A-021

データストリームに対する頻出値問題を解くアルゴリズムの実証実験
Comparative experiments on algorithms for the frequent items problem over data streams

鳥谷部　直弥 *　　谷　陽太 *　　喜田　拓也 *
Naoya Toriyabe　　Yota Tani　　Takuya Kida

1. はじめに

データストリームに対する重要な問題の一つに，**頻出アイテム問題**（**頻出値問題**）がある．いま，ある集合 Σ の要素（**アイテム**）が絶え間なく入力されるデータストリームを考える．また，Σ の各要素は自然数に一対一に写像されるものとする．このとき，頻出値問題とは，与えられた閾値以上の頻度でデータストリーム中に出現する自然数を見つけ出すことである．データストリームの性質上，すべての値の出現回数を保持するのは多くの記憶容量を必要とするため，頻出値問題を厳密に解くことは困難である．したがって，ここでは，出力に偽陽性の解が含まれることを許容する問題を考える．

Cormode と Hadjieleftheriou [1] によると，頻出値問題は二種類の問題設定（ϕ 頻出値問題と ϵ 近似頻出値問題）に大別され，既存手法は三つの方策（計数法，分位数法，スケッチ法）に分類されると述べられている．本稿では，それらのうち計数法の手法である k-reduced Bag 法 [4] と Space Saving 法 [3]，Lossy Counting 法 [2] の三つの手法に注目する．元々は，前者二つは ϕ 頻出値問題に対して提案されたものであり，最後の一つは ϵ 近似頻出値問題に対して提案されたものである．そこで，本稿では，これら三つの手法が両方の問題に適用可能であることを示し，それぞれの問題に対する性能比較を行った．

2. 準備

定義 1 (ϕ 頻出値集合)**.** 値 i に対して，その頻度を f_i とする．ϕ **頻出値集合**とは，N 個の値と閾値 $\phi\,(0<\phi<1)$ を与えたとき，集合 $A=\{i\,|\,f_i>\phi N\}$ に対して，条件 $i\in A\Rightarrow i\in\mathcal{S}$ を満たす集合 $\mathcal{S}$ のことをいう．

定義 2 (ϵ 近似頻出値集合)**.** 値 i に対して，その頻度を f_i とする．ϵ **近似頻出値集合**とは，N 個の値と閾値 $\phi\,(0<\phi<1)$, エラーパラメータ $\epsilon\,(0<\epsilon<\frac{\phi}{2})$ を与えたとき, 集合 $A=\{i\,|\,f_i>\phi N\}, B=\{i\,|\,f_i<(\phi-\epsilon)N\}$ に対して，条件 $(i\in A\Rightarrow i\in\mathcal{S})\wedge(i\in B\Rightarrow i\notin\mathcal{S})$ を満たす集合 $\mathcal{S}$ である．

ϕ 頻出値集合を出力する問題を ϕ **頻出値問題**といい，ϵ 近似頻出値集合を出力する問題を ϵ **近似頻出値問題**という．

次に，値の推定頻度のみを用いて，ϵ 近似頻出値集合を求める手法を説明する．まず，Manku と Motwani の論文 [2] の補題 4.4 より，以下の定理 1 が成り立つ．

定理 1. 値 i に対して，その頻度を f_i，推定頻度を $\hat{f}_i$ とする．すべての値に対して，$\hat{f}_i\le f_i\le\hat{f}_i+\epsilon N$ が成り立つとき，$\mathcal{S}'=\{i\,|\,\hat{f}_i>(\phi-\epsilon)N\}$ は ϵ 近似頻出値集合である．

Metwally らの論文 [3] の補題 2 より，以下の定理 2 を導くことができる．

定理 2. 値 i に対して，その頻度を f_i，推定頻度を $\hat{f}_i$ とする．すべての値に対して，$\hat{f}_i-\epsilon N\le f_i\le\hat{f}_i$ が成り立つとき，$\mathcal{S}'=\{i\,|\,\hat{f}_i>\phi N\}$ は ϵ 近似頻出値集合である．

3. 頻出値問題を解く手法

ϵ 近似頻出値問題に対する手法を ϕ 頻出値問題に適用するためには $\epsilon=0$ とすればよい．一方，ϕ 頻出値問題に対する手法を ϵ 近似頻出値問題に適用するためには，まず，頻度が ϵN より大きい値を必ず含むように ϵ 頻出値集合を求めることを考える．このとき，値の頻度と推定頻度との差が ϵN で抑えられるならば，定理 1 または定理 2 より，ϵ 近似頻出値集合を得ることができる．紙数の都合上，証明は省略するが，k-reduced Bag 法も Space Saving 法もこの条件を満たしている．

3.1 k-reduced Bag 法

この手法は，$k=\lceil\frac{1}{\phi}\rceil$ に対して，$k-1$ 個の値 i とその推定頻度 $\hat{f}_i$ の組 $(i,\hat{f}_i)$ を管理する．

データストリームから値 i が入力されたとき，$(i,\hat{f}_i)$ が保持されているならば，その推定頻度に 1 を加える．そうでない場合，新たに組 $(i,1)$ を保持する．ただし，管理している組の個数が $k-1$ であった場合には，管理しているすべての組の推定頻度を 1 ずつ減らし，推定頻度が 0 になったものを管理しているデータ構造から削除する．N 個の値を入力するまでに行われる削除操作の回数は，高々 $\lfloor\frac{N}{k}\rfloor$ 回である [4].

3.2 Lossy Counting 法

この手法は，値 i とその推定頻度 $\hat{f}_i$，値登録時における入力済みの値の数を n としたとき，$\Delta_i=\lfloor\phi n\rfloor$ の値の組 $(i,\hat{f}_i,\Delta_i)$ を管理する．

データストリームから値 i が入力されたとき，$(i,\hat{f}_i,\Delta_i)$ が保持されているならば，その推定頻度に 1 を加える．そうでない場合，新たに組 $(i,1,\Delta_i)$ を保持する．また，入力された値の個数が $\frac{1}{\phi}$ の倍数になったとき，$\hat{f}_i$ と Δ_i の和がその時点での Δ 値よりも小さい組を，管理しているデータ構造から削除する．ここで，Manku と Motwani の論文 [2] の補題 4.4 より，任意の値 i に対して $\hat{f}\le f\le\hat{f}+\phi N$ が成り立つ．

3.3 Space Saving 法

この手法は，$k=\lceil\frac{1}{\phi}\rceil$ に対して，$k-1$ 個の値 i とその推定頻度 $\hat{f}_i$ の組 $(i,\hat{f}_i)$ を管理する．

* 北海道大学， Hokkaido University

データストリームから値 i が入力されたとき，$(i, \hat{f}_i)$ が保持されているならば，その推定頻度に 1 を加える．そうでない場合，新たに組 $(i, 1)$ を保持する．ただし，管理している組の個数が k であった場合には，推定頻度が一番小さい組を管理しているデータ構造から削除し，新たな組として $(i, \hat{f}_{\min} + 1)$ を保持する．ここで，$\hat{f}_{\min}$ は削除された値の推定頻度である．Metwally らの論文 [3] の補題 2 より，保持されている推定頻度の最小値は高々 $\lfloor \frac{N}{k} \rfloor$ である．保持する組を入れ替える操作によって生じる真の頻度と推定頻度の差はこの最小値に等しいので，任意の値に対して $\hat{f} - \phi N \leq f \leq \hat{f}$ が成り立つ．

4. 実験

4.1 データ

データとして，Zipf 則に基づく人工データを用いた．人工データのデータサイズは 1000 万，$|\Sigma|$ は 100 万とした．データの偏りを示す指標となる Zipf 則のパラメータ s は，0.5 から 2.5 まで 0.5 刻みで変化させた．s の値が小さいときはデータは一様に近く，s の値が大きいときはデータの偏りが大きい．ϕ と ϵ はそれぞれ $\phi = 0.003$，$\epsilon = 0.001$ とした．

4.2 方法

ϕ 頻出値問題と ϵ 近似頻出値問題の両方に対して，前節で説明した三つの手法を実装し，各データに適用した．実装言語には C++ を用いた．値を管理するデータ構造としてはチェイン法を用いたハッシュ表を採用した．実行環境として，OS は ubuntu 18.04 LTS，g++ コンパイラ (g++ 7.3.0) を用いた．実験では，ϕ 頻出値問題に対して，各手法の実行時間を求めた．また，ϵ 近似頻出値問題に対して，各手法の実行時間と適合率を求めた．

4.3 結果と考察

表 1 に，ϕ 頻出値問題と ϵ 近似頻出値問題を三つの手法を用いて解いた場合の実行時間を示す．ここで，表の KRB, LC, SS はそれぞれ，k-reduced Bag 法，Lossy Counting 法，Space Saving 法を指す．s を大きくした場合，すべての手法で実行時間が小さくなった．これは，すでにデータ構造に保持されている値の頻度を増やす回数が多くなり，値の削除及び入れ替え操作の回数が，相対的に少なるためであると考えられる．s が小さいとき，Space Saving 法は他の手法と比べて実行時間が大きくなった．これは，一度の削除操作で複数の組を削除する可能性がある k-reduced Bag 法と Lossy Counting 法に対して，Space Saving 法が一度の操作で一つの値の組しか入れ替えないという性質による．s が小さいとき，データ構造内に新たな値が入力される回数が増えるため，値の削除及び入れ替え操作の回数が増える．Space Saving 法の性質から，s が小さいことによる操作回数の増加が他の手法と比べて大きくなる．したがって，実行時間が大きくなる．

表 2 に ϵ 近似頻出値問題を三つの手法を用いて解いた場合の出力の個数を示す．各手法の再現率はすべて 1 である．そのため，適合率は出力の個数が正解数に近いほど高くなる．Space Saving 法の適合率は，他の二つの手法と比べて，s によらず高い値になった．ここで，Space Saving 法は定理 2 に基づく手法である．定理 2 に基づく手法は，出力する値の頻度と推定頻度の差が小さいほど適合率が高くなる．Space Saving 法は，出力する値の頻度が高い手法であり，推定頻度との差が小さくなりやすい．したがって，Space Saving 法の適合率は他の手法と比べて高くなる．

表 1: Zipf 則パラメータ s に対する実行時間 (ms)

s	ϕ 頻出問題			ϵ 近似頻出問題		
	KRB	LC	SS	KRB	LC	SS
0.5	1368	1638	107829	1382	1632	109302
1.0	1320	1454	62034	1305	1477	57179
1.5	943	977	2930	945	968	3947
2.0	885	901	945	881	901	980
2.5	855	893	860	860	888	889

表 2: ϵ 近似頻出値問題における Zipf 則パラメータ s に対する各手法の出力の個数

s	正解数	KRB	LC	SS
0.5	0	0	0	0
1.0	22	25	34	22
1.5	25	32	33	25
2.0	14	17	17	14
2.5	9	10	10	9

5. おわりに

本稿では，k-reduced Bag 法 [4] と Lossy Counting 法 [2]，Space Saving 法 [3] の三つの手法が ϕ 頻出値問題と ϵ 近似頻出値問題の両方に適用可能であることを示し，それぞれの問題に対する性能比較を行った．

今後の課題として，計数法を他の方策（分位数法，スケッチ法）に属する手法と比較することや，適合率が高く，データの偏りが小さい場合にも高速で動く手法の開発が挙げられる．手法の開発にあたり，ϕ 頻出値問題や ϵ 近似頻出値問題を解く確率的手法や，近似的手法に関しても考慮していく必要がある．

参考文献

[1] Graham Cormode and Marios Hadjieleftheriou. Methods for finding frequent items in data streams. *The VLDB Journal*, 19(1):3–20, 2010.

[2] Gurmeet Singh Manku and Rajeev Motwani. Approximate frequency counts over data streams. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 346–357, 2002.

[3] Ahmed Metwally, Divyakant Agrawal, and Amr El Abbadi. Efficient computation of frequent and top-k elements in data streams. In *International Conference on Database Theory*, pages 398–412. Springer, 2005.

[4] Jayadev Misra and David Gries. Finding repeated elements. *Science of computer programming*, 2(2):143–152, 1982.