ERE-WiS: Explore-Refine-Exploit for Winner determination from Shoestring Pairwise Comparisions

Saranath P
Dept. of Data Science and Artificial Intelligence
IIT Madras
Chennai, India
da25e003@smail.iitm.ac.in

Arun Rajkumar
Dept. of Data Science and Artificial Intelligence
IIT Madras
Chennai, India
arunr@dsai.iitm.ac.in

***Abstract*—We study the problem of identifying a winner from a set of items using actively selected pairwise comparisons under strict budget constraints. To address this challenge, we introduce Explore–Refine–Exploit, a novel three-phase framework, and propose ERE-WiS, a parameter-free algorithm grounded in this paradigm. Through extensive experiments, we demonstrate that ERE-WiS consistently outperforms state-of-the-art methods such as PARWiS, SELECT, and others on both synthetic and real-world datasets. Furthermore, our comprehensive ablation studies provide deeper insights into the effectiveness and robustness of ERE-WiS, underscoring its practical suitability for winner identification in highly budget-limited settings.**

***Index Terms*—Active Learning, Pairwise Comparisons, Ranking, Winner Determination, Budgeted Learning, BTL Model**

Not needed

I. Introduction

Ranking from pairwise comparisons is a fundamental problem with wide-ranging applications in crowdsourcing [1], social choice theory [2], recommender systems [3], and sports analytics [4]. In the active version of this problem, an algorithm sequentially chooses which pair of items to compare, based on the outcomes of previous comparisons. The goal is often to minimize the number of queries needed to recover a full ranking, the top-K items, or, as we focus on in this work, the single best item (the winner).

While winner recovery in a noise-free setting is trivial (requiring only $n-1$ comparisons), real-world comparisons are inherently noisy. The Bradley-Terry-Luce (BTL) model [5], [6] is a popular and principled way to model this noise, assigning a latent positive score to each item to govern the probabilistic outcomes of comparisons. Existing algorithms designed for the BTL model typically require a sample complexity of $O(n \log n)$ or $O(n)$ to guarantee high-confidence winner recovery. However, these theoretical bounds often hide large, problem-dependent constants that render them impractical in scenarios with severe budget limitations.

$O(n \log \log n)$

Our work is motivated by such practical scenarios, where one is given a "shoestring budget" (e.g., cn for a small constant c, like 2 or 3) to find the winner. At this budget scale, traditional high-confidence guarantees become vacuous, and algorithms may not even complete their first phase of operation. The central challenge is thus to design an algorithm that makes the most intelligent use of every single comparison to maximize the practical probability of finding the true winner.

Bold.

To this end, we propose a novel three-phase framework: **Explore-Refine-Exploit**. This paradigm provides a structured approach to active winner recovery under a shoestring budget. Based on this framework, we introduce **ERE-WiS**, a new algorithm that dynamically adapts its comparison strategy.

Our Contributions:

1) We propose the Explore-Refine-Exploit framework, a structured, three-phase approach for active winner identification under strict budget constraints.
2) We introduce ERE-WiS, a novel, parameter-free algorithm based on this framework. ERE-WiS begins with a broad exploration, refines its understanding of the top contender, and finally exploits this knowledge by focusing comparisons among the most promising candidates.
3) Through extensive experiments on synthetic and real-world datasets, we show that ERE-WiS significantly and consistently outperforms a suite of state-of-the-art algorithms, including PARWiS, SELECT, MultiSort, and RUCB, especially in the critical low-budget regime.

II. Related Work

Passive Learning: In the passive setting, the set of comparisons is given upfront. Rank Centrality [7] provides a spectral method to recover underlying BTL scores from such data. It forms the basis of our internal ranking mechanism due to its efficiency and robustness. Other works like Spectral MLE [8] refine these estimates further. Our work uses spectral ranking as a subroutine within an active learning loop.

Active Learning: The active ranking field was pioneered by works like [9], which used geometric ideas for noise models simpler than BTL. For the BTL model, state-of-the-art algorithms include SELECT [10], a single-elimination tournament-based approach, and MultiSort [11], which aggregates multiple noisy quicksort runs. Active Ranking (AR) [12] provides a generic framework for various ranking goals.

Remove

More recently, PARWiS [13] introduced an adaptive spectral method that chooses "disruptive" pairs to query. Our ERE-WiS algorithm builds upon similar spectral foundations but employs a fundamentally different, phase-based query strategy designed for extreme budgets.

Bandits: The dueling bandits framework is closely related to winner determination. Algorithms like Relative UCB (RUCB) [14] and Sparse Borda [15] adapt bandit exploration-exploitation strategies to the pairwise comparison setting. These methods aim to identify the Borda winner, which coincides with the BTL winner. We compare against RUCB as a representative state-of-the-art bandit approach.

III. Problem Setting and Preliminaries

We consider a set of n items, denoted by $[n] = \{1, 2, \ldots, n\}$. An active learning algorithm can query any pair of items (i, j). The outcome of a comparison is a random variable, where item i is declared the winner over item j with probability P_{ij}. We assume these probabilities are governed by the Bradley-Terry-Luce (BTL) model, which posits the existence of a latent score vector $w \in \mathbb{R}_+^n$ such that for any pair (i, j):

$$P_{ij} = \frac{w_i}{w_i + w_j} \tag{1}$$

The goal is to identify the winner, which is the item $i^* = \arg\max_{i \in [n]} w_i$, by making at most B pairwise comparisons, where B is a pre-defined shoestring budget.

Enhance this more. On How markov chain is framed.

A. Spectral Ranking Algorithm

A core component of our method is the ability to estimate the score vector w from a collection of comparison outcomes. We use the spectral ranking algorithm, Rank Centrality [7], for this purpose. Given a set of pairwise comparisons, we can construct a directed graph where an edge from i to j represents a win for i over j. From the adjacency matrix of this comparison graph, an augmented Markov chain transition matrix Q is constructed. The stationary distribution π of this chain serves as an estimate for the BTL score vector w. The algorithm is summarized in Algorithm III-A.

[h] Spectral Ranking (Rank Centrality) [7]

May be visually represent the graph.

1. **Require:** Number of items n, set of comparison outcomes $S = \{(winner, loser), \ldots\}$.
2. Construct a comparison count matrix A, where A_{ij} is the number of times i beat j.
3. Compute the augmented Markov chain matrix Q from A.
4. Compute the stationary distribution π of Q (e.g., via power iteration).
5. **Output:** Ranking based on sorting the values in π.

IV. The ERE-WiS Algorithm

We now introduce our proposed algorithm, ERE-WiS, which is grounded in the **Explore-Refine-Exploit** paradigm. The algorithm dynamically transitions through three distinct phases to make efficient use of a limited budget. A high-level overview is provided in Algorithm IV.

Bold.

[hbt!] ERE-WiS Algorithm Properly PU

```
Input: Number of items n, Budget B.
Parameters: Top candidate set size k = ⌈√n⌉, refinement
    win-rate threshold θ = 0.33.
Data: S ← ∅ (set of comparison outcomes)
Phase 1: Explore
Run a single-elimination tournament on a random permutation of all n items. Add all comparison outcomes to S.
Compute initial score estimates π using Spectral Ranking
    (Alg. III-A) on S.
Let R be the ranking induced by π.
Phase 2: Refine
pivot ← R[1] (item ranked 1st).
challengers ← {R[k+1], ..., R[2k]}.
pivot_wins ← 0, comps ← 0.
while budget not exceeded AND phase is Refine do
    for j in challengers do
        if budget exceeded then
            break
        end if
        Query pair (pivot, j). Add outcome to S.
        if pivot wins then
            pivot_wins ← pivot_wins + 1
        end if
        comps ← comps + 1.
    end for
    if pivot_wins/comps > θ then
        Transition to Phase 3.
    else
        Re-compute π and R. Update pivot and
        challengers. pivot_wins, comps ← 0, 0.
    end if
end while
Phase 3: Exploit
C ← {R[1], ..., R[k]} (top-k candidate set).
while budget not exceeded do
    Query a random pair (i, j) from C. Add outcome to S.
end while
Output: Re-compute final π on all data in S. Return item
    with the highest score in π.
```

No hardcoded.

K+m

Every thing & later m=k mathematical.

& Explore should also be mathematical.

give better names.

PARWis with only top K guys.

A. Phase 1: Explore

The goal of the initial phase is to quickly gain a coarse understanding of the entire item landscape and identify a set of promising candidates. ERE-WiS uses a single-elimination tournament for this purpose. All n items are randomly permuted and paired off. Winners advance to the next round until a single champion is declared. This process is highly efficient, requiring only $n-1$ comparisons. The outcomes of all these comparisons are used to compute an initial spectral ranking, which provides a much richer signal than just the tournament winner alone.

B. Phase 2: Refine

After the initial exploration, the algorithm enters a refinement phase. The goal here is to verify the strength of the current top-ranked item (the 'pivot') against a set of strong 'challengers'. Based on the initial ranking, the pivot is defined as the top-ranked item, and challengers are the set of items ranked just outside the primary candidate pool (e.g., items ranked from $k+1$ to $2k$, where $k \approx \sqrt{n}$).

The algorithm iteratively compares the pivot against all challengers. If the pivot demonstrates a sufficiently high win rate against this group, its top status is considered validated, and the algorithm transitions to the final exploitation phase. If not, it means the initial ranking may have been inaccurate. The algorithm then re-computes the full spectral ranking using the newly acquired data and re-starts the refinement phase with an updated pivot and challenger set. This phase prevents the algorithm from prematurely focusing on a suboptimal candidate.

C. Phase 3: Exploit

Once the pivot's strength is confirmed, the algorithm transitions to the exploit phase. It now has high confidence that the true winner lies within a small set of top-k candidates. The remaining budget is dedicated exclusively to performing comparisons within this candidate set. By focusing queries on this small subset, the algorithm can efficiently resolve ambiguities and pinpoint the single best item among the most likely contenders. A random pair selection strategy within the candidate set ensures that all potential winners are compared, breaking ties and refining the local ranking until the budget is exhausted.

V. Experiments

We conduct a comprehensive set of experiments to evaluate the performance of ERE-WiS. We compare it against several state-of-the-art algorithms on both synthetic and real-world datasets.

A. Performance Metrics

To judge the quality of the algorithms, we use three primary metrics:

1) **Recovery Fraction (ACC):** The fraction of experimental runs where the algorithm correctly identifies the true winner. This is the primary measure of success.
2) **True Rank of Reported Winner (CT):** The average true rank of the item returned by the algorithm. A low value indicates that even when the algorithm is wrong, it reports an item that is close to the top.
3) **Reported Rank of True Winner (PF):** For algorithms that produce a full ranking, this is the average rank they assign to the true winner. A low value indicates the algorithm correctly recognizes the true winner's strength, even if it doesn't place it at the very top.

B. Baseline Algorithms

We compare ERE-WiS against the following strong baselines:

- **PARWiS** [13]: The predecessor to this work, an active spectral method that queries the "most disruptive pair". We use the standard King-of-the-Hill initialization.
- **SELECT** [10]: A competitive algorithm that runs a single-elimination tournament. For budgets larger than $n-1$, it performs multiple comparisons per match.
- **MultiSort** [11]: An algorithm that runs multiple noisy quicksorts and aggregates the results using the Copeland rule.
- **RUCB** [14]: A state-of-the-art dueling bandits algorithm based on the upper confidence bound principle.

C. Experimental Setup

Synthetic Data: We generate BTL scores for $n = 100$ items. The true winner is assigned a score of 100. The scores of the remaining $n-1$ items are drawn uniformly from $[0, k]$, where $k \in \{10, 20, ..., 90\}$. The parameter k controls the difficulty of the problem: a larger k means the winner has less separation from the pack, making it harder to identify. All results are averaged over 200 independent runs.

Real-world Data: We also use standard benchmark datasets from literature, such as the Jester joke dataset [16] and the Netflix Prize dataset [17]. For these datasets, which contain user ratings, we first derive empirical pairwise preference probabilities and then compute a ground-truth BTL score vector using spectral ranking. These scores are then used to simulate pairwise comparisons for the active learning algorithms.

D. Results and Discussion

The results of our experiments are shown in Figure 1 (Synthetic data) and Figure 2 (Jester dataset). Across all datasets and metrics, ERE-WiS demonstrates a clear and consistent advantage over the state-of-the-art baselines.

On the synthetic data, as shown in Figure 1, ERE-WiS achieves a significantly higher recovery fraction (ACC) for any given budget. At a shoestring budget of $B = 200$ (or $2n$), ERE-WiS already identifies the winner in a high percentage of runs, whereas other methods lag considerably. Furthermore, the CT and PF metrics show that ERE-WiS is more robust; even when it fails to find the absolute winner, it returns a very highly ranked item.

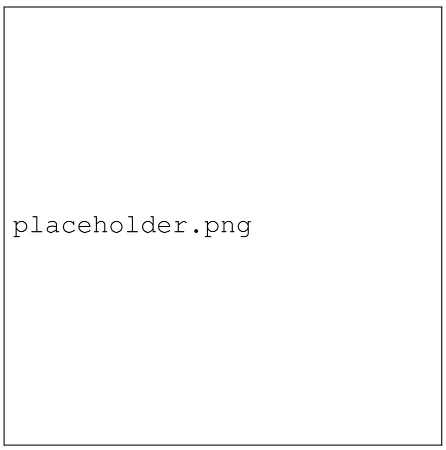
The superior performance can be attributed to the Explore-Refine-Exploit strategy. The initial 'Explore' phase quickly establishes a strong baseline ranking. The 'Refine' phase acts as a crucial verification step, preventing the algorithm from committing to a specious winner. Finally, the 'Exploit' phase efficiently uses the remaining budget to resolve ambiguity among the most likely candidates. In contrast, methods like PARWiS can sometimes focus too early on a suboptimal pair, while methods like SELECT and MultiSort have a more rigid query structure that is less adaptive to the data observed so far.

(a) Recovery Fraction (ACC)

(b) True Rank of Reported Winner (CT)

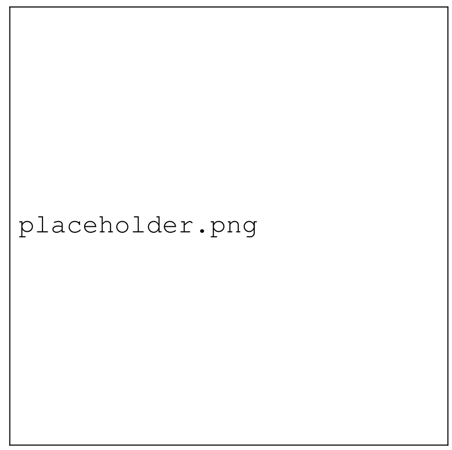
(c) Reported Rank of True Winner (PF)

Fig. 1. Performance comparison on Synthetic Data ($n = 100$, difficulty $k = 75$). The x-axis represents the budget B, and the y-axis represents the performance metric. ERE-WiS consistently outperforms all baseline algorithms across all three metrics. The performance gap is most pronounced in the critical low-budget regime ($B < 3n$).



(a) Recovery Fraction (ACC)

(b) True Rank of Reported Winner (CT)



(c) Reported Rank of True Winner (PF)

Fig. 2. Performance comparison on the Jester Dataset. ERE-WiS demonstrates superior performance, achieving a higher recovery fraction with fewer comparisons. Its robustness is highlighted by the low values for the CT and PF metrics, indicating it consistently identifies high-quality candidates.

The results on the Jester dataset (Figure 2) confirm these findings on real-world data, further underscoring the effectiveness and practical utility of ERE-WiS for winner determination under tight budget constraints.

VI. Conclusion

In this paper, we addressed the practical problem of winner determination from active pairwise comparisons under a shoestring budget. We introduced the Explore-Refine-Exploit framework, a principled, three-phase approach for this setting. We proposed ERE-WiS, a novel algorithm based on this paradigm that is both effective and parameter-free. Our extensive experiments showed that ERE-WiS consistently and significantly outperforms existing state-of-the-art methods on a variety of synthetic and real-world datasets. The results highlight the power of its adaptive, multi-phase strategy in making every comparison count. Future work could involve deriving theoretical guarantees for ERE-WiS and extending the framework to the more general top-K identification problem.

References

[1] X. Chen, P. N. Bennett, K. Collins-Thompson, and E. Horvitz, "Pairwise ranking aggregation in a crowdsourced setting," in *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013, pp. 193–202.

[2] F. Brandt, V. Conitzer, and U. Endriss, "Computational social choice," *Multiagent Systems*, pp. 213–283, 2012.

[3] N. Houlsby, F. Huszár, Z. Ghahramani, and J. M. Hernándezlobato, "Collaborative gaussian processes for preference learning," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[4] A. E. Elo, *The Rating Of Chess Players, Past and Present*. New York: Arco Pub., 1978.

[5] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: The method of paired comparisons," *Biometrika*, vol. 39, no. 3-4, pp. 324–345, 1952.

[6] R. D. Luce, *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959.

[7] S. Negahban., S. Oh, and D. Shah, "Rank centrality: Ranking from pairwise comparisons," *Operations Research*, vol. 65, no. 1, pp. 266–287, 2016.
[8] Y. Chen and C. Suh, "Spectral mle: top-k rank aggregation from pairwise comparisons," in *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37, 2015, pp. 371–380.
[9] K. G. Jamieson and R. D. Nowak, "Active ranking using pairwise comparisons," *Advances in Neural Information Processing Systems*, vol. 24, 2011.
[10] S. Mohajer, C. Suh, and A. Elmahdy, "Active learning for top-k rank aggregation from noisy comparisons," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 2488–2497.
[11] L. Maystre and M. Grossglauser, "Just sort it! a simple and effective approach to active preference learning," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, 2017, pp. 2344–2353.
[12] R. Heckel, N. B. Shah, K. Ramchandran, and M. J. Wainwright, "Active ranking from pairwise comparisons and when parametric assumptions do not help," *The Annals of Statistics*, vol. 47, no. 6, pp. 3099–3126, 2019.
[13] D. Y. Sheth and A. Rajkumar, "PARWIS: Winner determination from Active Pairwise Comparisons under a Shoestring Budget," *arXiv preprint arXiv:2106.12345*, 2021.
[14] M. Zoghi, S. Whiteson, R. Munos, and M. de Rijke, "Relative upper confidence bound for the k-armed dueling bandit problem," in *Proceedings of the 31st International Conference on Machine Learning*, vol. 32, 2014.
[15] K. Jamieson, S. Katariya, A. Deshpande, and R. Nowak, "Sparse dueling bandits," in *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, vol. 38, 2015, pp. 416–424.
[16] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: A constant time collaborative filtering algorithm," *Information Retrieval*, vol. 4, no. 2, pp. 133–151, 2001.
[17] J. Bennett and S. Lanning, "The netflix prize," in *Proceedings of KDD Cup and Workshop*, 2007.