Computer Security Symposium 2017
23 - 25 October 2017

文書からの脅威と脆弱性知識の自動抽出

山嵜 麿与[†1]　神谷 造[†1]

概要：効率的なインシデントの発見や対応には、各組織に関係する脅威と脆弱性に対する知識が不可欠である。WEB上の文書には多様な知識が含まれているが、構造化されていない膨大な文書から必要となる知識を得ることは容易ではない。そこで本稿では、文書から脅威と脆弱性に関係する知識を自動抽出する手法を提案する。従来手法とは異なり、自然言語処理における固有表現認識と関係抽出手法を用いることで、複雑化する脅威と脆弱性に関係する知識を自働抽出できる。さらに、教師有り学習で必要となるデータセットを実際に人手で作成し実施した評価実験の結果を報告する。

キーワード：脅威情報，脆弱性情報，固有表現認識，関係抽出

Automatic Extraction of Threat and Vulnerability Knowledge from Documents

Mayo YAMASAKI[†1]　Itaru KAMIYA[†2]

Abstract: Due to effective incident discovery and response, organizations need to have knowledge about threats and vulnerabilities related themselves. Although documents on the WEB contain various knowledge, these documents are unstructured, therefore it is not easy to obtain necessary knowledge. In this paper, we propose an automated method to extract threat and vulnerability knowledge from documents. Unlike existing methods, our method using natural language processing techniques called named entity recognition and relation extraction for extracting complex threat and vulnerability knowledge. Furthermore, we developed a dataset required for supervised machine learning and show the results of evaluation experiments.

Keywords: Threat Information, Vulnerability Information, Named Entity Recognition, Relation Extraction

1. はじめに

サイバー攻撃の増加に伴い，インシデントの発見と対処に必要となる脅威や脆弱性情報が，WEB 上に日々公開されている．一月間に，約 60,000 件のセキュリティブログと 1,000 件のセキュリティ研究報告書が公開されているとされるが[a]，これらの文書は構造化されていなために，膨大な文書の中から必要となる情報を探索し把握することは容易ではない．従って，効率的に必要な情報を得るために，膨大な文書から構造化された知識を抽出する技術が必要である．

非構造化文書から目的に応じた構造を抽出する技術は，自然言語処理の分野では情報抽出タスクと知られている．情報抽出タスクは，固有表現認識[1]と関係抽出[2]サブタスクに分割でき，それぞれ F1 値で 90%程度[3]と 85%程度[4]の精度で解けることが報告されている．

そこで本稿では，これらの自然言語処理技術を用いて，セキュリティに関する非構造化文書の中から，脅威と脆弱性に関連する構造化された知識を抽出する手法を提案する．これにより，膨大な文書の中から必要な脅威や脆弱性情報を，効率的に得ることが可能となる．

本研究の貢献は次の通りである．

- 脅威と脆弱性に関する構造を自動で抽出するために，STIX（Structured Threat Information eXpression）2.0[b]を参考とした情報抽出タスクを新たに提案する．図 1 に入力文と、出力される構造化された知識グラフの例を示す． 図 1 の出力例では，*Rig Exploit Kit* がマルウェア，*Adobe Flash Player* が製品，*CVE-2015-8651* が CVE（Common Vulnerabilities and Exposures）を意味する固有表現であることを示している．また，*Rig Exploit Kit* から *Adobe Flash Player*，及び *Rig Exploit Kit* から *CVE-2015-8651* に標的の関係が，*CVE-2015-8651* から *Adobe Flash Player* に帰属の関係があることを示している．
- 提案するタスクを評価するためのデータセットを人手で構築した．データセットへの正解ラベルの付与基準と，データセットの詳細について報告する．
- 教師あり学習手法を用いた評価実験の結果を報告する．評価実験では，統合後の固有表現認識と関係抽出の正解ラベルの自動付与精度が，それぞれ F1 値で約 8 割と約 7 割の精度で抽出可能であることを示す。

†1 NTT セキュアプラットフォーム研究所
NTT Secure Platform Laboratories

a https://www-03.ibm.com/press/us/en/pressrelease/49683.wss

b https://oasis-open.github.io/cti-documentation/

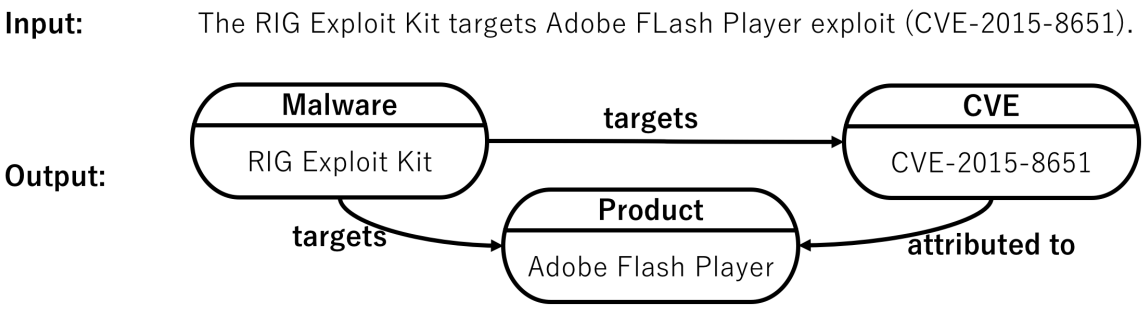

図 1 非構造化文とそこから抽出した脅威と脆弱性知識の例

2. 関連研究

代表的な固有表現認識タスクとして，CoNLL-2003 がある[1]. このタスクでは, 固有表現の種別として *LOCATION*, *PERSON*, *ORGANIZATION*, *MISC* が定義されており，近年では深層学習を用いた手法で，高い精度が報告されている[5]. サイバーセキュリティに関連する固有表現認識タスクとして，Joshi ら[6]や Bridges ら[7]は脆弱性に関する固有表現認識を提案している．Ramnani ら[8]は，*Campaigns*，*Threat Actors*，*IOC*（*Indicator of Compromise*），*TTP*（*Tactics, Techniques and Procedures*）, *Exploit Target* という 5 種類の表現を定義し，*Vulnerability in <Exploit Target>*のようなパターンマッチによる認識手法を提案している.

また，代表的な関係抽出タスクとして Semeval-2010 Task 8[2]がある. このタスクでは, *Cause-Effect*, *Instrument-Agency*, *Product-Producer* ， *Content-Container* ， *Entity-Origin* ， *Entity-Destination*，*Component-Whole*，*Member-Collection*，*Message-Topic* という 8 種類の名詞句間の関係を識別する．関係抽出タスクにおいても，深層学習を用いた手法が高い精度を報告している[4]. サイバーセキュリティに関連する関係抽出タスクとしては，脆弱性情報間に関する関係種別を提案している Jones らの手法がある[9].

提案する情報抽出タスクでは，脅威と脆弱性情報に対する固有表現認識タスクと，それに続く関係抽出を行うことにより, 図 1 の様に, 各固有表現間の意味関係を認識する.

3. 提案タスク

提案する脅威と脆弱性知識の自動抽出タスクでは，入力として文の集合をとり，各文から固有表現とそれらの関係を抽出し，すべての文の抽出結果を結合することにより，一つの知識グラフを出力する．この章では，提案する固有表現認識と関係抽出，及び，抽出結果を結合する 3 つのサブタスクの詳細について述べる.

3.1 固有表現認識

固有表現認識では, 与えられた文を文字の系列とみなし，系列ラベリングにより，文内の固有表現位置と種別を識別する. 図 1 の例文に固有表現認識を行った例を以下に示す.

> The [RIG Exploit Kit]$_{Malware}$ targets [Adobe Flash Player]$_{Product}$ exploit ([CVE-2015-8651]$_{Cve}$).

この例では固有表現の位置を鉤括弧で，鉤括弧に続く下付き文字でその種別を表している．固有表現の種別は文字列に対して重なりなく付与される.

脅威と脆弱性情報に関する固有表現種別を表 1 に示す.

表 1 固有表現種別の一覧

固有表現種別	意味
AttackPattern	サイバー攻撃手法の名称.
Campaign	サイバーキャンペーンの名称.
Cve	共通脆弱性識別子.
Domain	FQDN(完全修飾ドメイン名).
Hash	ハッシュ値.
Identity	個人，組織の名称.
Industry	業種，業界，産業の名称.
Ip	IPv4 アドレス.
Malware	マルウェアの名称.
Product	ソフトウェア，ハードウェア，サービス，製品の名称.
Region	地域の名称.
Role	個人の役割，職種の名称.
ThreatActor	サイバー攻撃者，グループの名称
Time	時間表現.
Version	バージョンの名称.

3.2 関係抽出

関係抽出では，固有表現 e_1，e_2 とそれらを含む文が与えられたときに，e_1，e_2 間の関係種別と方向を識別する.

図 1 の例にある *RIG Exploit Kit* から *CVE-2015-8651* への標的の関係を *targets(RIG Exploit Kit, CVE-2015-8651)*と表現する．別名を意味する *aliases* を除き，関係は有向であるため，*targets(CVE-2015-8651, RIG Exploit Kit)*は成立しない.

関係は同一文内の固有表現間だけに定義されるものとし，文をまたぐ関係は考慮しない.

脅威と脆弱性に関する関係種別を表 2 に示す.

表 2 関係種別の一覧

固有表現種別	意味
attributed-to(e_1, e_2)	e_1 は e_2 に帰属，起因する．
aliases(e_1, e_2)	e_1 と e_2 は同義語である．
indicates(e_1, e_2)	e_1 は e_2 を指し示す．
observed-in(e_1, e_2)	e_1 が e_2 に観測される．
uses(e_1, e_2)	e_1 が e_2 を使用する．
targets(e_1, e_2)	e_1 が e_2 を攻撃の標的とする．

3.3 結果の統合

前述した固有表現認識と関係抽出により，各文内に存在する脅威と脆弱性に関する知識は構造化された．最後に，全体を統合することにより，単一の知識グラフを作成する．

図 1 に示した通り，知識グラフは，固有表現種別とそれが付与された文字列を属性として持つノードと，関係種別を属性として持つエッジから構成される．従って，結果の統合サブタスクでは，文の集合と各文内のノードとエッジの集合が与えられたときに，全てのノード間，エッジ間が統合可能か否かを識別する．

4. データセット

WEB 上の英語で記述されたセキュリティ関連文書に対して，固有表現認識と関係抽出サブタスクための正解ラベルを付与し，評価用のデータセットを構築した．

4.1 固有表現認識の正解ラベル付与基準

正解ラベルの付与揺れを減らすために，固有表現種別の付与に対して，主に以下の基準を設けた．

- 固有表現に連続するその固有表現の分類を意味する文字列に対しても，固有表現と同一の種別を付与する．例えば，*[Mac OS]*$_{Produc}$ や，*[CTB Locker]*$_{Malware}$ のように付与した．
- 固有表現内に含まれる記号文字も付与対象とする．例えば，*[realstatistics(.)info]*$_{Domain}$ のように付与した．
- URL やファイル名，ソースコード等を意味する文字列内に含まれる表現は付与対象としない．
- マルウェアの分類を意味する表現には攻撃手法の意味も含まれるが，*AttackPattern* の付与対象とはしない．例えば，*[key logging]*$_{AttackPattern}$ のように付与し *keylogger* には *AttackPattern* を付与しなかった．
- *Malware* は，あるマルウェアファミリの名称に加え，セキュリティベンダの検知名称も付与対象とした．
- 特定の実装を意味しないプロトコル等の仕様は，*Product* の付与対象としない．
- *Version* はバージョン番号と名称に付与する．例えば，*[Android]* $_{Product}$ *[Marshmallow]*$_{Version}$ のように付与した

4.2 関係抽出の正解ラベル付与基準

正解ラベルの付与揺れを減らすために，関係種別の付与に対しても，主に以下の基準を設けた．

- 与えられた文内にて，明示的に示されている関係だけを付与対象とし，前後の文の情報は考慮しない．
- 正解データの付与作業を効率化するために，特定の固有表現種別間に成立する関係種別に制限を設けた．固有表現 e_1 から e_2 の関係種別を r としたときに成立しうる e_1 と e_2 の固有表現種別の集合を表 3 に示す．ただし，***ALL*** はすべての固有表現種別の集合を，記号-は差集合演算をそれぞれ意味する．また *aliases* には，同一の固有表現間にだけ付与する規則を設けた．

表 3 関係種別に対する制限

関係種別 r	e_1 の固有表現種別集合	e_2 の固有表現種別集合
attributed-to	{Campaign}	{ThreatActor, Identity, Role, Industry, Region}
attributed-to	{Cve}	{Product, Version}
attributed-to	{Identity}	{Role, Industry, Region}
attributed-to	{Product}	{Identity}
attributed-to	{ThreatActor}	{Identity, Role, Industry, Region}
attributed-to	{Version}	{Product}
targets	{AttackPattern, Version, Campaign, Domain, Identity, Industry, Role, Malware, Product, Ip, Region, ThreatActor}	{Domain, Ip, Identity, Role, Industry, Region, Cve, Product, Version}
observed-in	**ALL** - {Time}	{Time}
uses	{AttackPattern, Version, Malware, Product, }	{Malware, Product, Version}
uses	{Campaign, ThreatActor}	{AttackPattern, Version, Malware, Product}
uses	{Identity, Industry,Region, Role}	{AttackPattern, Version, Malware, Cve, Product}
indicates	{Domain}	{AttackPattern, Version, Campaign, Ip, Malware, ThreatActor, Identity, Role, Industry, Region, Product}
indicates	{Hash, Ip}	{AttackPattern, Version Campaign, Malware, ThreatActor, Identity, Role, Industry, Region, Product}

4.3 評価用データセットの概要

アノテーションツールである brat[c]を用いて，作業者 5 名により正解ラベルの付与作業を行った．文分割及びトークン化を Stanford CoreNLP[d]で行い，トークン単位でのラベル付けを行った．200 文書（10,106 文, 225,127 トークン, 15,976 トークン種，一文平均トークン数 22）に対して正解ラベル付けを行い得られた結果を表 4 と表 5 に示す．種別 O は，いずれの種別にも該当しないトークンや関係を意味している．表 4 の固有表現種は，文字列と固有表現種別が同じ固有表現の異なり数である．また表 5 の関係種は，始点と終点の固有表現種が同一でかつ，関係種別が同じ関係の異なり数である．

c http://brat.nlplab.org
d https://stanfordnlp.github.io/CoreNLP/

表 4 データセット内の固有表現ラベル

固有表現種別	ラベル数	固有表現数	固有表現種数
AttackPattern	1373	882	219
Campaign	499	269	58
Cve	113	106	54
Domain	2312	581	228
Hash	1266	864	764
Identity	3380	1890	812
Industry	886	625	212
Ip	520	221	139
Malware	3419	2117	403
Product	3263	2167	568
Region	1719	1461	311
Role	498	373	122
ThreatActor	928	679	124
Time	1817	851	399
Version	228	208	89
O	202906	-	-
合計	225127	13294	4502

表 5 データセット内の関係ラベル

関係種別	関係数	関係種数
attributed-to	537	420
aliases	414	328
indicates	82	79
observed-in	1041	971
uses	713	574
targets	1293	1079
O	286206	-
合計	290286	3451

5. 実験

4 章で述べた評価用データセットを用いて，抽出精度を計測する評価実験を実施した。

5.1 データセット

前処理として Stanford CoreNLP による文分割を行い，各文に対して，トークン化，品詞タグ付け，構文解析，固有表現認識（脅威と脆弱性に関連しない固有表現種別）を行った.

また, 評価用データセット中に存在する極端に短い文は，ノイズになると考えられるため，長さ 5 以上の 8,677 文だけを評価実験の対象とした．これにより固有表現数は 13,294 から 13,069 個, 関係数は 4,080 から 4,079 となった.

系列ラベリングで固有表現認識を行う場合，IO 形式や IOB 形式，IOBES 形式等がある．IO 形式では，複数のトークンからなる固有表現とその種別 LABEL がある場合に固有表現内のトークンを意味する I-LABEL とそれ以外のトークンである O を用いる表現形式である. また IOB 形式では，固有表現の開始トークンを意味する B-LABEL を導入し， IOBES ではこれらに加え，終わりのトークンを意味する E-LABEL と単一のトークンからなる固有表現を意味する S-LABEL を用いる表現形式である.

本実験では, これらの前処理を行った 3 種類の形式（IO, IOB，IOBES）のデータセットを用いた.

データセットを無作為に分割し，8 割を訓練・開発データと，残りの 2 割を評価データとして実験を行った.

5.2 実験手法

5.2.1 固有表現認識手法

脅威と脆弱性に関する固有表現認識では, Ratinov らの手法[10]を参考に，以下に示す特徴量と識別モデルである CRF（Conditional Random Fields）[11]を用いた.

- トークンの表層系
- トークンの品詞タグ
- トークンの固有表現種別(Stanford CoreNLP で取得した，脅威と脆弱性に関連しない固有表現種別)
- トークンが句読点か否か
- トークンの記号と数字を削除した表層系
- トークン文字の前 4 文字と後 4 文字
- 階層的クラスタリング手法である Brown Clustering[12]を行い, トークンの階層クラスの根から 4，6，10，15 クラスまでの経路
- トークン内の長さ 2-5 の文字 N-gram
- demonyum であるか否か
- Wikipedia のページタイトルから取得した語彙に，トークンが含まれているか否か

これらの特徴量は，系列ラベリング中のあるトークンとその前後 2 トークンから取得し，あるトークンの前後 8 トークンに含まれる表層系の集合（Bag of Words）も特徴量に加えた。

CRF の *L1*, *L2* 正則化パラメータ C_1, C_2 はそれぞれ，10^{-5} から 10^{5} までの *10* 倍毎の範囲で 15 回の Random Search[13]を行い決定した．Random Search では，訓練・開発データを 4 分割した交差検証を行い, マクロ F 値により評価した.

5.2.2 関係認識手法

脅威と脆弱性に関する関係認識では，Rink らの手法[14]を参考に, 次に示す特徴量と識別関数である SVM（Support Vector Machine）[15]を用いた．以下に，脅威と脆弱性に関する固有表現 e_1 と e_2 及び, それらを含む文 s が与えられたときの特徴量を示す.

- e1，e2 それぞれに含まれるトークンの表層系，品詞タグ，Stanford CoreNLP の固有表現タグ，脅威と脆弱性に関する固有表現タグ，WordNet[e]上にあるトークンの上位語
- e1 のトークンまたは e2 トークンに含まれる表層系，Stanford CoreNLP の固有表現タグ，WordNet 上にあるトークンの上位語

e https://wordnet.princeton.edu

- e1，e2 間にあるすべてのトークンの表層系，表層系の前 5 文字，品詞タグ，Stanford CoreNLP の固有表現タグ，WordNet 上にあるトークンの上位語
- e1，e2 の外側にあるトークンの表層系を 1 つずつ
- e1 と e2 の距離（トークン数）
- e1 と e2 を含む係り受け木に関して，係り受け距離 1 の関係と表層系，距離 1 の関係と VerbNet [f]のクラス，距離 2 の関係と VerbNet のクラス，距離 2 の関係と文内の位置関係（BEFORE，BETWEEN，AFTER）

Rink らの手法では，関係種別の分類とその方向の分類を行う 2 種類の SVM を用いているが，評価用データセットでは精度の改善が見られなかったため，1 種類の SVM で関係の分類を行った．

SVM は線形カーネルを用い，one-vs-the-rest で多クラス分類を行った．ペナルティ項 C は，10^{-5} から 10^{5} までの 10 倍毎の範囲範囲で全探索を行い決定した．固有表現認識と同様に，訓練・開発データを 4 分割した交差検証を行い，マクロ F 値により評価した．また評価用データセットが不均衡であるため，分類クラス数と各クラスの事例数の積で全事例数を割った値を用い，C に重み付けした．

5.2.3 結合手法

各文から得られた脅威と脆弱性に関する固有表現と関係を以下の規則に基づき結合した．

- 全文から得られた固有表現の中で，種別と大文字に変換後の文字列が同一の固有表現を結合．
- 全文から得られた関係の中で，始点となる固有表現同士と終点となる固有表現同士が同一で，かつ，関係種別が同一の関係を結合する

5.3 実験内容と結果

5.3.1 固有表現認識

まず初めに，IO，IOB，IOBES 形式のデータセット間で，固有表現認識精度の比較を行った．ラベル毎のマクロ F 値による評価結果を表 6 に示す．前述した手法を用いて，それぞれのデータセットに対して 3 回計算したマクロ F1 値の平均値を求めた．この結果から，IO 形式での固有表現認識が，最も精度が高いことがわかる．

表 6 各形式でのラベル毎のマクロ F1 値の比較

形式	開発データ	評価データ
IO	**0.82**	**0.83**
IOB	0.75	0.75
IOBES	0.66	0.66

次に，IO 形式のデータセットに対して，各ラベルの認識精度の評価を行った結果を表 7 に示す．先程と同様，3 回計算した結果の平均値を示す．

表 7 固有表現認識の精度

固有表現種別	Precision	Recall	F1
AttackPattern	0.81	0.61	0.70
Campaign	0.91	0.87	0.89
Cve	0.97	0.79	0.86
Domain	0.92	0.82	0.87
Hash	0.98	0.98	0.98
Identity	0.82	0.71	0.76
Industry	0.69	0.54	0.60
Ip	0.95	0.87	0.91
Malware	0.80	0.76	0.78
Product	0.82	0.87	0.84
Region	0.81	0.49	0.61
Role	0.91	0.83	0.87
ThreatActor	0.91	0.94	0.92
Time	0.97	0.61	0.74
Version	0.80	0.76	0.78
O	0.98	0.99	0.99
平均/合計	0.88	0.78	0.83

5.3.2 関係認識

関係認識でも前述した手法を用いて 3 回計算した精度の平均値を求めた．結果を表 8 に示す．

表 8 関係認識の精度

関係種別	Precision	Recall	F1
attributed-to	0.45	0.54	0.49
aliases	0.75	0.70	0.71
indicates	0.65	0.35	0.44
observed-in	0.68	0.89	0.77
uses	0.56	0.66	0.60
targets	0.68	0.83	0.74
O	1.00	0.99	1.00
平均/合計	0.68	0.71	0.68

5.3.3 結果の結合

全ての文から得られた固有表現と，それらの関係を統合し，重複する固有表現と関係を排除した．結合後の知識グラフに対して 3 回計算した精度の平均値を，表 9 に示す．

表 9 統合後の知識グラフ上での精度

	Precision	Recall	F1
固有表現	0.85	0.74	0.79
関係	0.66	0.76	0.71

5.3.4 学習曲線

評価用データセット全体を 5 分割した交差検証により求めた，それぞれの学習曲線を図 2 と図 3 に示す．図 2 は訓練文数に，図 3 は訓練関係数に対するマクロ F 値の推移と 1 標準偏差の範囲を示している．

f https://verbs.colorado.edu/~mpalmer/projects/verbnet.html

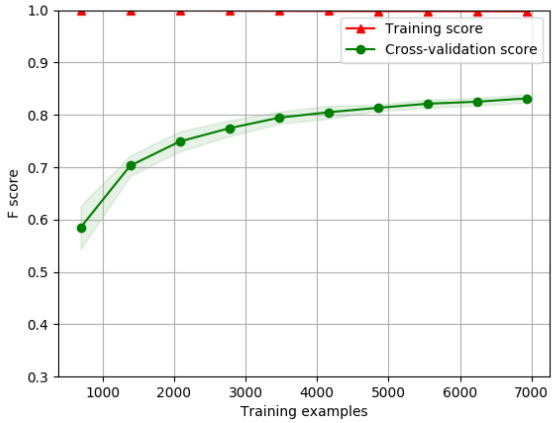

図 2 固有表現認識の学習曲線

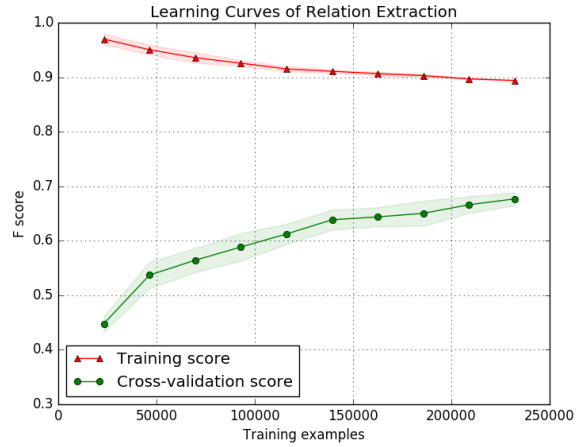

図 3 関係認識の学習曲線

6. 考察

表 6 では IO 形式での精度が最も高かった．これは，固有表現種別数を K としたときに，IO 形式では $K+1$ クラスであるが，IOB 形式では $2K+1$，IOBES 形式では $4K+1$ クラスになるため，IOB や IOBES 形式では，正解ラベルの少ないクラスが発生し，精度が低下したと考えられる．従って今後は，正解ラベル数のより多いデータセットを作成し，改めて精度を評価する必要がある．

また表 7，8，9 より，固有表現認識が約 8 割，関係抽出が約 7 割の精度で実現できることがわかった．今後は，インシデントの発見や対処で活用するために要求される精度の基準を検討する必要がある．

7. おわりに

本稿では，複雑化するサイバー攻撃に関する脅威と脆弱性知識を自動で構造化するための情報抽出タスクを新たに提案した．この提案タスクでは，固有表現認識と関係抽出，結果の統合により知識グラフを構築する．

さらに，提案タスクを評価するためのデータセットを人手で構築し，固有表現認識では約 8 割の精度で，関係抽出では約 7 割の精度で、自動抽出可能であることを示した．

参考文献

[1] Tjong Kim Sang, Erik F., and Fien De Meulder. "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition." Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4. Association for Computational Linguistics, 2003.

[2] Hendrickx, Iris, et al. "Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals." Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions. Association for Computational Linguistics, 2009.

[3] Xu, Mingbin, and Hui Jiang. "A FOFE-based Local Detection Approach for Named Entity Recognition and Mention Detection." arXiv preprint arXiv:1611.00801 (2016).

[4] Cai, Rui, Xiaodong Zhang, and Houfeng Wang. "Bidirectional Recurrent Convolutional Neural Network for Relation Classification." ACL (1). 2016.

[5] Chiu, Jason PC, and Eric Nichols. "Named entity recognition with bidirectional LSTM-CNNs." arXiv preprint arXiv:1511.08308 (2015).

[6] Joshi, Arnav, et al. "Extracting cybersecurity related linked data from text." Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on. IEEE, 2013.

[7] Bridges, Robert A., et al. "Automatic labeling for entity extraction in cyber security." arXiv preprint arXiv:1308.4941 (2013).

[8] Ramnani, Roshni R., Karthik Shivaram, and Shubhashis Sengupta. "Semi-Automated Information Extraction from Unstructured Threat Advisories." Proceedings of the 10th Innovations in Software Engineering Conference. ACM, 2017. APA

[9] Jones, Corinne L., et al. "Towards a relation extraction framework for cyber-security concepts." Proceedings of the 10th Annual Cyber and Information Security Research Conference. ACM, 2015.

[10] Ratinov, Lev, and Dan Roth. "Design challenges and misconceptions in named entity recognition." Proceedings of the Thirteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2009.

[11] Lafferty, John, Andrew McCallum, and Fernando CN Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." (2001).

[12] Brown, Peter F., et al. "Class-based n-gram models of natural language." Computational linguistics 18.4 (1992): 467-479.

[13] Bergstra, James, and Yoshua Bengio. "Random search for hyper-parameter optimization." Journal of Machine Learning Research 13.Feb (2012): 281-305.

[14] Rink, Bryan, and Sanda Harabagiu. "Utd: Classifying semantic relations by combining lexical and semantic resources." Proceedings of the 5th International Workshop on Semantic Evaluation. Association for Computational Linguistics, 2010.

[15] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: a library for support vector machines." ACM transactions on intelligent systems and technology (TIST) 2.3 (2011): 27.