既知の悪性 URL 群と類似した特徴を持つ URL の検索

孫 博† 秋山 満昭‡ 八木 毅‡ 森 達哉†

†早稲田大学 基幹理工学部
169-8555 東京都新宿区大久保 3-4-1
{sunshine,mori}@nsl.cs.waseda.ac.jp

‡NTT セキュアプラットフォーム研究所
180-8585 東京都武蔵野市緑町 3-9-11
{akiyama.mitsuhiro, yagi.takeshi}@lab.ntt.co.jp

あらまし ドライブバイダウンロード攻撃やフィッシング等の Web ユーザを標的とした脅威に対する有効な対策手段として URL ブラックリストが広く利用されている．一般に URL ブラックリストの構築とは個々の URL に対して悪性である場合にそのことを示すラベルを付与する処理である．インターネットにはラベルが付与されていない未知の URL が非常に大量に存在するので，それらの URL を整理し，既存の悪性 URL に対して類似検索を可能にする事は有意義である．本研究は任意の悪性 URL 群と類似した URL を検索する方法を提案する．応用例として，特定の Exploit kit を利用したことがわかっている悪性 URL 群が手元にあるとき，それらの悪性 URL 群と近い性質を持った URL を未知の URL 群から検索するケースなどが考えられる．検索結果は類似度スコアでソートすることが可能である．本研究ではそのような類似要素検索を実現するアルゴリズムとして Bayesian Sets を用いる．類似性を判定するための特徴としては，URL 構成文字列，ドメイン情報，IP アドレス等の様々な情報を総合的に利用する．実データを用いて提案手法の有効性を検証した結果および応用例を報告する．

Searching URLs that have similar features to the existing malicious URLs

Bo Sun† Mitsuaki Akiyama‡ Takeshi Yagi‡ Tatsuya Mori†

†Waseda University
3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555
{sunshine,mori}@nsl.cs.waseda.ac.jp

‡NTT Secure Platform Laboratories
3-9-11 Midoricho, Musashino-city, Tokyo 180-8585
{akiyama.mitsuhiro, yagi.takeshi}@lab.ntt.co.jp

Abstract URL blacklists/blocklists are widely deployed as an effective means to mitigate various threats targeting web users; e.g., Drive-by-download attacks, phishing scam, etc. Broadly speaking, building a blacklist is a process of picking up malicious URLs to label them. Since there are enormous amount of *unlabeled* URLs in the wild, it is useful to organize them and make it *searchable* to the set of existing malicious URLs. This work proposes a novel method that enables such search. An example application of the method is as follows. Given a set of known malicious URLs that make use of a particular exploit kit, the method searches similar URLs to those malicious URLs from a set of unknown URLs. The search results can be sorted with similarity scores. In this work, we adopt the Bayesian sets as a search algorithm that detects similar items to a given set of items. We extract various features such as characterisrtics of strings in URLs, IP address, domain name information etc. We validate and demonstrate the effectiveness of our approach using the real world URLs.

1　はじめに

背景: Google，Amazon，Facebook，Akamai に代表されるインターネットの “Hyper giants” の登場を契機に，インターネットにおける情報流の中心は再び Web に収斂つつある [1]．これに伴いサイバー犯罪もやはり Web が中心的な舞台となった．Web を舞台としたサイバー犯罪の手口は様々であるが，最初のトリガーとなるコンピュータへの侵入方法として Web ブラウザの脆弱性を標的としたドライブバイダウンロード攻撃が今なお盛んに利用されている．最近の事例として，ドライブバイダウンロード攻撃を実行するクライムキットの一つである Magnitude Exploit Kit は `PHP.net` への Web アクセスや Yahoo の広告スペースで利用され，一週間で約 600 万円以上の収益を不正に獲得したことが報告されている [2]．

現在 Web セキュリティの脅威を防ぐための手段として URL Blacklist が広く活用されている．URL Blacklist は個々の URL の評判情報を管理するものであり，予め悪性であることがわかっている URL をリストとして管理し，リストにマッチする URL へのアクセスを遮断する．URL Blacklist の精度やスケーラビリティを向上するための様々な研究が行われている．文献 [3] でも指摘されているように，広大な Web 空間に対して有効な URL Blacklist を構築する上で鍵となるのは

- 大規模データに対応すること
- 未知 URL から悪性 URL を発見すること

を満たすことである．そのためには，未知 URL を何らかの基準に沿って整理し，既知の知識に対して**検索可能**にすることが重要である．

提案: 本研究は任意の悪性 URL 群と類似した URL を検索する方法を提案する．従来より既知の悪性 URL の特徴を学習し，未知の悪性 URL を検出する技術が提案されてきた．これに対し，本研究は特定の性質を持つことが予めわかっている悪性 URL の集合に対し，その集合に属する確率が高い URL を未知の URL 群から検索することに新規性がある．すなわち単純な悪性・良性の判定ではなく，悪性の中でも特定の性質をもつ URL を抽出することに特徴がある．具体的な応用例として，Web クライアント型ハニーポット等の適用によって特定の Exploit kit を利用することがわかっている悪性 URL 群が手元にあるとき，それらの URL と近い性質を持った URL を検索することで，同じ Exploit kit を利用している可能性が高い URL を収集できる．そのような類似 URL を優先的に解析することにより，攻撃の特徴や詳細を早期に明らかにすることが期待できる．

本研究ではそのような類似要素検索を実現するアルゴリズムとして Bayesian Sets [4] を用いる．類似性を判定するための特徴としては，URL 構成文字列，ドメイン情報，IP アドレス等の様々な情報を総合的に利用する．実データを用いて提案手法の有効性を検証した結果および応用例を報告する．

貢献: 本研究の主要な貢献は下記のとおりである．

- 未知の URL 群から既知の悪性 URL 群と類似した URL を抽出する検索技術を提案した
- 同一 Exploit kit を利用した悪性 URL の検索，フィッシング URL の検索に関して精度良く検索ができること，および検索結果の具体例を示した．

本論文の構成は以下の通りである．はじめに 2 章では関連研究と本研究の比較を示す．次に 3 では本研究の提案手法を示す．4 章で提案手法の評価に用いるデータの詳細を述べた後，5 章にて提案手法の評価結果を示す．6 章は本研究のまとめと今後の展望を示す．

2　関連研究

悪性 URL の検知・分類では様々な手法が提案されている．以下では手法として機械学習を利用する技術，および利用しない技術の 2 種類に分けて整理し，本研究との比較を示す．

教師あり機械学習を利用する技術

以下に示す研究はいずれも教師あり機械学習のアプローチにより悪性 URL を検知する技術である．それぞれどのような特徴を使ったか，どのようなアルゴリズムを使ったかという点で整理する．Choi ら [5] は特徴として URL が他のサイトに引用される回数，Web コンテンツ，DNS，DNS Fast Flux，ネットワーク通信等，様々な特徴を使った手法を提案した．Ma ら [6] は特徴として URL 文字列とホスト情報を利用し，複数の教師あり機械学習を用いた性能評価を試みた．この結果，学習時間および誤検知率の観点からロジスティック回帰が最適であるという結論を導いた．Eshete ら [7] は主に URL 文字列や Web コ

ンテンツ特徴とし，やはり複数の教師あり機械学習アルゴリズムによる性能比較を行っている．実験の結果としては Random Tree の検知率が最良であることを示した．Xu ら [8] はネットワーク，ブラウザ，サーバ等の複数レイヤーで取得したデータを元に悪性 URL を検知する方式を提案した．主成分分析，Correlation feature selection (CFS)，情報利得等を利用して特徴選択を試みている．Canali らが開発した Prophiler [9] は，ハニーポットの悪性 URL 検知の負荷を低減することを目的として，クローラーで収集した URL から良性 URL を取り除くシステムである．特徴として HTML，Javascript，URL 文字列等の特徴を利用し，様々な教師あり機械学習アルゴリズムを利用した評価を行った．この結果，J48 決定木が最良の結果を与えたことを報告している．上述の研究はいずれもオフラインでのバッチ処理を前提としたものであるためオンラインで学習モデルを更新することができない．そこで Ma ら [10] はストリーム処理であるオンライン学習を適用することにより学習モデルをリアルタイムで更新する技術を提案した．

上述の関連研究は，いずれも教師有りの機械学習を利用しているため，事前にラベル付きの教師データを学習する必要がある．より良い検知率を得るためには大量の “ground truth” を用意しなくてはならない．しかし，教師データの収集はコストが高い問題がある．既知のブラックリストは時間が経過すると通信が不可能となり，それ以上の情報が収集できなくなる問題もある．本研究は，比較的少量の教師信号を用い，類似した悪性 URL を未知 URL 群から検索できることに利点がある．

機械学習を利用しない技術

Invernizzi ら [11] は Canali らと同様な目的意識のもと Evilseed システムを開発した．Canali らが複数の教師有り機械学習アルゴリズムを適用するのに対し，Invernizzi らは Google，Bing，Yacy 等の検索エンジンを悪性 URL の探索に活用した．検索用のクエリに関しては，Google’s Safe Browsing ブラックリストと Wepawet ハニーポットによって判定された悪性 URL を収集しておく．その後，それらを 5 種類の観点（ハイパーリンク，検索クエリ，SEO，度々メイン登録，DNS クエリ）から，主に検索エンジンを用いて悪性 URL に類似する未知の URL を収集する．しかし，EvilSeed は検索エンジンに強く依存するため，検索エンジンにインデックスされない悪性 URL を取得することができないというデメリットがある．

3 類似 URL 検索

本章では類似 URL 検索技術の詳細を示す．はじめに類似アイテム検索技術である Bayesian Sets の概要を示した後，Bayesian Sets が仮定するベルヌイ分布モデルに適合する形で URL の特徴を抽出する方法を示す．

3.1 Bayesian sets の概要

Bayesian Sets [4] は Google Sets [12] に触発されて Ghahramani らが開発したアルゴリズムである．Google Sets はユーザが入力した複数のクエリ集合に対し，それらの集合と関連が高いと考えられる応答を出力するサービスである [1]．例えばユーザが入力したクエリ集合: “Toyota”，“Nissan”，“Honda” に対し，Google Sets は “BMW”， “Ford”， “Audi”，“Mitsubishi”， “Mazda”， “Volkswagen”， ... といった文字列を関連度が高い順に出力する．

Ghahramani らは Google Sets の入出力をオンデマンド・クラスタリングの問題として定式化した．すなわち，ユーザが与えたクエリは何らかの共通な特徴を有するクラスタの部分集合であり，そのように定義されたクラスタの要素を確度が高い順に列挙する問題であると捉えることができる．ユーザが与えるクエリの組み合わせによって任意のクラスタを決定する点が面白い点である．彼らはそのような問題を解くためのアルゴリズムを Bayesian Sets と名づけている．以下では Bayesian Sets の概要を我々の問題の文脈で示す．$\mathbf{D}$ を URL 全体の集合とし，$\mathbf{x} \in \mathbf{D}$ を集合に属する個々の URL とする．ユーザはクエリとして比較的小さな URL の集合 $\mathbf{Q} \subset \mathbf{D}$ を用意する．クエリ集合 $\mathbf{Q}$ が与えられた条件の元で，$\mathbf{Q}$ と $\mathbf{x}$ の関連性の高さを測るメトリクスとして，以下の様なスコア S を導入する．

$$S(\mathbf{x};\mathbf{Q}) = \frac{P(\mathbf{x},\mathbf{Q})}{P(\mathbf{x})P(\mathbf{Q})} = \frac{P(\mathbf{x}|\mathbf{Q})}{P(\mathbf{x})}$$

Bayesian Sets のアルゴリズムは与えられた $\mathbf{Q}$ を使って $\mathbf{x} \in \mathbf{D}$ に対して上記のスコアを計算し，スコアが高い順に $\mathbf{x}$ を出力する．

[1] Google Sets は 2014 年 8 月現在，Google Sheets も含めてサービスを停止している．

i 番目の URL の特徴ベクトルを $\mathbf{x}_i = \{x_{i1}, \ldots, x_{im}\}$ とする．m は各々のアイテムが持つ特徴の数である．特徴ベクトルの要素は $x_{ij} \in \{0,1\}$ $(1 \leq j \leq m)$ の二値変数であり，パラメタ θ_j のベルヌイ分布: $P(x_{ij}|\theta_j) = \theta_j^{x_{ij}}(1-\theta_j)^{1-x_{ij}}$ でモデル化する．このときスコアは以下のように計算できる．

$$S(\mathbf{x}_i; \mathbf{Q}) = \frac{P(\mathbf{x}_i|\mathbf{Q})}{P(\mathbf{x}_i)} = \frac{\int P(\mathbf{x}_i|\theta)P(\theta|\mathbf{Q})d\theta}{\int P(\mathbf{x}_i|\theta)P(\theta)d\theta}$$

ベルヌイ分布のパラメタ θ の共役事前分布はベータ分布 $B(\alpha, \beta)$ である．このときスコア関数はハイパーパラメタ α, β を用いて以下のような簡便な式へと変形することができる [4].

$$\begin{aligned} S(\mathbf{x}_i; \mathbf{Q}) &= \frac{P(\mathbf{x}_i|\mathbf{Q}, \alpha, \beta)}{P(\mathbf{x_i}|\alpha, \beta)} \\ &= \prod_{j=1}^{m} \frac{\alpha_j + \beta_j}{\alpha_j + \beta_j + N} \left(\frac{\tilde{\alpha_j}}{\alpha_j}\right)^{x_{ij}} \left(\frac{\tilde{\beta_j}}{\beta_j}\right)^{1-x_{ij}} \end{aligned}$$

ここで $N = |\mathbf{Q}|$ であり，

$$\begin{aligned} \tilde{\alpha_j} &= \alpha_j + \sum_{\mathbf{x}_i \in \mathbf{Q}} x_{ij} \\ \tilde{\beta_j} &= \beta_j + \sum_{\mathbf{x}_i \in \mathbf{Q}} (1 - x_{ij}) \end{aligned}$$

である．実際に計算するスコアとしては対数をとった $\log S(\mathbf{x}_i; \mathbf{Q})$ を使うと便利である．ハイパーパラメター α, β は観測データから経験的に決めるものであり，例えば x_{ij} の全データにわたる平均値 $m_j = \sum_{\mathbf{x}_i \in \mathbf{D}} x_{ij}/|\mathbf{D}|$ を用いて $\alpha_j = cm_j$，$\beta_j = c(1-m_j)$ のように定める．ベータ分布の平均値は $\alpha_j/(\alpha_j+\beta_j)$ であり，これは m_j と一致する．本研究では [4] に習い $c = 2$ とした．アルゴリズムとしては与えられた全 URL の集合 $\mathbf{D}$ より予め α, β を計算しておき，与えられたクエリ集合 $\mathbf{Q}$ を用いて $\tilde{\alpha}, \tilde{\beta}$ を計算することによってスコアを計算できる．

3.2 URL の特徴抽出

本研究では URL の特徴として，URL の文字列のおよび URL に含まれる FQDN に対応する IP アドレスの特徴を利用する．特徴はドメインの whois 情報や，実際にダウンロードされるコンテンツの中身等を考慮することで更に拡張することが可能であるが，今後の課題とする．抽出した URL の特徴を表 1 に示す．Bayesian Sets のアルゴリズムはベルヌイ分布を用いて特徴ベクトルの要素をモデル化するため，すべての特徴を二値表現する必要がある．URL 長や平均トークン長等，整数や実数で表現される特徴に関しては数値の区間を量子化し，どの区間に存在したかで二値表現する手法を採用した．

表 1: 本研究で用いる URL の特徴．

No.	特徴のカテゴリ・説明	特徴数
1	URL 文字列長	11
2	ドメイン文字列長	11
3	パス文字列長	11
4	URL 文字列に含まれる数字の数	1
5	ドメイン文字列に含まれる数字の数	1
6	パス文字列に含まれる数字の数	1
7	パス文字列のトークン数	8
8	パス文字列の平均トークン長	8
9	パス文字列の最長トークン長	8
10	URL 文字列が “exe” を含むか	1
11	ドメインが IP アドレスか	1
12	IP アドレスの上位 24 ビット	24
	合計	86

No. 1〜3 の URL 特徴については全体データ $\mathbf{D}$ から計算可能な中間値と比較して大きいか小さいかという二値に加え，長さを p パーセンタイル値 $(p = 0, 10, 20, \ldots, 100)$ を用いて 10 個の区間に量子化し，どの区間に存在したかで数値を二値化する．例えばある URL の長さ x が 93.4 パーセンタイル値であるとき，その URL 長の特徴は 11 個の特徴で $\{1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1\}$ と二値表現できる．最初の 1 は x が中間値よりも大きいことを示し，最後の 1 は x が 90 パーセンタイル値から 100 パーセンタイル値の区間に存在することを示す．このように複数の解像度で数値を表現することにより，単純な大小関係に加えて実際の値としてどの程度かという両方の情報を表現することができる．また中間値，パーセンタイルは比較的安定した統計値であるため，一度計算しておけば数値を頻繁に更新する必要はない．

No. 4〜6 の特徴については中間値との大小関係を二値で表現している．No. 7〜9 の特徴では，”&”, ”%”, ”?”, ”/”, ”=”, ”-”, ”_”, ”.” の 8 種類の文字をデリミタとして文字列トークンを作成し，トークン数，平均トークン長さ及び最長トークンの長さを中間値との大小関係で二値表現する．最後に No. 12 の IP アドレスは上位の 24 ビットをそのまま 24 個の二値データとして利用する．

特徴 No. 1〜3 は Ma ら [6] の先行研究で利用し

た特徴を参考にし，パーセンタイル値による量子化の拡張をしている．No. 4〜11 の特徴は Xu ら [8] および Canali ら [9] の先行研究でも利用されたものに拡張を施している．最後に，本研究は同じ IP プレフィックスを有する悪性 URL が多く存在することを考慮した上で No. 12 の特徴を採用した．

4 データ収集環境とデータセット

この章では本研究の評価に用いたデータセットの収集方法およびデータの統計値を示す．

4.1 悪性 URL

提案手法では，悪性 URL 群と類似した URL を検索する．このため，本評価では，表 2 に示すように，複数の悪性 URL 群を用意した．

ドライブバイダウンロード攻撃において, マルウェア感染攻撃 URL 群は下記の URL により構成される．

- アクセスしたユーザを後に記述する攻撃 URL にリダイレクトする入口 URL
- Web ブラウザおよびプラグインの脆弱性を用いて攻撃コードを実行する攻撃 URL
- 攻撃コードを用いてダウンロードさせるマルウェアを配置したマルウェア配布 URL
- ダウンロードしたマルウェアを実行して挙動を解析した過程において新たなマルウェアをダウンロードした際のダウンロード URL

本データは，ドライブバイダウンロード攻撃を正確に検知して攻撃に関わる URL 群を特定できる Web クライアント型ハニーポット Marionette [13] と，他者への攻撃を防止しつつマルウェア解析環境をインターネットに接続して解析できるサンドボックス BotnetWatcher [14] を用いて，2011 年 8 月 2 日から 2014 年 7 月 12 日までの期間に収集した．

一方，フィッシング URL 群は，過去フィッシングに悪用されて公開サイト Phishtank [15] に掲載されている URL で構成される．本データは，2008 年 6 月 24 日から 2014 年 8 月 11 日までの期間に収集された URL であるが，2013 年までに悪用された URL 数は 1,427 程度であり，2014 年に悪用された URL 群が中心となっている．

表 2: 悪性 URL 群の内訳

悪性 URL 群	種別	URL 数
マルウェア感染攻撃 URL 群	入口 URL	823
	攻撃 URL	2,495
	マルウェア配布 URL	6,636
	ダウンロード URL	1,674
フィッシング URL 群	フィッシング URL	16,835

表 3: 良性 URL 群の内訳

良性 URL 群	URL 数
ALEXA	30,000
DMOZ	20,973

4.2 良性 URL

良性 URL のリファレンスとして良く利用される Alexa [16] および DMOZ [17] を用いる．前者はユーザのアクセス数が高いトップサイトであるのに対し，後者は世界中のボランティアエディタによって構築管理されている最大の Web ディレクトリである．

URL 文字列の特徴として，前者は FQDN のみで構成されるのに対し，後者は FQDN に加えてパス部を含んでいることが挙げられる．一般に悪性 URL はパス部を含むことが多いため，本研究では両者を半々の割合で混在させるアプローチをとった．表 3 に良性 URL 群の内訳を示す．

5 結果

本章では 3 つのケーススタディを通して提案方法の有効性を示す．表 2 に示されたカテゴリーの内，特に攻撃 URL (Exploit)，マルウェア配布 URL，およびフィッシングを対象とし，検索を行う．またそれぞれのカテゴリの検索には $|\mathbf{Q}| = N = 3$ とし，3 つの異なる URL をある特徴を持つ URL 群としてクエリで指定する．この数値は経験的に決定するものであるが，クエリに含まれる特徴にばらつきが無い方が良い結果を得られる可能性が高い．

5.1 クエリパターンの比較

Bayesian sets を用いたアプローチではいかに良いクエリを生成するかが良い結果を得るための重要な鍵となる．ここではそれぞれのカテゴリーに応じてどのようなクエリパターンが良いかを考察する．

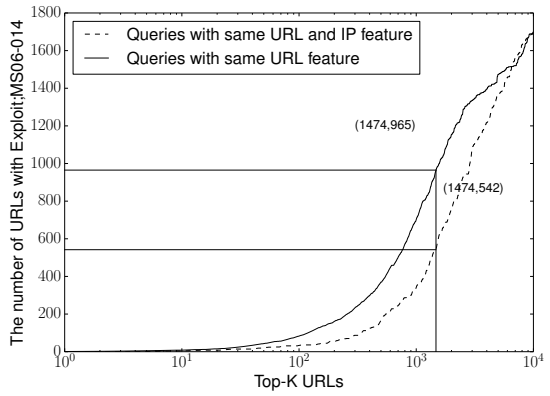

図 1: クエリパターンの違いが検索能力に与える影響の比較

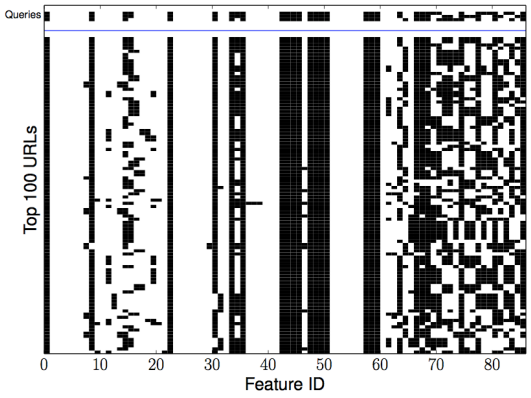

図 2: 攻撃 URL のクエリおよび検索結果の特徴ベクトル

検索に利用するクエリに関し，本研究の場合は下記のようなパターンを考えることができる．

- 類似した URL 文字列と異なる IP プレフィックスを持つ (U クエリパターンと呼ぶ)
- 類似した URL 文字列と同じ IP プレフィックスを持つ (UI クエリパターンと呼ぶ)
- 異なる URL 文字列と同じ IP プレフィックスを持つ (I クエリパターンと呼ぶ)

今回用いたデータでは IP プレフィックスが同じである場合，必ず URL 文字列に類似性があったため I クエリパターンとなる URL の候補は存在しなかった．攻撃 URL (Exploit MS06-014) に関して U クエリと UI クエリを構成して得られる検索結果の比較を図 1 に示す．検索の上位 URL に同攻撃 URL が含まれる割合は UI クエリパターンよりも U クエリパターンの方が高い．これは IP アドレスのばらつきを許容することで同じ Exploit Kit が複数のネットワークで利用されるケースを抽出可能になるからである．これはマルウェア配布 URL に関しても同様であった．フィッシング URL 群については U クエリパターンと UI クエリパターンの検索結果では UI クエリパターンの検索精度が高い結果を得た．

以上の結果をもとに，以下の実験では攻撃 URL，マルウェア配布 URL に関しては U クエリパターンを，フィッシング URL に関しては UI クエリパターンを利用する．

5.2 類似 URL 検索結果の詳細

攻撃 URL に関して前述したクエリパターンにしたがって任意に選択した 3 つの URL をクエリとして類似 URL を検索した結果を表 4 に示す．いずれも同じ Exploit Kit を利用した URL をクエリとして検索をかけたところ，同じ Expoloit Kit を利用した URL が検索されていることがわかる．ここでクエリで与えた URL の FQDN はいずれも異なり，それらに対応する A レコードの IP アドレスも異なる．IP アドレスの不一致性は検索結果にも反映されていることがわかる．一般に特定の Exploit Kit は同時期に利用されることが多いが，今回のクエリはいずれも 2011 年 12 月に収集したデータを使っている．いくつか例外もあるが，検索された URL もほぼ同時期に収集した URL であることがみてとれる．

図 2 は攻撃 URL のクエリおよび検索結果の URL に関して特徴ベクトルを可視化したものである．この図からも類似した特徴を持つ URL が検索できていることがみてとれる．

5.3 精度評価

前節で示した 3 つのケーススタディについて検索精度の評価を行う．これらはあくまでも例であり，他のクエリについても実施することが出来る．精度の評価として，検索結果の上位 10,000 件までにクエリと同じカテゴリの悪性 URL が含まれる検索ヒット率，および表 2 に示したすべての種別を含む広義

表 4: 攻撃 URL の検索結果

クエリ URL	種別	IP アドレス	取得日時
http://at1*****.com/main.php?page=045f7cd5dec4982a	攻撃 URL(Exploit;MS06-014)	79.137.237.**	2011/12/20
http://best*****.in/main.php?page=b1283ebc4a63b98d	攻撃 URL(Exploit;MS06-014)	78.111.51.**	2011/12/17
http://be*****.com/main.php?page=b778fa3b104bac2c	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/17
検索結果	種別	IP アドレス	取得日時
http://best*****.in/main.php?page=b1283ebc4a63b98d	攻撃 URL(Exploit;MS06-014)	78.111.51.**	2011/12/17
http://best*****.in/main.php?page=b1283ebc4a63b98d	攻撃 URL(Exploit;MS06-014)	78.111.51.**	2011/12/17
http://at1*****.com/main.php?page=045f7cd5dec4982a	攻撃 URL(Exploit;MS06-014)	79.137.237.**	2011/12/20
http://sky*****.net/main.php?page=ce57441e61ae2f12	攻撃 URL(Exploit;MS06-014)	46.137.87.**	2012/1/19
http://nh***.**x.com/main.php?page=403b4703851aa7ac	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/31
http://be*****.com/main.php?page=b778fa3b104bac2c	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/17
http://be*****.com/main.php?page=111d937ec38dd17e	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/17
http://the*****.net/main.php?page=5a56c997ffff2f79	攻撃 URL(Exploit;MS06-014)	46.249.59.**	2012/9/13
http://46.37.169.**/main.php?page=22d60c9a87becdf3	攻撃 URL(Exploit;MS06-014)	46.37.169.**	2012/5/12
http://diki*****.in/main.php?page=cc069b9ff187de82	攻撃 URL(Exploit;MS06-014)	79.137.237.**	2011/12/13
http://22pr*****.com/main.php?page=bbd8c7cc65c2cfb5	攻撃 URL(Exploit;MS06-014)	78.111.51.**	2012/4/26
http://pa***.**me/main.php?page=5a56c997ffff2f79	攻撃 URL(Exploit;MS06-014)	46.249.59.**	2012/9/25
http://pix***.**.tc/main.php?page=38b16bc50912741c	攻撃 URL(Exploit;MS06-014)	109.235.49.**	2012/3/8
http://lea*****.dumb1.com/main.php?page=5d05bfd9c0309a4b	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/27
http://dikar*****.c0m.li/main.php?page=45b88a869c6c23ae	攻撃 URL(Exploit;MS06-014)	46.249.37.**	2011/12/17

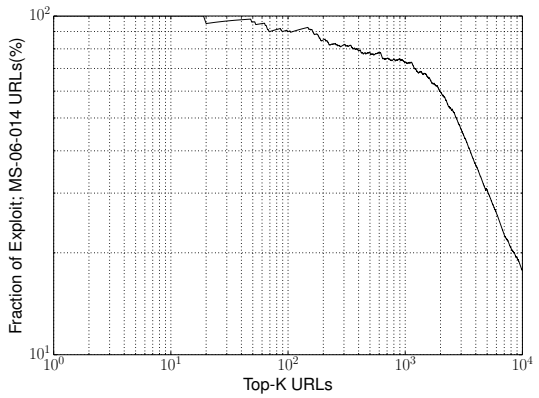

図 3: 攻撃 URL のヒット率

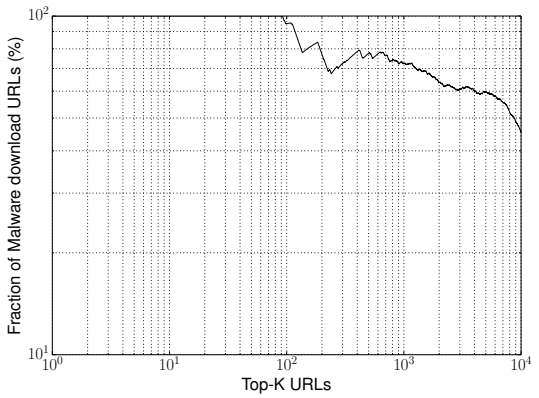

図 4: マルウェア配布 URL のヒット率

の悪性 URL が含まれる検索ヒット率を用いた分析を行う．

図 3 は攻撃 URL をクエリとして検索した際の攻撃 URL の検索ヒット率を示す．ヒット率が高いほど検索精度が高いといえる．検索スコアの上位 20 件まではヒット率が 100%であり，上位 1,000 件においても 70%の高いヒット率を維持している．同様にマルウェア配布 URL の検索ヒット率を図 4 に示す．検索上位 100 件までヒット率が 100%であり，上位 1,000 件においても 70% 超の比較的高い検索率を維持している．以上のように攻撃 URL とマルウェア配布 URL のクエリにおいては，クエリと類似した URL を精度よく検索できることがされた．一方，図 5 から見てとれるようにフィッシング URL 群の精度は不安定であり，上位 20 件までに良性 URL あるいは異なる種別の悪性が 8 件現れた．この結果より，類似したフィッシング URL を URL 文字列および IP アドレスのみから精度良く抽出することは困難であることが示唆される．様々な特徴を追加することによってフィッシング URL の検索精度を上げることは今後の課題である．

図 6 は 3 つのケーススタディについての悪性 URL のヒット率である．攻撃 URL とマルウェア配布 URL をクエリタイプとする場合，上位 10,000 件までにも高い精度を保っており，95%以上となっている．攻撃 URL とマルウェア配布 URL タイプのクエリは良性悪性 URL の判定に有効であることが示唆されている．一方，フィッシング URL 群の精度はフィッ

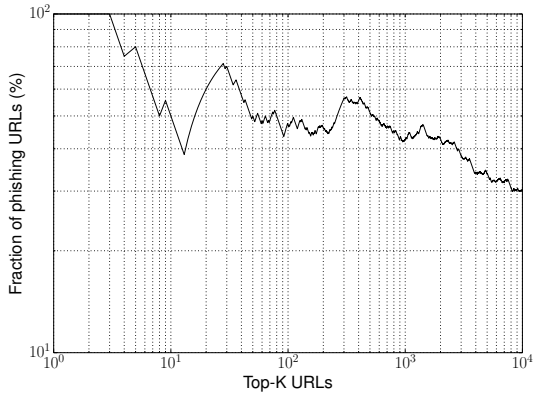

図 5: フィッシング URL のヒット率

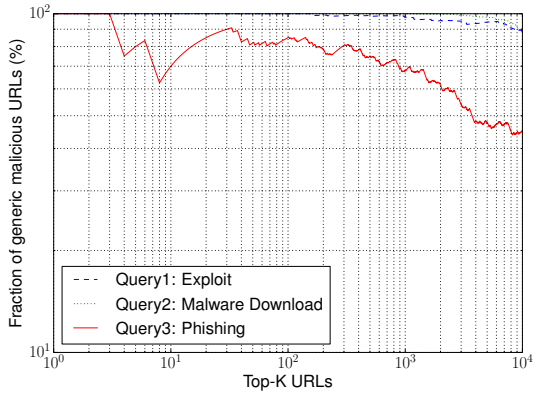

図 6: 悪性 URL のヒット率

シング URL ヒット率と同じように不安定となっている．フィッシング URL ヒット率と同様な原因を持つと考えられる．

6 まとめ

本研究は Bayesian Sets を利用することにより，任意の悪性 URL 群と類似した URL を検索する方法を提案した．様々な種別の悪性 URL を用いたケーススタディを通して提案方法の有効性を検証した．この結果，攻撃 URL とマルウェア配布 URL に関しては高いヒット率で類似 URL を検索できることを実証した．フィッシングに関しては必ずしも精度は良いとはいえないため精度向上が必要である．URL に関して取得可能な様々な特徴の追加，アルゴリズムの改良，解析データの大規模化は今後の課題である．

参考文献

[1] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian, “Internet inter-domain traffic,” *SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. –, Aug. 2010.

[2] Steve Ragan, “An in-depth look at one of the Web’s most famous crime kits.” `http://www.csoonline.com/article/2459925/malware-cybercrime/exposed-an-inside-look-at-the-magnitude-exploit-kit.html`.

[3] D. Canali, M. Cova, G. Vigna, and C. Kruegel, “Prophiler: A fast filter for the large-scale detection of malicious web pages,” in *Proc. WWW*, (New York, NY, USA), pp. 197–206, ACM, 2011.

[4] Z. Ghahramani and K. A. Heller, “Bayesian sets,” in *Proc. NIPS*, 2005.

[5] H. Choi, B. B. Zhu, and H. Lee, “Detecting malicious web links and identifying their attack types,” in *Proc. USENIX WebApps*, 2011.

[6] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond blacklists: learning to detect malicious web sites from suspicious urls,” in *Proc. KDD*, pp. 1245–1254, 2009.

[7] B. Eshete, A. Villafiorita, and K. Weldemariam, “Binspect: Holistic analysis and detection of malicious web pages,” in *Proc. SecureComm*, pp. 149–166, 2012.

[8] L. Xu, Z. Zhan, S. Xu, and K. Ye, “Cross-layer detection of malicious websites,” in *Proc. CODASPY*, pp. 141–152, 2013.

[9] D. Canali, M. Cova, G. Vigna, and C. Kruegel, “Prophiler: a fast filter for the large-scale detection of malicious web pages,” in *Proc. WWW*, pp. 197–206, 2011.

[10] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying suspicious urls: an application of large-scale online learning,” in *Proc. ICML*, p. 86, 2009.

[11] L. Invernizzi and P. M. Comparetti, “Evilseed: A guided approach to finding malicious web pages,” in *Proc. IEEE Symposium on Security and Privacy*, pp. 428–442, 2012.

[12] Google Sets. `http://en.wikipedia.org/wiki/List_of_Google_products#Discontinued_in_2011`.

[13] M. Akiyama, M. Iwamura, Y. Kawakoya, K. Aoki, and M. Itoh, “Design and implementation of high interaction client honeypot for drive-by-download attacks,” *IEICE Transactions*, vol. 93-B, no. 5, pp. 1131–1139, 2010.

[14] K. Aoki, T. Yagi, M. Iwamura, and M. Itoh, “Controlling malware http communications in dynamic analysis system using search engine,” in *Proc. IEEE CSS*, pp. 1–6, 2011.

[15] PHISHTANK, “Free community site for anti-phishing service.” `http://www.phishtank.com/`.

[16] ALEXA, “The web information company.” `http://www.alexa.com`.

[17] DMOZ, “Netscape open directory project.” `http://www.dmoz.org`.