GPU-based Parallelization of QuickScorer to Speed-up Document Ranking with Tree Ensembles

Francesco Lettich*, Claudio Lucchese†, Franco Maria Nardini†, Salvatore Orlando*,†, Raffaele Perego†, Nicola Tonellotto†, and Rossano Venturini‡,†

*Università Ca' Foscari Venezia, Italy
†ISTI, CNR, Pisa, Italy
‡Università di Pisa, Italy

Abstract. Scoring documents with learning-to-rank (LtR) models based on large ensembles of regression trees currently represents one of the most effective solutions to rank query results returned by large scale Information Retrieval systems. However, such scoring models are very complex, and when deployed in real Web Search Engine infrastructures they are constrained within strict time budgets. This calls for very fast and efficient solutions, able to exploit all the computational resources offered by a given system. This paper investigates the opportunities offered by modern graphic cards (GPUs) to efficiently exploit LtR complex models based on trees ensembles to rank documents. To this end we propose GPUScorer, a GPU-based parallelization of the state-of-the-art algorithm QuickScorer to score documents with tree ensembles. GPUScorer takes advantage of the huge computational power of GPUs to perform tree ensemble traversal by evaluating multiple documents simultaneously. We provide a concise experimental evaluation, and show that GPUScorer is able to achieve speedups up to 32x over the sequential version of QuickScorer.

In this work we propose GPUScorer, a GPU-based parallelization of QuickScorer [1], the state-of-the-art algorithm to score documents with tree ensembles.

GPUScorer proposes a *cache-conscious* approach that builds on the *blockwise* variant of QuickScorer – where tree ensembles can be conveniently partitioned in disjoint tree-blocks – to leverage the massive computational power offered by modern GPUs. More precisely, GPUScorer is made up by a processing pipeline structured in three different phases, where data structures and operations are conveniently redesigned or rearranged to take into account the features and limitations underlying the *execution model* and *memory hierarchy* characterizing the GPUs.

In the experimental evaluation we show how our approach is able to achieve consistent speedups with respect to QuickScorer, especially when the number of leaves per tree becomes consistent (up to 32× with 64 leaves, and 31× with 32 leaves).

References

1. C. Lucchese, F. M. Nardini, S. Orlando, R. Perego, N. Tonellotto, and R. Venturini, "Quickscorer: A fast algorithm to rank documents with additive ensembles of regression trees," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR).* ACM, 2015, pp. 73–82.