Distilling Contextual Embeddings Into A Static Word Embedding For Improving Hacker Forum Analytics

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

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

***Abstract*—Hacker forums provide malicious actors with a large database of tutorials, goods, and assets to leverage for cyber-attacks. Careful research of these forums can provide tremendous benefit to the cybersecurity community through trend identification and exploit categorization. This study aims to provide a novel static word embedding, Hack2Vec, to improve performance on hacker forum classification tasks. Our proposed Hack2Vec model distills contextual representations from the seminal pre-trained language model BERT to a continuous bag-of-words model to create a highly targeted hacker forum static word embedding. The results of our experimental design indicate that Hack2Vec improves performance over prominent embeddings in accuracy, precision, recall, and F1-score for a benchmark hacker forum classification task.**

Keywords— Hacker forums, static word embeddings, contextual embeddings, knowledge distillation, text classification

I. Introduction

Recently, the amount of cyber-attacks on critical organizational infrastructure has dramatically risen [1]. To protect against such attacks, organizations are increasingly looking for proactive measures to detect potential cyber-attacks before they can occur. Hacker forums are a prevailing source of tutorials and exploits that can and have been used in cyber-attacks [2]. These forums contain millions of posts from tens of thousands of unique authors that can provide valuable intelligence to organizations.

Hacker forums primarily operate through textual conversations between users. The post content found in hacker forums is often messy and grammatically incorrect, making automated textual analysis a non-trivial task. Carefully building a feature representation to capture the unique writing styles of hackers in various hacker forums can assist in downstream tasks (e.g., clustering, classification) for hacker forum analytics.

In this work, we developed a novel static word embedding titled Hack2Vec for the hacker forum analytics community. Hack2Vec draws upon state-of-the-art techniques for vector representations of text to create a novel embedding for the hacker forum analytics community.

The remainder of this paper is as follows. First, we review prior literature on hacker forum analytics and static and contextual word embedding techniques. Second, we present the research questions of this work. Third, we outline our research design. Fourth, we present the results of our experimental design. Finally, we conclude the research and offer several future directions.

II. Literature Review

We reviewed two areas of literature to form the foundation of this research: (1) hacker forum analytics, and (2) static word embeddings. First, we study hacker forum analytics to discover the prevalent techniques and goals of prior literature. Second, we review static word embeddings to understand how to create a novel feature representation as input to a model.

A. Hacker Forum Analytics

Hackers often congregate to specialized forums to discuss and share goods (e.g., SSNs, credit card numbers) and assets (e.g., exploit binaries, malware, source code) [3]. Recent extant literature has frequently focused on using the post content and source code found on hacker forums to classify malicious assets [3]-[7]. This literature often implements popular deep learning models, such as long short-term memory (LSTM) [7], Bidirectional LSTM (BiLSTM) [4], and diachronic graph embeddings (DGE) [5] to perform their analytics. Of these studies, one implemented the popular pre-trained embedding Global Vectors for Word Representation (GloVe) [8] model to process input to their BiLSTM model [4]. Another used a graph-of-words approach to create custom embeddings for hacker forum analytics [5].

From the literature, we note two research gaps. First, three of the five identified papers do not use a targeted embedding strategy for their input data. This is despite evidence that word embedding models can often improve classification performance for natural language processing (NLP) tasks (e.g., hacker forum text) [9]. Second, while the DGE model [5] is powerful for threat classification, it is only trained on 4,293 highly targeted exploit source code snippets. This approach can potentially miss general trends in hacker communities that are not exploit source code related (e.g., tutorials, binaries, etc.). To develop novel embeddings for the hacker forum analytics community, we require a mechanism that can effectively handle millions of general traditional hacker forum records.

B. Static and Contextual Word Embeddings

A static word embedding is a mapping of individual words to a dense n-dimensional vector. This embedding style was widely popularized in 2013 with the release of Word2Vec (W2V) [10]. W2V captures relational meaning between words using either a continuous bag-of-words (CBOW) model

(prediction of a target word given surrounding context words) or a Skipgram model (prediction of surrounding context words given target word). Subsequently, researchers considered that W2V did not properly capture global statistic information. The following year (2014), NLP researchers released GloVe, which implements a global co-occurrence matrix to find the relationships between words [8]. Following this, fastText replaced single vectors for each atomic token with summed n-gram vectors [11].

Recently, static word embeddings have been largely replaced with advances in contextual embeddings. The most popular of these methods is known as BERT [12]. BERT, and related contextual embedding models, place both contextual information and word representations into the same embedding. Contextual embedding models have greatly improved performance in NLP tasks that require sentence semantics understanding [13]. However, these models are computationally expensive and often have fixed length requirements (generally 512 tokens), necessitating an approach that can addresses these two issues. An increasingly popular method is to distill a feature representation from these contextual embedding models to a static embedding through decontextualized (removing context from each word) or aggregated (taking the minimum, maximum, average, or last vector) subword pooling [14].

III. Research Gaps and Questions

From the literature review, we identify two gaps. First, the current approaches to hacker forum embeddings are either out of date (e.g., GloVe) or very specific to the target task (e.g., DGE). Second, while contextual embedding models provide tremendous benefits, their fixed length and computational requirements make them inefficient for many tasks. To address these gaps, we pose the following research question for this study:

- How can static word embeddings be built from seminal contextual embedding models to create a novel and scalable hacker forum embedding?

IV. Research Design

This study aims to create a pre-trained hacker forum embedding, titled Hack2Vec, that distills knowledge from the contextual embedding model BERT to a CBOW-based static word embedding model. Our research design is comprised of three major components: data testbed and pre-processing, Hack2Vec framework, and evaluations (Figure 1).

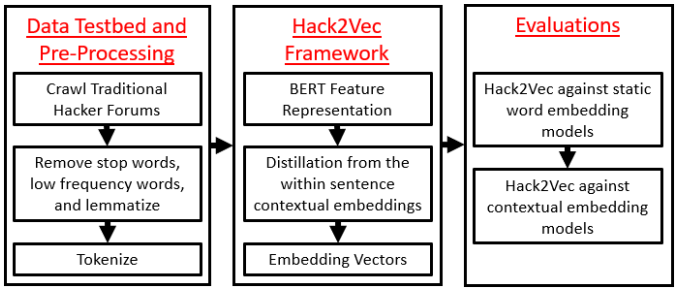

Fig. 1. Proposed Research Design

A. Data Testbed and Pre-Processing

Our data collection is comprised of seven prominent English hacker forums. These forums were selected based on input from domain experts and a depth-first search crawler. The crawler followed link stacks and kept records of each visited link to ensure an efficient and atomic data collection. A summary of our overall data collection is shown in Table 1.

TABLE I. Summary of Research Testbeds

Name	Start Date	End Date	Posts
0x00sec	4/13/2017	7/1/2021	9,161
Altenens	3/22/2010	7/1/2021	1,261,435
AntiOnline	4/10/2002	7/1/2021	291,914
Ciphers	5/1/2015	7/1/2021	51,612
go4expert	12/25/2004	7/1/2021	62,103
KernelMode	4/12/2018	7/1/2021	29,755
WildersSecurity	2/8/2002	7/1/2021	2,571,053
7 Sources	**2/8/2002 - 7/1/2021**		**4,277,033**

In total, we collected 4,277,033 posts from the seven hacker forums. The date range of our collection is from 2/8/2002 to 7/1/2021, providing a complete and historical testbed of hacker forum activity. From this data testbed, we removed all posts less than 100 characters in length (as these are generally uninformative), leaving 3,429,192 hacker forum posts. These remaining posts were stripped of unnecessary symbols, lower-cased, lemmatized and fed through the BERT tokenizer [12]. The BERT tokenizer creates word-level tokens and also deconstructs words into several potential subwords.

B. Hack2Vec Framework

We first pass our pre-processed inputs through a BERT uncased model to create a contextual representation of our data. Then, following prevailing literature [14]-[15], we distill the contextual representation into a CBOW static word embedding model through aggregated subword pooling. This design aggregates the contextual representation of each subword for each word in an input [14]. Specifically, the model takes a sample n sentences from the set of hacker forum inputs that contains a word w. Then, the contextual representation produced by the BERT model of w in each sentence is pooled together by taking the average of the vectors to associate w with one static vector. This process repeats for each word token in the corpus until all words have a single vector.

C. Evaluations

To evaluate each embedding in the context of hacker forum analytics, we used a gold-standard hacker forum dataset from the literature [4]. Each embedding is used as the first layer into a standard BiLSTM model to classify hacker forum posts into one of eight exploit categories (web application, denial of service, remote, local, SQL injection, cross-site scripting, file inclusion, and overflow).

We evaluated our proposed Hack2Vec embeddings with two sets of experiments. The first experiment compared Hack2Vec against static word embedding models prevailing in literature (W2V, GloVe, and fastText). We choose the 300-dimensional versions of these models for comparison. The second experiment compared Hack2Vec against prevailing

contextual embedding models (BERT) without distillation. We also train a BiLSTM model with no embedding model (Baseline in Table 2) to serve as a baseline. Both experiments use accuracy, F1, precision, and recall as metrics.

V. Results and Discussion

The results of our two experiments are summarized in Table 2. The top-performing embedding model for each metric appears in bold-face.

TABLE II. Experiment Results

Exper.	Embedding Model	Results			
		Accuracy	Precision	Recall	F1-score
1	Baseline	60.18%	64.78%	58.89%	61.11%
	W2V	61.48%	64.91%	61.01%	62.43%
	GloVe	63.05%	67.56%	59.71%	63.21%
	fastText	62.84%	65.14%	62.71%	64.02%
2	BERT	65.58%	65.47%	64.73%	65.17%
1, 2	Hack2Vec	**66.82%**	**67.19%**	**66.01%**	**66.52%**

*Note: Exper. = Experiment.

In Experiment 1, fastText is the best performing static word embedding model for hacker forum analytics in F1-score (64.02%), improving upon no embeddings (61.11% F1-score), W2V (62.43%), and GloVe (63.21%). However, fastText performs worse in all four tracked metrics when compared to the BERT embeddings in Experiment 2. Our proposed Hack2Vec model outperformed all models in both experiments in accuracy (66.82%), precision, (67.19%), recall (66.01%), and F1-score (66.52%). These results suggest that creating our static word embedding model outperforms the pre-trained static word embedding models (W2V, GloVe, fastText). Hack2Vec outperforming BERT also shows that the distillation of contextual representations from BERT to a static word embedding improved model performance, most likely due to the highly targeted nature of the hacker forum classification task.

In addition to performance, we also tracked the inference time for each embedding model. The static word embedding models had an average inference time of 0.479s, with fastText performing the fastest (0.412). BERT had an average inference time of 5.163s, over 10x slower than the static word embedding methods. Finally, our proposed Hack2Vec model had an average inference time of 0.596s, which is significantly faster than BERT, while providing better performance metrics.

VI. Conclusion and Future Directions

In this study, we aimed to develop a novel static word embedding that would improve performance for hacker forum analytics tasks. Our results indicate that distilling contextual representations from the seminal BERT model to a CBOW model offers a benefit to hacker forum classification tasks. To the best of our knowledge, this work uses the largest collection of hacker forum data (3,429,192 posts) and is the only research to incorporate contextual representations of hacker forum text.

We have considered several possible future directions for this work. We would like to test our Hack2Vec model on unsupervised tasks (e.g., social network analytics for prominent hackers) to further prove its value for hacker forum analytics. We would also like to test more contextual embedding models (e.g., RoBERTa) to see if there are significant performance differences from our proposed approach.

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).

References

[1] L. Jeffrey and V. Ramachandran, “Why ransomware attacks are on the rise — and what can be done to stop them” Public Broadcast Service, 2021. [Online]. https://www.pbs.org/newshour/nation/why-ransomware-attacks-are-on-the-rise-and-what-can-be-done-to-stop-them. [Accessed: 27-Jul-2021]

[2] S. Samtani, M. Abate, V. Benjamin, and W. Li. "Cybersecurity as an industry: A cyber threat intelligence perspective." The Palgrave Handbook of International Cybercrime and Cyberdeviance, pp. 135-154. 2020

[3] V. Benjamin, W. Li, T. Holt, and H. Chen, “Exploring threats and vulnerabilities in hacker web: Forums, IRC and carding shops,” 2015 IEEE International Conference on Intelligence and Security Informatics (ISI), pp. 85–90. 2015

[4] B. Ampel, S. Samtani, H. Zhu, S. Ullman, and H. Chen. "Labeling hacker exploits for proactive cyber threat intelligence: A deep transfer learning approach," 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), pp. 1-6., 2020

[5] S. Samtani, H. Zhu, and H. Chen. “Proactively Identifying Emerging Hacker Threats from the Dark Web.” ACM Transactions on Privacy and Security, 23(4), pp. 1–33. 2020

[6] 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

[7] I. Deliu, C. Leichter, and K. Franke, “Collecting Cyber Threat Intelligence from Hacker Forums via a Two-Stage, Hybrid Process using Support Vector Machines and Latent Dirichlet Allocation,” 2018 IEEE International Conference on Big Data, Big Data 2018, pp. 5008–5013, 2019

[8] M. Pennington, Jeffrey; Richard Socher; Christopher, “Glove: Global vectors for word representation,” in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532–1543, 2014

[9] T. Schnabel, I. Labutov, D. Mimno, and T. Joachims. "Evaluation methods for unsupervised word embeddings." In Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 298-307. 2015

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. "Distributed representations of words and phrases and their compositionality." In Advances in neural information processing systems, pp. 3111-3119. 2013

[11] A. Joulin, E. Grave, P. Bojanowski, T. Mikolov. "Bag of tricks for efficient text classification." In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pp. 427-431. 2017

[12] J. Devlin, M. Chang, K. Lee, and K. Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805. 2018

[13] S. Wang, W. Zhou, C. Jiang. "A survey of word embeddings based on deep learning." Computing 102, no. 3, pp. 717-740. 2020

[14] R. Bommasani, K. Davis, and C. Cardie. "Interpreting pretrained contextualized representations via reductions to static embeddings." In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 4758-4781. 2020

[15] P. Gupta, M. Jaggi. "Obtaining Better Static Word Embeddings Using Contextual Embedding Models." arXiv preprint arXiv:2106.04302. 2020