Corporate cyber-security event detection platform

Zheng Nie
Jingjing Feng
Steve Pomerville
Azadeh Nematzadeh
S&P Global
firstname.lastname@spglobal.com

Abstract

Among corporate stakeholders, meeting ESG standards(Environmental, Social and Governance) has grown significantly in recent years and is forecasted to continue growing. Cyber-security risks are a key ESG concern since cyber attacks may disrupt companies' operations, brand image, supply-chain, and affect stockholders' interests. We built a platform to detect corporate cyber-security events, in real-time, leveraging news sources and advances in language model pre-training. Specifically, we fine-tuned a deep learning transformer model on about 7,500 historical, supervised cyber-security news snippets using 3-folds cross-validation from companies between 2005 to 2019. The learned models are applied to detect cyber-security events about the companies and the best preforming model has achieved a F-Score of 0.88 in the cyber-security event prediction task and a F-score of 0.74 in predicting if a company has been a victim of a cyber-security event. The platform notifies its users in real-time through email alerts with relevant corporate cyber-security events and associated news articles.

Introduction

Mitigating cyber-security threats has conventionally been a high priority of corporations. Companies vigorously manage their defenses in anticipation of growing security threats by utilizing more sophisticated security products, implementing safety protocols, and educating their employees.

Until recently, cyber-security maintenance was mainly focused on keeping an operation uninterrupted from attacks. However, cyber-security risks are posed in other areas of a business' operation and companies may be open to cyber-security attacks through Merger and Acquisition(M&A) or supply-chain vulnerabilities. For instance, Marriott International's acquisition of Starwood Hotels and Resorts Worldwide in 2016 resulted in a serious data breach in 2018 when the Starwood guest reservation database was tampered with. Also, cyber-security threats have shown to have a broader impact on businesses including damaging the creditworthiness of companies or damaging investor and shareholder interests. For the first time in 2017 Equifax ratings outlook was downgraded due to cyber-security concerns.

Fundamental factors regarding cyber-security that investors and government officials evaluate include: how companies understand and mitigate cyber-security risks and how often and to what extent are they the victims of cyber-security threats. Investors seek to assess companies' cyber-security risks along with other risks. Cyber-security assessments can be achieved through an industry standard framework that investors can rely on throughout the investment cycle. However, while companies are obliged to report their financials regularly throughout the year, they do not formally disclose their security management activities, breakdowns, or vulnerabilities.

To assess companies cyber-security risks, companies and investors follow a cyber-security due diligence framework [1] and use tools such as interviews, surveys, and external audits such as SOC1. The current cyber-security assessment methods are repetitive and operationally inefficient. Also, there is not procedural validation or regulation in place to assure the trustworthiness of interviews or survey content.

The demand for cyber-security assessments of companies have risen, thus we designed a solution and built a platform to capture cyber-security risks associated with companies. Our goal is to detect cyber-security risks of companies, in real-time, and then to notify the users– for instance, the users could be corporate cyber-security risk analysts. The detected corporate cyber-security risks are threats that jeopardize confidentiality, integrity and availability of data and services through various mediums of cyber attacks. We leveraged advanced AI models and fine-tuned them for company cyber-security event detection tasks and introduced a platform that can monitor and detect cyber-security events for companies in real-time. We use news articles as the main data source of our platform. News sources report companies' security events in a timely manner and can fill the gaps where a lack of officially reported security documents exists. We used state-of-the-art pre-trained models, Wikipedia2Vec and BERT and fine-tuned BERT-Cased and BERT-Uncased deep learning models(BERT-C and BERT-U) for a news snippet classification task using labeled news snippets. We used a convolutional neural network model for sentence classification (Text-CNN) as a baseline (Kim 2014). Our la-

[1] http://www3.weforum.org/docs/WEF_Incentivizing_responsible_and_secure_innovation.pdf

belled data indicates first, if a news snippet is about a security event (prediction task I), and second, if a news snippet indicates that the company has been a victim of a security event(prediction task II). The second prediction task is more complex than first one as it requires learning the association between a company and a cyber-security event. To retrieve relevant news snippets, we built a corporate cyber-security expansion and a filtering model.

We tested the performance of our news snippet classification models on three test sets using a 3-fold cross validation technique. We compared the performance of our models for each prediction task with two inputs. Using the first input (I), we fine-tuned our models using cyber-security news snippets and the associated company. Using the second input(II), we fine-tuned our models only with news snippets. Thus, for each prediction task we compared four cases which were BERT-C I, BERT-C II, BERT-UC I, and BERT-UC II. BERT-Uncased using input I (BERT-UC I) was the best preforming model and it achieved a F-Score of 0.88 in predicting cyber-security events and a F-score of 0.74 in predicting if a company was a victim of a cyber-security event.

The platform is currently being used and tested by a small number of analysts and we are collecting online labels data to evaluate the performance of our models in an online learning setting.

Literature Review

Machine learning and AI has been applied to solving cyber-security problems and has proven to be beneficial in mitigating cyber-security threats (Apruzzese et al. 2018; Ferrag et al. 2020). For instance, LSTM-RNN classifiers have been used to detect different types of network attacks (Kumar, Goomer, and Singh 2018). An online deep learning architecture has been applied in insider threat detection (Tuor et al. 2017). Despite the large volume of research in building intrusion detection systems, there is a gap in building tools that can assess corporations' cyber-security risks (Alghamdi and Rastogi 2020). We treated corporate cyber-security risk assessment with an event detection solution.

Event detection methods, either supervised or unsupervised, have been applied to many problems including propaganda detection for online news (Barrón-Cedeno et al. 2019), fake news detection (Rashkin et al. 2017), and ”big event” detection on Twitter (Weng and Lee 2011). Recent works on event detection have leveraged deep-learning methods. Twitter traffic event detection methods have leveraged word embedding and supervised deep-learning algorithms including convolutional neural network (CNN) and recurrent neural network (RNN) for detecting traffic events (Dabiri and Heaslip 2019). An anomalous event detection in video scenes has been developed using an unsupervised deep learning framework (Xu et al. 2015) Abnormal event detection for video sequences have been built using deep learning models (Fang et al. 2016). Enhanced CNN models have been applied to detect crisis events from from social media data (Burel et al. 2017).

A lack of label data is one of the biggest challenges in applying deep learning methods for cyber-security tasks. CASIE is one of the first systems that provides a graph of cyber-security data by extracting information about cyber-security events from text (Satyapanich, Ferraro, and Finin 2020). A data set for cyber-security event detection was recently introduced that captures 30 cyber-security event types and is suited for deep learning evaluations (Trong et al. 2020). Deep learning has recently been applied in cyber-security detection. A CNN model was introduced to classify cyber threat indicators using Twitter data (Behzadan et al. 2018).

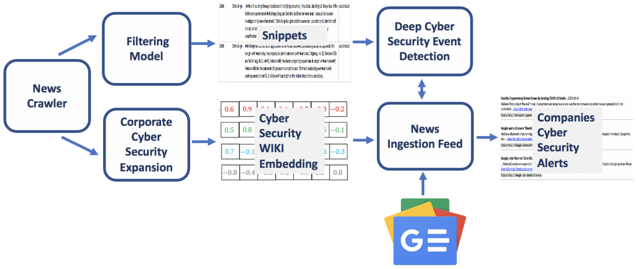

News Crawler
Filtering Model
Corporate Cyber Security Expansion
Snippets
Cyber Security WIKI Embedding
Deep Cyber Security Event Detection
News Ingestion Feed
Companies Cyber Security Alerts

Figure 1: The main components of the corporate cyber-security event detection platform

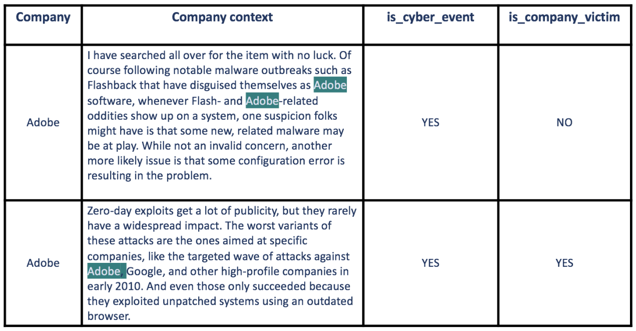
Methodology

We designed and built a platform that detects corporate cyber-security events in real-time. The platform allows users to select companies of interest. It then collects and analyzes cyber-security news relevant to those companies daily. After the deep event detection model is applied, users are notified through email if cyber-security events of their selected companies are found. Figure 1 shows the architecture of our corporate cyber-security risk model.

- **News crawler-** it collects historical news articles that are about cyber-security events.
- **Filtering model-** we built a model that filters out irrelevant cyber-security news snippets about a given company from a pool of collected historical news articles. Each company was represented by textual news snippets.
- **Deep cyber-security event detection-** we fine-tuned BERT to build a supervised deep model that classifies news snippets and detects cyber-security events of companies.
- **Corporate cyber-security expansion-** we built a model that leverages Wikipedia embedding to expand a given list of corporate cyber-security risks that jeopardize confidentially, integrity and availability.
- **News ingestion-** for a given set of selected companies, the news ingestion model uses Google Alerts and returns daily cyber-security news articles about the companies.
- **Company cyber-security alert-** it generates daily alerts about cyber-security events of selected companies.

News crawler

News articles are one of the main data sources in our analysis. The news crawler module uses Google Search to pull



Company	Company context	is_cyber_event	is_company_victim
Adobe	I have searched all over for the item with no luck. Of course following notable malware outbreaks such as Flashback that have disguised themselves as Adobe software, whenever Flash- and Adobe-related oddities show up on a system, one suspicion folks might have is that some new, related malware may be at play. While not an invalid concern, another more likely issue is that some configuration error is resulting in the problem.	YES	NO
Adobe	Zero-day exploits get a lot of publicity, but they rarely have a widespread impact. The worst variants of these attacks are the ones aimed at specific companies, like the targeted wave of attacks against Adobe Google, and other high-profile companies in early 2010. And even those only succeeded because they exploited unpatched systems using an outdated browser.	YES	YES

Figure 2: Examples of news snippets and two prediction labels

historical news articles about cyber risks. Google Search allows users to set the retrieval frequency such as daily, weekly, biweekly, monthly and yearly for search queries and the search results include news headlines, summaries, date published and news links.

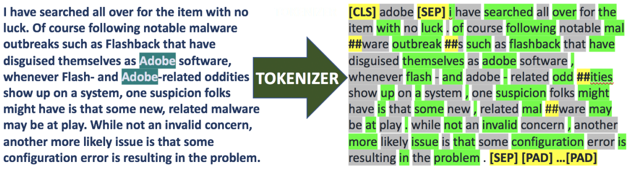
The module ran a search query for each of the cyber risk terms listed in the column 'Attacks and risks' of Table 1 with Google Search using Selenium for a yearly frequency. Selenium [2] is an automation testing framework for web applications which can also control a browser to navigate websites just like a human.

Filtering model

For a given company, we first retrieved historical cyber-security related news articles that have mentioned the company's name. We then extracted news snippets for the company in each of these articles. A snippet about a company includes three sentences including a sentence in which the company's name has appeared, the sentence before and the sentence after, see Figure 2 for an example.

Deep cyber-security event detection

Our deep cyber-security event detection model determines if a company has been a victim of a cyber-security attack. Our event detection model uses the output of our filtering model and classifies each snippet if it includes a security event or not. However, there are many cases in which a company is mentioned in the context of a security event but it is not necessarily the victim of the security event. For instance, cyber-criminals have used vulnerabilities in Adobe Flash or PDF to send malware to users. In these cases, Adobe is not the victim. To determine if a retrieved news snippet is about a corporate security breach we used two prediction labels. The first label predicts if the snippet contains a security event and the second label predicts if the security event is associated to the selected company. The two labels are *is_cyber_event* and *is_company_victim*. Each of the two labels have two values: 1 or -1. A positive value (1) for both labels indicates that a news snippet is about a security event and the selected company is a victim of the specified security event.



I have searched all over for the item with no luck. Of course following notable malware outbreaks such as Flashback that have disguised themselves as Adobe software, whenever Flash- and Adobe-related oddities show up on a system, one suspicion folks might have is that some new, related malware may be at play. While not an invalid concern, another more likely issue is that some configuration error is resulting in the problem.

[CLS] adobe [SEP] i have searched all over for the item with no luck . of course following notable mal ##ware outbreak ##s such as flashback that have disguised themselves as adobe software , whenever flash - and adobe - related odd ##ities show up on a system , one suspicion folks might have is that some new , related mal ##ware may be at play . while not an invalid concern , another more likely issue is that some configuration error is resulting in the problem . [SEP] [PAD] …[PAD]

Figure 3: The BERT tokenizer receives news snippets and generates the wordpieces.

Both of these prediction tasks are supervised binary snippets classification problems that we solved by fine-tuning BERT model (Devlin et al. 2018). BERT is a language model for transfer learning that uses a deep bidirectional transformer architecture (Vaswani et al. 2017) and is pre-trained on 16GB of BookCorpus and Wikipedia data to produce context-dependent embeddings. To apply BERT to snippet classification tasks, we included a fully connected layer on top of the BERT self-attention layers, which classifies the sentence embedding provided by BERT into two classes of interest for each of our prediction tasks.

Tokenization To prepare the input for BERT, we first tokenized news snippets using a trained BERT wordpiece model (Wu et al. 2016). BERT wordpiece embeddings maximize the language-model likelihood of the training data and generate a deterministic segmentation of any sequence of characters in which an input word will be broken into wordpieces. To be able to recover the word from the wordpiece, special boundary characters are added. The first token of every sequence input is set to [CLS] and the last is [SEP]. Figure 3 shows the input and output of the tokenizer in which ## marker is used to indicate split wordpieces.

The trained BERT wordpiece model has $30,000$ token vocabulary. BERT input character size is limited to 512 tokens and we set the maximum length to 256. Any inputs shorter than the maximum length of 256 will be padded at the end with a special token [PAD] while any longer input will be truncated to the maximum length.

Fine-tuning Given the limited number of labelled data in our study, training a new model from scratch was not feasible. Thus we turned to BERT pre-trained language model, which have made major breakthroughs in several natural language understanding tasks in recent years (Dai and Le 2015; Peters et al. 2018; Howard and Ruder 2018; Devlin et al. 2018). BERT pre-trained language model is trained on large corpus and can learn universal language representations and it can be fine-tuned on downstream NLP tasks. Fine-tuning requires a smaller amount of labelled data than training a new model from scratch (Radford et al. 2018).

Applying BERT in snippet classification tasks is done by adding a classification layer on top of the Transformer output. The final hidden state of the special token [CLS] is taken as the aggregate sequence representation and is input into a softmax classifier to predict the probability of *is_cyber_event* and *is_company_victim* using the fine-tuning objective function in the Equation 1.

[2] https://www.selenium.dev/

$$\theta^f max \sum_c I(y = c) log\, p(y = c), \qquad (1)$$

The fine-tuning objective function uses a cross-entropy loss where I is an indicator function, y is the class label, θ^f is the fine-tuned Transformer parameters, and the probability of the weight matrix W^Y is defined as follows:

$$p(y = c) = softmax(W^{Y^T}(z_0 \oplus)x) \qquad (2)$$

where z_0 is a [CLS] token and x is an input sequence of tokens.

The fine-tuning setup is as follow: the hidden size is set to 768 with 12 transformer blocks (Vaswani et al. 2017) and 12 self-attention heads. The batch size is set to 32 to fully utilize the GPU memory. We followed BERT paper (Devlin et al. 2018) and set the dropout probability to 0.1. The learning rate scheduler is slanted triangular learning rates (Howard and Ruder 2018): the base learning rate is $2e - 5$, and the warm-up proportion is 0.1. We empirically set the max number of the epoch to 3 and saved the best model on the validation set for testing.

Given the nature of malware names, we were curious to find out if BERT Cased (BERT-C) outperform BERT Uncased (BERT-UC) due to its capability to perform name entity recognition. We fine-tuned BERT-C and BERT-UC with identical hyper-parameters and evaluated the models on two inputs. First, the models take company names and snippets as an input (Input I), and is then able to learn the relationship between the two, and thus may predict if the security event is about the company. Second, the models only take news snippets as an input (Input II). Here we aim to evaluate the performance of our models in learning a selected company's association to security events given the fact that our models were not fine-tuned for the selected company. If we achieve a comparable performance we can scale our models to any new companies. The two inputs are

- I: [CLS]+Company tokens+[SEP]+snippets tokens
- II: [CLS]+snippets tokens

Corporate cyber-security expansion

Leveraging corporate analysts' insight and existing embedding we expanded a given set of 19 corporate cyber-security risks to their variations and domain specific phrases using Wikipedia embedding. The initial seed risks are the common cyber-security threats that jeopardize confidentiality, integrity and availability (CIA) (Von Solms and Van Niekerk 2013) and are tailored by expert corporate finance analysts to fit corporate financial cyber-security risk analysis, see Table 1. There are various keywords and phrases that have been used to refer to each of the initial seeds. The initial seeds may also miss important security threats. To unravel these issues we built a seed expansion model.

Wikipedia2Vec is a tool developed and maintained by Studio Ousia and is used for obtaining vector representations of words and entities from Wikipedia (Yamada et al. 2016). Wikipedia2Vec implements a conventional skip-gram model to learn the embeddings of words by predicting neighboring words given each word in a Wikipedia page. It also learns entity embedding by predicting neighboring entities in Wikipedia's link graph. It then place similar words and entities near one another in the vector space and learns embedding by predicting neighboring words given each entity after obtaining referenced entities and their neighboring words from links contained on a Wikipedia page.

	Attacks and risks
Confidentiality	Data breach
	Phishing
	Ransomware
	Malware
	Social engineering attack
	Packet sniffing
	Wiretapping
	Web Skimming
	Keylogging
	Password cracking
	Dumpster diving
Integrity	Salami Slicing
	Session hijacking
	Man in the middle
Availability	DOS: Denial of Service
	DDOS: Distributed Denial of Service
	SYN flood
	Zero-day vulnerability
	Crypto jacking

Table 1: The initial seeds that we used to build the risk expansion model are listed in the attacks and risks column.

The pre-trained Wikipedia embedding captures the semantic similarity among words and also between words and entities. We used the English Wikipedia embedding for this task, which is comprised of 300 dimensions embedding for around 4.5 million words and entities. We calculated the pairwise cosine similarity with the following equation 3 among seed terms from table 1 and 4.5 million words and entities in Wikipedia's embedding. We then expand each seed term with the top 10 most similar words and entities from Wikipedia's embedding.

$$\frac{s^T w}{|s|.|w|} \qquad (3)$$

News ingestion

For each seed term from Table 1, we use the top 10 terms with the highest cosine similarity. These expanded cyber-security terms form an alert trigger term list which we use as search terms to query Google Alerts. Google Alerts is a service that enables users to track news by matching the search terms set by users. We added the list of terms to Google Alerts and collected any results from the news feed on an hourly basis. The Google Alerts output is similar to Google Search in which it includes the news headlines, summaries, publish date and news article links. We then used the news crawler and filtering modules introduced earlier to retrieve news snippets from news articles links. It then uses

	Negative	Positive
is_cyber_attack	5,554	2,135
is_company_victim	6,323	1,366

Table 2: We manually labeled 7,689 news snippets. The size of negative and positive labels for each prediction task are shown.

our event detection module to determine the probability of is_cyber_attack and is_company_victim for all news snippets.

Company cyber-security alerts

Our News alert module sends an automatic daily alert email to users if a cyber-security event was found for their selected companies. The alert includes the articles associated with the cyber-security event.

Since retrieved news articles may include multiple news snippets, we first assigned a score to an article by selecting the highest prediction probability of its snippets and filtered out the news articles with a prediction probability below 0.7. We set the probability threshold low to not miss the positive class. The remaining news articles are ranked by their prediction score and will be included in the alert email.

Evaluation

Using the news crawler module we retrieved around 25,000 cyber-security news articles from 2005 to 2019.

For the sake of this study we focused on 22 companies. The list of 22 companies were given to us by expert cyber-security analysts and the companies are known to have dealt with cyber-security risks in the past. We applied a filtering model to the collected 25,000 historical cyber-security news articles and retrieved over 50,000 snippets for the 22 companies.

Deep cyber-security event detection

We sampled and labeled up to 7,689 of 50,000 news snippets in an iterative process to keep the heterogeneous samples across years. Our labeled data is imbalanced and largely skewed toward negative labels for both *is_cyber_attack* and *is_company_victem* prediction tasks, see Table 2. Since our data is not inherently significant to drift over time, BERT is capable of handling imbalanced classes with no additional data augmentation (Madabushi, Kochkina, and Castelle 2020; Devlin et al. 2018).

To build a training set, a validation set, and a test set, we split the data by time to avoid data leakage. Multiple news articles are often published about the same cyber-security event, splitting the data at random may lead to include articles related to the same event in both the training set and the test set.

We evaluated the performance of our model by using three sets of data using a rotating sliding window method to break the training set, the validation set, and the test set, see Figure 4. Since our label data is limited, we used a rotating sliding window to achieve a reasonable size of training data. The 3-fold cross validation sets are created by selecting data in 2017 (Fold 3), 2018(Fold 2), 2019 (Fold 1) as test data. The prior year's data was taken as a validation set and the rest of the years were used as a training set. Table 4 shows the sample size at each fold.

Due to the imbalanced nature of our data, we chose a F1-Score and an AUC-ROC score as model evaluation metrics. We also reported precision and recall to show how our models handle False positives and False negatives. We compared the performance of BERT Cased (BERT-C) and BERT Uncased (BERT-UC) given two sets of supervised inputs, while input I includes pairs of snippets and their associated company token, the input II only includes the snippets. Here we are mainly interested in finding out to what extend the additional information in input I decreases the prediction error of is_company_victim prediction task. If the the deviation among errors is not significant, we can scale our model to be tested for a larger set of companies using our current fine-tuned models without worrying about obtaining a new set of label data for any newly introduced companies.Table 5 shows the average evaluation metrics across three folds for two prediction tasks across two models with two inputs.

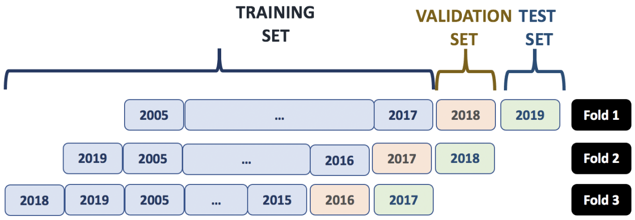


Figure 4: The sliding window to build a training set, a validation set and a test set.

	Class	Fold 1	Fold 2	Fold 3
Training set	Negative Positive	4,100 1,599	4,483 1,668	4,452 1,469
Validation set	Negative Positive	658 211	413 256	689 410
Test set	Negative Positive	796 325	658 211	413 256

Table 3: We evaluated the performance of our models using three sets of data. The size of data in each set is shown for is_cyber_event.

	Class	Fold 1	Fold 2	Fold 3
Training set	Negative Positive	4,631 1,068	5,051 1,100	4,892 1,029
Validation set	Negative Positive	740 129	532 137	899 200
Test set	Negative Positive	952 169	740 129	532 137

Table 4: We evaluated the performance of our models using three sets of data. The size of data in each set is shown for is_company_victim.

While both prediction tasks achieved satisfactory evaluation scores, our models overall performed better in predicting cyber-security events than predicting if a company is a victim of a cyber-security event. BERT-UC with Input I performed the best with an average F-score 0.87 for the is_cyber_attack prediction task and an average F-score 0.74 for the is_company_victim prediction task 2.

BERT-C model with Input I and II achieved a similar average F-score of 0.87 in predicting is_cyber_attack. However, in the remaining cases input I had a slightly higher average F-score. In the majority of cases BERT-UC performed slightly better than BERT-C.

We also reported the AUC-ROC score. While The ROC (Receiver Operator Characteristic) curve estimates a threshold in which two classes are separated in TPR(true positive rate) against a FPR (false positive rate) plot, AUC is used as a summary of ROC curve. Our models achieve a high average AUC-ROC score which shows that our classifiers are able to predict more numbers of true positives and true negatives than false positives and false negatives.

To show how our models tolerate false positive and false negative rates we reported Precision and Recall as well. Average Precision and Recall scores for prediction task 1 are consistent across all models, however BERT-UC shows a higher average Precision score for prediction task 2 and thus lower false positive. Our models attain a higher Recall score than Precision in predicting is_cyber_attack which indicates a low false negative rate meaning that our models are returning a majority of all positive results.

We also show the evaluation metrics for two prediction tasks across two models with two inputs for each fold Table 6. The table also includes the evaluation results for Text-CNN (Kim 2014). The Text-CNN model is trained on the same training data with filter windows 1, 2, 3, 5 and 36 filters for each of the 5 epochs. We used publicly available glove vectors that were trained on 840 billions words as the initialized word vectors.

Corporate cyber-security risk expansion

Our cyber-security risk expansion model extends the initial security risks to large sets which is then used to retrieve relevant cyber-security news articles. Table 7 shows the top five phrases that have been found using Wikipedia embedding. We used T-distributed Stochastic Neighbor Embedding(t-SNE) visualization technique (Maaten and Hinton 2008) to show the extended terms are clustered together. The t-SNE algorithm maps multi-dimensional data to two or more dimensions, where points which were initially distant are also located far away, and close points are also converted to close ones.

We visualized the 300D Wikipedia2Vec vectors in a 2D space. Figure 5 illustrates a pattern where similar words that are generated from the same seed term are closer to each other as well as the between-group relations of different seed terms.

Conclusion

The growing importance of corporate cyber-security risk analysis has generated a rising demand for efficient tools and automatic processes that can assess cyber-security risks in a timely manner. We leveraged advances in AI and NLP to develop a platform that detects cyber-security events and informs corporate cyber-security analysts on a daily basis. Our deep cyber-security event detection model is built on BERT pre-training and we fine-tuned BERT for the cyber-security event detection task. While our model achieves low prediction errors, it can be improved. In the future, as we collect more label data, we aim to pre-train BERT. Also, instead of only using the final hidden representation we will try concatenating hidden representations from the top four hidden layers. Finally, We will fine-tune each layer with decayed learning rates as the learning rates are smaller for the top layers than they are for the lower layers, the lower layers of the BERT model may contain more general information.

References

Alghamdi, W. N. M.; and Rastogi, R. 2020. An efficient data flow material model (DFMM) for cyber security risk assessment in real time server. *Materials Today: Proceedings* .

Apruzzese, G.; Colajanni, M.; Ferretti, L.; Guido, A.; and Marchetti, M. 2018. On the effectiveness of machine and deep learning for cyber security. In *2018 10th International Conference on Cyber Conflict (CyCon)*, 371–390. IEEE.

Barrón-Cedeno, A.; Da San Martino, G.; Jaradat, I.; and Nakov, P. 2019. Proppy: A system to unmask propaganda in online news. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 9847–9848.

Behzadan, V.; Aguirre, C.; Bose, A.; and Hsu, W. 2018. Corpus and deep learning classifier for collection of cyber threat indicators in twitter stream. In *2018 IEEE International Conference on Big Data (Big Data)*, 5002–5007. IEEE.

Burel, G.; Saif, H.; Fernandez, M.; and Alani, H. 2017. On semantics and deep learning for event detection in crisis situations .

Dabiri, S.; and Heaslip, K. 2019. Developing a Twitter-based traffic event detection model using deep learning architectures. *Expert systems with applications* 118: 425–439.

Dai, A. M.; and Le, Q. V. 2015. Semi-supervised sequence learning. In *Advances in neural information processing systems*, 3079–3087.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* .

Fang, Z.; Fei, F.; Fang, Y.; Lee, C.; Xiong, N.; Shu, L.; and Chen, S. 2016. Abnormal event detection in crowded scenes based on deep learning. *Multimedia Tools and Applications* 75(22): 14617–14639.

Ferrag, M. A.; Maglaras, L.; Moschoyiannis, S.; and Janicke, H. 2020. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications* 50: 102419.

Howard, J.; and Ruder, S. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146* .

Methods	is_cyber_event				is_company_victim			
	F1-Score	AUC-ROC	Precision	Recall	F1-Score	AUC-ROC	Precision	Recall
BERT-C I	0.87	0.97	0.83	0.91	0.73	0.95	0.75	0.73
BERT-C II	0.87	0.97	0.83	0.91	0.7	0.94	0.66	0.77
BERT-UC I	0.88	0.98	0.84	0.93	0.74	0.95	0.77	0.73
BERT-UC II	0.86	0.97	0.83	0.9	0.73	0.96	0.73	0.74

Table 5: We show the average evaluation results across three folds for BERT Cased(BERT-C) and BERT Uncased (BERT-UC) models for two different inputs. Input I includes news snippets and company tokens associated with the news snippets. Input II only includes news snippets.

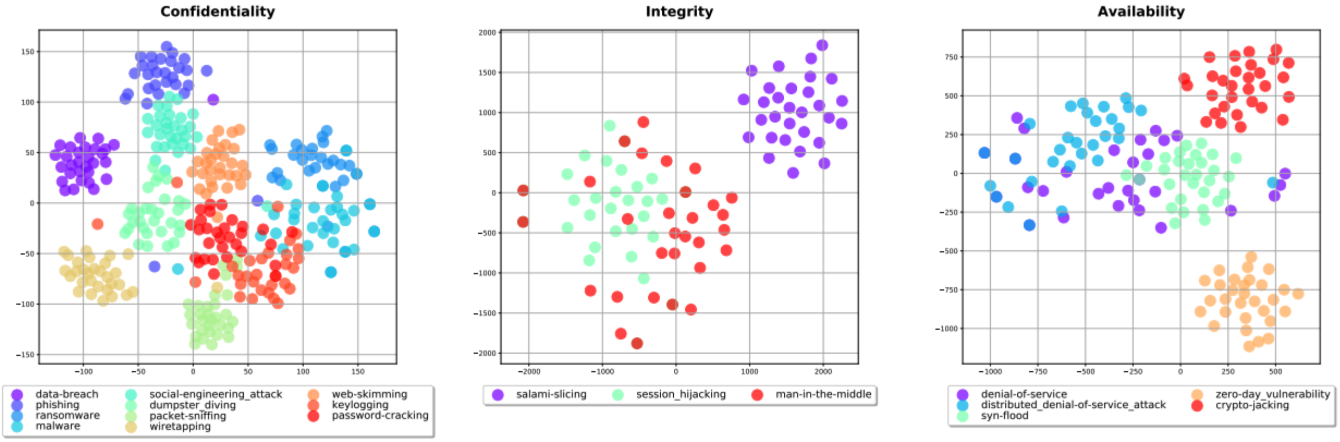


Figure 5: Visualizing wiki Wikipedia2Vec in a 2D space using t-SNE and we see how expanded terms for each seed differ from other groups.

Kim, Y. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* .

Kumar, J.; Goomer, R.; and Singh, A. K. 2018. Long short term memory recurrent neural network (lstm-rnn) based workload forecasting model for cloud datacenters. *Procedia Computer Science* 125: 676–682.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9(Nov): 2579–2605.

Madabushi, H. T.; Kochkina, E.; and Castelle, M. 2020. Cost-Sensitive BERT for Generalisable Sentence Classification with Imbalanced Data. *arXiv preprint arXiv:2003.11563* .

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* .

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding with unsupervised learning. *Technical report, OpenAI* .

Rashkin, H.; Choi, E.; Jang, J. Y.; Volkova, S.; and Choi, Y. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, 2931–2937.

Satyapanich, T.; Ferraro, F.; and Finin, T. 2020. CASIE: Extracting Cybersecurity Event Information from Text. *UMBC Faculty Collection* .

Trong, H. M. D.; Le, D. T.; Veyseh, A. P. B.; Nguyen, T.; and Nguyen, T. H. 2020. Introducing a New Dataset for Event Detection in Cybersecurity Texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5381–5390.

Tuor, A.; Kaplan, S.; Hutchinson, B.; Nichols, N.; and Robinson, S. 2017. Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. *arXiv preprint arXiv:1710.00811* .

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Von Solms, R.; and Van Niekerk, J. 2013. From information security to cyber security. *computers & security* 38: 97–102.

Weng, J.; and Lee, B.-S. 2011. Event detection in twitter. *Icwsm* 11(2011): 401–408.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* .

Xu, D.; Ricci, E.; Yan, Y.; Song, J.; and Sebe, N. 2015. Learning deep representations of appearance and

score	**Methods**	**is_cyber_event**			**is_company_victim**		
		Fold1	Fold2	Fold3	Fold1	Fold2	Fold3
F1-Score	BERT-C I	0.89	0.86	0.87	0.71	0.77	0.73
	BERT-C II	0.89	0.84	0.88	0.72	0.74	0.66
	BERT-UC I	0.91	0.87	0.87	0.72	0.76	0.76
	BERT-UC II	0.89	0.85	0.86	0.71	0.76	0.74
	Text-CNN I	0.89	0.86	0.85	0.68	0.70	0.68
	Text-CNN II	0.87	0.86	0.85	0.68	0.69	0.67
Precision	BERT-C I	0.86	0.81	0.82	0.76	0.71	0.79
	BERT-C II	0.86	0.79	0.85	0.77	0.69	0.54
	BERT-UC I	0.9	0.83	0.81	0.83	0.74	0.75
	BERT-UC II	0.88	0.82	0.81	0.77	0.72	0.71
	Text-CNN I	0.86	0.86	0.78	0.72	0.69	0.72
	Text-CNN II	0.84	0.85	0.82	0.71	0.67	0.70
Recall	BERT-C I	0.93	0.91	0.91	0.67	0.84	0.68
	BERT-C II	0.92	0.89	0.92	0.67	0.79	0.86
	BERT-UC I	0.93	0.91	0.95	0.64	0.78	0.77
	BERT-UC II	0.91	0.88	0.91	0.66	0.81	0.77
	Text-CNN I	0.92	0.87	0.91	0.64	0.71	0.65
	Text-CNN II	0.91	0.89	0.90	0.66	0.71	0.65
AUC-ROC	BERT-C I	0.97	0.98	0.97	0.95	0.97	0.95
	BERT-C II	0.97	0.97	0.97	0.96	0.96	0.90
	BERT-UC I	0.98	0.98	0.98	0.95	0.96	0.95
	BERT-UC II	0.98	0.97	0.96	0.96	0.97	0.93
	Text-CNN I	0.97	0.97	0.97	0.94	0.94	0.94
	Text-CNN II	0.97	0.97	0.97	0.94	0.94	0.94

Table 6: We show the evaluation results for BERT Cased(BERT-C), BERT Uncased (BERT-UC), and Text-CNN models for two different inputs. Input I includes news snippets and company tokens associated with the news snippets. Input II only includes news snippets.

motion for anomalous event detection. *arXiv preprint arXiv:1510.01553* .

Yamada, I.; Shindo, H.; Takeda, H.; and Takefuji, Y. 2016. Joint learning of the embedding of words and entities for named entity disambiguation. *arXiv preprint arXiv:1601.01343* .

Attacks and risks	Expanded terms	Attacks and risks	Expanded terms
Data Breach	Data breach incidents Medical data breach Inform affected customers Identity theft Hacking-related	Session hijacking	Cookie Tampering Cookie hijacking Firesheep Request forgery Sidejacking
Phishing	Phishing Attacks Phishing Emails Social engineering Spear phishing Phishing Scams	Man In The Middle	Mitm Man In The Middle Attack Cross site request forgery Replay attack Session hijacking
Ransomware	Cryptolocker Cryptowall Wannacry Ransomware attacks Malware	Denial of Service	Distributed denial of service attack DoS DoS attack APDoS Syn flood
Social engineering Attack	Spear Phishers Socially Engineered Attacks Right-to-Left Override Data-entry Phishing Relies on Social Engineering	Syn Flood	Mac Flooding Network-layer attacks State Exhaustion Udp amplification attacks Man in the middle position
Dumpster Diving	Gaining Unauthorized Physical Criminally Fraudulent Process Committing Identity Theft Cryptographic Secrets Keystroke Loggers	Zero-day Vulnerability	Zero-day flaw Zero-day exploit Critical vulnerability Security flaw Security hole
Packet Sniffing	Packet analyzer Encryption Enabled Cold-booting Packet Sniffer Log Your Keystrokes	Malware	Malicious software Trojan Ransomware Banking trojans Malicious code
Wiretapping	Nsa Warrantless Covert Listening Device Telephone Tapping Wire Tapping Wiretap	Salami Slicing	Salami attack Grey hat hacking Networks ransomware Crypto anarchists Destructive cyber attacks
Web Skimming	Romance Scams Magecart Like Credit-card Stealing Crypto Thieves Credit-card Stealing	Distributed denial of service	Denial of service attack DoS attack DDoS attack Massive distributed denial Gbps attack
Keylogging	Keystroke Logging Logging Keystrokes Form Grabbing Recording Keystrokes Keystroke Logger	Crypto Jacking	Cryptojacking Crypto mining Coinhive Ransomware Crypto-mining malware
Password Cracking	Trojan Backdoor Trojans SQL Injection Bug Serious Security Hole Second-stage Backdoor		

Table 7: The initial seeds and their associated top five expanded phrases are shown.