applied sciences

MDPI

Article

Detecting Cyber Threat Event from Twitter Using IDCNN and BiLSTM

Yong Fang [1,2], Jian Gao [1], Zhonglin Liu [3,*] and Cheng Huang [1,2]

1 College of Cybersecurity, Sichuan University, Chengdu 610065, China; yfang@scu.edu.cn (Y.F.); myndtt@stu.scu.edu.cn (J.G.); codesec@scu.edu.cn (C.H.)
2 Guangxi Key Laboratory of Cryptography and Information Security, Guilin 541004, China
3 Information Security Institute, Sichuan University, Chengdu 610065, China
* Correspondence: jungleforsa@gmail.com

Received: 20 July 2020; Accepted: 22 August 2020; Published: 26 August 2020

Abstract: In the context of increasing cyber threats and attacks, monitoring and analyzing network security incidents in a timely and effective way is the key to ensuring network infrastructure security. As one of the world's most popular social media sites, users post all kinds of messages on Twitter, from daily life to global news and political strategy. It can aggregate a large number of network security-related events promptly and provide a source of information flow about cyber threats. In this paper, for detecting cyber threat events on Twitter, we present a multi-task learning approach based on the natural language processing technology and machine learning algorithm of the Iterated Dilated Convolutional Neural Network (IDCNN) and Bidirectional Long Short-Term Memory (BiLSTM) to establish a highly accurate network model. Furthermore, we collect a network threat-related Twitter database from the public datasets to verify our model's performance. The results show that the proposed model works well to detect cyber threat events from tweets and significantly outperform several baselines.

Keywords: event detection; cyber threat; Twitter data; machine learning

1. Introduction

With the popularity of the network today, cyber threats happen every moment, which seriously interferes with people's lives and causes social unrest. According to the World Economic Forum in 2019 [1], the incidence of data fraud or theft and cyber attacks has dramatically increased economic risks and societal risks. The frequency and severity of cyber attacks and cyber crimes are only expected to increase [2]. Timely detection of cyber threats can effectively reduce the losses of relevant organizations and provide reliable evidence and proof for the prosecution at the trial stage [3].

Twitter is one of the top-ten most visited websites on the world wide web. As a social networking and microblogging service, many public events are published on the platform every day. People also share cyber threat events in their tweets, such as zero-day exploits, ransomware, data leaks, security breaches, DDoS, vulnerabilities, etc. [4]. Twitter can thus help researchers measure the interest raised by specific topics and automatically detect some unexpected cyber threat events in real-time [5].

However, detecting cyber threat incidents on Twitter is still challenging. Although tweets are limited to 140 characters, people's writing styles are flexible, word abbreviations are used frequently, and various forms of expression intensify and word modification [6]. Second, the content of advertising tweets is so varied and confusing, and people tend to abuse hashtags in their tweets to get views and attention. Those all make automatic detection of tweets very difficult.

Deep learning is widely used in the domain of pattern recognition. Furthermore, the classification problem, such as text classification and image classification, also has shown efficiency. However, most deep learning tasks are single-task learning now. For complex problems, researchers need to

decompose them into simple independent subproblems and solve them separately [7]. The multitask learning (MTL) [8] method can deal with related tasks concurrently. The related tasks share a portion of the factor so that each task can gain additional information learned by other tasks during the learning process. Multitask learning with an association in this way is usually better than single-task learning and also can reduce the probability of overfitting [9].

In this paper, we present a multitasks approach to merge two tasks that detect the cyber threat event from Twitter and Named Entity Recognition (NER) for tweets. We collect numerous tweets and select and filter the samples that meet the task's criteria manually. We also use the Latent Dirichlet Allocation (LDA) [10] model to get the keywords of each tweet to get more critical information from tweets. To obtain a more objective evaluation of results, we conducted single-task experiments and compared several baseline methods. We also compared our approach with the previous experiment method. Moreover, the result showed that the MTL model could get an improvement in cyber threat event detection while being able to maintain the performance of NER. The paper's contributions can be summarized as follows.

- We combine the existing machine learning algorithms and propose a approach to detect cyber threat events from tweets effectively. The proposed model got excellent results and achieved an f1-score rate of 96.4% under 5-fold cross-validation on cyber threat event detection.
- We propose the MTL model to improve the cyber threat event detection task's performance while maintaining the NER performance on the dataset. Moreover, the result showed that the proposed model could achieve more outstanding performance by comparing the f1-score with previous work.

The rest of the paper is organized as follows. In Section 2, we have a review of related work about event detection on Twitter and the cyber threat event detection. In Section 3, we introduce the proposed model architecture in detail. In Section 4, we introduce the data set used in this work, introduce the experimental operation in detail, and show the experimental environment and experimental evaluation criteria. Next, we present the experimental results and evaluate them with previous work. In Section 5, we summarize the conclusion and propose future works.

2. Related Work

2.1. Event Detection on Twitter

Event detection based on Twitter is widespread. Since 2010 and even earlier, researchers use Twitter as an extensive database of data source which to analyze and extract events. There is a wide variety of events to detect, including controversial events [11], sports-related events [12,13], geospatial events [14], and general events [15–18]. The researchers studied different types of events, and the focus of their tweets varied. Phuvipadawat et al. [19] extract features of the hashtags, followers, and the timestamps from the tweets to detect breaking news event. Cordeiro et al. [20] focus more on the tweet topics and frequency of the hashtags. Kaleel et al. [21] detect the events based on timestamps, geolocations, and cluster size. Multimodal et al. [22] detect events by analyzing the tags in the dataset. They all achieved excellent results. Many researchers pay more attention to the detection algorithm. Dabiri et al. [23] utilize the deep learning architectures for detecting traffic incidents. Saeed et al. [24] propose a novel approach named Weighted Dynamic Heartbeat Graph (WDHG) to detect events from the Twitter stream. Nazir et al. [25] combine average moving threshold algorithm and Gaussian algorithm detection for signal detection in tweets sentiment and top hashtag. Sani et al. [26] use locality sensitive hashing to approximately find similar items and incremental clustering to implement a practical real-time event detection algorithm. Researchers try to capture the characteristics of tweets and use appropriate algorithms to solve the problems they study.

2.2. Cyber Threat Event Detection

The detection of cyber threat events is a hot topic. Kang et al. [27] propose a network assessment framework to help cyber defenders better understand the global situation. Qiu et al. [28] thoroughly

analyze the relevant features of sentences to classify cyber attacks. They also pointed out that the trigger matching method is most suitable for event type detection, and the performance of the embedded word feature model trained with a large corpus is much better than other models. Khandpur et al. [29] propose a new query extension strategy based on convolutional kernel and dependency resolution to detect a wide range of network attacks, including DDoS, data breaches, and Twitter account hijacking. However, this approach is highly computationally intensive due to the cost of autoencoder training. Le Sceller et al. [30] propose an automatic, self-learned framework to detect, geolocate, and categorize cyber threat events based on a set of keyword seeds. This algorithm is good at detecting known threats, but may not be good at detecting potential cyber threat events. Bose et al. [31] use an unsupervised machine learning approach to the detection of cyber threat events on Twitter. This approach can extract the tweet terms that are characterized as named entities, keywords, or both to get an importance score and enables the ranking of cyber threat event by it. However, the feature processing of text is too one-sided and easy to overfit. Finally, Ji et al. [32] takes a supervised learning approach by proposing a novel multitask learning-based model to detect cyber threat event on tweets datasets. This method is characterized by complex treatment and general practical effects.

2.3. Multi-Task

MTL improves the generalization ability of models by using shared representations to perfect the learning process for different tasks. Moreover, in some cases, the multitasks model can provide better performance than the single-tasks model. Zhang, Zhanpeng, and Luo et al. [33] investigate the possibility of improving the detection of facial landmark robustness through multitask learning and get good results. Søgaard et al. [34] present a multitask learning architecture with a deep two-way neural network. Experimental results show that the low-level tasks can be better retained at the bottom, so that high-level tasks can use the shared representation of the underlying tasks. The architecture of Wehrmann et al. [35] includes a Convolutional Neural Network (ConvNet), which has two different outputs for classifying the sentiment and language of tweets, and the model has a noticeable effect.

3. The Proposed Model Architecuture

Just as shown in Figure 1, our multi-task learning integrates the two subtasks: cyber threat event detection and NER. They share the same word embedded layer. To explain the model clearly, we will introduce the model architecture of the two subtasks separately.

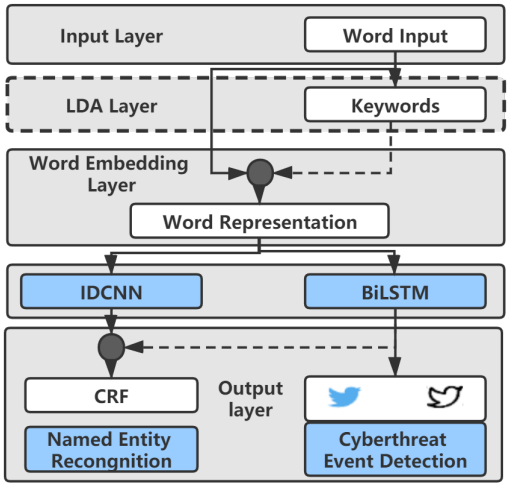

Figure 1. The architecture of multitask Learning in our method.

3.1. Cyber Threat Event Detection

We implemented the cyber threat event detection using the LDA and BiLSTM. Moreover, the architecture of cyber threat event detection is described below.

The layer of LDA: LDA can predict the distribution of topics in documents. It can give each document the topic in the document set in the form of the probability distribution. In this layer, we use LDA to extract the subject keywords of tweets, which can summarize the tweet's content to a certain extent and facilitate the detection of cyber threat events.

The layer of word Embedding: This layer functionality just uses the trained language model (GLOVE [36] or word2vec [37]) that maps the words to the value vector, as the Equation (1) shows. The word vector model replaces each word with its corresponding vector, which can reflect the semantic distance and the relationship between words, then gets a news article matrix $\boldsymbol{X} \in \mathbb{R}^{n \times d}$. Where the n is the number of the word in the tweets, and the d is the dimension of word embedding,

$$\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_i) \tag{1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^d$ is the d-dimensional word vector corresponding to the i-th word in the tweets.

The BiLSTM Model: The content of a tweet is limited, and the sentences will be shorter after removing the useless information. The BiLSTM network takes into the sequential order of words in the sentence and can better capture bidirectional semantic dependencies of words. Moreover, the architecture of BiLSTM as shown in the Figure 2.

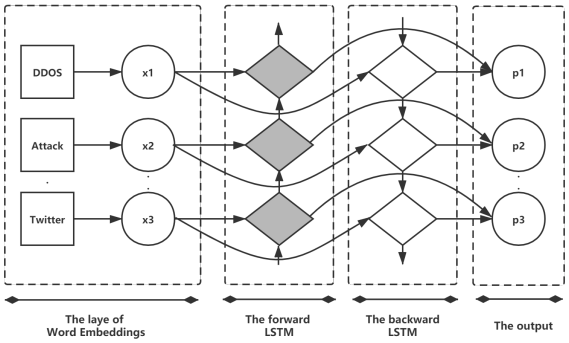

Figure 2. The architecuture of BiLSTM.

The layer of Output: We use the *Dropout* to select the feature in the network's output layer to prevent promote generalization and overfitting. Feature nodes are used by a fully connected *softmax* layer, which outputs the probability of tweets containing the cyber threat event information.

3.2. Named Entity Recognition

In the NER task, we aim to extract information from the tweets. Illustrated in Figure 3, our model is based on an Iterated Dilated Convolutional Neural Network (IDCNN) [38] and BiLSTM. We used the same word embedding layer of cyber threat event detection, which can map each word to a sequence, and use the IDCNN model and BiLSTM model to extract features. Because the dimensions extracted by the two models are different, the dimensions need to be compressed, transformed, and then spliced. Finally, the spliced result is used as the input of the CRF layer. The BiLSTM has been introduced earlier; next, we will introduce the IDCNN.

The IDCNN Model: Just as shown in Figure 4, the realization of dilated convolution is the same as that of standard convolution, except that dilated convolution is filled with gaps in the convolution kernel. Therefore, it can expand the convolution kernel's perception horizon and obtain multi-scale information, which is beneficial to the neural network to obtain the context information of tweets and improve the model's performance. In the model, we first conduct two conventional one-dimensional convolution operations on the word embedding layer's output, followed by a dilated convolution operation to replace the maximum pooling layer in the standard convolutional neural network finally connect the output with two full connection layers.

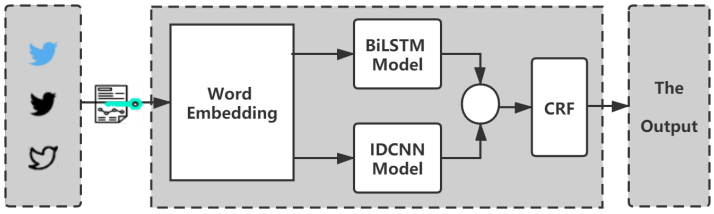

Figure 3. The architecture of Named Entity Recognition (NER) in our method.

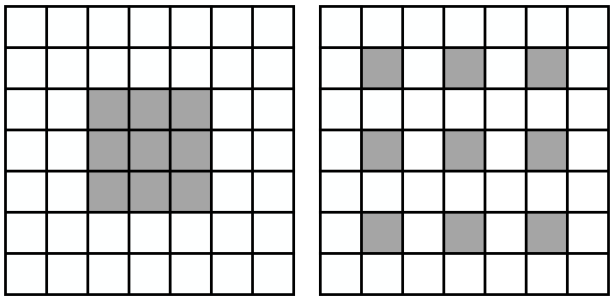

Figure 4. The dilated convolution diagram. (**a**) The standard convolution, and the size of convolution kernels is 3 × 3. (**b**) The dilated convolution, and the size of convolution kernels is 3 × 3 and the dilation rate is 1.

The layer of Output: BiLSTM network can capture the sequence dependence of sentences; simultaneously, the IDCNN network can better capture the sentence context information. The matrix obtained by combining the output results of the two can enrich the sentence information. The two models can only extract statement characteristics, but cannot predict the current entity tags, and cannot consider the dependencies between tags. The CRF algorithm can consider the adjacent relation between tags and obtain the global optimal tag sequence, so we use it as the output layer of the NER task. CRF defines a tag transfer score, where the score from an input sequence to a tag sequence can be expressed as the below Equations (2)–(4),

$$s(x, y) = \sum_{i=1}^{n} \left(W_{y_{i-1}, y_i} + P_{i, y_i} \right) \tag{2}$$

where x is the given sequence and y is the corresponding tag sequence. W is the transformation matrix, $W_{i,j}$ represents the label transfer score, and P_{i,y_i} represents the score of the y_i tag of the character. The maximum likelihood function of W is as below.

$$L = \sum_{i=1}^{n} \log \left(P \left(y_i \mid x_i \right) \right) + \frac{\lambda}{2} \|\theta\|^2 \tag{3}$$

where λ and θ are the regularization parameters, and P represents the corresponding probability from the original sequence to the prediction sequence; its formula is expressed as below.

$$P(y \mid x) = \frac{e^{s(x,y)}}{\sum_{y \in Y_x} e^{s(x,y)}} \tag{4}$$

4. Experiments Setup

Our experiment aims to detect tweets with information about the cyber threat event. This supports the hypothesis that the use of multi-task learning improves model detection performance and increases the reliability of experimental results because the probability of overfitting is reduced during multi-task

training. To this end, we first introduce the datasets and evaluation metrics. Next, we determine the parameters and data feeding methods used in the experiment. We then compare the performance of our method with several traditional methods. Finally, we performed experiments with additional public data sets to verify the feasibility of our approach.

4.1. The Datasets and Experimental Environment

The Data Collection. To verify our method's validity, we collected lots of tweets using Twitter's API over a period from 1 January 2019 to 1 June 2020 using the keywords seeds related to network security for model training. Such as "DDOS", "ransomware", "data breach", "phishing", and others. We also collect the tweets from the user of "@realDonaldTrump" as the raw samples that are not related to cybersecurity, the raw data can be seen in the file(https://github.com/das-lab/Cyberthreat-Detection).

The Data Preprocessing. In order to train more accurate models, the raw tweets need to be fully preprocessed. We removed the tweets with repetitive content and remove meaningless characters in tweets, including emoticons and remove the words that appeared less than twice in all tweets. We replace the URL that appears in the tweet. Then, we tokenize tweets using the Natural Language Toolkit (NLTK). Moreover, we use the Stanford CoreNLP to extract named entities. After preprocessing, the labeled dataset is balanced as 2525 cyber threat event tweets and 2410 non-cyber threat event tweets.

The Experimental Environment. The experimental environment is shown in Table 1. We show the Python library we used and the operating system version and associated configuration of the machine.

Table 1. Experimental environment configuration.

Items	Configuration
OS	Ubuntu 16.04.3 TLS
The system configuration	CPU: Intel i7-7700, RAM: 16G, GPU: GeForce GTX 2080 8G
The library of python	keras, Scikit-learn, gensim, Matplotlib, pandas, numpy, gensim

4.2. Metric

For evaluation of the cyber threat event detection task and NER task, as shown in Table 2, the confusion matrix has four categories: TP indicates that tweets with a threatening event are actually be detected as threatening. FP indicates that tweets without a threatening event are actually be detected as threatening. In the same way, FN indicates that tweets without a threatening event are actually be detected threatening. TN indicates that tweets without a threatening event are actually be detected non-threatening. We use precision, recall, and f1-score to measure the performance of the classifier. The specific calculation formula is shown as follows.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{F1-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

As shown in the equations, the precision rate represents the accuracy of predicting threat event samples, and the recall rate represents the probability of being predicted as threat event samples in the actual threat event samples. The accuracy rate can judge the accuracy rate of the model, and the f1-score comprehensively reflects the precision rate and accuracy rate. The higher the values, the better the prediction result of the model. Similarly, for the NER task, we calculated the f1-score to evaluate the performance of the model.

Table 2. Confusion matrix.

	With Threat	Without Threat
Predicted with threat	TP	FP
Predicted without threat	FN	TN

4.3. Train Process

The Keywords of Tweets. As we collect data through keyword seeds, the data have artificial distinguishing features. Therefore, before using the LDA model to get the topic distribution of the dataset and its keywords, we remove the keyword seeds from the tweet, which can effectively reduce the extra features brought from the process of data collection. This preprocessing operation is critical and makes the experiment more rigorous.

Model Setting. In the process of model training, we use ninety percent of the data for the training model, and ten percent of the data for validating model practicability and validity. Moreover, we use the 1D standard convolutional layer with 256 filters of size equal to three words. Moreover, the 1D dilated standard with 512 filters of size equal to four words. The dropout_ratio was 0.5 and the Bidirectional LSTM cell with hidden vectors dimension of 64.

Data Feed. To compare the reliability of the experiment, we not only did the multitasks experiments, but did single task experiments separately. Moreover, in the single task, the cyber threat event detection task and the NER task are under 5-fold cross-validation.

4.4. Evaluation and Result

Single-Task Models

Tables 3 and 4 show the result of the cyber threat event detection task and the NER task. According to the results in Table 3, the use of LDA can improve cyber threat event detection performance. In the NER task, IDCNN with BiLSTM and CRF are better than common entity recognition methods, such as BiLSTM+CRF and IDCNN+CRF.

Table 3. The results of cyber threat events detection.

Method		Precision (%)	Recall (%)	F1-Score (%)
Without LDA	non-cyber threat	95.5	95.3	95.3
	cyber threat	95.4	95.5	95.4
	avg/total	95.4	96.4	95.4
With LDA	non-cyber threat	96.4	96.4	96.6
	cyber threat	96.3	96.5	96.5
	avg/total	**96.4**	**96.4**	**96.4**

Table 4. The results of NER task.

Method	Precision (%)	Recall (%)	F1-Score (%)
BiLSTM+CRF/avg	91.7	93.6	92.7
IDCNN+CRF/avg	91.4	94.5	93.1
BiLSTM+IDCNN+CRF/avg	**92.4**	**94.5**	**93.4**

Multi-Task Models

As previously observed, we compared our model with several competitive baseline methods, as shown in Table 5, and our proposed model performed better; moreover, compared with the single task,

the multi-task learning can effectively improve the f1-score of NER task when the f1-score of cyber threat detection task is roughly unchanged. At the same time, we also feed our model with the data in [39], the dataset has 11,073 cybersecurity-related tweets, and the entity has been labeled. As shown in Table 6, the results showed that the indicators improved to a certain extent using our method.

Table 5. The results of multi-task model.

Method		Precision (%)	Recall (%)	F1-Score (%)
BiLSTM (detection) BiLSTM+CRF(NER)	Cyber threat detection	95.9	95.9	95.9
	NER	91.6	93.4	92.5
BiLSTM (detection) IDCNN+CRF(NER)	Cyber threat detection	95.7	95.7	95.7
	NER	91.6	94.7	93.2
BiLSTM (detection) BiLSTM+IDCNN+CRF(NER)	Cyber threat detection	96.6	96.6	**96.6**
	NER	92.4	95.4	**93.8**

Table 6. The results of the multi-task using data in [39].

Method		Precision (%)	Recall (%)	F1-Score (%)
WordRNN + CharRNN [39]	Cyber threat detection	-	-	92.2
	NER	-	-	94.0
Our method	Cyber threat detection	95.1	97.4	**96.2**
	NER	94.1	94.0	**94.1**

5. Discussion

In this paper, we focus on the two main tasks: detection of cyber threat events on tweets and the NER task. We conducted the related control experiments separately. Finally, we fused the two separate subtasks. The results show that the multitask model can achieve the effectiveness of two subtasks executed separately. That simplifies the complexity of using deep neural networks dramatically. At the same time, we demonstrate our proposed method's performance by comparing multiple baseline methods, and results show that our proposed method has more outstanding performance. The use of Twitter is very active, and it is also essential to promptly detect new cybersecurity incidents. In the future, we will explore faster and more accurate ways to identify cyber threat events in tweets; we also will make further exploration of event detection and entity recognition to extract the specific cyber threat events from tweets.

Author Contributions: Conceptualization, Y.F., G.J., Z.L., and C.H.; Methodology, C.H. and G.J.; Software, J.G. and Z.L.; Validation, J.G. and Z.L.; Formal Analysis, G.J.; Investigation, Z.L.; Resources, C.H.; Data Curation, G.J.; Writing—Original Draft Preparation, G.J. and Z.L.; Writing—Review & Editing, Y.F. and C.H.; Supervision, C.H.; Funding Acquisition, C.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (No.61902265), Frontier Science and Technology Innovation Projects of National Key Research and Development Program (No.2019QY1405), Sichuan University Postdoc Research Foundation (No.2019SCU12068), Guangxi Key Laboratory of Cryptography and Information Security (No.GCIS201921), and the Fundamental Research Funds for the Central Universities.

Acknowledgments: We thank those anonymous reviewers whose comments/suggestions helped improve and clarify this manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. World Economic Forum. The Global Risks Report 2019. Available online: http://www3.weforum.org/docs/WEF_Global_Risks_Report_2019.pdf (accessed on 25 August 2020).

2. Satyapanich, T.; Ferraro, F.; Finin, T. CASIE: Extracting Cybersecurity Event Information from Text. *Umbc Fac. Collect.* **2020**, *34*, 8749–8757. [CrossRef]
3. Noor, U.; Anwar, Z.; Amjad, T.; Choo, K.K.R. A machine learning-based FinTech cyber threat attribution framework using high-level indicators of compromise. *Future Gener. Comput. Syst.* **2019**, *96*, 227–242. [CrossRef]
4. Yagcioglu, S.; Seyfioglu, M.S.; Citamak, B.; Bardak, B.; Guldamlasioglu, S.; Yuksel, A.; Tatli, E.I. Detecting Cybersecurity Events from Noisy Short Text. *arXiv* **2019**, arXiv:1904.05054.
5. Mazoyer, B.; Cagé, J.; Hervé, N.; Hudelot, C. A French Corpus for Event Detection on Twitter. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 6220–6227.
6. Da Costa Abreu, M.; Araujo De Souza, G. Automatic offensive language detection from Twitter data using machine learning and feature selection of metadata. In Proceedings of the IEEE World Congress on Computational Intelligence (IEEE WCCI), Glasgow, UK, 19–24 July 2020.
7. Ruder, S. An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv* **2017**, arXiv:1706.05098.
8. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
9. Baxter, J. A Bayesian/information theoretic model of learning to learn via multiple task sampling. *Mach. Learn.* **1997**, *28*, 7–39. [CrossRef]
10. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
11. Popescu, A.M.; Pennacchiotti, M. Detecting controversial events from twitter. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, Toronto, ON, Canada, October 2010; pp. 1873–1876.
12. Lanagan, J.; Smeaton, A.F. Using Twitter to detect and tag important events in sports media. In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, 17–21 July 2011.
13. Nichols, J.; Mahmud, J.; Drews, C. Summarizing sporting events using twitter. In Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces, Lisbon, Portugal, February 2012; pp. 189–198.
14. Walther, M.; Kaisser, M. Geo-spatial event detection in the twitter stream. In Proceedings of the European Conference on Information Retrieval, Moscow, Russia, 24–27 March 2013; pp. 356–367.
15. Zhou, X.; Chen, L. Event detection over twitter social media streams. *VLDB J.* **2014**, *23*, 381–400. [CrossRef]
16. D'Andrea, E.; Ducange, P.; Lazzerini, B.; Marcelloni, F. Real-time detection of traffic from twitter stream analysis. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2269–2283. [CrossRef]
17. Pierce, C.E.; Bouri, K.; Pamer, C.; Proestel, S.; Rodriguez, H.W.; Van Le, H.; Freifeld, C.C.; Brownstein, J.S.; Walderhaug, M.; Edwards, I.R.; et al. Evaluation of Facebook and Twitter monitoring to detect safety signals for medical products: An analysis of recent FDA safety alerts. *Drug Saf.* **2017**, *40*, 317–331. [CrossRef] [PubMed]
18. Hasan, M.; Orgun, M.A.; Schwitter, R. Real-time event detection from the Twitter data stream using the TwitterNews+ Framework. *Inf. Process. Manag.* **2019**, *56*, 1146–1165. [CrossRef]
19. Phuvipadawat, S.; Murata, T. Breaking news detection and tracking in Twitter. In Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Washington, DC, USA, 31 August–3 September 2010, pp. 120–123.
20. Cordeiro, M. Twitter event detection: Combining wavelet analysis and topic inference summarization. In Proceedings of the Doctoral Symposium on Informatics Engineering, Porto, Portugal, 26–27 January 2012; pp. 11–16.
21. Kaleel, S.B.; Abhari, A. Cluster-discovery of Twitter messages for event detection and trending. *J. Comput. Sci.* **2015**, *6*, 47–57. [CrossRef]
22. Yılmaz, Y.; Hero, A.O. Multimodal event detection in Twitter hashtag networks. *J. Signal Process. Syst.* **2018**, *90*, 185–200. [CrossRef]
23. Dabiri, S.; Heaslip, K. Developing a Twitter-based traffic event detection model using deep learning architectures. *Expert Syst. Appl.* **2019**, *118*, 425–439. [CrossRef]
24. Saeed, Z.; Abbasi, R.A.; Razzak, M.I.; Xu, G. Event detection in Twitter stream using weighted dynamic heartbeat graph approach. *arXiv* **2019**, arXiv:1902.08522.
25. Nazir, F.; Ghazanfar, M.A.; Maqsood, M.; Aadil, F.; Rho, S.; Mehmood, I. Social media signal detection using tweets volume, hashtag, and sentiment analysis. *Multimed. Tools Appl.* **2019**, *78*, 3553–3586. [CrossRef]

26. Sani, A.M.; Moeini, A. Real-time Event Detection in Twitter: A Case Study. In Proceedings of the 2020 6th International Conference on Web Research (ICWR), Tehran, Iran, 22–23 April 2020; pp. 48–51.
27. Kang, M.H.; Mayfield, T. A cyber-event correlation framework and metrics[C]//System Diagnosis and Prognosis: Security and Condition Monitoring Issues III. International Society for Optics and Photonics. *SPIE* **2003**, *5107*, 72–82.
28. Qiu, X.; Lin, X.; Qiu, L. Feature representation models for cyber attack event extraction. In Proceedings of the 2016 IEEE/WIC/ACM International Conference on Web Intelligence Workshops (WIW), Omaha, NE, USA, 13–16 October 2016; pp. 29–32.
29. Khandpur, R.P.; Ji, T.; Jan, S.; Wang, G.; Lu, C.T.; Ramakrishnan, N. Crowdsourcing cybersecurity: Cyber attack detection using social media. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, November 2017; pp. 1049–1057.
30. Le Sceller, Q.; Karbab, E.B.; Debbabi, M.; Iqbal, F. Sonar: Automatic detection of cyber security events over the twitter stream. In Proceedings of the 12th International Conference on Availability, Reliability and Security, Reggio Calabria, Italy, August 2017; pp. 1–11.
31. Bose, A.; Behzadan, V.; Aguirre, C.; Hsu, W.H. A novel approach for detection and ranking of trendy and emerging cyber threat events in twitter streams. In Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Vancouver, BC, Canada, 27–30 August 2019; pp. 871–878.
32. Ji, T.; Zhang, X.; Self, N.; Fu, K.; Lu, C.T.; Ramakrishnan, N. Feature driven learning framework for cybersecurity event detection. In Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Vancouver, BC, Canada, 27–30 August 2019; pp. 196–203.
33. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. Facial landmark detection by deep multi-task learning. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 94–108.
34. Søgaard, A.; Goldberg, Y. Deep multi-task learning with low level tasks supervised at lower layers. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics Volume 2: Short Papers, Berlin, Germany, 7–12 August 2016; pp. 231–235.
35. Wehrmann, J.; Becker, W.E.; Barros, R.C. A multi-task neural network for multilingual sentiment classification and language detection on twitter. In Proceedings of the 33rd Annual ACM Symposium on Applied Computing, Pau, France, 9–13 April 2018; pp. 1805–1812.
36. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, October 2014; pp. 1532–1543.
37. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
38. Strubell, E.; Verga, P.; Belanger, D.; McCallum, A. Fast and accurate entity recognition with iterated dilated convolutions. *arXiv* **2017**, arXiv:1702.02098.
39. Dionísio, N.; Alves, F.; Ferreira, P.M.; Bessani, A. Towards end-to-end Cyberthreat Detection from Twitter using Multi-Task Learning. In Proceedings of the IJCNN 2020, Glasgow UK, 19–24 July 2020.