2009-IOT-4（4）
2009 / 3 / 5

DNS クエリグラフを用いた悪性ドメイン名リスト評価

石橋圭介† 豊野剛† 佐藤一道† 岩村誠†

† 日本電信電話株式会社 NTT 情報流通プラットフォーム研究所
東京都武蔵野市緑町 3–9–11

あらまし DDoS やスパム，フィッシングに用いられるボットネットはインターネット上の最も重要な脅威の一つとなっている．ボットネット検出の一手段として悪性ドメイン名リストに基づく DNS クエリ監視手法が挙げられる [1]. この手法ではボットネット観測などによって得られる悪性ドメイン名リストに基づき，そのドメイン名の名前解決を行っているユーザをボットネット感染ホストとして特定する．しかしながら，そのような手法で取得された悪性ドメイン名リストはボットネットの全挙動を反映していない可能性がある．逆にいくつかのボットネットは接続性確認のために著名な検索サイトへアクセスすることがあり，このような場合悪性ドメイン名リストに著名サイトが登録されてしまう問題がある．本稿では，悪性ドメイン名リストにおけるその精度に関する上記課題を，ドメイン名とそのドメイン名の名前解決を行うユーザの関係からなるグラフを用いて向上する手法を提案する．試験的に評価し，悪性ドメイン名リストに掲載されていないドメイン名でワーム感染ホストが特有に名前解決を行うドメイン名を抽出出来ることを確認した.

キーワード DNS, Botnet，グラフカーネル

Evaluation of Black Domain List by Using DNS Query Graph

Keisuke ISHIBASHI†, Tsuyoshi TOYONO†, Kazumichi SATO†, and Makoto IWAMURA†

† NTT Information Sharing Platform Laboratories, NTT Corporation
Midori-cho 3–9–11, Musashino-shi, Tokyo, Japan

Abstract Botnet hosts, which can be controlled by malicious operators for executing DDoS attacks or spamming, have been one of the major concerns of Internet security. One of the promising approaches for detecting those hosts is monitoring DNS traffic and detecting botnet-infected hosts by using black domain names. However, black domain name lists obtained through these methods may have a problem in their accuracy in that they do not cover all domain names caused by botnets. In addition, they may contain domain names that are not involved with botnet activities. In this paper, we propose a method to improve the accuracy of a black domain name list by using a DNS query graph, which composed with nodes of hosts and domains edeges representing query-relationship. Intuitively, domain names resolved by hosts that resolve many black domain names are also expected to be black, and domain names resolved by many hosts that do not resolve any black domain names are expected to be white. Thus, the DNS query graph indicates to us which domain names might be black or white. We also propose approximating a graph kernel value with random walk sampling because calculating a graph kernel requires operation of adjacency matrices of graphs, which may be difficult for huge graphs. We experimentally applied the proposed method using a black domain name list and DNS traffic. The result shows that most of the white domains mislisted as black can be removed.

Key words DNS, Botnet, Graph Kernel

1. Introduction

DDoS やスパム，フィッシングに用いられるボットネットはインターネット上の最も重要な脅威の一つとなっている．ボットネット対策の一環として，ボットネット感染ホスト検出があり，その一手法として悪性ドメイン名リストに基づく DNS クエリ監視手法が挙げられる [1]. この手法ではボットネット検体解析などによって得られる，ボットネット感染ホストがアクセ

スする悪性ドメイン名リストを用いて，そのリスト中のドメイン名の名前解決を行っているユーザをボットネット感染ホストとして特定する．この手法ではボットネット感染ホストが送信する全トラヒックのうち，一部トラヒック監視のみで効率的にボットネット感染ホストを検出できるという利点がある．

しかしながら，ボットネットはその対策から逃れるために，検体，挙動を頻繁に変更しており，上記手法で取得された悪性ドメイン名リストであっても，時々刻々変遷するボットネットの全挙動を捕らえているわけではなく，ボットネットがアクセスするドメイン名をリストアップできていない可能性があり，この場合，ボットネット感染ホストの検出漏れにつながることになる．

逆にいくつかのボットネットはインターネット接続性確認，もしくはカモフラージュののために著名な検索サイトへアクセスすることがあり，このような場合悪性ドメイン名リストに人気サイトが登録されてしまう問題がある．この場合それら人気サイトにアクセスした通常のホストをボットネット感染ホストと誤検出することになる．

本稿では，悪性ドメイン名リストの精度に関する上記課題を，ユーザの DNS 名前解決情報を用いて向上する手法を提案する．ここで利用する DNS 名前解決情報はドメイン名とそのドメイン名の名前解決を行うユーザの関係であり，これはホスト，ドメイン名をノード，それらの間の名前解決関係をエッジとしてグラフ表現が可能であり，以降 DNS クエリグラフと呼ぶ (図 1)．

本提案方式は以下の仮定を用いる：

（1） 悪性ドメイン名の名前解決を行っているホストの多数から名前解決されるホストはやはり悪性の可能性が高い．

（2） 悪性ドメイン名リストのドメイン名であっても，悪性ドメイン名リストのドメイン名を全く名前解決行わない多数のホストから名前解決される場合は悪性でない可能性が高い．

筆者らは以前この仮定を同様に用いて大量メール拡散型ワーム感染ホストをベイズ推定により特定する手法を提案した [2]．しかしながら，上記手法では，上記仮定により発見された新たな悪性ドメイン名の情報を再帰的に利用することはしておらず，あくまであらかじめ与えた悪性ドメイン名リストとクエリ元ホストの共通度が高いドメイン名を悪性として抽出するのみであった．

本稿では，グラフ内のノード間類似性を与えるグラフカーネル手法を適用して悪性ドメイン名リストにリストされているドメイン名に類似しているドメイン名抽出等を試みる．この場合，悪性ドメイン名リストと直接クエリ元ホストを共有していないホストの類似度も計算することができ，結果的に再帰的に悪性ドメインを抽出できる可能性がある．さらに大規模グラフではグラフカーネル計算が困難であるため，その近似としてグラフ上のランダムウォークサンプリングによってその近似値を推定する手法も提案する．試験的に評価し，悪性ドメイン名リストに掲載されていないドメイン名でワーム感染ホストが特有に名前解決を行うドメイン名を抽出出来ることを確認した．

以下本稿の構成は以下の通りである．2. 節で背景として DNS に関する概略を述べ，さらに今回 DNS の名前解決グラフに適用するグラフカーネル手法を紹介する．3. 節でグラフカーネルを用いたドメイン名鑑類似度計算，およびドメイン名の悪性スコア計算法を述べ，4. 節にそのランダムウォークシミュレーションによる近似法を述べる．5. 節で提案方式を試行的に検証した結果を紹介し，6. 節で結論を述べる．

2. 背　　景

2.1 DNS 名前解決

Domain Name System(DNS) はクライアントホスト（スタブレゾルバ），キャッシュサーバ（フルサービスレゾルバ），権威サーバ（コンテンツサーバ）の三種類から構成される．クライアントホストは名前解決時に名前問い合わせをキャッシュサーバに送信し，キャッシュサーバは問い合わせドメイン名に対する権威サーバを検索し，権威サーバに対して名前問い合わせを送信する．問い合わせ内容としては，ドメイン名に対する IP アドレス (A レコード)，メールサーバ (MX レコード) や権威サーバ (NS レコード) が代表的である．権威サーバからの応答を受けたキャッシュサーバは，クライアントホストにその応答を返すとともに，その応答内で指定されている期間（Time to live, TTL）だけ応答内容をキャッシュし，TTL 内に受信した他のクライアントホストからの問い合わせに対してはキャッシュ内容を応答する．本稿では，このクライアントホストとそのホストが名前解決問い合わせを行ったドメイン名の関係を DNS クエリグラフと呼ぶ．

2.2 グラフカーネル

グラフカーネルはグラフ内のノード間類似性をその接続関係に基づいて与える手法である．いくつかのグラフカーネルが提案されており，テキストマイニングにおいて文書ー単語の二部グラフに適用し，類似単語の抽出などの応用が報告されている．[3]〜[7]．本稿ではドメイン名間の類似性に同手法を適用する．以下，いくつかのグラフカーネル手法を紹介する．

2.2.1 拡散カーネル

拡散カーネルにおけるノード間類似度は，熱源をあるノード

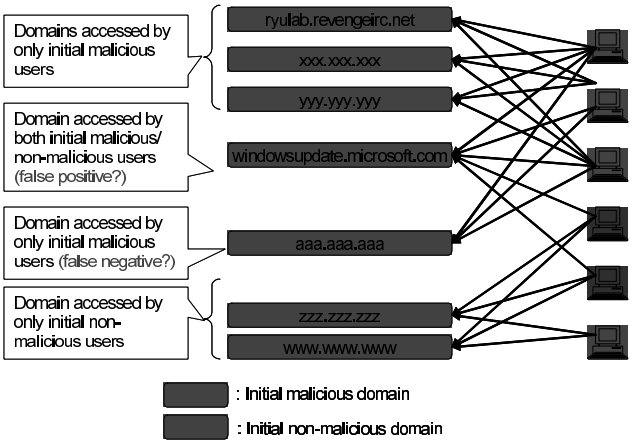

図 1　DNS Query Graph

に置き，グラフのリンク上に沿って熱を拡散させたときに一定時間後の別ノードの温度として考えることが出来る．具体的な拡散カーネル計算法は以下の通りである．$A = \{a_{ij}\}$ をグラフの隣接行列, D を対角行列でその値が A の列の和（ノード次数）となっている行列，また $L = A - D$ とする．このとき拡散カーネルは下記で与えられる：

$$K_D(\tau) := \exp(\tau L) = \sum_{n=0}^{\infty} \frac{(\tau L)^n}{n!} \tag{1}$$

ここで τ は拡散カーネルのパラメータであり，熱を拡散させる時間と解釈することが出来る．

$K_D(\tau)$ の (i, j) 要素は推移速度行列が L で与えられる連続マルコフ過程の τ 時間後のノード i からノード j への推移確率であり，拡散カーネルにおいてはこの値をノード i とノード j 間の類似度と見なす．τ が増大するにつれてホップ数が多いノード間の類似度も増加し，$\tau \to \infty$ で連結成分同士のノード間類似度は全て同一となる．

2.2.2 *ノイマンカーネル*

A, D を前節と同様とし，$N = AD^{-1}$ とする．N はグラフ上のマルコフ連鎖遷移確率行列である．このとき正規化ノイマンカーネルは下記で与えられる：

$$K_{NN}(\beta) := \frac{N}{E - bN} = N \sum_{n=0}^{\infty} (\beta N)^n. \tag{2}$$

ここで N が確率遷移行列なので，N^n の (i, j) 要素はノード i を起点とするグラフ上のランダムウォークが n ステップ後にノード j に到着する確率であり，$K_{NN}(\beta)$ の (i, j) 要素は，それら確率を β^n で重みづけた遷移確率の和となる．

2.3 カーネルの計算例

ここでは，カーネルの具体例を示すため，ドメイン名 100, ホスト数 1000 のダミーデータを生成し，各種カーネルを計算する．ドメイン名，ホストとも次数分布はべき分布として，著名ドメイン名，ヘビーユーザを生成し，クエリ先ドメイン名とクエリ元ホストの関係はランダムに生成した．図 2 に生成したドメイン名-ユーザの関係を示す．黒のノードがドメイン名，白のノードがユーザに対応している．

上記ドメイン名-ユーザ関係に関して，正規化ノイマンカーネル行列のドメイン部分のみの 100x100 行列を図 3,4,5 に示す．各セルの明るさがそのセルの値を示している．左から $\beta = 0.1, 0.5, 0.95$ の値である．β が小さいときは自ノード以外との類似度が低いが，β が大きくなるにつれて，他ノード（ドメイン名）との類似度が上昇していくのがわかる．またノード次数が高いドメインの類似度が上昇している．なお他ドメイン名と関連度度を持たないノードは孤立しているノード（他と共通ユーザを持たないドメイン名）である．

同様に拡散カーネル行列を図 6,7,8 に示す．左から $\tau = 0.01, 1, 100$ の値である (明度は差が見えやすいように変更している)．正規化ノイマンカーネルと同様の傾向が見られるが，類似度の上昇傾向が正規化カーネルと異なっており，ノード次数が高いドメインの類似度上昇の傾向は見られない．これはノード次数が高いドメイン名は他ノードへの推移率が高くなり，結果的に当該ノードの類似度上昇が平準化するためである．

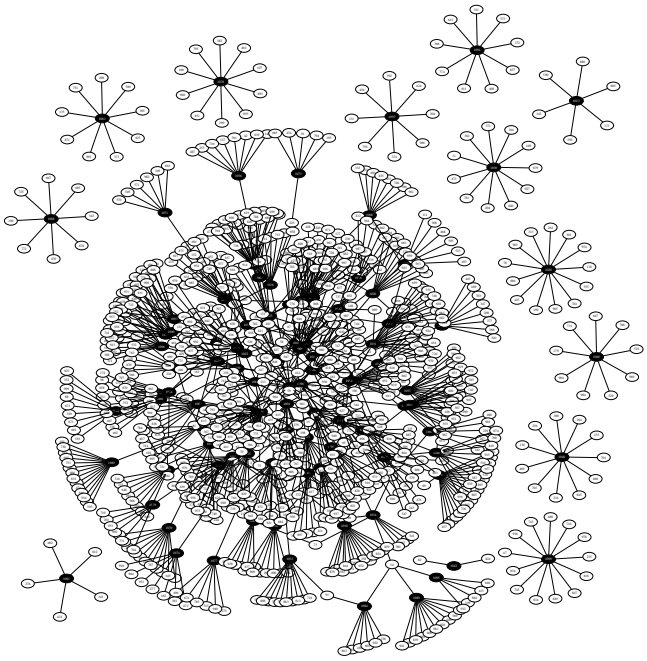

図 2　DNS Query Graph (Synthesized)

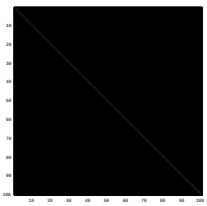

図 3　$\beta = 0.1$

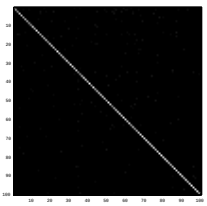

図 4　$\beta = 0.5$

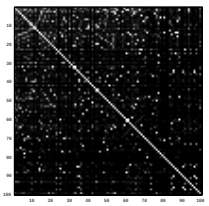

図 5　$\beta = 0.95$

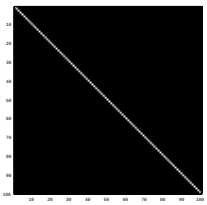

図 6　$\tau = 0.01$

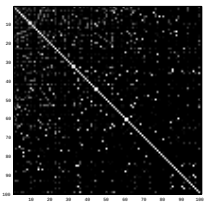

図 7　$\tau = 1$

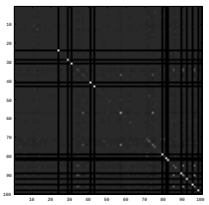

図 8　$\tau = 100$

3. ノード間類似度を用いた悪性度計算

本節では前節で述べたグラフカーネルによるノード間類似度を用いて，ドメイン名の悪性度計算を行う手法を述べる．ノード i,j 間の類似度を与えるグラフカーネル $K = \{k_{ij}\}$ を用いて，下記のとおり悪性ドメイン名リストのドメイン名との類似度の総和として，ドメイン名 j の悪性スコアを計算することが出来る：

$$b(j) := \sum_{i \in BL} k_{ij}, \tag{3}$$

ここで BL は悪性ドメイン名リスト内のドメイン名の集合である．このように計算した悪性スコアが高いドメイン名は悪性ドメインリストになくとも，悪性ドメイン名リストのドメインの

名前解決を行ったホストから名前解決されているために悪性の可能性が高いと考えることが出来る．

しかしながらこのスコア計算法では，アクセス数の高いポピュラーなドメイン名も悪性スコアが高くなってしまうという問題点が残る．これは，多数のホストから名前解決されるホストは，悪性ドメイン名リストのドメイン名を名前解決したホストからも名前解決され，結果的にこれらドメイン名との類似スコアが高くなってしまうためである．この問題を回避するため，1. 節で述べた仮定 2 を用いる．まず，良性スコアを下記式の通り悪性ドメイン名リストにないドメイン名との類似度の和で与える：

$$w(j) := \sum_{i \notin BL} k_{ij}. \tag{4}$$

さらに悪性スコアと良性スコアの比を下記で与える：

$$s(j) := \frac{b(j)}{w(j) + b(j)}. \tag{5}$$

悪性ドメインリストのドメインとのみ類似度が高いドメイン名はスコア比が高く，一方で，アクセス数の高いポピュラーなドメイン名は悪性スコアと良性スコア双方が高いためにスコア比は低くなると期待され，スコア比を取ることによって，悪性の可能性が高いドメイン名のみの抽出が可能になると期待される．

4. ランダムウォークシミュレーションによる推定

グラフカーネルを計算することによってドメイン名間の類似度を計算することが可能となり，悪性スコア，及びスコア比を計算することが出来る．しかしながら，グラフカーネルの計算には行列計算が必要となり，ドメイン名，ホスト数とも膨大となる大規模グラフでの計算は一般的に困難である．本稿では，グラフカーネルの値をランダムウォークシミュレーションによって近似することを考える．

ドメイン名 i を出発点とし，図 2 に示されるようなドメイン名-ユーザ，ユーザ-ドメイン名の関係のランダムウォークを考える．2. 2 節で述べた拡散カーネル，正規化ノイマンカーネルともマルコフ遷移の遷移確率行列から計算されるため，マルコフ遷移をグラフ上のランダムウォークによってシミュレーションし，遷移確率を推定出来ると考えられる．

4. 1 ノイマンカーネルのランダムウォークサンプリング

まず正規化ノイマンカーネルを疑似するランダムウォークサンプリングを考える．$N = AD^{-1}$ はこのグラフ上の離散マルコフ遷移の遷移確率行列を示すので，ノード i を起点とするランダムウォークの 2 ステップ先にドメイン名 j が現れる確率は N^2 の (i, j) 成分となる(注1)．従ってドメイン名 i から出発する n ステップのランダムウォークを多数生成し，n ステップ先にドメイン名 j が発生する頻度に β^{n-1} を乗じたものは，式 (2) の第 n 項 (i, j) 要素に収束する．$n = 1, \ldots$ に対してランダムウォークを実施し，上記の値の和を取ることによって正規化ノイマンカーネルの (i, j) 要素の値が得られる．

n に関する繰り返しを避けるためには，十分長いランダムウォークを生成し，その中の各ステップ毎にドメイン名の出現頻度を計算する手法もあるが，ランダムウォーク長を自動的に決定することも可能である．ドメイン名 i から出発し，次ノードに到達するたびに確率 $1 - \beta$ で終了するランダムウォークを生成する．終了確率 $1 - \beta$ のランダムウォーク長分布は $\Pr[\text{ランダムウォーク長} \geqq n] = \beta^{n-1}$ となるため，このランダムウォーク中におけるドメイン名 j の発生確率は，$\sum_{n=1}^{\infty} (\beta N)^n$ の (i, j) 成分となり，これは正規化ノイマンカーネル $K_{NN}(\beta)$ の (i, j) 成分に $1/\beta$ を乗じた値と一致する．従ってノード i を起点とする終了確率 $1 - \beta$ のランダムウォークを K 回生成し，その中におけるノード j の出現回数をカウントすることによって，正規化ノイマンカーネルの値 k_{ij} は以下で推定できる：

$$k_{ij} := \frac{1}{K\beta} \sum_{k=1}^{K} \text{ランダムウォーク中のノード } j \text{ 出現回数}.$$

従って，DNS クエリグラフの各ドメイン名を出発して，確率 $1 - \beta$ で終了するランダムウォークを十分多く行い，起点ドメイン名が悪性ドメインリストにあるか否か別に上記値を計算することで悪性，良性スコア，およびスコア比を計算できる．

4. 2 拡散カーネルのランダムウォークシミュレーション

次に拡散カーネルを疑似するランダムウォークサンプリングを考える．拡散カーネルは連続時間マルコフ過程の遷移確率なので，連続時間ランダムウォークシミュレーションを行えば遷移確率を推定できるが，ここでは実際のシミュレーションのために微小時間 Δt で離散化する．

推移速度行列が $L = A - D$ で与えられる連続マルコフ過程の微小時間 Δt 後のノード i からノード j への推移確率は d_i を対角行列 D の i 番目の成分，a_{ij} を A の (i, j) 成分として

$$\Pr[i \to i] = \exp(-d_i \Delta t)$$
$$\Pr[i \to j] = (1 - \exp(-d_i \Delta t)) \frac{a_{ij}}{d_i}$$

で与えられる．従って，1 ステップ先の遷移確率が Δt 時間後の推移確率に従う $\frac{\tau}{\Delta t}$ ステップのランダムウォークを生成し，最終ノードの出現頻度を計算することによって連続時間マルコフ過程の τ 後の推移確率（拡散カーネルの値）を得ることが出来ると考えられる．

5. 実 験 例

本節では実際の DNS トラヒックに対して提案手法を試行的に適用した結果を述べる．まず正規化ノイマンカーネルによる悪性ドメイン抽出として，316 ドメイン名からなる悪性ドメイン名リストを用い，それらリストに含まれるドメイン名を起点とし，次ドメイン名に到着すると確率 0.9 で終了するランダムウォークを 10,000 回，そうでないドメイン名に対しては 100 回生成し，グラフカーネルの値を推定した．

まず正規化ノイマンカーネルの近似計算による結果を示す表 1 は，悪性ドメイン名リストに含まれていないが，スコア比が高い上位 20 ドメイン名である．これらはマルウェア配布サイ

(注1)：DNS クエリグラフはドメイン名とホストの二部グラフなので，1 ステップ先にはドメイン名は表れない．

表 1　Score ratio of domains not in the black domain name list (top20)

Rank	$b(j)$	$w(j)$	$s(j)$	Domain (j)
1	0.089	1.200	0.069	nadsam0.info
2	0.062	1.110	0.053	proxim.ntkrnlpa.info
3	0.043	1.010	0.041	qualitydrug-store.com
4	0.041	1.000	0.039	fbi32.cheapdf.com
5	0.041	1.040	0.038	d.Felony-Productions.net
6	0.037	1.060	0.034	ksacoool.com
7	0.037	1.080	0.033	tassweq.com
8	0.027	1.040	0.025	pop.comcom2.com
9	0.027	1.050	0.025	20.20.140.59.in-addr.arpa
10	0.029	1.150	0.025	jns.dns02.com.ar
11	0.037	1.470	0.025	smtp3.google.com
12	0.037	1.480	0.024	smtp2.google.com
13	0.028	1.160	0.023	jns.dns01.com.ar
14	0.041	1.730	0.023	smtp1.google.com
15	0.025	1.090	0.022	ovbijkbqgr.hn.org
16	0.025	1.120	0.022	oegona.hn.org
17	0.022	1.030	0.021	webipcha.cn
18	0.022	1.010	0.021	fumegb.yi.org
19	0.024	1.100	0.021	nnkpuji.hn.org
20	0.022	1.020	0.021	ufyimtttqwh.afraid.org

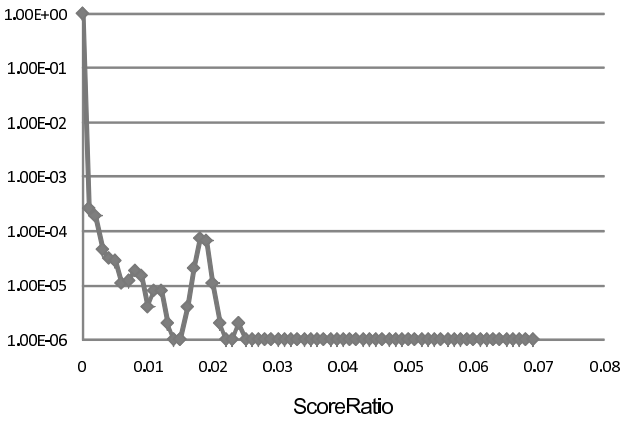

図 9　Histogram of Score Ratio

トなどであり，提案手法を用いることで，悪性ドメイン名リストにないドメイン名であるが，悪性と思われるドメイン名を抽出することが出来た．著名検索サイトのドメイン名も含まれているが，これらは MX レコードの問い合わせであった．通常のホストはメール送信は所属組織のメールサーバ経由で行うため，直接 MX レコードを問い合わせることはなく，スパム送信等が疑われるが，これらの分析は今後の課題である．

悪性ドメイン名リストにないドメイン名のスコア比分布を図 9 に示す．大多数のドメイン名はスコア比 0.01 以下の良性ドメイン名であるが，スコア比 0.02 付近にピークがあり，これらドメイン名は悪性であることが疑われる．

逆に悪性ドメイン名リストに含まれているがスコア比が低い下位 20 ドメイン名を表 2 に示す．検索サイトや OS アップデート配布サイトなどであり，提案手法により，悪性ドメイン名リストに含まれている人気ドメイン名を低いスコア比を持つドメイン名として排除できることがわかる．

表 2　Score ratio of domains in the black domain name list (top20)

Rank	$b(j)$	$w(j)$	$s(j)$	Domain (j)
1	1.259	558.330	0.002	wpad
2	1.100	232.460	0.005	www.google.com
3	1.064	153.730	0.007	download.windowsupdate.com
4	1.171	93.470	0.012	time.windows.com
5	1.176	64.900	0.018	www.microsoft.com
6	1.010	19.730	0.049	crl.microsoft.com
7	1.067	17.870	0.056	checkip.dyndns.org
8	1.013	10.820	0.086	download.microsoft.com
9	1.009	10.650	0.087	csc3-2004-crl.verisign.com
10	1.005	9.240	0.098	google.com
11	1.024	9.010	0.102	crl.verisign.com
12	1.008	8.230	0.109	www.yahoo.com
13	1.016	6.680	0.132	windowsupdate.microsoft.com
14	1.003	4.830	0.172	msn.com
15	1.017	4.050	0.201	mx1.hotmail.com
16	1.013	3.770	0.212	mailin-03.mx.aol.com
17	1.016	3.580	0.221	mailin-01.mx.aol.com
18	1.062	3.580	0.229	mailin-04.mx.aol.com
19	1.014	3.220	0.239	mailin-02.mx.aol.com
20	1.019	1.340	0.432	microsoft.com

同様に拡散カーネル手法のランダムウォークシミュレーションを実施した結果を表 3，4 に示す．シミュレーション回数は前節と同じであり，拡散パラメータは 10 とした．表 4 の悪性ドメインリストのドメインであるがスコア比下位のものは表 2 と重なりが多いが，表 3 の悪性ドメインリストのドメインであるがスコア比下位のものは表 1 と差違が見られる．

しかし，正規化ノイマンカーネルによって得られた図 9 で確認されたスコア比 0.2 付近のピークに存在するドメイン名に関して，拡散カーネルの結果と比較したところ，拡散カーネルの結果においてもスコア比上位に表れた．仮に図 9 においてスコア比 0.17 以上の 319 ドメイン名に関して拡散カーネルのスコア比上位 319 ドメイン名と比較したところ 92%が一致しており，これらドメイン名は悪性である可能性が高いと考えられる．

6. ま と め

本稿では，DNS クエリグラフを用いて悪性ドメイン名リストの精度向上する手法を提案した．

DNS クエリグラフ上のノード間類似度を計算し，悪性ドメイン名リストとそれ以外のドメイン名との類似度を比較することによって，リストに含まれていないが悪性度の高いドメイン，リストに含まれているが，人気ドメインで悪性度の低いドメイン名を抽出する．実トラヒックに適用することにより，提案手法の有効性を部分的に確認した．

文　献

[1] J. Kristoff, “Botnets, detection and mitigation: DNS-based techniques,” Information Security Day, Northwestern University, July, 2005.

表 3　Score ratio of domains in the black domain name list (top20)

Rank	$b(j)$	$w(j)$	$s(j)$	Domain (j)
1	0.204	0.001	0.997	fbi32.cheapdf.com
2	0.205	0.001	0.996	d.Felony-Productions.net
3	0.142	0.001	0.996	czyzmd.dynserv.com
4	0.155	0.001	0.996	qualitydrug-store.com
5	0.146	0.001	0.995	ufyimtttqwh.afraid.org
6	0.144	0.001	0.995	fumegb.yi.org
7	0.126	0.001	0.995	fynfrmil.hn.org
8	0.127	0.001	0.995	wmzztcd.3-a.net
9	0.125	0.001	0.995	zqptrlup.dynserv.com
10	0.157	0.001	0.995	ksacoool.com
11	0.127	0.001	0.995	hhviuirfg.hn.org
12	0.127	0.001	0.994	svoawdtq.afraid.org
13	0.126	0.001	0.994	kbgapiqvmp.afraid.org
14	0.123	0.001	0.994	gojzqd.1dumb.com
15	0.127	0.001	0.994	zibjyomx.hn.org
16	0.126	0.001	0.994	shqkbrhyq.dynserv.com
17	0.156	0.001	0.994	tassweq.com
18	0.144	0.001	0.994	yotppbxot.yi.org
19	0.126	0.001	0.994	wsytgqh.afraid.org
20	0.126	0.001	0.994	awohxoxhz.afraid.org

表 4　Score ratio of domains in the black domain name list (top20)

Rank	$b(j)$	$w(j)$	$s(j)$	Domain (j)
1	0.385	1.256	0.235	download.windowsupdate.com
2	0.387	1.257	0.235	www.google.com
3	2.718	8.492	0.242	wpad
4	0.038	0.113	0.249	crl.microsoft.com
5	0.013	0.039	0.251	csc3-2004-crl.verisign.com
6	0.008	0.021	0.267	google.com
7	0.243	0.590	0.291	time.windows.com
8	0.030	0.067	0.312	download.microsoft.com
9	0.178	0.366	0.327	www.microsoft.com
10	0.025	0.051	0.331	crl.verisign.com
11	0.002	0.004	0.351	msn.com
12	0.012	0.020	0.386	www.yahoo.com
13	0.044	0.050	0.471	windowsupdate.microsoft.com
14	0.002	0.002	0.520	mailin-03.mx.aol.com
15	0.003	0.002	0.559	mailin-02.mx.aol.com
16	0.004	0.002	0.597	mx1.hotmail.com
17	0.003	0.002	0.601	mailin-01.mx.aol.com
18	0.001	0.001	0.619	www.whatismyip.org
19	0.231	0.122	0.654	checkip.dyndns.org
20	0.003	0.001	0.804	www.showmyip.com

[2] K. Ishibashi, et al., “Detecting Mass-Mailing Worm Infected Hosts by Mining DNS Traffic Data,” Workshop on mining network data (MineNet-05), August 2005.

[3] R. I. Kondor, “Diffusion Kernels on Graphs and Other Discrete Input Spaces,” ICML 2002.

[4] J. Kandola, et al., “Learning semantic similarity.” NIPS, 2003.

[5] T. Ito, et al., “Application of kernels to link analysis,” ACM KDD, 2005.

[6] 鹿島久嗣, “カーネル法による構造データの解析,” 信学技報 (NLC2004-126),2005 年 2 月.

[7] 小町守, 工藤拓, 新保仁, 松本裕治, “カーネル法を用いた意味的類似度の定義と ブートストラップの一般化,” 言語処理学会第 14 回年次大会　2008 年 3 月 20 日.