機械学習を用いた Cyber Threat Intelligence の構造化と横断的分析

藤井 翔太[1,2]　川口 信隆[1]　重本 倫宏[1]　山内 利宏[3]

概要：サイバー攻撃の増加・巧妙化に伴い，CTI（Cyber Threat Intelligence）を収集・分析し，最新の脅威情報へ追従することがより重要となっている．一方で多くの CTI は自然言語で記述されており，分析には多くのコストを要する．加えて，様々な組織が別々に情報を発行しており，横断的な分析が困難である．そこで，我々は CTI を自動的に共通フォーマットで構造化することにより，分析を支援する CyNER の研究を進めている [7]．これにより，効率化や横断的な分析の実現が期待される．本稿では，CTI の構造化における固有表現抽出の精度向上手法について述べる．評価では，サイバーセキュリティドメインのコーパスを用いて学習したモデルにより，固有表現抽出の F 値を最大 2.6 ポイント向上できることを示した．また，CTI を横断的に分析し，既存のレピュテーションサイトには含まれない IOC を抽出可能なこと，97%以上の IOC が単一の情報源にのみ含まれること，および長期的かつ複数の攻撃グループにまたがって悪用されている IOC を自動で抽出できることを示し，CyNER が CTI 分析の効率化に寄与する見込みを得た．

キーワード：Cyber Threat Intelligence，情報抽出，固有表現抽出，関係抽出，STIX

Machine Learning-based Cyber Threat Intelligence Construction and Crossover Analysis

Shota Fujii[1,2]　Nobutaka Kawaguchi[1]
Tomohiro Shigemoto[1]　Toshihiro Yamauchi[3]

Abstract: Cybersecurity threats have been increasing and sophisticated. In such circumstances, to keep up with the latest threat information with CTI is important. However, most of the CTI is written in natural language, which makes the analysis costly. To solve this problem, we have been studying CyNER, the method of supporting analysis by automatically structuring CTI. In this paper, we describe a method for improving the accuracy of named entity recognition in CTI structuring. Specifically, we construct a domain-specific language model to improve the recognition accuracy. In the evaluation, we showed the F-measure of named entity recognition can be improved by up to 2.6 points. We also showed CyNER can extract IOCs that are not included in the existing reputation services. We also found that more than 97% of IOCs are contained in a single source, and that some IOCs have been utilized for a long time by multiple attack groups.

Keywords: Cyber Threat Intelligence, Information Extraction, Named Entity Recognition, Relation Extraction, STIX

1. はじめに

年々サイバー攻撃が増加・高度化しており，サイバー脅威インテリジェンス（CTI: Cyber Threat Intelligence）を収集し，最新の脅威情報に追従することがますます重要となっている．CTI には，新規の脆弱性・マルウェアの情報，攻撃者の手口，およびそれらに対する対策手法等が記載されている．また，攻撃を検知する指標として IOC（Indicator Of Compromise）が含まれることも多い．IOC は，例えば不審サイトの IP アドレスや URL，マルウェアのハッシュ値等から成る．こうした情報をファイアウォールや侵入検知システムの検知ルールとして活用することにより，攻撃の検知が可能となる．このように，CTI に含まれるマルウェア情報，脆弱性情報，および IOC 等を適切に抽出して活用することにより，検知ルールの構築や攻撃傾向の分析が可能となる．

一方で，CTI は，まずブログ，ベンダレポート，および公的機関情報等の形で，非構造化データとして配信されることが多く，それらの情報が公開されてから構造化されるまでの間にはタイムラグが存在し，1 ヶ月以上を要する場合もある [1]．このため，最新の脅威情報に追従するには，非構造化データを分析・活用する必要がある．しかし，例えばセキュリティブログに限定しても月 60,000 件以上 [2] と日々大量の CTI が発行されることが知られている．このすべてを人手で分析することは現実的ではなく，網羅性を確保するのが困難である．また，多くの CTI は自然言語で記述されていることから，単純に機械処理を実施することも容易ではない．こうした課題を解決するには，自然言語で記述された CTI を機械処理可能な形に構造化し，効率的な分析の支援を行うことが重要である．

こうした背景を受けて，辞書やオントロジを作成することによって，非構造化データの構造化を試みる研究がある [3-6]．ただし，セキュリティ分野では，新たなマルウェアの出現や脆弱性の発見，コードネームの付与等により，新語が生まれやすいことから，継続的な辞書やオントロジのメンテナンスが容易ではない．また，URL や IP アドレス等の IOC は形式が定まっているため，正規表現を用いることで抽出可能ではあるものの，どのようなマルウェアや攻

1 株式会社日立製作所
Hitachi, Ltd.
2 岡山大学 大学院自然科学研究科
Graduate School of Natural Science and Technology, Okayama University
3 岡山大学 学術研究院自然科学学域
Graduate School of Natural Science and Technology, Okayama University

撃者に用いられるものであるかといった文脈情報が欠落してしまうため，それだけでは分析や検知ルールとしての適用可否の判断に活用することが困難である．

これらの課題を緩和するべく，固有表現抽出と関係抽出を用いた CTI の構造化手法である CyNER の研究を進めている [7]．CyNER は，マルウェア名や脆弱性名のように，サイバーセキュリティの文脈で着目すべき固有表現を抽出することにより，分析の効率化を狙うとともに，固有表現間の関係を抽出することにより，文脈情報を維持した形で構造化を図るものである．文献 [7] においては，CyNER の基本方式を示し，一定の精度で CTI を構造化できることとその処理時間が実業務の範囲内であることを実証した．しかし，固有表現抽出の精度に向上の余地がある点と CyNER を用いた CTI の分析を実施できていない点が課題としてあった．

そこで，本稿では，提案手法における固有表現抽出の精度向上手法について述べる．具体的には，サイバーセキュリティ分野のコーパスを用いて学習を行い，ドメインに特化した言語モデルを構築することで認識精度の向上を図る手法を提案する．また，CyNER を用いて実際の CTI を構造化し，横断的に分析することにより，活用可能性を示す．

本稿の貢献は以下の通りである：

- CTI の構造化に際して，ドメインコーパスを用いて学習した言語モデルを活用することにより，汎用の言語モデルを用いた場合よりも固有表現抽出の F 値を最大 2.6 ポイント向上できることを示した．
- CyNER を用いて 40 の情報源から得た 52,292 件の CTI を構造化するとともに，270,047 件の IOC を抽出し，横断的な分析を実施した．この分析の中で以下の事実を明らかにし，CyNER の活用可能性を示した．
  - CyNER で抽出した IOC の網羅性を既存のレピュテーションサービスと比較し，既存サービスには含まれていない IOC を抽出できることを示した．
  - 抽出した IOC の 97%以上にあたる 262,174 件の IOC が一つの情報源にのみ含まれていたことを明らかにした．
  - 19,010 件の IOC は継続的に報告されており，1 年以上の間複数の攻撃グループにまたがって悪用されているものもあることを明らかにした．

2. CTI の構造化に係る背景

2.1 サイバーセキュリティにおける構造化フォーマット

サイバーセキュリティ分野において，セキュリティ情報の機械可読化や共通フォーマットでの情報交換を目的に，様々な構造化フォーマットが策定されている．例えば，STIX [8]（Structured Threat Information eXpression）が挙げられる．STIX は，以下に示す 2 つの情報から構成される：

(1) SDO（STIX Domain Object）：サイバーセキュリティの文脈におけるドメイン語のオブジェクト

(2) SRO（STIX Relationship Object）：SDO 間の関係性

STIX は，上記の情報を組み合わせることにより，様々なセキュリティ情報を構造化情報として記述できるものとなっている．他にも，STIX と同様に広範なセキュリティ情報を取り扱うことのできる MISP [9] や IOC に特化した OpenIOC [10] 等がある．

2.2 課題

前述のように，サイバーセキュリティに関する情報を構造化データとして取り扱うことのできるフォーマットは幾つか存在し，これらを活用することにより効率的に情報を処理することが期待できる．

一方で，CTI は，まずブログ，ベンダレポート，および公的機関情報等の形で非構造化データとして配信されることが多い．このため共通フォーマットの利点を享受しつつ最新の情報を取り扱うためには，非構造化データを構造化する必要がある．しかし，日々大量に公開される情報を全て人手で構造化するのは現実的ではないという課題がある．

3. 提案手法

3.1 残存課題

我々は，2.2 節で述べた課題を解決するために，文献 [7] において，CTI を自動的に共通フォーマットで構造化し，効率的な分析の支援を行うことを目的した CyNER を提案した．CyNER は，マルウェア名や脆弱性名のように，サイバーセキュリティの文脈で着目すべき固有表現を抽出することにより，分析の効率化を狙うとともに，固有表現間の関係を抽出することにより，文脈情報を維持した形で構造化を図るものである．文献 [7] においては，CyNER の基本方式を示し，一定の精度で CTI を構造化できることとその処理時間が実業務の範囲内であることを実証した．しかし，以下の 2 点が課題として残存している．

（**課題1**）固有表現抽出の精度に向上の余地がある

文献 [7] では，固有表現抽出に際して，BERT [11] に基づく事前学習済みの言語モデルを活用した．BERT は，深層学習モデルの transformers をベースとした機械学習手法であり，様々なタスクにおいて高い性能を達成しているものである．一方で，公開されている事前学習モデルは，Wikipedia に含まれる文章のような一般的なコーパスを用いて学習されており，専門的なドメインに対しては性能が低下することが知られている [12]. サイバーセキュリティのドメインについても同様の性能低下が起きている可能性があり，精度向上について検討する必要がある．

（**課題2**）CTI 分析における有用性の評価が未実施である

文献 [7] では，CTI の構造化に係る精度や処理時間の評価までは実施したものの，CyNER を用いた CTI の分析にまでは至っていない．本研究の目的として，

CTI 分析の効率化を掲げており，CyNER を活用して実際の CTI を分析し，効率化に資する結果が得られるか検証する必要がある．

以降では，上述の課題への対応について述べる．なお，（**課題 1**）への対応については，本章における以降の節で，（**課題 2**）への対応については，評価の章（4 章）で述べる．

3.2 事前学習による固有表現抽出の精度向上

本節では，（**課題 1**）への対応としての固有表現抽出精度の向上について述べる．

前述のように，BERT をはじめとした事前学習済みの言語モデルは広く公開されており，それらを用いることで固有表現抽出が可能である．一方で，専門分野については分野固有のコーパスで事前学習を行うことで，当該モデルをベースにした各種タスクの精度が向上することが知られている [13]．そこで，サイバーセキュリティ分野のドメインコーパスを用いて独自に事前学習モデルを構築することにより，固有表現抽出の精度向上を図る．

サイバーセキュリティドメインの事前学習モデルを構築するに際して，まず，言語モデルを構築するための学習データとすべく，CTI を公開している Web ページをクローリングし，ドメインコーパスの候補として収集する．次に，収集した CTI から学習のノイズとなる不要な情報を削除し，学習用の文章を抽出する．具体的には，本文部分を抽出するために，HTML タグや JavaScript 等の不要な情報を除去する．また，本文部分においても，見出しや箇条書きのような文章となっておらず，学習に不要なものも残存している．そこで，文献 [14] を参考に，以下の処理を行うことによって不要な情報を除去し，ドメインコーパスとする．

- 5 文未満のページ
- 3 語未満の行
- snort 等のシグネチャと思われる行（“{” や “$” で始まる行）

最後に，ここまでに構築したドメインコーパスを用いて，事前学習を実施し，言語モデルを構築する．

上記の方法により，CTI の構造化に際しての固有表現抽出の精度向上を図る．

3.3 全体像

文献 [7] の手法に，前述の事前学習を加えた CyNER の全体像を**図 1** に示す．処理の流れは以下の通りである．

(1) 情報収集

CTI を発信しているサイトをクローリングし，記事を収集する．

(2) 前処理

後段の処理のために，本文の抽出や IOC のリファングといった前処理を行う．

(3) 事前学習

前処理済みのテキストを蓄積しておき，例えば一定期間ごとに 3.2 節の手法で事前学習モデルを構築する．

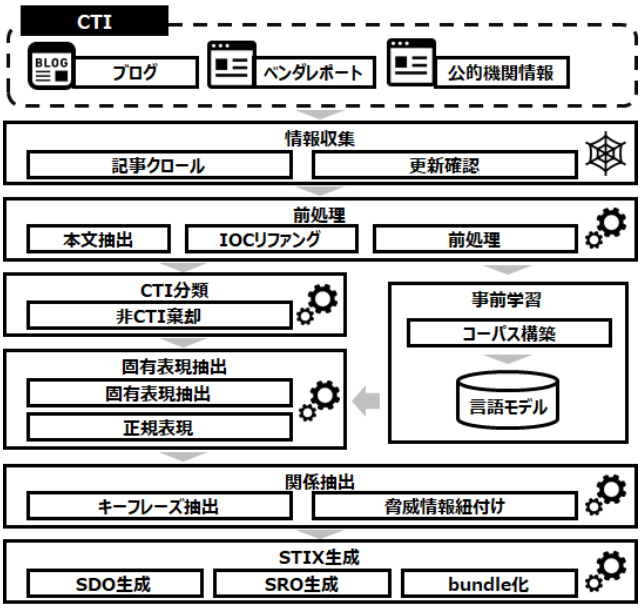


図 1　提案手法の全体像

表 1　提案手法における固有表現

STIX 項目	抽出項目	説明	例	抽出方法
Attack Pattern	name	攻撃パターン名	Spear Phishing	固有表現抽出
Campaign	name	キャンペーン名	Operation Aurora	固有表現抽出
Grouping	name	脅威アクタ名	APT10	固有表現抽出
Identity	name	名前	Hitachi, Ltd.	固有表現抽出
Indicator	pattern	IOC	URL，ハッシュ値，IP アドレス	正規表現
Malware	labels	マルウェア種別	ransomware	固有表現抽出
	name	マルウェア名	WannaCry	固有表現抽出
Tool	name	ツール名	metasploit	固有表現抽出
Vulnerability	name	脆弱性名	CVE-2014-0160 HeartBleed	正規表現 固有表現抽出

(4) CTI 分類

CTI を提供するブログや公式ページの中には，製品やセミナの紹介記事を含むものもある．それらは CTI ではないため，CTI か否かを判定する 2 値分類器によって CTI のみを抽出し，そうでないものを棄却する．

(5) 固有表現抽出

文献 [7] において STIX の SDO の仕様に沿って定義した項目（**表 1**）を固有表現抽出によって抽出する．この際，先に構築した言語モデルを利用することで，精度向上を図る．

(6) 関係抽出

同じく文献 [7] で STIX の SRO の仕様に沿って定義した SDO 間の関係（**表 2**）を抽出する．

(7) STIX 生成

最後に，抽出した SDO と SRO を STIX のフォーマットとして整形し，出力する．

以上の処理によって，より高い精度での CTI の構造化を実現し，CTI 分析の効率化を図る．

表 2　関係抽出のルール

主体	客体	関係性
indicator － ハッシュ値 － ファイル名	attack_pattern campaign_name malware_name threat_actor_name	indicates
malware_name	indicator － URL － IP アドレス	communicates-with

4. 評価

4.1 評価項目

先述の設計に沿って提案手法のプロトタイプを実装し，以下の評価を実施した．まず，（**課題 1**）に係る内容として以下の評価を実施した．

（**評価 1**）固有表現抽出精度

CyNER は，（**課題 1**）への対処として，ドメインコーパスを用いて事前学習を行い，固有表現抽出の精度向上を図る．そこで，事前学習言語モデルによって精度が向上するか否かを適合率，再現率，および F 値の軸から検証する．

また，（**課題 2**）に係る内容として，CyNER を用いて CTI を分析し，その活用可能性を検証するべく以下の評価を実施した．

（**評価 2**）IOC の網羅性検証

CyNER によって CTI を自動的に構造化することにより，人手で構造化するよりも効率的かつ網羅的に情報を活用できることが期待される．そこで，CyNER で CTI を構造化し，そこから抽出した IOC や IOC に紐づく情報の網羅性をデファクトのサービスとの比較を通して評価する．

（**評価 3**）情報源の関係性の活用可能性検証

CyNER は複数の情報源からなる CTI 群を統一のフォーマットで構造化し，一元的に取り扱うことができる．そこで，CyNER で抽出した IOC と情報源間の関係性を洗い出し，活用可能性を検討する．

（**評価 4**）時系列情報の活用可能性検証

CyNER を用いることにより，過去から現在の CTI や IOC の時系列での情報を一元的に取り扱うことができる．そこで，時系列の観点で CTI や IOC を分析するとともに，その活用可能性について検討する．

4.2 データセット

各評価を実施するにあたり，既存研究や実務者へのヒアリングを基に CTI を発信している 40 のサイトを選定した．

また，各サイト用にクローラを実装し，2001 年 6 月～2020 年 12 月の間に公開された 75,652 件の CTI 候補を収集した後，以下に示すデータセットを評価用に構築した．

表 3　抽出された IOC の数と内訳

IOC 種別	IOC 数
ハッシュ値	50,323
URL	184,349
IP アドレス	35,375
合計	270,047

表 4　評価環境

CPU	Intel Xeon E5-2698 v4 (2.2 GHz, 20 cores)
メモリ	256 GB 2133 MHz DDR4 LRDIMM
GPU	Tesla V100（VRAM 32 GB）×4 (VRAM 128 GB)
OS	Ubuntu 18.04

・言語モデル学習用データセット

収集した CTI に 3 章で述べた前処理を施し，言語モデル学習用のドメインコーパスとした．なお，本データセットは，約 3,000,000 行，約 320MB となった．

・固有表現抽出用データセット

収集した CTI をランダムにピックアップし，固有表現に関してアノテーションした CTI を 100 件分用意した．なお，本データセットは，13,479 文・193,027 単語から成り，のべ 4,562 の固有表現を含む．

以降の評価は上記のデータセットを用いて実施したものである．なお，（**評価 2**）～（**評価 4**）における分析対象は，上述の 75,652 件のうち，CTI 分類器によって CTI と判定された 52,292 件である．また，抽出された IOC のユニーク数は 270,047 であり，その内訳は**表 3** の通りである．なお，Alexa Top10,000 に含まれる URL や RFC で定義されているプライベート IP アドレスは，偽陽性の可能性が高いため，除外している．また，評価は**表 4** に示す環境で行った．

4.3 評価結果

4.3.1 評価 1：固有表現抽出精度

本評価では，70%にあたる 70 記事分を学習に，残りの 30 記事分を検証用に用いた．この際，学習用データを更に 70%の訓練用セットと 30%検証用セットに分割して学習を実施した．また，実装には，言語モデルを取り扱うことのできるライブラリである Hugging Face Transformers [15] を活用し，モデルには同ライブラリで利用可能な事前学習モデルのうち，代表的な BERT（BERT-BASE, BERT-LARGE）に加え，その後発である ALBERT [16]（ALBERT-BASE, ALBERT-LEARGE）と RoBERTa [17]（RoBERTa-BASE, RoBERTa -LEARGE）の合計 6 種類を利用した．また，学習パラメータはデフォルト値，エポック数は 3 とした．上記の設定で，モデルごとにドメインコーパスを用いた場合と用いない場合で精度を評価した．実験結果を**図 2** に示す．プレフィックスに「CTI-」を記載したもの（橙色）がドメ

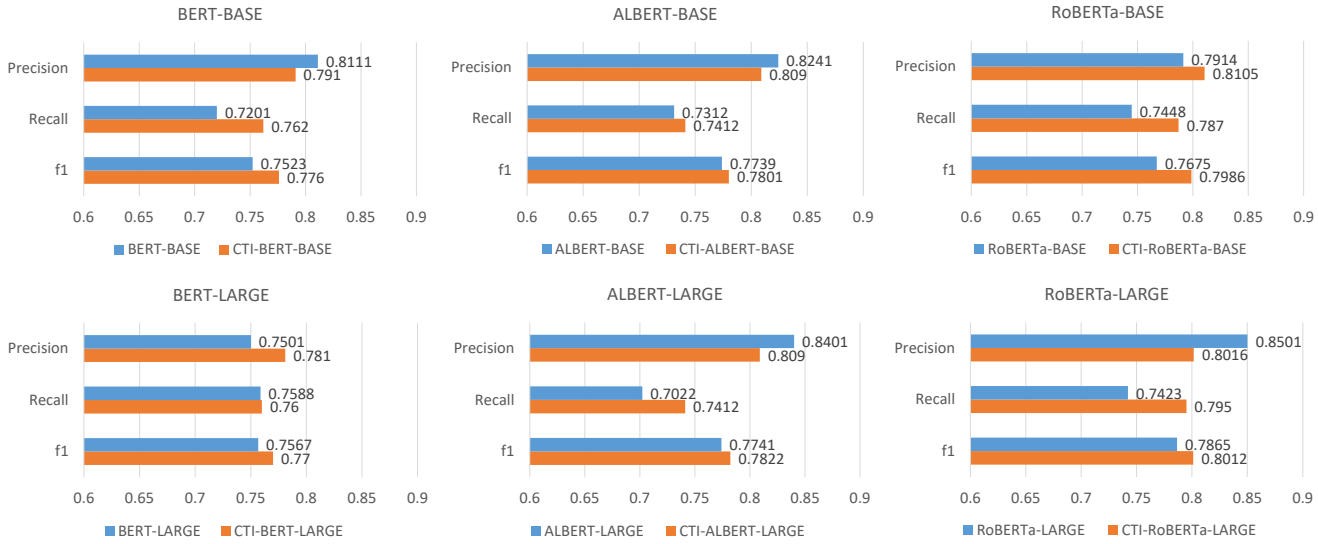


図 2　各モデルの固有表現抽出の精度

インコーパスを用いたモデル，記載していないもの（青色）がドメインコーパスを用いていないモデルの結果である．実験結果から何れのモデルにおいても，F 値が向上しており，最大で BERT-BASE において 2.4 ポイントが向上していることが分かる．また，RoBERTa-LARGE が F 値 0.8012 と全モデルの中で最大の精度となった．

この結果より，サイバーセキュリティの分野においても，ドメインコーパスを用いて事前学習を行うことにより，固有表現抽出の精度を向上できることが確認できた．なお，以降の評価は，RoBERTa-BERT を用いた場合の CyNER を用いて実施している．

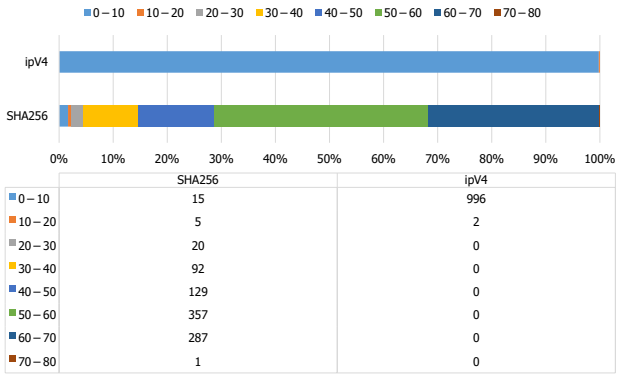
4.3.2　評価 2：IOC の網羅性検証

本評価では，CyNER から抽出した IOC が代表的な既存サービスである VirusTotal や AlienVault OTX に含まれるか否かを確認し，網羅性の比較を行った．この際，IOC としては一つ以上のマルウェアに紐づくものをランダムで選出した．選出した IOC の性質はマルウェアのハッシュ値（SHA256）と通信先（IPv4 アドレス）であり，それぞれ 1,000 件，合計 2,000 件選出し，比較に利用した．まず，カバー率に係る評価結果を**表 5** に示す．本評価では，前述の通り CyNER で抽出したマルウェアファミリと紐づく SHA256 と IPv4 アドレス 1,000 件ずつ，合計 2,000 件を基にカバー率の比較を行った．

まず OTX は SHA256 が 2.5%，IPv4 も 19.5%とカバー率が小さいことが分かる．OTX は人手の介在や専門家の登録に依る部分が比較的大きいことが原因であると考えられる．一方で，VirusTotal に関しては，SHA256 が 90.6%，IPv4 に関しては 99.8%と CyNER で抽出した IOC の多くが含まれるという結果になった．VirusTotal は OTX と比較して投入者が多いことから，今回の CyNER が情報源としたような公開されている CTI に記載されている IOC は，その多くが投入済みであるためだと推察される．ただし，CyNER で抽出した IOC のうち，どちらのサービスにも含まれていない IOC もあり，そうした IOC を自動かつマルウェアファミリと紐づけて文脈情報を有した状態で構造化できる点で有用であると言える．

表 5　各プラットフォームにおける IOC のカバー率

IOC 種別	手法	合計	含有数	割合
SHA256	CyNER	1,000	1,000	100%
	VirusTotal[a]	1,000	906	90.6%
	AlienVault OTX[b]	1,000	25	2.5%
IPv4	CyNER	1,000	1,000	100%
	VirusTotal	1,000	998	99.8%
	AlienVault OTX	1,000	195	19.5%



	SHA256	ipV4
0－10	15	996
10－20	5	2
20－30	20	0
30－40	92	0
40－50	129	0
50－60	357	0
60－70	287	0
70－80	1	0

図 3　VT に存在する IOC の AV エンジンによる検知数

次に，情報量の観点から比較を実施した．まず，OTX に関しては，前述の通りカバー率にやや欠ける点はあるものの，人手によってタグ付け等が可能であることから，CyNER と同等，あるいはそれ以上の情報を有する場合が多い．また，VirusTotal は，数十の AV 製品や URL スキャナによって，対象を検査することができる．このスキャン結

a VirusTotal は，Rotarua Limited の商標または登録商標である．
b AlienVault は，エイリアン ボールト，インコーポレーテッドの商標または登録商標である．

表 6 AVCLASS によってファミリ名を推定できた検体の数

IOC 種別	合計	推定数	割合
SHA256	906	362	39.95%

果を**図 3** に示す．ファイル（SHA256）に関しては，大部分が 30 以上の AV 製品で検知されており，この結果を活用することで対象がマルウェアか否かを高精度で推定することができる．

一方で，通信先（IPv4）に関しては，ほぼ全てが 10 以下のエンジンでしか検知されておらず，この結果のみを用いて対象が悪性か否かを推定することは容易ではないと言える．これは，通信先に関してはクローキングや well-known でないポートの利用等により，シンプルなスキャンでは悪性か否かの判断が容易ではないことが間接的な原因であると推察される．

さらに，VirusTotal のスキャン結果を基にマルウェアファミリが推定できるかを AVCLASS [18] を用いて検証した結果を**表 6** に示す．VirusTotal に含まれていた 906 件の検体のうち，約 40%は CyNER と同様にマルウェアファミリを推定できたものの，残りの約 60%は推定できなかった．これは，検体や AV エンジンによっては Generic.Trojan のように汎用的な名前で検知されるため，マルウェアファミリに紐づく情報が欠如していることが原因と言える．また，後段でマルウェア本体をダウンロードするような検体は，一連の攻撃の中では特定のマルウェアファミリに関連するものの，単体としては「Downloader」としてしか検知されないことがあるため，同様にマルウェアファミリに繋がる情報が得られなかった．これに対し CyNER は，IOC を CTI 中のマルウェアファミリ等と紐づけることが可能であるため，SHA256，IPv4 を問わず悪性か否かの判断可能性が高いと言える．また，検体の性質に依らず，CTI 中で言及されている攻撃と紐づけるため，Downloader であっても後段に降ってくるマルウェアファミリに紐づけることが可能であり，この点でも優位性があると言える．

以上の様に，CyNER によって複数の情報源から収集した CTI を一元的に構造化することにより，デファクトのサービスを単体で利用するよりも IOC や IOC に紐づく情報を網羅できた．

4.3.3 評価 3：情報源の関係性の活用可能性検証

ここでは，IOC と情報源間の関係性を述べる．先述の通り，今回は 40 のサイトを情報源としているが，IOC 毎にどの情報源に含まれるかを洗い出し，当該 IOC を含む情報源の数をまとめた．この結果を図 4 に示す．

まず，今回の調査範囲では，全体の 97%以上にあたる 262,174 件の IOC が一つの情報源にのみ含まれていた．この結果は，特定の情報源にしか含まれない IOC が多数存在していることを意味しており，前節の評価項目でもある網羅性を向上するためには，情報源を増やすことが望ましいことを示唆している．他方で，2 つ以上の情報源に含まれる IOC も 7,678 件存在していた．例えば，過去に猛威を振るった WannaCry のハッシュ値[c]は 4 つの情報源に，BadRabbit のハッシュ値[d]は 5 つの情報源に含まれていた．このように複数の情報源に含まれている IOC は，より脅威度の高いものである可能性が高いと言える．

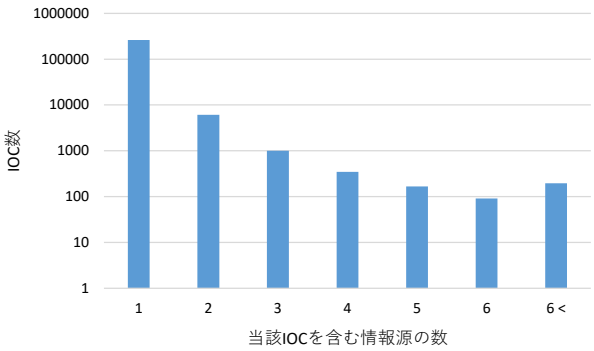


図 4　IOC を含む情報源の数

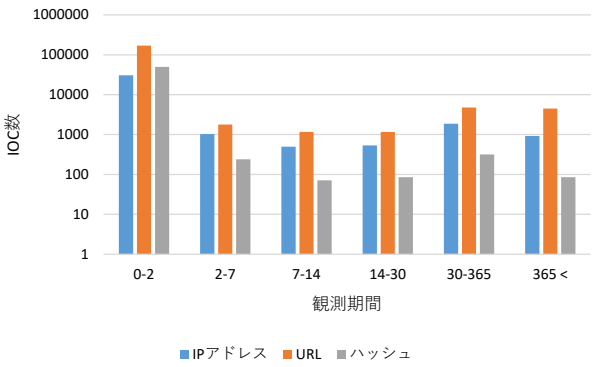


図 5　IOC の種別ごとの観測期間

以上より，IOC を含む情報源の数を算出し，その数が多いものを抽出することにより，より危険性の高い IOC を自動的に抽出できる可能性があると言える．

4.3.4 評価 4：時系列情報の活用可能性検証

ここでは，IOC の時系列情報としての活用可能性を検証する．IOC 毎に CTI で初めて報告された日と最後に報告された日を記録し，その差分の日数を観測期間として定義し，この期間の調査を実施した．なお，1 度のみ報告されているものや複数報告されているものの報告日が同日の場合のものは観測期間を 0 としている．調査結果を IOC の種別（IP アドレス，URL，およびハッシュ）毎に分け，観測期間をその長さ毎にプロットした結果を**図 5** に示す．

図 5 より，何れの IOC 種別においても，観測期間が 0-2 日の間に収まっているものが多いことが分かる．特にハッシュ値は，全体の 98%以上にあたる 49,527 件がこの範囲に収まっており，本傾向が顕著に表れている．これは，ハッシュ値ベースのマルウェア検知は比較的容易であることや亜種によってハッシュ値が変更できることから，同一ハッシュのマルウェアは，長期的に利用され難い傾向があるこ

c f8812f1deb8001f3b7672b6fc85640ecb123bc2304b563728e6235ccbe782d85
d 8ebc97e05c8e1073bda2efb6f4d00ad7e789260afa2c276f0c72740b838a0a93

とに起因すると推察される．URL や IP アドレスに関しても，観測期間の短いものが多く，これらは攻撃毎に使い捨てられていると考えられる．

一方で，大多数ではないものの，19,010 件は 3 日以上の観測期間を有しており，特に URL や IP アドレスに関しては，長期的に報告されているものが散見された．例えば，59[.]188[.]0[.]197 は，観測期間が 792 日と比較的長期にわたって観測されていた IP アドレスである．同 IP アドレスは, 2014 年に Temper Panda グループのスピアフィッシングにおける C2 サーバとして報告されている．その後，2015 年に同じく Temper Panda グループの攻撃に用いられたという報告があり，同グループにおいて継続的に悪用されていた攻撃インフラの一つであると推察される．また，同 IP アドレスは，2015 年に APT16 グループの攻撃に用いられたという報告もされており，複数の攻撃グループで攻撃インフラを共有している可能性もあるといえる．

以上より，長期的に観測されている IOC を抽出することによっても，より危険性の高い IOC を自動的に抽出できる可能性があると言える．

5. 議論

5.1 活用可能性

評価の章で述べたように，特定の情報源にのみ含まれている IOC が多いことが分かった．そこで, CyNER により，それらを一元的にまとめ，ブロックリスト作成の自動化や手動のインシデント調査等に活用する用途が挙げられる．また，リソースの制限からより脅威度の高いもののみをブロックリストに追加したいという要求は現実的に存在すると考えられる．このような要求に対して，CyNER を用いることにより，特定の脅威に関連付けられたものを選択することや長期的，あるいは複数の情報源にまたがって報告されているものを選択するといった活用方法が挙げられる．さらに，CyNER を用いることで長期的な IOC を時系列に沿って提示することや複数の情報源にまたがって報告されている IOC をまとめて提示することができるため，先述の要求に対する業務の効率化が期待できる．

一方で，実用に際して考慮する必要のある点がいくつか存在する. 今回の精度評価では統一的に F 値を重用したが，タスクによって重要視する精度が異なる．例えば，人手でインシデントレスポンスする際の情報源としては，偽陽性を容認してでも網羅性が重要視される．他方で，ブロックリストを作成する際は，正常通信を過検知するのは望ましくないため，真陽性が重要視される．このように，タスクに応じてどの指標を重要視するかを選択する必要がある．また，CTI には潜在的に偽陽性が含まれることが示唆 [19] されており，偽陽性に対して機微な用途に際しては，別途フィルタを導入する等の対処が望ましい場合もある. また，ドメインコーパスを用いて固有表現抽出の精度を向上したが，事前学習が必要になるため，その分処理時間が増大する．今回主に活用した RoBERTa-LARGE は，3 エポックの学習に約 17 時間を要した. 場合によっては，より軽量なモデルの活用等も検討する必要がある．

5.2 制限事項

URL や IP アドレスが未知の手法でデファングされていた場合，リファングすることができず，正規表現で抽出できない可能性がある．ただし，多くの場合 CTI 提供サイトの単位ではデファングの方式は統一されているため，この場合，サイトごとに一度リファングルールを整備すれば対応可能である．

また，評価の章で述べたように，単一の情報源にのみ含まれる IOC が多数存在する．今回は 40 の情報源を用いたが，これ以外の情報源も存在しうることから，今回の実験で得た情報は，必ずしも包括的なものではない．

5.3 研究倫理

本稿における評価用の CTI を収集する際，同一のサイトから情報を取得する場合は，アクセス毎に一定の間隔をおいている．加えて，設計の章で述べた通り，記事の更新有無を確認し，更新がない場合はそれ以上のアクセスを試みないようにしている．これらの施策により，CTI 配布サイトに対する負荷を低減し，実験を行った．

6. 関連研究

辞書やオントロジを作成することによって，非構造化データの構造化を試みる研究がある [3-6]．ただし，セキュリティ分野では, 新たなマルウェアの出現や脆弱性の発見，コードネームの付与等により，新語が生まれやすいことから，継続的な辞書やオントロジのメンテナンスが容易ではない．こうした課題を緩和するべく，提案手法と同様に機械学習ベースの自然言語処理によって非構造化データの構造化を試みる研究もある [20-24]．特に，iACE [24] は，固有表現の抽出だけでなく．グラフマイニングを用いて IOC に係る文脈情報の抽出を試みるものである．ただし，本稿で言及したような遠距離にある IOC との関係性抽出は行っていない．

構造化以外にも CTI の活用に焦点を当てた研究が多数実施されている．FeatureSmith [25] は，CTI をテキストマイニングすることにより特徴量を生成し，Android[e]マルウェアを検出するモデルを自動構築する．文献 [26] も CTI から検出ルールを自動構築する研究である．TTPDrill [27] は, CTI をテキストマイニングし, 記載内容を攻撃手口（TTPs）や Cyber Kill Chain に割り当てるものであり，ChainSmith [28] は, CTI から抽出した IOC の役割を推定するものである．また，POIROT [29] は，audit ログと CTI をそれぞれグラフ化して比較することにより, Threat Hunting を行うものである．これらは，CTI を活用する有用な研究ではあるも

e Android は，グーグル エルエルシーの商標または登録商標である．

のの，今回 CyNER を用いて実施した CTI の横断分析を研究対象とはしていない．

また，CTI の横断分析を試みている研究も複数存在する．文献 [30] は，複数のブロックリストに含まれる IP アドレスやドメインを調査しており，多くの IOC が 1 つのリストに固有であることを明らかにしている．文献 [19] も同様に複数のブロックリストを調査したものである．本稿では，ブロックリストではなく非構造化状態の CTI に焦点を当て，構造化を図ったうえで横断的な分析を実施した．

7. おわりに

我々は，自然言語で記述された CTI の分析を効率化することを目的とし，自動で STIX へと変換する手法である CyNER の研究を進めている．本稿では，CyNER における固有表現抽出の精度向上手法について述べた．具体的には，サイバーセキュリティ分野のコーパスを用いて学習を行い，ドメインに特化した言語モデルを構築することで，認識精度の向上を図る手法を提案した．

評価では，約 3,000,000 行から成るドメインコーパスを構築し，言語モデルの学習を行うことにより，汎用の言語モデルを用いた場合よりも F 値を最大 2.6 ポイント向上できることを示した．また，CyNER を用いて 40 の情報源，52,292 件の CTI から IOC を 270,047 件抽出し，横断的な分析を実施した．この結果，既存のレピュテーションサービスには含まれていない IOC を抽出できることを示した．加えて，97%以上にあたる 262,174 件の IOC が一つの情報源にのみ含まれることを示した．CyNER では，CTI を横断的かつ一元的に分析できるため，このように情報源ごとに分断された IOC 群を一元的に取り扱うことができる．また，19,010 件の IOC は継続的に報告されていること，および 1 年以上の間，複数の攻撃グループにまたがって悪用される IOC の存在を示した．CyNER では，同じ IOC に紐付く CTI 群を時系列で表示することができる．さらに，複数の情報源にまたがる IOC や長期的に利用される IOC を強調して表示することもできる．これらにより，IOC に係る時系列での分析や長期にわたって利用されている危険性の高い IOC の抽出を支援することができる．上記の結果より，CyNER が CTI 分析の効率化に寄与する見込みを得た．

今後の課題としては，各タスクの精度向上やより大規模なデータセットを用いた評価が挙げられる．

本稿中で使われているシステム・製品名は，各社の商標または登録商標である．

参考文献

[1] McNeil, N., Bridges, R.A., Iannacone, M.D., et al.: PACE: Pattern Accurate Computationally Efficient Bootstrapping for Timely Discovery of Cyber-security Concepts, *Proc. ICML 2013*, pp. 60-65 (2013).

[2] IBM: IBM Watson to Tackle Cybercrime, available from <https://www-03.ibm.com/press/us/en/pressrelease/49683.wss> (2021-02-07 accessed).

[3] Obrst, L., Chase, P. and Markeloff, R.: Developing an Ontology of the Cyber Security Domain, *Proc. STIDS 2012*, pp.49-56 (2012).

[4] Iannacone, M., Bohn, S., Nakamura, G., et al.: Developing an Ontology for Cyber Security Knowledge Graphs, *Proc. CISR 2015*, pp. 1-4 (2015).

[5] Lim, S.K., Muis, A.O., Lu, W., et al.: MalwareTextDB: A Database for Annotated Malware Articles, *Proc. ACL 2017*, pp.1557-1567 (2017).

[6] Mavroeidis, V. and Bromander, S.: Cyber Threat Intelligence Model: An Evaluation of Taxonomies, Sharing Standards, and Ontologies within Cyber Threat Intelligence, *Proc. EISIC 2017*, pp. 91-98 (2017).

[7] 藤井翔太, 川口信隆, 重本倫宏, 山内利宏：Cyber Threat Intelligence の構造化による分析支援手法の提案，情報処理学会研究報告，Vol. 2021-CSEC-92，No. 47，pp. 1-8 (2021).

[8] OASIS: Introduction to STIX, available from <https://oasis-open.github.io/cti-documentation/stix/intro.html> (2021-08-21 accessed).

[9] MISP project: MISP - Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing, available from <https://www.misp-project.org/> (2021-08-21 accessed).

[10] Mandiant: OpenIOC, available from <https://github.com/mandiant/OpenIOC_1.1> (2021-02-07 accessed).

[11] Devlin, J. and Chang, M.W., Lee, K., et al.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *Proc. NAACL 2019*, pp.4171-4186 (2019).

[12] Lee, J., Yoon, W., Kim, S., et al.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics*, Vol. 36, Issue 4, pp. 1234-1240 (2020).

[13] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, et al.: LEGAL-BERT: The Muppets straight out of Law School, arXiv (2020).

[14] C. Raffel, N. Shazeer, A. Roberts, et al.: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, *JMLR*, Vol. 21, No. 140, pp. 1-67 (2020).

[15] Hugging Face: Transformers, available from <https://huggingface.co/transformers/> (2021-02-07 accessed).

[16] Z. Lan, M. Chen, S. Goodman, K., et al: ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, *Proc. Eighth International Conference on Learning Representations* (*ICLR 2020)*, pp. 1-17 (2020).

[17] Y. Liu, M. Ott, N. Goyal, et al.: RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv:1907.11692. (2019).

[18] Sebastián, S. and Caballero, J.: AVClass2: Massive Malware Tag Extraction from AV Labels, *Proc. ACSAC 2020*, pp. 42-53 (2020).

[19] Li, V. G., Dunn, M., Pearce, P., et al.: Reading the Tea Leaves: A Comparative Analysis of Threat Intelligence, *Proc. SEC 2019*, pp. 851-867 (2019).

[20] Mulwad, V., Li, W., Joshi, A., et al.: Extracting Information about Security Vulnerabilities from Web Text, *Proc. WI-IAT 2011*, pp. 257-260 (2011).

[21] Joshi, A., Lal, R., Finin, T., et al.: Extracting Cybersecurity Related Linked Data from Text, *Proc. ICSC 2021*, pp. 851-867 (2013).

[22] Jones, C.L., Bridges, R.A., Huffer, K.M.T., et al.: Towards a Relation Extraction Framework for Cyber-Security Concepts, *Proc. CISR 2015*, pp.1-4 (2015).

[23] Ramnani, R.R., Shivaram, K., Sengupta, S., et al.: Semi-Automated Information Extraction from Unstructured Threat Advisories, *Proc. ISEC 2017*, pp.181-187 (2017).

[24] Liao, X., Yuan, K., Wang, X., et al.: Acing the IOC Game: Toward Automatic Discovery and Analysis of Open-Source Cyber Threat Intelligence, *Proc. CCS 2016*, pp. 755-766 (2016).

[25] Zhu, Z. and Dumitraş, T.: FeatureSmith: Automatically Engineering Features for Malware Detection by Mining the Security Literature, *Proc. CCS 2016*, pp. 767-778 (2016).

[26] Feng, X, Liao, X., Wang, X., et al.: Understanding and securing device vulnerabilities through automated bug report analysis, *Proc. SEC 2019*, pp. 887-903 (2019).

[27] Husari, G., Al-Shaer, E., Ahmed, M., et al.: TTPDrill: Automatic and Accurate Extraction of Threat Actions from Unstructured Text of CTI Sources, *Proc. ACSAC 2017*, pp. 103-115 (2017).

[28] Zhu, Z. and Dumitras, T.: ChainSmith: Automatically Learning the Semantics of Malicious Campaigns by Mining Threat Intelligence Reports, *Proc. EuroS&P 2018*, pp. 458-472 (2018).

[29] Milajerdi, S. M., Eshete, B., Gjomemo, R., et al.: POIROT: Aligning Attack Behavior with Kernel Audit Records for Cyber Threat Hunting, *Proc. CCS 2019*, pp. 1795-1812 (2019).

[30] Leigh, M. and Spring, J. M.: Blacklist Ecosystem Analysis: Spanning Jan 2012 to Jun 2014, *Proc. WISCS 2015*, pp. 13-22 (2015).