ISP Early Research Project Report
Consistent Weighted Sampling and Applications

Mahmud Allahverdiyev

January 25, 2019

1 Abstract

Consistent Weighted Sampling has become very popular technique for generating sketches for large volume data and continuous data streams. Originally proposed by Ioffe [4], the technique has been applied to a number of practical problems ranging from big data and similarity checking to bioinformatics applications, including rapid microbiome analytics [6]. "Count-Min Sketch" and "Histosketch" schemes successfully made use of CWS, thus driving attention of the community to further analysis and improvements. During the ISP research, a new method for deriving Count-Min sketches has been implemented and the benchmarks of the implementation can be found in the following pages of the report.

The objective of this work will be a comparison between HULK and Count-Min sketch sample by the user in offline mode. The comparisons are going to be made according to min-max similarity kernels.

2 Introduction

Consistent Weighted Sampling can be referred as a generic method for sampling processes where two CWS samples for similar or identical data records should be close to each other in some predefined metric. The method and its derivations have been successfully used in finding similar files in the system, and similar or duplicate pages on the Web [2]. The most suitable distance metrics may vary from one application to another one; yet for the use cases above and many others, Jaccard similarity presented reasonably effective performance. Recalling that, Jaccard similarity two sets A and B is $\frac{|A \cup B|}{|A \cap B|}$. Considering the fact that the sets can be unweighted or weighted, the means for defining similarity measurements is not trivial. In order to deal with weighted sets (bags), a classical approach was to replicate each element as many times as its weight, then apply traditional CWS algorithm. However, since the cardinality of the sets, thus the weights are at least theoretically unbounded, it would not be tractable in any possible way to apply this procedure.

All in all, using classical probability theory axioms, it can be shown that $Pr[sample(A) = sample(B)] = \frac{|A \cup B|}{|A \cap B|} = \frac{\sum_{key} min(A(key), B(key))}{\sum_{key} max(A(key), B(key))}$ which is theoretically equal to Jaccard similarity between A and B. The larger the size of the sample chosen from the datasets is, the above formulation yields less error on approximating Jaccard similarity. As proposed in [4], generation of one CWS sample is done via choosing three random variables with the following parameters: $r_{i,j}, c_{i,j} \sim gamma(2,1)$ and $\beta_{i,j} \sim uniform(0,1)$. Then, following operations done on each element in the universe with positive weight:

$$y_{i,j} = exp(r_{i,j}(\left\lfloor \frac{log(weight_i)}{r_{i,j}} + \beta_{i,j}) - \beta_{i,j}) \right\rfloor \tag{1}$$

Then, the sketch value is:

$$s_{i,j} = \frac{c_{i,j}}{y_{i,j} exp(r_{i,j})} \tag{2}$$

. The resulting sketch value for this particular iteration is the minimum value of $s_{i,j}$ among all the data records.

[7] introduces a framework for Histosketch generation and maintaining it during continuous update routines. If the size of the sketch is k, then the first k incoming data records are added to the sketch without making any further comparison or modifications on the structure itself. Each of the new coming elements starting from $(k+1)$-th one, is hashed in the similar way and compared with current histosketch elements and updates are done if a new local minimum hash is found. Their method also touches the well-known phenomena, concept drift which is considered to be a typical problem with sketching methods. Concept drift arises in many situations, for example, if a data source is a restaurant providing eastern cuisine, and abruptly it becomes a pizzeria, the obvious matter is that the sketch for the customer data, should be sensitive to these changes. Their method uses gradual forgetting for updating sketches in both one-by-one and batch styled input streams [7].

Recently, Li [5] proposed a simple idea of simply forgetting the weights on the generated samples after CWS scheme is applied. Applied to SVM and some other machine learning frameworks, it was empirically demonstrated that only taking account of the keys, 0-bit CWS can still produce almost exactly same results compared to classical methods. Moreover, this scheme is expected to yield more extensive usage of CWS hashing into large-scale learning problems.

3 Related work

During the ISP period, Count-Min Sketch has been implemented [1] in C++ 14. Using functional programming features of the language, the sketch scheme was produced. The performance of the implementation has been checked over both artificial and real data sets of element streams with varying sizes. To generate artificial inputs, a small and fast library implemented which makes use of pseudo-random number generator and outputs data sets according to

the adjusted parameters, including the stream size, the real similarity value between the sets, and the number of distinct streams. As the second task, CWS was implemented over Count-Min in order to realize the comparison of HULK and the implemented CWS solution. Since HULK has the capability of setting the parameters of sketching procedure automatically, we manually configured our solution on distinct data set comparisons.

In the original Count-Min Sketch paper [3], the authors proposed the dimensions of the sketch matrix, namely, its width and depth as:

$$width = \left\lceil \frac{e}{error} \right\rceil \tag{3}$$

$$depth = \left\lceil -\ln \frac{1}{1 - certainty} \right\rceil \tag{4}$$

In practice, empirically, it has been shown that if the natural exponent and the base of the logarithm is to changed to 2, then there happens performance increase in several percents of magnitude. The artificial data sets have been queried with tuning of both type of parameters. The implementation of Count-Min sketch has been tested over datasets with varying parameters: key universe size, confidence level indicator for the table, error bound indicator for the table, and incoming stream size. The results are reasonably satisfatory; i.e. even for $confidence = 0.5$ and $tolerance = 1.0$, the mean absolute error was less than 0.35 which is reasonably sufficient for further sketching and other applications. The benchmarks of all the tests done, can be found at [1].

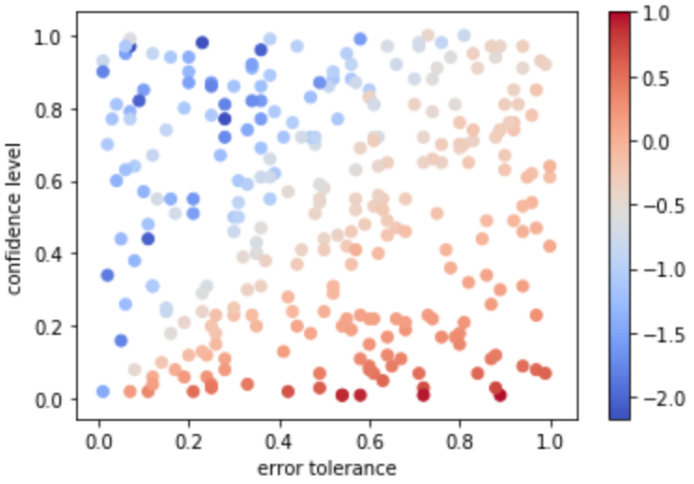

Figure 1: MAE's in log scale

4 Conclusion

The work showed that CWS over Count-Min Sketch can be a good candidate for estimating similarity measures between both binary and weighted sets. The experiments and observations indicated that this generic method result in fairly good performance. However, parameterization of the model is a must to outperform the classical approaches. The continuation of the work will be devoted to improve the effectiveness and efficiency of the implementation and method itself.

References

[1] Mahmud Allahverdiyev. Count-min sketch. January 2019. `https://github.com/tr0j4n034/Count-Min-Sketch`.

[2] Andrei Z. Broder, Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. Syntactic clustering of the Web. *Computer Networks and ISDN Systems*, 1997.

[3] Graham Cormode and S. Muthukrishnan. An improved data stream summary: The count-min sketch and its applications. *Journal of Algorithms*, 2005.

[4] Sergey Ioffe. Improved consistent sampling, Weighted Minhash and L1 Sketching. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2010.

[5] Ping Li. 0-Bit Consistent Weighted Sampling. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, 2015.

[6] Will PM Rowe, Anna Paola Carrieri, Cristina Alcon-Giner, Shabhonam Caim, Alex Shaw, Kathleen Sim, J Simon Kroll, Lindsay Hall, Edward O Pyzer-Knapp, and Martyn D Winn. Streaming histogram sketching for rapid microbiome analytics. *bioRxiv*, 2018.

[7] Dingqi Yang, Bin Li, Laura Rettig, and Philippe Cudre-Mauroux. HistoSketch: Fast similarity-preserving sketching of streaming histograms with concept drift. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2017.