Journal of Management Information Systems

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/mmis20

Semi-Supervised Cyber Threat Identification in Dark Net Markets: A Transductive and Deep Learning Approach

Mohammadreza Ebrahimi , Jay F. Nunamaker Jr. & Hsinchun Chen

To cite this article: Mohammadreza Ebrahimi , Jay F. Nunamaker Jr. & Hsinchun Chen (2020) Semi-Supervised Cyber Threat Identification in Dark Net Markets: A Transductive and Deep Learning Approach, Journal of Management Information Systems, 37:3, 694-722, DOI: 10.1080/07421222.2020.1790186

To link to this article: https://doi.org/10.1080/07421222.2020.1790186

JOURNAL OF MANAGEMENT INFORMATION SYSTEMS
2020, VOL. 37, NO. 3, 694–722
https://doi.org/10.1080/07421222.2020.1790186

Routledge
Taylor & Francis Group

Semi-Supervised Cyber Threat Identification in Dark Net Markets: A Transductive and Deep Learning Approach

Mohammadreza Ebrahimi[a], Jay F. Nunamaker Jr.[a], and Hsinchun Chen[a]

[a]Eller College of Management, University of Arizona, Tucson, AZ, USA

ABSTRACT

Dark Net Marketplaces (DNMs), online selling platforms on the dark web, constitute a major component of the underground economy. Due to the anonymity and increasing accessibility of these platforms, they are rich sources of cyber threats such as hacking tools, data breaches, and personal account information. As the number of products offered on DNMs increases, researchers have begun to develop automated machine learning-based threat identification approaches. A major challenge in adopting such an approach is that the task typically requires manually labeled training data, which is expensive and impractical. We propose a novel semi-supervised labeling technique for leveraging unlabeled data based on the lexical and structural characteristics of DNMs using transductive learning. Empirical results show that the proposed approach leads to an approximately 3-5% increase in classification performance measured by F_1-score, while increasing *both* precision and recall. To further improve the identification performance, we adopt Long Short-Term Memory (LSTM) as a deep learning structure on top of the proposed labeling method. The results are evaluated against a large collection of 79K product listings obtained from the most popular DNMs. Our method outperforms the state-of-the-art methods in threat identification and is considered as an important step toward lowering the human supervision cost in realizing automated threat detection within cyber threat intelligence organizations.

KEYWORDS

Dark net marketplaces; cyber threats; semi-supervised labeling; transductive learning; deep learning; long short-term memory; threat detection

Introduction

Dark Net Marketplaces (DNMs) are rich sources of information about malicious cybersecurity-related products, including *tools, manuals, tutorials*, and *personal and financial accounts* information. In recent years, the number of DNMs and the illegal products they host have increased constantly; their estimated turnover rose from $15-17M in 2012 to $150-180M in 2015 [60]. Cyber threat intelligence companies (e.g., Symantec, Splunk, FireEye, and McAfee) monitor the content of the dark web on a daily basis to identify and prioritize potential cyber threats as a crucial task in their operational intelligence process. Manual identification of these threats is prohibitive and takes a considerable amount of human analysts' time. Threat Identification is an integral part of Cyber Threat Intelligence [10]. Common cyber threats include malware, data breaches, or any other means of harm to an individual or organization in cyberspace.

We refer to malicious cybersecurity-related products in DNMs as "cyber threats" in this research. Figure 1 provides a screen capture of a product listing for hacking tools available at one of the most frequented markets, AlphaBay (e.g., Botnets, Exploits, Ransomware, etc.).

CONTACT Mohammadreza Ebrahimi ebrahimi@email.arizona.edu Eller College of Management, University of Arizona, 1130 E Hellen St., Tucson, AZ, 85721, USA.

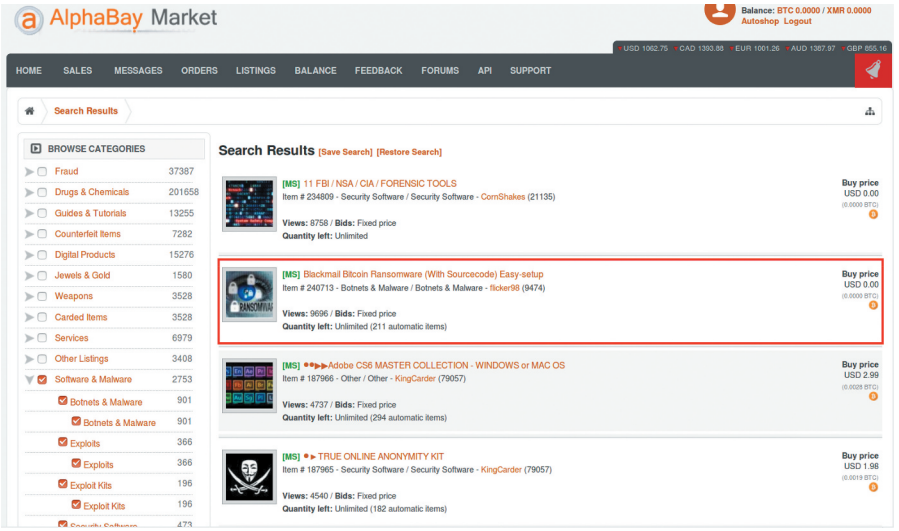

Figure 1. Listings of hacking products in a popular Dark Net Marketplaces (DNM).

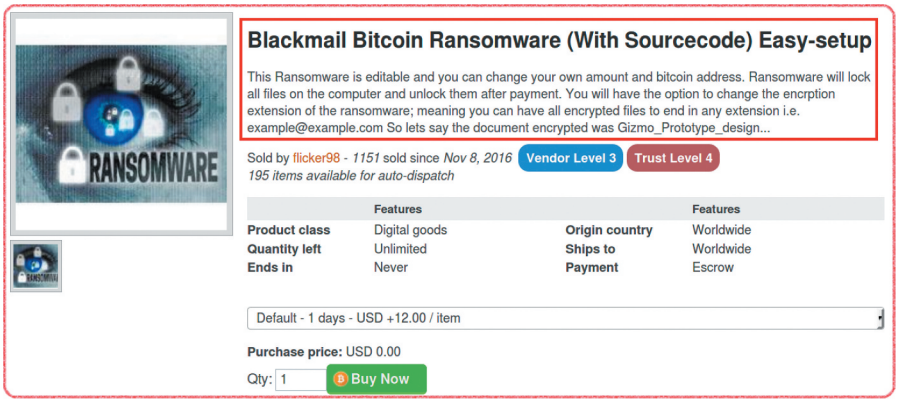

Figure 2. Editable ransomware for sale on the same marketplace.

Figure 2 depicts an editable ransomware and its corresponding information advertised on the same market (product description, seller name, payment method, quantity, etc.).

DNMs host a variety of cyber threats, as summarized in Table 1.

Threat information is intrinsically perishable. The information is only valuable if it is obtained in a certain period of time [18]. As a result, threat intelligence can lose its value within days or even hours [18]. This suggests that we need automated methods that can identify potential threats with minimum delay. Machine learning techniques appear to be a promising approach for automated threat identification [38, 45, 49]. However, two major challenges remain for leveraging machine learning techniques to this end. First, to learn a task, most machine learning methods need to be supervised by humans who provide a significant number of instances of ground-truth data (also known as labels) during the training process, which makes supervised learning labor-intensive. Because most of the content obtained from DNMs is originally unlabeled, manual labeling of DNM records

Table 1. Common major threats in Dark Net Marketplaces (DNMs).

Threat Category	Description
Ransomware	A type of malware that demands a ransom in exchange for some stolen functionality such as accessing encrypted data on the victim's hard drive [19,31].
Zero-day Attacks	Packages for exploiting vulnerabilities that have not been announced publicly by the software manufacturer [9].
Keyloggers	Tools that track the keystrokes on the victim's keyboard in a covert manner with malicious intent [43].
DDoS Attack Tools	Software packages that conduct Denial of Service attacks on servers.
SQL Injection Tools	Tools that manipulate insecure databases by attaching malicious scripts to information retrieval queries.
Mobile Malware	Other types of malicious software targeted at mobile operating systems such as Android and iOS [21].
Personal Accounts	Including breached personal and bank account information of victims.

Notes: DDOS, distributed denial of service; SQL, structured query language.

requires expensive human labor and expertise. The lack of labeled data in web applications [70] makes it impractical to directly apply supervised learning techniques to detecting cyber threats. Second, automated detection approaches (including machine learning) need to minimize false positives and false negatives to increase threat detection performance. However, false positives and false negatives tend to be antagonistic, which means efforts to reduce one of them often increases the other. This issue is not easily addressed in many application domains, including cyber threat identification.

To address the first challenge, the unlabeled data can be leveraged to improve the performance of classification tasks using semi-supervised learning [63, 72] which is characterized by learning in the presence of both labeled and unlabeled data [72]. Semi-supervised learning requires significantly less data because it can learn from underlying patterns in unlabeled data. This results in a sizable cost reduction for training the machine learning method. To address the second issue in threat identification, state-of-the-art machine learning approaches such as deep learning [36] can be enhanced to reduce false positives and false negatives. Accordingly, this study aims to propose a new approach that can:

(1) Effectively leverage the lexical characteristics (e.g., threat-related words such as "hack") and structural characteristics (e.g., predefined product categories in a platform such as "hacking tools") characteristics of DNM data for semi-supervised labeling to improve the classification performance and reduce the need for human labeling, and
(2) Reduce false positives and negatives for cyber threat identification tasks by developing Deep Long Short-Term Memory (LSTM) on top of the proposed semi-supervised labeling technique.

The proposed approach contributes to an important first step towards reducing the cost associated with human supervision within cyber threat intelligence organizations. At a managerial level, this study aims to facilitate operational intelligence in these organizations by providing a framework that can benefit from unlabeled data, which is often readily available as opposed to limited human-labeled data in the cybersecurity domain.

The organization of the paper is as follows. The second section provides a literature review of cyber threat intelligence, semi-supervised labeling, and deep text classification techniques. The third section details our proposed method. The fourth section is devoted to the experimental evaluation of the proposed method on a large dataset extracted from popular dark net marketplaces. The fifth section describes the managerial implications of the proposed method as well as the implementation considerations in cybersecurity organizations. Finally, the sixth section includes the conclusion and possible future directions.

Literature Review

Given the mentioned objectives, first, we review recent cybersecurity information science (IS) research to position our research among other cybersecurity studies. Second, cyber threat identification in the dark web is reviewed to understand the current threat identification methods. Next, we explain the use of transductive learning as a method of semi-supervised labeling for effective use of unlabeled data in DNMs with the goal of improving the classification performance of supervised models. Then, we review text classification using deep architectures to gain knowledge of deep learning text classification as an emerging method that is considered to be the state of the art in many fields [36]. Finally, we describe Long Short-Term Memory (LSTM), a state-of-the-art deep learning method in text classification that is closely related to our proposed method.

Recent IS cybersecurity research

To position our study, we recognize recent cybersecurity studies in three mainstream IS journals, including *MIS Quarterly, JMIS*, and *ISR* during the past four years. We build on the taxonomy given by Hui et al. [24], and categorize these studies into three main IS streams: Security compliance and users' behavior [1, 11, 25, 44, 61, 62], security risk management [2, 3, 23, 30, 59, 64, 67, 68], and predictive security analytics [7, 8, 38, 49, 56]. Table 2 provides a research taxonomy, which is not meant to be exhaustive. Rather, it aims to reflect the current state of the art within IS discipline.

While the first stream of research mainly focuses on the behavior of individuals who interact with cybersecurity systems, the second stream targets mitigation of the risk associated with cybersecurity threats and decisions. Unlike the first and second streams, which mainly focus on building descriptive, explanatory models to understand cybersecurity-related phenomena, the third stream of research tries to build statistical machine learning models to predict a variable of interest based on the characteristics of the cybersecurity data. However, given that these statistical models often learn from examples, most of the studies in this IS stream require human supervision, often manifesting in the form of constructing cost-sensitive, hand-crafted, labeled datasets for model training. Within the predictive security analytics, semi-supervised models can be used to significantly reduce the need for human-labeled datasets by being able to learn from "unlabeled" data as well as labeled data. This forms the focus and the contribution of our study in regards to other cybersecurity IS research (Figure 3).

Table 2. Selected recent cybersecurity research taxonomy from mainstream is publications.

Category	Year	Author(s)	Focus
Security compliance and users' behavior	2018	Moody et al.	Unification of extant behavioral models for security compliance [44]
	2018	Vance et al.	Studying users' habituation to security warnings [61]
	2017	Jensen et al.	Mitigating phishing attacks by user security training [25]
	2017	Wang et al.	Understanding behavioral responses to phishing attacks [62]
	2016	Chen and Zahedi	Studying user behavior in dealing with online security threats [11]
	2016	Anderson et al.	Reasons for users' habituation to security warnings [1]
Security risk management	2019	Yue et al.	The effect of discussions in hacker forums on DDoS attack victims [68]
	2018	Benaroch	A model for proactive risk mitigation in cybersecurity investments [3]
	2018	Karhu et al.	Securing software resources to prevent exploitation from hostile firms [30]
	2017	Temizkan et al.	The impact of software diversity on network security risk [59]
	2017	Yang et al.	Applying operational risk management to the security of financial organizations [67]
	2017	Hui et al.	The impact of law enforcement on deterring DDoS attacks [23]
	2017	Angst et al.	Reducing the incidence of security data breaches [2]
	2016	Wolff	Studying the effect of cyber defense tools in firms' security [64]
Predictive security analytics (Supervised Learning)	2019	Sun Yin et al.	De-anonymizing the transactions used by cybercriminals in the dark web [56]
	2019	Benjamin et al.	Guidelines for designing predictive models for cybersecurity research [7]
	2017	Samtani et al.	Designing a malware source code classification system [49]
	2016	Li et al.	Designing a text mining framework for key seller identification in cyber carding community [38]
	2016	Benjamin et al.	Designing a computational framework to identify key participants in hacker communities [8]

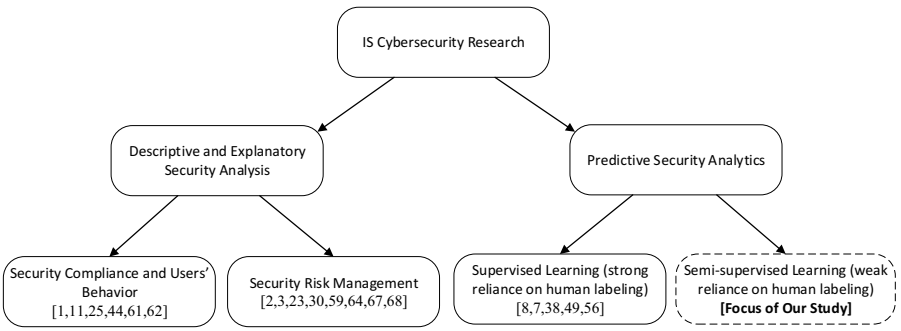

Figure 3. Position of our study among IS cybersecurity research.

Cyber threat identification in the dark web

Prior work on cyber threat identification within the dark web can be classified based on the targeted dark web platform, including hacker forums, carding shops (platforms for selling stolen financial information such as bank accounts), Internet Relay Chat (IRC) platforms (anonymous message boards), and dark net marketplaces. A branch of studies focuses on enabling data collection and conducting descriptive analytics in all platform types [6, 15].

Their goal is to identify the main data sources of cyber threats on the dark web. A major stream of relevant studies focuses on cyber threat detection within hacker forums [13, 47, 49, 50]. Schäfer et al. [50] and Portnoff [47] identify cyber threat topics such as botnets or exploits and aim to discover their trends within hacker forums. Samtani et al. [49] use Support Vector Machine (SVM) for classifying malware source codes or forum posts and, then, find the dominant topics in each of the malware categories using topic modeling. Similarly, Deliu et al. [13] identify the relevant posts to cybersecurity in forums and apply topic modeling to discover the main topics discussed by hackers. Some other studies [4, 5, 58] use neural language models to gain insight into the language used by hackers in writing the forum posts and communicating with each other. Also, some studies focus on identifying key hackers in IRC channels [8, 52] and carding shops [37, 38]. To the best of our knowledge, the work of Nunes et al. [45] is the only study that focuses on threat identification in dark net marketplaces. The study uses SVM as the classifier and co-training as a semi-supervised labeling technique and includes approximately 11K products, of which 25% (roughly 2,870 samples) were manually labeled. The results showed that even though using co-training results in increased recall (a normalized measurement of false negatives), it also leads to the reduction of precision (a normalized measurement of false positives) as an undesirable side effect. While our proposed approach requires only approximately 3% (1,678 samples) of the data to be labeled manually, it increases both precision and recall simultaneously. This is important from the practical point of view in cyber threat detection scenarios, where it is desirable to reduce the number of falsely identified threats as well as the number of false negatives.

Semi-supervised labeling through transductive learning

Most prior classification techniques are supervised learning methods and cannot be directly trained on the unlabeled DNM data. Under certain conditions, using unlabeled data in the learning process can lead to better results than supervised learning [26, 63]. Transductive SVMs (TSVMs) have yielded promising results in text classification as a way to leverage unlabeled data [53, 54]. We propose a TSVM based algorithm for automated labeling of the DNM data. The proposed algorithm labels the entire DNM dataset as a preliminary step in order to boost the performance of the proposed supervised LSTM. The algorithm is discussed in Section 3. Here we highlight the main concepts of TSVM.

Joachims [26] explored the use of TSVM, as an extension of SVM, in text classification through rigorous analysis. SVM uses convex optimization to obtain the optimum hyperplane that separates data objects (web documents in our case) into two classes, positive or negative. In the case of cyber threat detection, positive be a cyber threat, and negative would be a non-threat. Unlike regular SVM, TSVM optimizes a non-convex constrained problem, which provides TSVM with the flexibility to learn from unlabeled data through an iterative process. The TSVM iterative algorithm works best when the ratio of positive samples (cyber threats in our case) in the unlabeled data is known.

Figure 4 contrasts the decision hyperplanes for inductive and transductive SVMs. The decision hyperplane obtained by transductive SVM (solid line) leads to a better separation of positive and negative examples (denoted by + and -) compared to the hyperplane obtained from regular SVM (dashed line). TSVM leverages unlabeled samples by testing the different combinations of assigning labels to unlabeled data and selecting the feasible combination that minimizes the cost function defined in section 3-1.

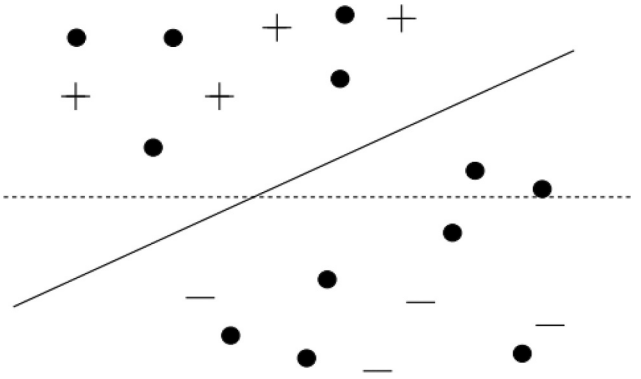

Figure 4. The comparison of hyperplanes in inductive and transductive classification (image from Joachims, 1999 [26]). The + and – signs denote positive (cyber threat) and negative (non-threat) data objects, respectively, while dark circles represent unlabeled samples. The dashed line shows the decision hyperplane for inductive learning with possible misclassified unlabeled samples. The solid line depicts the decision hyperplane for transductive learning.

Text classification using deep learning architectures

Recent use of deep learning architectures in text classification has led to the advent of the state-of-the-art method in this application field. The success of deep learning architectures is mainly owed to their ability to eliminate the manual and tedious feature engineering necessary for traditional (non-deep) machine learning approaches. Properly applied, a deep learning classifier could, for example, distinguish illegally copied software, hacking tools, and stolen credit card information, which may not be found through using traditional machine learning approaches. Here we summarize important studies in this area based on the input granularity (i.e., sentence or document) as well as the deep learning method that was employed. Larochelle and Bengio [35] used Restricted Boltzmann Machines (RBMs) for document classification in a semi-supervised setting. Liu [40] used RBMs to extract high-level features from a document for the subsequent application of SVM. Finally, Zhou et al. [71] used RBMs for unsupervised feature extraction in a Deep Belief Network to conduct sentiment classification on a document. Some studies conduct the analysis at the sentence level [14, 32, 55]. [14] utilized Convolutional Neural Network (CNN) for sentiment classification. Socher et al. [55] introduced Recursive Neural Tensor Network to accomplish the same task. In Kim [32], the author designed a general-purpose architecture for seven different common Natural Language Processing (NLP) tasks including text classification. CNNs have also shown promising results in text classification at the document level. Johnson and Zhang [28] applied the convolution layer directly to the high-dimensional text without use of word embedding. The authors in Ebrahimi et al. [16] used the same approach to identify predatory chat conversations. Lai et al. [34] adopted CNN in a recurrent structure and Zhang et al. [69] applied CNN on character inputs and achieved promising results on eight large-scale datasets including 3.65M Amazon reviews. Recurrent neural networks, specifically LSTM and Gated Recurrent Units (GRUs), have been shown to outperform the other deep learning methods in several text classification tasks. Liu et al. [39] employed LSTM in a multi-task learning setting to accomplish subjectivity and sentiment analysis. Tang et al. [57] used GRUs to aggregate sentence-level sentiments to obtain classification at document level. Finally, Johnson and Zhang [29] introduced the

application of region embedding and LSTM to achieve the state of the art on several text classification benchmark datasets.

Among deep learning methods previously discussed, recurrent neural networks including LSTM are capable of handling time dependencies in sequential data such as text [20]. As a result, LSTM outperforms other deep learning methods in text sequence modeling and text classification domain [29]. LSTM was originally introduced by Hochreiter and Schmidhuber in their seminal work [22] as a variant of recurrent neural networks. It aims to remedy the problem of vanishing or exploding gradients in backward propagation of errors during the training of RNNs [22]. Bidirectional LSTM is a variant of LSTM in which the output at time ***t*** also depends on the future inputs in the sequence in addition to the previous inputs [51]. Gated Recurrent Unit (GRU) is another variant that simplifies the LSTM memory cell by removing the output gate and replacing the input and forget gates with update and reset gates [12].

Here we describe the basic LSTM and in Section 3 we describe our improved algorithm for the cyber intelligence application domain. Gating is a major component in LSTM and is embedded in a network unit called *memory cell*, which mimics "*memorizing*" and/or "*forgetting*" the effect of the current input signal in the current time step. Figure 5 shows the original structure of a memory cell. The figure was reconstructed from Hochreiter and Schmidhuber [22] with minor changes.

Figure 5. (left) A LSTM Network with one hidden layer and two memory blocks. (right) An enlarged memory cell with two gates. (Image was reconstructed based on Hochreiter and Schmidhuber, 1997 [22] with minor changes.)

In Figure 5, S_c denotes the current state. The loop-back arrow shows that the memory cell may forget (or take into account) its state in the previous time step by adding it to the current input (i.e., gy^{in}) in order to prevent the error from decaying when it needs to be propagated. We adapt an LSTM architecture proposed by Johnson and Zhang [29] and describe our architecture in Section 3.

Research gaps and questions

Informed by the LSTM's analytical merits in processing sequential data such as text, next we discuss the relevant LSTM literature in IS to identify potential research gaps. Recurrent neural networks and specifically LSTMs have recently been used by IS scholars mainly in

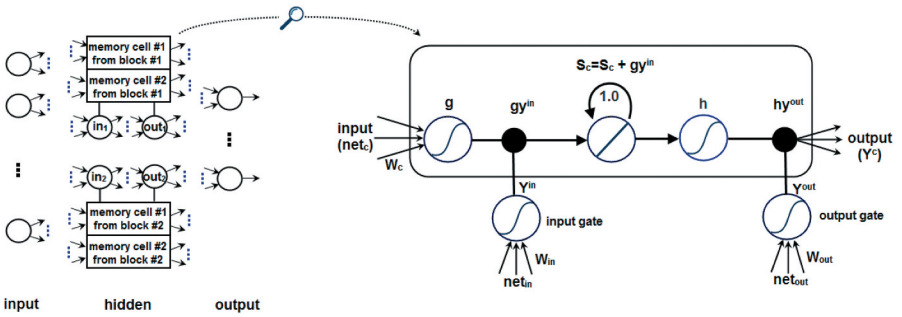

Figure 5. Proposed cyber threat identification design.

health care and social media analytics [41, 65, 66]. Liu et al (Forthcoming) utilize LSTM to identify medical terms in YouTube videos. Xie et al. [65] use LSTM to process user-generated textual content on WebMD. Finally, Xie and Zhang [66] employ LSTM to process Medicare patients' hospitalization data. Common in all of these studies, manual data labeling is a challenge in training the LSTM in a supervised manner. Due to the high cost, such a tedious process is often performed on small sets of instances and does not leverage the underlying knowledge embedded in the widely accessible unlabeled data. Similarly, within text classification literature, most studies rely on conventional supervised learning with manually labeled data. It is useful to leverage unlabeled data via semi-supervised learning in order to reduce the training cost.

Also within the cyber threat detection domain, prior research has focused on cyber threat detection in hacker forums and carding shops. It is also necessary to be able to identify cyber threats in dark net marketplaces. Given these identified gaps, our research aims to address the following research questions:

(1) Can we leverage unlabeled DNM data to improve cyber threat classification performance?
(2) Does applying a deep learning-based classifier on the DNM data lead to further improvement of the classification results?

Motivated by these questions, in this study, we propose a deep learning framework based on a novel deep learning method that benefits from semi-supervised labeling for cyber threat detection in the dark web. The novelty of our approach is twofold: 1) to our knowledge, our proposed approach is the first semi-supervised deep learning method for cyber threat detection. 2) We contribute to the text classification literature by enhancing traditional supervised LSTM to semi-supervised settings. Finally, we contribute to the cybersecurity IS knowledge base by providing a framework that can alleviate the cost associated with acquiring human-label data for building analytical models on sequential data as an important task for realizing operational intelligence. In the next section, we present our proposed design to address the questions previously mentioned.

Proposed cyber threat identification design: A transductive and deep learning approach

Our research design is composed of four main steps as shown in Figure 6. The first two steps are discussed in Section 4-1. Here we focus on the main two components of our proposed method: the proposed semi-supervised labeling through transductive learning and threat identification via deep bidirectional LSTM networks. While both components aim to improve the performance of automated cyber threat detection, they are motivated by different purposes: the first component is intended to leverage the knowledge from unlabeled data. However, the role of the second component is to leverage expensive human-labeled data to the fullest extent. We detail the underlying process for each of these two components in the following subsections.

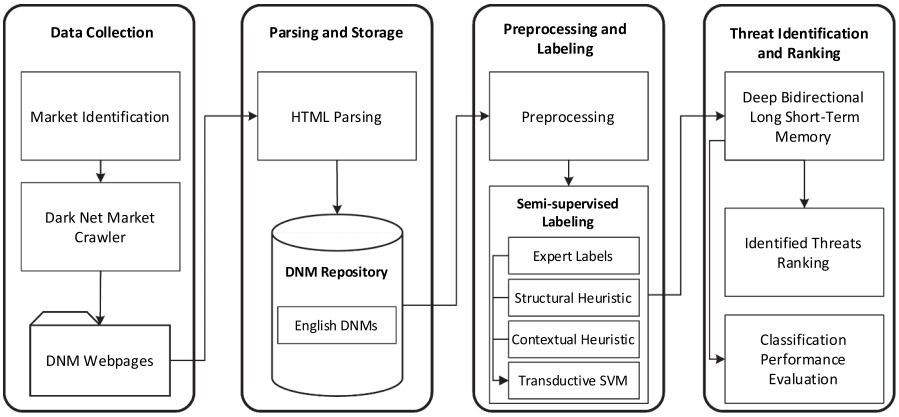

Figure 6. Proposed cyber threat identification design.

Semi-supervised labeling

Given the success of TSVM in semi-supervised learning, we are motivated to provide a text labeling algorithm using TSVM as an underlying semi-supervised labeling method, which can subsequently be employed to complement advanced supervised text classification techniques with the ultimate goal of improving the performance of automated cyber threat detection. Unlike inductive SVM, TSVM solves a non-convex constrained optimization problem to maximize the margin in the presence of unlabeled data. Let L and U denote the number of labeled and unlabeled samples, respectively. Let y_i denote the labels of labeled samples. We used the following TSVM objective function as described in Sindhwani and Keerthi [53]:

$$\begin{aligned} &\min_{w}\left(\frac{\lambda}{2}w^2+\frac{1}{2L}\sum_{i=1}^{L}l\left(y_i,w^Tx_i\right)+\frac{\lambda'}{2U}\sum_{j=1}^{U}l\left(y_i',w^Tx_j'\right)\right)\\ &\text{subjectto}:\ \frac{1}{U}\sum_{j=1}^{u}\max\left(0,\text{sign}\left(w^Tx_j'\right)\right)=r \end{aligned} \tag{1}$$

The first two terms in the minimization are the same as classic SVM. λ is the regularization parameter and l is a loss function; W is the weight vector; y'_i is chosen from {+1,-1} as the label of unlabeled data; and λ' controls the influence of unlabeled data. The minimization problem is subject to the constraint that the fraction r of the unlabeled data must be classified as positive. Even though both λ and λ' can be tuned by cross-validation, there is no formalized way to estimate r [53]. However, the choice of r is crucial to the quality of semi-supervised labeling. We denote the estimated parameter as $\hat{r}$. We exploit lexical and structural characteristics of DNM data to develop domain-specific heuristics for estimating this important parameter.

Heuristics

Structural and lexical characteristics of DNM data can lead us to powerful heuristics about content. We developed two heuristic functions based on the lexical and structural characteristics of document d in the DNM dataset. We design the heuristics to complement

each other. As a general property, the structural heuristic helps reduce false negatives, and the lexical heuristic helps reduce false positives. We define each heuristic as follows.

Webpage structure heuristic (H_1: Structural heuristic). An example of a listing category was shown in the left navigation bar in Figure 1. This heuristic leverages the listing structure of a product's webpage. It labels d as negative if it is listed in a category that is not related to cybersecurity. For example, it is not likely that cyber threats are found in the listing of *drugs* or *arms*. As a result, H_1 is highly reliable in identifying negative labels. Sellers tend to advertise their products in relevant listings to maximize visits. Therefore, it is reasonable to assume that H_1 helps avoid false negatives. However, applying H_1 may produce many false positives because it simply labels all documents that appear in listings such as *digital goods* as positive. Thus, it is necessary to design another heuristic to lower false positives.

Keyword matching heuristic (H_2: Lexical heuristic). Let w be a hand-curated list of cybersecurity-related keywords (the list is attached as Appendix 1). Let $N_{w,d}$ be the number of words in w that appear in document d. The occurrence of at least $k \geq 0$ elements of w in d indicates that it is related to cybersecurity (k denotes the number of security-related keywords that appear in d). Accordingly, this heuristic labels d as positive if and only if $N_{w,d} \geq k$, and as negative if otherwise. If k is large enough, the heuristic would be conservative enough that no false positives are allowed. As a result, unlike H_1, H_2 can be designed to reduce false positives.

Heuristic design and robustness. To circumvent introducing extra parameters to the learning process of our approach, we avoid designing complex structural and lexical heuristics. Despite the simplicity, throughout our evaluations, we show that the defined heuristics perform well in practice. Robustness of these heuristics against adversarial measures, which could potentially be taken by sellers to bypass automated threat identification, needs to be justified. The adversaries often do not have control over the structural heuristic since the structure is designed by the platform administrator. However, they can attempt to mislead the lexical heuristic by adding irrelevant keywords to the description of the products they aim to sell on the market. To justify the robustness of our lexical heuristic against adversarial measures, we draw on an important property of DNMs, which is the significant authority of the moderator or administrator in approving the advertised products after they are submitted by sellers [46]. As one of their main roles, moderators aim to facilitate the keyword-based product search for buyers, and thus they often do not allow sellers to add irrelevant keywords to their product description. Even if an adversary tries to include irrelevant keywords in their descriptions to distort the defined lexical heuristic, these changes would often be counteracted by administrators or moderators, based on the rules of the market. If a seller insists on performing such actions they are very likely to be banned from posting their products on the market [46]. Overall, the robustness of the heuristics against the adversarial input changes is an important aspect that needs to be taken to account in designing automated cyber threat detection approaches.

Parameter estimation

As noted in Section 3-1, having a viable estimate of the ratio of positive instances in unlabeled data (i.e., parameter r) is crucial to semi-supervised labeling. Hence, we aim to estimate this parameter in our proposed semi-supervised labeling process. The labeling process begins with applying the described heuristics on unlabeled data sequentially, as illustrated in Figure 7. The output of the heuristic functions is used as an intermediary result to estimate parameter r based on Algorithm 1.

TSVM uses the estimated parameter as well as a small subset of the data (3% in our case) that has been labeled by experts to produce labels for unlabeled data. The resulting labeled data is used to train a Deep LSTM network in a supervised manner. Algorithm 1 details the semi-supervised labeling procedure proposed for labeling the DNM dataset.

Algorithm 1. Semi-supervised data labeling using heuristics and TSVM

Input: Original corpus D (composed of labeled and unlabeled data)

Output: labels for unlabeled documents in corpus D

\\Initialization: (steps 1 and 2)

1. Let $D_L \subset D$ and $D_U \subset D$ denote the expert-labeled and unlabeled subsets of the dataset D respectively.

2. Define the heuristic functions H_1 and H_2 as already discussed to process document $d \in D$.

3. Apply H_1 to D_U and obtain the labeled dataset D_{H1}. Construct subset $D_{H1}^{neg} \subset D_{H1}$, with the negative samples obtained from applying H_1.

4. Apply H_2 to $D_U - D_{H1}^{neg}$ and obtain the labeled dataset D_{H2}. Construct subset $D_{H2}^{pos} \subset D_{H2}$, with the positive samples obtained from applying H_2.

5. Train TSVM classifier with setting the ratio of positive samples in unlabeled data $r = \frac{D_{H2}^{pos}}{D_U}$ and using the combination of expert-labeled and unlabeled subsets $\{D_L \cup D_U\}$.

6. Apply the obtained model on the rest of the unlabeled data points (i.e., $D_U - D_{H1}^{neg} \cup D_{H2}^{pos}$) or any other validation set.

Given the sensitivity of TSVM to the choice of r, in practice, it is very useful to have information about the optimal r so that choosing non-viable values can be avoided. To this end, knowing the lower bound of r can be very valuable to the success of semi-supervised labeling. It can be shown that the estimated r in our

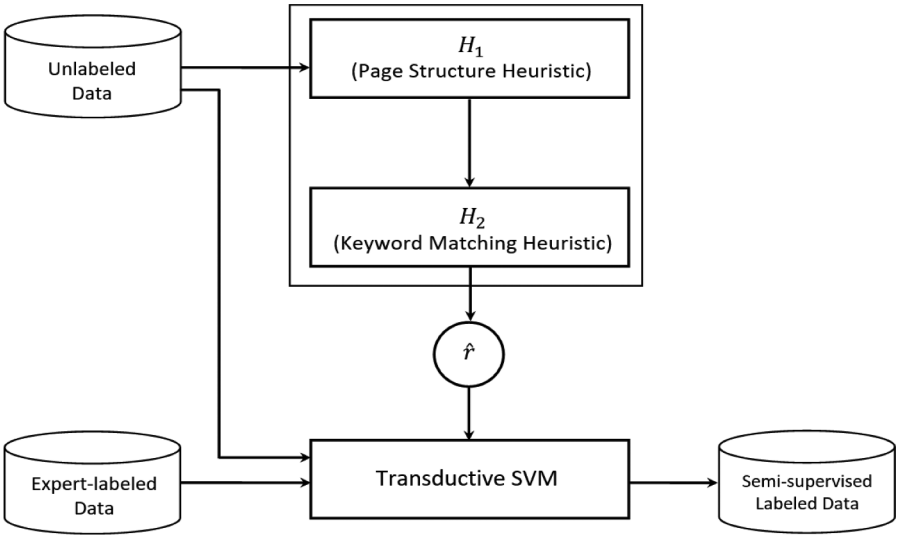

Figure 7. Data labeling based on transductive SVM.

approach leads to the lower bound for the optimal value of r under mild assumptions (see Appendix 1).

To test the empirical validity of the estimation method, we applied our parameter estimation method to the DNM dataset. Since r is a ratio, it ranges between 0 and 1. First, we obtain r^*, the pseudo-optimal value of r, by exhaustively testing all the possible values with 2-digit precision (100 tests). Then, we compare the estimated value obtained from the method ($\hat{r}$) to the optimal value (r^*). We expect the estimated value to be reasonably close to the optimal value. We also expect the estimated value to be less than the optimal value since it is the lower bound of r^*. Note that exhaustively testing the values for r is significantly time-consuming and impractical. Here we conducted such an expensive test merely to show the empirical validity of our estimation.

By applying Algorithm 1 on the DNM dataset, the unlabeled data is estimated to contain 14,636 positive samples and 44,258 negative samples. The estimated r would be $\frac{14.636}{(14,636+44258)} \cong 0.248$.

Figure 8 shows the changes in the classification accuracy of TSVM for different values of r with 2-digit precision. We see that the optimal value of r is0.25, while the method estimated $\hat{r} = 0.248$. This shows that the proposed parameter estimation method yields results that are comparable to the optimal value.

Being equipped with this estimation, we proceed to the next step of threat identification via a deep LSTM network (see the steps shown in Figure 6).

Threat identification and ranking

As shown in Figure 6, after semi-supervised labeling, we adopt a variant of deep bidirectional LSTM to perform threat identification on the DNM dataset. The labels obtained from the semi-supervised labeling process are used as the input of our LSTM architecture

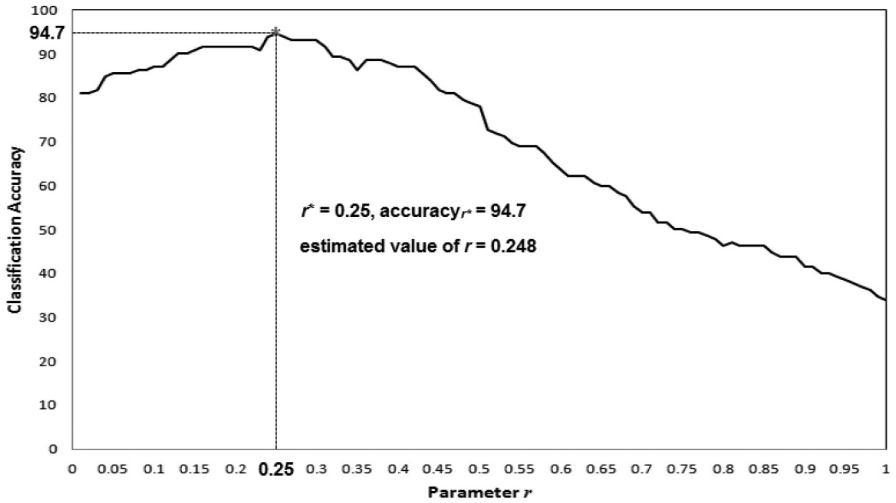

Figure 8. Changes of accuracy for different values of r with 2-digit precision: The estimated value is significantly close to the pseudo optimal value.

which in turn outputs the probability of being a cyber threat. As noted in Section 2-4, LSTM is the state of the art for sequence classification. In our architecture, the input to LSTM is the sequence of words obtained from associated product descriptions on DNMs. Inspired by Johnson and Zhang [29], we remove input and output gates and customize the LSTM for the threat identification task as follows.

$$\left.\begin{aligned} f_t &= \tanh\left(W^f x_t + U^f h_{t-1} + b^f\right) \\ g_t &= \tanh\left(W^g x_t + U^g h_{t-1} + b^g\right) \end{aligned}\right\} \; c_t = c_{t-1} \odot f_t + \; g_t \Rightarrow h_t = \tanh(c_t) \tag{2}$$

where f_t and h_t denote the output of the forget gate and hidden state, respectively; and g_t denotes the hidden state of the memory cell calculated based on the current input and the hidden state in the previous time step. The current cell state (c_t) is calculated based on the element-wise multiplication of the output of the forget gate and previous cell state. W and U are weight matrices, and *tanh* is the hyperbolic tangent function used as the nonlinear activation function [20]. For input z, tanh is calculated as follows:

$$\tanh(z) = (\exp(z) - \exp(-z))/(\exp(z) + \exp(-z)) \tag{3}$$

The choice of tanh as activation function is discussed as a hyper-parameter in Section 4-2-1.

Figure 9 depicts the bidirectional LSTM architecture that we adopted for the purpose of threat identification in dark net marketplaces.

The input documents are encoded as one-hot vectors in which the presence or absence of a word is denoted by one or zero, respectively. The LSTM layer comprises two chains of 100 memory cells linked in opposite directions. The pooling layer aggregates the hidden states produced by the LSTM layer. Consistent with Goldberg [20] and Johnson and Zhang [29], we used *max pooling* as the most common pooling operation to get the most

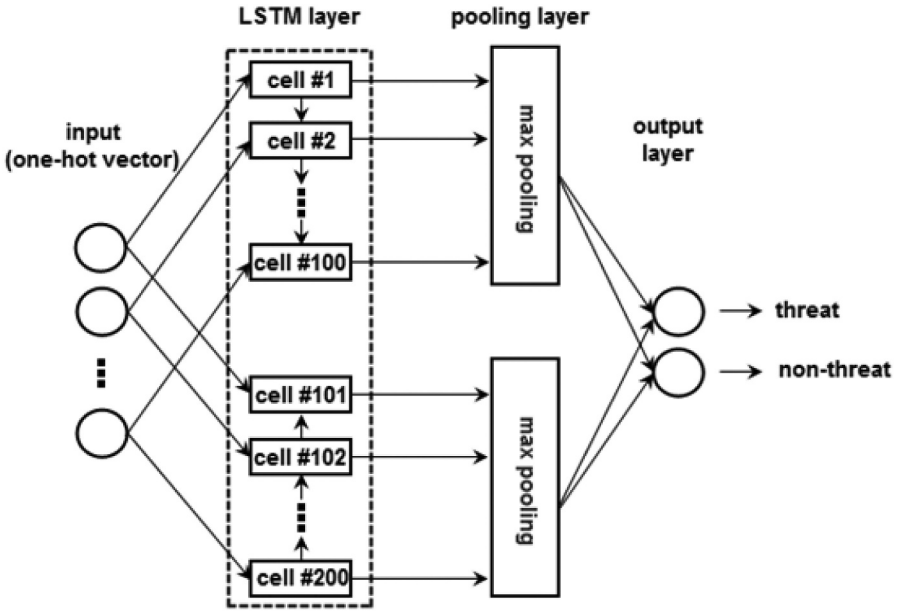

Figure 9. Threat identification network architecture.

salient information across the hidden states emitted from LSTM layer. Finally, the output layer receives the salient information about the product descriptions and classifies the document as threat or non-threat. The LSTM network is trained on the data obtained from the described semi-supervised labeling method. The final classification output can be viewed as a score that represents the extent to which a product is a cyber threat. The LSTM output can be ranked to eliminate the results with low scores.

Method evaluation

In this section, we discuss the data collection process and evaluate our proposed method on the collected data. To the best of our knowledge, the collected dataset is the largest among the threat identification research on dark net marketplaces. Using this dataset, we design several experiments to evaluate the effectiveness of our model versus several state-of-the-art benchmark methods. We verify that the proposed semi-supervised labeling method improves the classification performance of supervised learning schemes such as SVM and LSTM. We also show that the proposed deep LSTM network architecture is able to leverage the labeled data to yield superior threat identification performance.

Data collection and testbed

Nine dark net marketplaces were identified, including the most popular markets at the time of this study. The markets were selected from top markets in deepdotweb.com,[1] a reputable dark net news website. Seven of these markets were in English. As analyzing multilingual content of the Dark Net Markets [17] is out of the scope of our study, we focused on the English markets to construct our testbed. A web crawler was designed and

implemented for traversing the onion links (a common type of network address in hidden web) in a breadth-first manner. The crawler incorporates several techniques to circumvent anti-crawling measures in dark net marketplaces. The crawler can resume the crawling process by loading the state in which it is terminated. It is also capable of waiting for a user's response to access CAPTCHA-protected contents. Finally, it leverages random waits between retrievals to avoid revealing any discoverable pattern on web servers. We store the raw data obtained by the crawler in a local file system for further version tracking and indexing purposes as suggested in Benjamin et al. [6].

Data pre-processing

The DNM webpages stored in the file system are parsed to obtain product descriptions, product reviews, vendor descriptions, and vendor reviews. Parsed documents are stored in a relational database. Product descriptions contain the major cues about threats in dark net marketplaces. Therefore, we use product descriptions to build our research testbed. The product descriptions were pre-processed by tokenization, character normalization (e.g., converting to lower-case), and conversion to UTF-8 encoding [29]. These pre-processing steps aim to ensure that arbitrary ways of writing the same word will be treated identically and the text is suitable for input into the subsequent learning algorithms. Table 3 summarizes our testbed obtained by applying the described pre-processing steps. The dataset is available at https://github.com/mohammadrezaebrahimi/JMIS-DarkNetMarketData.

In the processing step, null and non-valid values, as well as duplicate products cross-posted on different markets, are eliminated. Table 4 shows the label distribution of the dataset after preprocessing.

Experiments

Given the aforementioned dataset, we designed three main categories of experiments in order to evaluate our proposed method. The first experiment includes the state-of-the-art

Table 3. DNM dataset.

Market Name	No. of Products	No. of Sellers	Collection Date
Valhalla	12,192	497	Oct. 21, 2016
Dream Market	25,602	723	Jan. 05, 2017
Hansa	14,149	513	Nov. 18, 2 016
Alphabay	25,118	1,577	Jan. 12, 2017
Minerva	683	N/A	Nov. 28, 2016
SilkRoad3	813	641	Jan. 02, 2017
Apple Market	877	112	Nov. 13, 2016
Total	**79,434**	**> 4,063**	-

Table 4. Distribution of labels in the DNM dataset

Category	Description	Label	No. of Docs	Total No. of Docs	%
Labeled data	Labeled by human expert for the purpose of training via cross-validation	Pos.	464	1,687	~2.78%
		Neg.	1,223		
	Randomly selected from the entire population and labeled by human expert for the purpose of validation	Pos.	31	132	~0.22%
		Neg.	101		
Unlabeled data	The majority of the data is unlabeled.	-	58,878	58,878	~97.0%

benchmark methods used in recent cyber threat detection studies. For comprehensiveness, we did not limit this category to methods that have been used in IS discipline. These methods, which utilize only human-labeled documents in their training process, include common machine learning algorithms such as *k*-Nearest Neighbors [13], Random Forest [47], SVM [47,49], as well as advanced deep learning methods such as CNN [13], and LSTM [17]. Consistent with text classification literature, common machine learning algorithms use weighted term frequency as their input features, while deep learning methods use the raw text sequence as their input [29]. The second category uses the human-labeled data along with the unlabeled data through transductive learning. The third category pertains to our proposed approach, which leverages TSVM labels for unlabeled data in addition to the human-labeled data via LSTM. We adopt F_1-score, a widely accepted performance measure in information retrieval and text classification tasks [42], which has also been widely used in IS studies for cyber threat detection [38, 49]. Table 5 summarizes the evaluation results. Statistical significance of the evaluations was conducted via pair-wise *t*-test between the performance obtained from the proposed method and each benchmark method (Table 6).

In our experiment environment, the University of Arizona's high-performance computing cluster, El-Gato, was used for running the proposed LSTM network on NVIDIA Kepler K20X GPU with 2,688 CUDA cores and 6GB of memory. Sindhwani's implementation[2] of TSVM was used for TSVM tests. The LSTM implementation in the Context3.0 library [27] was used for LSTM tests. In terms of computational burden,

Table 5. Performance comparison of SVM and LSTM with/without using TSVM labels.

Experiment Category	Method	Parameter Settings*	Learning Scheme	Performance on Validation Set (%)			
				Accuracy	Precision	Recall	F_1-Score
Without semi-supervised labeling	*k*-NN	Number of neighbors: 3	Supervised	81.82	64.00	51.61	57.14
	LR	Regularization Parameter: 1000	Supervised	87.88	68.29	90.32	77.78
	Random Forest	Number of trees: 1500, Max tree depth: 120	Supervised	87.88	67.44	93.55	78.38
	SVM	Nu = 0.1, kernel = linear	Supervised	91.67	76.32	93.55	84.09
	CNN	# of convolutional layers = 2, # of neurons: 64 and 128, # of max pooling layers = 2, activation = ReLU	Supervised	89.39	69.77	96.77	81.08
	LSTM	Number of cells: 500, activation: Tanh, layer type: bidirectional	Supervised	91.67	75.00	96.77	84.51
Direct use of unlabeled data by TSVM	TSVM (Sindhwani, 2006)	Optimization method = Deterministic Annealing, ratio of positive samples in unlabeled data = 0.28, regularization parameter = 0.001	Semi-supervised	93.18	77.50	**100.00**	87.32
With semi-supervised labeling	TSVM+LSTM (Our proposed approach)	Number of cells: 100, activation: Tanh, layer type: bidirectional	Semi-supervised	**94.70**	**83.33**	96.77	**89.55**

Note: Only parameters that led to the best results for each method are shown. Numbers in bold are the best performance. *k*-NN, *k*-Nearest Neighbors; LR, Logistic Regression; SVM, Support Vector Machine; CNN, Convolutional Neural Network; LSTM, Long Short-Term Memory; TSVM, Transductive Support Vector Machine.

Table 6. P-values obtained from *t*-test on F_1-score, accuracy, precision, and recall for comparing the proposed method (LSTM + SVM) against the benchmark methods.

	k-NN	LR	Random Forest	SVM	CNN	LSTM	TSVM
F_1-score	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***
Accuracy	< 0.001***	< 0.001***	< 0.001***	< 0.001***	0.003**	0.006**	0.009**
Recall	< 0.001***	< 0.001***	< 0.001***	< 0.001***	-	-	-
Precision	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***	< 0.001***

Note: - The corresponding method has a higher or equal value, and thus the null hypothesis cannot be rejected.
*$p < 0.05$. **$p < 0.01$. ***$p < 0.001$.

training the proposed model involves both CPU (for TSVM) and GPU (for LSTM). Conducting TSVM for more than 60,000 data points in our dataset requires less than 2GB of memory for less than 0.2 hours on a 3.3 GHz Intel CPU. Conducting the LSTM training requires less than 3GB of memory for less than 0.3 hours on a GPU with the aforementioned specifications. The prediction time for each data point is in the order of milliseconds suggesting that the trained model can be used in real-time settings. Based on this empirical analysis, the computational cost of training and prediction is highly affordable with commodity hardware in a timely manner.

When semi-supervised labeling is not used (i.e., the baseline), LSTM (F_1 = 84.51%) outperforms linear SVM (F_1 = 84.09%). Also, leveraging the unlabeled data through Algorithm 1 improves the classification performance by almost 3% (from 84.09% to 87.32%). In addition, using the described LSTM architecture on the results obtained from algorithm 1 leads to another two-percent improvement (from 87.32% to 89.55%). The results of conducting *t*-test on multiple runs (n = 10) of our algorithm against the benchmark methods show that the performance gain is statistically significant (Table 6). It is observed that using the proposed semi-supervised labeling improves the classification performance of both SVM and LSTM by approximately 3% and 5%, respectively. Note that since the same random seeds were used for cross-validation in all experiments, the same data instances were chosen in each fold across all experiments. Otherwise, the results may not be comparable due to the inconsistent assignment of different samples to each fold. It is also worth mentioning that when the LSTM uses the labeled data obtained by the aforementioned semi-supervised labeling, the number of required memory cells decreases from 500 to 100, implying that the proposed method facilitates learning with simples models.

We note that cyber threat detection in DNMs often entails a non-severe form of data imbalance (e.g., the ratio of positive to negative instances is 1 to 3 in our case). Although we found that this does not prevent the algorithms from leaning in our application domain, it is very important to recognize that accuracy cannot be a viable performance measure in this domain due to data imbalance. As a result, scholars have suggested using the harmonic mean of precision and recall, F_1-score, which is not sensitive to the data imbalance [38, 49].

To gain a better insight into the performance improvement in the presence and absence of unlabeled data, we compute and compare the area under the receiver operating characteristic (ROC) curve in addition to F_1-scores. Figure 10 compares the performance of LSTM networks with and without semi-supervised labeled data in terms of the area under the ROC curve. The ROC curve indicates the true positive rate versus the false positive rate and is a common method of evaluating the performance of a binary classifier. Ideally, the area under curve should be close to 1. By leveraging the

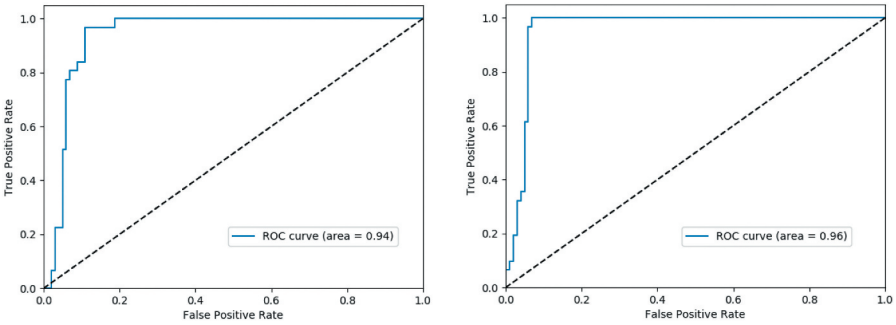

Figure 10. The area under ROC curve for LSTM Network without (left, 0.94) and with (right, 0.96) semi-supervised labeling. The area has increased by 2% through using the proposed labeling method.

unlabeled data, the area under curve has increased from 0.94 to 0.96. Considering the relatively large number of products available in DNMs (hundreds of thousands), and the potential cost caused by these threats, 3% and 2% increase in the classification accuracy and AUC can potentially translate to saving millions of dollars for companies and individuals whose systems and customers can be affected by these threats if not discovered and mitigated in early stages.

Finally, to investigate the quality of the training process in LSTM we illustrate the changes in train/test error versus *the number of iterations* as a parameter of LSTM learning. Figure 11 compares the reduction of squared error with and without using semi-supervised labeling. In the absence of semi-supervised labeling, the ultimate test error is about 0.1, while it reduces to approximately 0.05 by using the semi-supervised labeling method. This error reduction during training can signify the increase in classification accuracy using the proposed method, which implies the prevention of a sizable financial loss in the long term.

Practical considerations for model training and parameter selection

In this section, we discuss significant practical considerations in model training. It is critical that machine learning models are generalizable and perform well on unseen data that is not provided in the training process. To ensure the generalizability of the model, it is important to

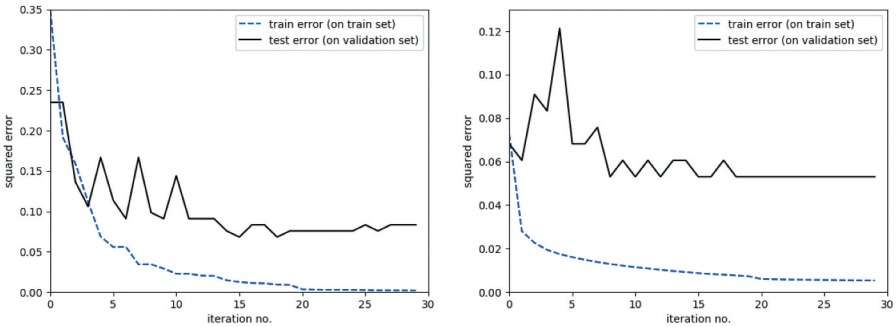

Figure 11. The test error for LSTM Network without (left) and with (right) semi-supervised labeling. The final test error reduced from 0.1 to 0.05 in the presence of semi-supervised labeling.

tune the parameters (e.g., the kernel choice in SVM, learning rate in LSTM) as well as the hyper-parameters (e.g., number of neurons and type of activation functions in LSTM) in a process known as 'rotation estimation' or 'k-fold cross-validation' [33, 48]. This process involves a systematic choice of $k-1$ training and one testing sets to avoid overfitting (i.e., memorizing the input patterns). Consistent with machine learning and cyber threat detection literature [49], in our experiments, models were trained using 10-fold cross-validation. Within machine learning literature, Kohavi (1995) shows that $k=10$ is the best choice for ensuring the generalizability even if computation power permits choosing higher values for k. It is also shown that choosing k to be smaller than 10 could lead to obtaining biased models (i.e., with low generalizability). One hyper-parameter that can drastically affect the quality of training is the type of activations for neurons in the LSTM architecture. Activations are non-linear functions that determine whether a neuron should activate during the training process. While a large variety of activation functions exist, two families of activations are most commonly used in deep learning tasks [20]: 1) Rectified Linear family, including traditional ReLu, and Leaky ReLu, and 2) logistic family, including Tanh and Sigmoid. While the first family of activations has shown promise in image processing and computer vision applications, the activations in the second family are more common for recurrent deep architectures that process sequential data such as LSTM. Among the logistic family, Tanh is often preferred over sigmoid in LSTM architectures [20] due to its ability in carrying the gradients (partial derivative of the error) in long input sequences. This property is helpful in avoiding a common learning issue known as "vanishing gradients" that often occurs with sigmoid activation. Given the results form cross-validation, we found that Tanh activation function provides the best accuracy in our case, which is consistent with the literature.

Discussions and empirical examples

To demonstrate the benefits of the proposed method in practice, we demonstrate the output of our model and compare it to our baseline for several examples sampled from the validation set. Table 7 shows two examples with financial themes that have been successfully identified as threats or detrimental products. The first product is a carding tool that facilitates making fake credit/debit cards in order to withdraw cash from victims' accounts. The second product is a collection of breached bank account information from one of the largest banks in the United States. The closer the predicted value for a product is to 1, the more likely that it is a detrimental product.

Even though one can interpret any predicted value greater than 0.5 as a potential threat, a good model would assign higher (closer to one) predicted values to highly potential threats and lower predicted values (closer to zero) to non-threats. The above examples show that the model is able to assign numbers close to 1 to highly potential cyber threats such as bank accounts' information and carding tools, which is a desirable outcome in this application domain.

To further illustrate the benefits of the proposed method in practice, we investigate the examples that our proposed model is able to identify correctly while the SVM model cannot. Table 8 illustrates examples of detrimental products that SVM cannot identify, but our method is able to identify.

Table 7. Examples of correctly identified detrimental products. Predicted values close to 1 indicate higher probability of being a threat.

Product Description Excerpts	Explanation	Predicted Value
"THE CARDER'S HOLY GRAIL -Antidetect 6.5 -Fraud Browser -Socks5/Proxy+VPN -List of Bins -Worth $550! This is what you need if you plan on making any money from carding. If you want to be a Professional Carder and raise your success rate 100% then you need this software. This software will make you card like a professional and you will be able to have a much higher accept rate. Stop wasting money!!!!! Stop Wasting your cards!!!! […]"	The seller offers a customized "safe" tool that is specifically designed to conduct credit card fraudulent activities.	0.83
"*Paypal* Bank of America Account Logins $500+ balance Once payment has been made and confirmed, you will receive these details: Username and password, account name, last 4 digits of account number, summary. […] HIGHLY recommended to use the account with the linking and instantly confirming to a PP account (must be a USA based PP) and adding $ from the account to PP or checking out using PP and the bank account as payment method (if the PP is setup correctly for this feature)"	The vendor provides breached bank accounts. Cyber criminals can monetize the information via performing fraudulent transactions.	0.87

Table 8. Examples of threats that the baseline method cannot identify but the proposed method is able to identify.

Error Type	Product Description Excerpts	Explanation	True Category	SVM's Response	Our Method's Response
False negative	"Basic Carding for Newbies Almost Free By this guide I hope most of beginners will find Answers of their most main questions."	A beginner's guide on how to do fraudulent activities on bank cards and personal bank accounts.	Threat	Non-threat	Threat
	"Advance Hacking Exposed Part 6"	A collection of hacking tools and guides.	Threat	Non-threat	Threat

As seen in Table 8, the proposed approach can identify threats that have very short descriptions. Short documents are problematic for typical bag-of-words approaches to classify due to the very low dimension of the input and the high dimensionality of vector space. The proposed method mitigates this problem by leveraging unlabeled data as well as using bidirectional LSTM. The proposed bidirectional LSTM extracts more information from short documents compared to simple bag-of-words approaches. Table 9 illustrates examples of non-detrimental products that SVM wrongly identifies as threats, but our method is able to correctly identify as non-threats.

The example shown in Table 9 illustrates that while the occurrence of tokens such as "KB" and "pdf" misleads the baseline method to determine as a detrimental product, the proposed method is able to correctly classify the document as a non-threat.

Table 9. Example of non-detrimental products that the baseline method wrongly identifies as threats but the proposed method is able to correctly identify.

Error Type	Product Description Excerpts	Explanation	True Category	SVM's Response	Our Method's Response
False positives	"DO IT YOURSELF GUIDE PACK 700+ 773 KB → 30 Quick Fixes For Everyday Disasters.pdf 837 KB → 4x8 Utility Trailer-Drawings.pdf 337 KB → 4x8 Utility Trailer-Instructions.pdf 233 KB → A Guide To Building Outdoor Stairs.pdf […]"	A collection of legal "Do It Yourself" e-books	Non-threat	Threat	Non-threat

Managerial implications and implementation considerations

Our proposed method can be utilized in two main categories of organizations to help fulfill their business missions: 1) Cyber threat intelligence organizations and 2) Identity theft protection companies. As an integral part of their mission, cyber threat intelligence companies (e.g., FireEye, IBM, and Symantec) monitor the content of the deep web on a daily basis to identify and triage the potential cyber threats. Manual threat identification is time-intensive, consuming many hours of valuable human resources (i.e., analysts). Similarly, identity theft protection companies (e.g., LifeLock, IdentityGuard, and Experian) tackle monitoring the dark web in order to detect and prioritize potential incidents of identity theft. At the managerial level, our method is an important step towards lowering the human supervision cost in realizing automated threat detection within cyber threat intelligence organizations and identity theft protection companies. Overall, given the large number of products available in DNMs (hundreds of thousands), and the potential damage caused by these threats, the reported increase in the detection performance can potentially translate to saving millions of dollars for companies and individuals whose systems and customers can be compromised by these threats if not discovered and mitigated early on.

To provide insight into the implementation and deployment of our approach within these organizations, we highlight several important considerations. Although implementing the presented semi-supervised method can significantly help to alleviate the need for manual data labeling, it is important to note that machine learning methods often have limitations in their performance if the data pattern changes over time. In consequence, these methods need to be re-trained to update their parameters or hyper-parameters in the course of time. Consequently, it is challenging to devise machine learning methods that are able to adjust their parameters in a dynamic manner in order to remain effective. We also note that while in practice, it is possible to replace the LSTM component of our method with traditional machine learning classifiers for which enough background knowledge exists within the organization, based on the experiments, we recommend using a family of methods that are able to account for time dependency in encountering sequential data obtained from textual contents in the dark web.

Conclusion and future directions

Given the increasing popularity of online marketplaces, anonymous dark net marketplaces have continued to evolve during the past few years. They serve as a promising source of intelligence for proactive cyber threat intelligence. Early and effective identification of cyber threats hosted by DNMs helps to avoid significant financial loss at organization and individual level.

We proposed a new transductive and deep learning approach that leverages the structural and lexical characteristics of DNMs to conduct transductive learning, which significantly reduces the need for manual labeling in this critical application domain. Our proposed approach improves cyber threat detection by reducing the number of falsely identified threats as well as the number of missing threats while using minimum human-labeled data. We showed that training an LSTM network with semi-supervised labeling yields state-of-the-art performance in this domain.

The empirical evaluation of our method on a DNM dataset reveals that unlike the recent semi-supervised labeling technique (co-training) used for threat identification in the literature, our method increases both precision and recall simultaneously. Also thanks to transductive learning, our method effectively uses less than 3% human-labeled documents, while the state-of-the-art co-training method uses almost 25% human-labeled data. The managerial implications and implementation considerations were discussed. We believe the idea of using semi-supervised data labeling to improve the classification performance of supervised models and minimize manual labeling provides promising avenues for future research seeking to classify large volumes of documents extracted from the dark web. However, devising machine learning methods that are able to adjust their parameters in a dynamic manner still remains a challenging task in this application domain. Another promising research direction is developing approaches to analyze non-English dark web platforms as they can differ in cyber threats and hacker assets they provide.

Notes

1. https://www.deepdotweb.com
2. http://vikas.sindhwani.org/svmlin.html

Funding

This material is based upon work supported by the National Science Foundation under Grant No. NSF 1936370 (SaTC).

References

1. Anderson, B.B.; Vance, A.; Kirwan, C.B.; Jenkins J.L.; and Eargle, D. From warning to wallpaper: Why the brain habituates to security warnings and what can be done about it. *Journal of Management Information Systems*, *33*, 3 (2016), 713–743.
2. Angst, C.M.; Block, E.S.; D'arcy, J.; and Kelley, K. When do IT security investments matter? Accounting for the influence of institutional factors in the context of healthcare data breaches. *MIS Quarterly*, *41*, 3 (2017), 893–916.
3. Benaroch, M. Real options models for proactive uncertainty-reducing mitigations and applications in cybersecurity investment decision making. *Information Systems Research*, *29*, 2 (2018), 315–340.

4. Benjamin, V.; and Chen, H. Developing understanding of hacker language through the use of lexical semantics. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, Baltimore, MD, 2015, pp. 79–84.
5. Benjamin, V.; and Chen, H. Identifying language groups within multilingual cybercriminal forums. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, Tucson, AZ, 2016, pp. 205–207.
6. Benjamin, V.; Li, W.; Holt, T.; and Chen, H. Exploring threats and vulnerabilities in hacker web: Forums, IRC and carding shops. In *IEEE International Conference on Intelligence and Security Informatics (ISI), 2015*. Baltimore, MD, USA: IEEE, 2015, pp. 85–90.
7. Benjamin, V.; Valacich, J.S.; and Chen, H. DICE-E: A Framework for Conducting Darknet Identification, Collection, Evaluation with Ethics. *MIS Quarterly*, *43*, 1 (2019), 1–22.
8. Benjamin, V.; Zhang, B.; Nunamaker Jr, J.F.; and Chen, H. Examining hacker participation length in cybercriminal Internet-relay-chat communities. *Journal of Management Information Systems*, *33*, 2 (2016), 482–510.
9. Bilge, L.; and Dumitras, T. Before we knew it: An empirical study of zero-day attacks in the real world. In *Proceedings of the ACM Conference on Computer and Communications Security*. Raleigh, NC, USA: ACM, 2012, pp. 833–844.
10. Chandrasekar, K.; Cleary, G.; Cox, O.; Lau, H.; Nahorney, B.; Gorman, B.O.; O'Brien, D.; and Wallace S. *Internet Security Threat Report*. Symantec Corporation, 2017. Available at https://www.symantec.com/content/dam/symantec/docs/reports/istr-22-2017-en.pdf (accessed on January 17, 2020)
11. Chen, Y.; and Zahedi, F.M. Individuals' Internet security perceptions and behaviors: Polycontextual contrasts between the United States and China. *MIS Quarterly*, *40*, 1 (2016), 205–222.
12. Cho, K.; Van Merriënboer, B.; Bahdanau, D.; and Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. In *Empirical Methods in Natural Language* Processing *(EMNLP)*. Doha, Qatar, 2014.
13. Deliu, I.; Leichter, C.; and Franke, K. Extracting cyber threat intelligence from hacker forums: Support vector machines versus convolutional neural networks. In *International Conference on Big Data*. Boston, MA: IEEE, 2017, pp. 3648–3656.
14. Dos Santos, C.N.; and Gatti, M. Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. In *International Conference on Computational Linguistics (COLING),* Dublin, Ireland. 2014, pp. 69–78.
15. Du, P.-Y.; Zhang, N.; Ebrahimi, M.; Samtani, S.; Lazarine, B.; Nolan, A.; Dunn, R.; Suntwal, S.; Angeles, G.; Schweitzer, R.; and Chen, H. Identifying, collecting, and presenting hacker community data: Forums, IRC, carding shops, and DNMs. In *IEEE International Conference on Intelligence and Security Informatics (ISI),* pp. 70–75. IEEE, Miami, FL, 2018.
16. Ebrahimi, M.; Suen, C.Y.; and Ormandjieva, O. Detecting predatory conversations in social media by deep Convolutional Neural Networks. *Digital Investigation*, *18*, (2016), 33–49.
17. Ebrahimi, M.; Surdeanu, M.; Samtani, S.; and Chen, H. Detecting cyber threats in non-english dark net markets: A cross-lingual transfer learning approach. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*. Miami, Florida: IEEE, 2018, pp. 85–90.
18. Farnham, G.; and Leune, K. Tools and standards for cyber threat intelligence projects. *SANS Institute. Extraído el*, *5*, (2013).
19. Gazet, A. Comparative analysis of various ransomware virii. *Journal in Computer Virology*, *6*, 1 (2010), 77–90.
20. Goldberg, Y. Neural network methods for natural language processing. *Synthesis Lectures on Human Language Technologies*, *10*, 1 (2017), 1–309.
21. Grisham, J.; Samtani, S.; Patton, M.; and Chen, H. Identifying mobile malware and key threat actors in online hacker forums for proactive cyber threat intelligence. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, Beijing, China, 2017, pp. 13–18.

22. Hochreiter, S.; and Schmidhuber, J. Long short-term memory. *Neural computation*, *9*, 8 (1997), 1735-1780.
23. Hui, K.-L.; Kim, S.H.; and Wang, Q.H. Cybercrime deterrence and international legislation: evidence from distributed denial of service attacks. *MIS Quarterly*, *41*, 2 (2017), 497–523.
24. Hui, K.L.; Vance, A.; and Zhdanov, D. Securing digital assets. In A. Bush and A. Rai (eds.), *MIS Quarterly Research Curations*. online, 2016. Accessed on July, 26, 2020: https://www.misqresearchcurations.org/blog/2017/5/10/securing-digital-assets-1
25. Jensen, M.L.; Dinger, M.; Wright, R.T.; and Thatcher, J.B. Training to mitigate phishing attacks using mindfulness techniques. *Journal of Management Information Systems*, *34*, 2 (2017), 597–626.
26. Joachims, T. Transductive inference for text classification using support vector machines. In *International Conference on Machine Learning (ICML)*. 1999, pp. 200–209.
27. Johnson, R. CONTEXT v3: Convolutional neural networks and LSTM for text categorization in C++ on GPU. *CONTEXT v3*, 2017. Available at http://riejohnson.com/cnn_download.html (accessed on January 18, 2020)
28. Johnson, R.; and Zhang, T. Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. In North American Chapter of the Association for Computational Linguistics (*NAACL*). Denver, Colorado, 2015.
29. Johnson, R.; and Zhang, T. Supervised and semi-supervised text categorization using LSTM for region embeddings. In *International Conference on Machine Learning (ICML)*. New York City, NY, 2016, pp. 526–534.
30. Karhu, K.; Gustafsson, R.; and Lyytinen, K. Exploiting and defending open digital platforms with boundary resources: Android's five platform forks. *Information Systems Research*, *29*, 2 (2018), 479–497.
31. Kharraz, A.; Robertson, W.; Balzarotti, D.; Bilge, L.; and Kirda, E. Cutting the Gordian knot: A look under the hood of ransomware attacks. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 2015, pp. 3–24.
32. Kim, Y. Convolutional neural networks for sentence classification. In *Empirical Methods in Natural Language Processing* (*EMNLP),* pp. 1746–1751. Doha, Qatar, 2014.
33. Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press, Montreal, Canada, 1995, pp. 1137–1145.
34. Lai, S.; Xu, L.; Liu, K.; and Zhao, J. Recurrent convolutional neural networks for text classification. In *Association for the Advancement of Artificial Intelligence* (*AAAI)*. Austin, Texas, 2015, pp.2267–2273.
35. Larochelle, H.; and Bengio, Y. Classification using discriminative restricted Boltzmann machines. In *Proceedings of the 25th International Conference on Machine Learning*. Helsinki, Finland: ACM, 2008, pp. 536–543.
36. LeCun, Y.; Bengio, Y.; and Hinton, G. Deep learning. *Nature*, *521*, 7553 (2015), 436–444.
37. Li, W.; and Chen, H. Identifying top sellers in underground economy using deep learning-based sentiment analysis. In *Intelligence and Security Informatics Conference (JISIC)*. Hague, Netherlands: IEEE, 2014, pp. 64–67.
38. Li, W.; Chen, H.; and Nunamaker Jr, J.F. Identifying and profiling key sellers in cyber carding community: AZSecure text mining system. *Journal of Management Information Systems*, *33*, 4 (2016), 1059–1086.
39. Liu, P.; Qiu, X.; and Huang, X. Recurrent neural network for text classification with multi-task learning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. AAAI Press, New York, NY, 2016, pp. 2873–2879.
40. Liu, T. A novel text classification approach based on deep belief network. In *International Conference on Neural Information Processing*. Vancouver, Canada: Springer, 2010, pp. 314–321.
41. Liu, X.; Zhang, B.; Susarla, A.; and Padman, R. Go to YouTube and call me in the morning: Use of socialmedia for chronic conditions. *MIS Quarterly*, *44*, 1 (2020), 257–283.

42. Manning, C.D.; Raghavan, P.; and Schütze, H. *Introduction to Information Retrieval*. New York: Cambridge University Press, 2008.
43. McAfee. What is a Keylogger? *McAfee*, 2013. https://www.mcafee.com/blogs/consumer/family-safety/what-is-a-keylogger (accessed on January 18, 2020)
44. Moody, G.D.; Siponen, M.; and Pahnila, S. Toward a unified model of information security policy compliance. *MIS Quarterly*, *42*, 1 (2018), 285–311.
45. Nunes, E.; Diab, A.; Gunn, A.; Marin, E.; Mishra, V.; Paliath, V.; Robertson, J.; Shakarian, J.; Thart, A.; Shakarian, P. Darknet and deepnet mining for proactive cybersecurity threat intelligence. In *IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, Tucson, AZ, 2016, pp. 7–12.
46. Odabas, M.; Holt, T.J.; and Breiger, R.L. Governance in online stolen data markets. In *The Architecture of Illegal Markets: Towards an Economic Sociology of Illegality in the Economy,* edited by Jens Beckert and Matías Dewey. Oxford University Press. Retrieved 26 Jul. 2020, from: https://www.oxfordscholarship.com/view/10.1093/oso/9780198794974.001.0001/oso-9780198794974-chapter-5.
47. Portnoff, R. The Dark Net: De-Anonymization, classification and analysis. 2018. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-5.html (accessed on January 18, 2020)
48. Refaeilzadeh, P.; Tang, L.; and Liu, H. Cross-validation. In *Encyclopedia of Database Systems,* edited by LIU L., ÖZSU M.T. Springer, Boston, MA.
49. Samtani, S.; Chinn, R.; Chen, H.; and Nunamaker Jr, J.F. Exploring emerging hacker assets and key hackers for proactive cyber threat intelligence. *Journal of Management Information Systems*, *34*, 4 (2017), 1023–1053.
50. Schäfer, M.; Fuchs, M.; Strohmeier, M.; Engel, M.; Liechti, M.; and Lenders, V. BlackWidow: Monitoring the dark web for cyber security information. In *11th International Conference on Cyber Conflict (CyCon)*. Tallinn, Estonia: IEEE, 2019, pp. 1–21.
51. Schuster, M.; and Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*, 11 (1997), 2673–2681.
52. Shao, S.; Tunc, C.; Al-Shawi, A.; and Hariri, S. Autonomic author identification in internet relay chat (IRC). In *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*. Aqaba, Jordan: 2018, pp. 1–8.
53. Sindhwani, V.; and Keerthi, S.S. Large scale semi-supervised linear SVMs. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2006, pp. 477–484.
54. Sindhwani, V.; Niyogi, P.; Belkin, M.; and Keerthi, S. Linear manifold regularization for large scale semi-supervised learning. In *Proceedings of the 22nd ICML Workshop on Learning with Partially Classified Training Data*. Bonn, Germany, 2005, pp. 80-83.
55. Socher, R.; Perelygin, A.; Wu, J.Y.; Chuang, J.; Manning, C.; Ng, A.; Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Seattle, USA: CiteseerX, 2013, pp. 1631–1642.
56. Sun Yin, H.H.; Langenheldt, K.; Harlev, M.; Mukkamala, R.R.; and Vatrapu, R. Regulating cryptocurrencies: a supervised machine learning approach to de-anonymizing the bitcoin blockchain. *Journal of Management Information Systems*, *36*, 1 (2019), 37–73.
57. Tang, D.; Qin, B.; and Liu T. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2015, pp. 1422–1432.
58. Tavabi, N.; Goyal, P.; Almukaynizi, M.; Shakarian, P.; and Lerman, K. Darkembed: Exploit prediction with neural language models. In *AAAI Conference on Artificial Intelligence*. New Orleans, Louisiana, 2018, pp. 7849-7854.
59. Temizkan, O.; Park, S.; and Saydam, C. Software diversity for improved network security: Optimal distribution of software-based shared vulnerabilities. *Information Systems Research*, *28*, 4 (2017), 828–849.
60. The Economist. Shedding light on the dark web. 2016. https://www.economist.com/comment/3188575 (accessed on January 17, 2020)

61. Vance, A.; Jenkins, J.L.; Anderson, B.B.; Bjornn, D.K.; and Kirwan, C.B. Tuning out security warnings: A longitudinal examination of habituation through fMRI, eye tracking, and field experiments. *MIS Quarterly*, *42*, 2 (2018), 355–380.
62. Wang, J.; Li, Y.; and Rao, H.R. Coping responses in phishing detection: An investigation of antecedents and consequences. *Information Systems Research*, *28*, 2 (2017), 378–396.
63. Wang, J.; Shen, X.; and Pan, W. On transductive support vector machines. *Contemporary Mathematics*, *443*, (2007), 7–20.
64. Wolff, J. Perverse effects in defense of computer systems: When more is less. *Journal of Management Information Systems*, *33*, 2 (2016), 597–620.
65. Xie, J.; Liu, X.; Zeng, D.; and Fang, X. Understanding reasons for medication nonadherence: an exploration in social media using sentiment-enriched deep learning approach. In *International Conference on Information Systems (ICIS)*. Seoul, South Korea, 2017, pp. 1-11.
66. Xie, J.; and Zhang, B. Readmission risk prediction for patients with heterogeneous hazard: A trajectory-aware deep learning Approach. In *International Conference on Information Systems (ICIS)*. 2018.
67. Yang, S.; Hsu, C.; Sarker, S.; and Lee, A.S. Enabling effective operational risk management in a financial institution: An action research study. *Journal of Management Information Systems*, *34*, 3 (2017), 727–753.
68. Yue, W.T.; Wang, Q.; and Hui, K.L. See no evil, hear no evil? Dissecting the impact of online hacker forums. *MIS Quarterly*, *43*, 1 (2019), 73–95.
69. Zhang, X.; Zhao, J.; and LeCun, Y. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2015, pp. 649–657.
70. Zhou, D.; Bousquet, O.; Lal, T.N.; Weston, J.; and Schölkopf, B. Learning with local and global consistency. In *Neural Information Processing Systems* (*NeurIPS)*. 2003, pp. 321–328.
71. Zhou, S.; Chen, Q.; Wang, X.; and Li, X. Hybrid deep belief networks for semi-supervised sentiment classification. In *COLING*. CiteseerX, 2014, pp. 1341–1349.
72. Zhu, X.; and Goldberg, A. Introduction to semi-supervised learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *3*, 1 (2009), 1–130.

About the Authors

Mohammadreza Ebrahimi (ebrahimi@email.arizona.edu) is a doctoral student in the Department of Management Information Systems and a research associate in the Artificial Intelligence Lab at the University of Arizona. He received his Master's degree in Computer Science from Concordia University, Canada. His research interests include statistical machine learning, natural language processing, social media analytics, and text mining. His work has appeared in journals, including *Digital Forensics, Applied Artificial Intelligence*, and was presented at IEEE ISI conference, as well as in a chapter of *Data Mining Trends and Applications in Criminal Science and Investigations*.

Jay F. Nunamaker Jr. (jnunamaker@cmi.arizona.edu) is Regents and Soldwedel Professor of MIS, Computer Science and Communication, and director of the Center for the Management of Information and the National Center for Border Security and Immigration at the University of Arizona. He received his Ph.D. in Operations Research and Systems Engineering from Case Institute of Technology. Dr. Nunamaker has held a professional engineer's license since 1965. He was inducted into the Design Science Hall of Fame and received the LEO Award for Lifetime Achievement from the Association for Information Systems. He was featured in the July 1997 issue of *Forbes Magazine* on technology as one of eight key innovators in information technology. His specialization is in the fields of system analysis and design, collaboration technology, and deception detection. The commercial product GroupSystems ThinkTank, based on his research, is often referred to as the gold standard for structured collaboration systems. He founded the MIS Department at the University of Arizona and served as department head for 18 years.

Hsinchun Chen (hchen@eller.arizona.edu) is Regents Professor and Thomas R. Brown Chair in Management and Technology at the Eller College of Management, University of Arizona. He received his Ph.D. in Information Systems from New York University. He is author or editor of 20 books, 300 journal papers, and 200 refereed conference articles covering digital library, data/text/web mining, business analytics, security informatics, and health informatics. He served as the lead Program Director of the Smart and Connected (SCH) Program at the National Science Foundation (NSF). Dr. Chen founded the Artificial Intelligence Lab at University of Arizona, which has received $50M+ research funding from the NSF, National Institutes of Health, National Library of Medicine, Department of Defense, Department of Justice, Central Intelligence Agency, Department of Homeland Security, and other agencies. He is a Fellow of ACM, IEEE, and AAAS.

Appendix 1

Here we show that the proposed approach to estimate the value of parameter r in TSVM can provide the lower bound for the optimal value of r. This lower bound serves as a guide to avoid choosing non-viable values for r in practice.

Assumptions

Assume heuristics $H_1$and H_2 partition the unlabeled dataset D_U and the negative documents classified by H_1 and H_2 have zero false negative and false positive rates, respectively. Let H_1 and H_2 be applied on the dataset in the same order described in steps 3 and 4 of Algorithm 1; also, assume parameter λ' was tuned and fixed by cross-validation.

Proposition

Let $r^* = \frac{Pos^*}{Neg^*+Pos^*}$ be the optimal value of parameter r in which Pos^* and Neg^* denote the true number of positive samples and negative samples in the unlabeled data, respectively. Also, let $Pos_{H2} = \left|D_{H2}^{pos}\right|$, $Neg_{H1} = \left|D_{H1}^{neg}\right|$, and $Neg_{H2} = \left|D_{H2}^{neg}\right|$ denote the number of positive/negative samples identified by each heuristic. The estimated value $\hat{r} = \frac{Pos_{H2}}{Neg_{H1}+Neg_{H2}+Pos_{H2}}$ is a lower bound for r. That is, $\hat{r} \leq r^*$; $\hat{r} \in (0, 1]$.

Proof

Under the assumption that does not produce any false negatives and when H_2 does not produce any false positives, the negative documents classified by H_2 (i.e., Neg_{H2}) contain some false negatives, $\alpha \geq 0$ and, thus, contain the true number of negative samples, $Neg^* = Neg_{H1} + Neg_{H2} - \alpha$. That is,

$$Neg_{H2} = Neg^* - Neg_{H1} + \alpha \tag{4}$$

Conversely, false negatives α are ideally supposed to be assigned to D_{H2}^{pos}. That is,

$$Pos_{H2} = Pos^* - \alpha \tag{5}$$

By plugging (2) and (3) into the given ratio for $\hat{r}$ in the proposition, we have:

$$\hat{r} = \frac{Pos^* - \alpha}{Neg_{H1} + \ Neg^* - Neg_{H1} + \alpha + Pos^* - \alpha} = \frac{Pos^* - \alpha}{Neg^* + Pos^*} \leq r^* \tag{6}$$