Twitterで共有されるAndroidアプリケーション配布ページの実態調査および検知手法の検討

三村 隆夫[1,a)]　巻島 和雄[1]　岩本 一樹[1]

概要：スマートフォン向けOSであるAndroidでは，多数のアプリが提供されている．通常，それらのアプリは，公式アプリストアであるGoogle Playを経由して提供されるが，ウェブサイトなどインターネット上で直接配布することも可能である．しかし，直接配布においては，悪性な挙動を行うアプリの配布事例も報告されている．我々の先行研究では，匿名利用が可能なソーシャルネットワーク（SNS）であるTwitterでウェブサイトが共有されて配布されるアプリの情報収集・調査を行い，Twitterで共有されるAndroidアプリに一定のリスクがあることを示した．本稿では，ユーザへの注意喚起の実現に向けて，アプリを配布しているウェブページの実態調査を行い，調査結果から抽出した特徴量を用いて，悪性なAndroidアプリを配布するウェブページの自動検知手法を検討し評価した．

キーワード：SNS，Android，サードパーティマーケット，悪性サイト検知

A study of web pages distributing Android applications shared in Twitter and consideration of a detecting method

Takao Mimura[1,a)]　Kazuo Makishima[1]　Kazuki Iwamoto[1]

Abstract: There are a large number of applications that are developed for Android OS. Although they are generally distributed via Google Play, the official application store, those apps can be distributed directly on the Internet. There are, however, reports on the direct distribution of apps with malicious behavior. Our previous study gathered and examined information about Android apps that were shared in Twitter, one of the social network services accepting anonymities, and showed that the use of the apps shared there had a certain risk. In this paper, we investigate web pages distributing Android apps that have been obtained from Twitter and evaluate classifiers that use features extracted from the pages in order to detect ones that distribute malicious apps.

Keywords: SNS, Android, third party market, malicious website detection

1. はじめに

スマートフォン向けOSであるAndroidでは，多数のアプリが提供されている．通常，それらのアプリは，Googleが運営する公式ストアであるGoogle Playで配布が行われているが，インターネット上で直接配布することも可能である．公式アプリストアでは，運営のポリシー上，配布が禁止されたコンテンツが存在 [1] するほか，国や地域によってはその利用が制限されている [2]．それ以外の配布方法としては，サードパーティが運営するマーケット [3] や開発者による直接配布 [4] などがある．Google Play以外で配布されるアプリをインストールするためには，ユーザが端末設定を変更し，明示的な許可を与える必要がある．

Androidに対する主要な脅威の一つとして，アプリとして配布されるマルウェアが挙げられる．Androidアプリとは，APK（Android PacKage）形式のファイルであり，ウェブサーバにより手軽に配信することが可能であるが，

[1] 株式会社セキュアブレイン
SecureBrain Corporation
[a)] *takao_mimura@securebrain.co.jp*

悪性な動作を行うアプリの配布事例も報告されている [5]．そのため，悪性なアプリの端末へのダウンロードを防止することは，セキュリティ上，有効である．

我々の先行研究 [6] では，ソーシャルネットワークサービス（SNS）の一つである Twitter について，ツイートを介して情報共有される APK に関する分析を行い，その実態を報告した．配布時の状況および表層分析による調査結果として，VirusTotal によるファイル検査で検知される APK を多数配布するサイトの存在を確認した．さらに，サードパーティマーケット上で配布される APK に一定のリスクが存在し，取り扱いに注意が必要であることを示した．

Twitter で情報共有された APK をユーザがインストールする場合，その典型的なシナリオは以下のようになる．まず，ユーザはツイート内の APK に関する短い説明を確認し，含まれる URL をクリックする．次に，表示されたウェブページの内容を確認した後，APK の URL をクリックしダウンロードを行う．そして，ダウンロードされた APK を端末上で開き，インストールプロセスを開始する．

上記シナリオでは，ツイートやウェブページ，URL といった情報項目が出現する．これらがもし悪性な APK に特徴的であるならば，事前の警告が可能だと考えられる．

本稿では，アプリを配布するウェブページの調査を行い，悪性なアプリを配布するページを検知する手法を検討，評価した．以下，2 章では Twitter からのデータ収集について説明する．3 章では使用するデータセットについて述べる．4 章では検知手法および実験結果について述べる．5 章では 3 章，4 章の分析および考察を行う．6 章では関連研究について述べる．最後に，7 章では本稿のまとめを行う．

2. Twitter を利用した APK 収集

本稿では，Public Streaming API[7]（track statuses/filter，キーワード “apk”）を用いて Twitter に投稿されるテキスト（ツイート）および包含される URL を収集し，当該 URL とそのリンク先に含まれる URL からコンテンツを取得した．取得コンテンツには HTML 等が含まれるため，以下の条件に全て合致するファイルを APK として抽出した．

- ZIP フォーマットである
- `classes.dex` および `AndroidManifest.xml` が存在
- `META-INF/` が存在

上記手順によるデータ収集を 2017 年 6 月 22 日から 2019 年 5 月 26 日にかけて実施した．APK ファイルとして 86,761 件を収集し，VirusTotal File Report（FR）[8] が利用可能な 66,996 件を本稿での調査対象（以下，データセット APK）とした．

データセット APK に対する VirusTotal FR では，未検知が 34,972（52.2%）件，検知数 1 以上が 32,024（47.8%）件であった（**図 1**）．先行研究 [6] のデータセット APK と比較しても，未検知・検知数 1 以上の割合の傾向に大きな変化は見られていない．ただし，検知数が 10 以上と多数のセキュリティベンダに危険性が認識されている APK については，先行研究では 545 件，1.6%であったのに対し，本データセット APK では 3,544 件，5.3%と割合の増加が確認されている．VirusTotal FR での検知とは，各セキュリティベンダがユーザに対して警告すべき対象とみなしているものであり，以下，本稿では検知数が 1 以上の APK を要注意 APK と呼ぶ．

表 1 評価データセットの概要

項目	件数
APK 数（要注意数）	66,972（32,016）
APK 配布 URL	86,981
アクセスページ	100,597
ツイート	127,598

データセット APK の月別の取得件数については，収集期間が短い 2017 年 6 月と 2019 年 5 月を除き，最小で 1,150 件（2018 年 9 月）から最大で 4,645 件（2017 年 10 月）と増減は見られるが，継続的に APK 情報が共有されている実態が確認されている（**図 2**）．要注意 APK の割合としては，2019 年 4 月に割合が 20%と低い数値も見られるが，全体的には 40 から 60%程度で推移する傾向を示している．

3. データセット

3.1 概要

本稿の観測範囲では，APK にリンクされた URL が直接ツイートされるケースは非常に少なく，37 件の APK に関する 43 件の URL のみが確認されている．それ以外の APK 配布 URL については，リンク先のコンテンツに含まれていた．以下，APK にリンクされた URL を APK 配布 URL，APK 配布 URL を含むウェブページをアクセスページと呼ぶ（**図 3**）．

なお，HTTP アクセスでは，リダイレクトにより別の URL への遷移が発生することがある．悪性な APK のダウンロードを事前に警告するという文脈では，警告のタイミングはハイパーリンクをクリックする前である．そのため，リダイレクト発生前の APK 配布 URL を使用した．

以降の検討では，APK 配布 URL が直接ツイートに含まれており，アクセスページが確認できない APK を除外する．評価データセットを**表 1** に示す．ある APK が複数の URL で配布されることがあるため，APK 配布 URL 数は，APK 数より多くなっている．また，これにより，前述の 37 件には除外されない APK があることに注意されたい．

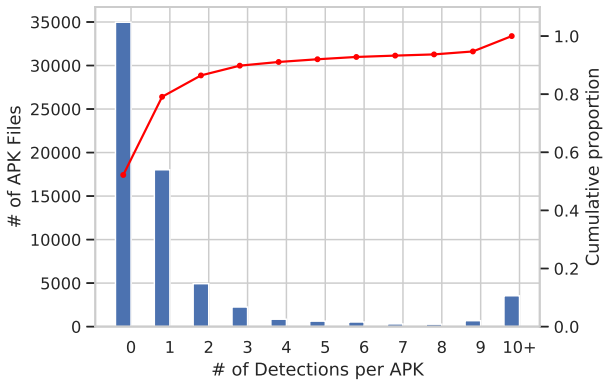

図 1　VirusTotal File Report による APK 検知数

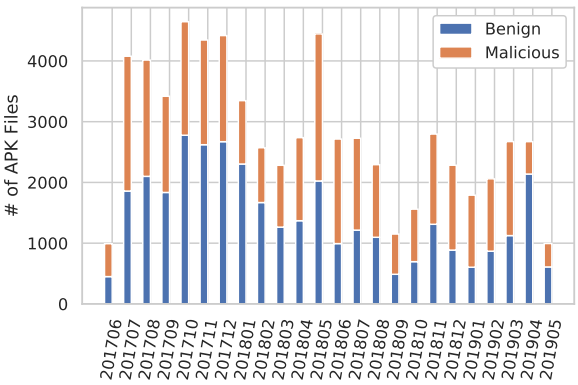

図 2　月別の APK 取得件数の傾向

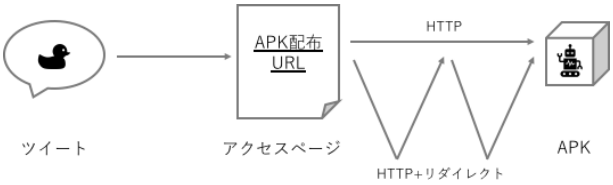

図 3　アクセスページと APK 配布 URL

3.2　APK 配布 URL

3.2.1　トップレベルドメイン・IP アドレス

配布される APK 数と要注意 APK について，APK 配布 URL のトップレベルドメイン（以下，TLD），および IP アドレスから推定した国名コードを基準とした関係を**図 4**，**図 5** に示す．ただし，もともと IP アドレスで指定された 219 件の URL については，同じ TLD グループとしている．また，国名コードについては，2019 年 8 月にドメインの名前解決を行い，取得に成功した 2,361 件の IP アドレスに対応する情報を MaxMind GeoIP[9] を用いて取得した．

配布 APK 数と要注意 APK の割合にはほとんど相関は見られないが，TLD および国名コードを基準とすると，要注意 APK の割合が大きく異なっている．なお，配布 APK 数が少数（10 以下）の場合，要注意 APK 割合が 0.0 や 1.0 といった数値となっているが，これはサンプル数の少なさに起因すると考えられる．

要注意 APK 割合が高い TLD としては，.io，.co，.org，.info，.in が，低い TLD としては，.jp，.moe，.ai，.br，.de が確認されている．

国名コード別については，配布 APK 数が 100 を超えるケースでは，要注意 APK の割合が 0.2 から 0.5 に分散しているが，NL（オランダ）では 0.92 と高い数値を示している．これは，NL には，要注意 APK 割合が非常に高いアプリ配布サイトのドメインが複数含まれているためである．このドメインは，NL 全体の約 85%を占めている．

直接 IP アドレスで指定されていた APK 配布 URL については，要注意 APK 割合は 38.8%であった．

3.2.2　FQDN

APK 配布 URL の FQDN を基準として，配布 APK 数と要注意 APK の割合の関係を**図 6** に示す．これらの URL に出現する FQDN 数としては，IP アドレスが直接指定されているものを含めて，2,571 件であった．FQDN 別に見ると配布 APK 数と要注意 APK の割合は大きくばらついているが，これらの値にほとんど相関は見られない．

最も多く APK を配布する FQDN では，8,418 件の配布が確認されているが，要注意 APK の割合は 7.1%と低い数値となっている．この FQDN では，公式ストアのアプリを再配布するとされており，割合が低い一因だと考えられる．一方，数千件の APK を配布し，なおかつ，要注意 APK の割合が 50%を超える FQDN が複数確認されている．これらは，インターネット上でアプリを配布するサードパーティのマーケットであると考えられている．

3.3　アクセスページ

アクセスページのコンテンツは，HTML 形式である．このコンテンツからの情報取得として，ウェブページのメタ情報を定義するために利用される，HTML の `meta` タグに着目した．メタ情報の一例としては，検索エンジン向けに設定するウェブページ概要を表す文字列がある．

ここでは処理対象として，`meta` タグの属性に `title`，`description`，`keyword` のいずれかを含む名称が指定されている場合に，`content` 属性に指定された値を抽出，連結することで文字列を生成した．

生成した文字列について，language-detection[10] を用いて言語種別を推定した．language-detection は，ISO 639-1 で規定される言語コードを推定するライブラリである．言語推定の結果を**表 2** に示す．最も多い推定言語コードは，en であり全体の 77.8%を占めている．次に多い種別は，文字列が設定されていない場合を含め，language-detection が言語判定に失敗したケースである．それ以降は，id，es，tr が続き，上位 10 種類で全体の 99.2%を占めている．な

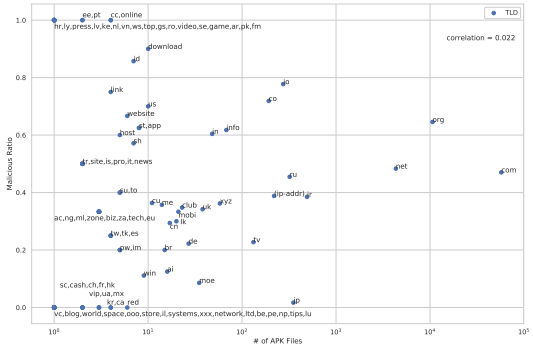

図 4　配布 APK 数と要注意 APK 割合の相関 (トップレベルドメイン)

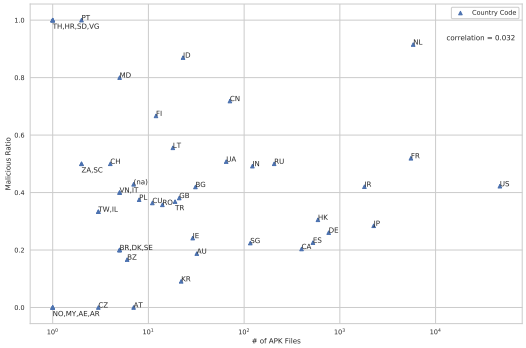

図 5　配布 APK 数と要注意 APK 割合の相関 (国名コード)

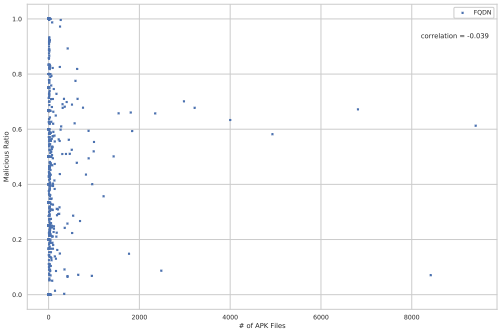

図 6　配布 APK 数と要注意 APK 割合の相関 (FQDN)

表 2　HTML meta タグ文字列の推定言語種別

順位	言語コード	言語名	件数	割合	累積
1	en	English	78,304	77.8%	77.8%
2	(判別不可)	-	11,631	11.6%	89.4%
3	id	Indonesian	5,549	5.5%	94.9%
4	es	Spanish	991	1.0%	95.9%
5	tr	Turkish	794	0.8%	96.7%
6	de	German	738	0.7%	97.4%
7	pt	Portuguese	689	0.7%	98.1%
8	fr	French	530	0.5%	98.6%
9	ar	Arabic	377	0.4%	99.0%
10	fa	Persian	174	0.2%	99.2%

お，ja については，19 番目に位置している（件数 36）．

3.4　ツイート

ツイートには，言語に応じて最大で 140 文字または 280 文字のテキストデータが含まれる [11]．

language-detection による言語推定の結果を**表 3** に示す．最も多い推定言語コードは，en であり全体の 70%を占めている．それ以降は，sw，id，判別不可，de が続き，上位 10 種類で全体の 90.6%を占めている．

Twitter のユーザとしては，日本に属するユーザの割合が 2 番目に高いことが報告されているが [12]，ja については，8 番目に位置している．これは，本稿でのデータ収集の対象としてキーワードに “apk” とだけ指定していることから，日本語のみのツイートは収集対象外となっていることが要因の一つだと考えられる．

表 3　ツイートの推定言語種別

順位	言語コード	言語名	件数	割合	累積
1	en	English	88,667	69.5%	69.5%
2	sw	Swahili	6,995	5.5%	75.0%
3	id	Indonesian	4,767	3.7%	78.7%
4	(判別不可)	-	3,491	2.7%	81.4%
5	de	German	3,415	2.7%	84.1%
6	pt	Portuguese	2,286	1.8%	85.9%
7	es	Spanish	2,086	1.6%	87.5%
8	ja	Japanese	1,473	1.2%	88.7%
9	sv	Swedish	1,230	1.0%	89.7%
10	ar	Arabic	1,193	0.9%	90.6%

4.　検知手法

本章では，検知手法として，3 章で示したデータからの特徴抽出方法，分類器への特徴量の適用方法，および，評価実験について述べる．分類器として Support Vector Machine（SVM）アルゴリズムを採用しており，その実装として LIBSVM[13] を利用した．カーネルには，RBF カーネルを指定した．

4.1　教師ラベル

本稿のデータセット APK では，VirusTotal FR での検知数に基づいて要注意 APK を定義している．要注意 APK を陽性クラス（Positive Class），それ以外の APK を陰性クラス（Negative Class）としてラベル付けを行った．

4.2　特徴量

特徴量の取得元（以下，データソース）を以下に示す．

- APK 配布 URL
- アクセスページ（3.3 節の手順で文字列を抽出）

表 4　特徴量の組み合わせパターン

方式	APK 配布 URL	アクセスページ	ツイート	特徴量の次元数
#1	✓			89,273
#2	✓	✓		116,350
#3	✓		✓	104,283
#4	✓	✓	✓	125,580

- ツイート

これらのデータソースから取得した情報に対して，以下の処理を適用することで特徴量化を行った．

テキストの特徴量化

(1) 指定されたテキストについて，非英数字の文字を基準としてトークンに分割

(2) 分割により生成された各トークンに対して，ステミングを適用．ステミングの実装としては，goporterstemmer[15] を利用

一例として，入力値に `https://www.example.com/apks/Awesome-New-Game-Mod-v1.2.3.apk` を与えた場合，出力される特徴量は以下のようになる．

```
http␣www␣exampl␣com␣apk
awesom␣new␣game␣mod␣v1␣2␣3␣apk
```

4.3 分類器

生成した特徴量を LIBSVM への入力とするため，SVM-light[14] のデータフォーマットに変換した．1 行が一つの訓練またはテストデータに対応し，先頭に教師ラベル（Positive Class +1，Negative Class -1）に空白を続けて特徴量を記述する．特徴量には予め一意な識別子となる整数を割り当てておき，その識別子と対応する値をコロン（:）で連結した形式で表現する．特徴量に対応する値として，存在を示すため 1 を設定する．ただし，一度しか出現しない特徴量については，除外する．

特徴量を**表 4** の通りに組み合わせ，分類器を構築した．以下，この特徴量の組み合わせをそれぞれ方式 N と呼ぶ．

4.4 評価基準

以下の尺度を用いて，分類器の評価を行う．

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

表 5　グリッドサーチにより決定したハイパーパラメータ

方式	Cost	Gamma
#1	$\log_2 3.0$	$\log_2 -5.0$
#2	$\log_2 9.0$	$\log_2 -11.0$
#3	$\log_2 1.0$	$\log_2 -3.0$
#4	$\log_2 7.0$	$\log_2 -11.0$

TP および TN については，陽性クラス，陰性クラスが正しく分類された件数であり，FP および FN は間違って分類された件数を示す．(1) は，分類結果において，正しく分類された割合を示す．(2) は，陽性と分類されたデータにおいて，正しく予測された割合を示す．(3) は，陽性クラスのデータについて，正しく分類された割合を示す．(4) は，Precision と Recall の調和平均である．

4.5 評価実験

4.5.1 ハイパーパラメータ

SVM で用いるハイパーパラメータを決定するため，LIBSVM に同梱のグリッドサーチを行うツール（`grid.py`）を用いてパラメータ探索を行った．10,000 件分の部分データを対象に実行しており，決定したパラメータを**表 5** に示す．

4.5.2 交差検証

分類器の評価を行うため，交差検証（k=5）により行った実験結果を**表 6** に示す．本実験では，F-measure による評価において，APK 配布 URL のみを用いる組み合わせ方式 #1 が最も良い結果となった．ツイートを用いる方式#3，アクセスページのコンテンツを用いる方式#2 が続き，全ての特徴量を用いる方式#4 が最も低い数値となった．

4.5.3 ホールドアウト法

交差検証と併せて，ホールドアウト法での評価実験を実行した．全体の半数である 43,490 件を訓練データとして利用し，残りの 43,491 件をテストデータとした．

実験結果（表 6）としては，交差検証と同様に，結果の良い順に方式#1，#3，#2，#4 となった．

5. 考察

評価実験

実験では，3 種類の特徴量を組み合わせた分類器に対する F-measure による評価により，交差検証では 82.1%から 87.1%，ホールドアウト法では 80.1%から 85.0%という数値が得られ，要注意 APK の配布 URL の検知に一定の効果があることが確認された．しかし，交差検証，ホールドアウト法ともに，利用する特徴量の種類を増やすことで精度が低下する結果となった．以下では，ホールドアウト法での実験において分類器による予測が正しく行われず，FP または FN となったテストデータ（以下，予測失敗テストデータ）の分析結果を示す．

各方式で重複する予測失敗テストデータの分布を**図 7** に示す．予測失敗テストデータの総数は 11,491 件である．こ

表 6　分類器の評価結果 (CV：交差検証，HO：ホールドアウト法)

	Accuracy		Precision		Recall		F-measure	
方式	CV	HO	CV	HO	CV	HO	CV	HO
#1	87.1%	86.5%	87.3%	86.3%	86.9%	83.7%	87.1%	85.0%
#2	82.4%	82.2%	82.1%	80.6%	83.0%	80.3%	82.5%	80.4%
#3	84.8%	84.4%	84.8%	83.5%	84.9%	82.0%	84.9%	82.8%
#4	81.9%	82.0%	81.5%	80.9%	82.6%	79.3%	82.1%	80.1%

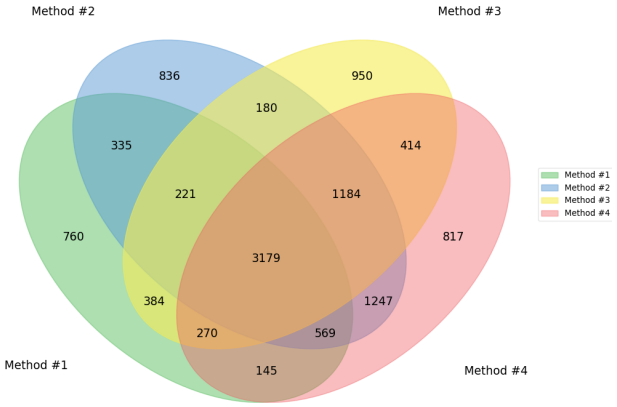

図 7　各方式で重複する予測失敗テストデータの分布

表 7　予測失敗テストデータを他の方式で正しく予測できた割合

方式	#1	#2	#3	#4
#1	-	26.6%	30.9%	29.0%
#2	44.5%	-	38.5%	20.3%
#3	40.2%	29.8%	-	25.6%
#4	46.8%	21.0%	35.5%	-

のうち，3,179 件のテストデータについては，全ての方式で正しく予測できなかった．各方式だけで発生する予測失敗としては，それぞれ 760 から 950 件程度が確認された．

ある方式での予測失敗テストデータについて，他の方式では正しく予測された割合を**表 7** に示す．各方式においてばらつきは見られるが，方式#1 については，予測失敗テストデータの平均 29%を他の方式では正しく予測できたことを示す．これは，分類器モデルに適用される特徴量の違いにより，正しく識別できるテストデータが異なることを示している．

一般に，パスやクエリパラメータなどの URL 構成要素は，ウェブサイトの設計や使用される CMS（Content Management System）によって大きく異なる．APK 配布件数の多い FQDN に対する予測失敗テストデータの割合を**表 8** に示す．項目 a) や b) では，方式#1 に比べて方式#2 や#3 の数値が低下し精度の向上が見られた一方，項目 h) では数値が大幅に上昇し精度が低下している．これは，前述の特徴量の違いにより正しく識別できる対象が変化することを示している．

APK 配布 URL の構造の違いによる影響を精査するため，パスとクエリパラメータの命名規則が大きく異なる 2 つの FQDN を対象とした方式#1 による予測結果の比較を**表 9** に示す．ここでは，URL における APK 情報の出現有無が大きな相違点である．APK 情報が出現する FQDN では，配信 URL のパスにアプリ名と思われる文字列が含まれている（例：`/7/Skype\%20Lite-1.15.0.28121-release.apk`）．APK 情報が出現しない FQDN では，配信 URL のパスは常に固定値であり，クエリパラメータのみが変化する仕組みとなっている（例：`/wp-content/themes/APKM[***]/download.php?id=337833`）．前者の FQDN で配布されるテストデータに対する予測結果では，(5) および (6) により算出される FPR，FNR がそれぞれ 14.7%，5.1%となっており，Positive，Negative へのバランスの良い対応が示されている．一方，後者の FQDN では，予測結果は Positive の 1 件を除き，全て Negative であった．これにより，FPR は 0.0%，FNR は 99.8%となり，極端に低い F-measure 値が算出された．このサイトで配信される APK の検知には適していないことが示されている．

$$FPR(FalsePositiveRate) = \frac{FP}{FP + TN} \tag{5}$$

$$FNR(FalseNegativeRate) = \frac{FN}{FN + TP} \tag{6}$$

以上より，本稿の検知手法では，要注意 APK の配信 URL の検知に一定の効果があること，APK 配信 URL の構造に大きな影響を受けること，ならびに，特徴量にアクセスページとツイートを組み合わせることで検知可能な対象を広げる効果が期待できることが確認された．ただし，全体的な検知精度については，改善が必要な水準である．

特徴量

本稿の特徴量は，文字列を分割することで生成したトークン単位で生成される．トークンとなる単語の一例としてアプリ名があるが，APK 配信 URL，アクセスページ，ツイートのいずれでも出現することがある．データソースに固有または重複して出現する特徴量の分布を**図 8** に示す．APK 配信 URL では固有の特徴量が多数得られているのに対して，アクセスページ，ツイートは，特徴量数が相対的に小規模であり，重複する割合はそれぞれ 48.4%，31%と高い数値であった．

特徴量の分布について，教師ラベルを基準とした傾向を**表 10** に示す．APK 配信 URL では，それ以外のデータソースと比較すると，異なる教師ラベル間での重複が相対的に少ないことが確認されたそのため，他の特徴量より有

表 8　FQDN 別の予測失敗テストデータの割合

	FQDN 名	件数	方式			
			#1	#2	#3	#4
a)	www.apkm[***].com	6,441	6.8%	6.5%	5.8%	6.8%
b)	webservices.apt[***].com	2,943	0.4%	0.2%	0.2%	0.2%
c)	www.medi[***].com	2,184	13.1%	17.3%	12.9%	15.4%
d)	download.devi[***].com	1,744	7.1%	7.1%	6.1%	8.4%
e)	dl3.ap[***].org	1,444	8.9%	11.8%	8.2%	11.2%
f)	sirius.andro[***].com	1,358	10.2%	12.2%	11.0%	12.1%
g)	dl3.uap[***].com	1,237	13.3%	15.3%	11.6%	15.4%
h)	s1.re[***].com	1,146	12.4%	34.7%	33.5%	35.7%
i)	dl2.ap[***].org	1,139	8.3%	9.2%	6.7%	10.5%
j)	dl2.uap[***].com	1,117	8.6%	11.5%	7.6%	12.0%

表 9　APK 配信 URL の構造による方式#1 の予測結果の比較

FQDN 名	APK 件数（要注意割合）	予測		結果		
		Positive	Negative	F-measure	FPR	FNR
dl2.uap[***].com	1,117（63.5%）	733	384	93.3%	14.7%	5.1%
www.apkm[***].com	6,441（6.8%）	1	6,440	0.5%	0.0%	99.8%

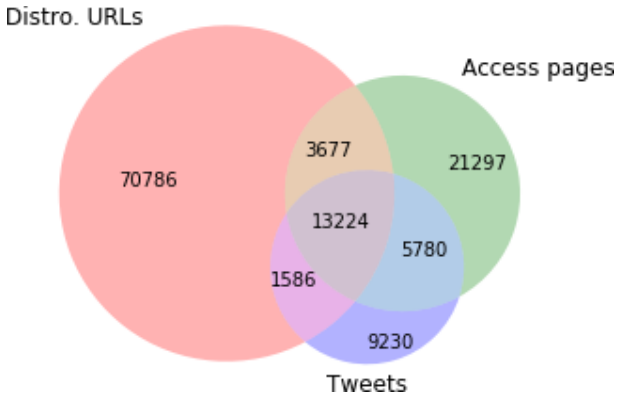

図 8　各データソースが生成する特徴量の重複の分布

表 10　教師ラベルを基準とした特徴量の重複の割合

特徴量	件数	共通	Negative	Positive
APK 配信 URL	89,273	9.5%	46.1%	44.4%
アクセスページ	43,978	35.7%	27.8%	36.5%
ツイート	29,820	26.5%	32.9%	40.7%

用性は高いと考えられる．

特徴量のトークン化では，非英数文字を基準とする文字列分割を行っている．しかし，この処理手順では，空白で単語を区切る英語，スラッシュや慣例的にハイフン等で単語を区切る URL やパスには対応可能であるが，単語を続けて記述する日本語などには対応ができない．今回の処理対象としたアクセスページおよびツイートについては，英語と推定される文字列が大部分を占めていたが，例えば日本語を多数含むデータを扱う場合，形態素解析による分割等を考慮する必要がある．

アクセスページ（運送会社詐称アプリ配布事例）

本稿のデータセットでは，運送事業会社を詐称し悪性な APK を配布すると考えられるアクセスページが確認されている．HTML の構造としては，画像コンテンツ等は正規のウェブページから読み込ませ，悪性な APK のみをそのページのドメインからダウンロードさせる仕組みとなっていた．`meta` タグには情報が設定されていなかった．これらのウェブページは，配布する APK をダウンロード，インストールさせるため，フィッシング目的で利用されていると推定されている．そのため，検索エンジン経由でのアクセスを考慮する必要性がないため，`meta` タグに情報が設定されていないと考えられる．

また，上記の悪性な APK のアクセスページとして，ウェブページのセキュリティ検査を行う正規のサイトも含まれていた．これは，悪性なアプリの存在を注意喚起するため，APK 配信 URL の検査結果を Twitter で報告したことによりデータセットに含まれたと考えられる．

ホールドアウト法による実験では，上述の悪性な APK と考えられるデータが，訓練データに 8 件，テストデータに 9 件が含まれていた．方式#2 および#4 で 1 件ずつ FN が発生したが，それ以外は正しく予測が行われた．今回は，FQDN 名やパスに特徴的な文字列が利用されたため，正しい予測になったと考えられるが，一般的な文字列が使用された場合，別の特徴量が必要になる可能性がある．

6. 関連研究

悪性なコンテンツにリンクされた URL を検知する研究は多数行われており，文献 [16], [17], [18] などがある．嶌田ら [16] は，セキュリティツールの長期ログを分析することで RIG ExploitKit の新しい IOC（Indicator of Compromises）を導出する手法を提案している．芹田ら [17] は，悪性ウェブサイトの URL のパスに共通して出現する文字列

に着目し，その類似性に基づいてクラスタリングを行い，悪性サイトを検知するための正規表現を生成する手法を提案している．孫ら [18] は，Bayesian Sets を用いることで既知の悪性 URL 群に類似する URL を検索する手法を提案している．

Twitter から取得した URL に対する研究としては，文献 [19], [20] などがある．野村ら [19] は，過去に報告された Drive by Download 攻撃のトラフィックから抽出された特徴量が論文執筆時点でも有効かを調査しており，評価での良性データとして Twitter から収集した URL を利用している．Halwar ら [20] は，相関関係に基づいて疑わしい URL を検知する WARNINGBIRD を中心として，Twitter で共有される URL を検査する手法をまとめている．

機械学習で用いる特徴量の拡充としては，Takahashi ら [21] が，マーケットでのカテゴリ情報および Android アプリの説明文を取得し，SVM の特徴量として適用する手法を提案している．

7. まとめと今後の課題

本稿では，Twitter で情報共有される APK について，配布 URL やアクセスページの分析を行い，要注意 APK の割合が高いトップレベルドメインや FQDN が存在することを明らかにした．また，APK 配布 URL やアクセスページ，ツイートから抽出した特徴量を用いて SVM による分類器を構築し，評価実験により要注意 APK の配布 URL の検知に一定の効果があることを示した．

今後の課題としては，検知精度の向上が挙げられる．まず，本稿の分類器では使用していない情報について，特徴量としての利用可否を検討する．例示すると，国名コードを始めとする Geo IP 情報や，アクセスページの HTML コンテンツに含まれる `meta` タグ以外の情報などである．次に，データソースの違いを特徴量に反映させる仕組みを検討する．本方式では，同一トークンは同じ特徴量として扱われるが，データソースを区別することによる分類器への影響を確認する．そして，これらの特徴量の指定や組み合わせを評価することで，検知手法の改善を図る．

謝辞 本研究は，国立研究開発法人情報通信研究機構の委託研究「Web 媒介型攻撃対策技術の実用化に向けた研究開発」の成果の一部です．ご協力いただいた皆様に，深く感謝します．

参考文献

[1] “Google Play, Developer Policy Center”, https://play.google.com/intl/en/about/developer-content-policy/
[2] “Supported locations for distribution to Google Play users”, https://support.google.com/googleplay/android-developer/table/3541286
[3] “Amazon Appstore for Android”, https://www.amazon.com/androidapp
[4] “Fortnite”, http://fortnite.com/android
[5] “安心相談窓口だより”, https://www.ipa.go.jp/security/anshin/mgdayori20181129.html
[6] 三村 隆夫，巻島 和雄，岩本 一樹．ソーシャルネットワークで共有される Android アプリケーションの実態調査．コンピュータセキュリティシンポジウム 2018 論文集,2018(2),113-120
[7] “Filter realtime Tweets”, https://developer.twitter.com/en/docs/tweets/filter-realtime/api-reference/post-statuses-filter
[8] “VirusTotal Public API v2.0”, https://www.virustotal.com/en/documentation/public-api/
[9] “MaxMind, GeoIP2”, https://dev.maxmind.com/geoip/
[10] “langdetect”, https://github.com/Mimino666/langdetect
[11] “Tweeting Made Easier”, https://blog.twitter.com/official/en_us/topics/product/2017/tweetingmadeeasier.html
[12] “Leading countries based on number of Twitter users as of July 2019 (in millions)”, https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/
[13] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 3, Article 27 (May 2011), 27 pages.
[14] “SVMlight”, http://svmlight.joachims.org
[15] “go-porterstemmer”, https://github.com/reiver/go-porterstemmer
[16] 嶌田 一郎，太田 敏史，山田 明，ユーザ環境観測による RIG Exploit Kit の長期観測と時間変化に対して頑強な攻撃検知．コンピュータセキュリティシンポジウム 2018 論文集,2018(2),1154-1161
[17] 芹田 進，藤井 康広，角田 朋，吉竹 利織，大鳥 朋哉，木城 武康，寺田 真敏，URL 正規表現自動生成による悪性通信検知手法に関する一考察．コンピュータセキュリティシンポジウム 2015 論文集,2015(3),242-249 (2015-10-14)
[18] 孫 博，秋山 満昭，八木 毅，森 達哉．既知の悪性 URL 群と類似した特徴を持つ URL の検索．コンピュータセキュリティシンポジウム 2014 論文集,2014(2),1-8 (2014-10-15)
[19] 野村 敬太，猪俣 敦夫，Rig Exploit Kit を利用した Drive by Download 攻撃における 分類評価．コンピュータセキュリティシンポジウム 2018 論文集,2018(2),327-332
[20] D.Halwar, Jyoti, Kadam, Sandeep, Desale, Vrushali. (2015). Detection of Suspicious URL in Social Networking Site Twitter: Survey Paper. International Journal of Computer Applications. 110. 6-8. 10.5120/19334-0688.
[21] T.Takahashi, T.Ban, T.Mimura, K.Nakao, ”Fine-Grained Risk Level Quantification Schemes based on APK Metadata,” The 22nd International Conference on Neural Information Processing, 663 - 673, Springer, November, 2015.