International Journal of Early Childhood Special Education (INT-JECSE)
DOI:10.9756/INTJECSE/V14I6.308 ISSN: 1308-5581 Vol 14, Issue 06 2022

Security Issues and Defensive Approaches in Deep Learning Frameworks

Karanam rupesh, BTech, Department of CSE, rupesh 18083@gmail.com.
Karanam venkata Sai, B.Tech mechanical engineering, venkatsai1677@gmail.com
Chinnapareddy bhargav sandeep, B.Tech electronics communication and engineering, Bhargavsandeep786@gmail.com

ABSTRACT: The development of deep learning frameworks is a major step forward for AI and has many potential applications. However, security risks associated with deep learning systems are a key impediment to their widespread use. Any attempt by malicious insiders or outsiders to compromise deep learning frameworks will have far-reaching consequences for people's everyday lives. We get things off with a rundown of the deep learning algorithm structure and a careful analysis of its vulnerabilities and dangers. Here, we provide a comprehensive categorization technique for security worries and defensive methods in deep learning frameworks, and we establish connections between different types of threats and the countermeasures that may be taken against them. We also look at a real-world scenario where security flaws in deep learning were present. We conclude with a discussion of future directions and challenges for deep learning architectures. We hope that our efforts will pique the attention of the academic and business sectors in furthering the study of and developing solutions for the security challenges presented by deep learning frameworks.
Topics covered include: adversarial examples; deep learning frameworks; defensive techniques; security concerns.

1. INTRODUCTION

The development of deep learning (DL) algorithms has altered the way that many problems involving large amounts of real-world data are approached, including the management of patient data for disease prediction [1, the performance of autonomous security audits from system logs [2 and the development of self-driving cars using visual object detection [3]. However, research into the privacy and security vulnerabilities of DL-based systems has been conducted in order to protect against cyberattacks. If the input data is inaccurate, the output of a DL-based system might be right or undesired. Blocking the camera lens [5] or jamming the sensors [4] of a self-driving vehicle, for example, might have serious consequences for its performance. Similarly, biometric identification systems that rely on facial recognition might provide false positive results [6] if noise is introduced into the picture or if digitally modified glasses are superimposed on the face [7].

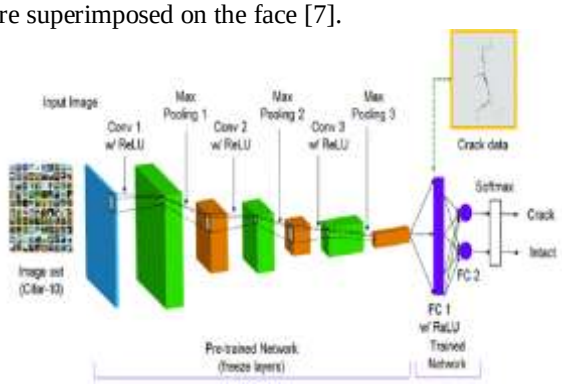

Figure 1: Illustrative Illustration

A defense system was developed to fend off such assaults. For instance, the gradient masking method was proposed by Goodfellow et al.[9]. According to him and his colleagues[11], a defense system that incorporates a wide range of countermeasures is more effective than any one of them in handling hostile situations. Applications built on deep learning are used extensively in the real world, especially in areas where privacy and safety are of paramount significance. The material universe may serve as a counterexample. An adversary may confuse autonomous cars, for instance, by tampering with a traffic sign recognition system[11]. There have been additional assessments of the safety issues of deep learning frameworks. The security flaws identified by Xu et al.[12] may be broken down into three categories: poisoning, escaping, and black-box/white-box attacks. According to Tariq et al[13]categorization .'s approach, attacks may be categorized as either causative, exploratory, targeted, or indiscriminate. However, a unified and systematic perspective on deep learning system security and defense measures was missing from the aforementioned research. When compared to previous polls, our categorization centers on attack stage, adversary

knowledge, attack frequency, attack target, and assault scope. Concerns about deep learning's impact on users' personal information were discussed in Bae et al[14] .'s paper. However, the authors relied heavily on mathematical ideas and algorithms when evaluating security flaws. We also make use of diagrams to show the general concepts and specifics of many attacks. We also discuss the serious software issues present in deep learning frameworks. Qiu et al. [15] discussed the training and testing phases of AI attack techniques, but they did not investigate the connections between these stages and attack and defense mechanisms. We made a solid connection between assaults and defenses against them. We also discussed unanswered questions and future directions for deep learning architectures. Therefore, we did a comprehensive study of the safety of deep learning frameworks and analyzed all the relevant literature.

2. LITERATURE REVIEW

Geospatial data to images: A deep-learning framework for traffic forecasting [1]:

With the advent of deep-learning technology, researchers are aiming to apply deep learning to the subject of traffic forecasting to achieve substantial breakthroughs, similar to what has been done in voice recognition and picture classification research in recent years. Here, we take a look at recent research that uses deep learning methods to address traffic forecasting problems based on geographic data. Using the foundation of prior work, we provide a deep-learning framework that uses state-of-the-art deep-learning methods like Convolutional Neural Networks (CNNs) and residual networks to analyze geographical data in the form of photographs. To showcase the efficacy and ease of our framework, we design the New York cab pick-up/drop-off forecasting issue, and we show that our framework significantly outperforms traditional techniques like Historical Average (HA) and AutoRegressive Integrated Moving Average (ARIMA) (ARIMA).

A better lineup suggestion algorithm for Dota 2 using a bidirectional LSTM [2]:

As the e-sports industry has expanded rapidly in recent years, it has generated a wealth of easily accessible data based on technical requirements. These characteristics make data mining and deep learning ideal tools for guiding players and developing successful tactics. Dota 2 has a large fan base and is one of the most popular e-sports in the world. Because of the importance of a hero's match to the outcome of the game, players often find it difficult to decide which heroes to field in a given fight. In this study, we provide a better solution to this problem by using a bidirectional Long Short-Term Memory (LSTM) neural network model for lineup suggestions in Dota2. The model uses the CBOW model from the Word2vec model to generate hero vectors. The CBOW model can predict how a word will be used in a sentence. Words become heroes, sentences become lineups, and word vectors become hero vectors with the help of the model used in this article. After the first four heroes have been selected, it makes a suggestion for the fifth hero to round out the team, fixing many issues with previous recommendation systems.

Countermeasures Against Deep Learning Adversaries [4]:

Due to its fast development and remarkable achievements in a broad variety of applications, deep learning is being applied in many safety-critical scenarios. The recent discovery that deep neural networks are vulnerable to adversarial instances, or input samples that are purposely created, is a major breakthrough. When testing and deploying deep neural networks, adversarial examples may be quite successful while being undetected to humans. In addition to other risks, deep neural networks are susceptible to attack from adversarial instances if they are used in life-or-death situations. Therefore, there is a lot of focus on arguments against and defenses of counterexamples. In this research, we categorize the various methods for generating adversarial instances, analyze their pros and cons, and assess recent findings on adversarial examples for deep neural networks. The taxonomy's practical uses are investigated, including those in hostile settings. We go further into countermeasures for combative circumstances, analyzing both the challenges and possible solutions.

Detecting memory-related security flaws using a feature-based method [5]:

The software industry has a big challenge when it comes to the development of safe software systems as a direct result of human error or design defects that result in vulnerabilities. In order to improve vulnerability detection, researchers usually use the source code components of vulnerabilities. While these tactics have proven effective, most studies to far have merely provided a conceptual description of vulnerabilities rather than identifying them explicitly. In this research, we provide a novel approach to vulnerability detection that makes use of vulnerability attributes to locate memory-related flaws (MRVDAVF). Specifically, our approach uses three distinct techniques to improve vulnerability detection. We begin by providing an enhanced Control Flow Graph (CFG) and Pointer-related CFG to characterize the features of several common vulnerabilities including memory leak, doublefree, and use-

after-free (PCFG). Then, the Feature Judging (FJ) method and the Vulnerability Judging (VJ) algorithm based on Vulnerability Features are utilized to identify memory-related security flaws (VJVF). Lastly, the proposed model is tested on three examples from the Juliet Test Suite. The experimental results prove the practicality and efficiency of the proposed method.

Neural networks have several interesting characteristics [6]:

Deep neural networks, which are incredibly expressive models, have lately achieved state-of-the-art performance on voice and picture recognition tasks. Their ability to communicate themselves is what makes them effective, but it also causes them to pick up answers to problems that seem impossible at first glance. In this research, we highlight two such characteristics. Before anything else, we use many methods of unit analysis to establish that unique high-level units are identical to random linear combinations of high-level units. This suggests that the upper layers of neural networks are less responsible for encoding semantic information than the space itself. In a second finding, we find that somewhat discontinuous input-output mappings may be learned efficiently by deep neural networks. In particular, we find that we can force the network to misclassify a picture by delivering a precise undetectable perturbation, which is recognized by increasing the network's prediction error. Furthermore, the shape of these perturbations is not random, thus the same input may lead to a different misclassification by a network trained on a different subset of the dataset.

3. METHODOLOGY

While deep learning has its uses, it is not without its limitations. Recent studies have shown that even a well-behaved deep learning network may be easily fooled by well crafted adversarial samples. The most advanced Deep Neural Network was fooled by extremely probabilistic perturbations made by Szegedy et al (DNN). Therefore, the cases in which a DNN gave a wrong classification are known as adversarial samples.

However, security issues are a major roadblock to the widespread use of deep learning systems. Any attempt by malicious insiders or outsiders to compromise deep learning frameworks will have far-reaching consequences for people's everyday lives.

Negatively, assaults against deep learning frameworks from either within or outside the organization will have far-reaching consequences for people's everyday lives.

We get things off with a rundown of the deep learning algorithm structure and a careful analysis of its vulnerabilities and dangers. We provide a highly extensive categorization technique for security challenges and defense tactics in deep learning systems, linking different types of assaults to their countermeasures.

Advantages We look at a real-world example of a security problem using deep learning.

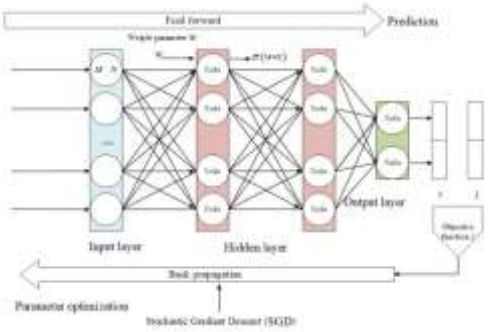

Fig.2: System architecture

MODULES:

We have built the following components to put the aforementioned project into action:

MODULES:

Step One: Gathering Information

See the interdependencies between different factors. an illustration of all of the essential elements from the dataset. Additionally, the dataset is cut in two, with one third used for training algorithms and the other third used for testing. Both the training and testing datasets include examples from each class in about the proper proportions. How the paper's datasets were divided into training and testing sets.

Second, we must do some preliminary processing of the data.

There may be inconsistencies in the data because to missing values. In order to improve the algorithm's efficiency and provide more accurate results, preprocessing the data is essential. Eliminating the outliers and converting the variables are two steps that need to be taken. To solve these issues, we use the map function.

Choice of Models, Third

Predicting and recognizing patterns so as to provide acceptable responses after understanding them is at the heart of machine learning. With the help of ML algorithms, we can analyze and learn from data patterns. When an ML model tries again, it improves based on its previous failures. The success of a model may be judged by splitting the data into a training set and a test set. To train our models, we separated the data into a Training set (consisting of 70% of the total dataset) and a Test set (consisting of the remaining 30%). It was thus critical to use many metrics to evaluate our model's predictions.

Speculate on the end outcome

The effectiveness of a system design is validated by means of a test set. Evolution analysis refers to the process of describing and modeling patterns or trends for things whose behavior varies through time.

4. IMPLEMENTATION

DNN processing consists of two phases: training and prediction. During the training phase, the network is taught to make predictions based on the existing data; these learnt parameters are then utilized during the inference phase to make predictions about the unknown data[14]. The standard DNN training process is shown in Fig. 2. In order to minimize the cost function, a neural network is often trained by accumulating parameters from known examples. The cost function determines the difference between the predicted value from the model and the observed value from the sample. The DNN training phase cannot be completed without forward and backward propagations. During the feed-forward phase, input propagates through the layer to compute the output. Next, the gradient descent technique is used to minimize the gap between the predicted label and the actual label. In the inference phase, when the model makes use of the prediction findings, it just moves the input forward and considers the output as a prediction.

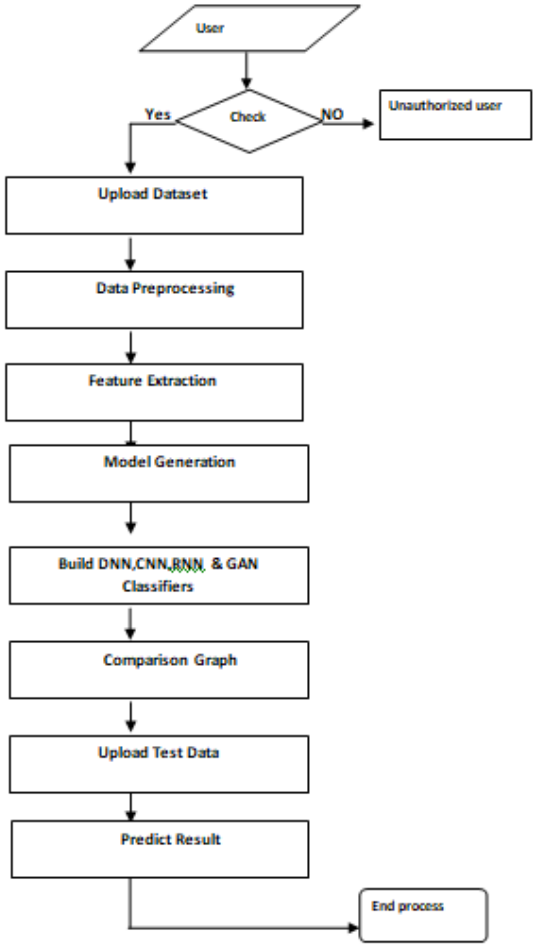

Fig.3: Dataflow diagram

The following are some places where DL has privacy and security issues.

1) Problems with DL models Different stages of DL are vulnerable to different kinds of attacks, the most common of which being evasion attacks during the inference phase and poisoning attempts during the training phase.

To defend DL models, 2) The different defense strategies presented may be broadly classified into two categories: anti-escape and anti-poisoning. Consequently, we may classify the methods used to foil evasion assaults as either empirical (including gradient masking, resilience, and detection) or certified.

Thirdly, DL-based systems are vulnerable to privacy assaults from a variety of sources, including service providers, information silos, and end users.

Fourth, in the event of a privacy breach, the newest cryptographic protection measures, such as differential privacy, safe multiparty computing, and homomorphic encryption, will be put into effect.

5. DEFENSE TECHNIQUES AGAINST DEEP LEARNING MODELS

There are well-established countermeasures to both poison and avoidance attacks. When it comes to the latter, you may choose between certified defenses that have been proven effective in the past and empirical defenses that use past data to counteract known evasion techniques.

Evasion attack countermeasures (1)

Several methods have been proposed to prevent attacks on DL-based systems that attempt to avoid detection (adversarial attacks). When protecting against adversarial instances is crucial, for instance, Kurakin et al. recommended employing adversarial training to increase robustness against evasion efforts. Adversarial training enables us to classify defense tactics into three major categories: gradient masking, robustness, and detection.

The gradients used in attacks may be hidden using a method called gradient masking. The three most common methods utilized in this tactic are shattered gradients, stochastic gradients, and vanishing/exploding gradients.

b) Robustness: Gradient obfuscation may not be the best option in a white-box setting, and it may be better to boost robustness instead. One technique to make a model more secure is to train it to provide the same output from both clean and adversarial instances, and then either penalize the difference between them or regularize the model to reduce the attack surface.

In order to increase the safety of a DL-based system and guarantee that tainted input may be ignored, it is deemed as equally (if not more) important to preserve the capacity to recognize attacks during the inference phase. Most detection methods don't need modifying the classifier, making them easy to implement and suitable for blending with existing defenses.

A lot of defenses' efficacy can only be established experimentally in the context of common attack types, therefore d) Certified approach is essential. Even a highly empirically-supported classifier may be vulnerable to more sophisticated assaults. Nonetheless, it is possible to show that some classifiers, often DNNs, are robust if they reliably provide an output for a collection of input variants.

Second, protecting yourself against poisoning attacks:

Steinhardt et al. [11] provide a defensive strategy for data cleansing that aims to remove contaminated records from the available data. The recommended online learning approach supplies the worst-case test loss from each assault, as well as examples of attacks that may be mounted. Koh et al. employed influence functions to monitor model predictions and identify training data points that had the greatest bearing on a given prediction. Although their theory does not apply to nonconvex and nondifferentiable models, they provided evidence that approximation influence functions may be effective in the face of poisoning attacks. These characteristics allow a defender to zero in on data with a high impact score. Using this method seems to be a more effective technique for removing contaminated instances than just searching for data points with large training losses.

6. CONCLUSION

This article describes the core composition structure and principles of deep learning and then go on to discuss the security challenges that arise in the context of deep learning applications. Also, it proves that there are many hostile samples out there designed to undermine deep learning. By studying adversarial algorithms, we may get a deeper understanding of the underlying mechanisms at work in deep learning's training and prediction processes. This study compiles and analyzes various incidents of deep learning attacks during the last several years, and it also provides a list of defense tactics against countermeasure technologies. Further, specific examples of software failures in certain implementations are provided. In deep learning, predictions are easily swayed by even modest changes, which shows that the underlying framework has serious faults and is hindering the field's progress. For static tasks, such as image categorization, deep learning may provide very precise predictions. However, in dynamic real-time environments with complex interconnections, it is easy to make errors and incorrectly judge emerging

circumstances. In addition, there is a technological impasse related to AI. Therefore, there are far-reaching effects associated with comprehending the security considerations underpinning deep learning architectural methods.

7: WORK TO COME (1) New offensive and defensive methods are constantly being developed using deep learning techniques. Both parts have undergone extensive evolution, with the former beginning with the discovery of deep learning's fragility and the latter culminating in the emergence of multiple safeguards. Both are being developed and improved in tandem using this method.

(2) The availability of adversarial examples allows for deeper learning algorithms to be developed. The large variations in deep learning prediction outputs from even tiny perturbations show that deep learning algorithms need a great deal of time. Recent advancements, although helpful, remain inadequate and immature.

REFERENCES

[1] W. W. Jiang and L. Zhang, Geospatial data to images: A deep-learning framework for traffic forecasting, Tsinghua Science and Technology, vol. 24, no. 1, pp. 52–64, 2019.

[2] L. Zhang, C. B. Xu, Y. H. Gao, Y. Han, X. J. Du, and Z. H. Tian, Improved Dota2 lineup recommendation model based on a bidirectional LSTM, Tsinghua Science and Technology, vol. 25, no. 6, pp. 712–720, 2020.

[3] H. M. Huang, J. H. Lin, L. Y. Wu, B. Fang, Z. K. Wen, and F. C. Sun, Machine learning-based multi-modal information perception for soft robotic hands, Tsinghua Science and Technology, vol. 25, no. 2, pp. 255–269, 2020.

[4] X. Y. Yuan, P. He, Q. L. Zhu, and X. L. Li, Adversarial examples: Attacks and defenses for deep learning, IEEE Trans. Neural Netw. Learn. Syst., vol. 30, no. 9, pp. 2805– 2824, 2019.

[5] J. C. Hu, J. F. Chen, L. Zhang, Y. S. Liu, Q. H. Bao, H. Ackah-Arthur, and C. Zhang, A memory-related vulnerability detection approach based on vulnerability features, Tsinghua Science and Technology, vol. 25, no. 5, pp. 604–613, 2020.

[6] C. Szegedy, W. Zaremba, I. Sutskever I, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, Intriguing properties of neural networks, arXiv preprint arXiv: 1312.6199, 2013.

[7] A. Athalye, N. Carlini, and D. Wagner, Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, arXiv preprint arXiv: 1802.00420, 2018.

[8] Y. T. Xiao, C. M. Pun, and J. Z. Zhou, Generating adversarial perturbation with root mean square gradient, arXiv preprint arXiv: 1901.03706, 2019.

[9] I. J. Goodfellow, J. Shlens, and C. Szegedy, Explaining and harnessing adversarial examples, arXiv preprint arXiv: 1412.6572, 2014.

[10] J. W. Su, D. V. Vargas, and K. Sakurai, One pixel attack for fooling deep neural networks, IEEE Trans. Evol. Comput., vol. 23, no. 5, pp. 828–841, 2019.

[11] W. He, J. Wei, X. Y. Chen, N. Carlini, and D. Song, Adversarial example defense: Ensembles of weak defenses are not strong, in Proc 11th USENIX Workshop on Offensive Technologies, Vancouver, Canada, 2017.

[12] G. W. Xu, H. W. Li, H. Ren, K. Yang, and R. H. Deng, Data security issues in deep learning: Attacks, countermeasures, and opportunities, IEEE Comm. Mag., vol. 57, no. 11, pp. 116–122, 2019.

[13] M. I. Tariq, N. A. Memon, S. Ahmed, S. Tayyaba, M. T. Mushtaq, N. A. Mian, M. Imran, and M. W. Ashraf, A review of deep learning security and privacy defensive techniques, Mobile Inf. Syst., vol. 2020, p. 6535834, 2020.

[14] H. Bae, J. Jang, D. Jung, H. Jang, H. Ha, and S. Yoon, Security and privacy issues in deep learning, arXiv preprint arXiv: 1807.11655, 2018.

[15] S. L. Qiu, Q. H. Liu, S. J. Zhou, and C. J. Wu, Review of artificial intelligence adversarial attack and defense technologies. Appl. Sci., vol. 9, no. 5, p. 909.