Errata for "SpaceSaving±: An Optimal Algorithm for Frequency Estimation and Frequent Items in the Bounded-Deletion Model"

Fuheng Zhao
UC Santa Barbara
fuheng_zhao@ucsb.edu

Divyakant Agrawal
UC Santa Barbara
agrawal@cs.ucsb.edu

Amr El Abbadi
UC Santa Barbara
amr@cs.ucsb.edu

Ahmed Metwally
Uber, Inc.
ametwally@uber.com

Claire Mathieu
CNRS and IRIF
clairemmathieu@gmail.com

Michel de Rougemont
University Paris II and IRIF
m.derougemont@gmail.com

ABSTRACT

This errata article points out an implicit assumption in the work of four of us published in VLDB 2022. The SpaceSaving± algorithm in bounded deletion data stream presented in the paper implicitly assumed deletions happen after all insertions. When insertions and deletions are interleaved, that algorithm may severely underestimate item's frequency. We first illustrate this phenomenon by an example and then present a modified algorithm with minor changes to allow interleaving between insertions and deletions. We also include a pointer to a full analysis of the new algorithms.

PVLDB Reference Format:
Fuheng Zhao, Divyakant Agrawal, Amr El Abbadi, Ahmed Metwally, Claire Mathieu, and Michel de Rougemont. Errata for "SpaceSaving±: An Optimal Algorithm for Frequency Estimation and Frequent Items in the Bounded-Deletion Model" . PVLDB, 17(4): 643 - 643, 2023.
doi:10.14778/3636218.3636221

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/ZhaoFuheng/SpaceSavingBoundedDeletionModel.

Algorithm. Recall the SpaceSaving± summary with lazy delete described in Algorithm 3 of [3]: the insertions follow the steps described in [1]; for deletions, the algorithm decrements the deleted item's count if it is monitored in the summary, and ignores the deletion otherwise. We first discuss the issue and then include a pointer to the fix.

Counter Example: The problem occurs when the inherited count during insertion is not monotoned. Let $\alpha \sim 3/2$, let k be an even number, and consider a SpaceSaving± summary with ϵ set to $\frac{3}{2k}$ which lead to k entries. The correctness of frequency estimation requires: $\forall x \in U, |\hat{f}(x) - f(x)| < \epsilon(I-D)$, where I (resp. D) denotes the total number of insertions (resp. deletions).

Assume items are drawn from a universe $\{a_1, \ldots a_{k+1}\}$. We construct a bounded deletion stream in which a_1, the most frequent item in the stream, is not in the final summary. Here is the stream, where a_i (resp. $-a_i$) denotes an insertion (resp. deletion) of a_i:

$$a_2 a_3 \cdots a_{k+1} (a_1 a_2 \cdots a_{k+1})^k [(-a_i)^{1+k/2} a_1^{k/2} a_i]_{i=2,3,\ldots,k+1}.$$

Analysis. The total number of insertions is $I \sim (3/2)k^2$ and the total number of deletions is $D \sim k^2/2$, so this is a valid bounded deletions stream: $D \leq (1-1/\alpha)I$ with $\alpha \sim 3/2$. The frequency of a_1 is $f(a_1) \sim k^2/2$, and for $i \geq 2$ the frequency of a_i is $f(a_i) = 1 + k/2$. The most frequent item is a_1 by far. Correctness of the algorithm requires the frequency estimations to have an additive error of at most $\epsilon(I-D) \sim \epsilon k^2 \sim 3k/2$.

In SpaceSaving±, after the first $k + k(k+1)$ operations, the summary contains the following items and counts: $S = \{(a_2, k+2), (a_3, k+2), \ldots, (a_{k+1}, k+2)\}$. The $1 + k/2$ deletions of a_2 bring a_2's count down to $(k+2)/2$. As a result, when a_1 arrives, a_2 is evicted and a_1 inherits its count. After processing the $k/2$ insertions on a_1, a_1's count is $k+1$, the minimum count, the last insertion of a_2 replaces a_1, and the summary doesn't contain a_1. Similarly, for every substream $(-a_i)^{1+k/2} a_1^{k/2} a_i$ for $i = 3, 4, \ldots, k+1$, the summary starts and ends will not contain a_1.

At the end, a_1 is absent from the summary. We know $\hat{f}(a_1) = 0$ and $f(a_1) \sim (3/2)k^2/2 > \epsilon k^2$. Hence, SpaceSaving± doesn't guarantee to solve the frequency estimation problem when operations are interleaved. We note that this stream is also a counterexample for the non-lazy version of SpaceSaving± proposed in [3].

Corrected algorithm. We now outline how to fix the algorithm so that all the results from [3] (frequency estimation, frequent items, and top-k problems in the bounded deletion model with $O(\frac{\alpha}{\epsilon})$ space) hold for the new algorithm.

Assume an item x is evicted at time t with a minimum count of c. To avoid underestimation, x must inherit a count no less then c when x is inserted again at time $t' > t$. To ensure that, we separate the count into two fields: an *insert count* and a *delete count*. Deletions will only affect the delete count. The estimated frequency of an item is then the difference between the insert and the delete counts. See [2] for details.

REFERENCES

[1] Ahmed Metwally, Divyakant Agrawal, and Amr El Abbadi. 2005. Efficient computation of frequent and top-k elements in data streams. In *International conference on database theory*. Springer, 398–412.
[2] Fuheng Zhao, Divyakant Agrawal, Amr El Abbadi, Claire Mathieu, Ahmed Metwally, and Michel de Rougemont. 2023. A Detailed Analysis of the SpaceSaving± Family of Algorithms with Bounded Deletions. arXiv:2309.12623 [cs.DB]
[3] Fuheng Zhao, Divyakant Agrawal, Amr El Abbadi, and Ahmed Metwally. 2021. SpaceSaving±: An Optimal Algorithm for Frequency Estimation and Frequent items in the Bounded Deletion Model. *arXiv preprint arXiv:2112.03462* (2021).

Proceedings of the VLDB Endowment, Vol. 17, No. 4 ISSN 2150-8097.
doi:10.14778/3636218.3636221