Introducing a New Dataset for Event Detection in Cybersecurity Texts

Hieu Man Duc Trong[1]*, **Duc Trong Le**[1]*, **Amir Pouran Ben Veyseh**[2]*,
Thuat Nguyen[1]* and **Thien Huu Nguyen**[2]†
[1] Hanoi University of Science and Technology, Hanoi, Vietnam
[2] Department of Computer and Information Science, University of Oregon,
Eugene, OR 97403, USA
{hieu.mdt161530,duc.lt161082,thuat.nh163964}@sis.hust.edu.vn
{apouranb,thien}@cs.uoregon.edu

Abstract

Detecting cybersecurity events is necessary to keep us informed about the fast growing number of such events reported in text. In this work, we focus on the task of event detection (ED) to identify event trigger words for the cybersecurity domain. In particular, to facilitate the future research, we introduce a new dataset for this problem, characterizing the manual annotation for 30 important cybersecurity event types and a large dataset size to develop deep learning models. Comparing to the prior datasets for this task, our dataset involves more event types and supports the modeling of document-level information to improve the performance. We perform extensive evaluation with the current state-of-the-art methods for ED on the proposed dataset. Our experiments reveal the challenges of cybersecurity ED and present many research opportunities in this area for the future work.

1 Introduction

With the proliferation of cyber technologies (i.e., social networks, Internet of Things) in our daily life, the frequency of cyberattacks and cybercrimes is also rapidly increasing, potentially imposing serious threats to our cyber activities. Cybersecurity has thus become an important field for which a large amount of text data would be produced to report and discuss various aspects of cyber vulnerabilities. The expected sheer amount of this type of data calls for automatic techniques to analyze the cybersecurity text and extract useful knowledge. Among others, these techniques can help to detect the trends of the cyberattacks for better policy making or populate cybersecurity knowledge bases for automatic reasoning systems.

In this work, we examine Information Extraction technologies (IE) in Natural Language Processing (NLP) as a promising candidate for the knowledge extraction task from cybersecurity text. In particular, we focus on Event Detection (ED), an important task in IE that seeks to identify trigger words of specified types of events in text (Ahn, 2006; Ji and Grishman, 2008). ED is an actively studied task in IE where deep learning models have been the dominant approach to deliver the state-of-the-art performance (Nguyen and Grishman, 2015; Chen et al., 2015). For instance, consider the sentence:

Remote attackers to completely **takeover** *player accounts just by tricking users into clicking an unsuspectable link.*

An ED system for cybersecurity texts should be able to identify "*takeover*" as an event trigger word of the event type *ATTACK.User Compromise* in this case.

In order to enable the application of the ED methods in the cybersecurity domain, a crucial requirement has to do with the benchmark datasets to facilitate the development and evaluation of ED models. Unfortunately, most of the current benchmark datasets for ED (i.e., the ACE and TAC KBP datasets (Walker et al., 2006; Mitamura et al., 2015)) cannot serve this purpose as they have mainly concerned the common events in a person's life of the general domain (e.g., being born, getting married, or being arrested). In addition, the events in the general domain might involve substantial differences with those in the cybersecurity domain (i.e., the divergences in lexical forms, sentence structures and domain expertise), necessitating the development of cybersecurity-focused datasets to aid the research on ED and reveal the nature for the events in this domain. To this end, (Satyapanich et al., 2020) recently presents the first dataset for cybersecurity ED (called CASIE) that annotates event instances with rich annotation. However, this dataset involves at least three limitations that hinder future research in this area.

*The first four authors contribute equally to this paper.
† Corresponding author.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 5381–5390,
November 16–20, 2020. ©2020 Association for Computational Linguistics

First, CASIE only contains a small number of event types (i.e., five types) that fail to cover a wide range of important cyber attack/vulnerability types in reality (Simmons et al., 2014). This would limit the application of the systems and restrict the comprehensiveness of the analysis about cybersecurity events developed from the dataset. Second, the event triggers in CASIE tend to be correctly detected and classified without considering the document-level information (i.e., simply relying on the local contexts in the sentences of the triggers is sufficient). This is not desirable as based on our analysis, the necessary contexts to recognize the event triggers in the cybersecurity domain might span the entire long documents and the limited requirement for document context in CASIE would not be able to reflect the data distribution of the cybersecurity events for ED well. For example, the word "*attack*" can appear somewhere in a document to refer to some event mentioned far away at the beginning of the document. As the local context of "*attack*" does not present any information about the specific type of this attack, the long document context (i.e., up to the beginning of the document) would be crucial to successfully determine the actual event type for "*attack*" in this case. Last but not least, CASIE has not been comprehensively evaluated with the state-of-the-art ED systems, making it challenging to accurately estimate the difficulty/complexity of the dataset.

Consequently, in this work, we introduce a novel dataset for cybersecurity ED (called CySecED) that is manually annotated for 30 event types to better characterize the important cyber attacks and vulnerabilities reported in texts. CySecED involves event triggers whose types can only be predicted if the long-range document context is effectively captured, thus offering a more challenging dataset for ED. Finally, we extensively evaluate the best-performing ED models on CySecED. Our experiments show that the performance of current models is far behind the human performance on this dataset and further research is needed to improve the models' performance. We will publicly release CySecED to promote the future research on ED and NLP for the cybersecurity domain.

2 Data Collection and Annotation

We use the articles on the website "*The Hacker News*" (THN)[1] as the documents for cybersecurity event annotation in this work. THN (written in English) is a trusted and widely-acknowledged cybersecurity news platform that reports the latest cybersecurity news and in-depth coverage of current as well as future trends in cybersecurity.

	CASIE	**CySecED**
# event types	5	30
# positive examples	8,470	8,014
# negative examples	240,682	282,220
# sentences per document (average)	16.69	24.94

Table 1: Statistics for CASIE and CySecED. Negative examples refer to non-trigger words while positive examples are the annotated trigger words for the 30 event types of interest.

In order to create an ED dataset for the cybersecurity domain, we consult the cyberattack taxonomy in (Simmons et al., 2014) to select a set of 30 cybersecurity event types that occur frequently and have high impact in THN. In particular, the 30 event types are grouped into four following categories to reflect four different stages of a cyber attack/vulnerability: **DISCOVER**: a vulnerability in a software or system is detected or mentioned by some entity (i.e., hackers, engineers) (e.g., kernel flows, buffer overflow, back door), **PATCH**: some entity (i.e., software companies) fixes or shows how to fix a known vulnerability, **ATTACK**: an attacker exploits some vulnerability to impact the systems using some means (e.g., user compromise, viruses, spyware, worms, denials of services), and **IMPACT**: the consequence of an attack for a system (e.g., disrupt, breach/disclosure of information). The full list of the event types along with their descriptions and examples for our dataset are shown in Appendix A.

The articles, once crawled from THN, would be processed to extract the title and text content (i.e., removing other elements such as html tags, images, etc.). We recruit two undergraduate students who specialize in security and networking to perform the data annotation for the processed articles. Each student is trained with the annotation guideline about the 30 event types and does a group of exercises to be able to better distinguish the event types. Among others, our guideline only annotates a single word for each event trigger (i.e., the most important word to clearly express the event), following the practice in prior ED work (Nguyen and Grishman, 2015). Overall, we annotate 292 documents and achieve a Cohen's Kappa score of

[1] https://thehackernews.com/

0.79 (i.e., very close to the near-perfect agreement range of $[0.81, 0.99]$). Finally, in order to improve the quality of the dataset, a cybersecurity expert is asked to resolve the cases where the two annotators disagree, leading to the final version of CySecED. Table 1 presents some statistics and comparisons between CASIE (Satyapanich et al., 2020) and the proposed dataset CySecED.

3 Annotation Challenges

We find that cybersecurity event annotation is a challenging task where the major challenges involve the disagreement between the annotators for the subtle cases and the high level of necessary domain expertise to annotate the triggers.

First, for the annotation disagreement, the highest disagreement concerns the decisions to annotate a word as a trigger or not. For instance, consider the following sentence:

"*The mobile* **apps** *in question disguised as photo editing and beauty* **apps** *purporting to use your mobile phone's camera to take better pictures or beautify the snaps you shoot, but were found including code that performs malicious activities on their users' smartphone.*"

In this sentence, both annotators believe the first word "*apps*" is referring to a malware and should be annotated as a trigger word of an event of type *ATTACK.Spyware*. However, for the second word "*apps*", one annotator treats it as a neutral apps and does not label it while the other annotator considers both words "*apps*" as coreferred and mark them as trigger words. The other disagreements have to do with the confusion between the types of cyberattacks where the differences are subtle (i.e., *ATTACK.Virus* vs. *ATTACK.Worm*, *ATTACK.Trojan* vs. *ATTACK.Spyware*, and *ATTACK.Root_Compromise* vs. *ATTACK.Arbitrary_Code_Execution*).

Second, for the domain expertise challenge, in many cases, the understanding about the cybersecurity attacks and crimes is necessary to analyze the context and assign the appropriate event types for the trigger words. For instance, consider the following sentence as an example:

"*Numerous* **cyberattacks** *on automobile companies have been reported yesterday where the hackers used compromised machines to* **hit** *the websites with floods of traffics measuring up to 140Gbps.*"

In this sentence, "*cyberattacks*" and "*hit*" are the trigger words of the events of type *ATTACK.Denial of Service*. As "Denial of Service" is not directly expressed in the sentence, the annotators would need to understand that "*floods of traffics*" is usually associated with Denial of Service attacks to be able to assign event types for these trigger words. In fact, this also presents an unique challenge for ED models in this domain where the recognition of such semantic association is important to successfully perform the task.

Word	Event Count	Event Rate
attack	1,564	42.5%
vulnerability	659	75.8%
malware	544	61.5%
exploit	338	69.2%
infect	296	70.2%

Table 2: Event rates of the words with the highest event counts.

4 Data Analysis

This section conducts some additional analysis to gain a better insight into the proposed dataset CySecED. First, Table 2 demonstrates the ambiguity in CySecED by showing five words with the highest occurrence as event triggers in the dataset (i.e., Event Count), along with the percentage of times they are labelled as event triggers in CySecED (i.e., Event Rate) (Sims et al., 2019). Among others, this table shows that even for popular event trigger words, there are still some chance that they are not labeled as trigger words in the dataset and the models need to appropriately capture the context to correctly make a prediction in these cases.

In addition, we find that sentences mentioning some events in CySecED often contain at least two event trigger words for the event types. In other words, cybersecurity events in CySecED tend to co-occur with each other in the sentences, suggesting potential inter-dependencies between events. These dependencies can be exploited to further improve the ED performance for cybersecurity domain in the future research (Li et al., 2013; Nguyen et al., 2016a; Nguyen and Nguyen, 2019). In particular, among the sentences in CySecED, 45.4% of the sentences do not contain any event triggers, 50.0% of the sentences host at least two event triggers, and only 4.6% of the sentences involve a single event trigger. Among the event types, the highest co-occurrence frequency involves the co-occurrence of two *ATTACK* event triggers in the same sentences (i.e., amounting to 22% of the total sentences in

CySecED).

5 Evaluation

Models: There are two classes of models for ED in the literature, i.e., the sentence-level models (i.e., the models that only exploit the local context information in the sentences of the triggers) and the document-level models (i.e., the models that further consider the document-level contextual information for ED). This section aims to reveal the complexity of CySecED by evaluating the performance of the state-of-the-art models for ED in these model classes. In particular, for the sentence-level class, we focus on the following ED models:

• **CNN** (Nguyen and Grishman, 2015): a Convolutional Neural Network model (CNN) for ED.

• **DMCNN** (Chen et al., 2015): a CNN model for ED with Dynamic Pooling.

• **GCN** (Nguyen and Grishman, 2018): a Graph Convolutional Neural Network model (GCN) based on dependency trees for ED.

• **MOGANED** (Yan et al., 2019): an ED model with Multi-Order Graph Convolution and Attention. This is currently the state-of-the-art ED model with uncontextualized word embeddings in the general domain (i.e., the ACE 2005 dataset).

• **CyberLSTM** (Satyapanich et al., 2020): a LSTM model developed for ED in the cybersecurity domain that exploits different features (e.g., the dependency trees) for the input representation.

Regarding the document-level models, we consider the following representative ED models:

• **HBTNGMA** (Chen et al., 2018): a Collective ED model with Hierarchical and Bias Tagging Networks and Gated Multi-level Attention Mechanisms to exploit the document-level information.

• **DEEB-RNN** (Zhao et al., 2018): a Document Embedding Enhanced Bidirectional RNN for ED.

For the models in this work, we experiment with both the traditional uncontextualized word embeddings `word2vec` (Mikolov et al., 2013b) (i.e., the 300 dimension version) and the recent contextualized word embeddings BERT (i.e., the uncased base model) (Devlin et al., 2019) as the pre-trained word embeddings. For BERT, we additionally evaluate the BERT-based ED models in (Wang et al., 2019) (called **DMBERT**) and (Yang et al., 2019) (called **BERT-ED**) that are the sentence-level models with the best-reported ED performance on the ACE 2005 dataset. Finally, we tune the hyper-parameters for the models using the development data of the datasets in this paper.

Dataset	Training		Test		Development	
	#Pos	#Neg	#Pos	#Neg	#Pos	#Neg
CASIE	6,776	192,937	847	24,804	847	22,941
CySecED	6,382	224,684	835	29,152	797	28,384

Table 3: The size of datasets. #Pos and #Neg represent the numbers of positive and negative examples.

Results: The ED problem in this work is formulated as a word classification problem where given a sentence/document, the models need to predict the event types for its words. The set of event types includes a special type *Other* to indicate the words that are not event triggers (called the negative examples). The positive examples for ED correspond to the event trigger words. In order to evaluate the ED models on CySecED, we use 240 documents for training data, 30 other documents for test data, and the remaining 30 documents for the development data (i.e., the document split ratio of 80:10:10). In order to compare CySecED and CASIE, we also divide the documents in CASIE (Satyapanich et al., 2020) into the training, test, and development data using the 80:10:10 ratio to evaluate the ED models. We train all the ED models in this work with early stopping on the development datasets (i.e., we stop training the models once the performance on the development data decreases). Some statistics about the data splits for CySecED and CASIE are reported in Table 3 while Table 4 shows the performance of the models on the test datasets.

There are several important observations from the table. First, comparing CASIE and CySecED, we see that the performance of the current ED models on CySecED is in general much worse than those for CASIE. This indicates that CySecED is more challenging than CASIE for ED and the future work can use CySecED to evaluate the ED models for the cybersecurity domain. Also, the best performance of the models on CySecED (i.e., 68.4% with DEEB-RNN) is still far behind the human performance on this dataset (i.e., 81.0%), presenting much opportunities for the future research in this area. Second, comparing `word2vec` and BERT, we find that BERT mostly performs comparably or poorer than `word2vec` for different ED models and datasets, potentially due to the large difference between the training data for BERT and the cybersecurity domain. Third, among the sentence-level models, similar to the general domains (i.e., ACE 2005), MOGANED and BERT-ED still have

Model	word2vec		BERT	
	CASIE	CySecED	CASIE	CySecED
CNN	83.9	43.7	83.8	43.0
DMCNN	85.2	43.2	84.0	42.7
GCN	85.5	52.2	85.4	48.9
MOGANED	**86.0**	61.6	**86.5**	56.5
CyberLSTM	81.4	51.8	82.3	34.5
DMBERT	-	-	84.1	55.1
BERT-ED	-	-	84.7	58.1
HBTNGMA	85.9	60.9	85.0	62.4
DEEB-RNN	85.5	**68.4**	85.8	**65.5**

Table 4: The performance (F1 scores) of the models.

the best performance for the cybersecurity datasets. Also, the CyberLSTM model developed for CASIE in (Satyapanich et al., 2020) perform much worse than the state-of-the-art models for ED, showing that CyberLSTM is not sufficient to evaluate the complexity of the datasets for cybersecurity ED. Finally, we see that the document-level model (i.e., DEEB-RNN) is significantly better than sentence-lelvel models for CySecED. This is in contrast to CASIE where the document-level models are only comparable with the sentence-level models. This suggests the advantages of CySecED over CASIE that necessitate the modeling of document-level context information to achieve good performance and better reflect the challenges for cybersecurity ED in CySecED. To illustrate, we provide an example that can only be predicted with document-level information in CySecED in Appendix B. In addition, the large performance difference between the two document-level models (i.e., HBTNGMA and DEEB-RNN) highlights the importance to appropriately capture the document-context information and future research can consider this direction to develop effective models for cybersecurity ED.

6 Related Work

Prior work has applied NLP to perform several tasks for the cybersecurity domain, including privacy policy analysis (Peng et al., 2012; Pandita et al., 2013; Zhu and Dumitras, 2016), text analysis for cybersecurity with social media text (i.e., DDos attack detection, alert generation for threads and vulnerabilities using Twitter) (Mittal et al., 2016; Wang and Zhang, 2017; Sceller et al., 2017; Chambers et al., 2018; Perera et al., 2018; Alguliyev et al., 2019; Hasan et al., 2019), and report and timeline creation of cybersecurity events (Hackmageddon, 2019; PrivacyRight, 2019). However, none of these work considers the detection of event trigger words from cybersecurity articles as we do. Recently, (Lim et al., 2017) presents a dataset for general text-based malware behavior analysis that annotates 39 reports for malware actions and their attributes. This study is then extended in the SecureNLP SemEval evaluation (Phandi et al., 2018). However, different from our CySecED dataset, the annotated dataset in these work is very sparse (i.e., involving less than 5 examples for many labels), hindering the development of the deep learning models (Roy et al., 2019). Also, it does not annotate event triggers for rich event types as we do.

Finally, ED has been extensively studied in the literature (Liao and Grishman, 2010; Li et al., 2013; Nguyen and Grishman, 2015, 2016e; Chen et al., 2015; Nguyen et al., 2016g; Lu and Nguyen, 2018; Liu et al., 2016b, 2017; Chen et al., 2017; Hong et al., 2018; Lai et al., 2020b), partly due to the availability of the large evaluation datasets (i.e., the ACE and TAC KBP datasets (Walker et al., 2006; Mitamura et al., 2015) for the general domains, and the BioNLP datasets (Kim et al., 2009) for the biomedical domain). The closest works to our in the cybersecurity domain involve (Qiu et al., 2016) to extract events on Chinese news, (Khandpur et al., 2017) to perform cyberattack detection on Twitter, and (Satyapanich et al., 2019; Satyapanich et al., 2020) to present the CASIE dataset for event extraction. However, these datasets contain less event types and cannot support the document-level information for the models as CySecED. Finally, we notice some recent interests in new type extension learning, e.g., few-shot/zero-shot learning (Nguyen et al., 2016b; Huang et al., 2018; Lai and Nguyen, 2019; Lai et al., 2020a), that can be helpful to develop ED systems for cybersecurity domain.

7 Conclusion

We present a new dataset CySecED for event detection in the cybersecurity domain. Our dataset is manually annotated for 30 event types and provides sufficient data to develop deep learning models for this task. We extensively evaluate state-of-the-art models for ED on the proposed dataset, showing that the performance of these models is still much worse than the human performance. Our experiments also suggest that document-level information is necessary to perform ED for cybersecurity domain. In the future, we plan to extend our annotation to include event arguments and other properties of events.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*.

R. M. Alguliyev, Aliguliyev, R. M., and F. J. Abdullayeva. 2019. The improved lstm and cnn models for ddos attacks prediction in social media. In *International Journal of Cyber Warfare and Terrorism*.

Nathanael Chambers, Ben Fry, and James McMasters. 2018. Detecting denial-of-service attacks from social media text: Applying NLP to computer security. In *NAACL-HLT*.

Yubo Chen, Shulin Liu, Xiang Zhang, Kang Liu, and Jun Zhao. 2017. Automatically labeled data generation for large scale event extraction. In *ACL*.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *ACL-IJCNLP*.

Yubo Chen, Hang Yang, Kang Liu, Jun Zhao, and Yantao Jia. 2018. Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms. In *EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Hackmageddon. 2019. http://hackmageddon.com. In *Hackmageddon*.

Mahmud Hasan, Mehmet A Orgun, and Rolf Schwitter. 2019. Real-time event detection from the twitter data stream using the twitternews+ framework. In *Information Processing & Management*.

Yu Hong, Wenxuan Zhou, jingli zhang jingli, Guodong Zhou, and Qiaoming Zhu. 2018. Self-regulation: Employing a generative adversarial network to improve event detection. In *ACL*.

Lifu Huang, Heng Ji, Kyunghyun Cho, and Clare R. Voss. 2018. Zero-shot transfer learning for event extraction. In *arXiv preprint arXiv:1707.01066*.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *ACL*.

Rupinder Paul Khandpur, Taoran Ji, Steve T. K. Jan, Gang Wang, Chang-Tien Lu, and Naren Ramakrishnan. 2017. Crowdsourcing cybersecurity: Cyber attack detection using social media. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 2009. Overview of BioNLP'09 shared task on event extraction. In *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*.

Viet Dac Lai, Franck Dernoncourt, and Thien Huu Nguyen. 2020a. Extensively matching for few-shot learning event detection. In *Proceedings of the 1st Joint Workshop on Narrative Understanding, Storylines, and Events (NUSE) at ACL 2020*.

Viet Dac Lai and Thien Huu Nguyen. 2019. Extending event detection to new types with learning from keywords. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019) at EMNLP 2019*.

Viet Dac Lai, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020b. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks. In *EMNLP*.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *ACL*.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *ACL*.

Swee Kiat Lim, Aldrian Obaja Muis, Wei Lu, and Chen Hui Ong. 2017. MalwareTextDB: A database for annotated malware articles. In *ACL*.

Shulin Liu, Yubo Chen, Shizhu He, Kang Liu, and Jun Zhao. 2016b. Leveraging framenet to improve automatic event detection. In *ACL*.

Shulin Liu, Yubo Chen, Kang Liu, and Jun Zhao. 2017. Exploiting argument information to improve event detection via supervised attention mechanisms. In *ACL*.

Weiyi Lu and Thien Huu Nguyen. 2018. Similar but not the same - word sense disambiguation improves event detection via neural representation matching. In *EMNLP*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *NIPS*.

Teruko Mitamura, Zhengzhong Liu, and Eduard Hovy. 2015. Overview of tac kbp 2015 event nugget track. In *TAC*.

Sudip Mittal, Prajit Kumar Das, Varish Mulwad, Anupam Joshi, and Tim Finin. 2016. Cybertwitter: Using twitter to generate alerts for cybersecurity threats and vulnerabilities. In *ASONAM*.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016a. Joint event extraction via recurrent neural networks. In *NAACL*.

Thien Huu Nguyen, Lisheng Fu, Kyunghyun Cho, and Ralph Grishman. 2016b. A two-stage approach for extending event detection to new types via neural networks. In *Proceedings of the 1st ACL Workshop on Representation Learning for NLP (RepL4NLP)*.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *ACL*.

Thien Huu Nguyen and Ralph Grishman. 2016e. Modeling skip-grams for event detection with convolutional neural networks. In *EMNLP*.

Thien Huu Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *AAAI*.

Thien Huu Nguyen, Adam Meyers, and Ralph Grishman. 2016g. New york university 2016 system for kbp event nugget: A deep learning approach. In *Proceedings of Text Analysis Conference (TAC)*.

Trung Minh Nguyen and Thien Huu Nguyen. 2019. One for all: Neural joint modeling of entities and events. In *AAAI*.

Rahul Pandita, Xusheng Xiao, Wei Yang, William Enck, and Tao Xie. 2013. WHYPER: Towards automating risk assessment of mobile applications. In *22nd USENIX Security Symposium (USENIX Security)*.

Hao Peng, Christopher S. Gates, Bhaskar Pratim Sarma, Ninghui Li, Yuan Qi, Rahul Potharaju, Cristina Nita-Rotaru, and Ian Molloy. 2012. Using probabilistic generative models for ranking risks of android apps. In *ACM Conference on Computer and Communications Security, CCS '12*.

I. Perera, J. Hwang, K. Bayas, B. Dorr, and Y. Wilks. 2018. Cyberattack prediction through public text analysis and mini-theories. In *IEEE International Conference on Big Data (Big Data)*.

Peter Phandi, Amila Silva, and Wei Lu. 2018. SemEval-2018 task 8: Semantic extraction from CybersecUrity REports using natural language processing (SecureNLP). In *Proceedings of The 12th International Workshop on Semantic Evaluation*.

PrivacyRight. 2019. http://privacyrights.org. In *Privacyright clearing house*.

X. Qiu, X. Lin, and L. Qiu. 2016. Feature representation models for cyber attack event extraction. In *IEEE/WIC/ACM International Conference on Web Intelligence Workshops (WIW)*.

Arpita Roy, Youngja Park, and Shimei Pan. 2019. Predicting malware attributes from cybersecurity texts. In *NAACL-HLT*.

T. Satyapanich, T. Finin, and F. Ferraro. 2019. Extracting rich semantic information about cybersecurity events. In *IEEE International Conference on Big Data (Big Data)*.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. Casie: Extracting cybersecurity event information from text. In *AAAI*.

Quentin Le Sceller, ElMouatez Billah Karbab, Mourad Debbabi, and Farkhund Iqbal. 2017. SONAR: automatic detection of cyber security events over the twitter stream. In *Proceedings of the 12th International Conference on Availability, Reliability and Security*.

Chris B. Simmons, S. G. Shiva, Harkeerat Singh Bedi, and Dipankar Dasgupta. 2014. Avoidit: A cyber attack taxonomy. In *The 9-th Annual Symposium on Information Assurance (ASIA'14)*.

Matthew Sims, Jong Ho Park, and David Bamman. 2019. Literary event detection. In *ACL*.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. In *Technical report, Linguistic Data Consortium*.

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019. Adversarial training for weakly supervised event detection. In *NAACL-HLT*.

Zhongqing Wang and Yue Zhang. 2017. Ddos event forecasting using twitter data. In *IJCAI*.

Haoran Yan, Xiaolong Jin, Xiangbin Meng, Jiafeng Guo, and Xueqi Cheng. 2019. Event detection with multi-order graph convolution and aggregated attention. In *EMNLP-IJCNLP*.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *ACL*.

Yue Zhao, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2018. Document embedding enhanced event detection with hierarchical and supervised attention. In *ACL*.

Ziyun Zhu and Tudor Dumitras. 2016. Featuresmith: Automatically engineering features for malware detection by mining the security literature. In *ACM Conference on Computer and Communications Security, CCS '16*.

A Event Types in CySecED

There are 30 event types annotated in the proposed CySecED dataset. Tables 5 and 6 present these event types along with their descriptions and examples. Note that for the types and descriptions in this section, we consult the cybersecurity taxonomy in (Simmons et al., 2014). Also, we show the distribution of the event types in CySecED in Figure 1.

B The Necessity of Document-level Information for ED in CySecED

In order to demonstrate the importance of the document-level information for the ED task in CySecED, consider the following document from CySecED:

"All the hackers have been charged with **conducting** *numerous Distributed Denial-of-Service (DDoS)* **attacks** *on major U.S. banks, with Firoozi separately gaining unauthorized access to a New York dam's industrial automation control (SCADA) system in August and September of 2013.*

"This unauthorized access allowed Firoozi to repeatedly obtain information regarding the status and operation of the dam, including information about the water levels, temperature, and status of the sluice gate, which is responsible for controlling water levels and flow rates," a DoJ statement reads.

Luckily, the sluice gate had already been manually disconnected for the purpose of maintenance at the time Firoozi attacked.

The hackers' work allegedly involved Botnets – networks of compromised machines – that **hit** *major American banks, including Bank of America and J.P. Morgan Chase, as well as the Nasdaq stock exchange and knocked them offline."*

There are three event trigger words of type *ATTACK.Denial of Service* (i.e, the words in bold) in this document. Among those, the first two trigger words (i.e., "*conducting*" and "*attacks*") can be easily assigned to this event type based on their local context (i.e., the direct word "*Denial-of-Service*" in the same sentence). However, the classification of the third trigger word "*hit*" to *ATTACK.Denial of Service* is non-trivial as the local context with the nearby sentences does not provide sufficient evidences to determine the correct event type. In this case, it is crucial for the annotators and models to capture the document-level information by looking back further to the beginning of the document to realize "Distributed Denial-of-Service (DDoS) attacks" as the main topic of the document. This topic information can then help to correctly predict the event type for "*hit*". Note that the word "*hit*" in this case is only correctly predicted by the document-level model **DEEB-RNN** and cannot be recognized by the other sentence-level models in this work. Overall, this example illustrates the challenge to encode the document-level information for ED in CySecED, serving as a guidance for the future research in this area.

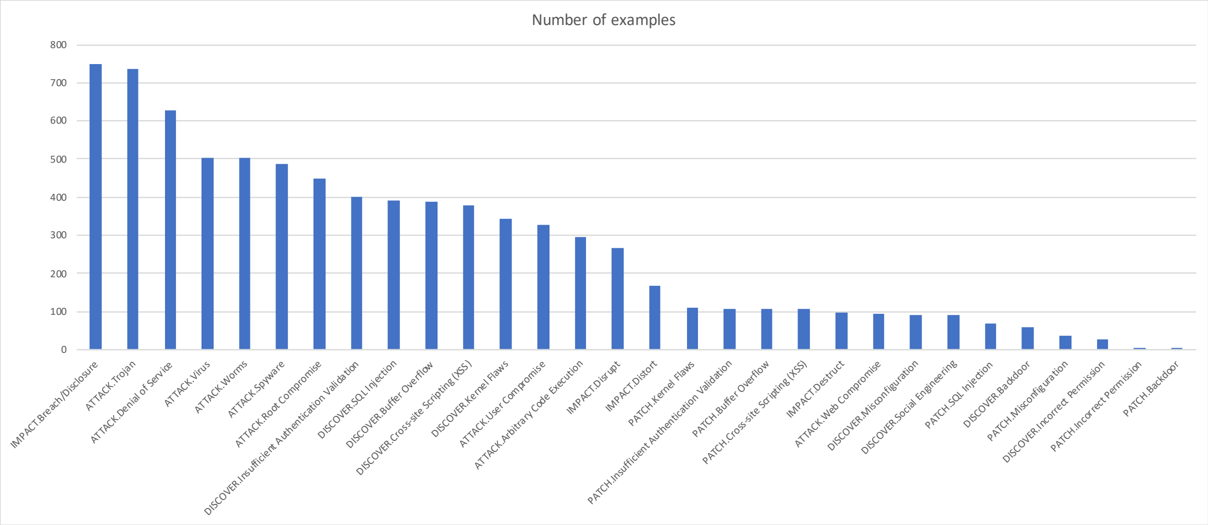

Figure 1: The distribution of the event types in CySecED.

ID	Type	Description	Example (triggers are highlighted)
	DISCOVER	A vulnerability in a software or system is detected or mentioned by some entity (i.e., hackers, engineers).	
1	DISCOVER.Misconfiguration	The discovery/mention of a configuration flaw within a particular application that enables hackers to gain access to a network or personal computer to cause a variety of attacks.	*Researchers* ***found*** *that due to lack of hostname verification, several banking applications were not checking if they connected to a trusted source.*
2	DISCOVER.Kernel Flaws	The discovery/mention of a kernel flaw within an operating system, which is the core code of an operating system, enabling hackers to gain certain privileges to exploit a vulnerability within the operating system.	*Another privilege-escalation* ***vulnerability*** *has been* ***discovered*** *in Linux kernel that dates back to 2005 and affects major distro of the Linux operating system, including Redhat, Debian, OpenSUSE, and Ubuntu.*
3	DISCOVER.Buffer Overflow	The discovery/mention of a buffer overflow where a buffer with weak or no bounds checking is populated with user supplied data. An attacker can exploit a buffer overflow vulnerability for an arbitrary code execution, often of privileges at the administrative level with the program running.	*The* ***vulnerability*** *was described as the stack buffer overflow* ***issue*** *and was* ***discovered*** *by Google's Project Zero staffer Gal Beniamini.*
4	DISCOVER.Insufficient Authentication Validation	The discovery/mention of a program failure to validate the authentication of an application and/or user sent to the program from a user. An attacker can exploit an insufficient authentication validation vulnerability and capture user credentials to impersonate a valid user, which commonly occurs within web applications.	*A critical security* ***vulnerability*** *has been* ***reported*** *in phpMyAdmin-one of the most popular applications for managing the MySQL database-which could allow remote attackers to perform dangerous database operations just by tricking administrators into clicking a link.*
5	DISCOVER.SQL Injection Flaw	The discovery/mention of injection flaws that are not properly validated and sent to an interpreter, usually due to some design flaw. An attacker can exploit this to inject arbitrary code, which commonly occurs within web applications.	*The* ***flaws****,* ***exist*** *in the Joomla version 3.2 to 3.4.4, include SQL injection* ***vulnerabilities*** *that could allow hackers to take admin privileges on most customer websites.*
6	DISCOVER.Cross-site Scripting (XSS)	The discovery/mention of XSS flaws that involve a design flaw not properly validated, allowing malicious scripts to be executed against a vulnerable application in a web browser.	*According to the in-depth technical details* ***shared*** *with The Hacker News, multiple Bigscreen* ***flaws*** *in question are persistent/stored cross-site scripting (XSS)* ***issues*** *that* ***reside*** *in the input fields where VR users are supposed to submit their username, room name, room description, room category in the Bigscreen app.*
7	DISCOVER.Backdoor	The discovery/mention of backdoor, a typically covert method of bypassing normal authentication or encryption in a computer.	*Security researchers have* ***discovered*** *a secret hard-coded* ***backdoor*** *in Western Digital's My Cloud NAS devices that could allow remote attackers to gain unrestricted root access to the device.*
8	DISCOVER.Incorrect Permission	The discovery/mention of an incorrect permission associated to a file or directory that consists of not appropriately assigning users and processes.	*The second* ***bug*** *(CVE-2018-1271)* ***resides*** *in Spring's Web model-view-controller (MVC) that allows attackers to execute directory traversal attack and access restricted directories when configured to serve static resources (e.g., CSS, JS, images) from a file system on Windows.*
9	DISCOVER.Social Engineering	The discovery/mention of a process of using social interactions to acquire information about a victim or computer system. These types of attacks provide quick alternatives in disclosing information to assist an attack that in normal circumstances may not be available.	*A critical security vulnerability has been reported in phpMyAdmin-one of the most popular applications for managing the MySQL database-which could allow remote attackers to perform dangerous database operations just by* ***tricking*** *administrators into* ***clicking*** *a link.*
	PATCH	Some entity (i.e., software companies) fixes or shows how to fix a known vulnerability.	
10	PATCH.Misconfiguration	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Misconfiguration.	*Users are strongly recommended to* ***change*** *default credentials for their devices to* ***prevent*** *against the malware.*
11	PATCH.Kernel Flaws	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Kernel Flaws.	*In response to the Horn's blog post, the maintainers of Ubuntu say the company would possibly* ***release*** *the* ***patches*** *for the Linux kernel flaw around October 1, 2018.*
12	PATCH.Buffer Overflow	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Buffer Overflow.	*In the security note accompanying iOS 10.3.1, Apple describes the issue as a stack buffer overflow vulnerability, which the company* ***addressed*** *by* ***improving*** *the input validation.*
13	PATCH.Insufficient Authentication Validation	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Insufficient Authentication Validation.	*"The* ***use*** *of authentication and authorization of messages, such as the one* ***provided*** *by Spring Security, can limit exposure to this vulnerability only to users who are allowed to use the application", the company suggests.*
14	PATCH.SQL Injection Flaw	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.SQL Injection Flaw.	*Yahoo Quickly* ***Fixes*** *SQL Injection Vulnerability Escalated to Remote Code Execution.*
15	PATCH.Cross-site Scripting (XSS)	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Cross-site Scripting (XSS).	*The latest Webmin and Usermin* ***releases*** *also* ***address*** *a handful of cross-site scripting (XSS) vulnerabilities that were responsibly disclosed by a different security researcher who has been rewarded with a bounty.*
16	PATCH.Backdoor	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Backdoor.	*Webmin developers have now* ***removed*** *the malicious backdoor in its software to* ***address*** *the vulnerability and* ***released*** *the clean versions, Webmin 1.930 and Usermin version 1.780.*
17	PATCH.Incorrect Permission	The mention/description of an update released by some entity to address a vulnerability of type DISCOVER.Incorrect Permission.	*It turns out that before receiving the latest* ***patch****, Guard Provider was downloading antivirus signature updates through an unsecured HTTP connection, allowing man-in-the-middle attackers sitting on open WiFi network to intercept your device's network connection and push malicious updates.*

Table 5: Event types along with their descriptions and examples in CySecED (to be continued in Table 6).

ID	Type	Description	Example (triggers are highlighted)
	ATTACK	An attacker exploits some vulnerability to impact the systems using some means. This can be a mention of an attack or the actions involved in the attack.	
18	ATTACK.User Compromise	A perpetrator gaining unauthorized use of user privileges on a host, as a user compromise.	*Remote attackers to completely **takeover** player accounts just by tricking users into clicking an unsuspectable link.*
19	ATTACK.Root Compromise	Gaining unauthorized privileges of an administrator on a particular host. This is also extended to include any elevated privileges above a normal user including administrative and/or root level privileges to a particular system.	*The flaws, exist in the Joomla version 3.2 to 3.4.4, include SQL injection vulnerabilities that could allow hackers to **take** admin privileges on most customer websites.*
20	ATTACK.Web Compromise	A website or web application using vulnerabilities to further an attack. An attack can occur through a web compromise, usually via cross-site scripting or SQL injection. Web Compromise involves the use of a malformed website or web application an attacker exploits for gain.	*Any malicious site can potentially **make** a victim's web browser **connect** to a My Cloud device on the network and **compromise** it.*
	ATTACK.Installed Malware	An attack that is launched via user-installed malware on a victim system, whether user installed or drive-by installation. Installed malware can allow an adversary to gain full control of the compromised systems leading to the exposure of sensitive information or remote control of the host. There are several subtypes for this type of attacks.	
21	ATTACK.Virus	A form of installed malware or a piece of code that will attach itself through some form of infected files, which will self-replicate upon execution of program. Viruses spread when the infected files they are attached to is transferred from one computer/device to another via the network, file sharing, or email attachments.	*The researcher demonstrated how a malicious attacker could have sent the victim's inbox to an external site, and **created** a virus that **attached** itself to all outgoing emails by secretly **adding** a malicious script to message signatures.*
22	ATTACK.Worm	A self-replicating computer program (a considerable threat to the internet today). Worms do not require human intervention to propagate as it is a self-replicating standalone program that enters a computer/device through a vulnerability in the system and takes advantage of file-transport or information-transport features throughout the network.	*Hajime botnet **works** much like Mirai by **spreading** itself via unsecured IoT devices that have open Telnet ports and uses default passwords and also uses the same list of username and password combinations that Mirai is programmed to use.*
23	ATTACK.Trojan	A harmful program that looks legitimate. Users are typically tricked into loading and executing it on their systems, allowing unauthorized backdoor access to a compromised system (a common way to introduce a victim into a multitude of attacks). Unlike viruses and worms, Trojans do not reproduce by infecting other files nor do they self-replicate. Trojans must spread through user interaction such as opening an email attachment or downloading and running a file from the Internet.	*Dubbed Trojan.Mirai.1, the new Trojan **targets** Windows computers and **scans** the user's network for compromisable Linux-based connected devices.*
24	ATTACK.Spyware	Similar to Trojans, Spyware is a type of malware programs that is covertly installed and infects its target. The difference concerns the purposes of the programs. In particular, Spyware aims to collect information from a computing system without owner's consent while Trojans can have numerous purposes and impact a system tremendously (e.g., ruining the system).	*A new variant of the X-Agent **spyware** is now **targeting** Apple macOS system that has previously been **used** in cyber **attacks** against Windows, iOS, Android, and Linux devices.*
25	ATTACK.Arbitrary Code Execution	Involves a malicious entity that gains control through some vulnerability injecting its own code to perform any operation the overall application has permission.	*Beniamini says this stack buffer overflow issue in the Broadcom firmware code could lead to remote code **execution** vulnerability, allowing an attacker in the smartphone's WiFi range to **send** and **execute** code on the device.*
26	ATTACK.Denial of Service	Denial of Service (DDoS) is an attack to deny a victim access to a particular resource or service.	*Dyn did not disclose the actual size of the **attack**, but it has been speculated that the DDoS **attack** could be much bigger than the **one** that **hit** French Internet service and hosting provider OVH that peaked at 1.1 Tbps, which is the largest DDoS **attack** known to date.*
	IMPACT	The consequence of an attack for a system.	
27	IMPACT.Distort	A distortion in information, usually when an attack has caused a modification of a file. When an attack involves distort, it is a change to data within a file, or modification of information from the victim.	*Last month, ransomware viruses hit two cities in Florida that made large ransom payments to gain back access to city files that were **encrypted** in the attacks.*
28	IMPACT.Disrupt	A disruption in services, usually from a Denial of Service. When an attack involves disrupt, it is an access change, or removal of access to victim or to information.	*The attack **prevented** prepaid customers from buying electricity units.*
29	IMPACT.Destruct	A destruction of information, usually when an attack has caused a deletion of files or removal of access. Destruct is the most malicious impact, as it involves the file deletion, or removal of information from the victim.	*All the attacker needs to do is trick the victims into clicking a specially crafted Facebook URL, as mentioned on his blog, designed to perform various actions like posting anything on their timeline, change or **delete** their profile picture, and even trick users into **deleting** their entire Facebook accounts.*
30	IMPACT.Breach/Disclosure	A disclosure of information, usually providing an attacker with a view of information they would normally not have access to and with the possibility of leading to other compromises.	*VPNFilter is a multi-stage, modular malware that can **steal** website credentials and **monitor** industrial controls or SCADA systems, such as those used in electric grids, other infrastructure and factories.*

Table 6: Event types along with their descriptions and examples in CySecED.