authors proposed a technique based on virtualization to create a scaled-down search engine to imitate a real one. The main idea in this technique is to maintain the overall performance behavior of a real search engine while reducing the computational requirements.

4 Final Remarks

This year's workshop papers encompassed a variety of topics; some of the presented works are follow-up on well-established research issues, ranging from efficient query processing [6] to resource selection on federated environments [2] and collection statistics estimation (using snippet sampling) [5]. On perspective, map-reduce applications are gaining an increasing interest in the community, especially those that involve similarity computations [3, 4]. The map-reduce programming paradigm allows experimentation and development of novel techniques for old problems in a new framework. In general, there is opportunity for exploring topics and solutions that were complicated to scale beforehand, and therefore making large-scale distributed computation more accessible, whilst facing new interesting problems. Finally, it is clear that there is a need in academia to keep up with real data corpora sizes and to experiment with the high number of machines employed in real-world commercial scenarios, which is not feasible in most of the situations [1].

5 Sponsors

We are grateful to Yahoo! and Twitter for their sponsorship.

References

[1] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso. Performance evaluation of large-scale information retrieval systems scaling down. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 36–39. CEUR WS, 2010.

[2] A. Kulkarni and J. Callan. Topic-based index partitions for efficient and effective selective search. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 19–24. CEUR WS, 2010.

[3] G. D. F. Morales, C. Lucchese, and R. Baraglia. Scaling out all pairs similarity search with MapReduce. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 25–30. CEUR WS, 2010.

[4] A. Stupar, S. Michel, and R. Schenkel. RankReduce – processing k-nearest neighbor queries on top of MapReduce. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 13–18. CEUR WS, 2010.

[5] A. Tigelaar and D. Hiemstra. Query-based sampling using snippets. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 7–12. CEUR WS, 2010.

[6] N. Tonellotto, C. Macdonald, and I. Ounis. Efficient dynamic pruning with proximity support. In *Proceedings of the 8th Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 31–35. CEUR WS, 2010.