Identifying and Categorizing Malicious Content on Paste Sites: A Neural Topic Modeling Approach

Tala Vahedi
Department of Management Information Systems
University of Arizona
Tucson, AZ, USA
talavahedi@email.arizona.edu

Benjamin Ampel
Department of Management Information Systems
University of Arizona
Tucson, AZ, USA
bampel@email.arizona.edu

Sagar Samtani
Department of Operations and Decision Technologies
Indiana University
Bloomington, IN, USA
ssamtani@iu.edu

Hsinchun Chen
Department of Management Information Systems
University of Arizona
Tucson, AZ, USA
hsinchun@arizona.edu

***Abstract*—Malicious cyber activities impose substantial costs on the U.S. economy and global markets. Cyber-criminals often use information-sharing social media platforms such as paste sites (e.g., Pastebin) to share vast amounts of plain text content related to Personally Identifiable Information (PII), credit card numbers, exploit code, malware, and other sensitive content. Paste sites can provide targeted Cyber Threat Intelligence (CTI) about potential threats and prior breaches. In this research, we propose a novel Bidirectional Encoder Representation from Transformers (BERT) with Latent Dirichlet Allocation (LDA) model to categorize pastes automatically. Our proposed BERT-LDA model leverages a neural network transformer architecture to capture sequential dependencies when representing each sentence in a paste. BERT-LDA replaces the Bag-of-Words (BoW) approach in the conventional LDA with a Bag-of-Labels (BoL) that encompasses class labels at the sequence level. We compared the performance of the proposed BERT-LDA against the conventional LDA and BERT-LDA variants (e.g., GPT2-LDA) on 4,254,453 pastes from three paste sites. Experiment results indicate that the proposed BERT-LDA outperformed the standard LDA and each BERT-LDA variant in terms of perplexity on each paste site. Results of our BERT-LDA case study suggest that significant content relating to hacker community activities, malicious code, network and website vulnerabilities, and PII are shared on paste sites. The insights provided by this study could be used by organizations to proactively mitigate potential damage on their infrastructure.**

Keywords – Paste sites, Pastebin, topic modeling, transformers, BERT, malicious content, exploit code, cyber threat intelligence

I. INTRODUCTION

Malicious hacking tools developed by cyber-criminals are becoming increasingly more complex and dangerous. Malicious cyber-activities are estimated to cost the global economy $445 billion annually [1]. Cyber-criminals often share malicious strategies and tools for launching cyber-attacks on information-sharing social media platforms [2]. A paste site is a large social media platform that allows millions of users to anonymously post large quantities of plain text content [3]. For example, Pastebin, a well-known paste site, currently has over 18 million registered users, 150 million public pastes, and over 16 million monthly visitors [5]. Personally Identifiable Information (PII), credit card numbers, exploit code, malware, and other sensitive content are often accessible on paste sites [4]. A Distributed Denial-of-Service (DDoS) attack posted publicly on Pastebin is presented in Fig. 1.

Fig. 1. Sample Pastebin Post with Metadata on Top and Malicious Source Code on Bottom for a Distributed Denial of Service (DDoS) Attack.

Each paste includes metadata such as paste title, author, post date, views, and expiration date. Below the post metadata are the plain-text post contents (e.g., DDoS at the bottom of Fig. 1), which may include various predominant keywords appearing in sequential order (e.g., "DDoS" on line 7 appears before "Attacking" on line 16 of Fig. 1). Overall, hundreds of thousands of illicit users share and contribute millions of pastes. Consequently, paste sites can serve as a valuable data source for understanding cyber-criminal tactics, techniques, and procedures (TTPs). Automatically categorizing content (e.g., extracting themes) in paste sites can assist in providing targeted Cyber Threat Intelligence (CTI) about potential cyber-threats and past cyber-breaches [3].

In this study, we propose a novel Bidirectional Encoder Representation from Transformers with Latent Dirichlet Allocation (BERT-LDA) topic model to automatically categorize pastes from large paste sites. The proposed BERT-LDA has two novelties. First, BERT-LDA leverages BERT, a prevailing neural network transformer architecture, to capture sequential dependencies within the text of a paste to represent each sentence. Second, BERT-LDA replaces the Bag-of-Words (BoW) used in the conventional LDA model

with Bag-of-Labels (BoL) that encompasses class labels at the sequence level when extracting topics.

The remainder of this paper is organized as follows. First, we review literature about paste sites, neural network-based topic models, and BERT. Second, we present our research gaps and questions. Third, we introduce our research design and BERT-LDA model. Fourth, we present our evaluation and case study results. Finally, we conclude this research and point to promising future directions for research.

II. LITERATURE REVIEW

We review three areas of literature for this study. First, we examined recent research studying paste sites to identify their objectives and prevailing methods. Second, neural network-based topic models were reviewed to identify prevailing unsupervised algorithms for automatically categorizing large quantities of text. Finally, we reviewed BERT to identify how the prevailing transformer-based model for text analysis could be incorporated into a topic model.

A. Paste Site Analysis

Paste sites allow users to freely share text online anonymously. Unlike other hacker social media platforms (e.g., forums) a paste site is a repository to store plain-text only (i.e., no multi-media). Recent literature analyzing paste sites have focused on using a single paste site as a data testbed [4], [6-9]. These studies have focused on identifying leaked PII (e.g., emails, passwords, phone numbers, credit cards, and secret keys) based on the metadata associated with each paste, and not the paste content [4], [6-10]. As a result, the knowledge about the major categories of content on paste sites is mostly unexplored and unknown. When choosing analytical methods, recent literature has primarily relied on classical machine learning algorithms [4], manual labeling and analysis [4][6], and theoretical methods [7][9][10] to analyze each paste's metadata and execute their research objectives. Such methods often cannot reveal the major themes (i.e., topics, categories) of content within paste sites. Since topic modeling is a common approach to automatically categorize large quantities of unstructured text [3], we review prevailing neural network-based topic models next.

B. Neural Network-based Topic Models

Topic modeling is a common approach to categorize text in social media platforms [3]. LDA is the prevailing unsupervised topic modeling approach for categorizing text in hacker social media platforms where there is limited prior knowledge [3][11]. LDA is an unsupervised generative statistical model that uses a BoW approach for representing text. Despite its prevalence, LDA's use of Bag-of-Words (BoW) often overlooks word sequences in text and limits performance [12]. In recent years, researchers have improved the performance of the LDA by incorporating neural network components to capture specific aspects of the input data (e.g., capture co-occurrences) [13]. There are three major categories of neural network topic modeling approaches:

- **Feed Forward Neural Network (FFNN)-based topic models**, such as DeepLDA [14] and Autoencoding Variational Inference for Topic Model (AVITM) [15] that use feed-forward neural networks to represent input texts as flat feature vectors for input into topic models.
- **Generative-based topic models**, such as Gaussian-Bidirectional Adversarial Training (G-BAT) [16] and Weibull Hybrid Autoencoding Inference (WHAI) [17], infer a deep probabilistic topic model with a generative encoder network (e.g., adversarial network) to capture the hierarchical document latent representations.
- **Recurrent Neural Networks (RNN)-based topic models**, such as TopicRNN [19] utilize Long Short-Term Memory (LSTM) and/or Bidirectional LSTM (BiLSTM) [18] to capture word positions and sequences of text for input into LDA or another topic model.

Among the different categories of neural network topic models, those that rely on recurrent architectures such as RNN, LSTM, and BiLSTM architectures can capture and learn from word positions, co-occurrences, and sequences. However, the performances of these models often suffer when operating on text with long contiguous sequences and dependencies (e.g., pastes). As a result, an alternative neural network architecture is required to capture and represent the lengthy sequential dependencies present in pastes. Therefore, we review BERT in the following sub-section.

C. *BERT*

BERT is a transformer-based Pre-trained Language Model (PTLM) that aims to represent unlabeled text [20]. BERT has consistently outperformed recurrent models in numerous unsupervised NLP tasks due to its ability to operate on significantly longer blocks of text than conventional LSTMs [20]-[21]. We present BERT's architecture in Fig. 2.

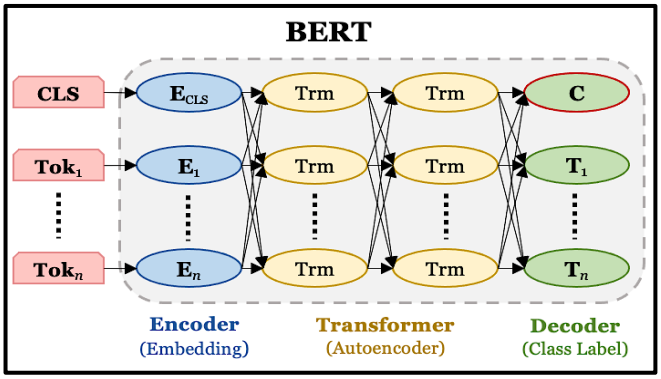

Fig. 2. BERT Architecture. Note: C = Class Label, CLS = Classification, E = Encoder, T = Token Output, Tok = Token, Trm = Transformer.

BERT includes three major components. First, an encoder reads, tokenizes, and creates a sentence embedding vector from the text input. Second, a transformer using an autoencoder learns contextual relationships between words in text (i.e., tokens) bidirectionally. Third, a decoder produces a class label for each sequence within the text. These class labels can be aggregated to produce a Bag-of-Labels (BoL) i.e., frequency of labels for all sequences in an input text. Despite the potential utility of BERT and its ability to produce a BoL that retains sequential information from long texts (and can therefore potentially address the limitations of prevailing

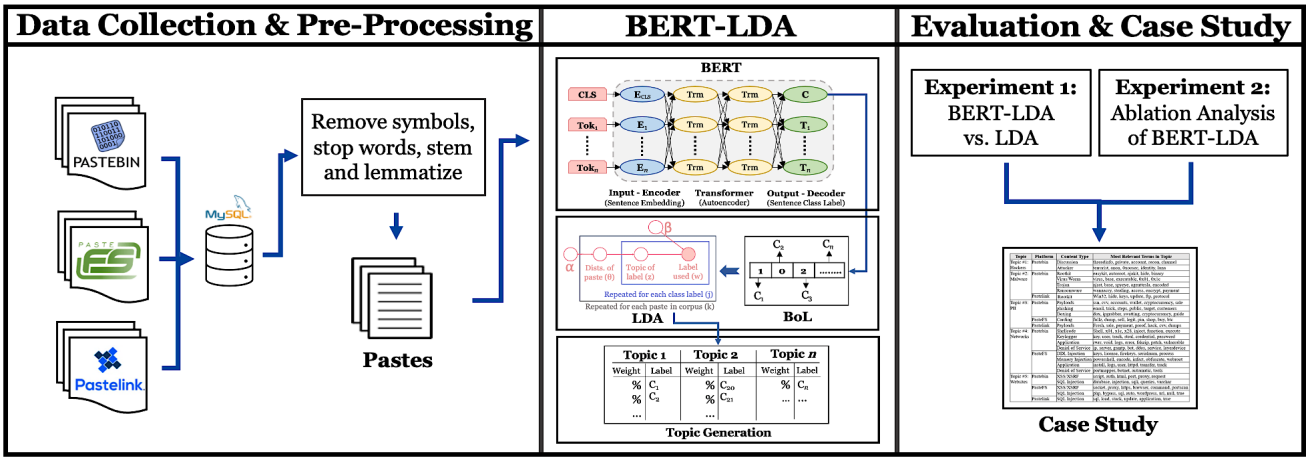

Fig. 3. Proposed Research Framework.

topic models), how to incorporate BERT into LDA to categorize content on paste sites requires additional study.

III. RESEARCH GAPS & QUESTIONS

We identified three research gaps from our literature review. First, prior studies limited their research to a single platform to perform targeted analysis (i.e., identification of breached records) instead of comprehensively analyzing all categories of pastes across multiple paste sites. Uncovering the full threat landscape of paste sites can provide targeted CTI about new threats and past breaches. Second, the prevailing topic model for categorizing hacker content in social media content, LDA, operates on a BoW model that can miss word order. Third, BERT uses transformers for unsupervised learning and can capture and represent word order and sequential dependencies as a BoL. However, how to integrate BERT into LDA requires further study. As such, we pose the following research questions for study:

- What categories of malicious content exist within prevailing paste sites?
- How can BERT be integrated into LDA to capture lengthy contiguous sequential dependencies from paste site content when categorizing pastes?

IV. RESEARCH DESIGN

Our proposed research framework includes three major components (Fig. 3): (1) Data Collection and Pre-Processing, (2) BERT-LDA, and (3) Evaluation & Case Study. We describe each major component in the following sub-sections.

A. *Data Collection & Pre-Processing*

We identified three prevailing paste sites for collection based on feedback from cybersecurity experts: Pastebin, PasteFS, and Pastelink. Custom web crawlers were developed to collect each paste and their associated metadata (e.g., title, author). A summary of our collection is presented in Table I.

TABLE I. SUMMARY OF DATA COLLECTION

Name	Start-End Date		Posts	Authors	Views
Pastebin	8/5/07	1/16/21	304,321	42,214	507,930,398
PasteFS	6/3/15	12/10/20	238,250	1,495	519,060
Pastelink	2/27/15	1/2/21	3,711,882	N/A	4,098,966
3 Sites	**8/5/07-1/16/21**		**4,254,453**	**43,709**	**512,548,424**

Our testbed includes over 4,254,453 pastes from 8/5/2007 to 1/16/2021 made by over 43,709 authors. Pastelink was the largest site in our collection, with 3,711,882 pastes. Overall, our collection exceeds the largest data collection in prior research by over 500,000 pastes [4]. Following best practice in hacker social media analytics, we pre-processed each paste by removing symbols, white space, special characters, and stop words and stemming, and lemmatizing each word [26].

B. *BERT-LDA*

Given the limitations of prevailing topic models as it pertains to categorizing long, contiguous text (e.g., pastes), we incorporate BERT into LDA. We present the proposed BERT-LDA in Fig. 4.

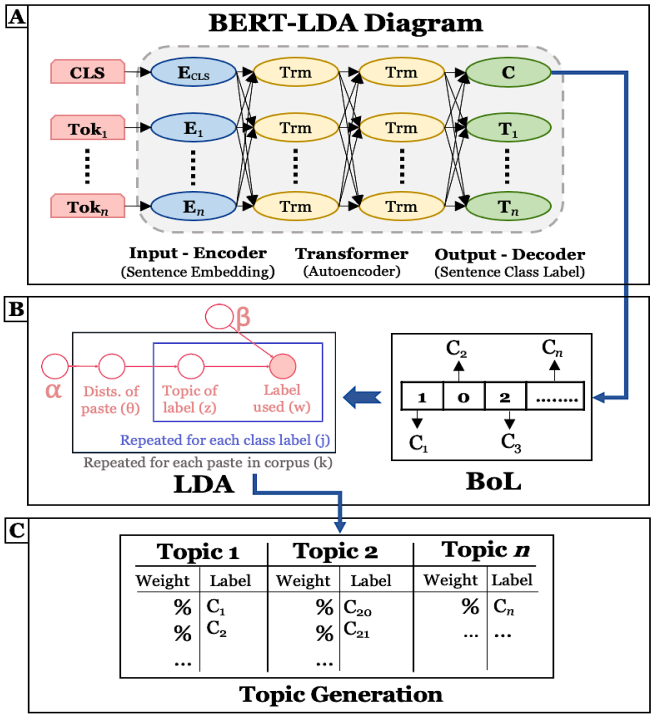

Fig. 4. Proposed BERT-LDA Architecture For Paste Categorization.

BERT-LDA consists of three major components:

A. BERT: BERT's encoder tokenizes each word in each sequence (sentence) for every inputted paste.

Tokens are used to create a sentence embedding to represent each sequence. The transformer's autoencoder reads sequences bidirectionally to create a sequence label.

B. BoL and LDA: the model replaces the traditional BoW in the conventional LDA model with the BoL (produced by BERT) to create a vocabulary of all unique labels. Replacing the BoW with a BoL helps to capture information at about each paste's semantics at the sequence-level, rather than at the word-level.

C. Topic Generation: LDA produces topics based on each paste's BoL.

BERT-LDA's core novelty resides in replacing LDA's BoW with a BoL generated from BERT. As such, we illustrate BERT-LDA's BoL functionality in further detail in Fig. 5. BERT-LDA's BoL operates as follows:

- **Step 1:** Each inputted paste is tokenized and sequenced. BERT inserts a "CLS" token at the beginning of first sentence and a "SEP" token at end of each sentence in a paste.
- **Step 2:** The sequenced tokens are passed to the encoder layer to create a sentence embedding vector.
- **Step 3:** The transformer then generates sequence level class labels from each vector (e.g., "ddos" and "attack").
- **Step 4:** All sequence labels are appended to a BoL to capture label frequencies at the sequence level.
- **Step 5:** The BoW in the conventional LDA is replaced with the BoL. Finally, LDA outputs topics based on the inputted BoL.

C. *Evaluation & Case Study*

We evaluated BERT-LDA with two sets of experiments. In Experiment 1, we compared BERT-LDA against the conventional LDA model, the prevailing approach for unsupervised topic modeling [3][11]. Experiment 2 is an ablation analysis for BERT-LDA that investigates how substituting the BERT in BERT-LDA with other prevalent PTLMs impacts performance. PTLMs used for ablation analysis include DistilBERT, Cross-lingual language model (XLM), and GPT-2. These models were selected due to their cutting-edge performance in various unsupervised text mining tasks [21]. DistilBERT's architecture is designed to have 40% less parameters and runs 60% faster than BERT [22]. XLM provides a robust pre-training method for cross-lingual understanding tasks [23]. Finally, GPT-2 includes a transformer decoder as the language model for text generation [24].

We executed both experiments on each of the collected paste sites (4,254,453 total pastes). The performances for each algorithm were measured using the perplexity metric. Perplexity measures how well a model predicts a given sample and is frequently used to compare topic models [25][26]. Lower perplexity scores indicate higher performances [25]. Benchmarking topic modeling algorithms in this fashion is a commonly accepted practice in hacker social media analytics literature [2] [26].

In addition to conducting experiments to evaluate BERT-LDA's performance, we executed a case study to demonstrate the BERT-LDA's potential practical utility for possible CTI applications. To execute the case study, we applied BERT-LDA to extract topics from each collected paste site. Following common practice in topic modeling literature, we manually assign names to each of the outputted topics [26].

V. RESULTS AND DISCUSSION

A. *Experiment 1: BERT-LDA vs LDA*

In Experiment 1, we compared the proposed BERT-LDA against the conventional LDA model. We present the results of Experiment 1 in Table II. The top performances are highlighted in bold-face.

TABLE II. EXPERIMENT 1 RESULTS: BERT-LDA VS LDA

Model	Topics	Pastebin	PasteFS	Pastelink
LDA	5	4,781.17	726.35	3,491.82
	10	4,016.33	546.82	3,491.82
	15	3,567.04	494.56	2,385.04
	20	2,475.64	446.88	1,121.44
BERT-LDA	**5**	**171.29**	**363.49**	**254.91**
	10	229.92	462.48	351.41
	15	276.91	521.12	403.65
	20	307.75	568.36	458.73

BERT-LDA achieved its best (i.e., lowest) perplexity scores across all platforms with 5 topics. In contrast, LDA

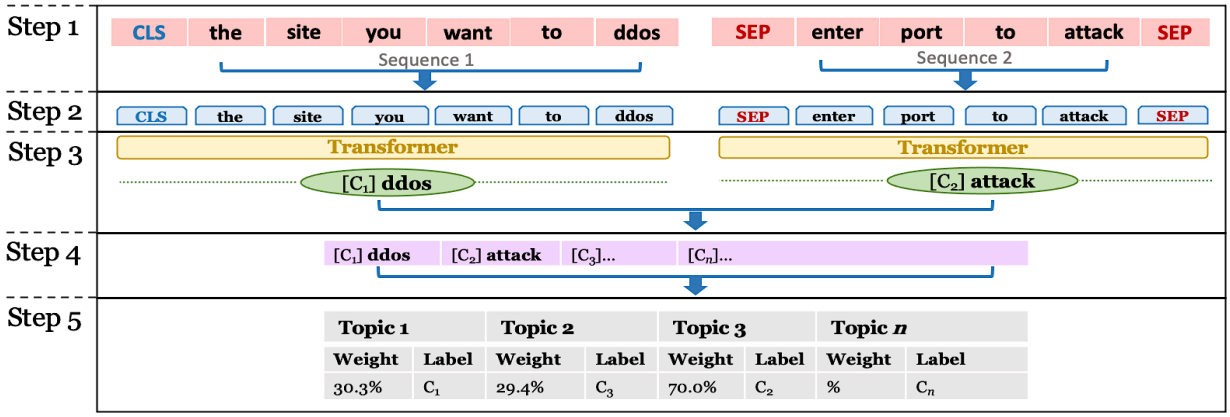

Fig. 5. Illustration of BERT-LDA's Process of Generating BOL for LDA.

achieved its best performances with 20 topics. With five topics, our BERT-LDA model outperformed the traditional LDA model for Pastebin (171.29 to 4,781.17), PasteFS (363.49 to 726.35), and Pastelink (254.91 to 3,491.82). BERT-LDA applies a BoL model to capture label frequencies and presumably nearly holds a dozen labels in its vocabulary. In contrast, LDA utilizes a BoW model. While this model captures word frequencies, it commonly has a vocabulary in the thousands. Therefore, it is plausible that the difference in label quantity between BERT-LDA's BoL and LDA's BoW resulted in BERT-LDA outperforming LDA.

B. Experiment 2: Ablation Analysis of BERT-LDA

In Experiment 2, we evaluated how BERT-LDA performed when BERT was replaced with alternative PTLMs perform. We present the results of Experiment 2 in Table III. The top performing algorithm is highlighted in bold-face.

TABLE III. EXPERIMENT 2 RESULTS: ABLATION ANALYSIS OF BERT-LDA

Model	Topics	Pastebin	PasteFS	Pastelink
XLM-LDA	5	270.80	402.60	322.07
	10	370.90	445.19	428.70
	15	439.69	452.28	496.46
	20	496.49	501.68	565.89
GPT2-LDA	5	173.84	422.58	227.87
	10	228.92	511.85	300.69
	15	265.09	570.37	346.09
	20	290.14	608.08	391.79
DistilBERT -LDA	5	256.49	**362.46**	274.79
	10	333.18	439.25	346.48
	15	386.73	491.31	406.58
	20	428.26	531.24	446.38
BERT-LDA (ours)	5	**171.29**	363.49	**254.91**
	10	229.92	462.48	351.41
	15	276.91	521.12	403.65
	20	307.75	568.36	458.73

Across all three platforms for each model, 5 topics resulted in lower perplexity scores than 10, 15 and 20 topics. GPT2-LDA outperformed XLM-LDA for Pastebin (173.84 to 270.80) and Pastelink (227.88 to 322.07), while only being 19.97 higher (422.58 to 402.61) on the PasteFS dataset. DistilBERT-LDA outperformed GPT-2-LDA and XLM-LDA for PasteFS (362.46), although underperformed against GPT2-LDA for Pastebin (256.49) and Pastelink (274.79). BERT-LDA achieved the best perplexity scores for Pastebin (171.30) and Pastelink (254.92), while only being 1.03 above the DistilBERT-LDA perplexity for PasteFS. PasteFS posts often exceed BERT's acceptable token range (i.e., 512 tokens) allowing DistilBERT's inference layer to be more powerful than BERT's. Overall, BERT-LDA's bidirectional transformer encoder layer shows clear improvements over XLM-LDA, GPT2-LDA, and DistilBERT-LDA.

C. Case Study Results

The proposed BERT-LDA model was applied to all pastes collected from Pastebin, PasteFS, and Pastelink platforms. We manually assigned names to five prevailing topics extracted by our model: (1) hackers, (2) malware, (3) networks, (4) websites, and (5) PII. We present case study results in Table IV. Results are sorted by topic, platform, content type, and relevant terms in the topic.

TABLE IV. TARGETED ANALYSIS AND FINDINGS FROM EXPERIMENTS. NOTE: DDLI = DYNAMIC-LINK LIBRARY INJECTION, DOS = DENIAL OF SERVICE, MI = MEMORY INJECTION, XSS/XSRF = CROSS SITE SCRIPTING/CROSS SITE REQUEST FORGERY, SQLI = SQL INJECTION.

Topic	Platform	Content Type	Relevant Terms in Topic
Topic 1: Hackers	Pastebin	Discussion	threadinfo, private, account, recon
		Attacker	terrorist, anon, 0xoosec, identity, luna
Topic 2: Malware	Pastebin	Rootkit	easykit, autoroot, ajakit, hide, binary
		Virus	virus, base, executable, 0x01, 0x1c
		Trojan	njrat, base, spyeye, agenttesla
		Ransomware	wannacry, stealing, access, encrypt
	Pastelink	Rootkit	win32, hide, keys, update, ftp
Topic 3: Network	Pastebin	Shellcode	shell, x01, x1c, x28, inject, function
		Keylogger	key, user, track, steal, credential
		Application	rwrr, void, logs, error, fakeip, patch
		DoS	ip, server, gsmtp, bot, ddos, service
	PasteFS	DDLi	keys, license, firekeys, serialnum
		MI	powershell, encode, infect, obfuscate
		Application	install, logs, user, httpd, transfer, track
		DoS	portmapper, botnet, automatic, tools
Topic 4: Website	Pastebin	XSS/XSRF	script, auth, html, port, proxy, request
		SQLi	database, injection, sqli, queries, char
	PasteFS	XSS/XSRF	socket, proxy, https, browser, portscan
		SQLi	php, bypass, sql, auto, wordpress, url,
	Pastelink	SQLi	sql, load, stack, update, app, true
Topic 5: PII	Pastebin	Stolen SSNs	ssn, cvv, accounts, wallet, sale
	PasteFS	Dumps	fullz, dump, sell, legit, pin, shop, buy
	Pastelink	Carding	fresh, sale, payment, proof, hack, cvv

We identified several noteworthy topics in the case study. First, Topic 1 included two distinct topics relating to hacker discussions and attackers. The keywords "threatinfo" and "channel" refer to hacker forum conversations and threads that might insinuate a cybercriminal's malicious intent. Topic 2 included four major malware types: (1) rootkits, (2) viruses, (3) trojans, and (4) ransomware. We discovered a significant ransomware termed "WannaCry Ransomware" on Pastebin, which is globally known for the major cyberattack on the U.S. National Health Service (NHS) in May of 2017. Topic 3 pertains to exploitable strategies for vulnerable networks and systems. Pastebin and PasteFS contain various Proof-of-Concept (PoC) exploits that could be used by hackers to carry out attacks. We discovered six unique exploits: shellcode, keylogger, application, DoS, DDLi, and memory injection. Similar to Topic 3, Topic 4 includes website application-related exploit content. Two major website application exploits were identified: cross site scripting/cross-site request forgery (XSS/XSRF) and SQL injections (SQLi). Additionally, instantaneous Proof-of-Concept (PoC) exploit code, lists of vulnerable websites, servers, and databases are uploaded on all three paste sites. Organizations can examine these exploit codes to identify attack, vulnerability, and malicious trends among adversarial actors for proactive CTI.

In addition to identifying topics pertaining to malicious actors and their exploits, we also identified three major categories of PII in Topic 5: (1) Stolen SSNs (2) dumps, and (3) carding. All three sites included breached and stolen data as well as other sensitive information, which may be associated with key terms such as "ssn," "account," "shop," and "cvv." These sensitive and personal information are often listed on paste sites for sale ensuing a data breach. Illicit users also share dark web links to Darknet Markets and anonymous financial intermediary platforms to commence sales.

Proactively identifying PII on paste sites can help alert organizations of potential breaches to their infrastructure.

VI. CONCLUSION AND FUTURE DIRECTIONS

Malicious cyber activities exact significant financial costs upon the global economy. Hackers are increasingly using social media platforms such as paste sites to share malicious content for cyber-attacks and breaches. In this study, we proposed a novel BERT-LDA to automatically extract major themes of malicious content from prevalent paste sites for potential CTI purposes. BERT-LDA operates by replacing the BoW used in the standard LDA with a BoL created by BERT that includes class labels at the sequence level. BERT-LDA outperforms the conventional LDA and BERT-LDA variants in terms of perplexity across three paste sites vs other PTLMs. We exhibited the potential practical utility of the proposed BERT-LDA in a case study that identified significant hacker community activity, malicious and destructive code, network and website vulnerabilities, as well as PII from prevailing paste sites. Organizations can use the findings of this study to proactively reduce potential damage on their infrastructure.

There are two promising directions for future study. First, future studies may concentrate on using key post metadata to perform threat trend analysis and detect emerging trends. Second, we can adapt deep transfer learning strategies implemented in prior CTI research to automatically classify the malicious code discovered by BERT-LDA [27]. Each direction can help provide more comprehensive CTI to combat emerging cyber-attacks.

VII. ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under grant numbers DGE-1921485 (SFS), OAC-1917117 (CICI), and CNS-1850362 (SaTC CRII).

VIII. REFERENCES

[1] V. Benjamin, J. S. Valacich, and H. Chen, "DICE-E: A framework for conducting Darknet identification, collection, evaluation with ethics," MIS Quarterly: *Management Information Systems*, vol. 43, no. 1, pp. 1–22, 2019.

[2] S. Samtani, M. Kantarcioglu, and H. Chen, "Trailblazing the Artificial Intelligence for Cybersecurity Discipline: A Multi-Disciplinary Research Roadmap," *ACM Transactions on Management Information Systems,* vol. 11 no. 4, Dec., pp. 1-19, 2020.

[3] S. Samtani, H. Zhu, and H. Chen, "Proactively Identifying Emerging Hacker Threats from the Dark Web: A Diachronic Graph Embedding Framework (D-GEF)," TOPS, *ACM Transactions on Privacy and Security,* vol. 23 no. 4, Aug., pp. 1-33, 2020.

[4] A. Riesco, E. Fidalgo, W. Al Nabki, F. Jáñez-Martino, and E. Alegre, "Classifying Pastebin Content Through the Generation of PasteCC Labeled Dataset," in Proceedings of HAIS, International Conference on Hybrid Artificial Intelligent Systems '19, 2019, pp. 456-467.

[5] Similarweb, "Pastebin Traffic, Ranking & Marketing Analytics," *Similarweb: Official Measure of the Digital World,* 2021. [Online]. Available: https://www.similarweb.com/website/pastebin.com/ [Accessed Sept. 18, 2020].

[6] M. Pakhari, N. Jamil, M, Rusli, and A. Rahim, "Implementation of Token Parsing Technique for Regex Based Classification of Unstructured Data for Cyber Threat Analysis," in Proceedings of ICMU, International Conference on Mobile Computing and Ubiquitous Networking '20, 2020, pp. 395-398.

[7] M. Brengel and C. Rossow, "Identifying Key Leakage of Bitcoin Users," in Proceedings of the RAID, Research in Attacks, Intrutions, and Defenses," '18, 2018, pp. 623-643.

[8] H. Imai and A. Kanaoka, "Time Series Analysis of Copy-and-Paste Impac on Android Application Security," in Proceedings of the AsiaJCIS, Conference on Information Security, '18, 2018, pp. 15-22.

[9] M. Squire and A. Smith, "The Diffusion of Pastebin Tools to Enhance Communication in FLOSS Mailing Lists," in Proceedings of the OSS, Open Source Systems: Adoption and Impact*,* '15, 2015, pp. 45-57.

[10] U. Eseryel, K. Wie, and K. Crowston, "Decision-making Processes in Community-based Free/Libre Open Source Software Development Teams with Internal Governance: An Extension to Decision-making Theory," in Proceedings of the CAIS, Communications of the Association for Information Systems*,* '20, 2020, pp. 484-510.

[11] W. Li, J. Yin, and H. Chen, "Targeting Key Data Breach Services in Underground Supply Chain," in Proceedings of the IEEE, International Conference on Intelligence and Security Informatics (ISI), '16, 2016, pp. 322-324.

[12] N. Barbieri, G. Manco, E. Ritacco, M. Carnuccio, and A. Bevacqua, "Probabilistic Topic Models for Sequence Data," in Proceedings of Machine Learning, '13, 2013, pp. 5-29.

[13] Y. Chai and W. Li, "Towards Deep Learning Interpretability: A Topic Modeling Approach," in Proceedings of the ICIS, International Conference on Information Systems*,* '19, 2019, pp. 1-10.

[14] M. Bhat, M. Kundroo, T. Tarray, and B. Agarwal, "Deep LDA : A new way to topic model," Journal of Information and Optimization Sciences, *Journal of Information and Optimization Sciences,* vol. 41, no. 3, pp. 823-834, 2020.

[15] A. Srivastava and C. Sutton, "Autoencoding Variational Inference For Topic Models," in Proceedings of ICLR, International Conference on Learning Representations, '17, 2017, pp. 1-12.

[16] R. Wang, X. Hu, D. Zhou, Y. He, and Y. Xi, "Neural Topic Modeling with Bidirectional Adversarial Training," in Proceedings of ACL, Association for Computational Linguisitics, '20, 2020, pp. 340-350.

[17] H. Zhang, B. Chen, and D. Guo, "WHAI: Weibull Hybrid Autoencoding Inference For Deep Topic Modeling," in Proceedings of ICLR, International Conference on Learning Representations, '18, 2018, pp. 1-15.

[18] A. Dieng, C. Wang, J. Gao, and J. Paisley, "TopicRNN: A Recurrent Neural Network with Long-Rance Semantic Dependency," in Proceedings of ICLR, International Conference on Learning Representations, '17, 2017, pp. 1-13.

[19] P. Zhou, Z. Qi, S. Zheng, J. Xu, H. Bao, and B. Xu, "Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling," *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers,* vol. 2, no. 1, pp. 3485–3495, 2016.

[20] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of the NAACL-HLT, Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, '19, 2019, pp. 4171–4186.

[21] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-Trained Models for Natural Language Processing: A Survey," *Science China Technological Sciences ArXiv,* vol. 63, no. 10, pp. 1872-1897, 2020.

[22] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter," in Proceedings of NIPS, Neural Information Processing Systems, '19, 2019, pp. 1-5.

[23] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzman, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised Cross-Lingual Representation Learning at Scale," in Proceedings of ACL, Association for Computational Linguisitics*,* '20, 2020, pp.8440-8451.

[24] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language understanding by Generative Pre-Ptraining," in Proceedings of Computer Science, '18, 2018, pp. 12.

[25] D. Blei, A. Ng, M. Jordan, "Latent Dirichlet Alloation," JMLR, *Journal of Machine Learning Research,* vol. 3, pp. 993-1022, 2003.

[26] W. Li, H. Chen, and J. Nunamaker Jr., "Identifying and Profiling Key Sellers in Cyber Carding Community: AZSecure Text Mining System," JMIS, *Journal of Management Information Systems,* vol. 33, pp. 1059-1086, 2016.

[27] B. M. Ampel, S. Samtani, H. Zhu, S. Ullman, and H. Chen, "Labeling Hacker Exploits for Proactive Cyber Threat Intelligence: A Deep Transfer Learning Approach," IEEE, International Conference on Intelligence and Security Informatics (ISI), no. November, 2020.