Faster Maximum Inner Product Search in High Dimensions (Reproducibility)

Viraj Prajapati (202311069), Parth Borad (202311033), Sarthak Vadher (202311040) and Dhairya Dave (202311011)

Abstract

Maximum Inner Product Search (MIPS) is a critical task in machine learning, often used in recommendation systems, where the goal is to identify the atom vector with the highest inner product relative to a query vector. Traditional MIPS algorithms scale as $O(\sqrt{d})$, making them computationally expensive in high-dimensional settings. The BanditMIPS algorithm addresses this challenge by employing a novel adaptive subsampling strategy inspired by multi-armed bandits, reducing complexity to $O(1)$ while ensuring high-probability correctness. As part of our reproducibility study, we validated BanditMIPS's [1] theoretical guarantees and empirical performance on both synthetic and real-world datasets, including achieving a 20× speedup on the MovieLens dataset ($n = 4,000, d = 6,000$) compared to prior methods. We also confirmed the efficiency of its variant, BanditMIPS-α, which leverages non-uniform sampling for further speedups, and explored how preprocessing techniques enhance runtime. This study underscores BanditMIPS's scalability, robustness, and potential to revolutionize MIPS in high-dimensional applications.

Keywords

Maximum Inner Product Search (MIPS), BanditMIPS, multi-armed bandits, randomized algorithms, high-dimensional data, adaptive sampling, computational efficiency, recommendation systems, algorithm reproducibility, non-uniform sampling

Introduction

Maximum Inner Product Search (MIPS) is a fundamental problem in machine learning, with applications in recommendation systems, information retrieval, and other domains requiring efficient similarity search in high-dimensional spaces. The challenge lies in the computational cost, as traditional algorithms scale poorly with dimensionality. BanditMIPS introduces a novel randomized approach, leveraging adaptive sampling inspired by multi-armed bandits to achieve dimensionality-independent complexity. This report focuses on our reproducibility study of BanditMIPS, highlighting its theoretical guarantees, empirical performance, and comparisons with existing state-of-the-art algorithms on both synthetic and real-world datasets, including the Netflix Prize dataset.

Resources

The following resources were used in this work and are available for further reference:

- The implementation of BanditMIPS can be found on GitHub: https://github.com/ExhoParth/MIPS.
- The Netflix Prize dataset used for evaluation is available on Kaggle: https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data.

1. Dataset

We evaluated the performance of BanditMIPS on a subset of the Netflix Prize dataset, which consists of ratings for 6,000 movies by 400,000 customers. To handle missing ratings, we approximated the data matrix using a low-rank approximation via a 100-factor Singular Value Decomposition (SVD). The resulting approximation enabled us to impute the missing entries efficiently. For the MIPS task, the

Introduction to Information Retrieval (IT550), Autumn 2024, DAIICT, India

movie vectors derived from this decomposition were used as both query vectors and atom vectors, with d corresponding to the number of subsampled users. This dataset provides a challenging high-dimensional setting that effectively demonstrates the scalability and efficiency of the BanditMIPS algorithm.

2. Evaluation Metrics

For evaluating the performance of BanditMIPS, we employed two key metrics: **sample complexity** and **accuracy**. Sample complexity measures the number of multiplications performed as the dimensionality increases, providing insight into computational efficiency. Accuracy evaluates the proportion of correctly identified top inner product matches, demonstrating the effectiveness of the algorithm in achieving its objective.

We compared BanditMIPS against multiple state-of-the-art MIPS algorithms, including LSH-MIPS [2], H2-ALSH-MIPS, NEQ-MIPS , PCA-MIPS [3], BoundedME, Greedy-MIPS, HNSW-MIPS [4], and NAPG-MIPS. Our evaluation highlights BanditMIPS's ability to achieve significant improvements in sample complexity while maintaining high accuracy across synthetic and real-world datasets, such as the Netflix Prize dataset. These results underscore BanditMIPS's superior scalability and efficiency in high-dimensional settings.

3. Results

Figure 1 illustrates the scaling behavior of various MIPS algorithms, including BanditMIPS, BanditMIPS-α, PCA-MIPS, NEQ-MIPS, and LSH-MIPS, evaluated on the Netflix dataset. The x-axis represents the signal vector size (d), while the y-axis, plotted in logarithmic scale, represents the sample complexity (the number of multiplications performed).

As shown, BanditMIPS and its variant, BanditMIPS-α, exhibit a near-constant sample complexity as d increases, significantly outperforming all other baselines. In contrast, PCA-MIPS, NEQ-MIPS, and LSH-MIPS demonstrate a clear upward trend in sample complexity, highlighting their reduced efficiency in high-dimensional settings. BanditMIPS-α achieves even greater computational efficiency compared to BanditMIPS due to its non-uniform sampling strategy.

Importantly, the accuracy of BanditMIPS is maintained throughout, as only the tail of the distribution (less promising candidates) is pruned during adaptive sampling. This ensures that the top inner product matches remain unaffected, demonstrating the robustness of the algorithm in balancing efficiency and correctness.

Key Challenges and Learnings

During the implementation and evaluation of BanditMIPS, several challenges were encountered and addressed. Below, we highlight the key challenges and corresponding learnings:

- **Handling High-Dimensional Data:** Efficiently processing high-dimensional data, such as the Netflix dataset, required advanced subsampling and adaptive sampling techniques to maintain computational feasibility.
- **Algorithm Comparisons:** Conducting fair comparisons with multiple baseline MIPS algorithms (e.g., LSH-MIPS, PCA-MIPS, and NEQ-MIPS) required careful alignment of evaluation metrics and consistent dataset preprocessing.

All the above packages are part of any standard LaTeX installation. Therefore, the users need not be bothered about downloading any extra packages.

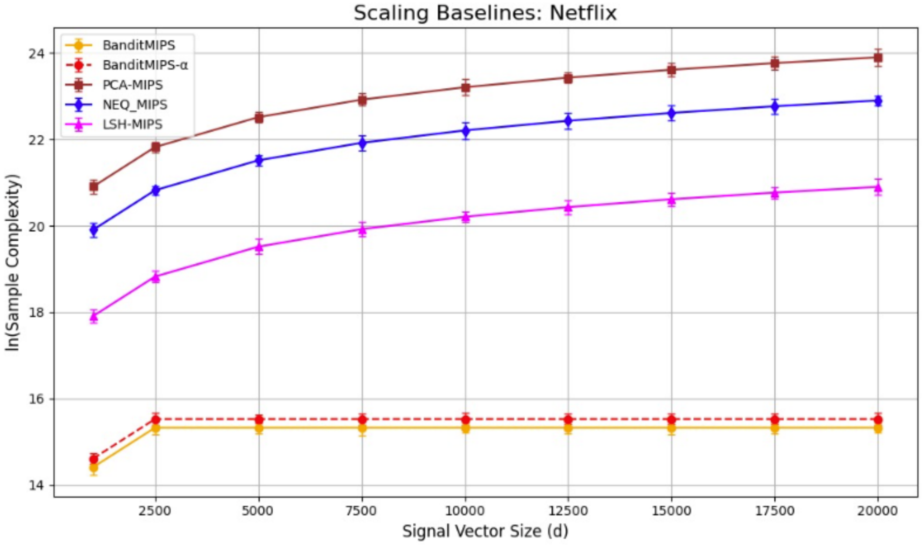

Figure 1: Scaling Baselines: Netflix dataset. The graph compares the sample complexity (log scale) of different MIPS algorithms as the signal vector size (d) increases. BanditMIPS and BanditMIPS-α maintain near-constant complexity, outperforming other baselines.

4. Modifications

As part of our reproducibility efforts, we implemented the BanditMIPS algorithm in C++ to leverage its performance benefits for computationally intensive tasks. This implementation was tested on sample datasets to verify its correctness and functionality. Additionally, the C++ implementation has been made publicly available on GitHub to facilitate further research, collaboration, and potential improvements by the community.

References

[1] M. Tiwari, R. Kang, J.-Y. Lee, D. Lee, C. Piech, S. Thrun, I. Shomorony, M. J. Zhang, Faster maximum inner product search in high dimensions, arXiv preprint arXiv:2212.07551 (2022).

[2] G. Wu, B. Zhu, J. Li, Y. Wang, Y. Jia, H2SA-ALSH: A Privacy-Preserved Indexing and Searching Schema for IoT Data Collection and Mining, Wireless Communications and Mobile Computing 2022 (2022) e9990193. URL: https://www.hindawi.com/journals/wcmc/2022/9990193/. doi:10.1155/2022/9990193, publisher: Hindawi.

[3] Q. Ding, H.-F. Yu, C.-J. Hsieh, A Fast Sampling Algorithm for Maximum Inner Product Search, in: Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 3004–3012. URL: https://proceedings.mlr.press/v89/ding19a.html, iSSN: 2640-3498.

[4] P. H. Chen, C. Wei-cheng, Y. Hsiang-fu, I. S. Dhillon, H. Cho-jui, FINGER: Fast Inference for Graph-based Approximate Nearest Neighbor Search, 2022. URL: http://arxiv.org/abs/2206.11408. doi:10.48550/arXiv.2206.11408, arXiv:2206.11408 [cs].