表層解析の効率化に向けた情報鮮度判別方式の評価

檜垣 龍徳[1] 竹原 一駿[1] 本部 建大[1] 楠目 幹[1] 西岡 大助[1] 西山 賢[2] 合田 翔[2] 最所 圭三[1]
喜田 弘司[1]

概要：サイバー攻撃の高度化に伴い，たとえ侵入されても，素早く，被害を最低限に抑えるための事前対策が必要の必要性が増している．そのために，日頃から脆弱性や攻撃などの情報を収集，分析する表層解析が重要である．しかし，技術力，人材共に充足していない組織などでは，満足に実施できていない．そこで我々は，表層解析を支援するシステム“CSICOS”を開発している．本論文では，セキュリティ情報に紐付いた外部の情報を調査することで，セキュリティ情報の公開日時，必要性を判定する手法を提案し，提案手法に対して実施した評価実験について述べる．評価実験では，実際に表層解析を業務で実施している企業のセキュリティ技術者を被験者に，提案手法の効果を測定した．評価実験の結果，公開日時の推定の評価が高いこと，最新情報及び動向の把握は改善されたこと，表層解析であってはならない情報の見落としが発生していることが判明した．

Evaluation of Information Freshness Discrimination Method for Improvement of Surface Analysis Efficiency

TATSUNORI HIGAKI[1] ICHITOSHI TAKEHARA[1] TATSUHIRO MOTOBU[1] MOTOKI KUSUME[1]
DAISUKE NISHIOKA[1] MASARU NISHIYAMA[2] SHOU GOUDA[2] KEIZO SAISHO[1] KOJI KIDA[1]

1. はじめに

近年，サイバー攻撃の高度化に伴って，早期に攻撃に気付くことや従来の入口対策では攻撃者の侵入を防ぐことが困難になっている．実際に，CrowdStrike 社の調査によると，”ある侵害が最初に発生した時から検知された時までの期間”を示すドエルタイムの平均は 2018 年から 2019 年の間に 10 日も伸びていることがわかっている [1]．こういった背景からサイバー攻撃対策の考え方として，侵入されることを前提として，侵入された時に被害を最小限に食い止めること，迅速に復旧することができるように事前対策をすることが重要視されている [2]．この体制を整えるためには，インシデントの発生直後に対応できるように，事前に複数のソースから原因となる脆弱性情報や攻撃動向などのセキュリティ情報を収集して突合することで，調査，分析をする表層解析が必要となる．表層解析は，CSIRT などに属して，組織のセキュリティ対策に従事するセキュリティ管理者が，定常業務として実施することが増えている．しかし，NRI のアンケートによると，アメリカとシンガポールでは 80%を超える企業が CSIRT を構築しているのに対して，日本では 36%弱の企業しか構築できていない．また，CSIRT を構築するために必要となるセキュリティ人材が充足していると回答した企業は，アメリカとシンガポールでは 80%を超えているのに対して日本では 10%以下といずれも低い水準にあることがわかる．こういった実態から，日本で満足に事前対策を実施できているのは極少数の組織であることが推測される [3]．

そこで我々は，事前対策で特に重要である表層解析を効率的に進める支援をするために，AI によるセキュリティ情報解析支援システム“CSICOS”を試作し，評価実験を実施した．その結果，表層解析においてセキュリティ情報の中でも初出の新しい情報，信頼性の高い情報，流行りの情報といった情報が特に重要視されることが確認できた [4]．我々は，この 3 つの観点をまとめて情報鮮度と呼ぶ．そこで，情報鮮度の観点の内，新しさに着目して、公開日時を

[1] 香川大学
Kagawa University
[2] (株)STNet
STNet, Inc.

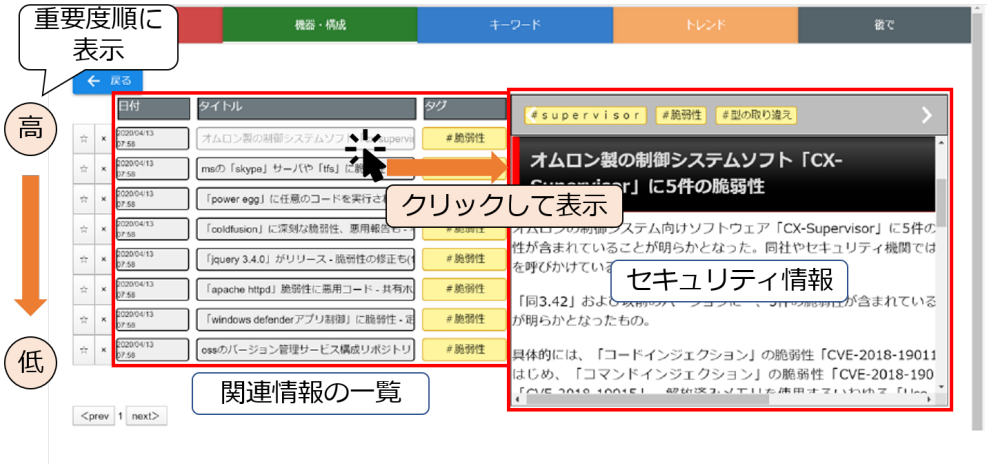

図 1　利用画面例

推定する手法を提案した [5]．本論文では，その評価実験の内容と結果について述べる．

2. CSICOS について

2.1 CSICOS の特徴

CSICOS の特徴を利用画面 (図 1) より説明する．CSICOS は，セキュリティ管理者が求める情報をすぐに見つけられるように提供できることが特徴である．画面左側に任意の話題の関連情報が一覧で表示され，タイトル領域を選択すると，画面右側に選択したセキュリティ情報が展開される．画面左側の関連情報は，代表語について多く言及している (重要度の高い) ものから並んでいる．画像の例では，脆弱性が関連情報群の代表語である．これらを画面の表示を実現するために，セキュリティ情報を SNS やベンダブログなどの複数のサイトから一括して自動で収集して，セキュリティ管理者が分析しやすいように内容ごとに分類して整理している．

2.2 CSICOS の構成

CSICOS は，セキュリティ情報を集める収集部，分析し整理する解析部，セキュリティ管理者に提供するフロント部の 3 つから構成され，各部はデータベースを介して連携している (図 2)．セキュリティ情報は収集部にて集められ，解析部で整理された後にフロント部によってセキュリティ管理者に提供される．本論文では，この内の解析部について述べる．

3. 表層解析について

3.1 表層解析の現状

表層解析は，攻撃への対策をいち早く講じるために，組織で被害が発生する前に様々なソースから公開されたセキュリティ情報を突き合わせることで，攻撃の兆候や特徴，シナリオなどを分析し，対策体制を整えるための作業である．

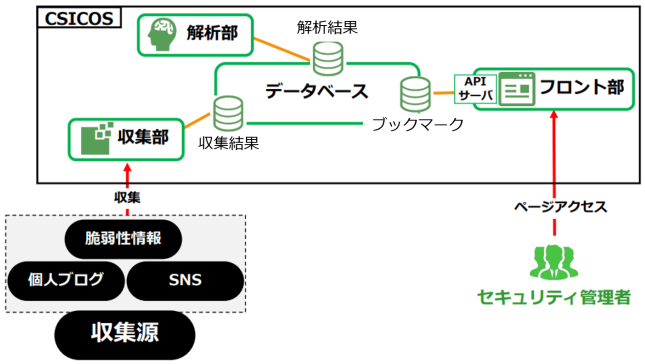

図 2　システム構成図

表層解析を一般化すると以下の流れで表すことができる (図 3)．

作業①　予め決めた複数 Web サイトを訪れて，新しい情報が公開されているか確認する．

作業②　自組織に影響を及ぼす可能性のある情報があるか分析する．

作業③　影響を及ぼす可能性のある情報の関連情報を充分に調査する．

作業④　調査したすべての情報を突合する．

作業⑤　突合結果を踏まえて対策を検討する．

これらの作業を定常業務として毎日複数名で数時間程度の限られた時間の中で実施している．作業③では，調査対象となる情報源が無数にあるために検索作業が繰り返し発生する．作業④では，調査した情報の中から必要な情報を判断するために，調査した情報量に応じた突合作業が繰り返し発生する．

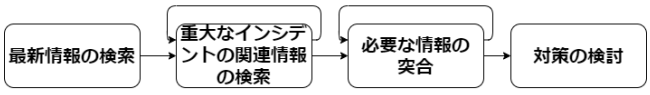

図 3　表層解析作業の流れ

表 1　SolarWinds Orion Platform の脆弱性に関する動向

日付	出来事
2020 年 12 月 8 日	FireEye がインシデントの発生を報告
2020 年 12 月 13 日	SolarWinds 社が SolarWinds Orion Platform の脆弱性を発表
2020 年 12 月 21 日	SolarWinds 社が同製品の新たな脆弱性を発表
2020 年 12 月 28 日	JVN が SolarWinds 製品の脆弱性を CVE-2020-10148 として発表

3.2　表層解析の課題

表層解析において分析するインシデントの動向は，日毎に変化する．例えば，SolarWinds 社 SolarWinds Orion Platform の脆弱性に関しては，初めに脆弱性の影響とはわかっていなかったが，2020 年 12 月 8 日に FireEye がインシデントを発表した．その後同月 13 日に SolarWinds 社から公式アナウンスが出されている．その後，同月 21 日に SolarWinds 社から同じ製品に関する新たな脆弱性が発表され，同月 28 日になって JVN が CVE-2020-10148 として発表をしている (表 1)．このためインシデントに対しては，関連するセキュリティ情報を時系列を追って分析して，常に最新の動向を把握する必要がある．しかし，Web 上で公開されているセキュリティ情報の中には，詳細な公開日時が明記されていないものもある．また，一般的に検索結果は記事の公開順になっていない場合がほとんどである．これらの要因から，インシデントの動向を時系列で追うこと，最新情報を把握することは困難になっている (課題①)．

また，セキュリティ情報の中には，動向を追う必要性の低いインシデントに関連するものや，信頼性の低いものもある．表層解析においては，これらの情報を正しく分析して取捨選択をする必要がある．しかし，専門知識を有した人材が少ない組織では，作業②において内容によっては必要な情報か分析できる人材がいないことがある．また，1 人が分析するセキュリティ情報の量が膨大となるため，作業④において 1 人にかかる負担が大きくなる上に時間がかかる．そのため，漏れなく速やかに必要性の高いセキュリティ情報を判断して分析することが困難になっている (課題②)．

4.　提案手法

4.1　提案手法の概要

3.2 節で述べたそれぞれの課題を解決する手法を提案する．課題①に関しては，セキュリティ情報に対して，本文や関連情報から，公開日時を推定して付与する公開日時推定の機能を提案する．これによって，最新の状況や動向の把握が用意になると考える．課題②に関しては，セキュリティ情報を参照している情報に基づいて，必要な情報か判定する不要判定の機能を提案する．これによって，必要性，重要性の高い情報のみを提供できるようになると考える．

4.2　公開日時推定

課題①に対しては，公開されているすべてのセキュリティ情報の公開日時を推定することで，時系列を追って分析するできるようになると考える．ここで，セキュリティ情報を参照する，詳細な公開日時のわかる外部ソースが見つけることができれば，参照されたセキュリティ情報は，少なくともそれより前に公開されたことがわかる．また，セキュリティ情報が参照されるのは公開日近辺が多くなると期待される．そのためまず，セキュリティ情報中に表記された時間情報を正規表現によって取得する．次に，セキュリティ情報を参照する外部ソースを調査する．調査した外部ソースの中で，投稿日時がセキュリティ情報から抽出した時間情報と合致するものを取り出す．取り出した外部ソースの投稿日時の分布を確認し，投稿された外部ソースが最も多い日の中で，1 番最初に投稿されたものの投稿日時をセキュリティ情報の公開日時として考える．このように外部ソースとセキュリティ情報双方の時間情報を併せて分析することで，セキュリティ情報の公開日時を推定できると考える (図 4)．

外部ソースとしては，公的組織のサイト，個人ブログ，Twitter を始めとした SNS が挙げられる．これらの外部ソースの特性を分析する．公的組織のサイトは，一般的に正確性，信頼性の高い内容で公開されることが期待されるため，内容の精査などのために公開までに時間がかかると考えられる．個人ブログは，公的組織のサイトよりは内容の精度に対する期待は薄いと考えられるが，ブログの体裁を整える以上，公開までに一定の時間を要すると考えられる．SNS，特に Twitter は，内容の精度に対する期待はその他のソースと比べて薄いと考えられ，内容も個人の感想やコメントのような体裁であるため，公開までに多くの時間は要しないと考えられる．また，すべての投稿に詳細な公開時間が付与されているため，セキュリティ情報の公開日時を推定するための外部ソースとして有効であると考えられる．そこで，Twitter 上で任意のセキュリティ情報を参照する投稿を検索し，その投稿の公開時間を参照することでセキュリティ情報の公開日時を推定する．これによって最新の状況，動向を容易に把握できるようになり，結果的に効率的に表層解析をすることができるようになると考える (仮説①)．

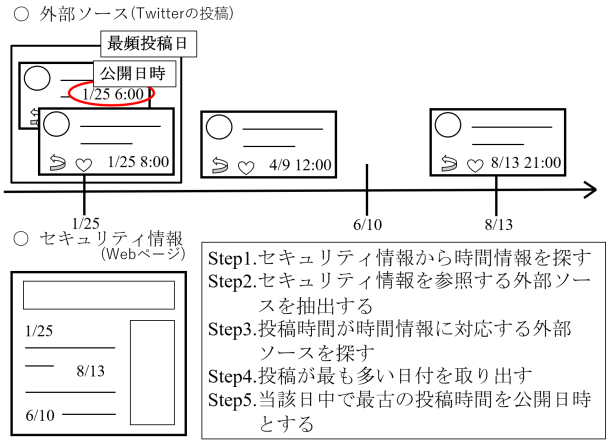

図 4　提案手法イメージ

4.3　不要判定

課題②に対しては，一般に，特定の情報を参照する場合は，取り上げられている内容が注目しているものであったり，よく整理されていて理解しやすい内容であったりと，何かしら役に立つ内容である場合が多いと考える．すなわち，外部ソースから参照されていない情報は，誰の役にも立たなかった不要な内容であると考えられる．そのため，セキュリティ情報を参照する外部ソースが発見できなかった場合に不要と判定する．外部ソースの中でも Twitter は，アクティブユーザが多く，ユーザごとの興味関心も多様であると考えられる．こういった中でも参照されないセキュリティ情報は，特に価値が低いと考えた．そこで本論文では，不要なセキュリティ情報を，Twitter でセキュリティ情報へのリンクを検索した結果が 0 件のものと定義する．この手法によって，事前に不要なセキュリティ情報を除くことができ，必要あるいは重要な情報のみを提供できるようになると考える (仮説②)．

5.　評価実験

5.1　基本情報

提案手法による効果の仮説の検証を目的として，(株)STNet で実際の業務で表層解析をしているセキュリティ管理者 9 名を被験者として，2020 年 12 月 15 日〜22 日の休日を除いた 6 日間で評価実験を実施した (表 2)．今回の評価実験では，システム全体としての評価を実施すると，評価結果がフロント部に左右される恐れがあると考えた．そこで，提案手法による結果に焦点を当てるために，システムのデバッグ画面を利用して実施した．デバッグ画面はシステムのすべてのデータベースと連携している。

表 2　実験環境

期間	2020 年 12 月 15 日〜2020 年 12 月 22 日
対象者	(株)STNet のセキュリティ管理者 9 名

5.2　評価実験の目的

仮説①について，提案手法によって実際に時系列で分析，最新情報の把握を支援できるか確認することを目的とする (目的①)．目的①は公開日時推定に対する実験をもって評価する．これらの結果は，実際に業務として作業している現場の人の感覚や印象にかかる部分が大きいと考えたため，アンケートによる定性的な評価を実施する．また仮説②について，提案手法によって不要なセキュリティ情報の判定が適切にできているか確認することを目的とする (目的②)．目的②は不要判定に対する実験をもって評価する．ここでは表層解析では，必要な情報を見落としてしまうと充分な分析ができなくなるため，提案手法によって必要な情報を誤って不要と判定していないかという点に焦点を当てて評価実験を実施する．

5.3　公開日時推定の評価手法

図 5 より実験の流れを説明する．図 5 中の画面はすべてデバッグ画面である．まず，検索画面の検索窓に検索語を 1 語入力し，全文検索の推定日時投稿順ボタンを押して検索する．今回の評価実験では，検索語として表 3 を例示した．検索すると検索結果画面で検索語に関連するセキュリティ情報が，提案手法によって推定した公開日時の順に並んで表示される．ここでの検索結果の上位 5 件に対して，記事 ID を選択して詳細画面で内容を確認する．確認対象を上位 5 件のみに絞ったのは，Google では検索結果の 6 番目の記事でクリック率が 5.1%と低く [6]，最近の動向や最新情報を把握することが効率的であったかを評価する上では，同等の件数で評価する必要があると考えたからである．確認した結果，検索意図に合うか，動向の把握に役立つ情報であるか評価して，良い情報であれば “◯”，そうでなければ “×” を表 4 の検索順位に対応する箇所に記録する．動向を追って調査する場合，日々継続的に検索する必要があるため，この作業を評価実験期間中毎日実施する．最終日に，表 5 のアンケートを実施する．推定した公開日時が実際の公開時間から大きく外れていると障害になると考えられるため，項目 1 で公開日時の推定結果の印象に関する質問を設けた．また，仮説①の検証のため，提案手法による提供順が検索語に関して動向を調査するという検索意図に合致していたか，項目 3 で検索結果の印象に関する質問を設けた．

5.4　不要判定の評価手法

実験の流れを図 5 を用いて説明する．検索画面のサイドバーの推定失敗実測順を押して，不要情報一覧の画面を確認する．不要情報一覧の画面には，日々収集したセキュリティ情報の中で，提案手法によって不要と判定したものが一覧で表示される．日々新たに一覧に追加されたセキュリ

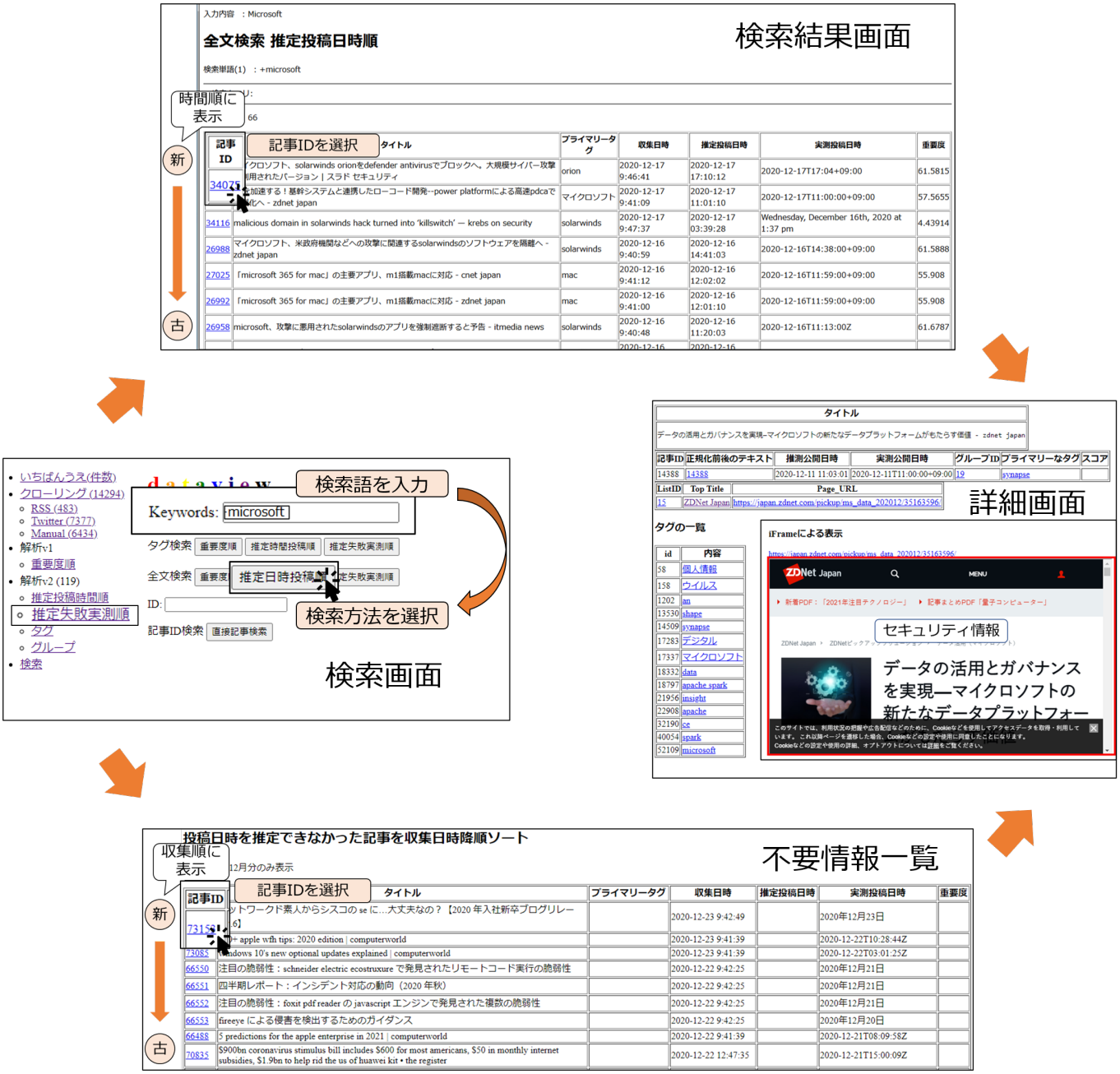

図 5　実験フロー

表 3　公開日時推定評価での検索語例

カテゴリ	検索語
企業	Microsoft，Cisco，Oracle，vmware，内閣サイバーセキュリティセンター
製品	Docker，Firefox，Chrome，Wordpress，Apache Tomcat
プログラミング言語	PHP，Java，Ruby，Perl，Go
インシデント	脆弱性，ランサムウェア，情報漏えい，不正アクセス，フィッシング
その他	bind，PoC，パッチ，エクスプロイト，クラウド

表 4　公開日時推定の評価シート

検索キーワード	ソート方法	表示順					備考
		1番目の記事	2番目の記事	3番目の記事	4番目の記事	5番目の記事	
脆弱性	推定日時	○	×	○	×	○	

表 5　アンケート項目

最終評価		良い	やや良い	やや改善余地あり	改善余地あり
アンケート回答日　2020年12月22日		80%以上	80%～50%	50%～20%	20%以下
1	推定公開日時の精度は満足できる結果でしたか？				
2	公開日時が推定されなかった記事は意図した結果でしたか？				
3	推定公開日時順での検索は意図した結果でしたか？				
自由記述欄					

表 6　不要判定の評価シート

記事ID	要因
41241	最近の動向を示した情報であるため重要

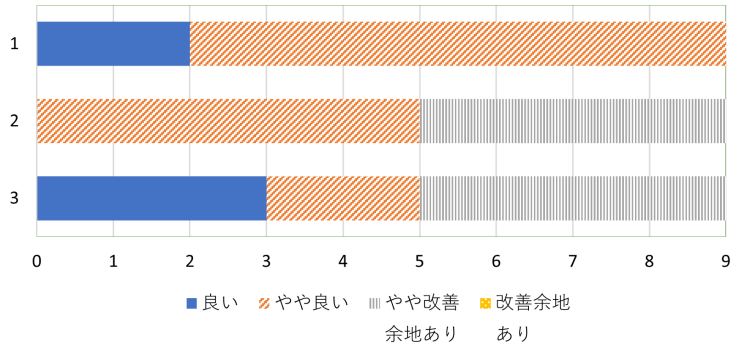

図 6　アンケート結果

表 7　自由記述回答 (抜粋)

記事が公開日順に表示されることで、運用者が意図・思考している操作が行えるようになった。
ある程度重要でない記事を省くことができているように思った。
可能であれば記事の更新日時を出せると更に良いと思います。
セキュリティ関連記事が下の方に押しやられていたこともあった。
推定に失敗していた記事に必要と思われる情報があったため、精度の向上に期待します。

ティ情報の記事 ID を押して，詳細画面にて内容を確認する．確認した中に，必要なセキュリティ情報があった場合，表 6 のように，セキュリティ情報の記事 ID と必要と判断した理由を併せて記録する．最終日に，表 5 のアンケートを実施する．仮説②に対して，表層解析に必要なセキュリティ情報を提供できているか，すなわち不要と判定されたセキュリティ情報がたしかに不要であるか，誤って不要と判定したセキュリティ情報が，表層解析において問題となるか調査するために，項目 2 で不要判定の結果に対しての質問を設けた．

6. 評価結果

アンケートの回答結果を図 6 に示す．また，自由記述の結果を表 7 に示す．仮説①に関連する，項目 1 に対して，2 人が「良い」，7 人が「やや良い」と回答し，項目 3 に対して，2 人が「良い」，3 人が「やや良い」，4 人が「やや改善余地あり」と回答した．自由記述では，「新しい情報だけが知れるため使いやすい」，「運用者が意図・思考している操作が行える」といった好意的な意見が多くあった．一方で，「記事の更新日時を出せると良い」，「セキュリティ関連記事が下に押しやられていることがある」といった意見も見られた．評価シートに関しては，仮説①に対する評価実験では，例として提示した検索語を 1 人あたり 5 あるいは 10 個担当し，それぞれの検索語に対して 2 人が評価した．公開日時推定の評価シートの結果は，1 番目の検索結果を 5 点，2 番目を 4 点，5 番目を 1 点というように，検索結果の上位からそれぞれ 5，4，3，2，1 点を持ち点として集計した．この中で必要と判断されたセキュリティ情報の合計点を，すべての検索結果に◯がつけられた場合の合計点で割って点数とした．そのため，得点範囲は 0～1 となる．例えば，2，4 番目に◯がつけられた場合は，6 点を 15 点で割った 0.4 が点数となる．点数を p，検索順位を i として，検索件数を n とすると計算式は (1) で表される．

$$p = \frac{\sum_{i=1}^{n}(n+1-i)\times c}{\sum_{i=1}^{n} i} \begin{cases} c=1 & (\text{回答が◯のとき}) \\ c=0 & (\text{それ以外}) \end{cases} \quad (1)$$

集計した結果をまとめたレーダーチャートを図 7 に示す．平均してインシデントのカテゴリが 0.82 点と最も高く，それ以外のカテゴリは 0.55〜0.6 点となった．また，日毎の評価点の変動は大きく，最低点と最高点の差はプログラミング言語のカテゴリの 0.08 点を除いて，0.25〜0.34 点の差があった．評価シートの備考欄には，検索語は含まれているがメインの内容ではないものや，セキュリティに関する内容ではないものが検索結果として出ているといった意見が見られた．例示した検索語以外では，1 名が“Apache Struts”，“CVE-2020-17049”，“SolarWinds”を検索しており，Apache Struts に関しては，備考欄にて評価者が実際に検索したときと差異のない順番で提供されていた，といった評価を得た．

仮説②に関連する，項目 2 に対しては， 5 人が「やや良い」， 4 人が「やや改善余地あり」と回答した．自由記述では，「ある程度需要のない記事を省くことができている」といった意見も見られたが，「不要と判定された記事の中に必要と思われる情報があった」，「検索結果にセキュリティに関係のない記事が混ざっている」といった意見が多く見られた．不要判定の評価シートを集計した結果，本手法によって不要と判断した 244 件のセキュリティ情報の内，33.2%の 81 件が評価者によって必要と判断された．必要と判断した要因としては，脆弱性に関する情報や防御策に関する情報，攻撃情報であることが多く挙げられた．

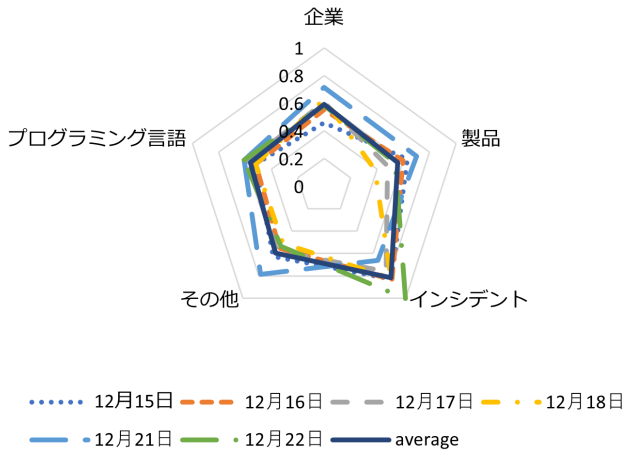

図 7　評価シート集計結果

7. 考察

7.1 公開日時推定への評価の考察

評価実験の結果，選択式のアンケートでは「やや良い」と回答した人を含めて，良いと感じた人が過半数を超えたこと，自由記述式のアンケートで好意的な意見が目立った．このことから，仮説①に対しては，推定した公開日時の順で記事を提供することで，困難であった最新情報及び動向の分析がしやすくなったとわかった．表層解析の課題の 1 つを軽減できたことから，本手法が表層解析の効率を高める一助となり得るといえると考える．Twitter の特徴である情報へのレスポンスの速さが，正確な公開日時を把握するという目的に合致していたと考えられる．評価シートに関しては全体的に，評価点の変動が大きくなっていたが，上位で提供されるセキュリティ情報に対しての得点の重みを大きく設定していたため，新しく収集されたセキュリティ情報の内容の影響を大きく受けた結果と考えられる．また，例示した検索語の中には企業名を始めとして関連する話題の多いものがあったため，1 語で検索したときに提供されるセキュリティ情報の内容に幅が出やすいことも原因として考えられる．

7.2 不要判定への評価の考察

Twitter からの参照の有無による不要な情報の判定では，33.2%の見落としがあったことがわかった．また，自由記述において，検索結果にセキュリティに関係のない記事が混ざっているという評価があったことから，不要な情報の提示 (誤検知) も一定数あったことがわかった．このことから，仮説②に対しては，Twitter のみに注目して参照されているかという観点だけでは充分ではないと考えられる．

見落としの要因としては，多くの多様なユーザがいる環境下であれば，重大な話題は情報が公開されてからすぐに参照するユーザがいると考え，公開されたと思われる日の内に参照されていなければ不要であると判断して，セキュリティ情報ごとに判定を一度しかしなかったことが考えられる．また，重要な内容ではあるものの，公開元のサイトが運営を開始したばかりなどの要因から知名度が低く，参照されなかったものがあると考えられる．

この改善案として，Twitter 以外の外部ソースから参照されている可能性もあるため，必要と判定されたセキュリティ情報の中で参照されていないか確認することで，見落としを減らすことができると考える．また，不要と判定した時点で公開されて間もなかったために見落としてしまう可能性が考えられるため，不要と判定されたセキュリティ情報に対しては複数回検証することで，判定精度を高めることができると考える．

誤検知の要因としては，Twitter の保有するユーザの興味が多様であるために，組織が必要とする内容以上の話題が参照されたことが考えられる．また，同じサイトから公開される記事の中でも必要な記事と不要な記事があると考えられるが，セキュリティ情報を公開しているサイトが

Twitter の公式アカウントを保有しているために，そのサイトで公開されるすべての記事が一度は参照されているために，余計に必要と判定されたセキュリティ情報があると考えられる．

この改善案として，セキュリティ情報を参照したユーザも含めた判定をすることが考えられる．今回の評価実験で表層解析を実施する人が必要とする話題を確認できたため，Twitter のユーザの中で，ユーザが参照したセキュリティ情報と，セキュリティ管理者が必要と判断したセキュリティ情報の一致率が高いユーザが参照していることを判定の軸として追加することで，判定精度の向上が期待できると考える．また，同様な理由から不要と評価されたセキュリティ情報の話題に傾向があれば，セキュリティ情報中の単語の出現傾向などを特徴ベクトルとして，学習することでも判定精度の向上が期待できると考える．

7.3 要望に関する考察

今回はセキュリティ情報の公開日時を推定することを目的とした提案及び実験を行ったが，自由記述の中で更新日時を出せるようになると更に良いという意見を得られた．IPA や JVN などの公的なサイトで公開されているセキュリティ情報であれば，更新があったセキュリティ情報のタイトルに更新と明示して，一覧に表示されるため，そのまま提供することでセキュリティ管理者は更新を認識することができる．一方で，個人ブログなどで公開されているセキュリティ情報に関しては，明示的に更新が表記されない場合がある．そのため，個人ブログを始めとした一部のサイトに関しては，更新日時を出す必要があると考える．セキュリティ情報は公開されたときと，更新されたときは比較的注目が集まると考えられるため，セキュリティ情報を参照する Twitter の分布が極大値となった日を取得することで，更新日時を出すことも実現できると考える．また，更新日時が要求される場面では，対象のセキュリティ情報は一度システムによって収集されていることが前提として考えられる．そのため，一度収集したセキュリティ情報は，毎日収集して差分を調べることでも更新を検出することができると考える．

また，抄訳したセキュリティ情報に対して，原著のセキュリティ情報が公開された日時を出せるようにしてほしいという意見も得られた．一般に抄訳したセキュリティ情報は，原著の情報を本文の中で明記している．そのため，これに関しては抄訳後のセキュリティ情報に対して原著の情報の公開日時を付与するのではなく，原著の情報と紐付けることで，充分に提供できると考える．

8. おわりに

表層解析を効率化するため，セキュリティ情報の公開日時の推定手法と不要判定手法を提案し，評価を実施した．評価実験の結果，公開日時の推定の結果に従って時系列でセキュリティ情報を提供することの有用性を確認できた．

その一方で，必要なセキュリティ情報の内 33.2%を不要と判定するなど，提供するセキュリティ情報そのものの判定に課題を残したため，今後，改良を進めさらなる評価実験を実施する．評価実験の中で新たな要望も得られたため，それに対しても検討を深める．また，多くの研究で扱われているのは JVN などによって構造化された脆弱性情報であるが [7][8]，我々が開発を進めている CSICOS では，より広範な情報源からセキュリティ情報を収集しているため，関連情報の判定精度を高めることで，今回確認できた最新情報や動向の把握の支援と併せて，最新の関連情報のないセキュリティ情報をインシデントの予兆として評価するといったことができないか検討する．

謝辞

本研究は株式会社 STNet との共同研究として取り組んでおり，今日に至るまで多大な御協力を賜り，本論文という 1 つの区切りに辿り着きました．この場でを借りて株式会社 STNet の皆様に感謝致します．

参考文献

[1] ScanNetSecurity: 95 日、検知までの期間が昨年から 10 日延びる 〜インシデントレスポンスから得られた CrowdStrike の知見, https://scan.netsecurity.ne.jp/article/2020/06/17/44216.html, (2021.05.01)

[2] JPCERT/CC: CSIRT ガイド, https://www.jpcert.or.jp/csirt_material/files/guide_ver1.0_20151126.pdf, (2021.05.01)

[3] NRI: セキュリティが経営戦略になる時代　企業における情報セキュリティ実態調査 2019, https://www.nri.com/-/media/Corporate/jp/Files/PDF/knowledge/report/cc/mediaforum/2019/forum278.pdf. (2021.05.01).

[4] 竹原一駿，檜垣龍徳ら，“AI によるセキュリティ情報解析支援システムの提案”，マルチメディア，分散，協調とモバイルシンポジウム 2020 論文集, pp.1606–1615, (2020)

[5] 檜垣龍徳ら, “表層解析の効率化に向けた情報鮮度判別方式の提案”, 令和 2 年度電気・電子・情報関係学会四国支部連合大会講演論文集, pp.18-5–18-5, (2020)

[6] SearchEngineJournal: Over 25% of People Click the First Google Search Result, https://www.searchenginejournal.com/google-first-page-clicks/374516/, (2021.05.02)

[7] 楠目幹，喜田弘司，最所圭三，“脆弱性情報を利用したゼロデイ攻撃対策システムにおける構成情報収集機能の実装及び脆弱性評価機能の設計”，電子情報通信学会技術研究報告, Vol.119, No.140, pp.1–6 (2019).

[8] 田島浩一，岸場清悟，近堂徹，渡邉英伸，岩田則和，西村浩二，相原玲二，“脆弱性診断と脆弱性情報公開サイトを用いた脆弱性更新通知機能の試作”，マルチメディア，分散協調とモバイルシンポジウム 2017 論文集, Vol.2017, pp.1661–1664 (2017).