情報処理学会研究報告
IPSJ SIG Technical Report

頻出時系列発見近似ストリームアルゴリズムとそのデータスクリーニングへの応用について

岡本 敦[†1] 鶴田 悠[†1] 正代隆義[†2]

ストリームデータに頻出する長さ 2 の時系列を列挙する近似ストリームアルゴリズムを提案し，このアルゴリズムが頻出アイテムを列挙する際のストリームアルゴリズムの近似基準であるイプシロン劣シノプスを満たすことを示す．さらに，提案したストリームアルゴリズムの有効性を示すために，インターネットログに頻出する時系列を抽出する実験と抽出した時系列によるインターネットログのスクリーニング実験を行ったので，その結果を報告する．

A Streaming Algorithm for Finding Frequent Sequences and Its Application to Network Traffic Screening

Atsushi Okamoto,[†1] Hisashi Tsuruta[†1]
and Takayoshi Shoudai[†2]

We propose a streaming algorithm for finding frequent sequences of length 2, and show that the proposed algorithm provides an epsilon-deficient synopsis, which is a criteria of frequency counts for items that have minimum support. Moreover, to show an effectiveness of our algorithm, we report an experimental result obtained by applying it to internet traffic data observed in darknet.

†1 九州大学大学院システム情報科学府情報学専攻
Department of Informatics, Kyushu University
†2 九州大学大学院システム情報科学研究院情報学部門
Department of Informatics, Kyushu University

1. 序 論

急速なインターネットの普及と高速化により，メモリ機能の向上よりもデータサイズの増大がより顕著となり，少ないメモリ領域で巨大なデータを高速に処理する技術が重要になってきている．そのような巨大データの処理に関する研究としてストリームアルゴリズム理論がある．ストリームアルゴリズムは履歴を全て記憶するわけではなく，必要な統計情報のみを蓄えて処理していく．

ストリームアルゴリズムを最初に形式化したのは N. Alon ら[1] の研究である．彼らはデータの出現数の頻度分布を計算するストリームアルゴリズムを提案した．R. M. Karp ら[2] は，ストリームデータ中で出現割合があらかじめ定められた γ 以上の頻出イベントを出力するストリームアルゴリズムを提案した．彼らのアルゴリズムは，使用するメモリ量が $1/\gamma$ で抑えられ，その計算時間はデータ量の線形時間である．また，G.S.Manku と R.Motwani[3] は，メモリサイズに制約がある環境ですべての頻出イベントとそれらの頻度を計算する手法として，誤り許容カウント法 (*lossy count method*) を提案した．彼らのアルゴリズムは頻出データの誤り具合の基準であるイプシロン劣シノプス (*ϵ-deficient synopsis*) を保持する．

ストリームアルゴリズムが必要とされる例として，インターネット上の脅威を早期に検知するためのインターネットログのオンライン解析があげられる．鶴田ら[5] は，ありふれた不正パケットを時系列パケットパタンで表現し，頻出する時系列パケットパタンに照合するパケットを削除することで，より悪質な不正パケットをあぶりだすデータスクリーニング手法を提案した．この手法はオフラインで保存されているデータに対して非常に効果的であるが，ストリームデータに対して単純に時系列パタンを保持・修正するには膨大なメモリが必要であり，また計算時間が急激に増大する．

本論文では，ストリームデータが与えられたときに，2 つのイベントが一定の時間差で頻出する長さ 2 の時系列を全て出力するストリームアルゴリズムを提案する．また，このアルゴリズムがイプシロン劣シノプスを保持することを証明する．さらに，実際のネットワークログに対して本アルゴリズムを適用し，長さ 2 の近似頻出時系列を用いたデータスクリーニングの実験結果について述べる．

2. 準 備

本章では，本論文で用いる用語の定義とアルゴリズムが目指す目標について述べる．

2.1 ストリームデータ

有限個の**イベント**の集合を $\mathcal{E}$ とする．以降，$\mathcal{E}$ に含まれるイベントを e, e' または $e_1, e_2, \ldots$ などと表す．時刻 t とイベント e の組 (e,t) を**事象**と呼ぶ．事象 (e,t) の集合を**ストリームデータ**とよぶ．ストリームデータ中の事象はすべて時刻の昇順に出現するものとする．ある 2 つの異なる事象 $a=(e,t)$ と $a'=(e',t')$ $(t<t')$ の時間差 $t'-t$ が T 以内 $(T>0)$ であるとき，時系列 (e,T,e') が**出現**したと言い，時系列 (e,T,e') を $(2,T)$**-時系列**とよぶ．次にストリームデータ $\mathcal{S}$ における $(2,T)$-時系列の出現回数を定義する．

定義 1 $(e_1,t_1),(e_2,t_2),(e_3,t_3)$ $(t_1<t_2<t_3)$ をストリームデータ $\mathcal{S}$ 中の 3 つの事象とする．(e,T,e') を $(2,T)$-時系列とする．次のいずれかの条件を満たすとき，$(2,T)$-時系列 (e,T,e') は 3 つの事象 $(e_1,t_1),(e_2,t_2),(e_3,t_3)$ で**重複**しているという．

(1) $e\neq e'$ かつ $e_1=e_2=e$ かつ $e_3=e'$ であり，かつ $t_3-t_1\leq T$.

(2) $e\neq e'$ かつ $e_1=e$ かつ $e_2=e_3=e'$ であり，かつ $t_3-t_1\leq T$.

(3) $e_1=e_2=e_3=e=e'$ であり，$t_2-t_1\leq T$ かつ $t_3-t_2\leq T$.

定義 2 $(e_1,t_1),(e_2,t_2),(e_3,t_3),(e_4,t_4)$ $(t_1<t_2<t_3<t_4)$ をストリームデータ $\mathcal{S}$ 中の 4 つの事象とする．(e,T,e') を $(2,T)$-時系列とする．次の条件を満たすとき，$(2,T)$-時系列 (e,T,e') は 4 つの事象 $(e_1,t_1),(e_2,t_2),(e_3,t_3),(e_4,t_4)$ で**跳越**しているという．

(1) $e_1=e_2=e$ かつ $e_3=e_4=e'$ であり，かつ $t_3-t_1\leq T$ かつ $t_4-t_2\leq T$.

定義 3 $\mathcal{S}$ をストリームデータとする．また，e と e' を異なるイベント $(e\neq e')$ とし，(e,T,e') を $(2,T)$-時系列とする．$\mathcal{S}[(e,T,e')]$ を次のように定める:

$$\mathcal{S}[(e,T,e')]=\{((e,t),(e',t'))\mid (e,t)\in\mathcal{S},(e',t')\in\mathcal{S},t<t',\text{かつ } t'-t\leq T\}.$$

$(2,T)$-時系列 (e,T,e') の**出現回数**を，次の条件を満たす $\mathcal{S}[(e,T,e')]$ の部分集合 $\mathcal{S}'$ のうち要素数最大のものの要素数とする．

(1) $\mathcal{S}'$ のどの 2 つの対 $((e_1,t_1),(e_2,t_2)),((e_3,t_3),(e_4,t_4))$ に現れる 4 つの事象は重複する 3 つの事象ではない．

(2) $\mathcal{S}'$ のどの 2 つの対 $((e_1,t_1),(e_2,t_2)),((e_3,t_3),(e_4,t_4))$ に現れる 4 つの事象は超越する 4 つの事象ではない．

下記にストリームデータ $\mathcal{S}$ に対し出現回数を計算するアルゴリズムを与える．

アルゴリズム 1

(e,T,e') を $(2,T)$-時系列とする．ストリームデータ $\mathcal{S}$ 中に現れる事象 (e',t') を時刻 t' が小さい順に読んでいき，その事象 1 つ 1 つに対して過去に時間差 T 以内で現れた事象 (e,t) のイベント e と組み合わせて時系列 (e,T,e') が出現したと見なす．重複を防ぐためにイベント x を引数に持つ $t_L(x)$ と時刻 τ を引数に持つ $\Omega(\tau)$ を用意する．t_L は引数のイベントが最後に現れた時刻を記憶し，Ω は引数の時刻の事象が同種のイベントからなる時系列 (x,T,x) に使われたかどうかを $true$ か $false$ で判断する．使われていたら $true$ で，使われていなければ $false$ である．任意の時刻 τ に対して，$\Omega(\tau)=false$，任意のイベント x に対して $t_L(x)=-T-1$ と初期化する．具体的にはある事象 (e',t') に対し，

(1) $e\neq e'$ となるイベント e と組み合わせる場合，

(a) $t'-T\leq t_L(e)$ (時間差判定)

(b) $t_L(e')<t_L(e)$ (重複判定)

を満たせば (e,T,e') が出現したと見なし，$(e,t_L(e))$ と (e',t') を (e,T,e') が出現した事象のペアとする．(このとき $t_L(e')$ は組み合わせ元の事象 (e',t') を除いたうちでの最後に現れた時刻である．事象 (e',t') による (y,T,e') となりうるイベント y と全て組み合わせを試すまで $t_L(e)$ は更新しない．)

(2) イベント e' と組み合わせる場合，

(a) $t'-T\leq t_L(e')$ (時間差判定)

(b) $\Omega(t_L(e'))=false$ (重複判定)

を満たせば時系列 (e',T,e') が出現したと見なし，$\Omega(t_L(e'))$，$\Omega(t')$ にそれぞれ $true$ を代入する．さらに，$(e,t_L(e))$ と (e',t') を (e,T,e') が出現した事象のペアとする．

ある事象 (e',t') に対するイベントの組み合わせが全て終わった後，$t_L(e')$ に t' を代入し次の事象へと読み進む．全ての時系列 (e,T,e') に対し出現したとされた回数をカウントし，出力要請時点で時系列とその出現回数を出力する．

補題 1 ストリームデータ $\mathcal{S}$ と任意の $(2,T)$-時系列 (e,T,e') に対して，アルゴリズム 1 は，$\mathcal{S}$ に現れる (e,T,e') の出現回数を正しく出力する．

証明．ストリームデータの 1 番目の事象から k 番目の事象までのデータを $\mathcal{S}_k$ とし，$\mathcal{S}_k$ 中での時系列 (e,T,e') の出現回数の定義 (定義 3) における集合 $\mathcal{S}'$ のうち要素数最大のものを $\mathcal{S}_k[(e,T,e')]$ とする．また，ストリームデータの 1 番目の事象から k 番目の事象までの時系列 (e,T,e') の出現回数について，アルゴリズム 1 により得られるイベント e と e' を含む事象のペアの集合を $\mathcal{A}_k[(e,T,e')]$ とする．次の 2 つの主張を k に関する数学的帰納法で示すことにより，本補題を得る．

主張 1. $e\neq e'$ のとき，$|\mathcal{S}_k[(e,T,e')]|=|\mathcal{A}_k[(e,T,e')]|$ である．

証明. $k+1$ 番目の事象が (e',t') であるとして一般性を失わない．$k+1$ 番目のデータを読

んだ時点での $(e, t_L(e))$ がアルゴリズム 1 においてペアとなる事象 (e', t'') を持たない場合は，$(e, t_L(e))$ と (e', t') とペアを組むので，$|\mathcal{A}_{k+1}[(e,T,e')]| = |\mathcal{A}_k[(e,T,e')]| + 1$ となり，題意は満たされる．従って，$(e, t_L(e))$ がアルゴリズム 1 でイベント e' のある事象とペアを作っているにもかかわらず，$|\mathcal{A}_{k+1}[(e,T,e')]| < |\mathcal{S}_{k+1}[(e,T,e')]|$ となる $\mathcal{S}_{k+1}[(e,T,e')]$ が存在するとして矛盾を導く．$\mathcal{S}_{k+1}[(e,T,e')]$ には (e', t') がある事象 (e, t) とペアとなって含まれている．$\mathcal{S}_{k+1}[(e,T,e')]$ から $((e,t),(e',t'))$ を削除した集合は $\mathcal{S}_k[(e,T,e')]$ となる集合のひとつである．帰納法の仮定より $|\mathcal{S}_k[(e,T,e')]| = |\mathcal{A}_k[(e,T,e')]|$ である．$\mathcal{S}_k[(e,T,e')]$ には，(e,t) をペアとなる事象は含まれていない．$t_L(e)$ の定義より，$t \le t_L(e)$ である．もし，$(e, t_L(e))$ が $\mathcal{S}_{k+1}[(e,T,e')]$ で事象 (e', t'') $(t_L(e) < t'' < t')$ とペアになっていれば，$(e,t),(e,t_L(e)),(e',t''),(e',t')$ は $\mathcal{S}_{k+1}[(e,T,e')]$ で重複または跳越をつくるので，$\mathcal{S}_k[(e,T,e')]$ では $t_L(e) < t'' < t'$ である事象 (e', t'') は $(e, t_L(e))$ とペアを作らない．すなわち，もし $t_L(e) < t'' < t'$ にイベント e' の事象 (e', t'') があれば，$\mathcal{S}_k[(e,T,e')]$ で (e', t'') と (e,t) はペアを作ることができる．これは，$|\mathcal{S}_k[(e,T,e')]| < |\mathcal{S}_{k+1}[(e,T,e')]|$ であることに矛盾する．したがって，$t_L(e) < t'' < t'$ に (e', t'') となる事象は存在しない．これは，$\mathcal{A}_k[(e,T,e')]$ で $(e, t_L(e))$ がペアを作っていることに矛盾する．したがって，$|\mathcal{A}_{k+1}[(e,T,e')]| = |\mathcal{S}_{k+1}[(e,T,e')]|$ である．(主張 1 の証明終)

主張 2. $e = e'$ のとき，$|\mathcal{S}_k[(e,T,e)]| = |\mathcal{A}_k[(e,T,e)]|$ である．

証明. $k+1$ 番目の事象を (e, t') として一般性を失わない．k 番目の事象までにイベント e の現れる時刻の列を $T = (t_1, t_2, \ldots, t_m)$ とする．次の条件を満たすように $T = (t_1, t_2, \ldots, t_m)$ を部分列に分割する．

- $T_i = (t_{i,1}, t_{i,2}, \ldots, t_{i,m_i})$，ただし，$|t_{i,\ell+1} - t_{i,\ell}| \le T$ $(1 \le \ell \le m_i - 1)$ かつ $|t_{i+1,1} - t_{i,m_i}| > T$ $(i \ge 1)$.

この部分列の数を K とおく．ここで，$\sum_{i\ge1}^{K} m_i = m$ である．明らかに $|\mathcal{S}_k[(e,T,e)]| = \sum_{i\ge1} \lfloor m_i/2 \rfloor$ が成り立つ．各部分列 T_i $(1 \le i \le K)$ の定義より，$t_L(e) = t_{K,m_K}$ である．もし，(e, t_{K,m_K}) がその K 番目の部分列で奇数番目の事象であれば，$\Omega(t_{K,m_K}) = false$ であり，偶数番目ならば $\Omega(t_{K,m_K}) = true$ である．$|t' - t_{K,m_K}| \le T$ ならば，$\Omega(t_{K,m_K}) = false$ のとき，(e, t_{K,m_K}) と (e, t') はペアを作ることができ，このが $\mathcal{A}_k[(e,T,e)]$ から最大の $\mathcal{S}_{k+1}[(e,T,e)]$ のひとつを作ることは明らかである．(主張 2 の証明終) □

2.2 誤り許容カウント法とイプシロン劣シノプス

この章では，G. S. Manku と R. Motwani[3] および徳山[4] に従って，誤り許容カウント法について述べる．

誤り許容カウント法は，ストリームデータ中のすべての頻出イベント (または頻出時系列) を，それらの誤差を認めた見積もり出現回数とともに出力する．入力として，イベント総数 N のストリームデータ $\mathcal{S}$ が与えられ，パラメータとして目標閾値 γ と誤り値 ϵ が設定される．ただし，$0 < \epsilon < 1$，$0 < \gamma < 1$，$\epsilon << \gamma$，$\epsilon^{-1} \le N$ である．

定義 4 イプシロン劣シノプスを保持するストリームアルゴリズムとは，次の 3 つの条件を満たすイベント集合 (または時系列集合) の出力を任意の時点で行えるアルゴリズムである．

(1) 真の出現回数が γN 以上のデータは必ず出力する．
(2) 真の出現回数が $(\gamma - \epsilon)N$ 未満であるデータは出力に含まれない．
(3) 出現回数の見積りが出力されるが，見積りは (真の出現回数 $- \epsilon N$) と真の出現回数の間の値である．

アルゴリズム 2 (**誤り許容カウント法**)

まず，ストリームデータ $\mathcal{S}$ 中にある N 個の事象を $\omega = \epsilon^{-1}$ ずつの区分にわけ，それをバケットと呼ぶ．誤り許容カウント法は，ユーザから出力が要請されるまでバケット単位でストリームデータ $\mathcal{S}$ を時刻の昇順に読み込んで計算していく．N/ω の余りも 1 つのバケットと見なすので総バケット数は $\lceil \epsilon N \rceil$ となる，

ストリームデータ $\mathcal{S}$ 中に現れる事象が持つイベントの集合を $\mathcal{E}$ とする．イベント $e \in \mathcal{E}$ に対し 3 つ組 (e,f,δ) を保持する．この 3 つ組の集合 $\mathcal{D}$ を**シノプス** (*synopsis*) とよぶ．ここで，f と δ は次の自然数を表す．

f: 計算途中の e の見積り出現回数．

δ: f の持つ最大許容誤差値．

1 つの e に対する 3 つ組は高々1 つしかない．アルゴリズムはストリームデータ $\mathcal{S}$ 上の事象を 1 つずつ読んでいきシノプス $\mathcal{D}$ を更新する．現在，アルゴリズムが $b_{current}$ 番目のバケットを計算しているとする．バケット中のストリームデータの事象を 1 つずつ読む毎に，次のいずれか一つを行う．

(1) **カウント更新**: 読んだ事象 (e,t) の e に対する 3 つ組みがすでにシノプス $\mathcal{D}$ にあれば，見積もり頻度値 f を一つ増やす．
(2) **挿入**: 読んだ事象 (e,t) の e に対する 3 つ組みがなければ，新しい 3 つ組みとして $(e, 1, b_{current})$ をシノプス $\mathcal{D}$ に挿入する．

$b_{current}$ 番目のバケットに含まれる全ての事象に対する計算が終わったら，次の削除判定を行う．

(3) **除去**: アルゴリズムはシノプス $\mathcal{D}$ に含まれる全ての 3 つ組みをチェックする．そして $f \leq b_{current} - \delta$ である f と δ の組を持つ 3 つ組を全て削除する．

削除判定が終了後，次のバケットに進み上記 3 つの作業を繰り返す．

最後に出力要求に従って，その時までのシノプスに含まれる 3 つ組のうち条件を満たす 3 つ組のイベントと見積り出現回数を出力する．

出力 ユーザーから出力が要請された時点で $f \geq (\gamma - \epsilon)N$ である f を持つ 3 つ組の e を頻出イベントとして，f をその見積もり出現回数として出力する．

定理 1 (G. S. Manku and R. Motwani[3]) アルゴリズム 2 はイプシロン劣シノプスを保持する．

3. 頻出 2 時系列ストリームアルゴリズム

本章では，ストリームデータ $\mathcal{S}$ に対して，誤り許容カウント法を用いた頻出 $(2,T)$-時系列抽出アルゴリズムを述べる．

3.1 頻出 2 時系列抽出のためのパラメータ

パラメータとして目標閾値 γ と誤り値 ϵ と許容時間差 T が設定され，$N = |\mathcal{S}|$ と γ, ϵ によりイプシロン劣シノプスを満たす誤り許容カウント法を行う. まずは準備としてバケットへの区分，バケット内 2 回走査，保持するシノプス $\mathcal{D}$ について述べる．

ストリームデータを $\omega = 2\epsilon^{-1}$ ずつのバケットにわける，アルゴリズムとして出力が要請されるまでバケット単位でストリームデータを読み込んで計算していく．N/ω の余りも 1 つのバケットと見なすので総バケット数は $\lceil \epsilon N/2 \rceil$ となる．またあるバケット 1 つを計算する際，まず始めに頻出イベント保持を目的とした走査を行い，その後頻出時系列保持を目的とした走査を行う．頻出イベントとして保持されているイベントのみを対象として時系列のペアを作り，それ以外のイベントは無視する．

ストリームデータ $\mathcal{S}$ 中に現れるイベントの集合を $\mathcal{E}$ とするとアルゴリズムはイベント $e \in \mathcal{E}$ に対し 3 つ組 (e, f, δ) を保持し．さらに任意のイベント x, y に対する時系列 (x, T, y) に対しても 3 つ組 $((x, T, y), F, \Delta)$3 つ組を保持する．シノプス $\mathcal{D}$ はそれら 3 つ組を保持する集合である．f と δ，F と Δ はそれぞれ次の自然数を表す.

- f: アルゴリズムの計算途中の e の見積り出現回数.
- δ: f の持つ最大許容誤差値.
- F: アルゴリズムの計算途中の (x, T, y) の見積り出現回数.
- Δ: F の持つ最大許容誤差値.

Procedure COUNT-EVENT;
input: ストリームデータ $\mathcal{S}$ 中に現れるイベント e, バケット番号 i;
begin
  if e についての 3 つ組がシノプス $\mathcal{D}$ に存在する **then**
    その 3 つ組の f の値を 1 増やす
  else
    3 つ組 (e,1,i-1) をシノプス $\mathcal{D}$ に挿入する
end.

図 1 手続き COUNT-EVENT

Procedure COUNT-SEQ:
input: ストリームデータ $\mathcal{S}$ 中に現れるイベント e, e' からなる (e, T, e'),
  シノプス $\mathcal{D}$, バケット番号 i;
begin
  if (e, T, e') についての 3 つ組がシノプス $\mathcal{D}$ に存在する **then**
    その 3 つ組の F の値を 1 増やす
  else
    3 つ組 ((e, T, e'),1,i-1) をシノプス $\mathcal{D}$ に挿入する
end.

図 2 手続き: COUNT-SEQ

3.2 誤り許容カウント法によるストリームアルゴリズム

アルゴリズム 3

$b_{current}$ 番目のバケットを読み込んで計算するときの動作を述べる．まず任意の時刻 τ に対して $\Omega(\tau) = false$，任意のイベント x に対して $t_L(x)$=-T-1 と初期化する．

頻出イベント走査 まずは頻出イベント保持のための走査をする．カウント更新，挿入，そして除去に関して第 2 章のアルゴリズム 2 と同じ動作を行う．当然，除去は時系列ではなくイベントに対する 3 つ組を対象に行う．すべての結果はシノプス $\mathcal{D}$ に保持する，それをこの $b_{current}$ 番目のバケットでの頻出時系列走査で頻出イベントの参照として使い，また $(b_{current} + 1)$ 番目のバケットでの頻出イベント走査にも $b_{current}$ 番目のバケットまでの頻出イベント走査の結果として引き継ぐ．

Procedure CLEAR-EVENT:
input: バケット番号 i, シノプス $\mathcal{D}$;
begin
  forall イベントの 3 つ組 $(e, f, \delta) \in \mathcal{D}$ に対して **do begin**
    if $f \leq i - \delta$ **then** (e, f, δ) をシノプス $\mathcal{D}$ から消去する
  end
end.

図 3 手続き: CLEAR-EVENT

Procedure CLEAR-SEQ:
input: バケット番号 i, シノプス $\mathcal{D}$;
begin
  forall 時系列の 3 つ組 $((e, T, e'), F, \Delta) \in \mathcal{D}$ **do begin**
    if $F \leq i - \Delta$ **then** $((e, T, e'), F, \Delta)$ をシノプス $\mathcal{D}$ から消去する
  end
end.

図 4 手続き: CLEAR-SEQ

頻出時系列走査 バケット内での時系列の出現回数は，シノプス $\mathcal{D}$ に 3 つ組が含まれる頻出イベントを持たない事象に対してはペアを作らず無視する．シノプスに保持された 3 つ組のイベントを含む事象に対してのみアルゴリズム 1 方法でカウントする，ただし，この走査の始めに任意のイベント x に対して $t_L(x)$=-T-1 で初期化するため $b_{current} - 1$ 番目のバケット以前に現れる事象に対してはペアを作らない．ペアを作り時系列 (e, T, e') が出現したと見なされた場合に，

(1) **カウント更新:** 発見した時系列 (e, T, e') に対する 3 つ組みがすでにシノプス $\mathcal{D}$ にあれば，見積もり頻度値 F を 1 つ増やす.
(2) **挿入**: 発見した時系列 (e, T, e') に対する 3 つ組みがシノプス $\mathcal{D}$ になければ，新しい 3 つ組みとして $((e, T, e'), 1, b_{current})$ をシノプス $\mathcal{D}$ に挿入する.

を行いカウントする.

$b_{current}$ 番目のバケットに含まれる全ての事象に対する計算が終われば，またアルゴリズム 2 のように次の削除判定を行う.

Algorithm FREQ2SEQ;
input: ストリームデータ $\mathcal{S}$, 時間差 T, ϵ, γ;
output: イプシロン劣シノプスを満たす $(2, T)$-時系列の集合とその見積り出現回数;
begin
  b を現在走査しているバケットの番号とする;
  $\mathcal{D}$ を空のシノプスとする; Ω と t_L を初期化する; $N = |\mathcal{S}|$ とする;
  for $b = 1$ **to** $\dfrac{\epsilon N}{2}$ **do begin**
    forall バケット b 内の事象 (e, t)(到着順で処理) **do** COUNT-EVENT$(e, b, \mathcal{D})$;
    CLEAR-EVENT$(b, \mathcal{D})$;
    forall 対応する 3 つ組がシノプス $\mathcal{D}$ に含まれるイベント e' を
      持つ事象 (e', t)(到着順で処理) **do begin**
      forall 対応する 3 つ組がシノプス $\mathcal{D}$ に含まれるイベント e **do begin**
        if $e \neq e'$ で時間差が許容以内で重複が無い **then**
          COUNT-SEQ$((e, T, e'), b, \mathcal{D})$;
        if $e = e'$ で時間差が許容以内で重複が無い **then**
          COUNT-SEQ$((e, T, e), b, \mathcal{D})$;
          $\Omega(t')$ と $\Omega(t_L(e))$ を *true* にする
      end;
      $t_L(e') := t'$
    end;
    t_L を初期化する; CLEAR-SEQ$(b, \mathcal{D})$;
  forall 保持されていて $F \geq (\gamma - \epsilon)N$ である時系列の 3 つ組 $((e', T, e), F, \Delta)$ **do**
    if $F \geq (\gamma - \epsilon)N$ **then** 時系列 (e, T, e') と見積り出現回数 F を出力
end.

図 5 アルゴリズム: FREQ2SEQ

(3) **除去**: アルゴリズムはシノプス $\mathcal{D}$ に含まれている時系列に関する全ての 3 つ組みをチェックする．そして $F \leq b_{current} - \Delta$ である F と Δ を持つ 3 つ組を全て削除する．この時系列の削除判定を以て，$b_{current}$ 番目のバケットの計算は終了し，次のバケットに移る．

出力 出力が要請された時点で $F \geq (\gamma - \epsilon)N$ である F を持つ 3 つ組の時系列 (e, T, e') を頻出イベントとして，F をその見積もり出現回数として出力する．

アルゴリズムの形式的な定義を図 1, 3, 2, 4, 5 にあげる．

3.3 イプシロン劣シノプスの証明

補題 1 より，時系列の出現回数はアルゴリズム 1 で出力される出現回数のことであり，それを前提に証明する．

補題 2 i 番目のバケット終了後のシノプスに保持された頻出イベントの集合を E_i, 頻出時系列の集合を S_i とする．任意の i とイベント e, e' で，次が成り立つ．

$(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$.

証明. 時系列 (e, T, e') が i 番目のバケットで挿入されたとする．(もし挿入される事がなければこの補題 3 は無条件に成り立つ．) この時系列が挿入されるためには i 番目のバケット終了後にイベント e, e' が頻出イベントとしてシノプスに保持されていなければならない．よって任意の時系列 (e, T, e') が挿入されたバケット i で $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ は成り立つ．次の $i+1$ 番目のバケット内でも，時系列 (e, T, e') の出現回数は e,e' の出現回数を超えないので同じく $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ が成り立つ．帰納法的に $d(\geq i)$ 番目のバケットでイベント e もしくは e' が頻出イベントから削除されるまで同じ事が言える．

もしイベント e がシノプスから削除されても，そのバケットでは $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ が成り立っているので対応する時系列 (e, T, e') もシノプスから削除されている．時系列 (e, T, e') が頻出時系列としてシノプスに保持されていない間は $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ は無条件に成り立ち、その後再び時系列 (e, T, e') が挿入されてもそのバケット以降で $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ が成り立つ．

よって任意のバケットで $(e, T, e') \in S_i \Rightarrow (e \in E_i) \cap (e' \in E_i)$ が成り立つ． □

補題 3 $b_{current}$ 番目のバケットを計算するとき，ある時系列 (e, T, e') が i 番目のバケットで最後に挿入されていたとする．i 番目のバケットから $b_{current}$ 番目のバケットまでに $2(b_{current} - i) + 1$ より多く出現していれば $b_{current}$ 番目のバケット計算終了後に頻出時系列としてシノプスに保持されている．

証明. $b_{current} = i + a$ とし，a $(a \geq 0)$ に関する数学的帰納法で証明する．

(1) $a = 0$ のとき，時系列 (e, T, e') が $b_{current}$ 番目のバケット中に 2 回出現するならば，イベント e, e' も 2 回出現しているので，e, e' は頻出イベントとして保持される．よって時系列 (e, T, e') はアルゴリズム 3 でも 2 回カウントされ 3 つ組は $((e, T, e'), 2, b_{current} - 1)$ となり，削除条件 $F \leq b_{current} - \Delta$ を満たさずに頻出として保持される．

(2) $b_{current} = i + a - 1$ のとき，時系列 (e, T, e') が $2(b_{current} - i) + 1 = 2a - 1$ より多く出現していれば $i + a - 1$ 番目のバケットで頻出時系列としてシノプスに保持されると仮定する．

$b_{current} = i + a$ のとき，時系列 (e, T, e') が $2(b_{current} - i) + 1 = 2a + 1$ 回より多く出現する場合，

- i 番目のバケットから $b_{current} - 1$ 番目のバケットまでに時系列 (e, T, e') は $2a - 1$ より多く出現しているとき．
  仮定より時系列 (e, T, e') は $b_{current} - 1$ 番目のバケット終了後に頻出時系列としてシノプスに保持されている．よってこのときの時系列 (e, T, e') の 3 つ組 $((e, T, e'), F, i)$ で $F > b_{current} - 1 - i$ である．この後 $b_{current}$ 番目のバケットまでに 2 回以上出現していて，補題 2 よりイベント e, e' も $b_{current} - 1$ 番目のバケット終了後に頻出なイベントとしてシノプスに保持されているので，$b_{current} - 1$ 番目のバケットでバケット間を跨がらない限りはアルゴリズムでカウントされる，時系列 (e, T, e') は $b_{current} - 1$ 番目のバケットと $b_{current}$ 番目のバケットとの間で高々1 回しか跨がらないので，$b_{current}$ 番目のバケットで時系列 (e, T, e') の F は最低でも 1 増えることになる．よって削除条件 $F \leq b_{current} - i$ を満たさず保持される．
- i 番目のバケットから $b_{current} - 1$ 番目のバケットまでに時系列 (e, T, e') は $2a - 1$ 回以下の v 回出現した場合．
  時系列 (e, T, e') は，$b_{current} - 1$ 番目のバケットまでに含まれてしまうものを除き，$b_{current}$ 番目のバケットまでに $2a + 1 - v$ より多く出現していることになる．時系列 (e, T, e') は $b_{current} - 1$ 番目のバケットと $b_{current}$ 番目のバケットとの間で高々1 回しか跨がらないので，時系列 (e, T, e') は $b_{current}$ 番目のバケット中に $2a - v$ より多く出現していると言え，それによりイベント e, e' も $b_{current}$ 番目のバケット中に $2a - v$ 回より多く出現している．$2a - v \leq 1$ よりイベント e, e' は $b_{current}$ 番目のバケットで頻出イベントとしてシノプス

に保持され，時系列 (e,T,e') はアルゴリズムで $2a-v$ 回より多くカウントされる．従って，時系列 (e,T,e') は $b_{current}$ 番目のバケット終了時に頻出時系列としてシノプスに保持される．

よって，$b_{current}=i+a$ でも $2(b_{current}-i)$ より多く出現する時系列は必ず $b_{current}$ 番目のバケット終了時に頻出時系列としてシノプスに保持される．故に任意の $b_{current}$ で成り立つ． □

定理 2 アルゴリズム FREQ2SEQ はイプシロン劣シノプスを保持する．

証明. 次の 3 つの主張を証明する．

主張 1. 出現回数の見積りが出力されるが，見積りは，(真の出現回数 $-\epsilon N$) と真の出現回数の間の値である．

証明. 対象となる時系列 (e,T,e') が i 番目のバケットで最後に挿入されたとする．$(i-1)$ 番目のバケット終了後のシノプスに時系列 (e,T,e') の 3 つ組が含まれていない事になるので補題 3 より，時系列 (e,T,e') は 1 番目のバケットから $i-1$ 番目のバケットまでに $2(i-1-1)+1=2i-3$ 回より多く出現していれば $i-1$ 番目のバケット終了時に頻出としてシノプスに保持される．実際には保持されていないので時系列 (e,T,e') は $i-1$ 番目のバケットまでに高々$2i-3$ 回しか出現していない．$i-1$ 番目のバケットまでに時系列 (e,T,e') が出現した回数を d，さらに $i-1$ 番目のバケットから現在計算中のバケットまでに時系列 (e,T,e') がバケット間を跨いだ回数を d' とすると，時系列 (e,T,e') の真の出現回数は見積り出現回数 F を用いて，$F+d+d'$ と書ける．現在計算中のバケットを b 番目のバケットとする．$d\le 2i-3$ と $d'\le b-(i-1)$ より，$F<F+d+d'\le F+i+b-2$ となり，$b\le\lceil\epsilon N/2\rceil$，$i\le\lceil\epsilon N/2\rceil$ から $b-1\le\epsilon N/2$，$i-1\le\epsilon N/2$ であるので，$F<F+d+d'\le F+\epsilon N$ となり F について変形すると，

$$F+d+d'-\epsilon N\le F\le F+d+d'$$

となる．$F+d+d'$ は時系列 (e,T,e') のストリームデータ全体での真の出現回数だから，見積り出現回数 F は，(真の出現回数 $-\epsilon N$) と真の出現回数の間の値である．(主張 1 の証明終)

主張 2. 真の出現回数が γN 以上のデータは必ず出力する．

証明. 対象となる時系列 (e,T,e') が i 番目のバケットで最後に挿入されたとする．$(i-1)$ 番目のバケット終了後のシノプスに時系列 (e,T,e') の 3 つ組が含まれていない事になるので補題 3 より，時系列 (e,T,e') は 1 番目のバケットから $i-1$ 番目のバケットまでに $2(i-1-1)+1=2i-3$ 回より多く出現していれば $i-1$ 番目のバケット終了時に頻出としてシノプスに保持される．実際には保持されていないので時系列 (e,T,e') は $i-1$ 番目のバケットまでに高々$2i-3$ 回しか出現していない．よって全体で γN 回以上出現している時系列 (e,T,e') は，i 番目のバケット以降に $\gamma N-(2i-3)$ 回以上出現していることになる．

また補題 3 より，時系列 (e,T,e') が i 番目のバケット以降に $2(\lceil\epsilon N/2\rceil-i)+2$ 以上出現するならば時系列 (e,T,e') は最後のバケット終了時のシノプスに必ず保持される．$2(\lceil\epsilon N/2\rceil-i)+2=2\lceil\epsilon N/2\rceil-2i+2$ で，$\lceil\epsilon N/2\rceil-1\le\epsilon N/2$ より，$2\lceil\epsilon N/2\rceil-2i+2\le\epsilon N-2i+4$ となるので，時系列 (e,T,e') が最後のバケット終了時のシノプスに保持される条件は i 番目のバケット以降に $\epsilon N-2i+4$ 回以上の出現とみることもできる．

ここで $\epsilon\ll\gamma$ より，$2\epsilon<\gamma$ としてよいので，$2\epsilon N<\gamma N$ より $\epsilon N+1<\gamma N$ となる．両辺に $-2i+3$ を加えて，$\epsilon N-2i+4<\gamma N-2i+3$ が得られる．全体で γN 回以上出現している時系列 (e,T,e') は i 番目のバケット以降で $\gamma N-(2i-3)$ 回以上出現しているので，時系列 (e,T,e') は必ず最後のバケットのシノプスに保持される．

真の出現回数が γN 回以上ならば，見積り出現回数 F は $(\gamma-\epsilon)N$ 以上なので，出力条件を満たす．よって題意は成り立つ．(主張 2 の証明終)

主張 3. 真の出現回数が $(\gamma-\epsilon)N$ 未満であるデータは出力に含まれない．

証明. 真の出現回数が $(\gamma-\epsilon)N$ 未満であるとき．見積もり出現回数 f はイプシロン劣シノプス 3 番目の条件よりその真の出現回数を超えない．故に出力条件 $F\ge(\gamma-\epsilon)N$ を満たさないので出力されない．(主張 3 の証明終) □

4. データスクリーニング実験

ボットネットなどによるインターネット上の脅威を早期に検知するためには，ダークネットなどのネットワーク観測による脅威の傾向把握が必要である．その際，古くからあるマルウェアによるありふれた不正パケット群が，ボットネットなどの悪質な脅威の検知を難しくする．本論文では，提案手法を用いたダークネットデータのデータスクリーニング実験について述べる．

実験データと方法について

あるダークネットが 2009 年 2 月 12 日 16 時 09 分から 19 時 55 分までに受信したパケットをそのまま記録した不正アクセス観測データを用いた．このデータをストリームデータ $\mathcal{S}$ とする (データの大きさ $N=401,295$)．パケットの受信時間は無視し，パケットの受信

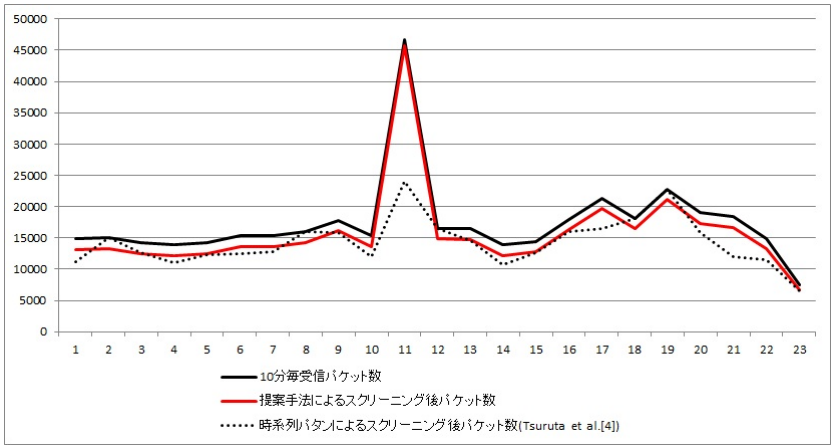

図 6 提案手法によるインターネットログのスクリーニング結果

順だけを考える．すなわち，あるパケットが $\mathcal{S}$ の i 番目に生起したイベントであるならば，そのイベントの受信時間 $t_i = i$ と定める．まず $\mathcal{S}$ に対し，第 3 章で提案した頻出 $(2,T)$-時系列発見近似ストリームアルゴリズムを適用して，頻出 $(2,T)$-時系列候補の $(2,T)$-時系列の集合を得た．各種パラメータは次のとおりに設定した: 目標閾値: $\gamma = 0.003$, 誤り値: $\epsilon = 0.0005$ 許容時間差: $T = 10$

実験は MacOSX 10.6.8, 2.53GHz Core 2 Duo, Memory 4GB で行った．得られた近似頻出 $(2,T)$-時系列集合に属す時系列に照合する出現を重複を許して削除した後のデータの大きさは 363,542 であった．また，近似頻出 $(2,T)$-時系列集合を得るまでの計算時間は 311.51(sec) であり，約 4 時間分のデータをほぼリアルタイムで処理できる手法であることがわかった．一方，悪質な不正パケットを洗い出すためのスクリーニング手法として，Tsuruta et al.[5] と比較した場合，悪質でないありふれたパケットであると思われるピーク時のパケットをスクリーニングできていないが，定常的に流れるありふれたパケットを削除する目的としては効果が期待できる (図 6).

5. 結論と今後の課題

本論文では，$(2,T)$-時系列の出現回数を定義し，イプシロン劣シノプスを保持する頻出 $(2,T)$-時系列データ抽出ストリームアルゴリズムを提案した．本アルゴリズムで出力される頻出 $(2,T)$-時系列の出現回数には超越する出現が無視されているが，これは現在読んでいるバケットから時間差 T だけ次のバケットを先読みすることで，超越する出現を出現回数に加えてもイプシロン劣シノプスを保持する頻出 $(2,T)$-時系列発見ストリームアルゴリズムを構築できる．詳細は省略する．

任意の自然数 k に対して，頻出 (k,T)-時系列の抽出を，イプシロン劣シノプスを保持しながら抽出するストリームアルゴリズムの開発が，今後の課題である．

謝　　辞

本研究の一部は国際連携によるサイバー攻撃の予知技術の研究開発 (総務省) の支援を受けたものである．

参 考 文 献

1) N.Alon, Y.Matias, and M.Szegedy. The space complexity of approximating the frequency moments *Proc. 28th ACM Symposium on Theory of Computing*, pages 20–29, 1996.
2) R.M.Karp, C.H.Papadimitriou, and S.Shenker. Simple Algorithm for Finding Frequent Elements in Streams and Bags. *ACM Transactions on Database Systems* **28(1)**, pp.51–55, 2003.
3) G.S.Manku and R.Motwani. Approximate Frequency Counts over Data Streams. *Proc. 28th International Conference on Very Large Data Bases (VLDB 2002)*, pages 346–357, 2002.
4) 徳山 豪．オンラインアルゴリズムとストリームアルゴリズム– アルゴリズム・サイエンスシリーズ 5. 共立出版, 2007.
5) H.Tsuruta, T.Shoudai, and J.Takeuchi. Network Traffic Screening Using Frequent Sequential Patterns. *Lecture Notes in Electrical Engineering* **110**, Springer, pages 363–375, 2012.