preserves the same theoretical properties as ICWS, but with reduced theoretical complexity in both time and space. We conduct extensive empirical tests of our PCWS algorithm and a number of state-of-the-art methods on five real-world text data sets for classification and information retrieval. The experimental results show that PCWS is able to achieve the same (even better) performance than ICWS with $1/5 \sim 1/3$ reduced empirical runtime and 20% reduced memory footprint. In the cases of large number of features, PCWS can save hundreds of GB of memory footprint, which makes it more practical in dealing with real-world data sets in the era of big data.

Existing similarity-preserving hashing techniques can only deal with nested binary sets [14] and tree-structured categorical data [4]. It will be interesting to extend CWS schemes to hash nested weighted sets, which not only encode the importance of feature but also preserve the *multi-level exchangeability* [4] of feature, in our future work.

Acknowledgment

This work is partially supported by ARC Discovery Grant DP140100545.

6. REFERENCES

[1] A. Z. Broder, M. Charikar, A. M. Frieze, and M. Mitzenmacher. Min-wise Independent Permutations. In *STOC*, pages 327–336, 1998.

[2] C.-C. Chang and C.-J. Lin. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011.

[3] M. S. Charikar. Similarity Estimation Techniques from Rounding Algorithms. In *STOC*, pages 380–388, 2002.

[4] L. Chi, B. Li, and X. Zhu. Context-preserving Hashing for Fast Text Classification. In *SDM*, pages 100–108, 2014.

[5] O. Chum, J. Philbin, A. Zisserman, et al. Near Duplicate Image Detection: Min-Hash and Tf-idf Weighting. In *BMVC*, pages 1–10, 2008.

[6] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni. Locality-sensitive Hashing Scheme Based on p-stable Sistributions. In *SOCG*, pages 253–262, 2004.

[7] E. Dumbill. A Revolution That Will Transform How We Live, Work, and Think: An Interview with the Authors of Big Data. *Big Data*, 1(2):73–77, 2013.

[8] S. Gollapudi and R. Panigrahy. Exploiting Asymmetry in Hierarchical Topic Extraction. In *CIKM*, pages 475–482, 2006.

[9] S. Gunelius. *The Data Explosion in 2014 Minute by Minute Infographic*, Jul 2014. http://aci.info/2014/07/12/the-data-explosion-in-2014-minute-by-minute-infographic.

[10] B. Haeupler, M. Manasse, and K. Talwar. Consistent Weighted Sampling Made Fast, Small, and Easy. *arXiv preprint arXiv:1410.4266*, 2014.

[11] T. H. Haveliwala, A. Gionis, and P. Indyk. Scalable Techniques for Clustering the Web. In *WebDB*, pages 129–134, 2000.

[12] P. Indyk and R. Motwani. Approximate Nearest Neighbors: towards Removing the curse of Dimensionality. In *STOC*, pages 604–613, 1998.

[13] S. Ioffe. Improved Consistent Sampling, Weighted Minhash and L1 Sketching. In *ICDM*, pages 246–255, 2010.

[14] B. Li, X. Zhu, L. Chi, and C. Zhang. Nested Subtree Hash Kernels for Large-scale Graph Classification over Streams. In *ICDM*, pages 399–408, 2012.

[15] P. Li. 0-Bit Consistent Weighted Sampling. In *KDD*, pages 665–674, 2015.

[16] P. Li and C. König. *b*-Bit Minwise Hashing. In *WWW*, pages 671–680, 2010.

[17] P. Li, A. Owen, and C.-H. Zhang. One Permutation Hashing. In *NIPS*, pages 3113–3121, 2012.

[18] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Identifying Suspicious URLs: an Application of Large-scale Online Learning. In *ICML*, pages 681–688, 2009.

[19] M. Manasse, F. McSherry, and K. Talwar. Consistent Weighted Sampling. *Unpublished technical report*, 2010.

[20] G. S. Manku, A. Jain, and A. Das Sarma. Detecting Near-duplicates for Web Crawling. In *WWW*, pages 141–150, 2007.

[21] M. Mitzenmacher, R. Pagh, and N. Pham. Efficient Estimation for High Similarities Using Odd Sketches. In *WWW*, pages 109–118, 2014.

[22] A. Rajaraman, J. D. Ullman, J. D. Ullman, and J. D. Ullman. *Mining of Massive Datasets*, volume 1. Cambridge University Press Cambridge, 2012.

[23] E. Schonfeld. *Google Processing 20,000 Terabytes A Day, And Growing*, Jan 2008. http://techcrunch.com/2008/01/09/google-processing-20000-terabytes-a-day-and-growing.

[24] A. Shrivastava. Exact Weighted Minwise Hashing in Constant Time. *arXiv preprint arXiv:1602.08393*, 2016.

[25] A. Shrivastava and P. Li. Densifying One Permutation Hashing via Rotation for Fast Near Neighbor Search. In *ICML*, pages 557–565, 2014.

[26] A. Shrivastava and P. Li. In Defense of Minhash Over SimHash. In *AISTATS*, pages 886–894, 2014.

[27] D. Sullivan. *Google Still Doing At Least 1 Trillion Searches Per Year*, Jan 2015. http://searchengineland.com/google-1-trillion-searches-per-year-212940.

[28] D. Tam. *Facebook processes more than 500 TB of data daily*, Jul 2014. http://www.cnet.com/news/facebook-processes-more-than-500-tb-of-data-daily.

[29] W. Wu, B. Li, L. Chen, and C. Zhang. Canonical Consistent Weighted Sampling for Real-Value Weighted Min-Hash. In *ICDM*, pages 1287–1292, 2016.

[30] D. Yang, B. Li, and P. Cudré-Mauroux. POIsketch: Semantic Place Labeling over User Activity Streams. In *IJCAI*, pages 2697–2703, 2016.