社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

編集距離に基づく インネットワークスパム検知手法

塚島洋介†　　中尾彰宏†

† 東京大学大学院　情報学環・学際情報学府　学際情報学専攻　総合分析情報学コース

あらまし　インターネットにおけるスパムメールの割合は依然として高く，2011 年の計測でメール全体の 80%以上がスパムであるという報告がある [1]．スパムの多くはボットネットから送信されている．ボットネットの規模は数百から数万と言われ，あるボットネットは１日に数十億通のスパムを送信することが出来るとされている．本研究で解決すべき問題は，ボットネットから送信されるスパムへの対策である．この問題への対応として，本研究では，編集距離に基づくネットワーク内部でのスパム検知を提案する．スパムの同一性に基づくクラスタリングを行い，スパムがボットネットから送信されているという事実に基づく判定値を定め，評価を行い 86%以上の精度でスパム検知可能であることを示す．本研究では，以前の我々の研究 [2] に加え，クラスタリングの自動化や処理の高速化に並列計算を用いる検討を行った．

キーワード　スパム，編集距離，kmeans，vp 木，最近傍探索，ネットワークプロセシング，GPGPU

An in-network spam filtering method using nearest neighbor search of edit distance

Yosuke TSUKAJIMA† and Akihiro NAKAO†

† Applied Computer Science Course, Graduate School of Interdisciplinary Information Studies, The University of Tokyo

Abstract　A huge amount of spam has become a serious threat to this communication means, since according to the existing research [1], spam accounts for as much as 80% of the entire Email messages. And a large amount of spam sent from bot-nets that are distributed widely across the Internet. Although the conventional spam filtering techniques are often implemented at the edge of the Internet. It makes sense to detect spam at the cross-road of the Internet, not only at the edges, because a large amount of spam sent from bot-nets. In this paper, we propose a novel in-network spam filtering technique that detects duplicate content of the packets flowing at the core of the network using edit distance. Our evaluation of the proposed method using the real packet trace shows that we can successfully detect spam with recall of 88% and precision of 86%. We also conduct feasibility study of our method, especially from the perspective as to whether we can perform our method in real-time at high speed routers and switches of today.

Key words　spam, edit distance,kmeans, vp tree, nearest neighbor search, network procceing，GPGPU

1. ま え が き

スパムメールは増加の一途を辿り，2011 年の計測でメール全体の 80%以上がスパムであるという報告がある [1]．本来必要であるメールの管理が大量のスパムの為に困難な状況になることや，スパムに添付されたウィルスによって，サーバ内の情報が外部に流出するなどの被害が生じている．スパムメールが与える悪影響は，フィッシング詐欺や添付ウィルスの感染といった直接的なものだけではなく，スパム除去を行うアプライアンスの導入を余儀なくされる間接的なコストもある．こうしたスパムに起因する，我が国の ICT インフラにおける年間損失額は年間数千億円にものぼるとの報告もある [3]．スパムの多くはボットネットから送信されている．あるボットネットは１日に数十億通のスパムを送信することが出来るとされている [4]．ボットネットはマルウェアなどに感染したボットと呼ばれる PC などから構成される．こうしたボットは指令・命令を送信するコン

ピュータから指示を受けスパムの送信や DDoS 攻撃などを実行する．ボットネットの規模は数百から数万と言われている [5]．近年，マイクロソフト社によって遮断された大規模ボットネット Rustock は約 100 万台のボットからなるボットネットワークであったと推定されている [4]．

本研究では，こうした状況を踏まえた以前の我々の研究 [2] に加え，同一性に基づくクラスタリングの自動化，及び処理の高速化を行うために GPGPU を用いた計算方法について検討を行った．

2. 関 連 研 究

関連研究として，スパムフィルタリング技術を挙げる．

2.1 スパムフィルタリング

スパムフィルタリングは大別すると 2 つに分類することができる．コンテンツフィルタリングとネットワークフィルタリングである．

2.1.1 コンテンツフィルタリング

コンテンツフィルタリングは，メッセージに対してフィルタリングを行う．コンテンツフィルタリングの基本方針は，メッセージに含まれる単語に値を付け，その値をもとに判定を行うことである．値の付け方は，ルールによるものとベイズ推定によるものがある．

ルールによる値付けとは，スパムに多く含まれる単語を抽出しておき，判定すべきメールがその単語を含んでいる場合に，その単語のスパムに含まれる度合いに応じて値をつける方法である．

ベイズ推定による値付けは，ある単語を持つメールが実際にスパムである条件付き確率で値を与える．

例えば，Ω をメール全体，$S \subset \Omega$ をスパムの集合，$H \subset \Omega$ を通常のメールの集合とする．$W = \{w_1, ..., w_N\}$ を S と H に含まれるメッセージから集めた全ての単語とする．

$$A_k := \{\omega \in \Omega \mid \text{メール}\omega\text{に単語 } w_k\text{が含まれている }\}$$

と定める．この時，メール m が幾つかの単語を含んでいる場合，$m \in \bigcap_i A_i =: A$ と書ける．従って，そのメールがスパムである確率 $P(S|A)$ は，ベイズの定理を用いて，

$$P(S|A) = \frac{P(S)}{P(A)} \prod_i P(A_i|S)$$

と計算することができる [8]．

2.1.2 ネットワークフィルタリング

ネットワークフィルタリングとは，ネットワーク層のヘッダー情報のみを用いて，スパムをフィルタリングする方式である．グレイリスティング [9]，ブラックリスト方式 [10], [11]，S25R [12] を挙げることができる．ブラックリスト方式は，スパムを送信した IP アドレスをブラックリストとして登録し，ブラックリストからのメールを拒否する方式である．例えば，Spamhaus [10] が提供する DNSBL や Barracua が提供する BRBL [11] などが挙げられる．グレイリスティングは，スパムを送信するホストが，SMTP サーバからエラー応答を受け取った際に，再送を行わないことを利用したフィルタリングである．再送を行うホストはスパムを送信するサーバではないとし，その IP アドレスをホワイトリストとして用いる [9]．S25R は IP アドレスからホスト名を逆引きし，逆引きできない場合や，ボットの疑いが高い場合メールの受信を拒否する方式である [12]．その他のネットワークフィルタリングとしては，TCP フィンガープリントを用いたスパムフィルタリング [13] を挙げることができる．この手法は，OS の TCP 実装によって異なる特徴量を用いボットを特定し，そこから送信されるメールをフィルタリングする方式である．

3. 研究の動機

現在のスパムフィルタリング手法は，主にネットワークの端点（エッジ）におかれたファイアウォールやアプライアンスを用いて，ネットワーク単位で独立実行されることが多い．この理由は，そもそもスパムフィルタリングをネットワーク内部で処理することが想定されておらず，また，高帯域でのフィルタリング技術も成熟していなかったことなどが挙げられる．しかし，近年のスパムはボットネットから送られる事が多く，同じ内容のスパムが多くの送信元から多くの宛先へ送信されている [7]．こうした従来方式のネットワークエッジでのスパム検知の問題点として 2 つ挙げる事が出来る．

問題点 1：冗長処理が発生している点

ボットネットから同一スパムが送信されているため，ネットワークエッジでスパム検知を行った場合，同一スパムに対して，各サブネットに存在するメールサーバ各々がスパム検知を実行している．更に，各サブネットに存在するメールサーバ同士は，独立してスパム検知を行っているため，スパムに関する情報を共有する必要がある．従って，全体として見た場合，冗長処理が発生しているといえる．

問題点 2：スパム検知の精度に悪影響が出る点

スパムはボットネットから同一のものが一斉に送信されることが多いが，ネットワークエッジでメールサーバがそれぞれスパム検知を行った場合，それぞれのメールサーバが受信するスパムは分散している分だけ減少する．従って，スパム検知に必要となる情報も減少する．特に，スパムが同一時刻に送られたという情報，すなわち，時間的相関の情報を喪失する．結果として，スパム検知の精度に悪影響が出る．

4. 目　　的

本研究の目的は冗長処理を削減する高精度のスパム検知を考案することで，ボットネットを検出し，将来のスパムを未然に防ぐ事である．ボットネットはスパムを大量に送信している．従って，スパムを検知することで，ボットネットの情報を収集することが可能である．そして，ボットネット情報を用いて，ブラックリストを作成し，そのブラックリストを用いて将来のスパムを未然に防ぐ方法が考えられる．

5. 提 案 方 式

こうした問題点の解決策として，従来方式とは異なるスパム

検知を提案する．図 1 はボットネットからスパムメールが送信される様子を模式的に示したものである．インターネットの構造は，規模の小さなネットワークが寄り集まりより大きなネットワークを形成する．そのため，あるネットワークから別のネットワークにパケットが送信される際には，パケットはネットワークとネットワークの接続部分を通過する．従って，ネットワーク内部では，スパムメールが集約される経路が存在する．そこで，この集約箇所であるネットワーク内部でのスパム検知を提案する．

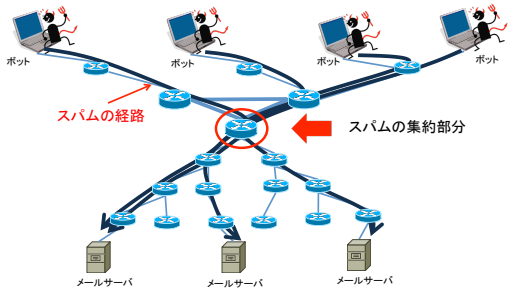

図 1　ボットネットからスパムメールが送信され集約される様子

具体的なスパム検知は，以下のステップを踏んで実行する．

1．ネットワーク内部のルータを通過するフローのうち，メールトラフィックを単位時間あたりのブロック毎に抽出する．

2．抽出したブロック毎にメールの同一性に基づいて，同一メール同士をひとまとめとするクラスタリングを行う．

3．クラスタリングの結果生じるクラスタ毎にスパム判定を実行する．

図 2 にスパム検知の流れを示す．

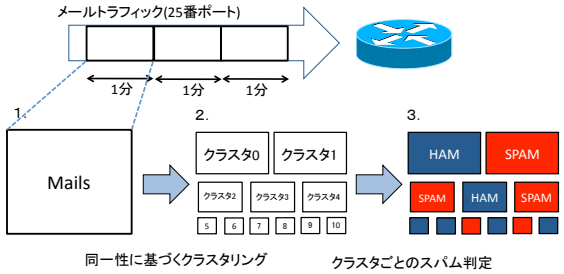

図 2　提案方式概要

6.　課題 1 :スパムの同一性検知

6.1　課題 1 の概要

提案方式を実行する際にまず課題となるのは，スパムの同一性の検知である．提案方式では，メールの同一性に基づいてクラスタリングを行うが，この同一性は，メールのペイロードの完全一致では得る事が出来ない．これは，メールのペイロードの一部がメール毎に変化するためである．ヘッダに記述されるメールの送信日時や，メールの識別番号などはメール毎に異なる．また，メールのテキスト部分と添付情報を区切る為の目印として用いられている MIME の boundary はテキスト部分と重複しないように，無作為に生成されることが多く，メール毎に異なったものとなる，更に，スパムの本文には，ランダムに改行文字などが挿入されたスパムや，フォーマットは同じであるが一部の文章が異なるといったスパムも存在する．従って，メールの同一性を得る為には，ペイロードの完全な一致ではなく，類似性を検知する方法が必要となる．

6.2　編 集 距 離

文字列の類似性を測る手法として，編集距離を用いる．編集距離は 2 つの文字列に対して距離を与える事ができ，この距離が近いものほど類似性が高いことを示している．編集距離は，文字列に含まれる文字の変換回数を距離としている．

メールのペイロード同士に対してこの編集距離を用い，距離を算出し距離が近いものを同一メールとすることで，メールの同一性を得る事ができる．

以下に編集距離の定義を述べる．

［定義 1］　x, y を文字列とする．編集距離 $d(x, y)$ とは，文字列 x に対して，以下の 3 つの操作を繰り返し行い，文字列 y と等しくするために必要な最小操作回数である．

挿入：文字列 x の任意の場所に，任意の 1 文字を挿入する．

削除：文字列 x の任意の 1 文字を削除する．

置換：文字列 x の任意の 1 文字を，任意の別の 1 文字に置換する．　□

編集距離は，距離の公理を満たす．すなわち，以下が，任意の文字列 x, y, z に対して成り立つ．

(1) $d(x, y) \geqq 0$

(2) $x = y \Leftrightarrow d(x, y) = 0$

(3) $d(x, y) = d(y, x)$

(4) $d(x, y) + d(y, z) \geqq d(x, z)$

編集距離を求める最も基本的なアルゴリズムは以下のような動的計画法を用いるものである [15].

［アルゴリズム 1］　x, y を文字列とする．$x[i]$ を文字列 x の i 番目の文字とし，$x[i...j]$ を文字列 x の i から j 番目の文字列とする．

$$t(i, j) = d(x[1...i], y[1...j]) \quad 1 \leqq i \leqq |x|, 1 \leqq i \leqq |y|$$

とおく．初期値として，$t(i, 0) = i$, $t(0, j) = j$ を用いる．$t(i, j)$ は

$$t(i, j) = \min \begin{cases} t(i-1, j-1) + d(x[i], y[j]) \\ t(i-1, j) + 1 \\ t(i, j-1) + 1 \end{cases}$$

によって，帰納的に求めることができる．$d(x, y) = t(|x|, |y|)$ なので，最終的に文字列 x と文字列 y の編集距離を求められる．　□

$t(i, j)$ をもとめる回数が $|x||y|$ であるので，計算時間は $O(|x||y|)$ である．

6.3　クラスタリング

編集距離を用いる事で，スパム同士の距離から類似性を距離で表すことができる．そこで，その距離を用いたクラスタリングによって，内容が同一なメールを集積することが可能となる．以下に，基準値による方法，k-means による手法を述べる．

6.3.1 基準値による方法

基準値によるクラスタリングは，距離に関して基準値を定め，その基準値以下の距離同士にあるものを同じクラスタとする方法である．

［定義 2］ 基準値を $T > 0$，x, y をメールとし，$d(x, y) < T$ を満たす時，x, y にパスがあると定める．

このとき，クラスタ C を連結グラフと定める．

ここで，連結グラフとは，グラフ上の任意の 2 点に対してパスが存在するものである． □

この時，メール全体を M，クラスタを $\{C_i\}_{i\in\Lambda}$ とすると，

$$C_i \neq \emptyset,\ C_i \cap C_j = \emptyset,\ M = \cap_{i\in\Lambda} C_i$$

が成り立ち，クラスタはメール全体を重なり無く分割することが分かる．

$C_i \neq \emptyset$ は，任意のメール x に対し，$d(x, x) = 0 < T$ が成立するためである．また，$C_i \cap C_j = \emptyset$ は，$\emptyset$ で無いと仮定すると，C_i, C_j は一つの連結グラフになる為成立する．$M = \cap_{i\in\Lambda} C_i$ は, 任意のメール x に対していずれかのクラスタは存在するので成立する．

6.3.2 k-means による方法

k-means と呼ばれるクラスタリングは自動的にクラスタリングを行うことが可能な方法であり，入力データとクラスタ数 k を与え，入力データから k 個の代表ベクトルを得る手法である [18].

アルゴリズムは以下のように書ける．

［アルゴリズム 2］ $D = \{x_1, ..., x_n\}$ を入力データ, k をクラスタ数とする．D から k 個の代表ベクトル $c_1, ..., c_k$ を任意に選び初期値とする．以下 1．と 2．を代表ベクトルが変化しなくなるまで，繰り返す．

1．クラスタ C_i を

$$C_i = \{x \in D | d(x, c_i) = \min\{d(x, c) | c \in \{c_1, ..., c_k\}\}\}$$

とする．すなわち，$x \in D$ に最も近い代表ベクトルが c_i であるとき，クラスタ C_i に x を入れる．

2．C_i の重心点を新たに代表ベクトルとする．

□

k-means は，$c_1, ..., c_k$ を変化させ，以下の評価関数 L を最小化しているととらえることが出来る．

$$L(c_1, ..., c_k) = \sum_{i=1}^{n} \min_{l=1,...,k} d(x_i, c_l)^2.$$

6.3.3 ギャップ統計量

k-means でクラスタリングを行う際には，予めクラスタ数 k を定めておく必要がある．データから自動的にクラスタ数 k を決定する方法が提案されており，その一つとしてギャップ統計量を用いるものがある [17]．ギャップ統計量を用いる方法は，データの分布に仮定をすることなくクラスタ数を求める事に特徴がある [18].

［定義 3］ X クラスタリングを行う入力データとし，Y を入力データ X のデータ数と同数で，一様分布であるデータとする．

L_k を X のクラスタ数 k に対する k-means における評価関数とする．M_k を Y のクラスタ数 k に対する k-means における評価関数とする．

ギャップ統計量 G_k を，

$$G_k = \log \frac{M_k}{L_k}$$

で定める． □

k を変化させてギャップ統計量の値が極大となる k が求めるクラスタ数である．

7. 課題 1 の検証

課題であったスパムの同一性に対応するため，スパムのペイロードの可変性に対し編集距離を定め，その距離をもとにクラスタリングを行い同一スパムを集積することを提案した．そこで，基準値によるクラスタリングと k-means によるクラスタリングがスパムのクラスタと通常メールのクラスタを分割できるかを検証する．

7.1 検証用データ

検証に用いるデータは，ある ISP を流れた 1 分間のトラフィックを用いる．このトラフィックに含まれる 25 番ポートのメールフローに含まれるメールを検証に用いた．

表 1 にあるように，メール総数は 1039 通．表 1 におけるインバウンドとは，ISP のネットワークに向けてに送られてきたフローのことを指し，アウトバウンドとは，ネットワークの外へ送られたフローの事を表す．全てのメールのうち，スパムは 420 通 (40.4%) であった．また，インバウンドメールの数は 589 通．このうち 321 通 (54.5%) がスパムであった．スパムであるかの判定は手動で行った．

表 1 検証用データ

	スパム	通常メール	計
インバウンドとアウトバウンド	420	619	1039
インバウンド	321	268	589

編集距離の計算には，各メールの最後尾文字列のバイナリデータを 16 進数に変換した 1023 文字を用いた (4092bit)．したがって，今回，編集距離の最大値は 1023 である．最後尾文字列を用いる理由は，メールのペイロードのヘッダ部分の可変性の影響を小さくする為である．

7.2 基準値を用いたクラスタリングの評価

基準値を用いてクラスタリングを行う時，クラスタは基準値 T に依存し，基準値 T を増加させた場合，本来は同一ではないメールを同一と判断してしまう状況が生じる．これは，基準値を用いたクラスタリングは，メール同士が基準値 T 以下である場合，それらが同じクラスタに属すとし，同一メールであるとしたためである．そこで，ひとつのクラスタにスパムと通常メールが混在しない度合いを表すために，クラスタリングの回収率 R_c，分離率 P_c を，以下のように定める．

［定義 4］ S をスパムの数，M をスパムを含むクラスタに含まれるメールの数 (クラスタサイズ) の合計とする．S_c をスパム

のうちクラスタサイズが 2 以上のクラスタに含まれるスパムの数を表す.

$$P_c = \frac{S_c}{M}$$

$$R_c = \frac{S_c}{S}$$

□

分離率は，クラスタの中にどの程度スパムと通常メールが混在しているかを表している．分離率が 100%である場合，スパムを含むクラスタに，通常のメールが含まれていないことを示している．

回収率は，意味をもつクラスタがどの程度存在するかを表している．クラスタのスパム判定を行う為には，クラスタサイズが 2 以上である必要がある．従って，このクラスタサイズが 2 以上のものが，意味をもつクラスタである．スパムと通常メールを完全に分けることができた場合，分離率は 100%となる．

図 3 は，60 秒フローに含まれる 1039 通に対してクラスタリングを行った際の，基準となる距離に対する分離率 (separation) と回収率 (recall) である．基準値 300 の時，分離率が 100%となることが見て取れる．

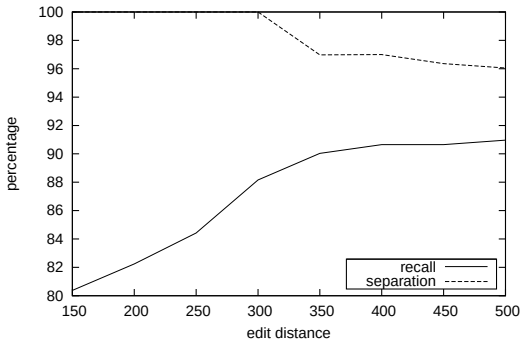

図 3　クラスタリングの回収率と分離率

図 4 は，上記回収率に分離率を乗算したものである．これは，効率性を表していると考えられる．編集距離が 300 で最大値となり，この値が最も効率的であるといえる．

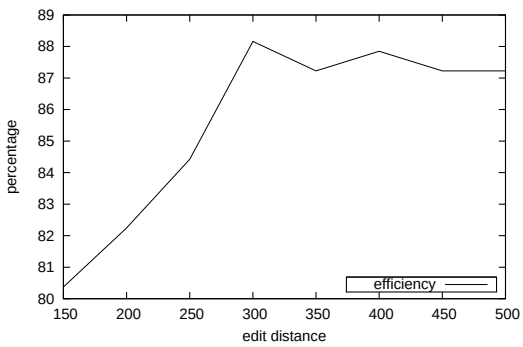

図 4　クラスタリングの効率性

以上より，基準値を用いてクラスタリングする際には，このデータに対しては基準値を 300 に設定することが妥当であり，基準値をこの値に設定することにより，スパムクラスタと通常メールクラスタを完全に分割することが可能である．

7.3　k-means を用いたクラスタリングの評価

次に，k-means を用いたクラスタリングの評価を行う．

k-means でクラスタリングを行う際に，クラスタサイズが小さいものが多数存在すると，クラスタリングの精度に影響が生じる事がある．こうした影響をのぞく為に，クラスタサイズが小さいものを取り除くことを行った．

k-means によるクラスタリングを，基準値によるクラスタリングで定めた分離率と同様の分離率を用いて評価を行う．スパムを含むクラスタに，通常メールが含まれないとき分離率が 100%となる．

k-means によるクラスタリングでは初期値によって分離率の値が変化する．そこで，k-means によるクラスタリングの評価を得るために初期値をランダムにし，複数回 k-means を実行しその分離率の平均を求めた．

評価関数に加える要素をクラスタの中心から近いもの 30%に限定してギャップ統計量を算出し，クラスタ数を決定し，その値を用いて k-means を行った際のグラフが図 5 である．横軸は取り除いたクラスタサイズを表している．

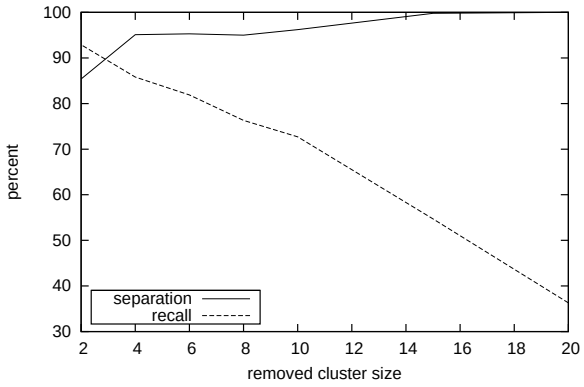

図 5　クラスタリングの分離率，回収率

7.4　課題１の結論

可変性を持つスパムのペイロードの同一性の検知と言う課題に対して，編集距離を用いたクラスタリングを適用した．基準値を用いたクラスタリングでは分離率 100%のとき回収率が 88%で，k-means を用いたクラスタリングでは分離率が 100%のとき回収率が 36%であるという結果を得た．従って，編集距離を用いてクラスタリングすることで同一なメールをまとめ，スパムクラスタから通常メールを分離できることを示した．

8.　課題２：検知の高速化

8.1　課題２の概要

本提案方式では，ネットワークの接続点であるメール情報が集約される箇所で，スパム検知を行うことを想定し，ネットワーク内部での処理を念頭においている．従って，集約した情報を一括して処理するので高速な処理が必要となる．

本提案方式に於ける最も処理に負荷が掛かる箇所は編集距離

の計算である．したがって，編集距離の計算時間の削減が全体の処理時間の効率化に繋がる．そこで，編集距離の計算に着目し，その時間的コストを検証する．

編集距離の計算時間の削減方法としては，処理の並列化や，計算回数の低減が考えられる．そこで，GPU を用いた計算の並列化とデータ構造を用いた計算回数の低減を検証する．

8.2 GPU による処理の並列化

本提案方式では，ルーターに流れ込んでくるフローを単位時間で区切ってブロック毎に処理を行う．ブロックに含まれるメールの数を N とすると，このメール同士の間の数は $N(N-1)/2$ であるので， 最大で $N(N-1)/2$ 回の距離の計算が必要となる．これらの距離の計算は互いに独立して実行可能である．そこで，この計算を GPU によって並列計算することで，計算時間の削減を行う方法を検討する．

8.3 最近傍探索

また，並列計算の他に，距離計算の回数をデータ構造を用いて削減する方法を検討する．距離集合に対して，最近傍探索を行うことの出来るデータ構造である vp 木を用いる [16]．vp 木は距離集合を二分木で表現するデータ構造である．従って，vp 木を用いると，距離集合に含まれる要素の中から，距離が一定値以下のものを計算コスト $O(\log_2 N)$ で探索することが可能である．従って，クラスタを効率的に構成することが出来る．以下が vp 木のアルゴリズムの概略である．

［アルゴリズム 3］ $\mathcal{S} = \{S_1, S_2, ..., S_k\}$, d を $\mathcal{S}$ 上の距離とする．$\mathcal{S}$ 上の vp 木 $\mathcal{V}$ は，以下のように構築する．

1. $\mathcal{S} = \emptyset$ のとき，空の木を作る．
2. $\mathcal{S} \neq \emptyset$ のとき，S_v を任意に $\mathcal{S}$ から選ぶ．(基準点 S_v を vantage point と呼ぶ) 基準値 M を

$$M = \mathrm{median}\{\mathrm{d}(\mathrm{S_i}, \mathrm{S_v}) \in \mathbb{R} \mid \forall \mathrm{S_i} \in \mathcal{S}\}$$

と定める．そして，

$$\mathcal{V}_l = \{S_i \in \mathcal{S} \mid d(S_i, S_v) < M,\ S_i \in \mathcal{S} \text{ かつ } S_i \neq S_v\}$$
$$\mathcal{V}_r = \{S_i \in \mathcal{S} \mid d(S_i, S_v) \geqq M,\ S_i \in \mathcal{S}\}$$

と $\mathcal{V}$ を分割する．この分割を再起的に繰り返し木を構築する． □

木の構築に必要な距離の計算回数は，木の高さが $\log_2 N$ で，同じ木の高さにおいて基準値 M を計算する回数が N なので，$O(N \log_2 N)$ である．

最近傍探索は以下のように行う．

［アルゴリズム 4］ Q を探索対象とし，Q との距離が r 以下であるものを $\mathcal{S}$ から探索するとき，以下を再帰的に実行する．

1. $d(Q, S_v) \leqq r$ を満たすとき S_v が求めるものである．
2. $d(Q, S_v) + r \geqq M$ の時，次に $\mathcal{V}_r$ を探索する．
3. $d(Q, S_v) - r \leqq M$ の時，次に $\mathcal{V}_l$ を探索する． □

上記条件 2 と 3 は共に成立する可能性があるので，探索を分割した両側の木に対して行う場合があることに注意が必要である．

9. 課題２の検証

本提案方式において最も計算負荷が高い編集距離の計算にかかる時間的コストの計測，及び，必要なリソースの見積もりを行う．ここでは，GPU を用いた並列計算による計算コストと最近傍探索を行う事の出来るデータ構造である vp 木を用いた計算コストの見積もりを行う．

9.1 GPU による編集距離計算の時間的コスト

60 秒フローに含まれるインバウンドメール 589 通に対して，それぞれの間の距離を GPU を用いて並列計算を行った．各メールの距離の計算はそれぞれ独立しているので，並列計算が可能である．

9.1.1 実 行 環 境

Nvidia 社 Tesla C1060 を用いて計算を実行した．Tesla C1060 の基本的なスペックは以下である．

表 2 Tesla C1060

ストリーミングプロセッサーコアの数	240
プロセッサーコアの周波数	1.296Ghz
メモリ合計	4GB
メモリ速度	800MHz
メモリインターフェース	512-bit GDDR3
メモリ回線速度	102GB/s

9.1.2 GPU と CPU の実行時間の比較

図 6 は，CPU で実行した際の時間と GPU で実行した際の計算時間の比較である．横軸はメールの数を表し，縦軸は実行時間を表している．比較に用いた CPU は Intel 社 Xeon W5590 3.33GHz(1 スレッド) である．

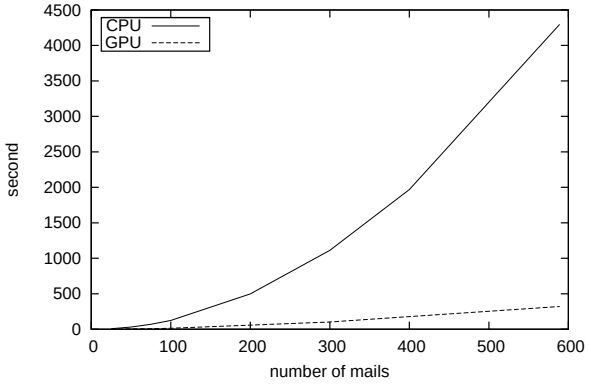

図 6 GPU と CPU の実行時間の比較

9.1.3 リアルタイム実行の為のリソースの見積もり

計算を実行するマシンを増やせばその台数に比例して，計算時間が減少する事が見込まれる．そこで，フローに対して台数がどれほど必要かを見積を示す．表 3 はその結果を示したものである．15 秒のフローに対しては，１台の GPU でフローに含まれるメールの間のそれぞれの編集距離を算出することが可能であり，60 秒フローに対しては，約 5 台程度の GPU が必要となる．

9.2 vp 木構築の時間的コスト

60 秒フローに含まれるインバウンドメール 589 通に対して，各メールの距離を編集距離で定めた距離集合に対する vp 木の

表 3　GPU の必要台数

フロー秒数	GPU 数
15 秒	0.86
30 秒	2.8
60 秒	5.3

構築時間の計測を行う．

vp 木構築の際の基準値 M の計算は，各距離計算が独立なので並列実行が可能である．そこで，pthread ライブラリを用いて並列計算を C 言語で実装した．以下の表 3 で示す実行環境で，vp 木構築の計算コストを評価するための実験を行った．

表 4　実 行 環 境

CPU	メモリ	OS
Xeon W5590x2 3.33GHz	24GB	Ubuntu 9.10

その結果，589 通のインバウンドメールを対象とした vp 木構築に完了する時間が 39.7 秒であった．用いたデータは 1 分間のトラフィックであるので，vp 木の計算コストだけを考えれば，リアルタイム処理ができる可能性がある．

9.3　課題２の結論

インネットワークを指向し提案方式において，課題となる計算時間の削減に関して，GPGPU を用いた並列計算による削減の検討とデータ構造を用いた計算コスト削減の検討を行った．結果として，GPGPU 5 台程度で 60 秒のフローの編集距離の計算を 60 秒以内に完了出来る事を示し，データ構造を用いる事で，約 40 秒で完了出来る事を示した．従って，リアルタイム処理の可能性を示した．

10.　課題３：スパムの判定

10.1　課題３の概要

課題 1 でメールのクラスタリングを編集距離に基づいて行うことが可能であることを示した．次に，クラスタリングしたメールが，スパムクラスタであるのか，通常のメールであるのかを判定する手法が必要となる．スパムの多くは，ボットネットから送信され，同様の内容にも関わらず，IP アドレスが全く異なる送信元から送られてきている傾向がある．同様の内容にも関わらず，送信元の IP アドレスが大きく異なる要因のひとつとして，多くのボットがあるひとりのスパマーによって制御されているといったことが挙げられる．近年，遮断された大規模ボットネットワークである Rustock には，スパムテンプレートが実装されており，それを用いてスパム送信されていたことが報告されている [4].

こうした状況を利用した，スパムクラスタと通常メールのクラスタを判別する判定方法が必要である．

10.2　判　定　値

上記のような状況を踏まえ，クラスタがスパムであるか通常のメールであるかの判定を，そのクラスタに属するメールの送信元が複数傾向であるか，単一傾向であるかによって行う．

次のように各クラスタに対して判定値 D を定める．

［定義 5］

$$D = \frac{\max\{\text{ ある送信元から送られるメールの個数 }\}}{\text{クラスタサイズ}}$$

□

判定値 D が基準とする値より大きい値の場合通常メールと判断し，判定値 D が基準とする値以下の場合スパムと判断する．

11.　課題３の検証

11.1　判 定 値 D

60 秒フローに含まれるメールに対して基準値を用いたクラスタリングを行い，判定値 D を算出する．

スパムクラスタとハムクラスタを判定する為に，基準値を用いてクラスタリングしスパムクラスタとハムクラスタ毎に判定値を求めた．表 5 は各クラスタの判定値を表したものである．表 5 にあるように各スパムクラスタの判定値 D の平均が 0.43，通常のメールクラスタの判定値 D の平均が 0.81 であった．従って，この判定値によって，スパムと通常のメールが判別可能である事がわかる．そこで，今回判定値として 0.6 を用い，$D > 0.6$ のクラスタを通常のメールクラスタ．$D \leqq 0.6$ のクラスタをスパムクラスタとする．

表 5　判 定 値 D

	スパム	ハム
D	0.43	0.81

11.2　再現率と適応率

スパム判定の性能を測定する指標として，再現率 R_f，適応率 P_f を以下のように定める．

［定義 6］　S をスパム全ての数，M_f を全てのメールのうちスパム判定によってスパムと判定されたメールの数とする．S_f を S と M_f の共通部分とする．

$$R_f = \frac{S_f}{S}$$

$$P_f = \frac{S_f}{M_f}$$

□

再現率はスパム判定によってどれ位のスパムを得られるかを表す回収率 (特定率) を表し，適応率はスパム判定がどれほど正確であるかという精度を表している．

本方式のスパム判定は，クラスタサイズが小さい場合，誤判定を行う可能性が高くなる．これは，スパム判定は，クラスタに含まれるメールの同一送信元の割合を用いて行うので，クラスタサイズが小さいと，判定のための情報量が十分でないためである．

図 7 は，589 通のインバウンドメールに対してスパム判定を行った際の，あるクラスタサイズ以上のクラスタに対する，スパム判定の再現率と適応率を示したものである．

再現率 88%の場合，適応率が 86%で，再現率が 55%であるとき，適応率が 100%であることが示されている．

11.3　従来方式との比較

60 秒フローに含まれる 589 通のメールに対して，従来方式

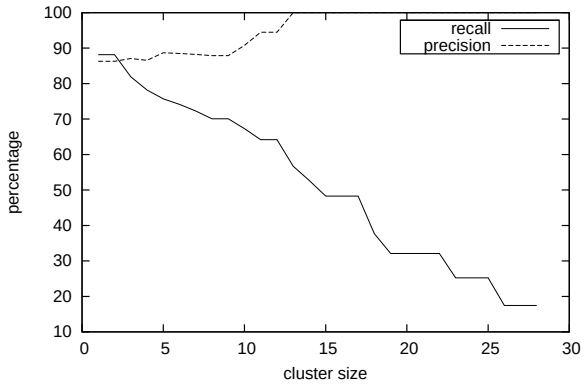

図 7　再現率と適応率

のスパムフィルタリングアプライアンスを適用し，適合率と再現率の提案方式との比較を行う．

従来方式のスパムフィルタリングアプライアンスが行う主なフィルタリングは，以下のフィルタリングを用いる．IP アドレス解析，ウィルススキャン，フィンガープリント解析，インデント解析，画像解析，ベイジアン解析，スパムルールスコアリング．

従来方式のスパムフィルタリングアプライアンスは，スパムスコアリングの閾値設定，ベイジアン解析のための学習が必要であり，閾値設定は推奨値，学習にはスパム 200 通，非スパム 200 通を用いた．

表 6 は，提案方式と従来方式と再現率・適応率の比較を示したものである．

表 6　提案方式と従来方式と再現率・適応率の比較

	提案方式	既存方式
再現率	88%	73%
適応率	86%	81%

11.4　課題３の結論

同一性に基づくクラスタリングによって構成したクラスタがスパムのクラスタであるかの判定の課題に対し，スパムがボットネットから多く送られてくることを利用した判定値 D を定義し，小規模クラスタに対する判定が精度 86%，回収率が 88%，大規模クラスタに対する判定が 100%，回収率が 57%であると言う結果を得た．従って，ボットネットから送信されるスパムを高精度で検知出来る事を示した．

12.　ま　と　め

12.1　貢　　献

本論文では提案方式の３課題に関して，以下の貢献を行った．

貢献１　スパムの同一性検知

スパムの同一性を文字列の類似度を表す編集距離を用いて検知し，その同一性に基づいて類似なメールをひとまとめにするクラスタリングを行うことが可能であることを示した．また，クラスタリングに関して k-means 及びギャップ統計量を用いてクラスタリングの自動化の検討を行った．

貢献２　検知の高速化

本提案方式で最も処理負荷が高い編集距離の計算に関して，編集距離の計算回数をデータ構造を用いて削減することで，スパム検知に必要な情報の構築を，入力データの時間内に完了することができることを示した．更に，本研究では，対象データに対して，リアルタイム処理を行うための GPGPU を用いた並列計算に必要なリソースの見積もりを行った．

貢献３　スパムの判定

ボットネットから送られたスパムが送信元が多様であるという特徴を用いることで，スパムの判定基準を定義し，この判定値がボットネットから送信されるスパムに対して高精度でスパム検知可能であることを示した．

12.2　今後の課題

今後の課題として，より多様なデータに対する基準値 T や判定値 D の評価を行う必要がある．また，編集距離以外の計算コストの検討や実環境での実行などが今後の課題としてあげられる．

文　　献

[1] Symantec，http://www.symantec.com/ (Ref 2011.2.8).
[2] 塚島洋介, 中尾彰宏, “編集距離の最近傍探索を用いたインネットワーク・スパムフィルタリング手法”, IEICE NS, 2011.
[3] 竹村 敏彦，若林 成嘉．”迷惑メールが日本経済に及ぼす影響の調査について”，日本データ通信， 2009.
[4] Battling the Rustock Threat, http://www.microsoft.com/ (Ref 2012.1.11).
[5] The top 10 spam botnets, http://www.techrepublic.com/ (Ref 2012.1.11).
[6] A. Hanemann, J. Boote, E.Boyd, J. Durand, L. Kudarimoti, R. Lapacz, D.Swany, S. Trocha , J. Zurawski, “Perfsonar: A service oriented architecture for multi-domain network monitoring”，ICSOC，2005
[7] A. Ramachandran, N. Feamster, “Understanding the network-level behavior of spammers”，SIGCOMM 2006.
[8] D.Heckerman,E.Horvitz,M.Sahami,S.Dumais, “A Bayesian approach to filtering junk e-mail”，Proc.AAAI，1998.
[9] Greylisting，http://www.greylisting.org/ (Ref 2011.2.8).
[10] Spamhaus，http://www.spamhaus.org/ (Ref 2011.2.8).
[11] Barracuda, http://www.barracudanetworks.com/ (Ref 2011.2.8).
[12] 浅見秀雄，“阻止率 99%のスパム対策方式の研究報告”，http://gabacho.reto.jp/anti-spam/anti-spam-system.html.
[13] H.Esquivel，T.Mori, “Router-level spam filtering using tcp fingerprints: Architecture and measurement-based evaluation”，CEAS，2009.
[14] V.I.Levenshtein，“Binary codes capable of correcting deletions, insertions, and reversals”，Cybernetics and Control Theory, 1966.
[15] T.F.Smith，“Identification of common molecular subsequences”，Journal of Molecular Biology, 1981.
[16] J.K.Uhlmann，“Satisfying general proximity/similarity queries with metric trees”，Information Processing Letters，1991.
[17] R. Tibshirani, G. Walther, T. Hastie, “Estimating the number of clusters in a data set via the gap statistic”, Journal of the Royal Statistical Society: Series B, 2001
[18] 金森 敬文, 竹之内 高志, 村田 昇, 金 明哲, “パターン認識”, 共立出版, 2009
[19] T. Bozkaya，M. Ozsoyoglu，“Distance-based indexing for high-dimensional metric spaces”，ACM SIGMOD，1997.