脅威を引き起こすアプリケーションを検出する手法についての考察
―紹介ページからの特徴量抽出について―

2016SE014 林勢也 2016SE037 川村隼大

指導教員：横森励士

1　はじめに

近年 Android を筆頭にスマートフォンが急速に普及し，スマートフォン上で動作させるアプリケーション（以下，アプリ）の需要が増加している．大量のアプリの中には悪意を持つものが存在しており，様々な脅威が日々引き起こされている．このような環境下では，アプリの紹介ページなど利用者に事前に公開される情報を利用して，脅威を未然に回避することが求められる．[4] では，アプリが利用する権限や紹介ページなどから入手可能な情報を用いて，より高い精度の機械学習で悪意をもつアプリの検出ができそうであることを示した．しかし，情報をアプリの紹介ページから直接得ていたので，集める特徴量を増やそうとした時に，すでに消去されているアプリの情報が得られないなど，継続した分析を行うのに不十分であった．

本研究では，特徴量の抽出に必要な情報を手元に残したうえで，手元のデータから特徴量を抽出する仕組みを作ることを目的とする．特徴量の抽出源となる情報を決定し，抽出源の定期的な取得を試みる．得られた情報源から情報を抽出し，機械学習に用いる表を作成する．その作業と並行し，その情報源から悪意を持つアプリかを判定する．特徴量を得るためのシステムを試作し，実際に公開されているアプリに対して入手を試みた結果を紹介する．さらに，入手したデータを元に，教師ありの機械学習手法を適用した結果を紹介する．現状の入手手法の課題と，得られた情報を用いて機械学習を行う際に直面した課題を考察し，悪意を持つアプリの検出を行う仕組みの実現につなげる．

2　背景技術

2.1　悪意を持つアプリが引き起こす脅威

代表的な Android アプリの配布サービスとして，Google Play[1] が運営されている．アプリを提供する側がアプリとともにタイトル名やアプリの説明文，スクリーンショットなどの画像，連絡先などを登録すると，マルウェアやウイルスなどの感染を機械的にチェックした上で，公開される．チェックを通り抜ければ，悪意を持つアプリもそのまま公開されてしまうので，利用者はアプリをインストールする際に [1] から与えられた情報をもとに自己判断を行う必要がある．Android 不正アプリ検出数の割合 [2] を表 1 に示す．表で示す通り，“アドウェア”が約 8 割を占めており，“情報窃盗／バックドア”が残りの部分の半数を占めている．ユーザ側は脅威への対策としてセキュリティアプリの導入が推奨されているが，そのどれもが一度アプリをインストールしてからチェックにかける方式をとっているので，インストールされた時点で何らかの被害を及ぼすアプリには効果が薄いと言える．

表 1　国内での不正アプリ検出種別割合 (2015) [2]

脅威の種類	割合
アドウェア	79.80 %
情報窃盗/バックドア	8.56 %
ネット詐欺	2.84 %
脆弱性悪用	1.46 %
プレミアム SMS 悪用	0.81 %
ランサムウェア	0.04 %
その他の不正アプリ	6.48 %

2.2　関連研究

Zhongmin らは，アプリのアクセス権限に対して，機械学習による分類分けを行うことで，悪意を持ったアプリの判別を行った [3]．[3] では，無料 Android アプリを対象として，APK ファイルで記述されている要求権限を抽出した．アプリのカテゴリとアクセス権限は，密接な関係があるとし，同種のカテゴリと異なる特徴を持つものは悪意を持ったものである可能性が高いと判断している．アプリが属するべきカテゴリを推測する形で機械学習を行った．安藤ら [4] は，アプリの紹介ページから情報を得て特徴量とし，アクセス権限やその他の機械学習の材料とすることで，悪意を持つアプリの検出の精度が向上するかを評価した．評価実験の結果，カテゴリごとにアプリを分けてから悪意をもつアプリを検出する手法を用いて，[3] のアプローチより精度の高い検出結果を得た．

3　特徴量抽出システムの実現

3.1　研究の動機

[4] では，アクセス権限や紹介ページからの情報を用いることで，より高い精度で悪意をもつアプリの検出ができた．情報収集の観点からは，権限を含めたそれらの情報を直接紹介ページを確認することで得ており，後から調査項目を増やそうとしたときに，悪意をもったアプリの検出において重要な既に消去されたアプリについて追加の情報が得られない．抽出に必要な情報は手元に必ず残し，手元のデータから特徴量を抽出する仕組みの実現が必要である．

3.2　研究の概要

この目的にしたがって，抽出された情報から悪意のあるアプリを検出するためのシステムの構築を行った．システムでは，事前に必要となりそうな情報の範囲を考察し，それらの情報をすべて保存する．例えば，紹介ページの情報として， Google Play 上の紹介ページや，公式サイトな

どをダウンロードして手元に残す．各調査項目は，手元に保管した情報から抽出を行う．利用する権限や紹介ページの情報についての調査項目ごとに情報を抽出し，表にまとめ，その表を用いて機械学習を行う．以下では，システムの全体像と自分たちが担当した範囲について紹介し，得られた情報を機械学習手法に適用することで，どのような結果が得られたか紹介する．

4　実現した特徴量抽出システム

4.1　システムの概要について

図 1 はアプリ情報取得システムの概要で，悪意のある Android アプリの検出を目的とする．各アプリから 2 種類の表を作成し，その表を用いてアプリが悪意を持つかどうかを判定する．1 つ目の表は，各アプリが利用する権限をまとめた表で，2 つ目の表は，各アプリの紹介ページなどから抽出した情報をまとめた表である．本研究では，図 1 の下部に相当する，各アプリの紹介ページなどから抽出した情報をまとめる部分を実現する．その過程で得た情報から Google Play 上で削除されたアプリなどを記録し，それらを悪意をもつアプリとみなし，学習材料に用いる．

4.2　抽出する特徴量について

特徴量を表現する表を作成するために，悪意を持つアプリが持つと考えられる特徴を想定し，ダウンロードする成果物を設定する．対象ごとに抽出する項目を作成し，特徴量とする．表 2 は, ダウンロードする対象と，抽出する内容を表した表である．例えば， Google Play の紹介ページからは，悪意を持つアプリはアプリごとに開発元を変えているので，悪意を持つアプリの場合は開発元が提供するアプリの数が極端に少ないと仮説を立て，開発者が他にアプリを提供している数を抽出する．このようにして得た 19 種類のデータに対して，ワンホットエンコーディング，ビニングなどの手法を用いて，機械学習へ入力するための 39 種類の特徴量を求めた．図 2 は，アプリの紹介ページなどから得られる情報の抽出部についての概要である．想定する入力は，アプリの集合についての情報である，$AP = \{AP_1, AP_2, AP_3, \ldots, AP_k\}$ である．出力として想定する表は 2 つ存在し，1 つ目は，そのアプリ集合のそれぞれのアプリから抽出する特徴量を $T = \{T_1, T_2, T_3, \ldots, T_k\}$ としたときの，$AP \times T$ を表現する表である．2 つ目の表は，アプリが悪意を持つかどうかを判定した結果を求め，そのアプリ集合を悪質と判定をした場合を $t = 1$ とし，悪質でないと判定をした場合を $t = 0$ として，AP_1 から AP_k までを判定した結果である．

4.3　悪意を持つアプリの決定方法について

悪意を持つアプリかを判定するにあたり，どのようなアプリが悪意をもつとするかを以下のように定義した．

アプリ紹介文での矛盾

紹介文の記述と矛盾するアクセス権限を要求しているものを対象とした．

セキュリティアプリでの検出

セキュリティアプリに搭載されているアプリスキャン機能によってプライバシー保護の観点から危険性があると判断されたものを対象とした．

アプリ配信の停止

Google Play ， App Store 上において該当アプリがすでに削除されているものを対象とした．

4.4　抽出の手順

準備

同じジャンルのアプリの情報を収集し，対象アプリの名前と Google Play での紹介ページの URL をまとめた集合として， AP を作成する．

手順 1

あらかじめ決めた取得する項目にしたがって，アプリ $(AP_1, AP_2, AP_3, \ldots, AP_k)$ ごとにダウンロード対象となる項目の URL を特定する．

手順 2

対応する URL を指定してダウンロードを行うバッチファイルを生成する．

手順 3

定期的にバッチファイルを起動し，ダウンロード対象を定期的に入手する．アプリが Google Play ，App Store 上から削除されていないかを合わせて調査する．

手順 4

入手したデータを分析し，抽出する情報を特徴量として抽出する．アプリ紹介文で矛盾の有無も調べる．

手順 5

アプリごとの特徴量をまとめ表にする．また，対象アプリが悪意を持つかどうかも判定し，表にする．

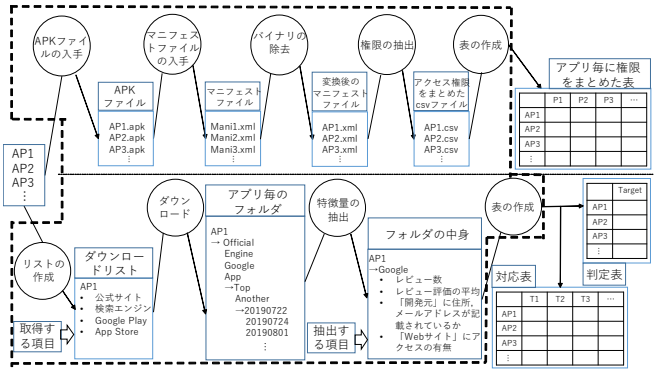

図 1　アプリ情報取得システム

5　データセットの作成と機械学習の適用結果

5.1　機械学習を行うためのデータセットの作成

表 3 で示すような 6 ジャンル計 758 個のアプリからなるデータセットを作成した．具体的な情報の入手方法は，2019 年 7 月〜8 月の間にバッチファイルを作成しながら，

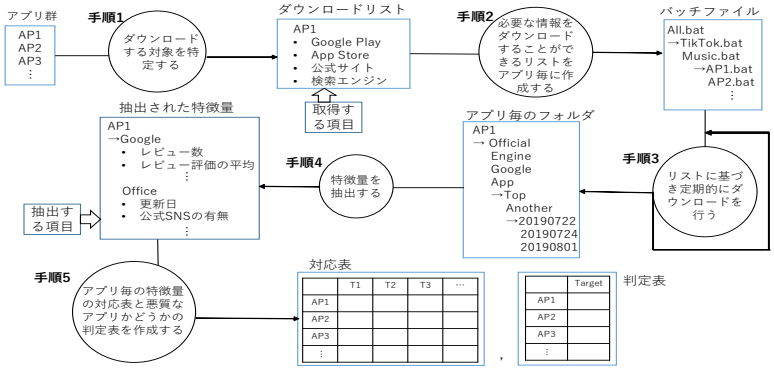

図 2 特徴量抽出システム

表 2 アプリ情報から入手した特徴量の一覧

項目	抽出する内容
Google Play での紹介ページ	カテゴリ
	レビュー数
	レビュー評価の平均
	住所，メールアドレスが記載されているか
	「Web サイトにアクセス」の有無
	開発者が他にアプリを提供している数
iOS 版の紹介ページ	App Store に存在するか
	レビュー数
	レビュー評価の平均
	開発者が他にアプリを提供している数
アプリの公式サイト	公式サイトの有無
	最新情報の更新日
	よくある質問の有無
	公式 SNS の有無
	公式 SNS の投稿数
	公式 SNS の更新日
アプリ名で検索した結果	検索件数
	検索結果の上位 3 つが関連しているか
	関連キーワードの数

表 3 アプリの紹介ページが特徴量であるデータセット

アプリ群名	サンプル数	悪質なアプリ数
出会い系	142	8
音楽	162	4
TikTok	127	39
お小遣い	140	5
漫画	105	2
カメラ	82	6

表 4 利用権限も特徴量として追加したデータセット

アプリ群名	サンプル数	悪質なアプリ数
出会い系	125	4
TikTok	86	21
お小遣い	115	3
漫画	100	0

2019 年 7 月〜10 月の間，対象アプリの情報を定期的に入手し，39 種類の特徴量を入手した．実際に機械学習を行おうとした際にサンプル数不足による問題が生じたので，ジャンル情報も特徴量としたうえで，1 つのアプリ集合として機械学習を行った．さらに，アプリ情報と利用権限の情報を組み合わせて機械学習を行った事例を紹介するために 4 ジャンル 426 個のアプリからなるデータセット (表 4) を用意した．そのデータセットでは，前述の特徴量に加えて， android が用意している利用権限を含む，計 354 種類のアクセス権限も入力となっている．利用権限の入手に失敗したケースが存在したので表 3 と比較して分かるように一部の悪質なアプリの情報が入手できなかった．

5.2 アプリ情報から入手した特徴量による分析の結果

実際に分析を行ったところ，k-最近傍法，サポートベクタマシンでは上手く機械学習を行うことができなかった．特徴量の厳選などを行って，関係のない情報を除去する必要があると考えられる．図 3 に示す通り，ランダムフォレスト，勾配ブースティング決定木では現在のデータセットに対して，偽陽性率が低い状態で，再現率が 7 割程度の精度を持つモデルが構築できそうであることがわかった．一方，線形モデルでは，グラフが直線に近い形になっており，現状あまり精度が高いといえず，特徴量の厳選などを行って，関係のない情報を除去する必要がある．

決定木では分岐をするにあたってどの特徴量をどれだけ重要視したかを見ることができる．ランダムフォレストと勾配ブースティング決定木について特徴量の重要度を確かめた．ランダムフォレストで重要度が高い特徴量は，「App Store に存在するか」，カテゴリである「お小遣い」，「App Store でのレビュー数」，「Google での検索件数」，「住所，メールアドレスが記載されているか」，の順であった．勾配ブースティング決定木では，「App Store に存在するか」，「お小遣い」，「Google での検索件数」，「App Store のレビュー数」，「住所，メールアドレスが記載されているか」の順であった．2 つのモデルの特徴量の重要度の上位 5 つは順位には違いがあったが，同じではあった．また，アプリ情報と利用権限の情報を組み合わせて機械学習を行った事例として，同様の手順でランダムフォレスト，勾配ブースディング決定木，線形モデルを用いて機械学習を行った．図 4 がその結果であるが，全体として図 3 の場合よりばらつきが多く，良い結果にはならなかった．理由として，複数のジャンルが含まれることで精度が低下していること，特徴量の厳選が十分でないことに加えて，テストデータ中に，悪質なアプリの量が少なく，機械学習として十分な精度を実現できなかった可能性がある．

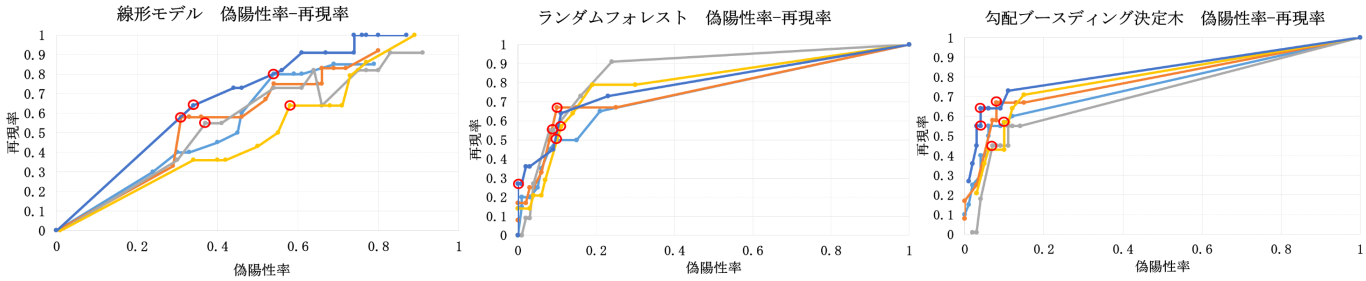

図 3　アプリの紹介ページの特徴量を使用した場合の偽陽性率-再現率グラフ

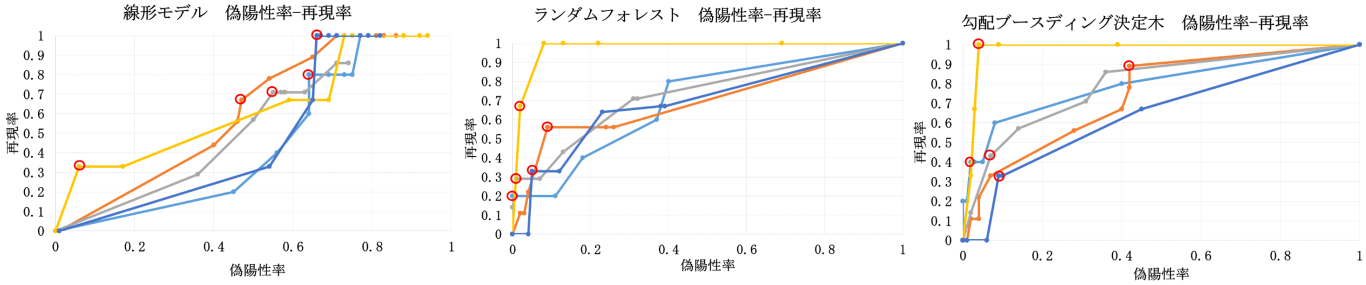

図 4　アプリの紹介ページの特徴量にアクセス権限を加えた場合の偽陽性率-再現率グラフ

6　考察

6.1　システムについての考察

特徴量抽出システム単体としては目的通りの結果を得ることができたが，データが大量に必要な場合，リストアップはしたがバッチファイルを作成して情報をダウンロードするまでの間にアプリが削除されてしまうことがあったので，効率良くアプリ情報を取得できる方法が不可欠である．アプリ情報や利用権限を取得する前に削除されたことによってアプリの順番や内訳が異なり，表をまとめるのに時間を要してしまったという事例が発生し，利用権限の分析結果との連携が上手くできなかった．アプリ情報とアクセス権限を，常に共有できる状態を保つことが必要である．現在は時間軸を考慮した特徴量は更新日や投稿数のみで，日数あたりの情報を考慮した情報も特徴量として考えられるので，そのような特徴量も追加したい．

6.2　得られた情報を用いて機械学習を行う際の課題

機械学習の結果が上手く得ることができなかった原因として，悪質なアプリのサンプル数が不足していたと考えられる．定期的にアプリのリストアップを行い，データの構築回数を増やす必要がある．また，現状では特徴量の厳選や加工について十分な配慮を行っておらず，必要のない情報が多く含まれていたことも考えられる．手法によって結果に差が出ていたことから，それぞれの手法にあった最適化を行い，余分なデータを削除する必要があると考える．決定木において重要な判断材料となっていた特徴量は，2つの手法ともほぼ同じとなり，有力な判断材料の一部は判明しつつあると考えられる．アプリ数を増やした後でも，同じ傾向が得られるかについて調査を行いたい．

7　まとめと今後の課題

本研究では，削除されたアプリから追加のデータ項目が必要な場合を考慮して判断材料となるデータを手元に残すような仕組みを作ることを目的として，データを入手し，管理するような仕組みを作った．実際のアプリに対してデータを入手し，機械学習を行い，手法によって結果に差が出ることを確認した．機械学習の結果の精度を向上させる方法として，データセットに含まれる悪質なアプリのサンプル数を増やすことと，上手く機能しなかった特徴量を求め厳選することが必要である．必要に応じて有効となりうる特徴量を考察して，データセットに加えたい．

参考文献

[1] Google play：https://play.google.com/store/

[2] トレンドマイクロ：“1000 万個を突破した Android 不正アプリの「これから」”，http://blog.trendmicro.co.jp/archives/12960

[3] Zhongmin Ma：“Android Application Install-time Permission Validation and Run-time Malicious Pattern Detection”，Master thesis of Virginia Polytechnic Institute and State University，2013.

[4] 安藤花風里，伊藤美惟：“脅威を引き起こすアプリケーションを アクセス権限などを用いて検出する手法についての考察 ”，南山大学 2018 年度卒業論文，2019.