sensing problem, using which we establish theoretical convergence guarantees for our algorithm. We conduct extensive experiments on real-world and synthetic datasets. These experiments demonstrate the advantages of using our algorithm over several baselines and confirm its strong robustness to both spammers and adversarial workers, highlighting its practical effectiveness in realistic crowdsourcing and reward learning settings.[a]

[a]The code and data is publicly available at `https://github.com/KaustubhShejole/BoRa_EM`.

1 INTRODUCTION

Reward learning from pairwise comparisons is a widely studied problem in machine learning, with applications in recommendation systems, information retrieval, social choice, and, more recently, fine-tuning of large language models. In this problem, given pairwise comparisons between K items, the goal is to learn a reward for each item that is "consistent" with the observed comparisons. For example, in recommendation systems, one observes the choices of users over recommended items and the goal is to learn a score for each item [Kalloori et al., 2018]; in social choice, one observes preferences of individuals over various policy alternatives and the goal is to aggregate the preferences into a single ranked list [List, 2022]; and in fine-tuning large language models, one observes pairwise comparisons between different model responses and the goal is to learn a reward for each response [Bakker et al., 2022, Stiennon et al., 2020].

Given the large-scale requirement of preference data in these applications, comparisons between items are often elicited using human crowdworkers [Velayati et al., 2025, Chen et al., 2013]. Due to this, many crowdsourcing marketplaces such as Amazon Mechanical Turk, Scale AI, Figure Eight, and Yandex Toloka have gained prominence in recent years. However, it has been observed that crowdworkers often exhibit large variability in the quality of labels [Gadiraju et al., 2015, Ding et al., 2022]. There may be several factors leading to this variability such as insufficient knowledge of the domain, lack of time/effort devoted to the tasks, and revenue maximizing/spamming behavior. The default approach adopted by many platforms is to identify a gold-standard dataset and evaluate worker competency on it [Chen et al., 2013]. However, in many scenarios such datasets are not available, and requiring workers to answer gold-standard questions can be inefficient and costly.

In this work, we study the problem of jointly estimating item rewards and worker competencies from pairwise comparison data, without requiring a gold-standard dataset. We

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:6204–6231.

adopt the Boltzmann-rational model, in which each item $j \in [K]$ has a latent reward r_j and each worker $s \in [M]$ has a competency parameter $\beta_s \in [-1, 1]$. The probability that worker s prefers item w over item l is given by

$$P[w \succ l \mid s] = \sigma(\beta_s(r_w - r_l)),$$

where σ denotes the logistic function. This model captures a spectrum of worker behaviors: $\beta_s = 1$ corresponds to an ideal annotator, $\beta_s = 0$ to a random spammer, and $\beta_s = -1$ to an adversarial worker. Given a dataset $\mathcal{D} = \{(w_i, l_i, s_i)\}_{i=1}^N$ of N pairwise comparisons, the goal is to jointly estimate the reward vector $\mathbf{r} \in \mathbb{R}^K$ and the competency vector $\boldsymbol{\beta} \in [-1, 1]^M$.

We derive an expectation-maximization (EM) algorithm for inference under this model. To derive the EM algorithm, we use a Pólya-Gamma augmentation [Polson et al., 2013], which introduces auxiliary variables to transform the logistic likelihood into a conditionally Gaussian form. This leads to a quadratic objective for the M-step which enables efficient alternating maximization over $\boldsymbol{\beta}$ and $\mathbf{r}$ in closed form (Section 4). While this alternating minimization can be solved efficiently, it is not guaranteed to converge to a global optimum.

To address this, we show that the M-step optimization can be cast as a rank-1 matrix sensing problem. Specifically, defining $X = \boldsymbol{\beta}\mathbf{r}^\top \in \mathbb{R}^{M \times K}$ and a linear operator $\mathcal{A} : \mathbb{R}^{M \times K} \to \mathbb{R}^N$, the M-step reduces to minimizing $\|\mathcal{A}(\boldsymbol{\beta}\mathbf{r}^\top) - \mathbf{u}\|_2^2$. We establish that $\mathcal{A}$ satisfies the Restricted Isometry Property (RIP) given a sufficient number of samples. Leveraging results from non-convex low-rank matrix recovery [Park et al., 2017], we prove that under the RIP condition, at each iteration of the EM algorithm, every local minimum of the M-step objective is near-global minimum. Using standard results for EM, we are able to show that the likelihood increases monotonically and converges to a stationary point of the likelihood function.

We conduct extensive experiments on both real-world and synthetic datasets. The experimental results demonstrate that our algorithm outperforms baseline methods, and more importantly, it remains robust in scenarios with large numbers of spammers of various types and achieves significant performance gains over other existing approaches in this case. Thus, we provide a unified framework capable of handling diverse spammer behaviors—from random clickers to malicious adversaries—offering a practical and theoretically grounded alternative for real-world crowdsourcing platforms where worker quality is heterogeneous and unknown *a priori*.

2 RELATED WORK

Learning rewards from pairwise comparisons has been widely used across many domains such as economics, operations research and machine learning. The classical Bradley-Terry-Luce (BTL) model [Bradley and Terry, 1952] assumes that each item is associated with an underlying reward, and pairwise comparisons between items are drawn by taking into account the reward difference between them. Specifically, the probability that item w is preferred over item l is given as $P(w \succ l) = \sigma(r_w - r_l)$, where $\sigma(\cdot)$ denotes the sigmoid function, and $\mathbf{r} = \{r_j\}_{j=1}^K$ represents the latent reward or score associated with each of the K items. The BTL model, however, does not capture the heterogeneity in the rewards observed by different crowdworkers.

Negahban et al. [2012] proposed RankCentrality (`RC`), which interprets pairwise comparisons as a directed graph, where nodes represent items and directed edges represent comparison outcomes. This formulation enables the use of random-walk–based methods for ranking items.

The closest to our work is Crowd-BT model [Chen et al., 2013] which extends the `BTL` model by incorporating worker reliability. For a worker s, the probability of preferring item i over j is modeled as $\Pr(i \succ j \mid s) = \beta_s\,\sigma(r_i - r_j) + (1 - \beta_s)\,\sigma(r_j - r_i)$, where r_i, r_j are item rewards and β_s captures the worker's competency. This model is effectively a mixture of two `BTL` models – one with reward $\mathbf{r}$ and the other with negative reward i.e., $-\mathbf{r}$. The mixture weight represents the competency of the crowdworker, with $\beta_s = 1/2$ representing a worker that labels uniformly at random. Crowd-BT is an alternating minimization approach to jointly learn the rewards and competencies. However, Chen et al. [2013] do not provide any theoretical guarantees, and they require a gold standard dataset to initialize worker competencies, which is not always available. On the other hand, our work adopts the Boltzmann rational choice model, which provides a more natural and theoretically grounded formulation for preference-based reward learning and alignment, and allows us to prove theoretical convergence results.

Another related work is Bugakova et al. [2019], which introduced the Factor-BT model to explicitly capture systematic biases present in crowdsourced pairwise comparisons. Unlike prior models, which treat task-related biases as noise or ignore them altogether, `FactorBT` explicitly models these biases by incorporating known but irrelevant task features (e.g., item position on screen, background design) that may influence worker decisions. For a comparison between items d_i and d_j by worker s, described by a feature vector $\mathbf{x}_{sij} \in \mathbb{R}^M$, where M is the feature dimension. The probability that d_i is preferred over d_j is given as follows $\Pr(d_i \succ_s d_j) = \sigma(\gamma_s) \cdot \sigma(r_i - r_j) + (1 - \sigma(\gamma_s)) \cdot \sigma(\langle \boldsymbol{b}_s, \mathbf{x}_{sij} \rangle)$, where r_i, r_j are the latent scores of items d_i and d_j, γ_s controls the susceptibility of worker s to bias and $\sigma(\gamma_s)$ can be considered as the the skill of the worker same as that of β_s in our formulation, $\boldsymbol{b}_s$ encodes the worker-specific sensitivity to task features, $\sigma(\cdot)$ is the sigmoid function. The model estimates parameters $\{r_i\}$, $\{\gamma_s\}$, and $\{\boldsymbol{b}_s\}$ by maximizing

the regularized log-likelihood of the observed pairwise comparisons. Unlike `FactorBT` [Bugakova et al., 2019], we do not explicitly model the influence of task-irrelevant factors such as item position, and we instead focus on learning worker competency jointly with rewards.

Jin et al. [2020] proposed two models that explicitly account for annotator competence: the Heterogeneous *Bradley-Terry--Luce* (HBTL) model and the Heterogeneous *Thurstone Case V* (HTCV) model. The HBTL model is based on the Boltzmann-Rational formulation given in Equation (2). In contrast, the HTCV model assumes that the probability of item w being preferred over item l by annotator s is given by

$$P(w \succ l; s) = \Phi\left(\frac{\beta_s(r_w - r_l)}{\sqrt{2}}\right), \tag{1}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of the standard normal distribution, and β_s captures the competence of annotator s. They perform alternating gradient descent for both these methods and provide theoretical convergence guarantees.

Bias-Aware Ranking from Pairwise comparisons (BARP) [Ferrara et al., 2024] extends the classic BTL model to account for evaluator biases in pairwise comparisons. Each evaluator is assigned a bias parameter that distorts the latent quality scores of items based on the group membership of items. By explicitly modeling the log-likelihood over items' latent scores and evaluators' biases, `BARP` uses an alternating optimization approach to jointly estimate both. In `BARP`, the probability that evaluator s prefers item i over j is modeled as $P(i \succ_s j) = \sigma((r_i - r_j) + \theta_s(\delta_{ig} - \delta_{jg}))$, where r_i, r_j are the latent item scores and $\theta_s\delta_{ig}$ represents the bias of evaluator s for the group g of item i. Unlike the standard BTL model, this formulation accounts for biased perception of item scores rather than true latent scores.

Organization Section 3 introduces the problem of jointly learning rewards and worker competence from pairwise comparisons. Section 4 presents the derivation of our proposed method (**BoRaEM**) for the **Bo**ltzmann **Ra**tional Model using Expectation–Maximization (**EM**), while Section 5 establishes theoretical guarantees for its convergence. Section 6 reports experimental results and analysis. Finally, Section 7 concludes with a discussion of future research directions.

3 PROBLEM FORMULATION

We consider a setting where there is a set of K items, and one observes outcomes of pairwise comparisons between these items, elicited from a set of M crowdworkers. Specifically, we are given a dataset $\mathcal{D} = \{(w_i, l_i, s_i)\}_{i=1}^N$ where there are N pairwise comparisons; the i-th comparison is performed by worker $s_i \in [M]$ where $w_i \in [K]$ is the winning item and $l_i \in [K]$ is the losing item such that l_i is not equal to w_i.

We assume that the outcomes of these comparisons are generated according to the Boltzmann-rational model which is an extension of the `BTL` model and has received a lot of attention recently [Daniels-Koch and Freedman, 2022, Jeon et al., 2020, Ziebart et al., 2010]. Similar to the `BTL` model, each item $j \in [K]$ is associated with a reward r_j, and the probability of winning a comparison is governed by pairwise reward differences. This model also considers the heterogeneity of human decision-making. Specifically, humans are considered rational decision-makers, where the quality of their decisions is modulated by a "rationality" parameter $\beta \in [-1, 1]$ [Daniels-Koch and Freedman, 2022, Jeon et al., 2020, Ziebart et al., 2010]. Formally, the probability that item w beats item l given worker s is given by

$$\begin{aligned} P[w \succ l; s] &= \frac{\exp(\beta_s r_w)}{\exp(\beta_s r_w) + \exp(\beta_s r_l)} \\ &= \frac{1}{1 + \exp(-\beta_s(r_w - r_l))} \\ &= \sigma(\beta_s(r_w - r_l)). \end{aligned} \tag{2}$$

The case $\beta_s = 1$ corresponds to a perfect annotator who makes decisions according to the underlying `BTL` model, $\beta_s = -1$ for an "adversarial" worker, and $\beta_s = 0$ for a "spammer" who labels uniformly at random.

Given a dataset of pairwise comparisons $\mathcal{D} = \{(w_i, l_i, s_i)\}_{i=1}^N$ that is drawn according to the Boltzmann-rational model with reward vector $\mathbf{r} = \{r_j\}_{j=1}^K$ and worker competency vector $\boldsymbol{\beta} = \{\beta_s\}_{s=1}^M$, the goal is to jointly estimate $\mathbf{r}$ and $\boldsymbol{\beta}$. Since the reward model is invariant to both additive shifts and scaling, as the likelihood depends only on the product $\beta_s(r_w - r_l)$, the transformations $r \to r + c$ and $r \to cr$, $\beta_s \to \beta_s/c$ leave the likelihood unchanged, resulting in non-identifiability. To address this, we enforce identifiability by centering the rewards to remove translation invariance and normalizing them to unit root-mean-square after each EM iteration, with a compensatory rescaling of worker competencies. This fixes the latent scale and selects a unique representative solution. Additionally, mild ℓ_2 regularization on rewards and a Gaussian prior on competencies improve numerical stability and prevent degenerate scaling. More details about implementation are provided in Appendix E.

4 BORAEM ALGORITHM

In order to solve this problem, one can formulate a log-likelihood objective based on the underlying model. Specifically, following Equation (2) for the probability that a worker s_i favors w_i over l_i:

$$p(w_i \succ l_i \mid s_i) = \sigma(\beta_{s_i}(r_{w_i} - r_{l_i})) \tag{3}$$

Let $\boldsymbol{\theta} = \mathbf{r}, \boldsymbol{\beta}$. The associated complete-data likelihood is

$$\mathcal{L}_{\text{complete}}(\boldsymbol{\theta}; \mathcal{D}) = \prod_{(w_i, l_i, s_i) \in \mathcal{D}} \sigma\left(\beta_{s_i}(r_{w_i} - r_{l_i})\right) \quad (4)$$

where $\mathcal{D}$ refers to the comparisons data.

However, direct maximum likelihood estimation (MLE) of the parameter vectors $\mathbf{r}$ and $\boldsymbol{\beta}$ is challenging as the objective is non-concave. This complicates both optimization and theoretical analysis.

To circumvent this issue, we adopt the **Pólya-Gamma (PG) augmentation framework** [Polson et al., 2013][1]. For a latent variable $\omega \sim \text{PG}(b, 0)$ where $b > 0$, and any $\psi \in \mathbb{R}$, the logistic likelihood admits the integral representation:

$$\frac{(e^{\psi})^a}{(1+e^{\psi})^b} = \frac{e^{\kappa\psi}}{2^b} \int_0^{\infty} \exp\left(-\frac{\omega\psi^2}{2}\right) p(\omega \mid b, 0)\, d\omega, \quad (5)$$

with $\kappa = a - b/2$, where $p(\cdot)$ is the density of ω.

We apply this identity to our model by setting $a = 1, b = 1$, and $\psi = \eta_i$ where $\eta_i := \beta_{s_i}(r_{w_i} - r_{l_i})$.

Thus, the probability of the i^{th} pairwise comparison can be written as:

$$\begin{aligned} p(w_i, l_i, s_i | \boldsymbol{\theta}) &= \sigma(\eta_i) \\ &= \frac{1}{2} e^{\eta_i/2} \int_0^{\infty} \exp\left(-\frac{\omega_i \eta_i^2}{2}\right) p(\omega_i) d\omega_i, \end{aligned} \quad (6)$$

where $\omega_i \sim \text{PG}(1, 0)$ is a latent Pólya-Gamma variable introduced for each i^{th} comparison.

So, the joint density $p(w_i, l_i, s_i, \omega_i | \boldsymbol{\theta})$ is given by

$$p(w_i, l_i, s_i, \omega_i | \boldsymbol{\theta}) \propto \exp\left(-\frac{\omega_i \eta_i^2}{2} + \frac{\eta_i}{2}\right) p(\omega_i) \quad (7)$$

Following Equation (7), the complete-data likelihood, takes an exponential form as follows:

$$\mathcal{L}_{\text{complete}}(\mathcal{D} \mid \boldsymbol{\theta}) \propto \frac{1}{2}\, \mathbb{E}_{\omega_i \sim \text{PG}(1,0)} \Big[\exp\big(-\frac{\omega_i \eta_i^2}{2} + \frac{\eta_i}{2} \big) \Big] \quad (8)$$

Further details are in Appendix A.2.

Expectation Maximization. The basis of our algorithm is as follows: using Equation (8), and given all ω_i, the log-likelihood objective over $\boldsymbol{\theta} = (\{r_j\}_{j=1}^K, \{\beta_l\}_{l=1}^M)$, simplifies to the following form (further details in Appendix A.2).

$$\begin{aligned} \ell(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}|} \sum_{(w_i, l_i, s_i) \in \mathcal{D}} \Big[&\frac{1}{2} \beta_{s_i} (r_{w_i} - r_{l_i}) \\ &- \frac{1}{2} \omega_i \beta_{s_i}^2 (r_{w_i} - r_{l_i})^2 \Big] \end{aligned} \quad (9)$$

Hence, this gives an EM algorithm that iteratively estimates the latent variables ω_i followed by log-likelihood maximization steps. Algorithm 1 describes this EM Algorithm. The Expectation step (E-step) and Maximization step (M-step) are as follows (details in Appendix A.3):

4.1 E-STEP

At iteration t, given the current estimates $\boldsymbol{\theta}^{(t)}$ we have

$$\eta_i^{(t)} = \beta_{s_i}^{(t)} (r_{w_i}^{(t)} - r_{l_i}^{(t)}), \quad \forall i \in [N] \quad (10)$$

Given $\eta_i^{(t)}$, the Pólya–Gamma random variable ω_i follows the distribution $\text{PG}(1,\ \eta_i^{(t)})$. Hence, we can evaluate the expectation of ω_i given the current parameters. Let $\kappa_i^{(t)} = \mathbf{E}[\omega_i \mid \eta_i^{(t)}]$. We show that (Appendix A.3.1)

$$\kappa_i^{(t)} = \begin{cases} \frac{1}{2\,\eta_i^{(t)}} \tanh\left(\frac{\eta_i^{(t)}}{2}\right), & \eta_i^{(t)} \neq 0 \\ \frac{1}{4}, & \eta_i^{(t)} = 0 \end{cases} \quad (11)$$

This gives the formulation for the Q function:

$$\begin{aligned} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) = \sum_{(w_i, l_i, s_i) \in \mathcal{D}} \Big[&\frac{1}{2} \beta_{s_i} (r_{w_i} - r_{l_i}) \\ &- \frac{1}{2} \kappa_i^{(t)} \beta_{s_i}^2 (r_{w_i} - r_{l_i})^2 \Big] \end{aligned} \quad (12)$$

4.2 M-STEP

In order to maximize the $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$ function in terms of $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{r})$, we use the Alternating Maximization (AM) algorithm. We first initialize $\boldsymbol{\theta}$ to $\boldsymbol{\theta}^{(t)}$. We update $\boldsymbol{\beta}$ corresponding to Step 9 of Algorithm 1 as follows (Appendix A.3.2)

$$\beta_s = \frac{\frac{1}{2} \sum_{i \in I_s} d_i}{\sum_{i \in I_s} \kappa_i^{(t)} d_i^2} = \frac{\sum_{i \in I_s} d_i}{2 \sum_{i \in I_s} \kappa_i^{(t)} d_i^2} \quad (13)$$

where $I_s = \{ i \in [N] : s_i = s \}$ as the index set of comparisons for worker s.

The reward vector $\mathbf{r}$ can be obtained uniquely by solving the linear system $\mathbf{Hr} = \mathbf{b}$, and imposing the constraint

[1] We encourage readers to also refer to a useful blog on Pólya-Gamma (PG) augmentation at `https://gregorygundersen.com/blog/2019/09/20/polya-gamma/`.

Algorithm 1 BoRaEM Algorithm

Input: Dataset of pairwise comparisons $\mathcal{D}$
Output: Parameter estimates $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{r})$

1: Initialize $\boldsymbol{\theta}^{(0)}$
2: $t \leftarrow 0$
3: **repeat**
4: **E-step:**
5: $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) \leftarrow \mathbb{E}_{\omega|\mathcal{D},\boldsymbol{\theta}^{(t)}}[\ell(\boldsymbol{\theta}; \mathcal{D}, \omega)]$ ▷ See (12)
6: **M-step (using AM):**
7: Initialize $(\boldsymbol{\beta}^{(0)}, \boldsymbol{r}^{(0)}) \leftarrow \theta^{(t)}$, $k \leftarrow 0$
8: **repeat**
9: $\boldsymbol{\beta}^{(k+1)} \leftarrow \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}, \boldsymbol{r}^{(k)} \mid \boldsymbol{\theta}^{(t)})$
10: $\boldsymbol{r}^{(k+1)} \leftarrow \arg\max_{\boldsymbol{r}} Q(\boldsymbol{\beta}^{(k+1)}, \boldsymbol{r} \mid \boldsymbol{\theta}^{(t)})$
11: $\boldsymbol{r}^{(k+1)} \leftarrow \boldsymbol{r}^{(k+1)} - \frac{\mathbf{1}^T \boldsymbol{r}^{(k+1)}}{n}$
12: $k \leftarrow k + 1$
13: **until** convergence
14: $\boldsymbol{\theta}^{(t+1)} \leftarrow (\boldsymbol{\beta}^{(k-1)}, \boldsymbol{r}^{(k-1)})$
15: $t \leftarrow t + 1$
16: **until** convergence
17: **return** $\boldsymbol{\theta}^{(t+1)}$

$\sum_{j=1}^{K} r_j = 0$, where the matrix $\mathbf{H}$ is defined as:

$$H_{jj} = \sum_{i \in W_j} \beta_{s_i}^2 \kappa_i^{(t)} + \sum_{i \in L_j} \beta_{s_i}^2 \kappa_i^{(t)},$$
$$H_{jl} = \sum_{i \in \mathcal{D}_{jl}} -\beta_{s_i}^2 \kappa_i^{(t)}, \quad j \neq l,$$

and, the vector $\mathbf{b}$ is defined as:

$$b_j = \sum_{i \in W_j} \frac{1}{2} \beta_{s_i} + \sum_{i \in L_j} \left(-\frac{1}{2} \beta_{s_i} \right),$$

where W_j and L_j denote the sets of comparisons in which item j won and lost, respectively, and $\mathcal{D}_{jl}$ is the set of comparisons between items j and l. This corresponds to Steps 10 and 11 of Algorithm 1. The linear system has a unique solution (Appendix A.3.2) and can be solved efficiently using conjugate gradient descent. The implementation details are provided in Appendix E.

5 CONVERGENCE ANALYSIS

In this section we show that the M-step objective can be written as a matrix sensing problem with a rank-1 parameter matrix $X = \boldsymbol{\beta} \boldsymbol{r}^\top$. This allows us to leverage standard results from low-rank matrix recovery to analyze the geometry of the Q-function and the behavior of EM.

Formally, Equation (12) can be stated in the following form.

$$\min_{\boldsymbol{\beta}, \boldsymbol{r}} \|\mathcal{A}(\boldsymbol{\beta} \boldsymbol{r}^T) - \mathbf{u}\|_2^2,$$

where $\mathbf{u} \in \mathbb{R}^N$ and the linear operator $\mathcal{A}: \mathbb{R}^{M \times K} \to \mathbb{R}^N$ as $\mathcal{A}(X) = (\langle A_1, X \rangle, \ldots, \langle A_N, X \rangle)^\top$ with matrices $A_i = 2c_i \alpha \, \boldsymbol{e_{s_i}} (\boldsymbol{e_{w_i}} - \boldsymbol{e_{l_i}})^\top$ where $\alpha = \sqrt{MK/2N}$, $c_i = \kappa_i^{(t)}$ for iteration t, e_x denotes a unit vector having value as 1 at index x. More details are in Appendix B.

We assume that the worker index and the two item indices (denoted by random variables B, W, and L, respectively) are independent and uniformly distributed, $B \sim \text{Unif}\{1, 2, \ldots, M\}$ and $W, L \sim \text{Unif}\{1, 2, \ldots, K\}$. The comparison outcome is then generated according to 2. Further, we assume that $\beta \in [-1, 1]$ and $r \in [-R, R]$ for some constant $R > 0$.

Under mild assumptions detailed in Appendix B, the linear operator $\mathcal{A}$ satisfies the Restricted Isometry Property, according to the following definition [Candes and Plan, 2010].

Definition (Restricted Isometry Property (RIP))**.** *A linear operator $\mathcal{A} : \mathbb{R}^{M \times K} \to \mathbb{R}^N$ with $(\mathcal{A}(X))_i = \langle A_i, X \rangle$ satisfies the restricted isometry property on rank-r matrices, with parameter δ_r, if the following set of inequalities hold for all rank-r matrices X:*

$$(1 - \delta_r)\|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1 + \delta_r)\|X\|_F^2.$$

This gives us the following lemma.

Lemma 1. *Define matrix operators $A_i \in \mathbb{R}^{M \times K}$ as $A_i = 2c_i \alpha \, \boldsymbol{e_{s_i}} (\boldsymbol{e_{w_i}} - \boldsymbol{e_{l_i}})^\top$, where $\alpha = \sqrt{MK/2N}$. Let $X \in \mathbb{R}^{M \times K}$ be any matrix of arbitrary rank r with $X\mathbf{1} = \mathbf{0}$, where $\mathbf{1}$ and $\mathbf{0}$ denote the all-ones and all-zeros vectors, respectively. Define the linear operator $\mathcal{A}: \mathbb{R}^{M \times K} \to \mathbb{R}^N$ as $\mathcal{A}(X) = (\langle A_1, X \rangle, \ldots, \langle A_N, X \rangle)^\top$. Then, for any $\eta > 0$, there exists $\mu > 0$ and $\delta > 0$ such that with probability at least $1 - \eta$:*

$$(1 - \mu)\|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1 + \mu)\|X\|_F^2,$$

provided that

$$N \geq C \, \frac{MK}{\delta^2} \ln\left(\frac{2}{\eta}\right),$$

where $C > 0$ is some constant.

Then, invoking Remark 3 from Park et al. [2017], we can state the following theorem.

Theorem 1. *Let $\{\boldsymbol{\beta}^{(t)}, \boldsymbol{r}^{(t)}\}_{t \geq 0}$ be the sequence of iterates generated by the EM algorithm. Suppose the M-step at each iteration t solves the optimization problem:*

$$(\boldsymbol{\beta}^{(t+1)}, \boldsymbol{r}^{(t+1)}) = \arg\min_{\boldsymbol{\beta}, \boldsymbol{r}} \|\mathcal{A}(\boldsymbol{\beta} \boldsymbol{r}^\top) - \mathbf{u}^{(t)}\|_2^2$$

having global minimum $(\boldsymbol{\theta}^*)^{(t+1)}$, where $u_i^{(t)} = \frac{\alpha}{c_i}$. If $R = 0.1$, the operator $\mathcal{A}$ satisfies Restricted Isometry Property with $\mu_2 = \mu_4 = 0.005$, and every local minimum $\boldsymbol{\theta}^{(t+1)} = \left(\boldsymbol{\beta}^{(t+1)}; \boldsymbol{r}^{(t+1)}\right)$ satisfies:

$$\textit{DIST}(\boldsymbol{\theta}^{(t+1)}, (\boldsymbol{\theta}^*)^{(t+1)}) \leq \frac{1250}{3\sigma_1(X_t^*)} \cdot \|\mathcal{A}(X_t^* - X_{1,t}^*)\|_2$$

where $X_t^* \mathbf{1} = \mathbf{0}$ and $X_{1,t}^*$ is the best rank-1 approximation of the true high-rank matrix X_t^*.

Therefore, each EM update computes a solution whose distance to the global optimum of the corresponding M-step objective is bounded by the intrinsic rank-one approximation error of the corresponding M-step objective. Thus, the estimation error at each iteration is controlled by $\|\mathcal{A}(X_t^* - X_{1,t}^*)\|_2$. At iteration t, if X_t^* is exactly rank-1, the algorithm recovers the global minimum of the function $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^t)$.

Following Proposition 2.7.1 from Bertsekas [1997], we can say the alternating minimization (AM) procedure used to solve the M-step converges to a stationary point, since our $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^t)$ function is differentiable and marginally convex. In our case, in each iteration of AM, we solve for $\boldsymbol{\beta}$ and $\mathbf{r}$ exactly. Therefore, Theorem 4.3 from Park et al. [2017] implies that alternating minimization escapes saddle points and converges to a local minimum with high probability, which, by Theorem 1, is close to the global minimum.

Finally, since the M-step at each iteration is solved using AM, which is a monotonic procedure [Jain and Kar, 2017], each EM iteration monotonically increases the likelihood. Standard convergence results for EM therefore imply that the sequence of iterates converges to a stationary point of the likelihood.

6 EXPERIMENTAL ANALYSIS

6.1 EVALUATION METRICS

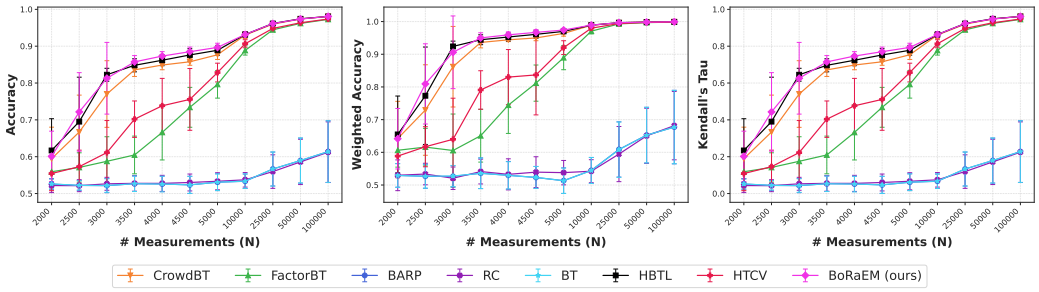
To quantitatively evaluate the performance of the proposed methods, we employ Kendall's Tau (τ), Accuracy (acc), and Weighted Accuracy ($wacc$). These metrics focus on the relative ordering of items rather than absolute value errors. Kendall's Tau is a standardized rank correlation that is robust to outliers, but it treats all pairwise swaps equally. Accuracy measures the probability of correctly ordering item pairs and is easy to interpret, though it ignores the magnitude of score differences. Weighted Accuracy penalizes larger ranking errors more strongly, but its value depends on the distribution of the scores. The definitions, including their respective advantages and limitations, are summarized in Table 3 in Appendix C. we evaluate all the representative crowdsourcing methods discussed in Section 2. More details are given in D.

6.2 SYNTHETIC DATASETS

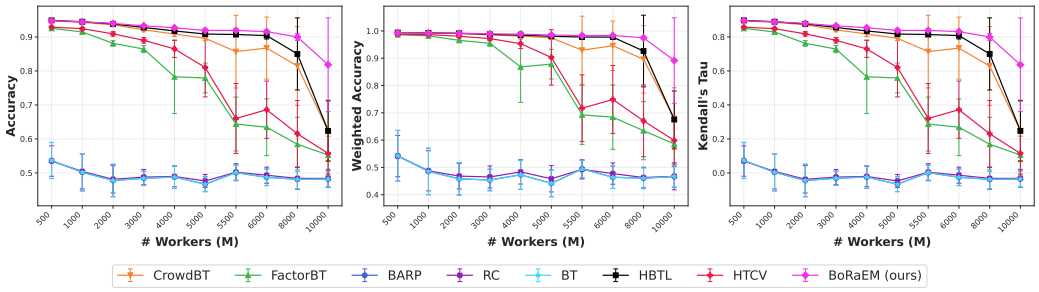
We will sample data (having N measurements) from the distribution given in Eq. (3), with K (number of items) and M (number of workers) to study their effect and we sample β uniformly from $[-1, 1]$. We use 10 random seeds for data sampling. Let k denote 1000 for specifying number of measurements, workers, or items. To study the effect of each variable, we vary N, K, M individually keeping the other two variables constant as follows:

1. **Varying N, Keeping K and M constant:** We conduct an experiment with fixed values of the number of items $K = 100$ and the number of workers $M = 500$, while varying the number of measurements $N \in \{2k, 2.5k, 3k, 3.5k, 4k, 4.5k, 5k, 10k, 25k, 50k, 100k\}$ to examine the effect of measurement volume on various techniques. As seen in Figure 1a, irrespective of number of measurements the performance of methods not modeling the annotator competence remains low, whereas the methods that model annotator competence while predicting rewards i.e., BoRaEM, HBTL and CrowdBT remains high. BoRaEM seems to be robust than CrowdBT in sparse measurements setting (i.e, for measurements less than 4000).
2. **Varying M, Keeping N and K constant:** We vary the number of workers $M \in \{500, 1k, 2k, 3k, 4k, 5k, 5.5k, 6k, 8k\}$ while fixing the number of items at $K = 200$ and the number of measurements at $N = 25k$, to analyze the impact of worker scale. As seen in Figure 1b, all methods suffers as the number of workers are increased (or the comparisons per worker get decreased). The performance decay of CrowdBT is seen to be much higher than BoRaEM and BoRaEM is quite stable.
3. **Varying K, Keeping N and M constant:** We vary the number of items $K \in \{50, 100, 250, 500, 1000, 1250, 1500, 1750, 2250, 2500\}$ while fixing the number of workers at $M = 20k$ and the number of measurements at $N = 1k$, in order to study the effect of item scale. As seen in Figure 1c, all methods suffers as the number of items are increased. The performance of BoRaEM, HBTL and CrowdBT decays after $K = 1000$, HBTL and BoRaEM seem to degrade slower than CrowdBT under larger N regimes.

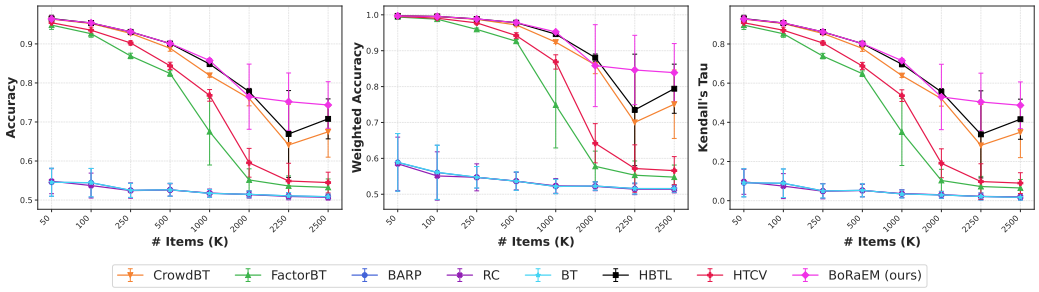
Methods such as `BT`, `BARP`, and `RC`, which do not explicitly model annotator competence or susceptibility to bias, exhibit consistently poor performance in the presence of adversarial workers. Appendix F presents results for the regime $\beta \in [0, 1]$, which contains no adversarial annotators. In this setting, these methods perform comparatively better due to absence adversaries. In contrast, `BoRaEM`, `HBTL`, and `CrowdBT`, which explicitly model annotator competence, achieve superior performance in denser regimes where sufficient observations are available to reliably estimate the additional parameters. However, their performance exhibits a modest decline in sparse regimes, where limited data constrains accurate parameter estimation and increases susceptibility to estimation noise. Hardware details are in Appendix J.



(a) Varying N. $K = 100$, $M = 500$.



(b) Varying M. $K = 200$, $N = 25000$.



(c) Varying K. $M = 20000$, $N = 1000$.

Figure 1: Varying N, K, M individually when $\beta \in [-1, 1]$.

6.3 REAL DATASETS

For evaluating crowdsourcing methods on real-world data, we use two datasets: the FaceAge dataset (IMDB-WIKI-SbS)[2] [Pavlichenko and Ustalov, 2021] and the Reading Difficulty (Passage) dataset[3] [Chen et al., 2013]. The dataset statistics are summarized in Table 1.

The FaceAge dataset [Pavlichenko and Ustalov, 2021] is specifically tailored for large scale evaluation of various crowdsourcing methods and is a widely adopted benchmark for crowdsourced pairwise comparison experiments [Ferrara et al., 2024]. On the FaceAge dataset, `BoRaEM` achieves the best performance, followed by `HBTL` and `CrowdBT`. The Passage dataset is comparatively much smaller. On the PASSAGE dataset, we observe that all methods achieve relatively low performance, with Kendall's tau values ranging from 0.29 to 0.38. This can be attributed to the small size of the dataset and the inherent difficulty of modeling preferences in reading difficulty, which is a cognitively complex and subjective task. On this dataset, `CrowdBT` achieves the highest performance, followed by `HBTL` and `BoRaEM`.

Table 2 presents the results on the FaceAge and Passage datasets. `BoRaEM`, `HBTL`, `HTCV`, `FactorBT` and `CrowdBT` remain competitive across both datasets. These results highlight the importance of jointly modeling worker competence and item rewards.

6.4 SPAMMER ADDITION ANALYSIS

We evaluate robustness under adversarial conditions by injecting synthetic spammers commonly studied in the crowdsourcing literature. Specifically, we consider four archetypes: (i) *random choosers*, who respond uniformly

[2] `http://www-personal.umich.edu/~kevynct/datasets/wsdm_rankagg_2013_readability_crowdflower_data.csv`

[3] `https://github.com/Toloka/IMDB-WIKI-SbS/tree/main`

Table 1: Summary statistics of the evaluation datasets.

Attribute	FaceAge Dataset	Reading Difficulty or Passage Dataset
Items (K)	9,150	472
Workers (M)	4,091	624
Comparisons (N)	250,249	11,763
Description	Comparison of two face images to identify the older individual.	Pairwise difficulty assessment of reading passages.
Ground Truth	Chronological age (years)	Standardized reading levels
Spammer Handling	Removal based on performance on "gold" instances (large age gaps).	Not specified in original study.

Table 2: Benchmark results on FaceAge and Passage datasets. Accuracy and Weighted Accuracy are reported as percentages, while Kendall's Tau is reported as a decimal. Best overall results are shown in **bold**.

Method	Accuracy (%)	Weighted Accuracy (%)	Kendall's Tau	Runtime (s)
FaceAge Dataset				
Simple BT	79.01	87.20	0.5754	0.89 ± 0.05
RC	78.04	86.44	0.5582	0.07 ± 0.03
FactorBT	79.16	87.30	0.5785	169.41 ± 1.16
BARP	79.08	87.26	0.5769	51.08 ± 2.85
CrowdBT	79.17	87.31	0.5787	138.16 ± 3.86
HTCV	79.17	87.33	0.5786	41.31 ± 1.41
HBTL	79.20	87.35	0.5793	83.10 ± 3.32
BoRaEM (Ours)	**79.21**	**87.36**	**0.5795**	6.55 ± 0.43
Passage Dataset				
Simple BT	68.04	74.33	0.3407	1.39 ± 0.13
RC	65.22	71.30	0.2892	0.02 ± 0.04
FactorBT	69.47	75.58	0.3676	100.85 ± 2.82
BARP	68.19	74.51	0.3435	32.05 ± 2.38
CrowdBT	**70.02**	**75.96**	**0.3779**	84.25 ± 1.84
HTCV	69.30	75.30	0.3643	46.19 ± 1.83
HBTL	69.53	75.46	0.3688	77.72 ± 2.85
BoRaEM (Ours)	69.59	75.56	0.3698	8.28 ± 0.71

at random without regard to item quality [Vuurens et al., 2011]; (ii) *position-biased* spammers, who systematically favor one side of the comparison (e.g., always selecting the left or right option), also referred to as uniform spammers in prior work [Day, 1969, Xu et al., 2016, Vuurens et al., 2011]; (iii) *malicious adversaries*, who intentionally provide inverted or misleading responses to degrade aggregation quality [Kleindessner and Awasthi, 2018, Steinhardt et al., 2016]; and (iv) a *combined* setting, where all aforementioned spammer types are present in equal proportion, reflecting a more realistic and heterogeneous adversarial environment. For fairness, each injected spammer is assigned the same expected number of comparisons as an original worker, ensuring performance differences arise from behavioral characteristics rather than annotation volume.

6.4.1 Results

We mainly focus on the FaceAge dataset, as it is a widely adopted benchmark for crowdsourcing [Pavlichenko and Ustalov, 2021, Ferrara et al., 2024]. Results on the Passage dataset are provided in Appendix H. Figure 2 presents the performance of different crowdsourcing methods in terms of Kendall's tau (τ) as the proportion of various types of spammers in the FaceAge dataset is varied. The results are averaged over 10 random seeds for generating spammers and 10 random seeds for random initializations. Similar trends are observed in terms of accuracy (acc) and weighted accuracy ($wacc$) (see Appendix G).

We add spammers following the procedure of Bugakova et al. [2019]. Specifically, if there are originally M an-

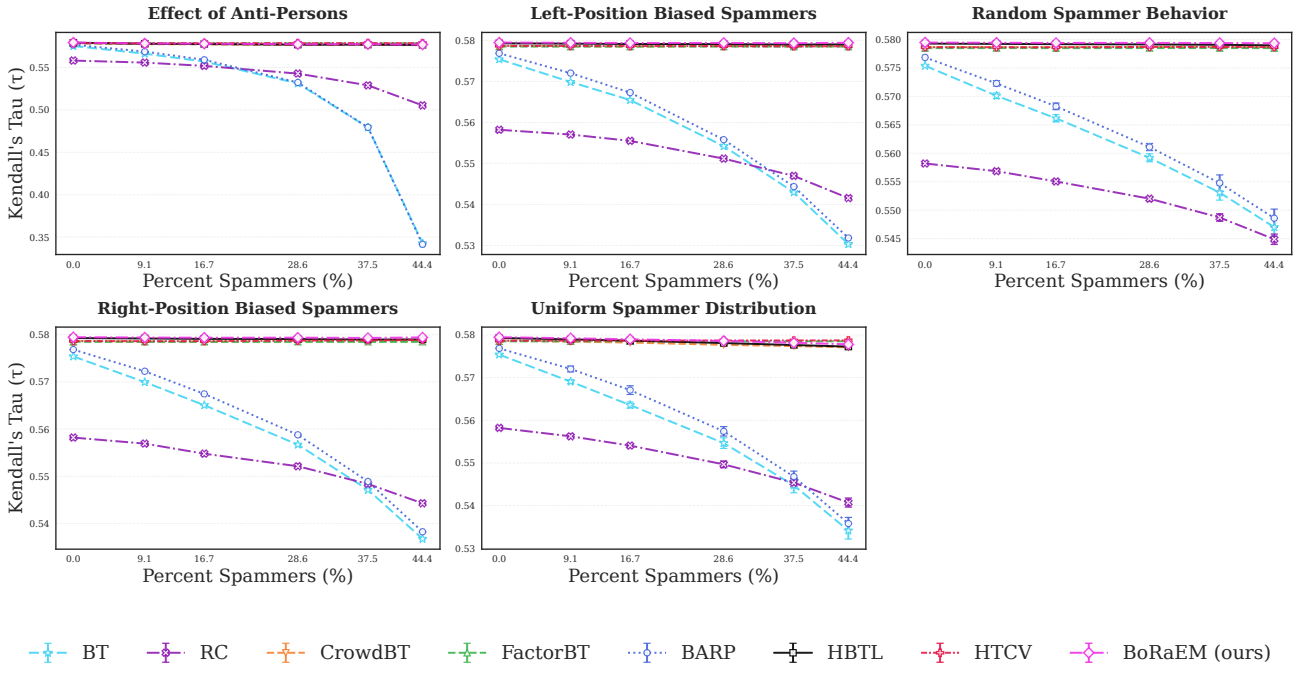


Figure 2: Kendall's Tau (τ) of different ranking methods under varying proportions of spammers for FACEAGE dataset.

notators, we add $0.1M$, $0.2M$, $0.4M$, $0.6M$, and $0.8M$ spammers according to the specified proportions. As the spammer proportion increases from 0% to 44.4%, `BoRaEM`, `HBTL`, `HTCV` consistently perform the best, followed by `CrowdBT` and `FactorBT`, while `BARP`, `BT`, and `RC` exhibit substantial performance drops. These results highlight the importance of explicitly modeling annotator competencies in crowdsourcing. Further, we have performed ablations over intialization and hyperparameters, and the results are detailed in Appendix I.

7 CONCLUSION

In this work, we addressed the problem of rank aggregation from pairwise comparisons in the presence of workers with varying competencies. We adapt the Boltzmann-rational model for modeling each pairwise comparison. We then use the Polya-Gamma random variable augmentation to simplify the logistic function to an exponential form enabling a simple and efficient EM formulation. We achieve tractable inference with theoretical convergence guarantees by treating this problem as a matrix sensing problem. Extensive experiments on both synthetic and real-world datasets demonstrate the advantages of using our algorithm over several baselines and the algorithm remains robust even in the presence of a large fraction of spammers, highlighting its effectiveness for reliable rank aggregation in crowdsourced settings.

Future work includes designing models for robust RLHF that can jointly predict item feature importance vectors while effectively identifying and mitigating the influence of spammers.

Author Contributions

Arpit Agarwal conceived the idea of modeling annotator competence using the Boltzmann Rational Model, identified its connection to the matrix sensing problem, and led the manuscript preparation. Kaustubh Shivshankar Shejole proposed the use of Pólya-Gamma augmentation, developed the method, conducted empirical studies, and generated the experimental visualizations. Tanish Agarwal developed the theoretical foundations of the proposed approach under the guidance of Arpit Agarwal and Avishek Ghosh.

Acknowledgements

We would like to thank the Organizing Committee, Area Chairs, Senior Area Chairs, Meta-Reviewer, and the anonymous reviewers of UAI 2026 for their valuable feedback, constructive suggestions, and insightful comments, which helped improve the quality and presentation of this work.

The first author gratefully acknowledges Kush Mangukiya for contributions during the early stages of this research and Pratham Shah, Manish Kumar for their continuous support and encouragement.

References

Michiel Bakker, Martin Chadwick, Hannah Sheahan, Michael Tessler, Lucy Campbell-Gillingham, Jan Balaguer, Nat McAleese, Amelia Glaese, John Aslanides, Matt Botvinick, and Christopher Summerfield. Fine-tuning language models to find agreement among humans with diverse preferences. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 38176–38189. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/f978c8f3b5f399cae464e85f72e28503-Paper-Conference.pdf.

D. P. Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, Mar 1997. ISSN 1476-9360. doi: 10.1057/palgrave.jors.2600425. URL https://doi.org/10.1057/palgrave.jors.2600425.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Nadezhda Bugakova, Valentina Fedorova, Gleb Gusev, and Alexey Drutsa. Aggregation of pairwise comparisons with reduction of biases. *arXiv preprint arXiv:1906.03711*, 2019.

Emmanuel J Candes and Yaniv Plan. Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *arXiv preprint arXiv:1001.0339*, 2010.

Xi Chen, Paul N Bennett, Kevyn Collins-Thompson, and Eric Horvitz. Pairwise ranking aggregation in a crowd-sourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202, 2013.

Oliver Daniels-Koch and Rachel Freedman. The expertise problem: Learning from specialized feedback. In *NeurIPS ML Safety Workshop*, 2022. URL https://openreview.net/forum?id=I7K975-H1Mg.

Ralph L Day. Position bias in paired product tests, 1969.

Xinyi Ding, Zhenjie Zhang, Zhuangmiao Yuan, Tao Han, Huamao Gu, and Yili Fang. Effectiveness of malicious behavior and its impact on crowdsourcing. In *CCF Conference on Computer Supported Cooperative Work and Social Computing*, pages 118–132. Springer, 2022.

Antonio Ferrara, Francesco Bonchi, Francesco Fabbri, Fariba Karimi, and Claudia Wagner. Bias-aware ranking from pairwise comparisons. *Data Mining and Knowledge Discovery*, 38(4):2062–2086, 2024. doi: https://doi.org/10.1007/s10618-024-01024-z.

Ujwal Gadiraju, Ricardo Kawase, Stefan Dietze, and Gianluca Demartini. Understanding malicious behavior in crowdsourcing platforms: The case of online surveys. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, pages 1631–1640, 2015.

Prateek Jain and Purushottam Kar. *Non-convex Optimization for Machine Learning*. 01 2017. ISBN 9781680833690. doi: 10.1561/9781680833690.

Hong Jun Jeon, Smitha Milli, and Anca Dragan. Reward-rational (implicit) choice: A unifying formalism for reward learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4415–4426. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/2f10c1578a0706e06b6d7db6f0b4a6af-Paper.pdf.

Tao Jin, Pan Xu, Quanquan Gu, and Farzad Farnoud. Rank aggregation via heterogeneous thurstone preference models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4353–4360, 2020. doi: https://doi.org/10.1609/aaai.v34i04.5860.

Saikishore Kalloori, Francesco Ricci, and Rosella Gennari. Eliciting pairwise preferences in recommender systems. In *Proceedings of the 12th acm conference on recommender systems*, pages 329–337, 2018.

Matthaeus Kleindessner and Pranjal Awasthi. Crowdsourcing with arbitrary adversaries. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2708–2717. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/kleindessner18a.html.

Christian List. Social Choice Theory. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2022 edition, 2022.

Sahand Negahban, Sewoong Oh, and Devavrat Shah. Iterative ranking from pair-wise comparisons. In F. Pereira, C.J. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/9adeb82fffb5444e81fa0ce8ad8afe7a-Paper.pdf.

Dohyung Park, Anastasios Kyrillidis, Constantine Carmanis, and Sujay Sanghavi. Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial*

Intelligence and Statistics, volume 54 of *Proceedings of Machine Learning Research*, pages 65–74. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/park17a.html.

Nikita Pavlichenko and Dmitry Ustalov. Imdb-wiki-sbs: An evaluation dataset for crowdsourced pairwise comparisons. *arXiv preprint arXiv:2110.14990*, 2021.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

Jacob Steinhardt, Gregory Valiant, and Moses Charikar. Avoiding imposters and delinquents: Adversarial crowdsourcing and peer prediction. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/0a87257e5308197df43230edf4ad1dae-Paper.pdf.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Rezvan Velayati, Andrei Gurca, Mehdi Bagherzadeh, and Wim Vanhaverbeke. Crowdsourcing effectiveness at a problem level: A meta-synthesis. In *Academy of Management Proceedings*, volume 2025, page 19087. Academy of Management Valhalla, NY 10595, 2025.

Jeroen Vuurens, Arjen P de Vries, and Carsten Eickhoff. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *Proc. ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR'11)*, pages 21–26, 2011.

Qianqian Xu, Jiechao Xiong, Xiaochun Cao, and Yuan Yao. False discovery rate control and statistical quality assessment of annotators in crowdsourced ranking. In *International conference on machine learning*, pages 1282–1291. PMLR, 2016.

Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010. URL https://www.cs.cmu.edu/~bziebart/publications/maximum-causal-entropy.pdf.

Finding the Signal in the Spam (Supplementary Material)

Kaustubh Shivshankar Shejole[1,*] **Tanish Agarwal**[1,*] **Arpit Agarwal**[1] **Avishek Ghosh**[1]

[*]Equal contribution
[1]IIT Bombay, Mumbai, India – 400076

A MORE DETAILS ON BORAEM DERIVATION

A.1 PÓLYA-GAMMA RANDOM VARIABLES

Following Polson et al. [2013], if ω is a Pólya-Gamma distributed random variable with parameters $b > 0$ and $c \in \mathbb{R}$, denoted $\omega \sim \mathrm{PG}(b, c)$, then it is equal in distribution to an infinite weighted sum of gamma random variables:

$$\omega \overset{d}{=} \frac{1}{2\pi^2} \sum_{k=1}^{\infty} \frac{g_k}{\left(k - \frac{1}{2}\right)^2 + \frac{c^2}{4\pi^2}}, \quad g_k \sim \mathrm{Gamma}(b, 1).$$

Here, "$\overset{d}{=}$" denotes equality in distribution, and $g_k \sim \mathrm{Gamma}(b, 1)$ are independent Gamma random variables. Note that this is not the density function. Instead, equality in distribution means that the PG random variable on the left-hand side has the same cumulative distribution function as the random variable on the right-hand side.

Polson et al. [2013] showed that all finite moments of ω admit closed form expressions; in particular,

$$\mathbb{E}[\omega] = \frac{b}{2c} \tanh\left(\frac{c}{2}\right). \tag{14}$$

They further proved the following two identities for $\omega \sim \mathrm{PG}(b, 0)$, which are central to Pólya-Gamma data augmentation for logistic models:

1. For any real ψ, one has
$$\frac{(e^{\psi})^a}{(1 + e^{\psi})^b} = 2^{-b} e^{\kappa\psi} \int_0^{\infty} e^{-\omega\psi^2/2}\, p(\omega \mid b, 0)\, d\omega, \tag{15}$$
where $\kappa = a - \frac{b}{2}$, and $p(\omega \mid b, 0)$ is the density of a $\mathrm{PG}(b, 0)$ random variable.
2. The conditional density of ω given ψ is again Pólya–Gamma with parameters b and ψ, that is,
$$(\omega \mid \psi) \sim \mathrm{PG}(b, \psi). \tag{16}$$

A.2 LOG LIKELIHOOD DERIVATION

Let us consider the problem formulation given in Section 3, where there is a set of K items, and one observes outcomes of pairwise comparisons between these items, elicited from a set of M crowdworkers. Specifically, we are given a dataset $\mathcal{D} = \{(w_i, l_i, s_i)\}_{i=1}^{N}$ where there are N pairwise comparisons; the i-th comparison is performed by worker $s_i \in [M]$ where $w_i \in [K]$ is the winning item and $l_i \in [K]$ is the losing item such that l_i is not equal to w_i.

Consider Equation (15). Let $\eta_i = \beta_{s_i}(r_{w_i} - r_{l_i})$. For $a = 1$, $b = 1$, and $\psi = \eta_i$, we have $\kappa = \frac{1}{2}$ and $p(\omega_i) = \mathrm{PG}(\omega_i \mid 1, 0)$, which allows us to write

$$\frac{1}{1+e^{-\eta_i}} = \frac{e^{\eta_i}}{1+e^{\eta_i}} = \frac{1}{2}e^{\eta_i/2}\int_0^\infty \exp\left(-\frac{\omega_i\,\eta_i^2}{2}\right)\,p(\omega_i)\,d\omega_i \tag{17}$$

Let $\boldsymbol{\theta} = \{\mathbf{r}, \boldsymbol{\beta}\}$. Therefore, referring to Equation (10) and (17), we can write each term of the likelihood as

$$p(w_i, l_i, s_i|\boldsymbol{\theta}) = \frac{1}{1+\exp\{-\beta_{s_i}(r_{w_i} - r_{l_i})\}} \tag{18}$$

$$\begin{aligned} &= \frac{1}{2}e^{\eta_i/2}\int_0^\infty \exp\left(-\frac{\omega_i\eta_i^2}{2}\right)\,p(\omega_i)\,d\omega_i \\ &= \frac{1}{2}\int_0^\infty \exp\left(\frac{\eta_i}{2} - \frac{\omega_i\eta_i^2}{2}\right)p(\omega_i)\,d\omega_i \end{aligned} \tag{19}$$

which induces the augmented joint density

$$p(w_i, l_i, s_i, \omega_i|\boldsymbol{\theta}) \propto \exp\left(-\frac{\omega_i\eta_i^2}{2} + \frac{\eta_i}{2}\right)p(\omega_i),$$

Taking the logarithm and ignoring additive constants that do not depend on $\boldsymbol{\theta}$, we get the log-likelihood for the i^{th} comparison as

$$\log(p(w_i, l_i, s_i, \omega_i|\boldsymbol{\theta})) \propto \log\left[\exp\left(\frac{1}{2}\eta_i - \frac{\omega_i}{2}\,\eta_i^2\right)\,p(\omega_i)\right] + \text{const} \tag{20}$$

$$\begin{aligned} &= -\frac{\omega_i\eta_i^2}{2} + \frac{\eta_i}{2}, \\ &= \frac{1}{2}\,\left(\beta_{s_i}(r_{w_i} - r_{l_i})\right) - \frac{\omega_i}{2}\,\left(\beta_{s_i}(r_{w_i} - r_{l_i})\right)^2. \end{aligned} \tag{21}$$

Summing over all N comparisons in $\mathcal{D}$ gives the *complete data* log-likelihood, given all ω_i,

$$\ell(\boldsymbol{\theta}, \omega_i; \mathcal{D}) = \frac{1}{2}\sum_{i\in[N]}\left[\,\beta_{s_i}(r_{w_i} - r_{l_i}) - \omega_i\,\left(\beta_{s_i}(r_{w_i} - r_{l_i})\right)^2\right], \tag{22}$$

up to additive constants independent of $\boldsymbol{\theta}$.

A.3 EXPECTATION-MAXIMIZATION

A.3.1 E-Step

Given the current parameter estimate $\theta^{(t)}$ and let $\boldsymbol{\omega}$ denote the set of all w_i for $i = 1$ to N comparisons, we compute

$$Q\left(\theta \mid \theta^{(t)}\right) = \mathbb{E}_{\omega|\mathcal{D},\,\theta^{(t)}}\left[\,\ell(\boldsymbol{\theta}, \boldsymbol{\omega}; \mathcal{D})\right] = \frac{1}{2}\sum_{(w_i,l_i,s_i)\in\mathcal{D}}\mathbb{E}_{\omega_i|\theta^{(t)}}\left[\,\beta_{s_i}\,(r_{w_i} - r_{l_i}) - \omega_i\,\beta_{s_i}^2\,(r_{w_i} - r_{l_i})^2\right].$$

Noting that the only term depending on ω_i is $\omega_i\,\beta_{s_i}^2\,(r_{w_i} - r_{l_i})^2$, we define

$$\kappa_i^{(t)} = \mathbb{E}\left[\omega_i \mid \theta^{(t)}\right].$$

Then,

$$Q\left(\theta \mid \theta^{(t)}\right) = \frac{1}{2}\sum_{(w_i,l_i,s_i)\in\mathcal{D}}\left[\beta_{s_i}\,(r_{w_i} - r_{l_i}) - \kappa_i^{(t)}\,\beta_{s_i}^2\,(r_{w_i} - r_{l_i})^2\right]. \tag{23}$$

Computing $\kappa_i^{(t)}$. At iteration t, we compute the conditional expectation $\kappa_i^{(t)} = \mathbb{E}[\omega_i \mid \eta_i^{(t)}]$ for the latent variables $\omega_i \sim \mathrm{PG}(1, \eta_i^{(t)})$, where $\eta_i^{(t)} = \beta_{s_i}^{(t)}(r_{w_i}^{(t)} - r_{l_i}^{(t)})$. Invoking Eq. (14) yields:

$$\kappa_i^{(t)} = \mathbb{I}\left(\eta_i^{(t)} \neq 0\right) \frac{\tanh\left(\eta_i^{(t)}/2\right)}{2\eta_i^{(t)}} + \mathbb{I}\left(\eta_i^{(t)} = 0\right) \frac{1}{4}. \tag{24}$$

A.3.2 M-Step

Updating β

$$Q\left(\theta \mid \theta^{(t)}\right) = \sum_{(w_i, l_i, s_i) \in \mathcal{D}} \left[\frac{1}{2}\beta_{s_i}\left(r_{w_i} - r_{l_i}\right) - \frac{1}{2}\kappa_i^{(t)}\beta_{s_i}^2\left(r_{w_i} - r_{l_i}\right)^2\right],$$

where $\kappa_i^{(t)} = \mathbb{E}\left[\omega_i \mid \theta^{(t)}\right]$. We first maximize Q with respect to each β_s. Let

$$I_s = \{\, i : s_i = s\}$$

denote the index set of comparisons done by worker s. Define $d_i = r_{w_i} - r_{l_i}$, for all $i \in I_s$. Then, the fraction of Q depending on β_s is

$$Q_k\left(\beta_s\right) = \sum_{i \in I_s}\left[\frac{1}{2}\beta_s\, d_i - \frac{1}{2}\kappa_i^{(t)}\beta_s^2\, d_i^2\right]. \tag{25}$$

For maximization, we require $\frac{\partial Q_k}{\partial \beta_s} = 0$, that is,

$$\frac{\partial Q_k}{\partial \beta_s} = \sum_{i \in I_s}\left[\frac{1}{2}\, d_i - \kappa_i^{(t)}\beta_s\, d_i^2\right] = 0, \tag{26}$$

which on rearranging gives

$$\frac{1}{2}\sum_{i \in I_s} d_i - \beta_s \sum_{i \in I_s}\kappa_i^{(t)}\, d_i^2 = 0 \Longrightarrow \beta_s = \frac{\frac{1}{2}\sum_{i \in I_s} d_i}{\sum_{i \in I_s}\kappa_i^{(t)}\, d_i^2} = \frac{\sum_{i \in I_s} d_i}{2\sum_{i \in I_s}\kappa_i^{(t)}\, d_i^2}.$$

Updating r We now maximize Q with respect to each r_k. Define the index sets

$$W_j = \{\, i : w_i = j\} \text{ and } L_j = \{\, i : l_i = j\}$$

where W_j denotes the set of comparisons where item j won, and L_j denotes the set of comparisons where item j lost. Then,

$$\frac{\partial Q}{\partial r_k} = \left[\sum_{i \in W_j}\left[\frac{1}{2}\beta_{s_i} - \beta_{s_i}^2\,\kappa_i^{(t)}\,\left(r_k - r_{l_i}\right)\right] + \sum_{i \in L_j}\left[-\frac{1}{2}\beta_{s_i} + \beta_{s_i}^2\,\kappa_i^{(t)}\,\left(r_{w_i} - r_k\right)\right]\right] = 0. \tag{27}$$

which on rearranging gives

$$r_k = \frac{A_k}{B_k}$$

where

$$A_k = \sum_{i \in W_j}\left[\frac{1}{2}\beta_{s_i} + \beta_{s_i}^2\,\kappa_i^{(t)}\, r_{l_i}\right] + \sum_{i \in L_j}\left[-\frac{1}{2}\beta_{s_i} + \beta_{s_i}^2\,\kappa_i^{(t)}\, r_{w_i}\right], \text{ and,}$$
$$B_k = \sum_{i \in W_j}\beta_{s_i}^2\,\kappa_i^{(t)} + \sum_{i \in L_j}\beta_{s_i}^2\,\kappa_i^{(t)}.$$

Observe that for fixed $\boldsymbol{\beta}$, these update equations can be written as a linear system $H\boldsymbol{r} = \mathbf{b}$ as follows.

$$B_k r_k - \sum_{i \in W_j}\beta_{s_i}^2\kappa_i r_{l_i} - \sum_{i \in L_j}\beta_{s_i}^2\kappa_i r_{w_i} = \sum_{i \in W_j}\left(\frac{1}{2}\beta_{s_i}\right) + \sum_{i \in L_j}\left(-\frac{1}{2}\beta_{s_i}\right)$$

The coefficient matrix H and vector $\mathbf{b}$ are given by

$$H_{jl} = \begin{cases} \sum_{i \in \mathcal{D}_{jl}} -\beta_{s_i}^2 \kappa_i^{(t)}, & j \neq l \\ B_j, & j = l \end{cases}$$

and

$$b_j = \sum_{i \in W_j} \frac{1}{2} \beta_{s_i} + \sum_{i \in L_j} \left(-\frac{1}{2} \beta_{s_i} \right)$$

By construction, H is symmetric, $H_{jl} \leq 0$ for $j \neq l$, and $H_{jj} = \sum_{l \neq j} |H_{jl}|$. Thus, H is diagonally dominant with non-negative diagonal entries, and it is positive semidefinite. We can further observe that $H\mathbf{1} = \mathbf{0}$. Now, suppose $\mathbf{x}$ satisfies $H\mathbf{x} = \mathbf{0}$ and let j be such that $x_j = \max_i x_i$. Using the zero row sum property, we get that

$$0 = (H\mathbf{x})_j = \sum_{l \neq j} H_{jl}(x_l - x_j)$$

Since $H_{jl} \leq 0$ and $x_l - x_j \leq 0$, each term is nonnegative, hence each term must be zero. Therefore, $x_j = x_l$ whenever $H_{jl} \neq 0$. If the comparison graph induced by the nonzero off-diagonal entries of H is connected, all entries of $\mathbf{x}$ are equal, hence $\mathbf{x} = c\mathbf{1}$. The centering constraint $\mathbf{1}^\top \mathbf{x} = 0$ forces $c = 0$, so $\mathbf{x} = \mathbf{0}$. Therefore, H is positive definite on the mean-zero subspace, and the constrained linear system admits a unique solution.

B CONVERGENCE ANALYSIS

Let $\mathcal{D}$ be the dataset of pairwise comparisons, $N = |\mathcal{D}|$ denotes the number of datapoints, $K = |\boldsymbol{r}|$ the number of items, and $M = |\boldsymbol{\beta}|$ the number of workers. For each worker $j \in [M]$, we constrain $\beta_j \in [-1, 1]$. We assume that the rewards are bounded as $r_i \in [-R, R]$ for all items $l \in [K]$ for some fixed constant $R > 0$. In each iteration t of the EM algorithm, we independently set

$$c_i^2 = \kappa_i^{(t)} = \frac{1}{2\eta_i^{(t)}} \tanh\left(\frac{\eta_i^{(t)}}{2} \right), \quad \text{where } \eta_i^{(t)} = \beta_{s_i}^{(t)} \left(r_{w_i}^{(t)} - r_{l_i}^{(t)} \right).$$

Since $\beta_{s_i} \in [-1, 1]$ and $r_{w_i}, r_{l_i} \in [-R, R]$, it follows that $|\eta_i| \leq 2R$.

The function $f(x) = \frac{\tanh(x/2)}{2x}$ is positive, even, and monotonically decreasing for $x > 0$. Therefore, there exists a constant

$$\zeta = \frac{1}{4R} \tanh(R) > 0$$

such that $c_i^2 \in [\zeta, 1/4]$ for all i.

For each worker s, let G_s denote the comparison graph induced by the worker's pairwise comparisons $\mathcal{D}_s$. We assume that G_s is acyclic.

Finally, we assume that the worker index and the two item indices (denoted by random variables B, W, and L, respectively) are independent and uniformly distributed as $B \sim \text{Uniform}([M]), W, L \sim \text{Uniform}([K])$. The comparison outcome is then generated according to 2. Under this model, we can show the following lemma.

Lemma 1. *Define matrix operators $A_i \in \mathbb{R}^{M \times K}$ for each $i \in [N]$ as follows:*

$$A_i^\top = 2c_i\alpha \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array} \begin{array}{c} \overset{\displaystyle s_i}{} \\ \begin{bmatrix} 0 & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & +1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & -1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ \\ \\ w_i \\ \\ l_i \\ \\ \end{array}$$

where $+1$ appears at position (w_i, s_i), -1 at position (l_i, s_i), with all other entries equal to 0, and $\alpha = \sqrt{MK/2N}$. Equivalently, $A_i = 2c_i\alpha\, \boldsymbol{e_{s_i}}(\boldsymbol{e_{w_i}} - \boldsymbol{e_{l_i}})^\top$. Let $X \in \mathbb{R}^{M\times K}$ be any matrix of arbitrary rank r with $X\mathbf{1} = \mathbf{0}$, where $\mathbf{1}$ and $\mathbf{0}$ denote the all-ones and all-zeros vectors, respectively. Define the linear operator $\mathcal{A}\colon \mathbb{R}^{M\times K} \to \mathbb{R}^N$ as $\mathcal{A}(X) = (\langle A_1, X\rangle, \ldots, \langle A_N, X\rangle)^\top$. Then, for any $\eta > 0$, there exists $\mu > 0$ and $\delta > 0$ such that with probability at least $1 - \eta$:

$$(1-\mu)\|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1+\mu)\|X\|_F^2\,,$$

provided that

$$N \geq C\,\frac{MK}{\delta^2}\ln\left(\frac{2}{\eta}\right),$$

where $C > 0$ is some constant.

Proof. We first show that $\mathcal{A}$ satisfies RIP in expectation, and we subsequently bound the deviation of $\|\mathcal{A}(X)\|_2^2$ from its expectation to achieve the desired result.

Consider $\langle A_i, X\rangle^2 = 4c_i^2\alpha^2(X_{s_i,w_i} - X_{s_i,l_i})^2$. Under uniform sampling of B, W, L and the assumption $X\mathbf{1} = \mathbf{0}$, we have that

$$\mathbb{E}_{B,W,L}\langle A_i, X\rangle^2 = \mathbb{E}_B\left[\mathbb{E}_{W,L}\left[\langle A_i, X\rangle^2 | B = s_i\right]\right]$$

Conditioned on $B = s_i$, we get the following expression.

$$\begin{aligned}
\mathbb{E}_{W,L}\left[\langle A_i, X\rangle^2\right] &= 4c_i^2\alpha^2\mathbb{E}_{W,L}\left[(X_{s_i,w_i} - X_{s_i,l_i})^2\right] \\
&= 4c_i^2\alpha^2\left(\mathbb{E}_{W,L}X_{s_i,w_i}^2 + \mathbb{E}_{W,L}X_{s_i,l_i}^2 - 2\mathbb{E}_{W,L}X_{s_i,w_i}X_{s_i,l_i}\right) \\
&= 8c_i^2\alpha^2\left(\mathbb{E}_{W,L}X_{s_i,w_i}^2 - \left(\mathbb{E}_{W,L}X_{s_i,w_i}\right)^2\right)
\end{aligned}$$

Now, note that

$$\mathbb{E}_{W,L}X_{s_i,w_i}^2 = \frac{1}{K}\sum_{j=1}^K X_{s_i,j}^2 \quad \text{and} \quad \mathbb{E}_{W,L}X_{s_i,w_i} = \frac{1}{K}\sum_{j=1}^K X_{s_i,j},$$

and define $S_b = \sum_{j=1}^K X_{b,j}$ and $T_b = \sum_{j=1}^K X_{b,j}^2$. Observe that $S_b = 0$ for every $b \in [M]$ since $X\mathbf{1} = \mathbf{0}$, which implies that $\mathbb{E}_{W,L}X_{s_i,w_i} = 0$, and $\mathbb{E}_{W,L}[\langle A_i, X\rangle^2] = 4c_i^2\alpha^2(\frac{2}{K}T_{s_i})$. Finally,

$$\mathbb{E}_{B,W,L}\langle A_i, X\rangle^2 = \frac{1}{M}\left(4c_i^2\alpha^2\cdot\frac{2}{K}\sum_{j=1}^M T_{s_j}\right) = \frac{1}{M}\left(4c_i^2\alpha^2\cdot\frac{2}{K}\sum_{j=1}^M\sum_{i=1}^K X_{ji}^2\right) = \frac{1}{M}4c_i^2\alpha^2\frac{2}{K}\|X\|_F^2 = \frac{4c_i^2}{N}\|X\|_F^2, \tag{28}$$

where the last equality follows from the definition of α. Summing over all N datapoints, we obtain:

$$\mathbb{E}_{B,W,L}\|\mathcal{A}(X)\|_2^2 = \sum_{i=1}^N \mathbb{E}_{B,W,L}\langle A_i, X\rangle^2 = \|X\|_F^2\sum_{i=1}^N\frac{4c_i^2}{N}$$

Note that $4\zeta \leq 4c_i^2 \leq 1$. Take $\epsilon = 1 - 4\zeta > 0$, such that

$$(1-\epsilon)\|X\|_F^2 \leq 4\zeta\|X\|_F^2 \leq \mathbb{E}_{B,W,L}\|\mathcal{A}(X)\|_2^2 \leq \|X\|_F^2 \leq (1+\epsilon)\|X\|_F^2. \tag{29}$$

Now, we will bound the deviation of $\|\mathcal{A}(X)\|_2^2$ from its expectation. Define random variables $Y_i = \langle A_i, X\rangle^2$; then $\sum_{i=1}^N Y_i = \|\mathcal{A}(X)\|_2^2$. We will first bound Y_i. Note that $Y_i \geq 0$. Further,

$$Y_i = \langle A_i, X\rangle^2 = 4c_i^2\alpha^2(X_{s_i,w_i} - X_{s_i,l_i})^2 \leq 4c_i^2\alpha^2(|X_{s_i,w_i}| + |X_{s_i,l_i}|)^2.$$

Since each $X_{ij}^2 \leq \|X\|_F^2$, we have that $|X_{ij}| \leq \|X\|_F$, and consequently,

$$0 \leq Y_i \leq 16c_i^2\alpha^2\|X\|_F^2 \tag{30}$$

Similarly, $\mathbb{E}Y_i^2 \leq \max Y_i \cdot \mathbb{E}Y_i$. We know from Equation (28) that $\mathbb{E}Y_i = \mathbb{E}\langle A_i, X\rangle^2 = \frac{4c_i^2}{N}\|X\|_F^2$, which gives

$$\mathbb{E}Y_i^2 \leq 16c_i^2\alpha^2\|X\|_F^2 \cdot \frac{4c_i^2}{N}\|X\|_F^2 = \frac{64c_i^4\alpha^2}{N}\|X\|_F^4 \leq \frac{64c_i^2\alpha^2}{N}\|X\|_F^4 \,, \tag{31}$$

where the final inequality follows from the fact that $c_i < 1$. Define random variables $Z_i = Y_i - EY_i$. Then, Z_i are independent and zero-mean, $|Z_i| \leq |Y_i| + |\mathbb{E}Y_i| \leq 2|Y_i| \leq 32c_i^2\alpha^2\|X\|_F^2$, and, $\mathbb{E}Z_i^2 \leq \mathbb{E}Y_i^2 \leq \frac{64c_i^2\alpha^2}{N}\|X\|_F^4$.

Then, using Bernstein's inequality on Z_i, we get,

$$\begin{aligned}\Pr\left(\left|\sum_{i=1}^N Z_i\right| > \delta\|X\|_F^2\right) &\leq 2\exp\left(-\frac{\delta^2\|X\|_F^4/2}{N\cdot\frac{64c_i^2\alpha^2}{N}\|X\|_F^4 + \frac{1}{3}\cdot 16\delta c_i^2\alpha^2\|X\|_F^4}\right) \leq 2\exp\left(-\frac{\delta^2\|X\|_F^4}{80c_i^2\alpha^2\|X\|_F^4}\right) \\ &= 2\exp\left(-\frac{\delta^2}{80c_i^2\alpha^2}\right).\end{aligned}$$

For this to be bounded above by η, we need

$$2\exp\left(-\frac{\delta^2}{80c_i^2\alpha^2}\right) = 2\exp\left(-\frac{\delta^2 N}{40c_i^2 MK}\right) \leq \eta \implies N \geq C\frac{MK}{\delta^2}\log\left(\frac{2}{\eta}\right),$$

for some constant $C > 0$. Therefore, with probability at least $1-\eta$, we get that

$$\begin{aligned}\left|\sum_{i=1}^N Z_i\right| \leq \delta\|X\|_F^2 &\implies \left|\sum_{i=1}^N \langle A_i, X\rangle^2 - \sum_{i=1}^N \mathbb{E}\langle A_i, X\rangle^2\right| \leq \delta\|X\|_F^2 \\ &\implies \left|\|\mathcal{A}(X)\|_2^2 - \mathbb{E}\|\mathcal{A}(X)\|_2^2\right| \leq \delta\|X\|_F^2 \\ &\implies (1-\mu)\|X\|_F^2 \leq \|\mathcal{A}(X)\|_2^2 \leq (1+\mu)\|X\|_F^2\end{aligned}$$

with $\mu = \epsilon + \delta$. □

We now show that the M-step optimization problem can be written as a matrix sensing problem. Recall that the M-step at iteration t maximizes

$$\begin{aligned}Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^t) &= \frac{1}{2}\sum_{i=1}^N \left(\beta_{s_i}\left(r_{w_i} - r_{l_i}\right) - c_i^2\beta_{s_i}^2\left(r_{w_i} - r_{l_i}\right)^2\right) \\ &= -\frac{1}{2}\sum_{i=1}^N \left(c_i^2\beta_{s_i}^2\left(r_{w_i} - r_{l_i}\right)^2 - 2\left(\frac{1}{2c_i}\right)c_i\beta_{s_i}\left(r_{w_i} - r_{l_i}\right) + \frac{1}{4c_i^2} - \frac{1}{4c_i^2}\right) \\ &= -\frac{1}{2}\sum_{i=1}^N \left(c_i\beta_{s_i}\left(r_{w_i} - r_{l_i}\right) - \frac{1}{2c_i}\right)^2 + \sum_{i=1}^N \frac{1}{8c_i^2}\end{aligned}$$

Therefore,

$$\begin{aligned}
\arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^t) &= \arg\max_{\boldsymbol{\theta}} \left(-\frac{1}{2} \sum_{i=1}^{N} \left(c_i \beta_{s_i} \left(r_{w_i} - r_{l_i} \right) - \frac{1}{2c_i} \right)^2 + \sum_{i=1}^{N} \frac{1}{8c_i^2} \right) \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \left(c_i \beta_{s_i} \left(r_{w_i} - r_{l_i} \right) - \frac{1}{2c_i} \right)^2 \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \left(2c_i \alpha\, \beta_{s_i} \left(r_{w_i} - r_{l_i} \right) - \frac{\alpha}{c_i} \right)^2 \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \left(\langle A_i, \boldsymbol{\beta} \boldsymbol{r}^\top \rangle - \frac{\alpha}{c_i} \right)^2
\end{aligned}$$

where the second equality follows from the fact that c_i are constants, and the fourth equality follows from the definition of matrices A_i. Define the vector $\mathbf{u}^{(t)} \in \mathbb{R}^N$ with $u_i^{(t)} = \frac{\alpha}{c_i}$. Then,

$$\arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^t) = \arg\min_{\boldsymbol{\theta}} \left\| \mathcal{A}(\boldsymbol{\beta} \boldsymbol{r}^\top) - \mathbf{u}^{(t)} \right\|_2^2,$$

We now invoke existing matrix sensing guarantees. Let $X_t^* \in \mathbb{R}^{M \times K}$ denote the underlying true matrix and $X_{1,t}^*$ denote its best rank-1 approximation. Define the distance between a factorization $(\boldsymbol{\beta}, \boldsymbol{r})$ and $X_{1,t}^*$ as

$$\text{DIST}(\boldsymbol{\beta}, \boldsymbol{r}; X_{1,t}^*) = \min_{(\boldsymbol{\beta}^*, \boldsymbol{r}^*) \in \mathcal{X}_1^*} \left\| \begin{bmatrix} \boldsymbol{\beta} \\ \boldsymbol{r} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\beta}^* \\ \boldsymbol{r}^* \end{bmatrix} \right\|_F$$

where $\mathcal{X}_1^*$ is the set of balanced rank-one factorizations for $X_{1,t}^*$, defined as

$$\mathcal{X}_1^* = \left\{ (\boldsymbol{\beta}^*, \boldsymbol{r}^*) : \boldsymbol{\beta}^* \in \mathbb{R}^M, \boldsymbol{r}^* \in \mathbb{R}^K, \boldsymbol{\beta}^* \boldsymbol{r}^{*\top} = X_{1,t}^*, \sigma_1(\boldsymbol{\beta}^*) = \sigma_1(\boldsymbol{r}^*) = \sigma_1(X_{1,t}^*)^{1/2}, \right\}.$$

Then, restating Remark 3 from Park et al. [2017],

Remark 1. ***(High-rank matrix sensing)*** *Suppose that X_t^* is of arbitrary rank, and let $X_{1,t}^*$ denote its best rank-1 approximation. Let $\mathbf{b} = \mathcal{A}(X^*)$ and let $(\boldsymbol{\beta}^*, \boldsymbol{r}^*)$ be a balanced factorization of $X_{1,t}^*$. If $0 \le \delta_2 \le \delta_4 < 0.005$, then, for any local minimum $(\boldsymbol{\beta}, \boldsymbol{r})$ the distance to $(\boldsymbol{\beta}^*, \boldsymbol{r}^*)$ is bounded by*

$$\text{DIST}(\boldsymbol{\beta}, \boldsymbol{r}; X_t^*) \le \frac{1250}{3\sigma_1(X_t^*)} \cdot \|\mathcal{A}(X_t^* - X_{1,t}^*)\|_2.$$

Using this, we can give the following theorem.

Theorem 1. *Let $\{\boldsymbol{\beta}^{(t)}, \boldsymbol{r}^{(t)}\}_{t \ge 0}$ be the sequence of iterates generated by the EM algorithm. Suppose the M-step at each iteration t solves the optimization problem:*

$$(\boldsymbol{\beta}^{(t+1)}, \boldsymbol{r}^{(t+1)}) = \arg\min_{\boldsymbol{\beta}, \boldsymbol{r}} \|\mathcal{A}(\boldsymbol{\beta} \boldsymbol{r}^\top) - \mathbf{u}^{(t)}\|_2^2$$

having global minimum $(\boldsymbol{\theta}^*)^{(t+1)}$, where $u_i^{(t)} = \frac{\alpha}{c_i}$. If $R = 0.1$, the operator $\mathcal{A}$ satisfies Restricted Isometry Property with $\mu_2 = \mu_4 = 0.005$, and every local minimum $\boldsymbol{\theta}^{(t+1)} = \left(\boldsymbol{\beta}^{(t+1)}; \boldsymbol{r}^{(t+1)}\right)$ satisfies:

$$\textit{DIST}(\boldsymbol{\theta}^{(t+1)}, (\boldsymbol{\theta}^*)^{(t+1)}) \le \frac{1250}{3\sigma_1(X_t^*)} \cdot \|\mathcal{A}(X_t^* - X_{1,t}^*)\|_2$$

where $X_t^* \mathbf{1} = \mathbf{0}$ and $X_{1,t}^*$ is the best rank-1 approximation of the true high-rank matrix X_t^*.

Proof. Define $X_t^* \in \mathbb{R}^{M\times K}$ as any matrix consistent with the equations $[\mathcal{A}(X_t^*)]_i = u_i^{(t)}$. From the definition of the sensing matrices A_i, this implies

$$u_i^{(t)} = \frac{\alpha}{c_i} = 2c_i\alpha(X^*_{t,\{s_i,w_i\}} - X^*_{t,\{s_i,l_i\}}) \implies X^*_{t,\{s_i,w_i\}} - X^*_{t,\{s_i,l_i\}} = \frac{1}{2c_i^2}$$

Under the assumption that for each worker s, the comparison graph is acyclic, and with the additional constraint that $\sum_{l\in[K]} X^*_{t,\{s_i,l\}} = 0$, this linear system is always consistent. Thus, a solution X_t^* exists.

Now, our operator $\mathcal{A}$ satisfies RIP on matrices X of arbitrary rank r with $X\mathbf{1} = \mathbf{0}$. Because the coefficients $c_i^2 \in [\zeta, 1/4]$, we can define $R = 0.1$, such that $\zeta > 0.249$ and $\varepsilon < 0.004$ and the RIP constants $\mu_r < 0.005$, independent of r.

Note that X_t^* may not be rank-1. So, we denote by $X^*_{1,t}$ its best rank-1 approximation. Then, by Remark 1, any local minimum of the objective

$$\min_{\boldsymbol{\beta},\boldsymbol{r}} \left\| \mathcal{A}(\boldsymbol{\beta}\boldsymbol{r}^\top) - \mathbf{u}^{(t)} \right\|_2^2$$

satisfies

$$\text{DIST}(\boldsymbol{\beta}, \boldsymbol{r}; X_t^*) \le \frac{1250}{3\sigma_1(X_t^*)} \|\mathcal{A}(X_t^* - X^*_{1,t})\|_2.$$

□

C EVALUATION METRICS: MORE DETAILS

Table 3: Summary of Quantitative Metrics. Here, g_i and s_i denote the ground truth and predicted quality scores for item $i \in D_g$, respectively.

Metric	Formulation	Pros	Cons
Kendall's Tau	$\frac{C-D}{\frac{1}{2}k(k-1)}$	Standardized rank correlation range; robust to outliers.	Treats all pairwise swaps with equal importance.
Accuracy	$\frac{\sum \mathbb{I}(g_i > g_j \wedge s_i > s_j)}{\sum \mathbb{I}(g_i > g_j)}$	Highly interpretable; measures correct ordering probability.	Does not account for the magnitude of score gaps.
Weighted Acc.	$\frac{\sum w_{ij}\,\mathbb{I}(g_i > g_j \wedge s_i > s_j)}{\sum w_{ij}\,\mathbb{I}(g_i > g_j)}$	Penalizes significant ranking errors more heavily.	Metric value depends on the score distribution.

Note: C and D are concordant and discordant pairs; $k = |D_g|$ is the number of items; and $w_{ij} = |g_i - g_j|$ represents the weight of the difference between ground truth scores.

Table 3 summarizes the quantitative metrics used to evaluate the predicted item rewards against the ground truth. Kendall's Tau measures the rank correlation between item pairs, providing a standardized indication of agreement while treating all swaps equally. Accuracy computes the probability of correctly ordering item pairs, offering intuitive interpretability but ignoring the magnitude of differences. Weighted Accuracy extends this by penalizing larger ranking errors more heavily, making it sensitive to the score gaps between items. Together, these metrics provide complementary perspectives on ranking performance.

D CROWDSOURCING METHODS CONSIDERED

In this section, we evaluate all the representative crowdsourcing methods discussed in Section 2. These include classical approaches such as the Bradley–Terry–Luce (BT) model Bradley and Terry [1952], RankCentrality (RC) Negahban et al. [2012], CrowdBT Chen et al. [2013], FactorBT Bugakova et al. [2019], and BARP Ferrara et al. [2024].

The implementations of RC, CrowdBT, and BARP are obtained from the publicly available repository released by Ferrara et al. [2024][1], while BT[2] and FactorBT[3] are sourced from their respective open-source repositories. These methods differ in how they model pairwise comparisons and annotator reliability. All methods were adapted to PyTorch and optimized for time efficiency.

E IMPLEMENTATION DETAILS OF BORAEM

We implement BoRaEM in PyTorch with full GPU support and deterministic seeding for reproducibility. Item rewards are initialized to zero, ensuring symmetry across items at the start of optimization. Like CrowdBT [Chen et al., 2013], we initialize worker competencies to a constant positive value (0.8), corresponding to a neutral reliability assumption. This avoids early instability while allowing competency differences to emerge purely from observed comparisons. Worker competencies are updated via closed-form aggregated statistics using vectorized accumulation. A Gaussian prior with fixed variance is incorporated as ridge regularization by adding prior precision to the denominator of each worker update. This shrinks poorly observed workers toward zero and prevents degeneracy. Within each EM iteration, reward and competency updates are alternated multiple times to accelerate convergence. After each outer iteration, identifiability constraints are enforced: rewards are centered to zero mean and normalized to unit root-mean-square magnitude, with worker competencies rescaled accordingly to preserve the likelihood. The competency prior variance σ_β^2 imposes a Gaussian prior on worker competencies β_w, which appears as an additional ridge term in the update:

$$\beta_s \leftarrow \frac{\sum_{(w,l)\in\mathcal{I}_s} \frac{1}{2}(r_w - r_l)}{\sum_{(w,l)\in\mathcal{I}_s} \kappa_{wl}(r_w - r_l)^2 + \sigma_\beta^{-2}},$$

where $\mathcal{I}_s$ is the set of comparisons performed by worker s, and κ_{wl} are the Polya-Gamma expectations defined in Equation (11). A smaller σ_β increases the prior precision σ_β^{-2}, shrinking β_w toward 0 (strong regularization), whereas a larger σ_β reduces the prior influence, allowing β_w to vary more freely. We set the default value of σ_β as 1.0.

The reward updates solve a linear system $(H + \lambda_r I)r = b$, where H depends on the worker competencies and Polya-Gamma expectations. The regularization coefficient λ_r is scaled with the mean of κ_{wl} to ensure numerical stability and prevent overfitting to sparse comparisons:

$$\lambda_r = \lambda_{r,\text{base}} \cdot \max(\bar{\kappa}, 10^{-12}),$$

where $\bar{\kappa} = \frac{1}{|\mathcal{D}|}\sum_{(w,l)\in\mathcal{D}} \kappa_{wl}$ is the mean Polya-Gamma expectation over all comparisons, and $\lambda_{r,\text{base}}$ is a small constant (default 10^{-2}). This acts as a data-dependent ridge, shrinking rewards slightly toward zero, improving convergence stability.

Convergence is monitored via the mean log-likelihood, and optimization terminates when its improvement falls below a fixed tolerance (10^{-6}).

Similar to BoRaEM, for HBTL, we set worker competencies to 0.8. Ablation studies are provided in Appendix I.

F SYNTHETIC DATASET EXPERIMENTS: $\beta \in [0, 1]$

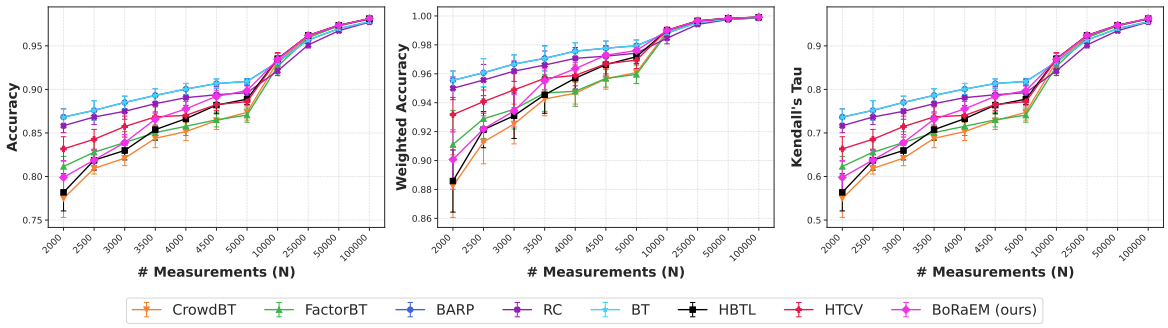
In this regime $\beta \in [0, 1]$, we do not consider malicious workers ($\beta \in [-1, 0]$). Figure 3 shows the results when we individually vary each parameter. The details are as follows:

1. **Varying N, Keeping K and M constant:** We conduct an experiment with fixed values of the number of items $K = 100$ and the number of workers $M = 500$, while varying the number of measurements $N \in \{2k, 2.5k, 3k, 3.5k, 4k, 4.5k, 5k, 10k, 25k, 50k, 100k\}$ to examine the effect of measurement volume on various techniques. As seen in Figure 3a, with less measurements the accuracy of methods such as BoRaEM, HBTL and CrowdBT is lower than other methods because more parameters need to be estimated from less number of measurements. Amongst, BoRaEM, HBTL and CrowdBT, we see that BoRaEM and HBTL consistently perform better than CrowdBT in sparser settimngs (e.g. till N = 10000).

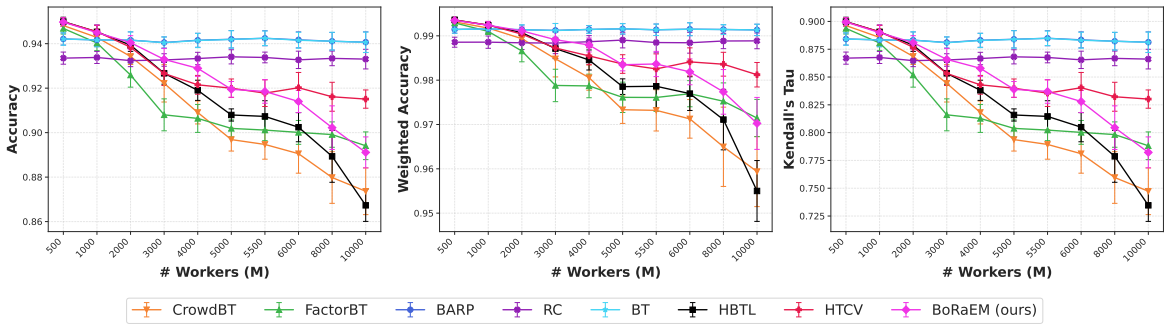
[1] https://github.com/Ambress92/Bias-Aware-Ranker-from-Pairwise-comparisons

[2] https://github.com/lucasmaystre/choix

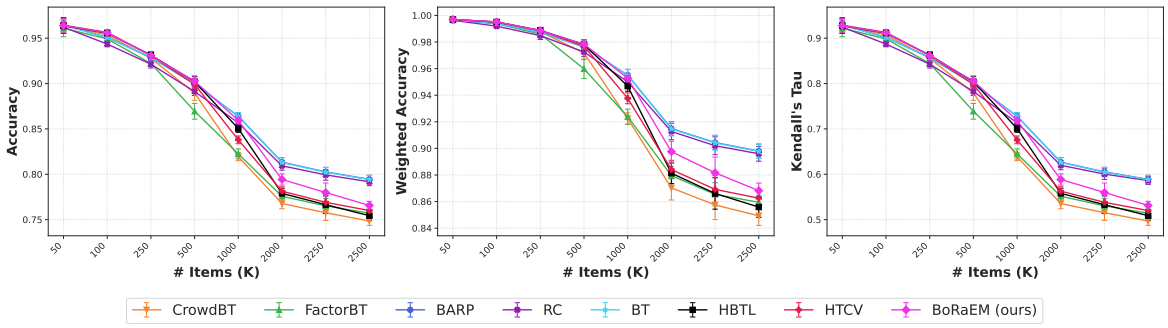
[3] https://github.com/Toloka/crowd-kit



(a) Varying N. $K = 100,\ M = 500$.



(b) Varying M. $K = 200,\ N = 25000$.



(c) Varying K. $M = 20000,\ N = 1000$.

Figure 3: Varying N, K, M individually when $\beta \in [0, 1]$.

2. **Varying M, Keeping N and K constant:** We vary the number of workers $M \in \{500, 1k, 2k, 3k, 4k, 5k, 5.5k, 6k, 8k\}$ while fixing the number of items at $K = 200$ and the number of measurements at $N = 25k$, to analyze the impact of worker scale. As seen in Figure 3b, HTCV seems to be the most stable alogrithm as compared to BoRaEM, HBTL and CrowdBT.

3. **Varying K, Keeping N and M constant:** We vary the number of items $K \in \{50, 100, 250, 500, 1000, 1250, 1500, 1750, 2250, 2500\}$ while fixing the number of workers at $M = 20k$ and the number of measurements at $N = 1k$, in order to study the effect of item scale. Similar to varying N and M, by varying K (Figure 3c), we observe the same trend where the performance of BoRaEM, HBTL, HTCV and CrowdBT get affected when number of item increases whereas BoRaEM and HTCV seem to be more robust than CrowdBT.

Although Figure 3 suggests that methods such as `BTL` and Rank Centrality (`RC`) are better than BoRaEM, HBTL, HTCV, FactorBT and CrowdBT in sparser settings for $\beta \in [0, 1]$, Section 6 (Figure 1) demonstrates how methods that do not model annotator competence can suffer significantly in the presence of adversaries. Section 6.4 further shows how these methods degrade with the addition of spammers and adversaries in real-world datasets.

G SPAMMER ADDITION IN FACEAGE DATASET: MORE DETAILS

In Section 6.4, we showed that results based on Kendall's tau demonstrate how methods such as `FactorBT`, `BT`, `BARP`, and `RC` degrade under spammer addition, whereas `BoRaEM`, `HBTL`, and `CrowdBT` remain robust due to their explicit modeling of annotator competence. Figures 4a and 4b present the results in terms of accuracy and weighted accuracy when various types of spammers are added to the FaceAge dataset. A similar performance pattern to that observed with Kendall's tau is evident for these metrics as well.

H SPAMMER ADDITION IN PASSAGE DATASET

In this section, we present the experimental results of spammer addition on the Passage dataset. Figures 5 and 6 shows the results using Kendall's tau as the evaluation metric. `CrowdBT`, `HBTL`, `HTCV`, `FactorBT` and `BoRaEM` exhibit robustness to spammers, whereas the remaining methods suffer substantially in their presence.

Similar trends are observed in Figures 6a and 6b, which report results in terms of accuracy and weighted accuracy, respectively. Models such as BT, FactorBT, Rank Centrality, BARP suffer heavily in presence of spammers and malicious adversaries. These findings further emphasize the importance of modeling annotator competence or susceptibility to bias in crowdsourced settings.

I ABLATION STUDY

I.1 INITIALIZATION OF WORKER COMPETENCIES

As discussed in Appendix E, we initialize all worker competency values to a fixed positive value (0.8) in case of BoRaEM and HBTL, corresponding to a neutral reliability assumption to avoid early instability and allowing competency differences to emerge purely from observed comparisons. Similarly, for CrowdBT, we follow Ferrara et al. [2024] and initialize worker competencies to 0.7.

In Section 2, we discussed about the objective function for FactorBT. For a comparison between items d_i and d_j by worker s, described by a feature vector $\mathbf{x}_{sij} \in \mathbb{R}^M$, where M is the feature dimension. The probability that d_i is preferred over d_j is given as follows

$$\Pr(d_i \succ_s d_j) = \sigma(\gamma_s) \cdot \sigma(r_i - r_j) + (1 - \sigma(\gamma_s)) \cdot \sigma(\langle \boldsymbol{b}_s, \mathbf{x}_{sij} \rangle) \tag{32}$$

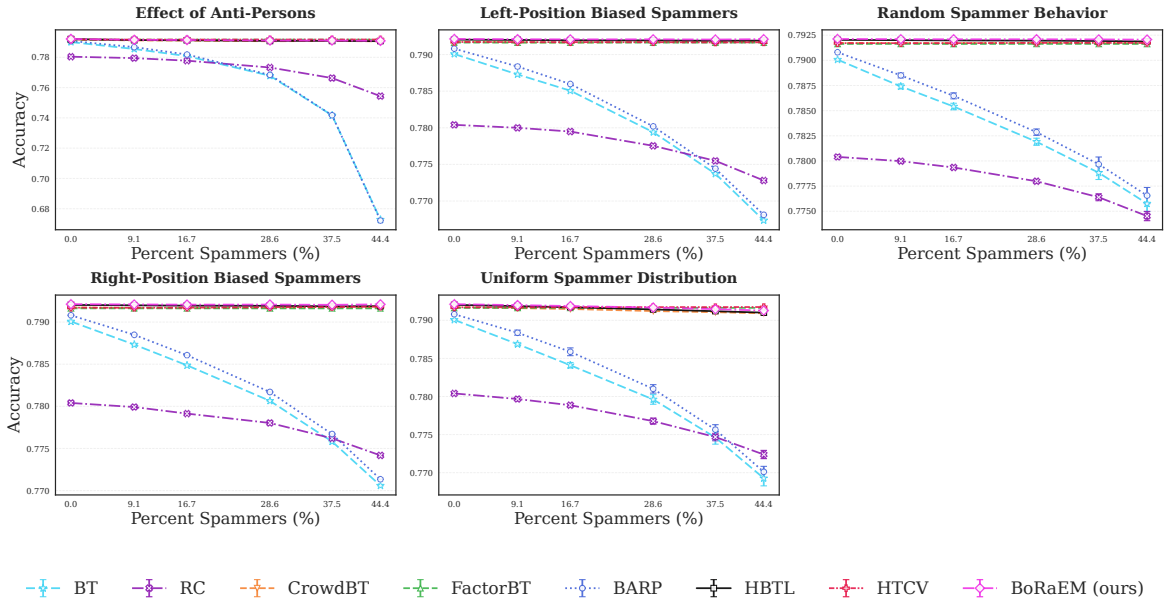
where r_i, r_j are the latent scores of items d_i and d_j, γ_s controls the susceptibility of worker s to bias and $\sigma(\gamma_s)$ can be considered as the the skill of the worker same as that of β_s in our formulation, $\boldsymbol{b}_s$ encodes the worker-specific sensitivity to task features, $\sigma(\cdot)$ is the sigmoid function. The model estimates parameters $\{r_i\}$, $\{\gamma_s\}$, and $\{\boldsymbol{b}_s\}$ by maximizing the regularized log-likelihood of the observed pairwise comparisons. For FactorBT, we uniformly initialize the worker logits (γ_s) to 0.7, which corresponds to initializing worker competencies as $\sigma(0.7) \approx 0.668$.

Effect of initialization. In this ablation study, we analyze the effect of initialization on the performance of BoRaEM, HBTL, and CrowdBT on the FaceAge dataset [Pavlichenko and Ustalov, 2021]. The initialization parameter represents annotator competency, denoted by β.

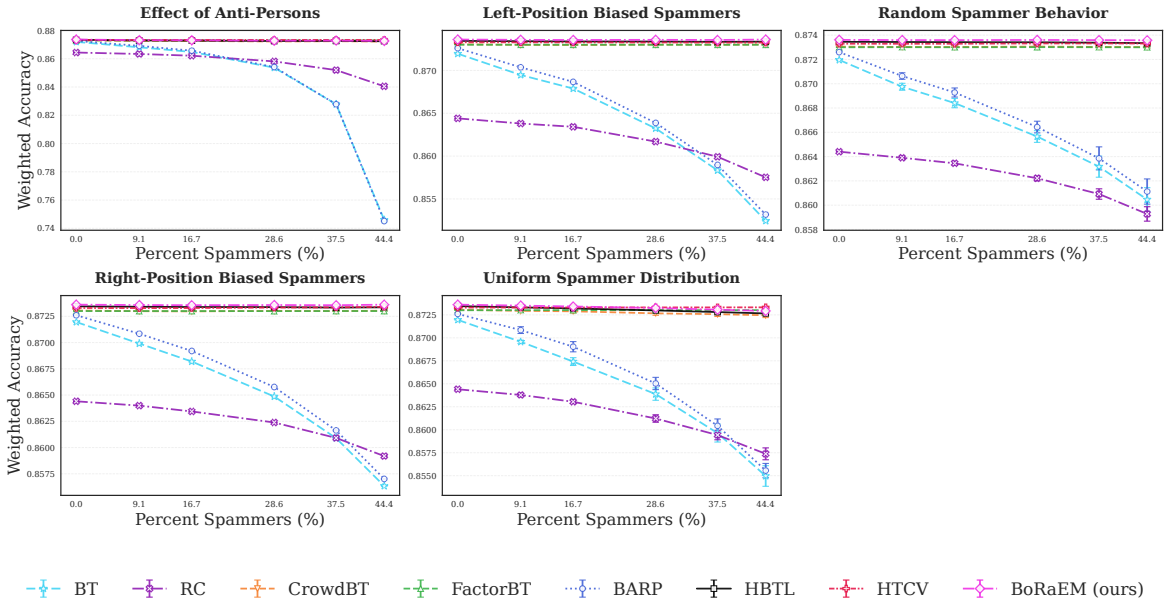
For BoRaEM and HBTL, we vary β from -1 to 1 in increments of 0.1. For CrowdBT, we vary β from 0 to 1 in increments of 0.1, consistent with its probabilistic interpretation.

Figures 7 and 8 present ablation studies on the effect of initialization for BoRaEM and HBTL, CrowdBT respectively.

Figure 9 presents an ablation study on the effect of initialization for FactorBT by varying the initial worker competency, $\beta_s = \sigma(\gamma_s)$. When β_s is initialized to 0, the component of $\beta_s \sigma(r_i - r_j)$ in Equation (32) is completely disabled, reducing the model to a pure worker-bias model that cannot learn meaningful item scores. Conversely, when β_s is initialized to 1, the worker-bias component, $\sigma(\langle \mathbf{b}_s, \mathbf{x}_{sij} \rangle)$, is effectively disabled at initialization, causing the model to behave as the standard Bradley–Terry model. Since $\gamma_s = \text{logit}(\beta_s)$, initializing β_s to 1 yields $\gamma_s \to \infty$, for which the skill gradients vanish because $\sigma'(\gamma_s) = \sigma(\gamma_s)(1 - \sigma(\gamma_s)) = 0$. For intermediate initializations, the model's performance remains stable, indicating that its performance is largely insensitive to the choice of initialization.

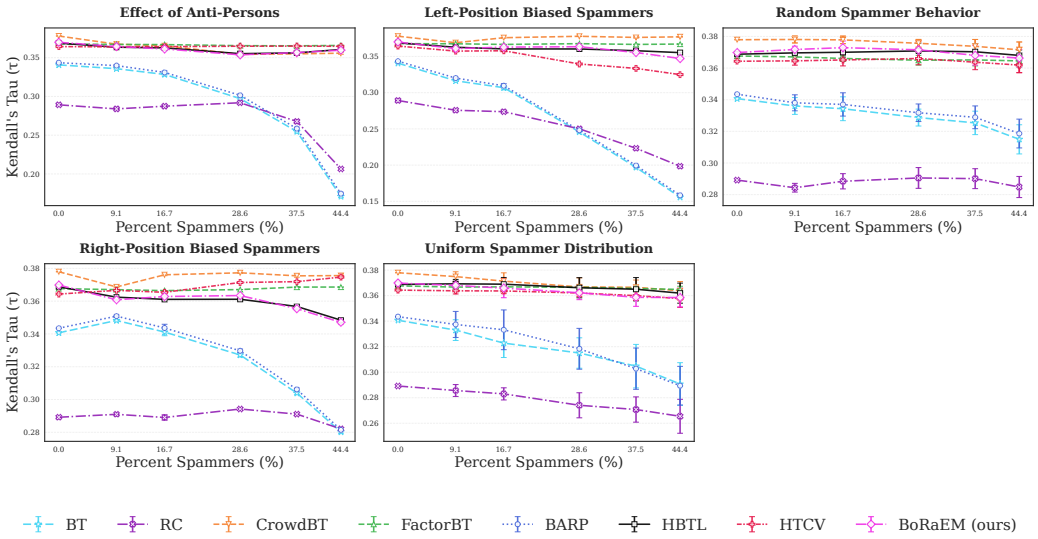


(a) Accuracy (acc)



(b) Weighted Accuracy ($wacc$)

Figure 4: Performance of different ranking methods under varying proportions of spammers introduced into the FACEAGE dataset.



(a) Kendall's tau (τ)

Figure 5: Performance of different ranking methods under varying proportions of spammers introduced into the PASSAGE dataset, measured using Kendall's tau (τ).

This study demonstrates the robustness of all methods to uniform initialization, where all workers are assigned the same initial competency value, with performance remaining largely stable across the initialization range.

I.2 SENSITIVITY TO MEAN LOG-LIKELIHOOD CONVERGENCE TOLERANCE

As described in Appendix E, we set the mean log-likelihood convergence tolerance parameter (ϵ) to 10^{-6}.

Figure 10 illustrates the sensitivity of BoRaEM, HBTL, HTCV, CrowdBT and FactorBT to different values of the convergence tolerance ϵ. The results show that both methods maintain stable performance across the range, with only minor gains in Accuracy and Kendall's Tau for stricter convergence, supporting the choice of $\epsilon = 10^{-6}$ as a robust default.

I.3 BORAEM-SPECIFIC ABLATION STUDIES

I.3.1 Sensitivity to Competency Prior Variance (σ_β)

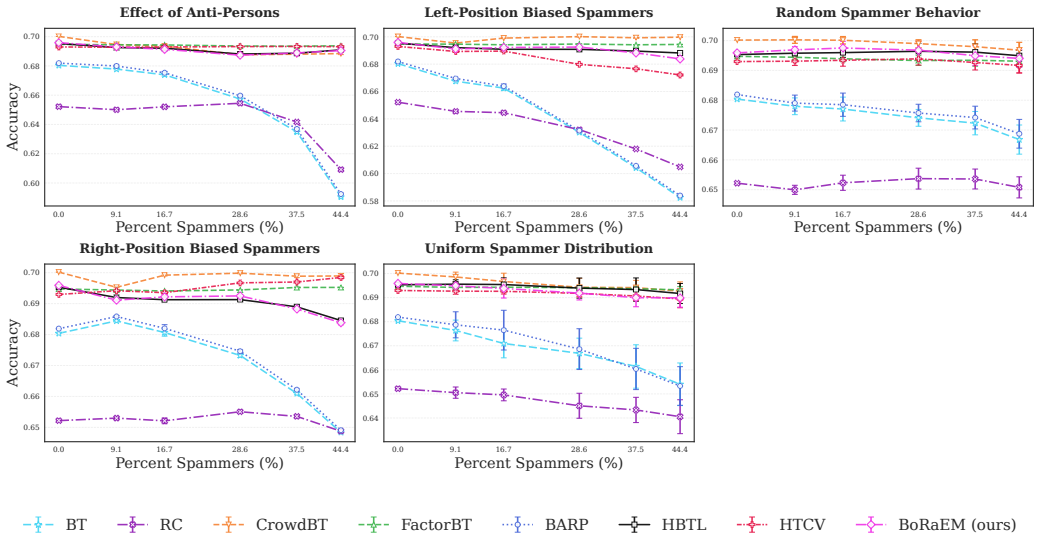
As described in Appendix E, we set the Competency Prior Variance parameter (σ_β) to 1.0.

The competency prior variance (σ_β^2) controls the strength of the Gaussian prior on worker competencies β_w in the EM updates. A smaller σ_β increases the prior precision σ_β^{-2}, which shrinks β_w toward 0 (strong regularization), while a larger σ_β reduces the prior influence, allowing β_w to vary more freely. In the extreme case of $\sigma_\beta \to \infty$, the prior is effectively removed, which can lead to overfitting and slightly lower stability in the estimates. The results above reflect this trade-off: moderate σ_β (e.g., 1.0) balances regularization and flexibility, producing optimal Accuracy and Kendall's Tau.

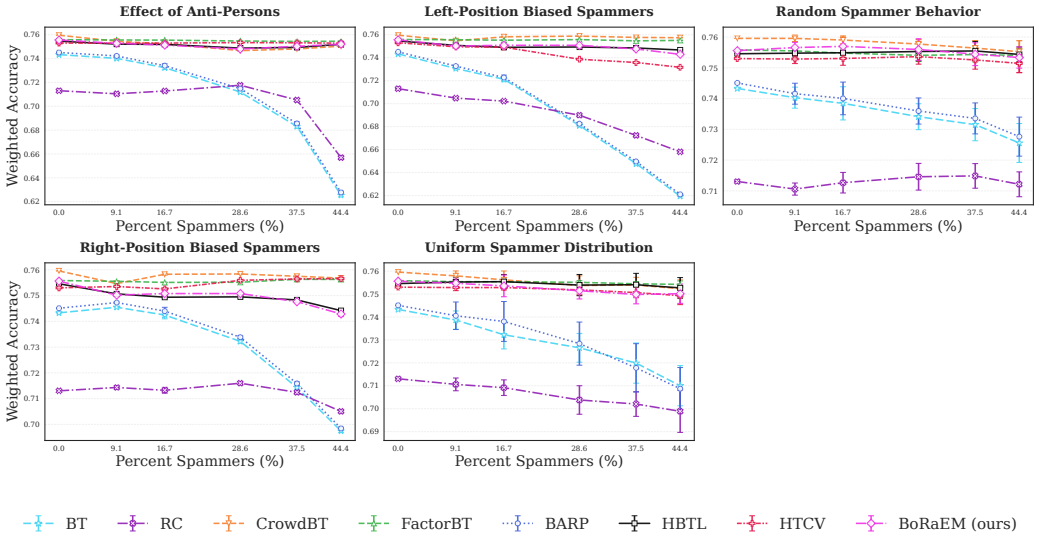
I.3.2 Sensitivity to Reward Regularization Coefficient ($\lambda_{r,\text{base}}$)

As described in Appendix E, we set the Reward Regularization Coefficient Parameter ($\lambda_{r,\text{base}}$) to 10^{-2}.

BoRaEM demonstrates remarkable robustness to the choice of $\lambda_{r,\text{base}}$ except at extreme values. This relative insensitivity suggests that while the regularization is essential for numerical conditioning in the linear system, the model's primary predictive power is driven by the underlying Pólya-Gamma latent variable framework rather than sensitive hyperparameter tuning.



(a) Accuracy (acc)



(b) Weighted Accuracy ($wacc$)

Figure 6: Performance of different ranking methods under varying proportions of spammers introduced into the PASSAGE dataset, measured using Accuracy (acc) and Weighted Accuracy ($wacc$).

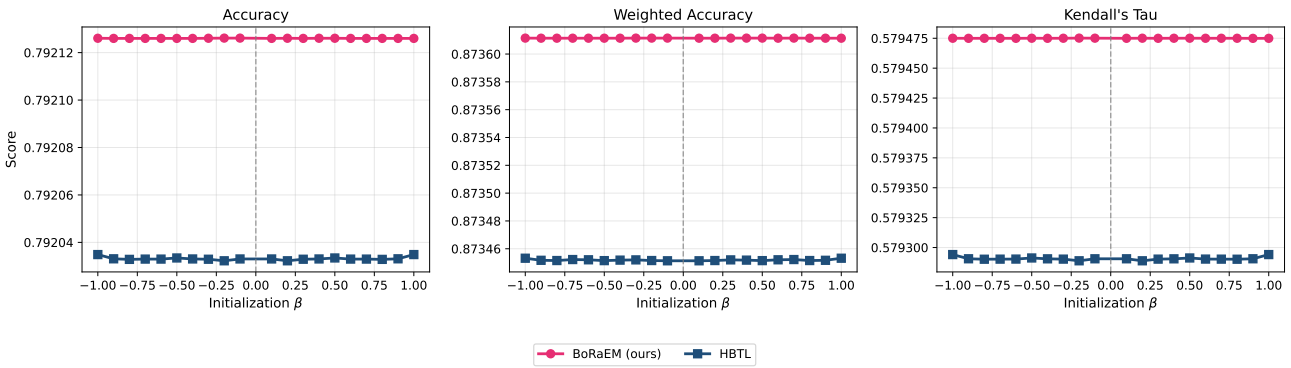


Figure 7: Effect of initialization on BoRaEM and HBTL. Both methods exhibit symmetry around $\beta = 0$, indicating that initializing with competency β or $-\beta$ leads to identical performance. This symmetry arises because their likelihood depends on the term $\sigma(\beta_s(r_w - r_l))$, which is invariant under simultaneous sign reversal of competency and score differences.

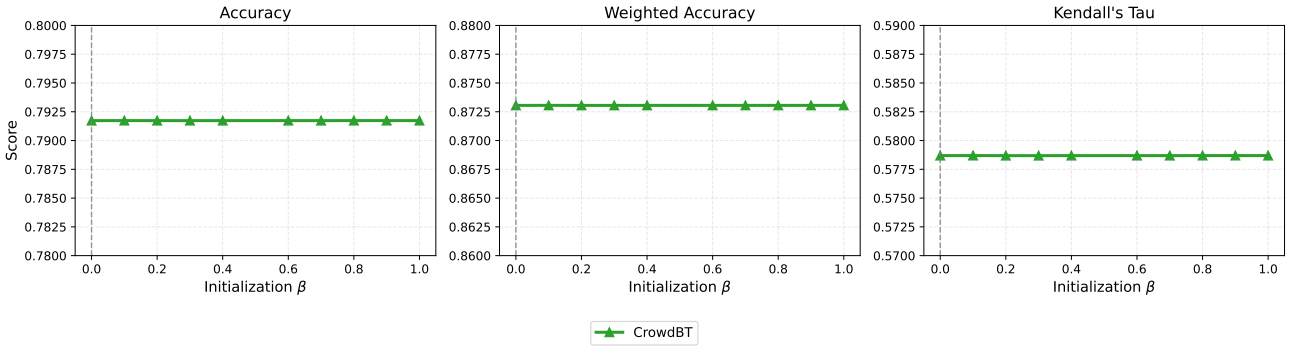


Figure 8: Effect of initialization on CrowdBT. The method exhibits symmetry around $\beta = 0.5$, meaning that initialization with β or $1-\beta$ produces identical results. This follows from the model structure, where β and $1-\beta$ represent complementary competency assumptions in the likelihood formulation.

J HARDWARE DETAILS

The experiments were conducted on a server with dual-socket Intel Xeon Gold 6348 CPUs (28 cores / 56 threads per socket, 2.6 GHz base, 3.5 GHz turbo, AVX-512), 503 GiB RAM, and 4 NVIDIA A100 GPUs. The L1, L2, and L3 caches are 2.6 MiB, 70 MiB, and 84 MiB, respectively, across 2 NUMA nodes. The system runs Ubuntu 22.04 LTS.

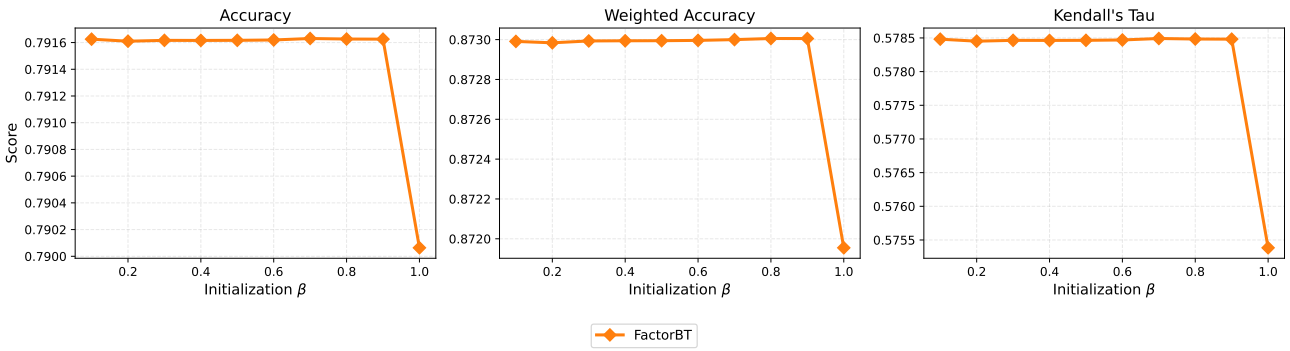


Figure 9: Effect of initialization for FactorBT by varying the initial worker competency, $\beta_s = \sigma(\gamma_s)$.

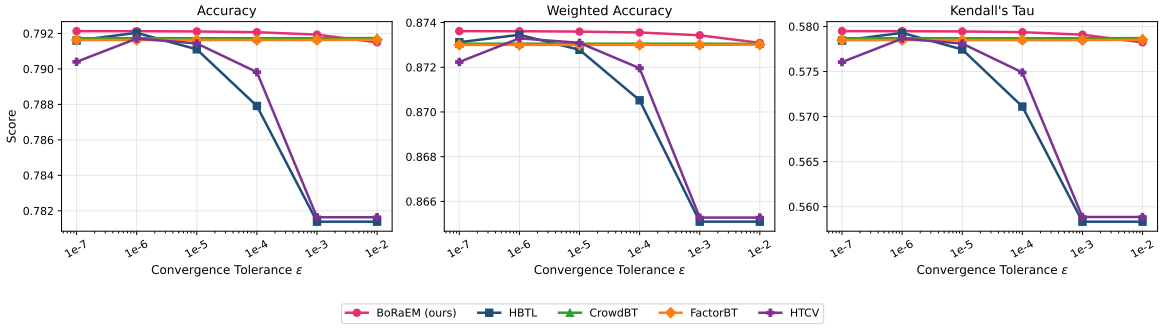


Figure 10: Impact of the mean log-likelihood convergence tolerance (ϵ) on the performance of BoRaEM, HBTL, HTCV, CrowdBT, and FactorBT. The x-axis is shown in decreasing order, from loose to strict convergence. Both methods remain largely stable across the tested range of ϵ, with minor improvements in Accuracy and Kendall's Tau observed for smaller values (stricter convergence). These results demonstrate that the EM updates converge reliably and are not overly sensitive to the choice of stopping criterion, confirming that the default setting of $\epsilon = 10^{-6}$ is within the optimal regime.

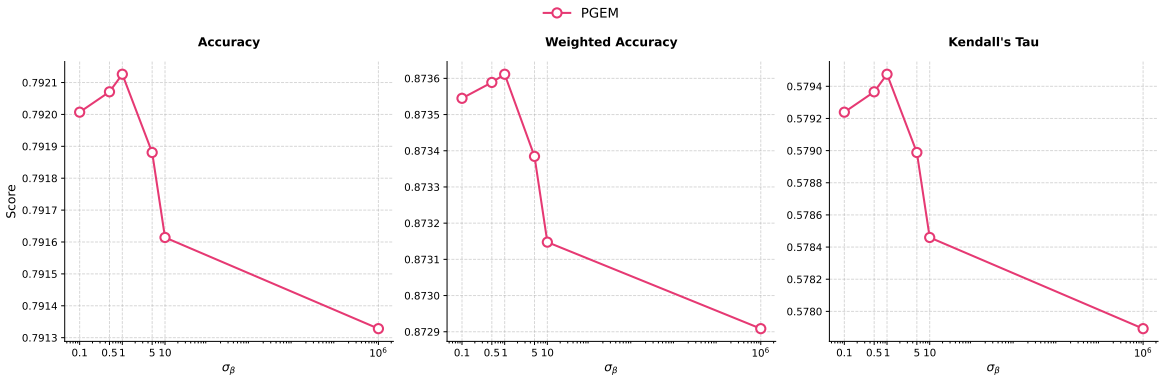


Figure 11: Effect of the competency prior variance (σ_β) on BoRaEM. This hyperparameter controls the strength of the Gaussian prior over worker competencies (β), where smaller values shrink competencies closer to zero and larger values allow more variability. The results show that both methods are largely stable across a wide range of σ_β, with minor improvements in Accuracy and Kendall's Tau when the prior is moderately loose. This indicates that the EM updates are robust to the choice of σ_β, and the default setting ($\sigma_\beta = 1.0$) provides a reasonable balance between regularization and flexibility.

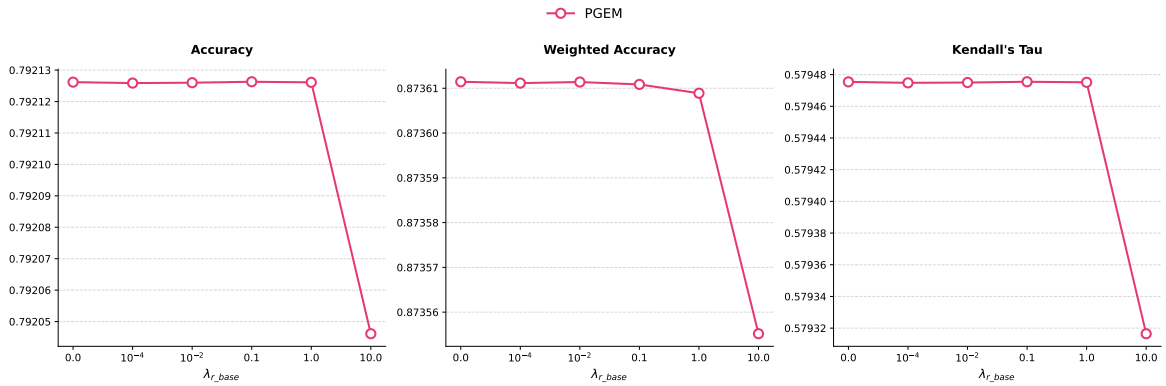


Figure 12: Effect of the reward regularization coefficient $\lambda_{r,\text{base}}$ on BoRaEM performance.