TI-NERmerger: Semi-Automated Framework for Integrating NER Datasets in Cybersecurity

Inoussa Mouiche and Sherif Saad
School of Computer Science, University of Windsor, ON, Canada

Keywords: Threat Intelligence, Named Entity Recognition, Data Annotation, Data Augmentation.

Abstract: Recent advancements highlight the crucial role of high-quality data in developing accurate AI models, especially in threat intelligence named entity recognition (TI-NER). This technology automates the detection and classification of information from extensive cyber reports. However, the lack of scalable annotated security datasets hinders TI-NER system development. To overcome this, researchers often use data augmentation techniques such as merging multiple annotated NER datasets to improve variety and scalability. Integrating these datasets faces challenges like maintaining consistent entity annotations and entity categories and adhering to standardized tagging schemes. Manually merging datasets is time-consuming and impractical on a large scale. Our paper presents TI-NERmerger, a semi-automated framework that integrates diverse TI-NER datasets into scalable, compliant datasets aligned with cybersecurity standards like STIX-2.1. We validated the framework's efficiency and effectiveness by comparing it with manual processes using the DNRTI and APTNER datasets, producing Augmented APTNER (2APTNER). The results demonstrate over 94% reduction in manual labour, saving several months of work in just minutes. Additionally, we applied advanced ML algorithms to validate the effectiveness of the integrated NER datasets. We also provide publicly accessible datasets and resources, supporting further research in threat intelligence and AI model developments.

1 INTRODUCTION

Threat intelligence, also known as entity recognition (TI-NER), is a specialized NLP task in the cybersecurity and threat intelligence domain. It identifies and classifies cybersecurity-related entities within unstructured text reports, such as malware, threat actors, indicators of compromise (IoCs), security tools, and vulnerabilities. Although manual analysis by security analysts is precise, the large volume and varied sources of daily threat reports make this approach impractical. To address this, researchers have turned to machine learning (ML) models to automate the extraction of actionable intelligence from these reports. Examples of such tools include AGIR (Perrina et al., 2023), TTPHunter (Rani et al., 2023a), Vulcan(Jo et al., 2022), AttackKG(Li et al., 2022), CyberRel(Guo et al., 2021), EXTRACTOR(Kiavash et al., 2021), and CyberEntRel(Ahmed et al., 2024). These deep learning tools depend on high-quality annotated datasets. In cybersecurity, the most common tagging schemes for annotating these entities in text sequences are BIO (beginning, inside, or outside) and BIOES (beginning, inside, outside, end, or single), although there is limited research on the costs of choosing one scheme over another. The effectiveness of TI-NER is further underscored by its integration with the Structured Threat Information eXpression (STIX) framework like STIX-2.1 [(Jordan et al., 2022)]. STIX organizes extracted entities in a standardized format, facilitating data sharing and analysis. It includes at least 19 entity types or STIX domain objects (SDOs) and 18 STIX cyber-observable objects (SCOs) or artifacts, each of which represents a unique entity commonly found in cyber threat intelligence (CTI) datasets.

However, a significant challenge in developing a dynamic and effective TI-NER AI model for real-world use is the scarcity of suitably scalable labelled datasets. These datasets need to be well-annotated and readily accessible to facilitate progress in the field. Additionally, they should encompass a diverse range of entity categories with many instances per category and include a substantial volume of tokens (Wang et al., 2020c). Additionally, adherence to the widely adopted STIX 2.1 specifications is essential, as these serve as a standard format for TI data exchange among security firms. Data augmentation (DA), by

Mouiche, I. and Saad, S.
TI-NERmerger: Semi-Automated Framework for Integrating NER Datasets in Cybersecurity.
DOI: 10.5220/0012867900003767

In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, pages 357-370
ISBN: 978-989-758-709-2; ISSN: 2184-7711

merging existing annotated NER datasets, offers a potential solution to this challenge. DA is the process of generating new data from existing data. Robust ML models require large and varied datasets for initial training, but sourcing sufficiently diverse real-world datasets can be challenging because of data silos, regulations, and other limitations (Ding et al., 2024; Zhou et al., 2020). While various DA techniques are available in the literature, it's important to note that they do not always guarantee improved dataset quality or subsequent model performance (Lin et al., 2024; Bakır et al., 2024). The DA approach consisting of integrating two or more NER datasets in cybersecurity presents several challenges, including inconsistencies in tagging schemes, number of labels, label names, and compliance with standards like STIX2.1. Merging NER datasets without addressing these issues degrades the model's performance. Manual merging processes are time-consuming and cumbersome, akin to re-annotating each dataset manually.

To address these challenges, this study introduces TI-NERmerger, a semi-automated framework for merging TI-NER datasets. Our framework streamlines the integration process, significantly reducing manual effort. Experimental results using two prominent open-source TI-NER datasets, DNRTI(Wang et al., 2020c) and APTNER(Wang et al., 2022), demonstrate that our framework saves over 94% of manual work, which would typically take several months, in just a few minutes.

Our key contributions can be summarized as follows:

- We introduced TI-NERmerger, a semi-automated framework designed to integrate threat intelligence NER datasets. A case example involving open-source NER datasets such as DNRTI and APTNER illustrates the framework's effectiveness and performance.
- We curated the DNRTI-STIX NER dataset, comprising $175,354$ tokens, $39,435$ labeled entities, and $6,580$ sentences. This dataset adheres to the STIX 2.1 data exchange standard.
- We created the curated 2APTNER dataset by merging DNRTI-STIX and APTNER. This more extensive augmented dataset contains $434,150$ tokens, $79,161$ labelled security entities, and $16,691$ sentences. It offers greater scalability than existing datasets and complies with the STIX 2.1 standard, establishing itself as the premier dataset for building robust NER AI models.
- We implemented deep learning models, including BiLSTM and BERT, to demonstrate the effectiveness of the curated DNRTI-STIX and 2APTNER datasets for TI-NER.

```
admin@338 B-HackOrg          Aquatic B-APT
uses O                       Panda E-APT
Poison B-Tool                leverages O
Ivy I-Tool                   Cobalt B-MAL
, O                          strike E-MAL
LaZagne I-Tool               , O
, O                          LaZagne B-TOOL
and O                        , O
Cobalt B-Tool                and O
strike I-Tool                njRAT B-MAL
to O                         to O
target O                     target O
financial B-Org              military B-IDTY
organizations I-Org          industries E-IDTY
in O                         in O
Westen B-Area                Hong B-LOC
China I-Area                 Kong E-LOC
; O                          and O
it O                         exfiltrate B-ACT
exfiltrates B-Purp            data E-ACT
data I-Purp                  over O
via O                        0.0.0.0 B-IP
1.1.1.1 O                    address

    A)                           B)
```

Figure 1: Sample threat information in BIO(A) and BIOES(B) format.

- To promote research in this field, in addition to the TI-NERmerger framework, we will make both the DNRTI-STIX and 2APTNER datasets available through our GitHub repository, accessible via the following link [1].

The paper proceeds as follows: Section 2 discusses the challenges motivating this study, Section 3 reviews previous research efforts, Section 4 outlines the methodology for merging TI-NER datasets, Section 5 introduces the TI-NERmerger framework, and Section 6 concludes the paper, summarizing key findings and contributions.

2 PROBLEM DEFINITION

We illustrate the research problem using a simple case example in Figure 1, which mirrors a real-world scenario. In this example, A and B represent two annotated TI-NER datasets collected from different sources. The objective is to merge these datasets into a single consolidated dataset suitable for training a robust AI model.

The analysis of datasets A and B reveals the following challenges:

1. Tagging schemes: Dataset A utilizes the BIO tagging scheme, whereas Dataset B employs the

[1] https://github.com/imouiche/TI-NERmerger

BIOES tagging scheme.

2. Label names and entity categories: Dataset A includes label names such as HackOrg, Tool, Org, Area, and Purp, while dataset B uses labels such as APT, MAL, TOOL, IDTY, LOC, ACT, and IP. This also highlights the difference in the number of entity types between A and B.
3. Annotation: There is inconsistency in entity annotation between the datasets. For example, "Cobalt Strike" labelled as B-Tool I-Tool in dataset A is annotated as B-MAL E-MAL, indicating it as malware instead of a tool like in A. Another inconsistency is between "financial organizations" labelled as B-Org I-Org in dataset A and "military industries" labelled as B-IDTY E-IDTY in dataset B. Both entity types ("Org" and "IDTY") identify the object being targeted by hackers or malware. The only difference is that "Org" is more specific.
4. Uncovered entities: Dataset B includes low-level indicators of compromise (IoCs), such as IP addresses, which are neglected in Dataset A.

Integrating datasets A and B without addressing these challenges will degrade the model's performance. While the manual process can be completed within minutes if A and B only contain one sentence each, real-world datasets like DNRTI(Wang et al., 2020c) and APTNER(Wang et al., 2022) contain tens of thousands of sentences, which makes the manual approach cumbersome and even intractable at large scale. In addition, datasets contain entities that span several tokens, making their identification and extraction more complex. This paper aims to alleviate these challenges by transitioning from the manual process to a semi-automated one, taking advantage of the fact that these datasets are already annotated and come from the same domain.

3 RELATED WORKS

Previous literature lacks any work explicitly targeting the development of a framework for merging TI-NER datasets. Given that this paper also aims to release suitably annotated NER datasets compliant with cybersecurity data exchange standards like STIX-2.1 (Jordan et al., 2022), we will review previous efforts in this direction to provide research context. (Zhou et al., 2018) conducted a comprehensive study in which they crawled 687 Advanced Persistent Threat (APT) reports published between 2008 and 2018. They then annotated 370 articles, focusing on 11 predefined indicators of compromise (IoC) entity types. (YI et al., 2020) introduced a novel NER approach called RDF-CRF, which combines regular expressions, a dictionary of known entities, and the conditional random field (CRF) algorithm. To evaluate the model, they created a NER dataset using $14,000$ web security reports, encompassing 22 predefined entity categories and featuring $7,413$ labelled entities. (Kim et al., 2020) designed a NER system that leveraged the character-level feature vector to detect cyber threats within unstructured text reports. To evaluate the performance of their model, they constructed a corpus that contained $498,000$ entity tags and 11 cyber keywords or entity names. (Guo et al., 2021) gathered security reports from diverse CTI sources, including APT reports, hacker forums, security bulletins, and more. They created a dataset named OSINT, consisting of 13,000 sentences, to assess the capabilities of CyberRel, a model designed for the simultaneous extraction of entities and relationships from security reports. (Marchiori et al., 2023) introduced the STIXnet model, which employs rule-based methods, NLP, and deep learning techniques to extract 18 STIX entities and relationships within security reports. As part of their work, the authors made available a sample of annotated APT groups, which they gathered by crawling data from the MITRE ATT&CK repository (Corporation, 2023).

Previous attempts to address the lack of large-scale and high-quality annotated NER datasets in cybersecurity have not gone unnoticed. Table 1 summarizes advancements in the NER domain in a comparative study. It is essential to highlight that, at present, all the annotated datasets mentioned are not publicly accessible except for DNRTI (Wang et al., 2020c) and APTNER (Wang et al., 2022). DNRTI covers only 13 entity categories and does not conform to the STIX 2.1 specification for sharing cyber threat intelligence (CTI) information (Wang et al., 2022). DNRTI-STIX is a newly generated TI-NER dataset that adheres to the STIX 2.1 standard. The integration of DNRTI-STIX and APTNER results in the augmented APTNER, also known as 2APTNER. The 2APTNER dataset surpasses existing datasets in terms of the number of tokens, annotated entities, and sentences, establishing itself as the largest NER dataset in the field of threat intelligence.

Table 1: The DNRTI-STIX2 and 2APTNER Datasets and their Comparison with Existing TI-NER Datasets.

Datasets	Open	# of entity types	# of tokens	# of labeled ents.	# of sents.	vocab size	# of Reports
(Zhou et al., 2018)	☒	11	1773638	69032	-	-	390
(YI et al., 2020)	☒	23	-	7413	-	-	14128
(Kim et al., 2020)	☒	11	498000	15720	13570	-	160
(Guo et al., 2021)	☒	-	-	75990	13000	-	-
(Marchiori et al., 2023)	☒	18	-	-	-	-	-
(Wang et al., 2020c)	☑	13	175461	36808	6592	9426	-
(Wang et al., 2022)	☑	21	258796	39726	10111	15608	-
DNRTI-STIX2	☑	21	175354	39435	6580	9444	-
2APTNER	☑	21	434150	79161	16691	16439	-

4 METHODOLOGY FOR INTEGRATING TI-NER DATASETS

This section outlines the step-by-step procedure followed in this paper for merging labelled TI-NER datasets in cybersecurity. After defining the datasets, the methodology comprises four main phases: Tag Representation, Entity Categories, Entity Mappings, and Annotation. The paper begins with a manual approach to establish the baseline for developing the automation framework known as Ti-NERmerger.

4.1 Datasets

The two datasets utilized for the experiment are DNRTI(Wang et al., 2020c) and APTNER(Wang et al., 2022), sourced from their respective repositories [(Wang et al., 2020b), (Wang et al., 2020a)]. We combined the training, testing, and validation sets into a unified dataset for each dataset. We conducted preprocessing to eliminate non-ASCII characters and incomplete sentences, and the resulting distribution of the number of sentences, labelled entities, and vocabulary size can be found in Table 1. The objective is to merge these datasets to create a more scalable annotated dataset for building robust NER AI systems. In this case, the resulting dataset is called augmented APTNER or simply 2APTNER.

The definitions and examples of each entity type are provided in Table 2. Additionally, the

4.2 Tag Representation

The goal here is to select the tagging scheme for the resulting dataset (2APTNER). DNRTI is labelled using the BIO (beginning, inside, or outside) scheme, while APTNER employs BIOES (beginning, inside, outside, end, or single). Since we only have two datasets, choosing between BIO and BIOES is optimal. To maintain simplicity and leverage the data granularity provided by BIOES, we opted for this format and this decision addresses the issue (1) stated in Section 2.

4.3 Entity Categories

This step tackles challenge (2) of the problem definition in Section 2 by specifying the entity categories for the target dataset (2APTNER). The DNRTI dataset comprises 13 entity types: *HackOrg*, *OffAct*, *SamFile*, *SecTeam*, *Time*, *Way*, *Tool, Idus, Org, Area, Purp*, and *Features*. In contrast, the APTNER dataset features 21 entity categories, including *APT, SECTEAM, LOC, TIME, VULNAME, VULID, TOOL, MAL, FILE, MD5, SHA1, SHA2, IDTY, ACT, DOM, ENCR, EMAIL, OS, PROT, URL*, and *IP*. Given that APTNER complies with the STIX 2.1 standard for data exchange, using its entity types ensures alignment with this standard. Therefore, utilizing APTNER as the base dataset and converting DNRTI to align with APTNER for seamless integration is beneficial.

4.4 Entity Mappings

This step involves defining possible entity mappings when aligning two datasets. Entity mappings elucidate the types of relationships that exist between entity types in different datasets. Once entity categories for the resulting or target dataset have been defined, up to four possible entity mappings can be distinguished. Due to this finite number, it becomes feasible to semi-automate the process. For DNRTI and APTNER, the four established mappings are illustrated with examples in Table 2.

1. 1-to-1 Mappings indicate a direct mapping between DNRTI and APTNER entities.
2. 1-to-many Mappings: they show the DNRTI entities or categories that were expanded into two or more APTNER features.
3. many-to-1 Mapping: as a reverse of 1-to-many mappings, they present those DNRTI entities merged into a single APTNER entity.
4. Uncovered Entities: this section introduces additional entities similar to APTNER, not initially included in the original DNRTI article but uncovered during the annotation process while converting DNRTI to align with APTNER, i.e. with 21 entity types. It is important to note that this mapping is optional as one may decide only to consider initially annotated entities.

The primary objective of this phase is to establish a foundation for seamless manual and automated harmonization of datasets.

4.5 Annotations or Alignments

This phase aims to tackle challenges (3) and (4) from Section 2. To resolve the inconsistency issue in entity annotation between both datasets, it is crucial to have a reliable reference source of truth. We relied on the MITRE ATT&CK framework (Corporation, 2023) as our primary point of reference to determine the correct entity types. The MITRE ATT&CK framework is a knowledge base of adversary tactics and techniques based on real-world observations. It is widely used in cybersecurity for threat intelligence, threat hunting, and incident response purposes. The example provided in Figure 1, utilizing the MITRE repository, highlights that "Cobalt Strike" in Dataset A should be classified as part of the Malware class rather than a Tool, thus offering enhanced precision in addressing inconsistency for a coherent integration. Addressing the challenge (4) is important but not mandatory. It involves identifying entities that were not initially included in the original dataset. The case example shown in Table 2 entails discovering entities such as DOM, ENCR, EMAIL, OS, PROT, URL, and IP in the DNRTI dataset. It is important as it helps increase the number of instances of these classes in the target dataset, thereby enhancing classification accuracy.

```
Similar O                     Similar O
to O                          to O
RIPTIDE B-OffAct              RIPTIDE B-ACT
campaigns I-OffAct            campaigns E-ACT
, O                           , O
APT12 B-HackOrg               APT12 S-APT
infects O                     infects S-ACT
target O                      target O
systems O                     systems O
with O                        with O
HIGHTIDE B-Tool               HIGHTIDE S-MAL
using O                       using O
a O                           a O
Microsoft B-Tool              Microsoft B-FILE
Word I-Tool                   Word E-FILE
( O                           ( O
.doc B-Tool                   .doc S-FILE
) O                           ) O
document O                    document O
that O                        that O
exploits O                    exploits O
CVE-2012-0158 B-Exp           CVE-2012-0158 S-VULID
. O                           . O
    a)                          b)
```

Figure 2: Sample conversion of DNRTI (a) to DNRTI-STIX(b).

After completing the analysis phases, the manual relabeling of DNRTI using BIOES format and the 21 predefined entity categories of the target dataset was initiated. This process involved four annotators: one PhD student and three master's students, all from a cybersecurity background. The process began with two one-hour meetings coordinated by the PhD student. During the first meeting, 25 sentences were re-labeled to serve as examples. At the end of the meeting, each student selected 10 sentences to annotate for the next meeting. In the second meeting, all 40 sentences were reviewed for better understanding. Subsequently, the remaining DNRTI sentences were distributed among all annotators, with the PhD student receiving 40% and each master's student receiving 20%. Annotators collaborated to address any confusion that arose during the annotation process and the consensus was obtained through a majority vote. The voting weight was distributed such that the PhD student's vote counted for 40%, while each master's student's vote counted for 20%. This distribution of voting power effectively resolved any tie situations. The manual process to align DNRTI with APTNER, ensuring adherence to the STIX 2.1 specification, took three months to complete. The resulting dataset, named DNRTI-STIX, will seamlessly merge with APTNER to create 2APTNER. This combined dataset offers a more scalable annotated TI-NER dataset for building reliable AI systems. A sample conversion of DNRTI to DNRTI-STIX is shown in Figure 2. For instance, the named entity "HIGHTIDE" initially labeled as "Tool" is changed to "Malware" according to the MITRE ATT&CK repository. Similarly, "Microsoft Word .doc" classified initially as "Tool" (i.e., "B-Tool I-Tool B-Tool"), becomes "B-FILE E-FILE S-FILE" after conversion.

Table 2: Entity Mappings aligning DNRTI with APTNER and STIX 2.1.

DNRTI Entities	APTNER Entities	STIX-2.1	Examples
1-to-1 Mappings			
HackOrg	APT	Threat groups	APT19, admin@338, MuddyWater
SecTeam	SECTEAM	Security teams	FireEye, MATI, Palo Alto Networks
Area	LOC	Location	China, Russia, North Korea
Time	TIME	Time	Sept 10, April 9th, 2016
1-to-many Mappings			
Exp	VULNAME	Exploit	EternalBlue, zero-day
	VULID	Vulnerability ID	CVE-2017-8759, CVE-2016-4117
Tool	TOOL	Tool	PowerShell, LaZagne
	MAL	Malware	SHIRIME, FinSpy, Clayslide
SamFile	MAL	Malware	Backdoor.APT.FakeWinHTTPHelper
	FILE	File	checker1.exe, .docs, Excel worksheets
	MD5	Hash value	12hj34ng34ghjdf802n3inf
	SHA1	Hash value	AA0FA4584768CE9E16D67D8C520...
	SHA2	Hash value	cca268c13885ad5751eb70371bbc9ce8c...
many-to-1 Mappings			
Idus	IDTY	Identity,	Military Industry, Financial Institutes
Org		Industry	Google, Technology organizations
OffAct	ACT	Attack patterns	Spear-phishing
Way		Attack patterns	Brute force
Purp		Attack patterns	Exfiltration, DoS
Features		Attack patterns	Lateral movement
Uncovered Entities			
-	DOM	Domain	adobe.com, mydomain1607.com
-	ENCR	Encryption methods	RSA, AES
-	EMAIL	Email	edmundj@chmail.ir, hostay88@gmail.com
-	OS	Operating system	Windows, Linux
-	PROT	Protocol	ssh, HTTP, POP3
-	URL	URL	https://github.com
-	IP	IP address	185.162.235.0, 0.0.0.0

4.6 Integration and Results

As shown in Table 3, the conversion of the DNRTI to DNRTI-STIX from using fine-grained BIOES format resulted in a total of $39,435$ labelled entities, adding $2,625$ entities to the original DNRTI. A slight reduction in the number of tokens and sentences for DNRTI-STIX can be observed, and this is primarily attributed to the removal of noisy data, including non-ASCII characters and incomplete sentences, during the migration process. Additionally, DNRTI-STIX features 21 entity categories that are the same as APTNER and, therefore, can be merged with no issues. Their integration gives rise to the 2APTNER dataset, which is more expansive and encompasses $434,150$ tokens, $79,161$ labelled security entities, and $16,691$ sentences. It provides increased scalability compared to existing datasets and adheres to the STIX 2.1 standard, solidifying its position for building real-world AI systems.

Figure 3 provides a visual representation of different class distributions that distinguish 2APTNER as the most scalable TI-NER dataset when compared to DNRTI-STIX and the leading APTNER. The labelling quality and effectiveness of the resulting datasets (DNRTI-STIX and 2APTNER) are assessed in the following sections.

4.7 Evaluations and Discussions

This section evaluates the quality and effectiveness of the curated DNRTI-STIX and 2APTNER datasets resulting from the manual labelling process. Various state-of-the-art (SOTA) algorithms in the literature

Table 3: Curated DNRTI-STIX and 2APTNER Datasets from Manual Approach.

Datasets	# ents type	# of tokens	# of labeled ents.	# of sents.	vocab size
DNRTI	13	175461	36808	6592	9426
DNRTI-STIX	**21**	**175354**	**39435**	**6580**	**9444**
APTNER	21	258796	39726	10111	15608
2APTNER	**21**	**434150**	**79161**	**16691**	**16439**

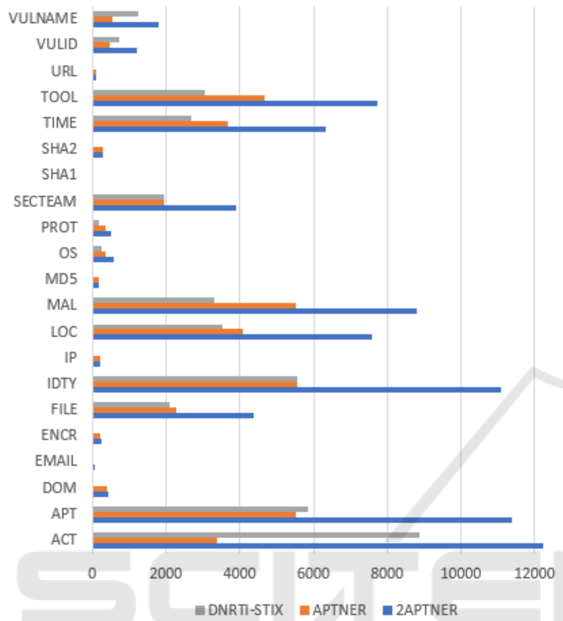

Figure 3: Comparison of DNRTI-STIX, APTNER, and 2APTNER.

have demonstrated significant performance on NER tasks. For this study, we implement one recurrent neural network (RNN)-based architecture like BiLSTM and one transformer-based model like BERT. The primary objective is to evaluate the effectiveness of the datasets rather than focusing on the performance of the models. Both BiLSTM and BERT have bidirectional capabilities, allowing them to capture information from past and future contexts, significantly enhancing their ability to comprehend the overall context of a sequence. This quality contributes to their impressive performance in NER tasks [(Huang et al., 2015), (Zhou et al., 2021), (Varghese et al., 2023), (Wang et al., 2020a), (Devlin et al., 2019)]. Moreover, BERT undergoes pre-training on an extensive corpus of text data before fine-tuning for specific downstream tasks, employing an attention mechanism to consider the entire context of a word within a sentence. BERT has gained prominence, particularly for its transformer architecture, which excels in capturing long-range dependencies. This study implements the base forms of these models to demonstrate the effectiveness of our datasets.

4.7.1 DNRTI-STIX vs DNRTI

The hyperparameters for each base model used in the experiment are detailed in Table 4. A Dropout of 0.2 was applied with BiLSTM to prevent overfitting. The datasets were split into training, test, and validation sets in a ratio of 7:1.5:1.5 for both models.

Table 4: Models' parameter settings.

parameters	**BERT**	**BiLSTM**
batch size	8	16
dropout	0.5	0.5
learning rate	1e-5	1e-5
epsilon	1e-6	1e-6
weight decay	0.001	0.001
hidden layer size	100	-
optimizer	Adam	Adam
embedding size	768	300
number of epochs	1	10

Table 5 provides a comparative summary of DNRTI-STIX and DNRT datasets for BiLSTM and BERT models. DNRTI-STIX features more unique entity tags (60) than DNRTI (27) due to the conversion of DNRTI to 21 entity categories aligned with the STIX standard. Despite covering a broader range of entity categories and exhibiting more diversity in entity types, DNRTI-STIX maintains relatively similar performance to DNRTI and even slightly outperforms it in terms of Precision (P), Recall (R), and F1 scores (F1) for both BiLSTM and BERT models. This highlights the quality of the manual relabeling process undertaken by the authors. As expected, the BERT model achieves higher Precision, Recall, and F1 scores compared to the BiLSTM model for both datasets, indicating its superior performance. For this reason, we used the BERT model to report the individual class classification for both datasets in Table 6.

This table comprehensively shows how effectively the model predicts each class, presenting per-class and overall performance metrics, including Micro Avg, Macro Avg, and Weighted Avg.

- The Micro Avg row represents the weighted average of precision, recall, and F1-Score across all classes, considering individual predictions and

Table 5: DNRTI-STIX vs DNRTI using BiLSTM and BERT models.

	DNRTI-STIX			DNRTI		
# unique entity tags	60			27		
Metrics	P	R	F1	P	R	F1
BiLSTM-CRF	0.68	0.70	0.69	0.67	0.70	0.68
BERT	**0.79**	**0.84**	**0.81**	0.77	0.82	0.80

support for each instance.

- The Macro Avg row displays the unweighted average of precision, recall, and F1-Score across all classes, treating all classes equally without considering class imbalances.
- The Weighted Avg rows provide a weighted average of precision, recall, and F1-Score, with each class's contribution weighted by its support after the split.

It's important to note that specific entity classes, such as *SHA1*, and *URL*, are not included in the report. This omission is due to the insufficient number of instances for these classes in DNRTI-STIX, as seen in Fig 3, and they were not considered during training and evaluation. Simultaneously, Table 6 presents the classification report for the BERT model on the original DNRTI dataset. This not only highlights the unique characteristics of the STIX 2.1 format in extracting more detailed entity information from TI-NER datasets but also underscores the quality of data relabeling done by the authors, maintaining the model's performance relatively high despite the increased number of entity categories in DNRTI-STIX.

4.7.2 DNRTI-STIX VS APTNER vs 2APTNER

The Augmented APTNER also known as 2APTNER is formed by merging the DNRTI-STIX and APTNER datasets. Table 7 displays the classification report for these datasets using BiLSTM and BERT models. DNRTI-STIX demonstrates superior performance compared to APTNER and 2APTNER. Previous studies have shown that BiLSTM struggles with larger datasets due to memory requirements. This is also seen in Table 7, while BERT consistently outperforms across all datasets.

Upon reviewing the BERT classification reports in Table 8 for both APTNER and 2APTNER datasets, it becomes evident that the "SHA1" entity class hurts 2APTNER with a contribution of 0.00. This is because the count of instances of this class was less than 50, and machine learning (ML) models typically require a minimum of 50 samples to understand the context within a sentence [(Rani et al., 2023b), (Pedregosa et al., 2011)]. However, the contribution of "SHA1" and "URL" was ignored for DNRTI-STIX. Thus, it is understood that DNRTI-STIX and 2APTNER perform similarly using the BERT model.

This concludes the demonstration of high-quality annotation and efficiency of the DNRTI-STIX and 2APTNER datasets obtained through the manual relabeling approach using the BiLSTM and BERT base form models. These datasets will serve as baselines for evaluating the TI-NERmerger framework proposed in this study, aimed aA common approach in data-centric AI is data augmentation or argumentation to meet these requirementsre already annotated and belong to the same domain, the framework aims to automate and optimize the dataset integration process.

5 TI-NERmerger: A SEMI-AUTOMATED FRAMEWORK FOR INTEGRATING NER DATASETS IN CYBERSECURITY: A CASE STUDY OF DNRTI AND APTNER

With the rise of data-centric AI and the emergence of Large Language Models (LLMs) like BERT, GPT-3, RoBERTa, and others, the importance of high-quality, scalable, and diverse datasets for training robust AI systems has become increasingly apparent. To meet these requirements, a common approach in data-centric AI is data augmentation or argumentation. This involves merging multiple open-source annotated datasets into a single, consolidated, and diverse dataset with the aim of significantly improving the resulting AI systems. However, integrating threat intelligence named entity recognition (TI-NER) datasets poses several challenges, as outlined in Section 2. These challenges include using different tagging formats, entity types, and inconsistency in entity annotation. The manual process to address these issues and align datasets for integration is time-consuming and becomes increasingly difficult when dealing with numerous datasets.

This section introduces TI-NERmerger, a semi-automated framework designed for merging TI-NER datasets. Leveraging that these datasets originate from the same domain and are already annotated for NER tasks, TI-NERmerger facilitates the transition from the current manual approach to a semi-

Table 6: DNRTI-STIX vs DNRTI Classification Report using the BERT Model.

	DNRTI-STIX			DNRTI			
Class	P	R	F1	**Class**	P	R	F1
ACT	0.72	0.80	0.76	Area	0.85	0.93	0.89
APT	0.80	0.88	0.84	Exp	0.96	0.98	0.97
DOM	1.00	0.80	0.89	Features	0.73	0.83	0.78
EMAIL	0.80	1.00	0.89	HackOrg	0.78	0.83	0.81
ENCR	0.75	0.60	0.67	Idus	0.79	0.80	0.79
FILE	0.78	0.89	0.83	OffAct	0.71	0.84	0.77
IDTY	0.78	0.81	0.79	Org	0.65	0.68	0.66
IP	0.67	1.00	0.80	Purp	0.63	0.74	0.68
LOC	0.85	0.91	0.88	SamFile	0.81	0.81	0.81
MAL	0.79	0.83	0.81	SecTeam	0.88	0.87	0.88
MD5	1.00	1.00	1.00	Time	0.87	0.91	0.89
OS	0.84	0.95	0.89	Tool	0.68	0.77	0.72
PROT	0.94	0.64	0.76	Way	0.73	0.64	0.68
SECTEAM	0.87	0.87	0.87				
SHA2	1.00	1.00	1.00				
TIME	0.85	0.90	0.87				
TOOL	0.70	0.71	0.70				
VULID	1.00	0.99	1.00				
VULNAME	0.86	0.90	0.88				
Micro Avg	0.78	0.84	0.81	Micro Avg	0.77	0.82	0.80
Macro Avg	0.84	0.87	0.85	Macro Avg	0.77	0.82	0.79
Weighted Avg	0.79	0.84	0.81	Weighted Avg	0.77	0.82	0.80

Table 7: DNRTI-STIX VS APTNER vs 2APTNER Classification Report using BiLSTM and BERT.

		DNRTI-STIX			APTNER			2APTNER		
Model	Metrics	P	R	F1	P	R	F1	P	R	F1
BiLSTM-CRF	Micro Avg	0.65	o.74	0.69	0.65	0.61	0.63	0.60	0.61	0.63
	Macro Avg	0.56	0.50	0.54	0.45	0.47	0.46	0.41	0.46	0.42
	Weighted Avg	0.66	0.74	0.69	0.59	0.61	0.60	0.60	0.60	0.60
BERT	Micro Avg	**0.78**	**0.84**	**0.81**	0.73	0.78	0.76	0.76	0.80	0.78
	Macro Avg	**0.84**	**0.87**	**0.85**	0.77	0.79	0.78	0.81	0.82	0.81
	Weighted Avg	**0.79**	**0.84**	**0.81**	0.73	0.78	0.76	0.76	0.80	0.78

automated one. This transition allows the annotation task, which typically takes several months, to be completed in just a few minutes.

Figure 4 illustrates the framework, which comprises four main components leading to the formation of the target or merged dataset. The framework is inspired by the manual process of integrating DNRTI and APTNER, as outlined in Table 2. The four components are classified into two phases: Analysis and Automation. The analysis phase includes Tag Representation, Entity Categories, and Entity Mappings. The automation phase involves annotation or alignment and integration into the target dataset. As depicted in Figure 4, the framework claims the capability of merging multiple datasets (denoted as *n*). In practice, this is achieved by merging two datasets at a time, and then the resulting dataset is merged with the next dataset in the sequence.

For clarity, we maintain the example of DNRTI and APTNER to describe the four components:

1. **Tag Representation.**
   Select the tagging scheme for the target dataset, such as BIO or BIOES. This decision should be influenced by the specific requirements of the NER dataset, the dataset's characteristics, and the desired level of detail in entity recognition ((Konkol and Konopík, 2015), (Alshammari and Alanazi, 2021)). The framework implements only these two tagging formats because almost all NER datasets in security use one of these formats. To repeat the experiment using our tool for the case of DNRTI and APTNER, we have chosen BIOES.

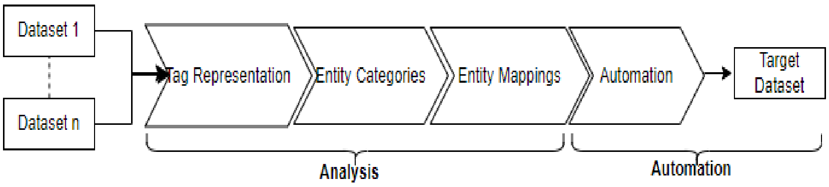

Figure 4: TI-NERmerger: Semi-automation framework to integrate NER datasets in cybersecurity.

Table 8: APTNER and 2APTNER Classification Reports using the BERT model.

Class	APTNER			2APTNER		
	P	R	F1	P	R	F1
ACT	0.56	0.68	0.62	0.62	0.68	0.65
APT	0.82	0.88	0.85	0.82	0.86	0.86
DOM	0.93	0.90	0.92	0.83	0.98	0.92
EMAIL	0.67	0.56	0.61	1.00	0.73	0.84
ENCR	0.85	0.92	0.89	0.76	0.85	0.80
FILE	0.72	0.75	0.74	0.77	0.74	0.76
IDTY	0.71	0.82	0.76	0.68	0.74	0.78
IP	0.94	0.94	0.94	0.94	0.97	0.96
LOC	0.88	0.91	0.89	0.84	0.89	0.86
MAL	0.71	0.72	0.72	0.72	0.80	0.76
MD5	0.69	0.63	0.66	0.93	0.97	0.95
OS	0.77	0.78	0.77	0.83	0.86	0.85
PROT	0.72	0.77	0.74	0.70	0.79	0.74
SECTEAM	0.87	0.89	0.88	0.80	0.82	0.81
SHA1	-	-	-	0.0	0.0	0.0
SHA2	0.77	0.97	0.86	0.98	1.00	0.99
TIME	0.87	0.91	0.89	0.77	0.82	0.79
TOOL	0.53	0.57	0.55	0.72	0.74	0.73
URL	0.89	0.71	0.79	0.78	0.50	0.61
VULID	1.00	0.99	0.99	1.00	1.00	1.00
VULNAME	0.52	0.51	0.51	0.76	0.76	0.76

This choice is based on its previous use in representing the APTNER dataset, which was larger and contained 21 entity categories, and it also adheres to the STIX standard.

2. **Entity Categories.**
   Thoroughly analyze the entity categories in each dataset and predefined the entity types for the final dataset. This task should be carried out by a domain expert who understands the specific requirements of the NER tasks. In the case example, the authors opted for the 21 predefined APTNER entity categories for the target dataset.

3. **Entity Mappings.**
   Establish distinct mappings between each dataset and the predefined entity types of the target dataset. Possible mappings include 1-to-1 mappings, 1-to-many mappings, many-to-1 mappings, and the discovery module. Refer to Table 2 for a visual representation of the four different mappings defined for the case of DNRTI and APTNER.

4. **Annotation or Alignment.**
   This component involves translating the different mappings established in the preceding phase into a programming language. The TI-NERmerger framework is implemented in Python and comprises six main modules, each of which can work and be executed independently. The first module reads the command-line inputs, while the next four modules implement each identified mapping: 1-to-1 mappings, 1-to-many mappings, many-to-1 mappings, and uncovered entities. Finally, the last module performs the merge and outputs the result. The component is approached as an active annotation initiative, with a domain expert in the loop who decides which module to run and provides the required entity classes. For the case example of DNRTI and APTNER:

   (a) The first module reads user inputs from the command line and applies any tagging conversion if needed. For example, a command-line input of "TI-NERmerger BIOES DNRTI APTNER 2APTNER" means the user wants to merge DNRTI and APTNER into a single dataset called 2APTNER using BIOES tagging. This module will automatically detect which DNRTI and APTNER does not align with this format and make the necessary conversion. In this case, DNRTI will be changed from BIO to BIOES.

   (b) The 1-to-1 mappings module assigns new labels, namely *APT*, *SECTEAM*, *LOC*, and *TIME*, to all DNRTI entities with labels *HackOrg*, *SecTeam*, *Area*, and *Time*, respectively.

   (c) The many-to-1 mappings module merges all DNRTI entities tagged as *Idus* and *Org* into *IDTY*; and all DNRTI entities annotated as *OffAct*, *Way*, *Purp*, and *Features* into *ACT*.

   (d) The 1-to-many mappings module implements an algorithm that uses Python Scrapy library to query ATT&CK repository(Corporation, 2023) and categorize all DNRTI entities labelled as *Tool* into either *TOOL* or *MAL* (malware). It defaults to *TOOL* if the software is not found on the MITRE platform. It works similar to the manual process to address inconsistency in entity annotation. This module employs regular expressions (regex) to parse all *SamFile* entities into *MAL*, *FILE*, *MD5*, *SHA1*, and *SHA2*. Similarly, it categorizes all *Exp* entities into *VULNAME* and *VULID* using regex. It's important to note that this module can also identify hacker groups defined in the ATT&CK repository.

(e) The discovery module reveals indicators of compromise (IoCs) such as *IP*, *URL*, *DOM*, *EMAIL*, and *PROT* from the DNRTI that were not originally considered. This was necessary to augment the number of instances in the target dataset, as these entities were annotated in APTNER. The module also identifies encryption algorithms (*ENCR*) and operating systems (*OS*) in DNRTI by matching unlabeled entities with pre-defined lists of encryptions and operating systems.

(f) The integration module merges both datasets, combining DNRTI and APTNER into a single dataset called 2APTNER.

This explains how we successfully completed the annotation task that took several months in just a few minutes.

The code of the whole TI-NERmerger framework spans over 1000+ lines, and we plan to release it along with various datasets to support continuous development and improvement.

5.1 Results and Discussions

To evaluate the effectiveness of the TI-NERmerger framework, we employed it to align the original DNRTI with APTNER, resulting in DNRTI-STIX. We compared the results with the manual process in Table 9. It is important to note that APTNER remained unchanged during the integration process until the final stage, where it was merged with DNRTI-STIX. APTNER was selected as the baseline for the final dataset because it adheres to the STIX-2.1 data exchange standard and offers a diverse range of entity categories. Consequently, inconsistencies were resolved by aligning DNRTI-STIX with APTNER. For example, as shown in Figure 1, the "financial organization" in dataset A, initially tagged as B-Org I-Org, was mapped to B-IDTY I-IDTY to align with dataset B during the merging process. In dataset B, the label category "IDTY" is used to identify the object or victim entity targeted by malware and hacker organizations.

Table 9 indicates that the framework successfully extended DNRTI of 13 entity types to DNRTI-STIX featuring 21 entity categories. Both the manual process and the TI-NERmerger framework resulted in datasets with the same number of entity types (21). The total number of tokens in the datasets is almost the same for both approaches, with a slight difference of 111 tokens. This is because noisy words and incomplete sentences were removed during the manual approach (hence $6,580$ for the manual process and $6,592$ for the TI-NERmerger framework).

We observe more labelled entities $(39,435)$ resulting from the manual process than the TI-NERmerger framework (37,335). This is due to the discovery of entities such as *IP*, *URL*, *DOM*, *EMAIL*, *PROT*, *OS*, and *ENCR* that were not initially considered in the original DNRTI dataset. The framework's discovery module faces challenges in uncovering an operating system or encryption algorithm when the entity name cannot be found in the predefined list of operating systems or encryption methods. Both approaches resulted in datasets with a similar number of sentences $(6,580$ for the manual process and $6,592$ for the TI-NERmerger framework). The vocabulary size of the datasets is also very close, with only a difference of 5 vocabulary items. These results deduce the figures depicted in Table 10 for the 2APTNER dataset, which is the outcome of merging DNRTI-STIX and APTNER. TI-NERmerger framework can produce datasets with comparable characteristics to those obtained through manual processes, demonstrating its effectiveness in automating the dataset integration process. In just a few minutes, the framework successfully accounted for over 94.67% of the annotated entities, a task that had previously taken several months using manual methods. This result could improve further if no new entities are uncovered from the dataset.

5.2 Evaluation and Discussions

The effectiveness of the datasets obtained using our TI-NERmerger framework is displayed in Table 11. The results from the manual approach serve as baselines to evaluate the framework's capability. The TI-NERmerger framework performs similarly to the manual approach, especially regarding Micro and Weighted Averages for both BiLSTM-CRF and BERT models on both datasets.

Table 12 presents the classification reports for each individual entity class for the DNRTI-STIX dataset using the BERT model. We observed only slightly better performance in favour of the manual approach. The absence of a line for the *EMAIL* entity class indicates that the framework did not uncover enough instances of this class. Similarly, the framework successfully identified 53 of 60 unique tags from the manual ap-

Table 9: DNRTI-STIX: Manual process Vs Framework.

Approach	# of ents type	# of tokens	# of labelled ents.	# of sents.	vocab size
Manual	21	175354	39435	6580	9444
TI-NERmerger	21	175465	37335	6592	9439

Table 10: 2APTNER: Manual Approach Vs Framework.

Approach	# of unique tags	# of tokens	# of labeled ents.	# of sents.	vocab size
Manual	21	434150	79161	16691	16439
TI-NERmerger	21	434261	77,061	16,703	16023

proach. As stated earlier, the missing 7 tags result from discovering entities that were initially ignored in the original dataset. This demonstrates that the overall performance of the framework remains above 94%.

5.3 Generalization, Advantages, and Limitations

1. Generalization:
   Our TI-NERmerger framework aligns with the widely adopted STIX-2.1 data exchange standard in cybersecurity, which defines a set of STIX Domain Objects (SDOs) and STIX Cyber-observable Objects (SCOs). Each object corresponds to a unique concept commonly represented in CTI datasets. STIX ensures that organizations can share CTI consistently and machine-readably, encouraging dataset owners to use STIX objects as entity types for different downstream tasks. Although these datasets may utilize different labelling and tagging schemes, their integration is facilitated once they establish mappings between entity categories and STIX baseline objects. The identification of the four possible mappings, outlined in Table 2, is supported by integrating MITRE ATT&CK within the STIX framework. This integration offers a detailed behavioural context that significantly enhances the understanding and differentiation of threat entities. This facilitates the effective alignment of entity categories with the established STIX objects. As a result, our TI-NERmerger framework claims strong generalizability across datasets utilizing STIX objects or a subset of STIX objects to define the entity categories. In other words, their integration is assured as long as CTI data can be mapped to the STIX standard. Conversely, in areas where these standard references are not guaranteed or are inapplicable, establishing mappings across CTI datasets or demonstrating their existence can be challenging.
2. Advantages:
   - The modules are independent of each other, flexible, and extensible, allowing them to be adapted for different purposes.
   - Large datasets, often annotated by groups of students, can be effectively combined using this framework. It ensures quality annotation and consistency when merging datasets from different groups.
   - The framework significantly streamlines labour-intensive work that typically takes several weeks to only a few minutes.
   - Small to medium-sized NER datasets are typically well-annotated, and the tool can be employed to create a scalable and high-quality labelled dataset.
   - It is designed to conform to the STIX 2.1 data sharing standard and functions effectively with datasets encompassing a diverse range of entity categories.
3. Limitation. The TI-NER framework has certain limitations:
   - The framework merges two datasets at a time. In the case of multiple datasets, the result of the first two datasets is merged with the third dataset, and so on. This requires running the model multiple times, assuming each dataset has peculiarities.
   - Despite its capability to uncover artifact entities initially overlooked in original datasets, the framework relies on the MITRE ATT&CK repository as the sole source of truth when classifying high-level security entities such as attack groups, tools and malware.

6 CONCLUSION AND FUTURE WORK

This study introduces TI-NERmerger, a semi-automated framework integrating threat intelligence named entity recognition (TI-NER) datasets in cybersecurity. It serves as a data augmentation tool designed to timely tackle the scarcity of scalable and

Table 11: Manual Approach vs TI-NERmerger: Classification Report using BiLSTM and BERT.

		Manual			TI-NERmerger			Manual			TI-NERmerger		
		DNRTI-STIX			DNRTI-STIX			2APTNER			2APTNER		
Model	Metrics	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BiLSTM-CRF	Micro Avg	0.65	o.74	0.69	0.65	0.61	0.63	0.60	0.61	0.63	0.61	0.62	0.62
	Macro Avg	0.56	0.50	0.54	0.45	0.47	0.46	0.41	0.46	0.42	0.44	0.47	0.45
	Weighted Avg	0.66	0.74	0.69	0.59	0.61	0.60	0.60	0.60	0.60	0.61	0.63	0.62
BERT	Micro Avg	**0.78**	**0.84**	**0.81**	0.80	0.84	0.90	0.76	0.80	0.78	0.75	0.79	0.77
	Macro Avg	**0.84**	**0.87**	**0.85**	0.77	0.80	0.78	0.81	0.82	0.81	0.79	0.80	0.79
	Weighted Avg	**0.79**	**0.84**	**0.81**	0.80	0.84	0.80	0.76	0.80	0.78	0.77	0.79	0.77

Table 12: DNRTI-STIX (Manual process Vs Framework): Classification Reports using the BERT model.

# of unique tags	**Manual**			**TI-NERmerger**		
	60			**53**		
Class	P	R	F1	P	R	F1
ACT	0.72	0.80	0.76	0.78	0.83	0.80
APT	0.80	0.88	0.84	0.80	0.86	0.83
DOM	1.00	0.80	0.89	0.83	1.00	0.91
EMAIL	0.80	1.00	0.89			
ENCR	0.75	0.60	0.67	0.70	0.56	0.68
FILE	0.78	0.89	0.83	0.94	0.95	0.95
IDTY	0.78	0.81	0.79	0.75	0.82	0.78
IP	0.67	1.00	0.80	1.00	1.00	1.00
LOC	0.85	0.91	0.88	0.83	0.88	0.86
MAL	0.79	0.83	0.81	0.88	0.87	0.88
MD5	1.00	1.00	1.00	1.00	1.00	1.00
OS	0.84	0.95	0.89	0.84	1.00	0.92
PROT	0.94	0.64	0.76	1.00	0.83	0.91
SECTEAM	0.87	0.87	0.87	0.90	0.91	0.90
SHA2	1.00	1.00	1.00	1.00	1.00	1.00
TIME	0.85	0.90	0.87	0.84	0.87	0.85
TOOL	0.62	0.68	0.65	0.69	0.75	0.72
VULID	1.00	0.99	1.00	1.00	1.00	1.00
VULNAME	0.86	0.90	0.88	0.93	0.96	0.95
Micro Avg	0.78	0.84	0.81	0.80	0.84	0.80
Macro Avg	0.84	0.87	0.85	0.77	0.80	0.78
Weighted Avg	0.79	0.84	0.81	0.80	0.84	0.80

diverse annotated NER datasets suitable for building robust AI systems efficiently. The framework's performance and capabilities are demonstrated by merging two prominent open-source NER datasets, DNRTI and APTNER, as a practical case study. By comparing against the manual approach as a baseline, TI-NERmerger efficiently covers over 94% of the manual work within a few minutes, a task that initially required several months to complete manually. The effectiveness of the resulting datasets (DNRTI-STIX and 2APTNER) from both approaches was evaluated using BiLSTM and BERT models. Merging DNRTI-STIX with APTNER produced the Augmented APTNER dataset, denoted as 2APTNER, which significantly surpasses existing TI-NER datasets. 2APTNER comprises 434,150 tokens, 79,161 labeled entities, 16,691 sentences, and 16,439 unique terms compliant with the STIX 2.1 data exchange standard. Looking forward, TI-NERmerger leverages the MITRE ATT&CK repository as the primary source of truth to address annotation inconsistency issues across datasets. This approach can be expanded to include other repository references to enhance model reliability and generalizability. Furthermore, the resulting datasets adhere to STIX 2.1 standards, covering diverse entities, making them valuable resources for extracting cyber threat intelligence (CTI) from security reports.

REFERENCES

Ahmed, K., Khurshid, S. K., and Hina, S. (2024). Cyberentrel: Joint extraction of cyber entities and relations using deep learning. *AComputers & Security*, 136.

Alshammari, N. and Alanazi, S. (2021). The impact of using different annotation schemes on named entity recognition. *Egyptian Informatics Journal*, 22:295–302.

Bakır, H., Çayır, A. N., and Navruz, T. S. (2024). A comprehensive experimental study for analyzing the effects of data augmentation techniques on voice classification. *Multimed Tools Appl 83*, page 17601–17628.

Corporation, T. M. (2015-2023). Att&ck.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

Ding, B., Qin, C., Zhao, R., Luo, T., Li, X., Chen, G., Xia, W., Hu, J., Luu, A. T., and Joty, S. (2024). Data augmentation using llms: Data perspectives, learning paradigms and challenges. *arXiv:2403.02990*.

Guo, Y., Liu, Z., Huang, C., Liu, J., Jing, W., Wang, Z., and Wang, Y. (2021). Cyberrel: Joint entity and relation extraction for cybersecurity concepts. *Information and Communications Security: 23rd International Conference, ICICS 2021, Chongqing, China*, page 447–463.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *ArXiv*.

Jo, H., Lee, Y., and Shin, S. (2022). Vulcan: Automatic extraction and analysis of cyber threat intelligence from unstructured text. *Computers & Security*, 120.

Jordan, B., Piazza, R., and Darley, T. (25 January 2022). Stix version 2.1.

Kiavash, S., Rigel, G., and N, V. V. (2021). Extractor: Extracting attack behavior from threat reports. *In: IEEE EuroS&P*, pages 598–615.

Kim, G., Lee, C., Jo, J., and Lim, H. (2020). Automatic extraction of named entities of cyber threats using a deep bi-lstm-crf network. *Int. J. Mach. Learn. & Cyber*, 11:2341–2355.

Konkol, M. and Konopík, M. (2015). Segment representations in named entity recognition. *International Conference on Text, Speech and Dialogue*, 9302.

Li, Z., Zeng, J., Chen, Y., and Liang, Z. (2022). Attackg: Constructing technique knowledge graph from cyber threat intelligence reports. In *Computer Security – ESORICS 2022: 27th European Symposium on Research in Computer Security, Copenhagen, Denmark, September 26–30, 2022, Proceedings, Part I*, Lecture Notes in Computer Science. Springer International Publishin.

Lin, C.-H., Kaushik, C., Dyer, E. L., and Muthukumar, V. (2024). The good, the bad and the ugly sides of data augmentation: An implicit spectral regularization perspective. *Journal of Machine Learning Research*, 25:1–85.

Marchiori, F., Conti, M., and Verde, N. V. (2023). Stixnet: A novel and modular solution for extracting all stix objects in cti reports. *ARES 23: Proceedings of the 18th International Conference on Availability, Reliability and Security*, 2(3):1–11.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., and Thirion, B. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Perrina, F., Marchiori, F., Conti, M., and Verde, N. V. (2023). Agir: Automating cyber threat intelligence reporting with natural language generation. *2023 IEEE International Conference on Big Data (BigData)*, pages 3053–3062.

Rani, N., Saha, B., Maurya, V., and Shukla, S. K. (2023a). Ttphunter: Automated extraction of actionable intelligence as ttps from narrative threat reports. *In Australasian Information Security Conference (AISC 2023)*, page 126–134.

Rani, N., Saha, B., Maurya, V., and Shukla, S. K. (2023b). Ttphunter: Automated extraction of actionable intelligence as ttps from narrative threat reports. *ACSW '23: Proceedings of the 2023 Australasian Computer Science Week*, page 126–134.

Varghese, V., S, M., and Kb, S. (2023). Extraction of actionable threat intelligence from dark web data. *2023 International Conference on Control, Communication and Computing (ICCC), Thiruvananthapuram, India*, pages 1–5.

Wang, X., He, S., Xiong, Z., Wei, X., Jiang, Z., Chen, S., and Jiang, J. (2020a). Aptner.

Wang, X., He, S., Xiong, Z., Wei, X., Jiang, Z., Chen, S., and Jiang, J. (2022). Aptner: A specific dataset for ner missions in cyber threat intelligence field. *2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Hangzhou, China*, pages 1233–1238.

Wang, X., Liu, X., Ao, S., Li, N., Jiang, Z., Xu, Z., Xiong, Z., Xiong, M., and Zhang, X. (2020b). Dnrti.

Wang, X., Liu, X., Ao, S., Li, N., Jiang, Z., Xu, Z., Xiong, Z., Xiong, M., and Zhang, X. (2020c). Dnrti: A large-scale dataset for named entity recognition in threat intelligence. *2020 IEEE 19th International Conference on Trust, Security, and Privacy in Computing and Communications (TrustCom), Guangzhou, China*, pages 1842–1848.

YI, F., JIANG, B., WANG, L., and WU, J. (2020). Cybersecurity named entity recognition using multi-modal ensemble learning. *in IEEE Access*, 8(10):63214–63224.

Zhou, S., Liu, J., Zhong, X., and Zhao, W. (2021). Named entity recognition using bert with whole world masking in cybersecurity domain. *2021 IEEE 6th International Conference on Big Data Analytics (ICBDA), Xiamen, China*, pages 316–320.

Zhou, S., Long, Z., Tan, L., and Guo, H. (2018). Automatic identification of indicators of compromise using neural-based sequence labelling. *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation, Hong Kong*.

Zhou, S., Zhang, J., Jiang, H., Lundh, T., and Ng, A. Y. (2020). Data augmentation with mobius transformations. *arXiv:2002.02917*.