機械学習を用いた脆弱性情報の深刻度評価

渡辺 龍[1,a)]　松中 隆志[1]　窪田 歩[1]　浦川 順平[2]

概要：脆弱性の深刻度や危険性を表す指標として，各種団体から公表される注意喚起を活用することができる．注意喚起はセキュリティに関わる団体やベンダーが，脆弱性や脅威の内容，攻撃による被害状況等から判断し，危険性の高い脆弱性やサイバー攻撃に対して注意を促すものであり，対象となった脆弱性は深刻度が高く，迅速な対処が必要であるものと考えられるからである．しかしながら，注意喚起の公表には，該当する脆弱性の公開から注意喚起の公表までに時間を要することがある．この時間を短縮できれば，脆弱性の深刻度をいち早く推測ことが可能である．このため，著者らは，公開された脆弱性が注意喚起に相当するような深刻度を持つ脆弱性であるかどうかを機械学習により予測する手法を考案した．本稿では，考案手法と精度検証のために実施した評価試験について報告する．

キーワード：セキュリティ対策，脆弱性対応，注意喚起，機械学習

Evaluating the severity of vulnerability information with machine learning

Ryu Watanabe[1,a)]　Takashi Matsunaka[1]　Ayumu Kubota[1]　Jumpei Urakawa[2]

Abstract: The security alerts issued by various organizations can be used as an indicator of the severity and danger of vulnerabilities. The alerts are issued by security-related organizations or vendors based on the published vulnerability information, threats, and damages caused by the attacks to warn the public of high-risk vulnerabilities or cyberattacks. However, it may take some time between the disclosure of the vulnerability and the release of the alert. If this delay can be shortened, it will be possible to guess the severity of the vulnerability earlier. For this purpose, the authors have proposed a machine learning method to predict whether a disclosed vulnerability is severe enough to publicize an alert. In this paper, the authors describe our proposed scheme and the evaluation tests we conduct to verify its accuracy.

Keywords: Security measure, Vulnerability management, Security alert, Machine learning

1. はじめに

近年，様々な分野で ICT 化が進み，多くの業務やサービス，セミナーや授業等がオンラインで実施・提供されている．しかしながら，同時に，ウイルス，マルウェア，ランサムウェアなどによる被害も増加している．サーバシステム，ネットワーク，デバイス，アプリケーションなどの脆弱性を悪用し，サーバや端末を不正に乗っ取り，サイバー攻撃の踏み台にするケースや，情報窃盗などの損害を与える行為の報告が後を絶たない．2020 年以降，新型コロナウイルス感染症 (COVID-19) は世界的に猛威をふるい，オンライン化はさらに進んだ．その結果，これまで見られなかったような攻撃や問題も顕在化してきており，システム運用におけるセキュリティ対策の重要性はますます高まっている．

脆弱性に関する情報とその脅威が日々報告され，併せて脆弱性への対策手法などが公開されている．しかしながら，その全てをタイムリーに収集し，危険性や深刻度を判

1　株式会社 KDDI 総合研究所
KDDI Research Inc.
2　KDDI デジタルセキュリティ株式会社
KDDI Digital Security Inc.
a)　ry-watanabe@kddi-research.jp

© 2021 Information Processing Society of Japan

断したうえで，実際の対策を施すことは大変困難である．実際のところ，報告される脆弱性の数は大変多い．図 1 は，National Vulnerability Database[1] に登録された脆弱性情報の件数 (CVE[2] の数) である，2017 年ごろから急速に増加し，2020 年にはおよそ 18,000 件と多くの脆弱性が報告されている．また，被害を及ぼす攻撃コードの半分は，対象となる脆弱性の公開から 2 週間以内に出現すること [3] が報告されており，セキュリティ対策においては，脆弱性や脅威に関する情報を適切に収集し，早急に対処することが重要である．

脆弱性対応に関する技術レポート [4], (2.2.2 項) では，取得すべき情報として，脆弱性情報や，最新のニュース，ベンダーサイトの他に，公的機関における注意喚起を挙げている．これらの情報を迅速に収集し，適切に活用することができれば，セキュリティリスクが顕在化する前に早期に対策を講じることができる．注意喚起 [5], [6], [7], [8], [9] はセキュリティに関わる団体やベンダーが，発覚した脆弱性や脅威，攻撃による被害状況等を把握し，危険性の高い脆弱性やサイバー攻撃に対して注意を促すものである．システム管理者は，注意喚起をきっかけにセキュリティ対策を見直すケースもあり，対応すべき脆弱性の優先度付けに必要な情報の一つとなっている．しかしながら，注意喚起の公表に関しては，ある脆弱性が公開されてから当該の脆弱性に基づく注意喚起が公表されるまでに時間がかかることがある．この時間差を解消することができれば，セキュリティ対策上，大きなメリットとなりえる．すなわち，ある脆弱性が公開された際に，その脆弱性が公的機関の注意喚起に相当するような深刻度の高い脆弱性であるかどうかを判別できれば，脆弱性の公開とほぼ同時にクリティカルな脆弱性であると認識することができ，優先して対策をすべき脆弱性であるかを迅速に判断することができる．

著者らは，セキュリティ対策支援の手法として，公開された脆弱性情報の深刻度を，機械学習を用いて推定する手法を提案している．これは，過去に公開された注意喚起と脆弱性に関する情報とを学習データとして，ある脆弱性が注意喚起に相当するような脆弱性であるかを機械学習により判定するものである．本稿では，提案手法を説明し，また，精度評価のため行われた評価試験についても報告する．機械学習による推定を実際に活用する際には，過去の事実から現在及び未来を予測することとなる．このための評価として，学習データとテストデータの時期を違えた評価も実施した．

2. 関連研究

脆弱性に関する情報共有を容易にすることを目的として，共通脆弱性識別子: CVE(Common Vulnerability Exposure) が制定されている [2], [10]．この CVE により，現在では，一つの脆弱性について固有の識別子（CVE-ID）

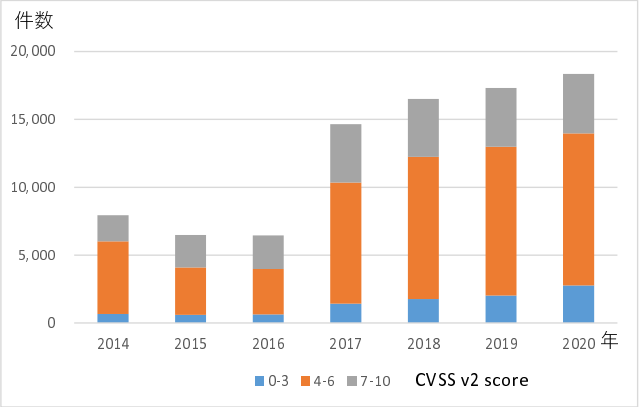

図 1 報告された脆弱性の件数 (CVE 数)
Fig. 1 Number of reported vulnerabilities (CVE).

を付与し，管理されるようになっている (ex. [1], [11])．脆弱性への対処に関し，上記の CVE とともに，脆弱性の深刻度を評価するための指標である共通脆弱性評価システム: CVSS(Common Vulnerability Scoring System) [4], (2.1 節), [12] が広く活用されている．CVSS では，定量的なスコアリング手法が定義 [13] されているが，CVSS の基本値の算出には，攻撃の容易性や守るべき情報資産の価値などが利用されており，実際に攻撃がされるかどうかは考慮していない．このため，CVSS スコアの高い脆弱性への対応を優先する方針では，実際の攻撃には対処しがたいという問題がある．この問題の対処のため，システムに被害が出るような攻撃が実際になされる可能性を考慮した手法が検討されている [14], [15], [16], [17]．例えば，[17] は，過去の脆弱性情報とそれに対応する攻撃コードが生成されたかどうかを学習データとして，機械学習により，新規脆弱性の深刻度を判別する手法を実現している．[18] は，脆弱性対応の優先度付けために，脆弱性の深刻度と侵害確率に基づく指標を統合するモデルを数理最適化によって実現している．このような検討・研究は進められているものの，公的機関が公表する注意喚起に相当するかどうかを指標とし，機械学習を用いた脆弱性の深刻度判別に関する検討は見られなかった．

この他，セキュリティ対策支援のために，機械学習を用いた先行研究は多い．脆弱性や脅威などのセキュリティ関連の投稿を抽出するために機械学習を用いる研究 [19], [20], [21] がその例である．

3. 脆弱性の深刻度予測

ここでは，提案手法である，ある脆弱性が注意喚起に相当するような深刻度を持つかどうかを予測する手法について述べる．具体的には，準備としての各種データの取得とその加工方法，利用した機械学習のアルゴリズム，性能評価試験とその結果を記載する．

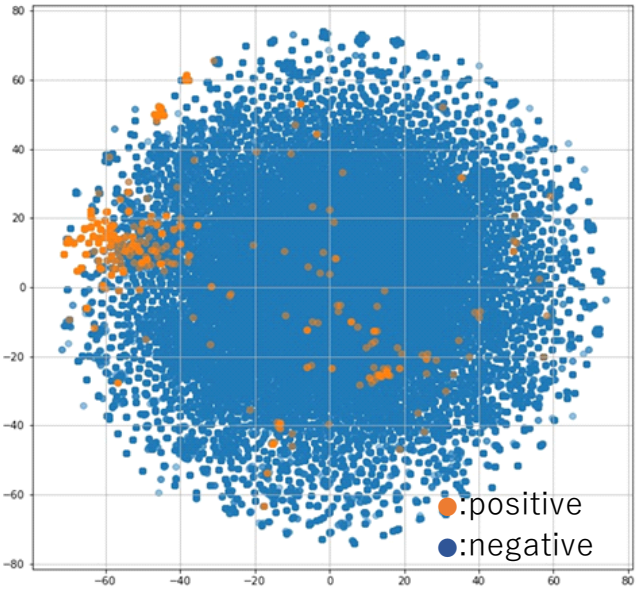

図 2　t-SNE により 2 次元化された特徴ベクトル
Fig. 2　2- dimensional feature vector by t-SNE.

3.1　対象とする注意喚起

本検討で予測のラベルとして利用した JPCERT/CC が公開する「注意喚起」[6] は，歴史が古く，1997 年から公開されている．このため，注意喚起に関する情報は，JPCERT/CC[6] より取得した．また脆弱性に関する情報は VulDB[22] から取得した．VulDB は，脆弱性に加え，脅威 (攻撃コードの公開) に関する情報を提供するデータベースである．VulDB が持つ情報には脅威に関する情報も含まれており，CVE の記述だけから得られる情報よりも活用できる情報が多いと考え，本研究での脆弱性情報の入手先とした．

3.2　モデル構築と評価試験

機械学習のモデル評価の準備として実施した，ラベルデータおよび特徴ベクトルの作成と，学習モデルの性能評価試験について記述する．

3.2.1　注意喚起ラベルデータの生成

取得した注意喚起に関する情報は，JSON 形式にデータを整形したうえで，機械学習のラベルデータとして活用した．

3.2.2　特徴ベクトルの作成

取得した脆弱性情報のレコードには以下の処理を行い，特徴ベクトルを作成した．

(1) ドキュメント作成
- VulDB から取得したレコードについて，entry.title, entry.summary, entry.details の各項目を結合
- CVE 番号が含まれないレコードと重複したレコードを除去

(2) ラベルの付与
- 脆弱性情報のドキュメントに含まれる CVE 番号が，注意喚起のラベルデータに含まれていれば正例として "1" を，含まれていなければ負例として "0" を付与
- 学習データに付与するラベルは，学習データと同じ期間の注意喚起のラベルデータを付与，テストデータについては，全データを使用してラベルを付与

(3) 正規化の実施
- 記号の除去[*1]
- ストップワードを除去
- 単語を原型に変換
- 全単語を小文字化

(4) ドキュメントのベクトル化
- BoW(Bag of Words) により上記ドキュメントのベクトル化を実施
- ベクトルの次元数削減のため，99%以上のドキュメントに含まれる単語と，0.01%以下のドキュメントにしか含まれない単語を除去

得られた特徴ベクトルを，次元削減アルゴリズムである t-SNE (t-distributed Stochastic Neighbor Embedding, t-分布型確率的近傍埋め込み) を用いて 2 次元に圧縮しプロットしたものを図 2 に示す．“positive” の点が左方中央よりに偏在してることから，特徴ベクトルに傾向があることが確認された．

3.2.3　単語の除去

また，モデルの汎用化のため，脆弱性情報の文章から短命であると思われる単語と無意味語の除去を行い，次元数の削減を実施した．汎用化のために削除した単語を次に記す．

- Windows update の更新番号
- CWE 番号
- 数字のみ，年号，バージョン番号
- 一文字語

表 1　モデル評価の諸元
Table 1　Specification of machine learning model evaluation.

項目	説明
データの期間	2017 年 1 月〜2019 年 12 月
データの件数	52,113 件 (正例: 778, 負例; 51,335)
学習アルゴリズム	logistic regression, random forest, xgboost
ハイパーパラメータ	logistic regression: C, penalty random forest: max depth, n estimatiors xgboost: max depth, learning rate, n estimatiors
チューニング	hyperopt によるオートチューニング
評価手法	5-fold cross validation
評価指標	ROC(Receiver Operator Characteristic), PR(Precision-Recall)

[*1] Python の string.puctuation に含まれる記号を除去した．

表 2　評価結果
Table 2　Evaluation results.

アルゴリズム	ROC AUC	PR AUC
logistic regression	0.939	0.415
random forest	0.963	0.624
xgboost	0.960	0.633

表 3　学習データとテストデータを違えた評価でのデータ期間
Table 3　Data period

データ	期間	正例数	負例数
学習データ	2017 年 1 月〜2019 年 6 月	629	39,219
テストデータ	2019 年 7 月〜2019 年 12 月	146	12,119

表 4　評価結果 (期間分離)
Table 4　Evaluation results (data period split).

アルゴリズム	ROC AUC	PR AUC
xgboost	0.978(0.960)	0.644(0.633)

括弧内は期間分離しない場合の評価結果 (表 2 参照)

表 5　単語 (ベンダー，製品) ごとの注意喚起件数
Table 5　Number of security alerts for vender product word.

単語	注意喚起件数
windows (ウインドウズ)	95
microsoft (マイクロソフト)	74
edge	65
adobe flash	60
internet explorer	49
oracle	19
apache	14
bind	13
apache struts	9
apache tomcat	6
cisco	3
paloalto	1
atlassian	1

対象とした期間中の注意喚起 181 件中の件数

表 6　false positive の上位 10 件
Table 6　false positive top ten.

CVE-ID	対象製品	予測値
CVE-2019-0746	Microsoft Edge	0.957
CVE-2019-0610	Microsoft Edge	0.952
CVE-2019-0779	Microsoft Edge	0.949
CVE-2019-1193	Microsoft Edge	0.910
CVE-2018-1258	Oracle Retail Customer Insights	0.894
CVE-2019-0783	Microsoft Windows Internet Explorer	0.878
CVE-2019-1407	Microsoft Windows Server	0.877
CVE-2019-1438	Microsoft Windows Server	0.877
CVE-2018-1258	Oracle Retail Markdown Optimization	0.848
CVE-2019-1192	Microsoft Edge	0.820

表 7　false negative の上位 10 件
Table 7　false negative top ten.

CVE-ID	対象製品	予測値
CVE-2019-5736	runc	5.20×10^{-6}
CVE-2019-18187	Trend Micro OfficeScan	2.81×10^{-5}
CVE-2019-9489	Trend Micro Apex One Management Console	5.42×10^{-5}
CVE-2019-11707	Mozilla Firefox	7.65×10^{-5}
CVE-2019-11539	Pulse Secure Pulse Connect Secure	8.10×10^{-5}
CVE-2019-3395	Atlassian Confulence Server / Data Center	1.46×10^{-4}
CVE-2019-11510	Pulse Secure Pulse Connect Secure	1.60×10^{-4}
CVE-2019-0158	Intel Graphics Performance Analyzer	2.54×10^{-4}
CVE-2019-0163	Intel Broadwell	2.65×10^{-4}
CVE-2019-3396	Atlassian Confulence Server	4.29×10^{-4}

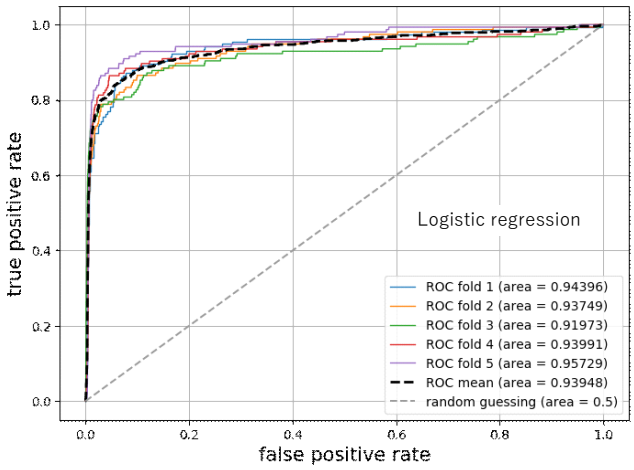

図 3 logistic regression の ROC 曲線

Fig. 3 ROC curve for logistic regression.

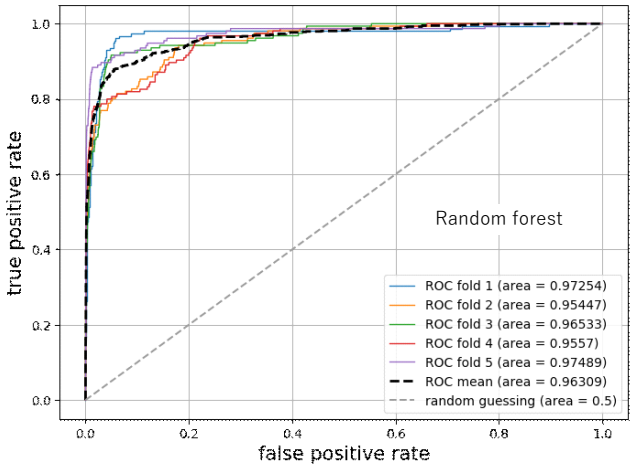

図 4 random forest の ROC 曲線

Fig. 4 ROC curve for random forest.

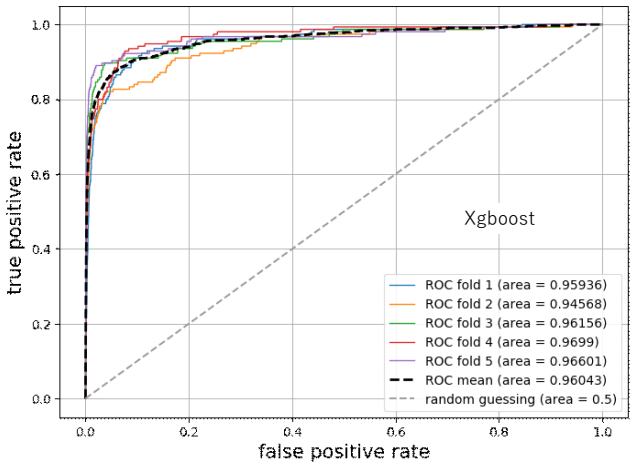

図 5 xgboost の ROC 曲線

Fig. 5 ROC curve for xgboost.

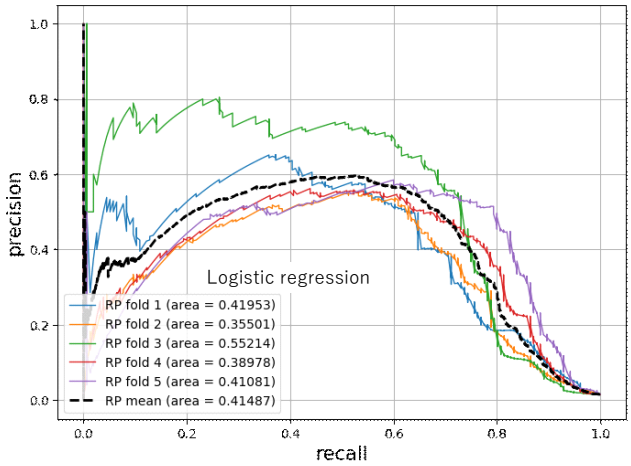

図 6 logistic regression の PR 曲線

Fig. 6 PR curve for logistic regression.

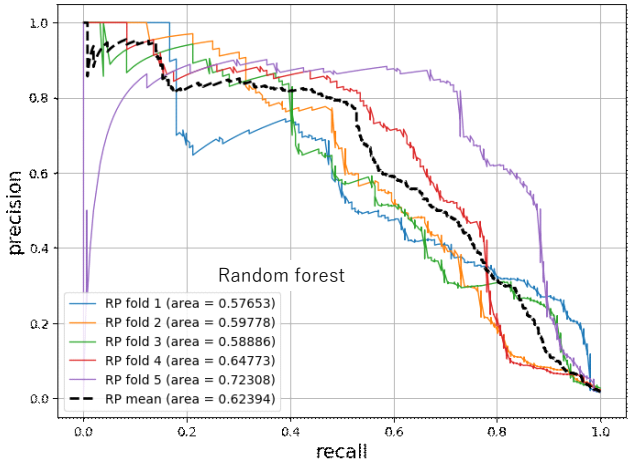

図 7 random forest の PR 曲線

Fig. 7 PR curve for random forest.

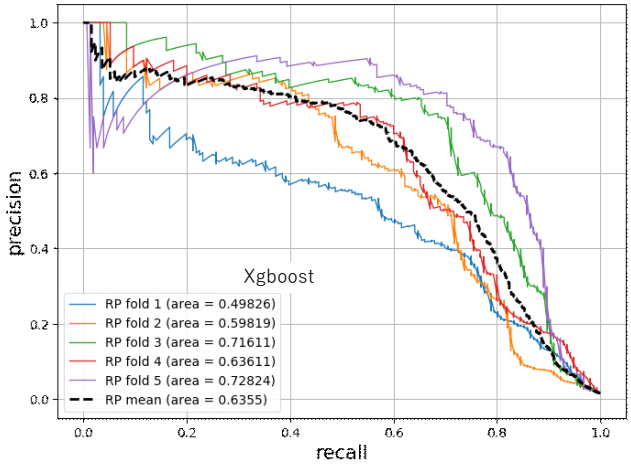

図 8 xgboost の PR 曲線

Fig. 8 PR curve for xgboost.

3.2.4 モデル評価結果

各モデルの評価結果である ROC AUC 及び PR AUC を表 2 に記す．どのアルゴリズムでも，ROC AUC 値では 0.93 以上であった．また，PR AUC の評価では， logistic regression を用いたケースで 0.41 程度，random forest と xgboost を用いたケースでは 0.63 程度の精度であった．モデルごとの ROC 曲線及び PR 曲線を，図 3〜図 8 に示す．

3.2.5 学習データとテストデータの時期を違えた場合の評価

本モデルの汎用性のさらなる検証のため，学習データとテストデータの期間を変更し，評価試験を実施した．この評価では，3 種のアルゴリズムのうち，先の検証で精度が最も安定していた xgboost を利用して実施した．この評価での，学習データとテストデータの分割方法を表 3 に示す．本評価では，学習データとテストデータの期間を分離し，モデル作成の時点で評価する未来に相当する期間の情報を含まないようにしている．評価結果を表 4 に示す．ROC AUC，PR AUC ともに，先の評価と同程度であることが確認された．本評価での ROC 曲線及び PR 曲線を，図 9，図 10 に記す．

3.3 評価結果に関する考察

3.3.1 精度

表 2 に示した評価結果より，3 種の機械学習のアルゴリズムとも ROC AUC による評価では大きな差はなかったが，PR AUC による評価では，logistic regression よりも，random forest と xgboost の方が，性能が高かった．これは，本評価での特徴ベクトルの傾向が，線形モデルに基づくアルゴリズムよりも，決定木ベースのアルゴリズムの方に適合しているためと考えられる．また，全体的に見て，xgboost が安定していることが確認された．表 8 に示される PR AUC 曲線で見た場合，もっとも精度の良い xgboost において，カバー率が 80%(recall 0.8) の時，正答率が 40%(precision 0.4) 程度であった．これは，注意喚起に相当するとの判断が正しい確率が 40%で，また本来注意喚起に相当する脆弱性のうち 8 割を拾うという精度であることを示している．また，表 4 に示された評価結果より，学習データとテストデータの期間を分離した場合の評価では，もとのモデルの精度が保たれていることが確認された．しかしながら，表 10 に示される PR AUC 曲線からは，カバー率 80%の時，正答率が 30%程度と精度的に劣化することが確認された．実際の運用を想定した場合，過去であるこれまでの脆弱性情報から，今後報告される脆弱性情報が注意喚起に相当する深刻度であるか否かを予測する必要があるため，こちらの値がより現実に近い形で利用した場合での精度となる．この精度が許容できるかどうかは，実運用の場に委ねらるため断言はできない．しかしながら，脆弱性対策の際の優先順位付けのための一指標にはなりえると考える．

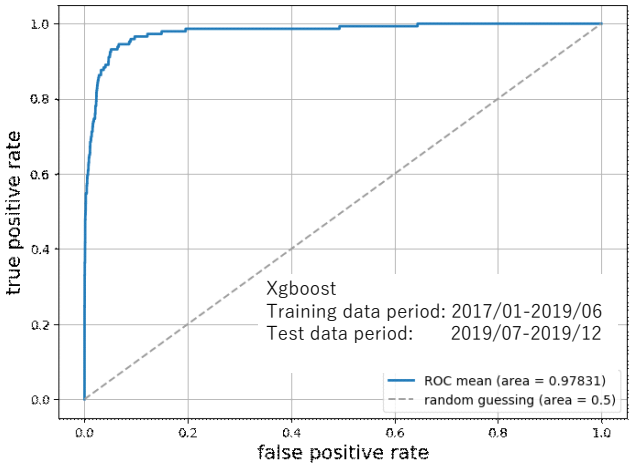

図 9 xgboost の PR 曲線 (期間分割)
Fig. 9 ROC curve for xgboost (data period split).

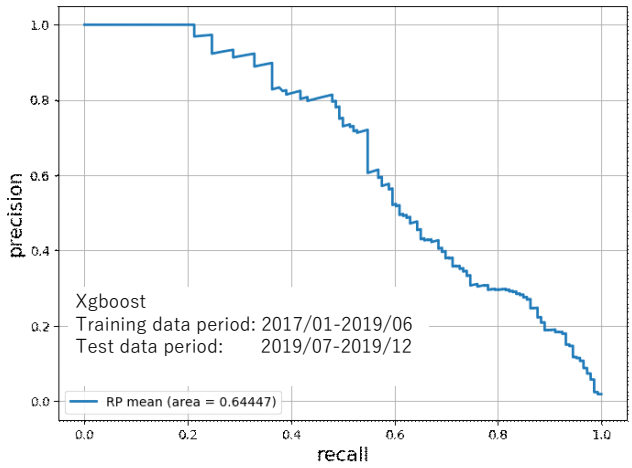

図 10 xgboost の PR 曲線 (期間分割)
Fig. 10 PR curve for xgboost (data period split).

3.3.2 誤検知

表 5 に示した様に，対象とした JPCERT/CC の注意喚起では，Microsoft, Adobe flash, Oracle 製品など著名なベンダーの製品に関する注意喚起が多い．このため，当該情報を含む事例は注意喚起に相当すると判断される傾向が強い．実際，誤検知のうち，false positive の上位である，負例のうち注意喚起に相当する可能性が高く算出されたケースは，表 6 に示した様に，同製品が多く，false negative の上位である，正例のうち注意喚起に相当する可能性が低く算出されたケースは，表 7 に示した様に，そうでないもので占められた．学習データの傾向に沿っているためこの結果は正しいと言える．この点を改善するには，データの増加に加え，別の情報源を活用することなどが考えられ．あるいは，出現頻度の上位/下位のワードに特化して別々のモデルを構築するなどモデルを分割する方法も考えられる．

4. おわりに

本稿では，セキュリティ対策への支援を目的として，機械学習により注意喚起に相当するような深刻度を持つ脆弱性情報を判別する手法について，モデル評価を実施した．学習データとテストデータの期間を違える場合も含め，アルゴリズムに xgboost を利用した評価試験では，ROC AUC で 0.96 以上，PR AUC で 0.63 以上の精度を確認した．本手法が，迅速な脆弱性対応に貢献できると期待する．

謝辞 本研究の一部は，総務省契約「脆弱性情報の高精度な深刻度・信頼度評価に関する実証実験の請負」の成果を含みます．ここに感謝の意を表します．

参考文献

[1] NVD: National Vulnerability Database, National Institute of Standards and Commerce (online), available from ⟨https://nvd.nist.gov⟩ (accessed 2021-05-01).

[2] CVE: Common Vulnerability Exposures, MITRE Corporation (online), available from ⟨https://cve.mitre.org/⟩ (accessed 2021-04-30).

[3] KENNA Security: PRIORITIZATION to PREDICTION Analyzing Vulnerability Remediation Strategies, KENNA Security (online), available from ⟨https://www.kennasecurity.com/resources/prioritization-to-prediction-report/⟩ (accessed 2021-05-18).

[4] IPA: 脆弱性対策の効果的な進め方 (実践編) 第 2 版〜脆弱性情報の早期把握、収集、活用のスゝメ〜, IPA Technical Watch, 独立行政法人 情報処理推進機構 セキュリティセンター (2019).

[5] IPA: 重要なセキュリティ情報一覧，情報処理推進機構（オンライン），入手先 ⟨https://www.ipa.go.jp/security/announce/alert.html⟩（参照 2021-05-06）.

[6] JPCERT/CC: 注意喚起，JPCERT コーディネーションセンター（オンライン），入手先 ⟨https://www.jpcert.or.jp/at/2021.html⟩（参照 2021-05-06）.

[7] 警視庁：警視庁セキュリティポータルサイト, 警視庁（オンライン），入手先 ⟨https://www.npa.go.jp/cyberpolice/⟩（参照 2021-05-06）.

[8] US-CERT: National Cyber Awareness System, Cybersecurity and Infrastructure Security Agency (online), available from ⟨https://us-cert.cisa.gov/ncas⟩ (accessed 2021-05-06).

[9] LAC WATCH: 注意喚起の記事，株式会社 LAC（オンライン），入手先 ⟨https://www.lac.co.jp/lacwatch/alert/⟩（参照 2021-05-06）.

[10] IPA: 共通脆弱性識別子 CVE 概説，独立行政法人 情報処理推進機構セキュリティセンター（オンライン），入手先 ⟨https://www.ipa.go.jp/security/vuln/CVE.html⟩（参照 2021-5-13）.

[11] JPCERT/CC: Japan Vulnerability Notes, JPCERT コーディネーションセンター (online), available from ⟨https://jvn.jp/⟩ (accessed 2021-05-013).

[12] CVSS: Common Vulnerability Scoring System, Forum of Incident Response and Security Teams (online), available from ⟨https://www.first.org/cvss/⟩ (accessed 2021-05-01).

[13] IPA: 共通脆弱性評価システム CVSS 概説，独立行政法人 情報処理推進機構セキュリティセンター（オンライン），入手先 ⟨https://www.ipa.go.jp/security/vuln/CVSS.html⟩（参照 2021-5-13）.

[14] Mehran, B., Lawrence, K. S., Stefan, S. and Geoffrey, M. V.: Beyond heuristics: learning to classify vulnerabilities and predict exploits, *Proc. 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 105–114 (2010).

[15] Carl, S., Octavian, S. and Tudor, D.: Vulnerability Disclosure in the Age of Social Media: Exploiting Twitter for Predicting Real-World Exploits, *Proc. 24th USENIX Security Symposium (USENIX Security '15)*, USENIX, pp. 1041–1056 (2015).

[16] Chaowei, X., Armin, S., Yang, L., Bo, L., Mingyan, L. and Tudor, D.: From patching delays to infection symptoms: using risk profiles for an early discovery of vulnerabilities exploited, *Proc. 27th USENIX Security Symposium (USENIX Security '18)*, USENIX, pp. 903–918 (2018).

[17] Jay, J., Sasha, R., Benjamin, E., Michael, R. and Idris, A.: Exploit Prediction Scoring System (EPSS), *arXiv:1908.04856* (2019).

[18] 山嵜麿与，佐々木満春：数理最適化による CVSS と侵害予測モデル統合，2021 年暗号と情報セキュリティシンポジウム，電子情報通信学会 (2021).

[19] Triet, H. M. L., David, H., Roland, C. and M., A. B.: PUMiner: Mining Security Posts from Developer Question and Answer Websites with PU Learning, *Proc. 17th International Conference on Mining Software Repositories*, ACM, pp. 350–361 (2020).

[20] Andrei, L. Q., Susan, M. and Keegan, B.: Eavesdropping Hackers: Detecting Software Vulnerability Communication on Social Media Using Text Mining, *Proc. The Fourth International Conference on Cyber-Technologies and Cyber-Systems*, IARIA, pp. 41–48 (2019).

[21] Andrei, L. Q., Susan, M. and Brian, K.: Detecting Hacker Threats: Performance of Word and Sentence Embedding Models in Identifying Hacker Communications, *Proc. 27th AIAI Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, CEUR, pp. 116–127 (2019).

[22] VulDB: Vulnerability Database, VulDB (online), available from ⟨https://vuldb.com/⟩ (accessed 2021-05-06).