CURATRON: Complete and Robust Preference Data for Rigorous Alignment of Large Language Models

Son The Nguyen
University of Illinois Chicago
Chicago, Illinois, USA
snguye65@uic.edu

Niranjan Uma Naresh
Independent
USA
un.niranjan@gmail.com

Theja Tulabandhula
University of Illinois Chicago
Chicago, Illinois, USA
theja@uic.edu

ABSTRACT

This paper addresses the challenges of aligning large language models (LLMs) with human values via preference learning (PL), focusing on incomplete and corrupted data in preference datasets. We propose a novel method for robustly and completely recalibrating values within these datasets to enhance LLMs' resilience against the issues. In particular, we devise a guaranteed polynomial time ranking algorithm that robustifies several existing models, such as the classic Bradley–Terry–Luce (BTL) [5] model and certain generalizations of it. To the best of our knowledge, our present work is the first to propose an algorithm that provably recovers an ϵ-optimal ranking with high probability while allowing as large as $O(n)$ perturbed pairwise comparison results per model response. Furthermore, we show robust recovery results in the partially observed setting. Our experiments confirm that our algorithms handle adversarial noise and unobserved comparisons well in both general and LLM preference dataset settings. This work contributes to the development and scaling of more reliable and ethically aligned AI models by equipping the dataset curation pipeline with the ability to handle missing and maliciously manipulated inputs.

1 INTRODUCTION

Large Language Models (LLMs) are highly advanced Artificial Intelligence (AI) systems capable of understanding, interpreting, and generating languages. The integration of AI chatbots like ChatGPT into our daily lives and businesses has had a profound impact on both society and industries [12]. These models have evolved from being specialized tools in specific fields to versatile assets that are increasingly applied in everyday activities and diverse work environments [31]. However, the success of GPTs/LLMs depends not only on their ability to generate responses and perform tasks well but also on their alignment with human values and expectations.

The prevalent method for aligning AI/LLMs currently involves preference learning (PL) through RLHF or Reinforcement Learning from AI Feedback (RLAIF) [26] using Proximal Policy Optimization (PPO) [38], or alternatively, employing Direct Preference Optimization (DPO) [35]. While PPO is a reinforcement learning (RL) technique within the RLHF pipeline, DPO directly integrates human preferences into the LLMs.

These techniques rely on collecting and curating high-quality pairwise human preference data, which presents several challenges. Gathering human feedback is slow and expensive and often results in incomplete or imperfect data [4, 26]. Furthermore, participants may intentionally provide inaccurate or harmful feedback due to malicious intentions, as pointed out by [6]. These factors can lead to unintended consequences in estimating rankings from preference datasets from models such as BTL. They pose a considerable challenge in ensuring the integrity and reliability of the preference datasets used for aligning LLMs, especially when scaling up the alignment process with large-scale responses and participants.

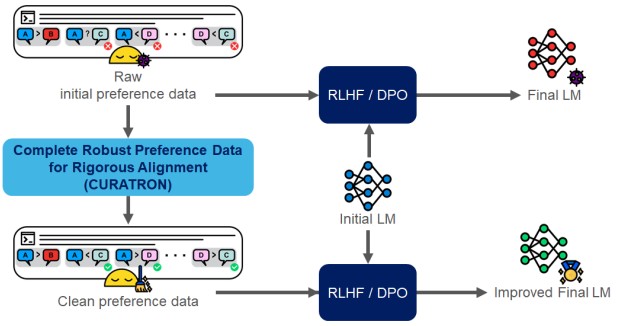

Figure 1: CURATRON corrects incomplete and adversarially corrupted preference data to improve RLHF/DPO alignment results compared to using the raw initial preference data.

Approaching the issues, we consider the following learning problem. Suppose there are n responses we wish to order based on a notion of comparison, between every pair of responses, with probabilistic outcomes. Further, we are given a set, $\aleph = \{(i, j, \{y_{ij}^k\})\}$, consisting of K independent pairwise comparison outcomes, denoted by $\{y_{ij}^k\} \in \{0, 1\}, k \in [K]$, between pairs of responses $(i, j) \subseteq [n] \times [n]$, a significant proportion of which might be corrupted by an adversary. In this passive learning setting, the concrete questions we wish to address are:

(1) Is it possible to identify the pairs whose comparison results were corrupted by an adversary?
(2) Having identified the corrupted results, as desired, is it possible to filter them out while computing a global ranking of the n responses?
(3) Is this task tractable statistically and computationally?
(4) If so, is it possible to construct a provably correct and efficient algorithm, and what are the associated properties?
(5) Further, does it work well in practice when we may also encounter unobserved data?

Our Contributions: We systematically answer the above questions in the affirmative. Specifically, our contributions are as follows.

(1) *Problem formulation:* We give a generic definition of (additive) adversarial noise, which can be handled for a broad class of statistical models, including the classic BTL model and also certain extensions of it such as the general Low–Rank (LR) models [37]. As is the case with standard estimation techniques, if the noise is not modeled and handled well, we

show that the quality of the estimated ranking could be quite bad, by quantifying the error of the estimated ranking with respect to the best possible ranking.

(2) *Algorithms & guarantees:*
- Under certain (information-theoretically tight) identifiability assumptions on the properties of the adversary, we develop a correct and efficient ranking method, Robust Preference Data for Rigorous Alignment (RORATRON), that guarantees ϵ-accurate high-probability learnability in a manner that is 'robust' and oblivious to the effects of the adversary. Our learning algorithm is provably characterized by polynomial time computational complexity.
- In practice, it is often the case that not all pairs are compared, and even the observed pairwise comparison data could be adversarially corrupted – we also develop Complete Robust Preference Data for Rigorous Alignment (CURATRON) and characterize the conditions for guaranteed robust recovery in this scenario. This results in a practical implication of enhancing preference data collection efficiency by automatically generating complete datasets from limited missing preference data.

(3) *Experiments:* Finally, we support our theoretical results by showing robust ranking results on both synthetic and real-world experiments. Our experiments demonstrate the potential of our method in helping create large-scale AI/LLMs that are more accurately aligned with human values using minimal human effort as we achieved high reconstruction accuracy despite severe data missing and corruption.

2 RELATED WORK

We now briefly present relevant work in: (1) LLM alignment with PL from human feedback, (2) ranking models and ranking algorithms that handle noise, and (3) robust subspace recovery methods, which will be needed for us to prove recovery results for ranking.

LLM Alignment with PL from human feedback: PL was initially developed to train agents in simulated environments to perform nuanced behaviors that are hard to define but easy to observe and recognize [8]. It has recently been found successful in aligning LLMs to human intentions and values such as harmfulness, helpfulness, factuality, and safety. Some of the methods of PL in LLMs are RLHF [34], RLAIF [4, 26], DPO/ψPO [35, 43, 47], and SLiC-HF [47]. However, these methods assume that there is high-quality human supervision through pairwise/ranking preference data, but in practice, this is often not the case [6]. Recent works such as KTO [13] also attempt to eliminate the need for pairwise human preferences, requiring only binary feedback on LLM outputs.

Ranking Models: In the BTL model, item i has an associated score w_i; then, the probability that item i is preferred over j is given by $P_{ij} = e^{-w_i}/(e^{-w_i} + e^{-w_j})$ where $\mathbf{w} \in \mathbb{R}^n$ is the BTL parameter vector to be estimated from data; here, $\mathbf{P} \in \mathbb{R}^{n \times n}$ is called the 'preference matrix'. A closely related model, in the non-active setting, is the recently proposed LR model [37] wherein a generic class of preference matrices is characterized to be those having low rank under transformations using certain functions; specifically, for BTL-like models, the logit function defined as $\psi(x) = \log(x/(1-x))$ turns out to right choice as shown in their paper. However, while their model accounts for missing information, they do not consider the harder problem of handling adversarial noise. Several robust ranking heuristics have been proposed (for example, [45, 49]) but these approaches do not have theoretical guarantees associated with them. The Sync-Rank algorithm, for handling different noise models as compared to the one considered in the preset work, was proposed in [11] and is based on spectral techniques. Another related work is [36] which proposes the so-called 'Generalized Low-Noise' (GLN) condition that $\forall i \neq j, P_{ij} > P_{ji} \implies \sum_{h=1}^{n} \alpha_h P_{hj} > \sum_{h=1}^{n} \alpha_h P_{hi}$ for $\alpha \in \mathbb{R}^n$. When $\alpha_h = 1, \forall h$ they analyze the sample complexity and show convergence properties of various popular ranking algorithms like:

(1) Maximum Likelihood (ML): this entails solving $\arg\max_{\mathbf{w}} \sum_{i<j} (\widehat{P}_{ij}(w_j - w_i) - \log(1+\exp(w_j - w_i)))$ where $\mathbf{w} \in \mathbb{R}^n$ is the BTL parameter vector and $\widehat{P}_{ij}$ is the empirical preference matrix.

(2) Rank Centrality (RC) [28]: here, one sorts items by their scores which are computed as the stationary distribution of an appropriately normalized empirical preference matrix; this approach has a known sample complexity guarantee of $O(n \log(n))$.

(3) Borda Count (BC) [23]: this heuristic involves ranking an item according to the fraction of times it beats other items.

For the general case α (which previous methods fail to handle), they also propose a noise-tolerant SVM-based method for rank aggregation. However, in the adversarial setting, we consider in this paper, GLN could be violated and hence requires a different algorithmic approach and analysis.

Robust Subspace Recovery: It is well-known that Principal Component Analysis (PCA), a ubiquitous technique for subspace identification, is not robust to outliers; this may be attributed to the fact that PCA is an L_2 optimization problem due to which grossly corrupted data points may perturb and skew the eigenvectors spanning the maximum variance subspace of the data points significantly.

The Robust PCA (RPCA) problem [30] addresses the following question: suppose we are given a data matrix $\mathbf{M}$ which is the sum of an unknown low-rank matrix $\mathbf{L}$ and an unknown sparse matrix $\mathbf{S}$, can we recover each of the component matrices? While several works [19, 46] analyze this problem, it is shown in [30] that, under information-theoretically tight assumptions, a simple iterative algorithm based on non-convex alternating projections of appropriate residuals provably yields an ϵ-accurate solution in $O(\log(1/\epsilon))$ iterations with an overall computational complexity of $O(n^2 r^2 \log(1/\epsilon))$ where r is the rank of $\mathbf{L}$. We will use this result, in particular, to derive guarantees for our ranking problem.

3 PROBLEM SETUP AND SOLUTION APPROACH

3.1 Notation

We first define some notation. We denote the set of all permutations of n LLM responses/items as $\mathcal{S}_n$. If not specifically defined, we use lower-case letters for scalars, upper-case letters for global constants,

lower-case bold-face letters for vectors and upper-case bold-face letters for matrices; specifically, $\mathbf{P}$ denotes a preference matrix. Let $\mathcal{P}_n := \{\mathbf{P} \in [0,1]^{n\times n} | P_{ij} + P_{ji} = 1\}$ denote the set of all pairwise preference matrices over n responses. Let the set of stochastic-transitive matrices be $\mathcal{P}_n^{ST} := \{\mathbf{P} \in \mathcal{P}_n | P_{ij} > 1/2, P_{jk} > 1/2 \implies P_{ik} > 1/2\}$. Let the set preference matrices described by the BTL model be $\mathcal{P}_n^{BTL} := \{\mathbf{P} \in \mathcal{P}_n | \exists \mathbf{w} \in \mathbb{R}^n \text{ s.t. } e^{-w_i}/(e^{-w_i}+e^{-w_j})\}$. Let $\psi : [0,1] \mapsto \mathbb{R}$ be a strictly increasing bijective L-Lipschitz function and define the class of low-rank preference matrices with respect to ψ as $\mathcal{P}_n^{LR(\psi,r)} = \{\mathbf{P} \in \mathcal{P}_n | \text{rank}(\psi(\mathbf{P})) \le r\}$ where $r \in [n]$; when we apply such a transformation to a matrix, it is applied entry-wise. In this paper, we take ψ to be the logit function.

For any matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$, let the infinity norm be denoted by $\|\mathbf{M}\|_\infty = \max_{i,j} |M_{ij}|$, the Frobenius norm be denoted by $\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n M_{ij}^2}$, the spectral norm be denoted by $\|\mathbf{M}\|_2 = \max_{\mathbf{x},\mathbf{y}\in\mathbb{R}^n} \mathbf{x}^\top \mathbf{M}\mathbf{y}$. Denoting the indicator function by $\mathbb{1}$, define the zero norm of a matrix to be the maximum number of non-zero elements in any row/column, ie, $\|\mathbf{M}\|_0 = \max(\max_j \sum_{i=1}^n \mathbb{1}(M_{ij} \neq 0), \max_i \sum_{j=1}^n \mathbb{1}(M_{ij} \neq 0))$. Let the Singular Value Decomposition (SVD) of a square matrix be given by $\mathbf{M} = \mathbf{U}\Sigma\mathbf{V}^\top$ where $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n\times r}$ are orthonormal matrices (whose columns are singular vectors) and $\Sigma \in \mathbb{R}^{r\times r}$ is the diagonal matrix of singular values. Now, $\mathbf{M}$ is said to be μ-incoherent if $\max\left(\max_i \left\|\mathbf{e}_i^\top \mathbf{U}\right\|_2, \max_i \left\|\mathbf{e}_i^\top \mathbf{V}\right\|_2\right) \le \mu\sqrt{r/n}$ where $\mathbf{e}_i$ denotes the i^{th} basis vector in $\mathbb{R}^n$. Also, let $\sigma_{\max} := \max_i \Sigma_{ii}$ and $\sigma_{\min} := \min_i \Sigma_{ii}$.

We define the distance between a permutation $\sigma \in \mathcal{S}_n$ and a preference matrix $\mathbf{P} \in \mathcal{P}_n$ as:

$$\text{dist}(\sigma, \mathbf{P}) := \binom{n}{2}^{-1} \sum_{i<j} \mathbb{1}\left((P_{ij} > 1/2) \wedge (\sigma(i) > \sigma(j))\right) + \binom{n}{2}^{-1} \sum_{i<j} \mathbb{1}\left((P_{ji} > 1/2) \wedge (\sigma(j) > \sigma(i))\right)$$

Note that the above loss function basically is the number of pairs on which the ordering with respect σ and $\mathbf{P}$ differ divided by the number of ways to choose two out of n responses. Finally, let $P_{\min} = \min_{i\neq j} P_{ij}$ and $\Delta = \min_{i\neq j} |\psi(P_{ij}) - \psi(1/2)|$.

3.2 Characterization of the Adversary

The following (weak) assumption characterizes the properties of the adversary. We shall see in the next section that it is information-theoretically tight in order to guarantee recovery in the solution approach that we propose. Note that this is a deterministic assumption; in particular, we do not have any distributional assumptions regarding the locations, the signs, or the magnitudes of the corruptions, and hence is very general.

Assumption 1. *The (additive) adversarial noise which corrupts a μ-incoherent preference matrix $\mathbf{P} \in \mathcal{P}_n^{LR(\psi,r)}$ is modeled by a skew-symmetric sparse matrix $\mathbf{S}$ so that the corrupted preference matrix $\mathbf{P^c} \in \mathcal{P}_n$ is given by $\mathbf{P^c} = \mathbf{P} + \mathbf{S}$. We assume the (deterministic) bounded degree condition that $\|\mathbf{S}\|_0 \le d < n$ such $d < n/512\mu^2 r$ where $r \le n$.*

So, why do existing non-robust algorithms not recover the true response ordering in the presence of an adversarial noise source? This question is answered by the following proposition which precisely quantifies how bad a ranking could be when an algorithm uses the corrupted pairwise preference matrix. The key idea is to construct an adversary that intentionally flips true comparison results.

Procedure 1 RPCA: Robust Principal Component Analysis

Input: $\mathbf{M} = \mathbf{L}^* + \mathbf{S}^*$, rank r of $\mathbf{L}^*$.
Output: $\widehat{\mathbf{L}}, \widehat{\mathbf{S}}$.

1: Solve the following optimization problem using Algorithm 1 of [30]:

$$\{\widehat{\mathbf{L}}, \widehat{\mathbf{S}}\} = \arg\min_{\mathbf{L},\mathbf{S}} \|\mathbf{M} - \mathbf{L} - \mathbf{S}\|_F \quad \text{s.t. } \text{rank}(\mathbf{L}) \le r, \|\mathbf{S}\|_0 \le d$$

2: **return** $\widehat{\mathbf{L}}, \widehat{\mathbf{S}}$.

Procedure 2 PR: (γ-approximate) Pairwise Ranking

Input: Preference matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$.
Output: Ranking $\widehat{\sigma}$.

1: Compute $\forall i, \quad v_i \leftarrow \sum_{j=1}^n \mathbb{1}(M_{ij} > 1/2)$.
2: **return** $\widehat{\sigma} \leftarrow \text{Sort}(\mathbf{v})$.

Claim 1 (**Upper bound on estimation error**). *Under Assumption 1 it is possible that* $\text{dist}(\widehat{\sigma}, \mathbf{P^c}) = O(1)$.

Proof. Assume that we are exactly given the entries of the preference matrix as opposed to sampling them. Note that in order to estimate a ranking from a given preference matrix, we still need to use a pairwise ranking procedure. Let $\widehat{\sigma} \in \mathcal{S}_n$ be the output of any Pairwise Ranking (PR) procedure with respect to an underlying preference matrix $\mathbf{Q} \in \mathcal{P}_n$. For a constant $\gamma > 1$, $\widehat{\sigma}$ is said to be γ-approximate if $\text{dist}(\widehat{\sigma}, \mathbf{Q}) \le \gamma \min_{\sigma\in\mathcal{S}_n} \text{dist}(\sigma, \mathbf{Q})$. Define the following distance which measures the fraction of response pairs over which two preference matrices $\{\mathbf{Q}, \mathbf{R}\} \in \mathcal{P}_n$ disagree.

$$\text{dist}(\mathbf{Q}, \mathbf{R}) := \binom{n}{2}^{-1} \sum_{i<j} \mathbb{1}\left((Q_{ij} > 1/2) \wedge (R_{ij} < 1/2)\right) + \binom{n}{2}^{-1} \sum_{i<j} \mathbb{1}\left((Q_{ij} < 1/2) \wedge (R_{ij} > 1/2)\right)$$

By Lemma 20 of [37], for $\mathbf{Q} \in \mathcal{P}_n^{ST}$ and $\mathbf{R} \in \mathcal{P}_n$, we have $\text{dist}(\widehat{\sigma}, \mathbf{Q}) \le (1+\gamma)\,\text{dist}(\mathbf{Q}, \mathbf{R})$. But note that it is possible that $\text{dist}(\mathbf{Q}, \mathbf{R}) = 1$ as it is easy to construct by $\mathbf{R}$ that disagrees with $\mathbf{Q}$ in every entry by simply setting $\mathbf{R} = \mathbf{Q}^\top$. Now, we may set $\mathbf{Q} = \mathbf{P}$ and $\mathbf{R} = \mathbf{P^c}$ for any algorithm that uses $\mathbf{P^c}$ for ranking; specifically, for the adversary satisfying Assumption 1, we can see by a direct counting argument that $\text{dist}(\mathbf{Q}, \mathbf{R}) \le \frac{d(2n-1-d)}{n(n-1)}$ which proves the claim. □

3.3 Solution Approach

This part of the paper identifies three scenarios/settings where missing and adversarially corrupted comparisons can affect the ranking results. We plan to tackle the three situations detailed in subsequent sections:

(1) *Fully observed and adversarially corrupted setting:* Some instances of the comparison results are adversarially corrupted. This scenario can happen when data quantity is prioritized over data quality in the data collection process, resulting in biased or malicious human feedback. Algorithm 3 `RORATRON` is proposed to solve this problem.
(2) *Partially observed and uncorrupted setting:* Not all pairs of responses are compared. This scenario can happen when data quality is prioritized over quantity in the data collection process. Observing all possible comparisons can be expensive and challenging, especially when there are many LLM responses to compare. Algorithm 4 `CORATRON` is proposed to solve this problem.
(3) *Partially observed and adversarially corrupted setting:* Both (1) not all pairs of responses are compared, and (2) some instances of the comparison results are adversarially corrupted. This scenario can happen when data quantity and quality are not met in the data collection process. This scenario will likely happen in a large crowd-sourced environment due to large-scale LLM responses and participants. Algorithm 5 `CURATRON` is proposed to solve this problem.

4 FULLY OBSERVED ADVERSARIAL SETTING

4.1 Algorithm

In this section, we answer Question 4. We present our main algorithm for robust passive ranking from pairwise comparisons in the presence of adversarial noise in Algorithm 3. The input data consist of the set of pairwise comparison results $\aleph = \{(i, j, \{y_{ij}^k\})\}$, $(i, j) \in [n] \times [n]$, $k \in [K]$, $y_{ij}^k \in \{0, 1\}$. The algorithm assumes the true rank of $\psi(\mathbf{P})$ as an input parameter; specifically, for the BTL model, we set $r = 2$. Algorithm 3 calls the following procedures:

(1) *Robust PCA (Procedure 1):* Note that Step 3 of Algorithm 3 uses a matrix low-rank plus sparse decomposition subroutine. To obtain our recovery guarantee, it is sufficient to use the robust PCA problem as a black-box method; for the precise details of this algorithm, we refer the reader to [30]. In particular, for our analysis, we use the noise-case guarantees in their paper. This is characterized by a (strongly-polynomial) running time of $O(n^2r^2\log(1/\epsilon))$ and guarantees ϵ-recovery of the component matrices under the conditions of Assumption 1 and Lemma 3.
(2) *γ-approximate pairwise ranking procedure (Subroutine 2):* Step 4 of Algorithm 3 calls a constant factor approximate ranking procedure. Specifically, we use the Copeland procedure [9] which has a 5-approximation guarantee [10] and involves sorting the responses according to a score of response i given by $\sum_{j=1}^{n} \mathbb{1}(\overline{P}_{ij} > 1/2)$.

4.2 Analysis

We begin with a useful short result followed by the statement and the proof of our main result that, with high probability, we achieve ϵ–accurate ranking in polynomial time using polynomial number of samples, despite the presence of adversarial noise. Precisely, Theorem 1 and Remark 1 address Question 3; Remark 2 addresses Question 1. In this context it is noteworthy that we present the result for LR models which strictly contain the BTL model while being much more general [37]; upon proving this result, we specialize it to the classic BTL model as well (Corollary 1).

Algorithm 3 RORATRON: Robust Preference Data for Rigorous Alignment

Input: Comparison dataset $\aleph = \{(i, j, \{y_{ij}^k\})\}$, true rank r.
Output: Ranking of n responses, $\hat{\sigma} \in \mathcal{S}_n$.
1: Estimate entries of $\widehat{\mathbf{P}}$ for $i \le j$ as:

$$\widehat{P}_{ij} = \begin{cases} \frac{1}{K}\sum_{k=1}^{K} y_{ij}^k & \text{if } i < j \\ 1/2 & \text{if } i = j \end{cases}$$

2: Set $\widehat{P}_{ij} = 1 - \widehat{P}_{ji}$ for all $i > j$.
3: Perform robust PCA: $\{\psi(\overline{\mathbf{P}}), \widehat{\mathbf{S}}\} \leftarrow \text{RPCA}(\psi(\widehat{\mathbf{P}}), r)$.
4: Using a pairwise ranking procedure after taking the inverse transform: $\widehat{\sigma} \leftarrow \text{PR}(\overline{\mathbf{P}})$.
5: **return** $\widehat{\sigma}$.

Lemma 1 (**Some properties of the logit function**). *Let $a, b, c \in (0, 1)$ such that $c = a + b$. Then, we have,*
(1) $\psi(c) = \psi(a) + \psi(a + b) + \psi(1 - a)$
(2) $\psi(a) + \psi(1 - a) = 0$.

Proof. Both follow by using the definition of the logit function that $\psi(a) = \log(a/(1 - a))$ and using the property that $\log(ab) = \log(a) + \log(b)$. □

Theorem 1 (**Provably good estimation of ranking in LR models in the presence of adversarial noise**). *Let $\mathbf{P} \in \mathcal{P}_n^{LR(\psi,r)}$ be the true preference matrix according to which the pairwise comparison dataset $\aleph = \{(i, j, \{y_{ij}^k\})\}$ is generated for all responses pairs (i, j) such that $k \in [K]$. Let $\widehat{\mathbf{P}}$ be the empirical preference matrix computed using $\aleph$. Let $\mathbf{S} \in [0, 1]^{n\times n}$ be the adversarial matrix that additively corrupts $\widehat{\mathbf{P}}$. Let ψ be L-Lipschitz in $[\frac{P_{\min}}{2}, 1 - \frac{P_{\min}}{2}]$ and $\psi(\mathbf{P})$ be μ-incoherent. Let each pair be compared independently $K \ge 16384\mu^2(1+\gamma)L^2n^2\log^2(n)/\epsilon\Delta^2$ times where $\Delta = \min_{i\neq j}\left|\psi(P_{ij}) - \psi(1/2)\right|$. Then, with probability atleast $1 - 1/n^3$, Algorithm 3 returns an estimated permutation $\widehat{\sigma}$ such that $\text{dist}(\widehat{\sigma}, \mathbf{P}) \le \epsilon$.*

Remark 1 (**Computational complexity**). *In Algorithm 3, Step 1 takes $O(n^2K) = O(n^4\log^2 n/\epsilon)$ time, Step 3 takes $O(n^2r^2\log(1/\epsilon))$, and Step 4 takes $O(n^2+n\log n)$ time. Thus, putting together the cost of these main steps, the overall computational complexity of our robust ranking algorithm for $\mathbf{P} \in \mathcal{P}_n^{LR(\psi,r)}$ is $O(n^4\log^2 n/\epsilon)$.*

Remark 2 (**Identifying adversarially corrupted pairwise comparisons**). *From Step 3 of Algorithm 3, using Theorem 2 of [30], we also have $\text{Supp}(\widehat{\mathbf{S}}) \subseteq \text{Supp}(\mathbf{S})$ and thus we can identify the corrupted pairwise comparison results.*

Remark 3 (**Missing data versus adversarially corrupted data**). *Note that the adversarial sparse noise we consider subsumes the setting when comparison results for certain pairs are missing as in [37] and hence directly applies in that situation. Moreover,*

since the support and magnitude of the corrupted entries of the preference matrix are unknown, the problem considered in this paper is harder; consequently, our sample complexity is $O(n^2)$ as opposed to $O(n \operatorname{poly}\log n)$ in their work.

Proof. Let $\widetilde{P}_{ij}$ be the empirical probability estimate of P_{ij}. Note that we compute $\widetilde{P}_{ij} = \frac{1}{K}\sum_{k=1}^{K} y_{ij}^k$ from the given pairwise comparison dataset, $\aleph = \{(i,j,\{y_{ij}^k\})\}$. Now, $\widehat{\mathbf{P}} = \widetilde{\mathbf{P}} + \mathbf{S}$. By Lemma 1, we may write the adversarially corrupted empirical probability estimate as $\psi(\widehat{\mathbf{P}}) = \psi(\widetilde{\mathbf{P}}) + \widetilde{\mathbf{S}}$ where $\widetilde{\mathbf{S}} = \psi(\widetilde{\mathbf{P}} + \mathbf{S}) + \psi(1 - \widetilde{\mathbf{P}})$. We have $\psi(\widetilde{\mathbf{P}}) = \psi(\mathbf{P}) + \widetilde{\mathbf{N}}$ where $\widetilde{\mathbf{N}} = \psi(\widetilde{\mathbf{P}}) - \psi(\mathbf{P})$. Now, this noise, $\widetilde{\mathbf{N}}$, is purely due to finite-sample effects which can be controlled (using concentration arguments given in the inequality ξ_3 below) by driving it down to as small a value as we want by ensuring large enough number of comparisons for each pair. Note that we input $\psi(\widehat{\mathbf{P}}) = \psi(\mathbf{P}) + \widetilde{\mathbf{S}} + \widetilde{\mathbf{N}}$ to Subroutine 1 and obtain $\psi(\overline{\mathbf{P}})$ as the output in Step 3 of Algorithm 3. Hence, using Theorem 2 from [30], if $\left\|\widetilde{\mathbf{N}}\right\|_\infty \le \sigma_{\min}(\psi(\mathbf{P}))/100n$, we have,

$$\left\|\psi(\overline{\mathbf{P}}) - \psi(\mathbf{P})\right\|_F \le \epsilon' + 2\mu^2 r\left(7\left\|\widetilde{\mathbf{N}}\right\|_2 + \frac{8n}{r}\left\|\widetilde{\mathbf{N}}\right\|_\infty\right)$$

after $T \ge 10\log(3\mu^2 r\sigma_1/\epsilon')$ iterations associated with Step 1 of Subroutine 1. Next, we have, with probability at least $1 - 1/n^3$,

$$\begin{aligned}\left\|\psi(\overline{\mathbf{P}}) - \psi(\mathbf{P})\right\|_F &\le \epsilon' + 2\mu^2 r\left(7\left\|\widetilde{\mathbf{N}}\right\|_2 + \frac{8n}{r}\left\|\widetilde{\mathbf{N}}\right\|_\infty\right)\\ &\overset{\xi_1}{\le} \epsilon' + 32\mu^2 n\left\|\widetilde{\mathbf{N}}\right\|_2 \overset{\xi_2}{\le} \epsilon' + 32\mu^2 n\tau\\ &\overset{\xi_3}{\le} n\sqrt{\frac{\epsilon}{1+\gamma}}\frac{\Delta}{2}\end{aligned}$$

where ξ_1 follows by using $r \le n$ and $\left\|\widetilde{\mathbf{N}}\right\|_\infty \le \left\|\widetilde{\mathbf{N}}\right\|_2$, ξ_2 follows by substituting for $\widetilde{\mathbf{N}}$ from Lemma 2 with $K \ge \frac{L^2 n^2 \log^2 n}{\tau^2}$, and ξ_3 is obtained using $\epsilon' = n\sqrt{\frac{\epsilon}{1+\gamma}}\frac{\Delta}{4}$, $\tau = \min\left(\sigma_{\min}(\psi(\mathbf{P}))/100, \sqrt{\frac{\epsilon}{1+\gamma}}\frac{\Delta}{128\mu^2}\right)$. Then using similar arguments as proof of Theorem 13 in [37], we obtain our result. □

Lemma 2 (**Concentration of sampling noise**). *Under the conditions of Theorem 1, let each response pair be compared such that the number of comparisons per response pair is $K \ge \frac{L^2 n^2 \log(n)}{\tau^2}$; with probability at least $1 - 1/n^3$, $\left\|\widetilde{\mathbf{N}}\right\|_2 \le \tau$.*

Proof. Let L be the Lipschitz constant of ψ and set $K \ge \frac{L^2 n^2 \log(n)}{\tau^2}$. Using the inequality that $\left\|\widetilde{\mathbf{N}}\right\|_2 \le n\left\|\widetilde{\mathbf{N}}\right\|_\infty$,

$$\begin{aligned}\Pr\left(\left\|\widetilde{\mathbf{N}}\right\|_2 \ge \tau\right) &\le \Pr\left(\left\|\widetilde{\mathbf{N}}\right\|_\infty \ge \frac{\tau}{n}\right)\\ &= \Pr\left(\exists(i,j) : \left|\psi(\widehat{P}_{ij}) - \psi(P_{ij})\right| \ge \frac{\tau}{n}\right)\\ &\le \sum_{i,j}\Pr\left(\left|\psi(\widehat{P}_{ij}) - \psi(P_{ij})\right| \ge \frac{\tau}{n}\right)\\ &\le \sum_{i,j}\Pr\left(\left|\widehat{P}_{ij} - P_{ij}\right| \ge \frac{\tau}{nL}\right) \le \frac{1}{n^3}\end{aligned}$$

□

Next, for completeness, we recall the following lemma (proved in Theorem 8 and Lemma 14 of [37]) which characterizes the incoherence constant μ of $\mathbf{P} \in (\mathcal{P}_n^{LR(\psi,2)} \cap \mathcal{P}_n^{ST})$ in Assumption 1.

Lemma 3 (**Incoherence of BTL and LR models**). *We have $\mathbf{P} \in (\mathcal{P}_n^{LR(\psi,2)} \cap \mathcal{P}_n^{ST})$ if and only if $\psi(\mathbf{P}) = \mathbf{u}\mathbf{v}^\top - \mathbf{v}\mathbf{u}^\top$ for $\mathbf{u} \in \mathbb{R}_+^n$ and $\mathbf{v} \in \mathbb{R}^n$ where $\mathbf{u}^\top\mathbf{v} = 0$. Moreover, $\psi(\mathbf{P})$ is μ-incoherent where $\mu = \sqrt{\frac{n}{2}}\left(\frac{u_{\max}^2}{u_{\min}^2} + \frac{v_{\max}^2}{v_{\max}^2}\right)^{1/2}$ where $u_{\min} = \min_i |u_i|$, $u_{\max} = \max_i |u_i|$, $v_{\min} = \min_i |v_i|$ and $v_{\max} = \min_i |v_i|$. We also have $\mathcal{P}_n^{BTL} \subset (\mathcal{P}_n^{LR(\psi,2)} \cap \mathcal{P}_n^{ST})$ since we may set $\mathbf{u} = \mathbf{1}$ where $\mathbf{1}$ is the all-ones vector and $\mathbf{v} = \mathbf{w}$ where $\mathbf{w}$ is the BTL parameter vector. In this case, we may rewrite $\mu = \sqrt{\frac{n}{2}}\left(1 + \frac{(w_{\max} - \overline{w})^2}{(w_{\min} - \overline{w})^2}\right)$ where $\overline{w} = \frac{1}{n}\sum_{i=1}^n w_i$.*

The following corollary makes precise our claim that up to $O(n^2)$ response pairs may be subject to adversarial corruption, but our RORATRON algorithm still recovers a good ranking.

Corollary 1 (**Recovery result for BTL model**). *Consider $\mathbf{P} \in \mathcal{P}_n^{BTL}$. Using Assumption 1, let the adversarial matrix be $\mathbf{S} \in [0,1]^{n\times n}$ satisfying $\|\mathbf{S}\|_0 \le n/1024\mu^2$ where μ is characterized as in Lemma 3. Then, with probability $1 - 1/n^3$, the output of Algorithm 3 with input $\widehat{\mathbf{P}}$ computed using $\aleph = \{(i,j,\{y_{ij}^k\})\}$ satisfies and $r = 2$, $\operatorname{dist}(\widehat{\sigma}, \mathbf{P}) \le \epsilon$.*

5 PARTIALLY OBSERVERED ADVERSARIAL SETTING

In this section, we consider the partially observed and adversarially corrupted comparison results setting. Both factors can be modeled in a unified manner by setting the corresponding missing entries of the preference matrix to zero (or a specific constant to account for numerical stability). We present our robust ranking algorithm for this setting in Algorithm 5 – this essentially involves using the 'OptSpace' matrix completion algorithm of [24] followed by using the robust PCA algorithm of [30] as sub-routines. We note at this point that, in the case when we are confident that the data are collected faithfully but we do not have the full data to work with, we can use OptSpace on its own to generate the full preference matrix from the incomplete one, as presented in Algorithm 4. We show in Experiment 7.3 below that in such a setting with extremely missing data, we can still complete the full matrix with minimal error.

While the recent work of [32] considers the incomplete data case, it leverages extra information provided in the form of side information (specifically, noiseless and complete item-related features) to derive recovery guarantees; however, their algorithm is still unable to handle the presence of pairwise comparisons corrupted in an adversarial manner as the required assumptions on the noise bounds are violated. We now derive the recovery guarantees as follows.

Theorem 2 (**Provably good estimation of ranking in BTL model in the presence of adversarial noise as well as missing data**). *Consider a similar notation as in Theorem 1 but let $\mathbf{P} \in \mathcal{P}_n^{BTL}$. Let $\Omega \subseteq [n] \times [n]$ be a set of compared response pairs. Assume Ω is drawn uniformly from all subsets of $[n] \times [n]$ of size $|\Omega|$ such that $|\Omega| \ge C'' n\log(n)$ and let the sparse noise satisfy $\|\mathbf{S}\|_\infty \le \Delta_w \frac{\log(n)}{C_\Delta n}$ where $\Delta_w := \min_{i,j}\left|w_i - w_j\right|$. Let the number of*

Algorithm 4 CORATRON: Complete Preference Data for Rigorous Alignment

Input: Comparison dataset $\aleph = \{(i, j, \{y_{ij}^k\})\}$, true rank r.
Output: Ranking of n responses, $\hat{\sigma} \in \mathcal{S}_n$.

1: Estimate entries of $\widehat{\mathbf{P}}$ for $i \leq j$ as:

$$\widehat{P}_{ij} = \begin{cases} \frac{1}{K}\sum_{k=1}^{K} y_{ij}^k & \text{if } i < j \text{ and } (i,j) \in \Omega \\ 1/2 & \text{if } i = j \text{ and } (i,j) \in \Omega \\ 1/2 & \text{if } (i,j) \notin \Omega \end{cases}$$

2: Set $\widehat{P}_{ij} = 1 - \widehat{P}_{ji}$ for all $i > j$.
3: Set $\mathbf{R} \leftarrow \text{OptSpace}(\psi(\widehat{\mathbf{P}})_\Omega)$.
4: Using a pairwise ranking procedure after taking the inverse transform: $\widehat{\sigma} \leftarrow \text{PR}(\mathbf{R})$.
5: **return** $\widehat{\sigma}$.

Algorithm 5 CURATRON: Complete Robust Preference Data for Rigorous Alignment

Input: Comparison dataset $\aleph = \{(i, j, \{y_{ij}^k\})\}$, true rank r.
Output: Ranking of n responses, $\hat{\sigma} \in \mathcal{S}_n$.

1: Estimate entries of $\widehat{\mathbf{P}}$ for $i \leq j$ as:

$$\widehat{P}_{ij} = \begin{cases} \frac{1}{K}\sum_{k=1}^{K} y_{ij}^k & \text{if } i < j \text{ and } (i,j) \in \Omega \\ 1/2 & \text{if } i = j \text{ and } (i,j) \in \Omega \\ 1/2 & \text{if } (i,j) \notin \Omega \end{cases}$$

2: Set $\widehat{P}_{ij} = 1 - \widehat{P}_{ji}$ for all $i > j$.
3: Set $\mathbf{R} \leftarrow \text{OptSpace}(\psi(\widehat{\mathbf{P}})_\Omega)$.
4: Use a robust PCA procedure: $\psi(\overline{\mathbf{P}}) \leftarrow \text{RPCA}(\mathbf{R})$.
5: Using a pairwise ranking procedure after taking the inverse transform: $\widehat{\sigma} \leftarrow \text{PR}(\overline{\mathbf{P}})$.
6: **return** $\widehat{\sigma}$.

comparisons per pair be $K \geq cn^4/\Delta_w$. Then with probability at least $1 - 2/n^3$, Algorithm 5 returns a ranking that satisfies $\text{dist}(\widehat{\sigma}, \mathbf{P}) \leq \epsilon$.

Remark 4 (**Robust Estimation of BTL Model in the Partially Observed Case**). *For the BTL model, Theorem 2 says $O(n \log n)$ pairs suffice to estimate the BTL model which matches bounds from [37]. Further, even in this incomplete comparison data case, we are able tolerate uniformly random additive sparse noise its maximum absolute entry scaling as the order of the BTL 'score-gap' divided by the number of responses upto logarithmic factors, ie, $\widetilde{O}(\Delta_w/n)$.*

Proof. From Lemma 3, we have $\psi(\mathbf{P}) = \mathbf{1}\mathbf{w}^\top - \mathbf{w}\mathbf{1}^\top$ for the BTL model where ψ is the logit function. Clearly, in this case, $\psi(\mathbf{P})$ is a real skew-symmetric matrix of rank $r = 2$. Since it is skew-symmetric, its eigenvalues, which are the roots of its charateristic polynomial, are of the form $\pm\lambda i$ for some $\lambda \in \mathbb{R}$ and $i = \sqrt{-1}$, and hence, $\sigma_{\min}(\psi(\mathbf{P})) = \sigma_{\max}(\psi(\mathbf{P}))$, ie, the condition number of $\psi(\mathbf{P})$, $\kappa = 1$. Now, we recall the spectral-lower bound from Corollary 2 of [18],

$$\sigma_{\min}(\psi(\mathbf{P})) \geq \frac{\|\psi(\mathbf{P})\|_F}{\sqrt{r(r-1)}} \geq \sqrt{\frac{n(n-1)}{2}}\Delta_w \tag{1}$$

where $\Delta_w = \min_{i,j} |w_i - w_j|$.

Let $\Omega \subseteq [n] \times [n]$ be a subset of all the response pairs with comparison results among which some might be corrupted by sparse noise, ie, $\psi(\widehat{\mathbf{P}}_\Omega) = \psi(\mathbf{P}_\Omega) + \widetilde{\mathbf{S}}_\Omega + \widetilde{\mathbf{N}}_\Omega$. Let $\mathbf{T} := \widetilde{\mathbf{S}}_\Omega + \widetilde{\mathbf{N}}_\Omega$. From Theorem 1.2 of [24], we have $\frac{1}{n}\left\|\psi(\widehat{\mathbf{P}}) - \psi(\mathbf{P})\right\|_F = \frac{1}{n}\|\mathbf{T} + \mathbf{M}\|_F \leq C\kappa^2 \frac{n\sqrt{r}}{|\Omega|}\|\mathbf{T}\|_2$ where $\mathbf{M}$ is the noise matrix after obtaining the completed matrix $\psi(\widehat{\mathbf{P}})$ from $\psi(\widehat{\mathbf{P}}_\Omega)$ using OptSpace. Using triangle inequality and noting that $|\Omega| \geq C'' n \log(n)$, the noise may be bounded as

$$\begin{aligned} \left\|\widetilde{\mathbf{N}}_\Omega + \mathbf{M}\right\|_\infty &\leq \left\|\widetilde{\mathbf{N}}_\Omega + \mathbf{M}\right\|_F \leq \|\mathbf{T}\|_2 \frac{\sqrt{2}Cn^2}{|\Omega|} + \left\|\widetilde{\mathbf{S}}_\Omega\right\|_F \\ &\overset{\zeta_1}{\leq} C' \frac{n}{\log(n)} \left\|\widetilde{\mathbf{S}}_\Omega\right\|_2 \end{aligned} \tag{2}$$

where C, C' and C'' are constants and ζ_1 is obtained by using the triangle inequality that $\|\mathbf{T}\|_2 \leq \left\|\widetilde{\mathbf{S}}_\Omega\right\|_2 + \left\|\widetilde{\mathbf{N}}_\Omega\right\|_2$, followed by setting $K \geq cn^4/\Delta_w$ for constant c and finally using $\left\|\widetilde{\mathbf{S}}_\Omega\right\|_F \leq \sqrt{n}\left\|\widetilde{\mathbf{S}}_\Omega\right\|_2$. Then, combining Equations 2 and 1, we have if

$$\begin{aligned} \frac{\log(n)}{C_\Delta n}\Delta_w \geq \left\|\widetilde{\mathbf{S}}_\Omega\right\|_2 &= \left\|\psi(\widehat{\mathbf{P}}) - \psi(\widetilde{\mathbf{P}})\right\|_2 \\ &\geq \left\|\psi(\widehat{\mathbf{P}}) - \psi(\widetilde{\mathbf{P}})\right\|_\infty \geq L\left\|\widehat{\mathbf{P}} - \widetilde{\mathbf{P}}\right\|_\infty \geq \|\mathbf{S}\|_\infty \end{aligned}$$

where C_Δ is a global constant and using Lemma 2, then we have the guarantee (along similar lines as that of Theorem 1 that Algorithm 5 returns an estimated permutation which satisfies $\text{dist}(\widehat{\sigma}, \mathbf{P}) \leq \epsilon$. □

6 GENERALIZATION TO OTHER RANKING MODELS

Related to the BTL model are many other binary choice models [14] such as the Thurstonian model [40]. In such models, the preference matrix has been shown to be low-rank under appropriate choices of ψ; for instance, for the Thurstonian models, the probit function turns out to be the right choice. For further details, we refer the reader to the work of [37].

Let $a, b, c \in (0, 1)$ such that $c = a + b$. Then, for any general non-linear L-Lipschitz function, we write $\psi(c) = \psi(a+b) = \psi(a) + \psi(a+b) - \psi(a)$. The error may be lower bounded by $|\psi(a+b) - \psi(a)| \geq Lb$. Thus, for any adversarial model wherein we have $\mathbf{P^c} = \mathbf{P} + \mathbf{S}$, we have:

$$\psi(\mathbf{P^c}) = \psi(\mathbf{P}) + (\psi(\mathbf{P} + \mathbf{S}) - \psi(\mathbf{P})) = \psi(\mathbf{P}) + \widetilde{\mathbf{S}}$$

where $\widetilde{\mathbf{S}}$ is also a deterministic sparse corruption matrix with the absolute value of the non-zero entries lower bounded by $L.\min_{i,j} S_{ij}$. With the appropriate ψ, $\psi(\mathbf{P})$ will be a low-rank matrix and hence Algorithm 3 and the associated recovery guarantee of Theorem 1 holds.

7 EXPERIMENTS

In this section, we answer Question 5. We now perform simulations in order to understand the performance of our robust ranking approach in practice in both general and LLM preference dataset settings.

7.1 Evaluation Criterion

We use several evaluations to assess our proposed methods' effectiveness against unobserved and adversarial corrupted comparisons.

7.1.1 Normalized Frobenius Error. First, To measure the relative error between two preference matrices in terms of their elements' magnitudes, we use the normalized Frobenius error (NFE). NFE between two matrices P and $\overline{P}$ is defined as:

$$NFE(P, \overline{P}) = \frac{\|P - \overline{P}\|_{Fro}}{\|P\|_{Fro}},$$

where the Frobenius norm, denoted as $\|A\|_{Fro}$, for a matrix A is calculated by:

$$\|A\|_{Fro} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2}$$

In this formula, a_{ij} represents the element of the matrix A in the ith row and jth column. The Frobenius norm is the square root of the sum of the absolute squares of all elements in the matrix. Thus, the numerator $\|P-\overline{P}\|_{Fro}$ calculates the Frobenius norm of the difference between the original and reconstructed matrices, and the denominator $\|P\|_{Fro}$ calculates the Frobenius norm of the original matrix. The ratio provides a measure of the relative error normalized by the magnitude of the original matrix.

7.1.2 Correlation Coefficient. Second, we compute the correlation coefficient for corresponding elements in these matrices to assess the similarity between the original matrix P and the reconstructed matrix $\overline{P}$. The correlation coefficient, denoted as $corr$, between the elements of these two matrices can be defined as:

$$corr(P, \overline{P}) = \frac{\sum_{i=1}^{n}(P_i - \langle P\rangle)(\bar{P}_i - \langle \bar{P}\rangle)}{\sqrt{\sum_{i=1}^{n}(P_i - \langle P\rangle)^2}\sqrt{\sum_{i=1}^{n}(\bar{P}_i - \langle \bar{P}\rangle)^2}},$$

where $\langle P\rangle$ and $\langle \bar{P}\rangle$ denote the mean values of the elements within the P and $\bar{P}$ matrices, respectively. n represents the total number of elements in each matrix.

This formula quantifies the linear relationship between the matrices' elements. A correlation coefficient close to 1 indicates a strong positive linear relationship, whereas a value close to -1 suggests a strong negative linear relationship. A coefficient around 0 implies no linear relationship.

7.1.3 Ranking Distance. Third, for ease of reference, we rewrite the dist $(\widehat{\sigma}, \mathbf{P})$ formula, which evaluates the distance between rankings obtained by corrupted and recovered matrices, previously defined in Section **??**:

$$\text{dist}\,(\widehat{\sigma}, \mathbf{P}) := \binom{n}{2}^{-1}\sum_{i<j}\mathbb{1}\left((P_{ij} > 1/2) \wedge (\widehat{\sigma}(i) > \widehat{\sigma}(j))\right) + \binom{n}{2}^{-1}\sum_{i<j}\mathbb{1}\left((P_{ji} > 1/2) \wedge (\widehat{\sigma}(j) > \widehat{\sigma}(i))\right),$$

where $\widehat{\sigma}$ is the global ranking after applying ranking procedure with $\bar{P}$.

7.2 Performance of Robust Ranking in General Setting

First, we begin with the BTL model. We generate synthetic pairwise comparison data and also adversarial sparse matrix as follows. We generate the entries of the BTL parameter vector $\mathbf{w}$ from $\mathcal{N}(0, \nu^2)$ followed by generating the ground truth preference matrix from with y_{ij}^k is sampled for all response pairs (i, j) for a fixed K. The adversarial sparse matrix $\mathbf{S}$ is generated as a skew-symmetric matrix where each entry is non-zero independently with probability d/n followed by generating a value for an entry from $U(5, 10)$ and then setting the sign to be positive with probability $1/2$; this corruption matrix is then added to the $\psi(\mathbf{P})$ to give $\psi(\mathbf{P^c})$ which is then input to our algorithm; the same $\mathbf{P^c}$ is used for the other algorithms as well.

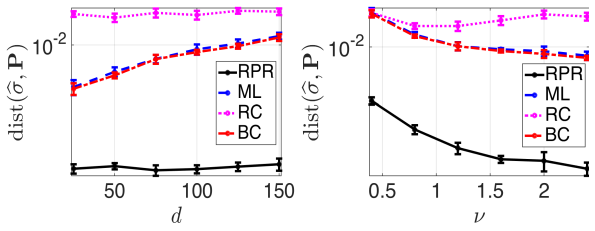

Figure 2: Robust recovery results of the BTL model: we fix $\nu = 2$ and vary d in the left plot; we fix $d = 100$ and vary ν in the left plot.

We take the number of responses to be $n = 500$. In plots in Figure 2, we compare the performance of our RPR approach using Algorithm 3 against well-known ranking algorithms, such as Rank Centrality (RC [28], Maximum Likelihood (ML) and Borda Count (BC) count [23], with special attention to robustness to the noise model that we consider in this paper. We vary two parameters namely, ν, spread of the BTL scores, and d, the density of adversarial corruption matrix. All our results averaged over five runs. We observe that our algorithm maintains low recovery error in spite of increasing the problem hardness, thus outperforming previous approaches in all cases.

7.3 Performance of Robust Ranking in LLM Preference Dataset

In this illustrative experiment, from the MT-Bench dataset [48], we collect the data of the first prompt "Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions" and its six responses from `GPT-3.5`, `GPT-4` [33], `Claude-v1` [3], `Vicuna-13B` [7], `Alpaca-13B` [39], and `LLaMA-13B` [41]. Additionally, we generated nine responses to the same prompt using `Llama-2-70B-chat-hf` [42], `Falcon-180B-chat` [2], `Openchat-3.5` [44], `Mixtral-8x7B-Instruct-v0.1` [22], `Mistral-7B-Instruct-v0.2` [21], `Gemini-pro` [15], `Dolphin-2.2.1-mistral-7B` [16], `Solar-10.7B-instruct-v1.0` [25], `Yi-34B-chat` [1] from Hugging Face's HuggingChat [20] and LMSYS's Chatbot Arena [48]. So we have $n = 15$ responses.

Next, we rank the responses using OpenAI's GPT-4 Turbo `GPT-4-1106-preview` [33]. This ranking helps us create the BTL parameter vector $\mathbf{w}$. We then sort this vector descendingly for visually accessible when building the corresponding preference matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$. With $\binom{n}{2}$ comparisons in $\mathbf{P}$, we randomly remove entries based on a specified deletion probability parameter, dp, to

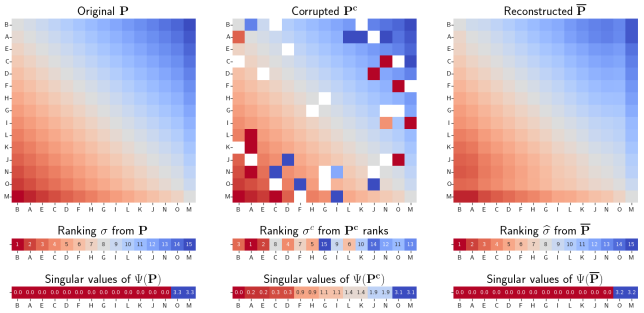

Figure 3: Left: Original matrix. Middle: corrupted matrix. Right: reconstructed matrix. The corrupted matrix has 10% adversarial corruptions and 10% of unobserved comparisons. We use our CURATRON algorithm to successfully recover the original matrix.

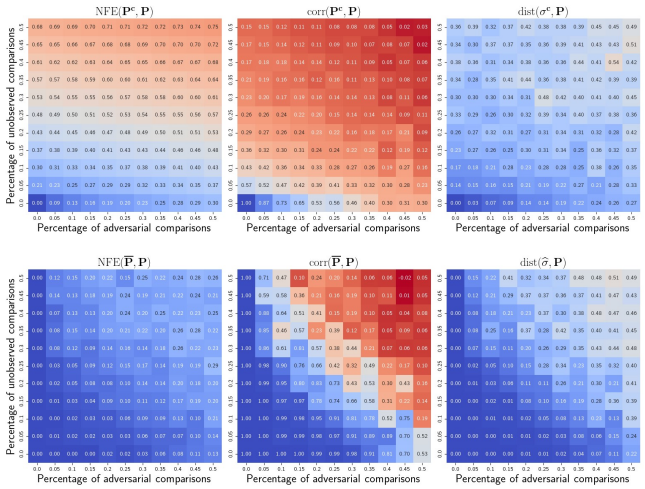

Figure 4: Average over 5 runs of reconstruction error, correlation, and distance between reconstructed ranking and original ranking for different percentages of unobserved and adversarial comparisons.

simulate unobserved comparisons. We then create an adversarial skew-symmetric sparse matrix, $\mathbf{S}$, using the given matrix $\mathbf{P}$ and an adversarial corruption probability parameter ap. When corruption is applied, it involves randomly selecting a value from $U(-5, 5)$ and then adding to the $\mathbf{P}$ to give $\mathbf{P}^c$ which is then become the input of our algorithm. It's important to note that $\mathbf{P}$ is a skew-symmetric matrix, any corruption must be applied to both ij and ji values.

Our experiment results visualized in Figure 3 show that $dp = 10\%$ and $da = 10\%$ can significantly affect the ranking of different models and the rank of the matrix when performing logit link transformation. The ranking can get altered quite badly when compared to the original matrix. Also, the logit link transformation of the corrupted matrix is high-rank which indicates that there are noises in the matrix. By using CURATRON to impute the missing comparisons and filter out the noisy sparse matrix, we successfully reconstruct the original matrix, which is low-rank when in logit link transformed form. As a result, we obtain the correct ranking. We also obtain noisy comparisons that can be used to identify responders with malicious intent and prevent them from continuing to alter results.

We now examine how our algorithm performs across different levels of unobserved and adversarially corrupted comparisons. In the plots shown in Figure 4, we compare the performance of our approach by varying two parameters, dp and ap. We use NFE, correlation, and ranking distance as defined earlier in section 7.1. Our results are averaged over 5 runs. When there is no adversarial noise, we can recover the original $\mathbf{P}$ with no NFE and perfect correlation and ranking, even if 50% of the comparison data was missing. This suggests that we may not need to collect all comparisons from humans to obtain the entire data. We observe that, with $n = 15$, we only need to obtain about $50 - 55\%$ of the 105 comparisons and fill in the rest with our algorithm to achieve a strict 0% NFE, perfect correlation, and ranking. On the other hand, when missing data is absent, our algorithm performs well with NFE of approximately 6%, even when 35% of the comparison data is adversarially corrupted. When both adversarial noise and missing data are present, we can achieve a low NFE of around 4% when both 15% of the comparison data is missing and 15% of adversarially corrupted comparisons (30% in total) affect $\mathbf{P}$.

8 CONCLUSION

Our study examines how missing information and distorted feedback can impact LLMs, potentially compromising their performance in terms of alignment with human values. We have proposed a robust algorithm for provably correct and efficient ranking responses in the BTL, LR, and general binary choice models. This robust ranking data is then input in the PL step. Further, we also handled the partially observed setting, wherein only some response pairs are compared, by integrating matrix completion techniques into our robust learning algorithm. In all cases, we provided statistical and computational guarantees using novel techniques. Through our comprehensive analysis, we hope to contribute to the ongoing discussion on AI safety by helping to create and scale LLMs/AGI models that align with human values and expectations. Some future research directions include tightening the recovery results for partially observed settings under weaker conditions (possibly using noisy-case extensions of [46]), exploring other notions of adversarial noise, and understanding the minimax optimal rates for ranking estimators under various noise models. We also plan to study the parametric non-active pairwise ranking setting, studying lower bounds and practical algorithms in the active setting similar to [17]. Furthermore, it would be interesting to investigate whether we can extend this approach to solve the entity corruption problem in retrieval models, as shown in [27]. Another research direction could be defining an alignment framework that expands DPO to various objective functions based on Rank Centrality [29]. Finally, we aim to examine the relationship between robust PL and model capacity, as this can shed light on the trade-offs between model complexity and generalization performance.

REFERENCES

[1] 01.ai. 2023. Yi-34B. https://www.01.ai. Accessed 03-03-2024.

[2] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The Falcon Series of Open Language Models. arXiv:2311.16867 [cs.CL]

[3] Anthropic. 2023. Introducing Claude. https://www.anthropic.com/news/introducing-claude. Accessed 03-03-2024.

[4] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862* (2022).

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

[6] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217* (2023).

[7] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality. https://lmsys.org/blog/2023-03-30-vicuna/. Accessed 03-03-2024.

[8] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[9] Arthur H Copeland. 1951. A reasonable social welfare function. In *Mimeographed notes from a Seminar on Applications of Mathematics to the Social Sciences, University of Michigan*.

[10] Don Coppersmith, Lisa Fleischer, and Atri Rudra. 2006. Ordering by weighted number of wins gives a good ranking for weighted tournaments. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*. Society for Industrial and Applied Mathematics, 776–782.

[11] Mihai Cucuringu. 2016. Sync-Rank: Robust Ranking, Constrained Ranking and Rank Aggregation via Eigenvector and SDP Synchronization. *IEEE Transactions on Network Science and Engineering* 3, 1 (2016), 58–79.

[12] Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. 2023. GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models. arXiv:2303.10130 [econ.GN]

[13] Kawin Ethayarajh, Winnie Xu, Dan Jurafsky, and Douwe Kiela. 2023. *Human-Centered Loss Functions (HALOs)*. Technical Report. Contextual AI. https://github.com/ContextualAI/HALOs/blob/main/assets/report.pdf.

[14] Peter C Fishburn. 1973. Binary choice probabilities: on the varieties of stochastic transitivity. *Journal of Mathematical psychology* 10, 4 (1973), 327–352.

[15] Gemini, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805* (2023).

[16] Eric Hartford. 2023. Dolphin-2.2.1-mistral-7b. https://huggingface.co/cognitivecomputations/dolphin-2.2.1-mistral-7b. Accessed 03-03-2024.

[17] Reinhard Heckel, Nihar B Shah, Kannan Ramchandran, and Martin J Wainwright. 2016. Active Ranking from Pairwise Comparisons and when Parametric Assumptions Don't Help. *arXiv preprint arXiv:1606.08842* (2016).

[18] Bill G Horne. 1997. Lower bounds for the spectral radius of a matrix. *Linear algebra and its applications* 263 (1997), 261–273.

[19] Daniel Hsu, Sham M Kakade, and Tong Zhang. 2011. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory* 57, 11 (2011), 7221–7234.

[20] Hugging Face. 2023. HuggingChat. https://huggingface.co/chat. Accessed 03-03-2024.

[21] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. arXiv:2310.06825 [cs.CL]

[22] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of Experts. arXiv:2401.04088 [cs.LG]

[23] Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. 2011. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming* 127, 1 (2011), 203–244.

[24] Raghunandan H Keshavan, Andrea Montanari, and Sewoong Oh. 2010. Matrix completion from noisy entries. *Journal of Machine Learning Research* 11, Jul (2010), 2057–2078.

[25] Dahyun Kim, Chanjun Park, Sanghoon Kim, Wonsung Lee, Wonho Song, Yunsu Kim, Hyeonwoo Kim, Yungi Kim, Hyeonju Lee, Jihoo Kim, Changbae Ahn, Seonghoon Yang, Sukyung Lee, Hyunbyung Park, Gyoungjin Gim, Mikyoung Cha, Hwalsuk Lee, and Sunghun Kim. 2023. SOLAR 10.7B: Scaling Large Language Models with Simple yet Effective Depth Up-Scaling. arXiv:2312.15166 [cs.CL]

[26] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267* (2023).

[27] Niranjan Uma Naresh, Ziyan Jiang, Ankit, Sungjin Lee, Jie Hao, Xing Fan, and Chenlei Guo. 2022. PENTATRON: PErsonalized coNText-Aware Transformer for Retrieval-based cOnversational uNderstanding. arXiv:2210.12308 [cs.LG]

[28] Sahand Negahban, Sewoong Oh, and Devavrat Shah. 2012. Iterative ranking from pair-wise comparisons. In *Advances in Neural Information Processing Systems*. 2474–2482.

[29] Sahand Negahban, Sewoong Oh, and Devavrat Shah. 2017. Rank Centrality: Ranking from Pairwise Comparisons. *Operations Research* 65, 1 (2017), 266–287. http://www.jstor.org/stable/26153541

[30] Praneeth Netrapalli, UN Niranjan, Sujay Sanghavi, Animashree Anandkumar, and Prateek Jain. 2014. Non-convex robust PCA. In *Advances in Neural Information Processing Systems*. 1107–1115.

[31] Son The Nguyen, Theja Tulabandhula, and Mary Beth Watson-Manheim. 2023. User Friendly and Adaptable Discriminative AI: Using the Lessons from the Success of LLMs and Image Generation Models. arXiv:2312.06826 [cs.AI]

[32] UN Niranjan and Arun Rajkumar. 2017. Inductive Pairwise Ranking: Going Beyond the n log (n) Barrier.. In *AAAI*. 2436–2442.

[33] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[34] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. arXiv:2203.02155 [cs.CL]

[35] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290* (2023).

[36] Arun Rajkumar and Shivani Agarwal. 2014. A Statistical Convergence Perspective of Algorithms for Rank Aggregation from Pairwise Data.. In *ICML*. 118–126.

[37] Arun Rajkumar and Shivani Agarwal. 2016. When can we rank well from comparisons of $O(n \log(n))$ non-actively chosen pairs?. In *29th Annual Conference on Learning Theory*. 1376–1401.

[38] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347 [cs.LG]

[39] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford Alpaca: An Instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca. Accessed 03-03-2024.

[40] Louis L Thurstone. 1927. A law of comparative judgment. *Psychological review* 34, 4 (1927), 273.

[41] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs.CL]

[42] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288 [cs.CL]

[43] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of LM alignment. arXiv:2310.16944 [cs.LG]

[44] Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023. OpenChat: Advancing Open-source Language Models with Mixed-Quality Data. arXiv:2309.11235 [cs.CL]

[45] Lidan Wang, Paul N Bennett, and Kevyn Collins-Thompson. 2012. Robust ranking models via risk-sensitive optimization. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 761–770.

[46] Xinyang Yi, Dohyung Park, Yudong Chen, and Constantine Caramanis. 2016. Fast Algorithms for Robust PCA via Gradient Descent. *arXiv preprint arXiv:1605.07784* (2016).

[47] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. 2023. SLiC-HF: Sequence Likelihood Calibration with Human

Feedback. arXiv:2305.10425 [cs.CL]
[48] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. arXiv:2306.05685 [cs.CL]
[49] Yan-Bo Zhou, Ting Lei, and Tao Zhou. 2011. A robust ranking algorithm to spamming. *EPL (Europhysics Letters)* 94, 4 (2011), 48002.