F-034

機械学習を用いた悪意のある URL の検出方法の提案

A proposal of a method of detecting malicious URLs using machine learning

張偉銘† 高田豊雄†
Zhang Weiming Toyoo Takata

1. はじめに

インターネットの普及に伴い，その影響は我々の生活の各面に深く浸透している．ユーザは URL を介してインターネット上の多種多様な情報に直接的あるいは間接的にアクセスする．しかし，不正なウェブサイトや金融詐欺といった危険も潜んでおり，URL には良質なものと悪質なものが混在している．フィッシング，トロイの木馬，マルウェアなど，多様な攻撃が悪意のある URL を通じて実施されている．2024 年 4 月のフィッシング報告件数[1]は 106,757 件となり，これは前月の 2024 年 3 月と比較すると約 9.9%増加している．このデータは，フィッシング攻撃が依然として増加傾向にあって，特に月間の増加率が顕著であることを示している．このような問題に対処するため，悪意のある URL を検出し，ユーザをこれらの脅威から保護するためのセキュリティ評価技術や手法の開発と研究が不可欠である．

悪意のある URL を検出する技術の歴史は，インターネットの発展と共に進化してきた．初期の対策手法は主にブラックリストに基づくものであったが，攻撃手法の多様化と高度化に伴い，これらの対策は限界を迎えた．そこで，動的な URL フィルタリング技術が登場した．これは，リアルタイムでの URL 解析を行い，悪意のある可能性のある URL を即座にブロックするものである．さらに，近年では機械学習や人工知能を利用した高度な検出技術が導入されており，これにより悪意のある URL の検出精度が飛躍的に向上している．クラウドベースのセキュリティサービスも登場し，これにより膨大なデータをリアルタイムで分析し，迅速に対応することが可能となった．また，ユーザの行動分析に基づく動的な対策も進化しており，ユーザの通常の行動パターンから逸脱する動きを検知して警告を発するシステムも開発されている．これにより，より個別化されたセキュリティ対策が可能となり，悪意のある URL からの脅威に対する防御が一層強化されている．

以上のように，悪意のある URL 検出は，技術の進歩と共に多様な方法で進化を遂げてきた．現在では，機械学習や人工知能，クラウドベースの技術を活用した先進的な対策が主流となっており，これによりユーザを効果的に保護することが可能となっている．

2. 関連研究

Shengら[2]はシグネチャに基づく悪意のあるURLの検出方法を提案している．提案手法では既知のマルウェアや侵入手法についてのシグネチャをデータベース化して大量に保管しており，ストレージ上のファイルや受信したデータをこれと照合することにより危機を検知する．新しいURLにアクセスするたびに，データベースのクエリーが実行される．もしそのURLがブラックリストに載っていれば，悪質と判断され，警告が表示される．この方法の主な欠点は，与えられたリストにない新しい悪意のあるURLを検出することが非常に困難である．また，検出率と適時性も低い．

一方，Heら[3]は悪意のあるURL検出の研究で，モデルに大きく貢献する最適な特徴部分集合を選択するという特徴選択方法を設計した．各特徴がランダムフォレストの各木にどれだけ貢献しているかを判断し，その平均値を取って，最終的に特徴間の貢献を比較する．この論文では，ランダムフォレストのジニ係数(Gini index)を特徴貢献度の計算に使用する．まず，ランダムフォレストのジニ重要度を用いて，初期特徴集合内の各特徴の寄与度を計算する．この論文でHeらは 28個のURL

A proposal of a method of detecting malicious URLs using machine learning

† Weiming Zhang and Toyoo Takata,
Graduate School of Software and Information Science,Iwate Prefectural University

特徴を抽出した．主にURLの構造特徴とセンシティブワード特徴である．詳細を表1に示す．

表1　Heらが示したURLの特徴[2]

Feature 0	ドメインの長さ
Feature 1	ドメインに数字が含まれる
Feature 2	ドメインに特殊文字が含まれる
……	……（論文中でも省略）……
Feature 27	ドメイン名の登録時間が2年未満かどうか

表1の通り，4番目から27番目までの特徴が論文中に掲載されていなかった．そのため，再現実験を行うにあたり，論文中の記述を基に特徴を探索し，最終的に26個の特徴を選択した，その結果を表2に示す．

表2　再現実験で用いた特徴の説明

Feature 0	urlの長さ	Feature 8	ドメインの数
Feature 1	IPアドレス	Feature 9	'https'の使用
Feature 2	ドメイン名の長さ	Feature 10	'/'の使用
Feature 3	数字の占める割合	Feature 11	'//'の使用
Feature 4	特殊文字の割合	Feature 12	ポート番号の使用
Feature 5	アルファベットの割合	Feature 13	ファイル拡張子
Feature 6	短いリンクの使用	……	
Feature 7	@記号を使用	Feature 25	パラメータ値に特殊文字が含まれる

Heらの手法では特徴は貢献度に基づいて並べ替えられ，貢献度の低い特徴から順に削除する．得られた特徴部分集合を使用して，同じデータセットでモデルをトレーニングする．モデルの精度，適合率，再現率，F1スコア，AUCを計算し，これら5つの評価指標の平均値を計算する．評価指標の平均値が初期特徴集合に近い場合，最小の特徴数を持つ部分集合を選択する．最終的に最適な特徴部分集合が得られる．

この方法により，必要な特徴の選択と無関係な特徴を減らすことが可能となる．しかし，特定の特徴の貢献傾向に過度に依存すると，悪意のあるURLと良いURLを判別する上で，微細な特徴を見逃す可能性がある．

そのため，本研究では2019年と2021年のデータセットを使って，Heらの論文で提案された方法で実験を行う． 実験結果を図1と図2に示す．実験結果から，二つの年の特k徴の寄与度ランキングが異なることが最重要の特徴が変化していることなどからも明らかである．実験結果から，特徴の貢献度は時間とともに変化する可能性がかなり高いことがわかった．そのため定期的にモデルを更新する必要がある．よって，より効率的な他の特徴抽出法を探す必要がある．

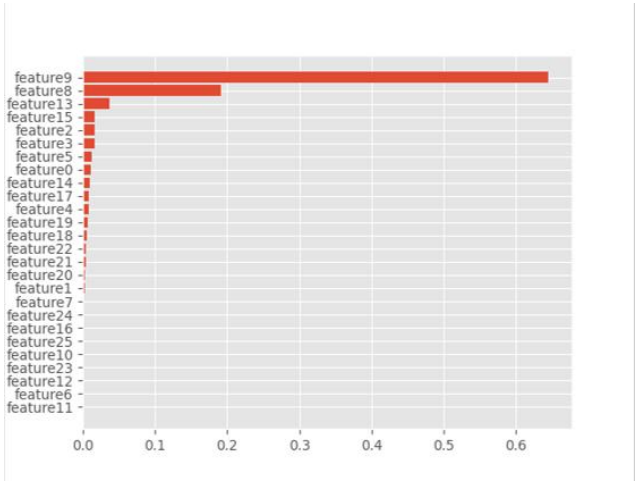

図1　2019年データセットの寄与度ランキング

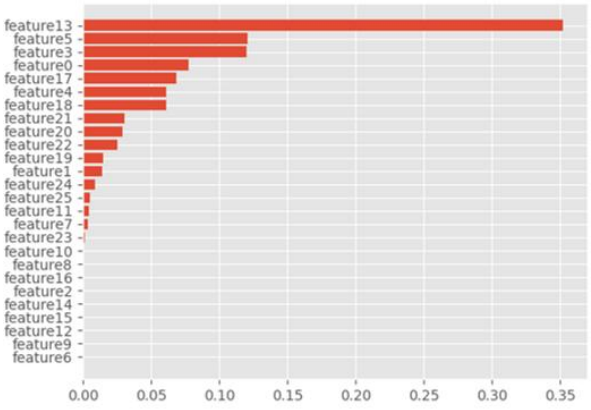

図2　2021年データセットの寄与度ランキング

Heらの論文では，ランダムフォレストが一番性能のよいアルゴリズムであると結論付けている．

ランダムフォレストは多数の決定木のアンサンブルモデルであり，その特性上，特徴量の相互関係や複雑なパターンを捉えきれない場合がある．悪意のあるURLはしばしば複雑なパターンや進化する手法を用いるため，これらを正確に検出するためには，ランダムフォレストよりも精度の高いモデルを探す必要がある．

3. 提案手法

3.1 TF-IDFによる特徴抽出

本研究では特徴抽出方法としてTF-IDFを採用する．ここでは簡単にTF-IDFを紹介する．TF-IDFは，ある単語がどれだけ重要かを評価するための統計的手法であり，元のテキストを機械学習アルゴリズムが使用できるベクトル形式に変換することができる．単語の重要性は，その単語が文書中に出現する回数に正比例して増加して，コーパス中での出現頻度に反比例して減少する．このアルゴリズムは次のように定義される：

$$tf \cdot idf = tf(t, d) \cdot idf(t, D)$$

ここで，tfは頻度であり，これは，文書dに単語tがどれだけ頻繁に現れるかを反映する．idfは逆文書頻度であり，これは，その単語が出現する文書の数に反比例する．Dはコーパスの文書である．最後に，各単語のTFとIDFを掛け合わせて，その単語のTF-IDFスコアを計算する．

3.2 TF-IDFと特徴寄与度の比較

Heらの特徴寄与度に基づく悪意のあるURL検出と本論文で提案するTF-IDFによる特徴抽出方法は，どちらも悪意のあるURL検出のために最も関連性の高い特徴を抽出する方法という点では同じである．

本節ではTF-IDFという特徴抽出方法が特徴寄与傾向に基づく方法よりも優れている点について詳しく説明する．

特徴寄与度に基づく悪意のあるURL検出が，データを詳細に分析して，どの特徴が悪意のあるURLに最も関連しているかを特定する必要がある．特徴の寄与度は時間とともに変化する可能性があるため，定期的にモデルを更新する必要があり，また，重要な特徴を見逃す可能性がある．

一方で，TF-IDFは，テキストデータから特徴を自動的に抽出する．全データセットに基づいて特徴を計算するため，新しいパターンに自然に適応する．単語レベルで動作するため，より細かい粒度の特徴表現を提供することができる．これにより大規模なデータを高い計算効率で処理することに適している．

3.3 gcForestモデルの採用

3.3.1 gcForestモデルについて

gcForest（multi-Grained Cascade Forest）は，様々なタイプの特徴を持つデータを処理するのに特に適した機械学習モデルである．これは2017年でZhouら[4]により，提案された．gcForestはカスケード構造を採用している．各層には多くのランダムフォレストがあり，多層構造となっている．

本研究ではカスケードフォレストの最大層数（max_layers)を5に設定した．5層に設定することで，モデルの複雑さと過学習リスクのバランスの取れた良い性能のモデルが得られた．次に各分類器に含まれる木の数を設定する．

各分類器の木の数（estimators）を10で設定することにより，モデルの性能を一定程度保証しつつ，計算コストが過度に高くならないように設定することができた．

本研究では各層は4つのランダムフォレストから成り立っており，2つのランダムフォレストと2つのエクストリームフォレストがある．各フォレストはデータに対して訓練を行い，図3，結果が出る．この結果はフォレストが生成するクラスベクトルと呼ばれる．過学習を避けるために，各フォレストに対する訓練データはk分割交差検証によって供給される．各層は最終的に4つのクラスベクトルを生成し，次の層では前の層の4つのクラスベクトルと元のデータを新しい訓練データとして使用する．このプロセスが積み重なっていき，最終層ではクラスベクトルを平均し，それを予測結果とする．

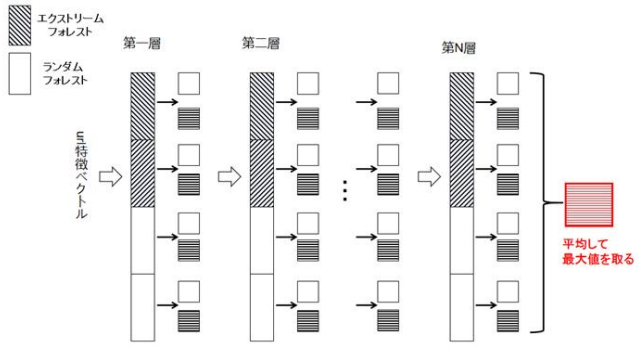

図 3　gcForest による悪意のある URL 検出の流れ

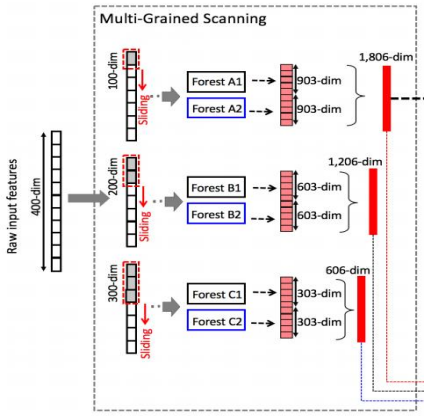

図4　gcForestの多粒度スキャン

3.3.2　gcForest のメリット

多くの学術研究で, gcForest（多粒度カスケードフォレスト）モデルは, 分類問題における顕著な効果により注目されている. がんサブタイプの識別[5], 服装識別[6]の目標分類まで, gcForestはさまざまなデータセットでその強力な分類能力を発揮している. これらの先行する成功した応用例に基づき, 悪意のあるURLの分類においてもgcForestモデルが高精度かつ効率的な識別を実現可能であることが予想される. したがって, 本研究では, 悪意のあるURLの識別とフィルタリングにおけるgcForestの可能性を探求し, インターネットセキュリティツールとしての有効性を実験的に検証する.

前節で述べた通りgcForest独自の多粒度スキャンメカニズム図4により, 局所的および全体的な特徴をよりよく捉えることができる. これはURLの文字の組み合わせや構造分析に特に重要である. gcForestモデルはカスケード構造を採用しており, 層ごとのランダムフォレストを通じて, より複雑なパターンを抽出することができる. これは, 悪意のあるURLの微細な特徴とルールを捉えるのに役立つ. gcForestは, ランダムフォレストの並列計算能力を利用して, 大規模データセットを効果的に処理する.

4.実験

4.1 実験手順

まず, 悪意のある URL と良い URL のデータセットを収集する. 次に, このデータセットの前処理を行い, URL をクリーンアップし, トークン化するなどして機械学習モデルが処理できる形式に変換する. 特徴抽出方法（例えば TF-IDF）を適用してから, データに基づいて機械学習モデルを訓練する. 訓練されたモデルの性能は再現率などの指標を用いて評価され, URL を正しく分類する効果を確認する.全体の流れを図 5 に示す.

図 5　提案手法による悪意のある URL 検出の流れ

4.2 データ収集

提案手法の妥当性を検証するために, まず検証用データセットを作成する. Patgiri ら[7]によれば,

80:20 の割合で分割する方が，より精度の高い分類が可能であることが示された．良い URL と悪意のある URL は kaggle から収集し，合計 40000 件の URL を取得した．これは 2023 年のデータセットである．良い URL は 20000 件,悪意のある URL が 20000 件である．この中で, 80%が訓練用データ, 20%がテスト用データである．データセットの具体的な数は，表 3 に詳細を示す．

表3 データセットの数

	悪意のあるURL	良いURL	合計
訓練データ	16000	16000	32000
テストデータ	4000	4000	8000

4.3 特徴エンジニアリング

URL（Uniform Resource Locator, 統一リソースロケーター）は，インターネット上のリソースを特定するためのアドレスである. URL の一般的な形式は次のとおりである：

プロトコル://ドメイン名:ポート番号/ファイルパス?クエリ

この形式を使って，インターネット上の特定のリソースを指し示し，アクセスすることができる．

4.3.1 データセットの前処理

まず, URL 内のプロトコル部分である[http], [https]などのフィールドは, URL の分類には基本的に影響を与えないため，これらの部分を取り除く．

次に，機械学習では URL をトークン化（分割）する必要があり，それを基にして URL の単語ベクトル表現を実現する．本研究では正規表現を使用してテキストを分割する．分割の例を以下の図 6 に示す：

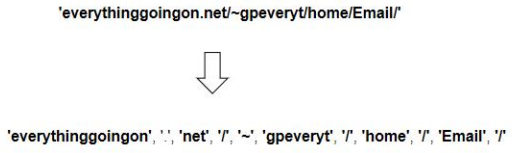

図6 URL分割の例

最後に, TfidfVectorizer ツールを使用して，元のテキストを，機械学習アルゴリズムが使用できる TF-IDF のベクトル形式に変換する．

4.3.2 TF-IDFによる計算結果

TF-IDFにより得られた結果を以下の図7に示す．図7は, TF-IDF値がゼロでない場合の単語とそのTF-IDF値を表示しており，各列は左から順に, Feature, Sample, TF-IDFを表す．詳細は以下の通り:

Feature: URLを分割した後の単語やフレーズ,

Sample: サンプル番号（データセット内のインデックス），

TF-IDF: 計算されたTF-IDF値.

図7からわかる通り,全体で231,048がゼロでないことが確認された.

	Feature	Sample	TF-IDF
0	com	0	0.138934
1	savings	0	0.990302
2	citypages	1	0.577626
3	com	1	0.078817
4	dave	1	0.511380
...	...	...	...
231043	dwtrade	39999	0.616393
231044	keybase	39999	0.508015
231045	php	39999	0.175215
231046	post	39999	0.427397
231047	www	39999	0.189367

図7 TF-IDFの計算結果

4.4 実験環境

本研究で使用したコンピュータのオペレーティングシステムは Windows で，メインクロック周波数は 2.5GHz である．また，システムのメモリは 16GB である．特徴抽出およびデータモデリングの実装には, Python 3.10 を使う．

4.5 実験モデルの評価指標

本論文では，モデルの分類性能を評価するために，正解率(Accuracy)，適合率(Precision)，再現率(Recall), F1 スコア(F1‐score)の 4 つの指標を用いた．

4.6　実験結果

まず, He らの研究において提案された特徴貢献傾向に基づく特徴選択方法と，本研究で提案する TF-IDF を用いた特徴抽出方法の比較実験を行う．He らの研究で最も精度が高かったランダムフォレストモデルを使う．実験結果を図 8 に示す.

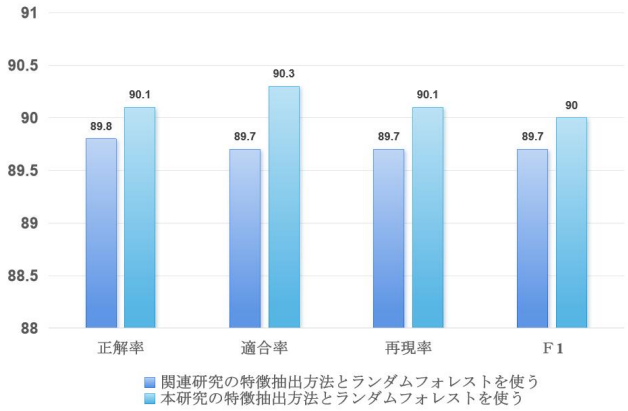

図 8　実験結果（1）

この結果から，本研究の提案手法である TF-IDF 特徴抽出方法の方がより精度が高いことがわかる．

次の実験では特徴抽出方法として本研究で提案した TF-IDF を使う．その上で識別手法として本研究で採用した gcForest と既存のランダムフォレストの比較実験を行う．実験結果を図 9 に示す.

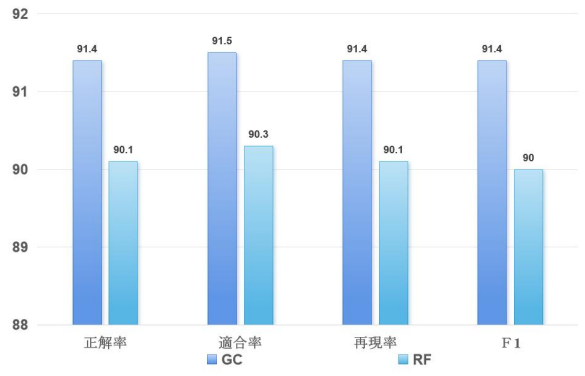

図 9　実験結果（2）

この結果によると, gcForest は,正解率，適合率，再現率, F1 スコアの各指標において，従来のランダムフォレストより高くなっている.

5.まとめ

本論文では，悪意のある URL の検出に関する一手法を提案した．特徴抽出には TF-IDF 法を利用し，実験結果から, TF-IDF がより優れた分類効果を示すことを確認した．また識別アルゴリズムとして, gcForest モデルを採用して，既存のランダムフォレストモデルと比較した．比較結果から，gcForest モデルはより良い結果を得られることができることがわかった.

参考文献

[1] フィッシング対策協議会，月次報告書_フィッシング報告状況, 2024.

[2] S. Sheng, B.Wardman, G.Warner, L.F. Cranor, J. Hong, and C. Zhang, An empirical analysis of phishing blacklists, Proceedings of Sixth Conference on Email and Anti-Spam (CEAS), 2009.

[3] S. He, J. Xin, H. Peng, and E. Zhang, Research on Malicious url Detection Based on Feature Contribution Tendency, IEEE 6th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), 2021.

[4] Z. Zhou, and J. Feng, Deep Forest: Towards an Alternative to Deep Neural Networks, IJCAI, pp. 3553–3559, 2019.

[5] Z. Chen, X. Sun, and L. Shen, An effective tumor classification with deep forest and self-training, IEEE Access,vol.9, pp.100944–100950, 2021.

[6] J. Dai, T. Wang, and S. Wang, A deep forest method for classifying e-commerce products by using title information, ICNC 2020, Big Island, HI, USA, February 17–20, 2020, pp. 1–5. IEEE, 2020.

[7] R. Patgiri, H. Katari, R. Kumar, and D. Sharma, Empirical Study on Malicious url Detection Using Machine Learning, Distrib.Comput.Internet Technol., Springer, pp. 380–388, 2019.