POSTER: Towards Understanding the Dynamics of Adversarial Attacks

Yujie Ji
Lehigh University
Bethlehem, PA, USA
yuj216@cse.lehigh.edu

Ting Wang*
Lehigh University
Bethlehem, PA, USA
ting@cse.lehigh.edu

ABSTRACT

An intriguing property of deep neural networks (DNNs) is their inherent vulnerability to adversarial inputs, which significantly hinder the application of DNNs in security-critical domains. Despite the plethora of work on adversarial attacks and defenses, many important questions regarding the inference behaviors of adversarial inputs remain mysterious. This work represents a solid step towards answering those questions by investigating the information flows of normal and adversarial inputs within various DNN models and conducting in-depth comparative analysis of their discriminative patterns. Our work points to several promising directions for designing more effective defense mechanisms.

CCS CONCEPTS

• **Security and privacy** → **Domain-specific security and privacy architectures**; • **Computing methodologies** → *Neural networks*; • **Mathematics of computing** → Information theory;

KEYWORDS

adversarial sample; deep neural network; mutual information

ACM Reference Format:
Yujie Ji and Ting Wang. 2018. POSTER: Towards Understanding the Dynamics of Adversarial Attacks. In *2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18), October 15–19, 2018, Toronto, ON, Canada.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3243734.3278528

1 INTRODUCTION

Recent years have witnessed the abrupt advances in deep learning [9], leading to breakthroughs in a number of long-standing artificial intelligence tasks. However, designed to model highly nonlinear, non-convex functions, deep neural networks (DNNs) are inherently vulnerable to adversarial inputs, which are maliciously crafted samples to trigger target DNNs f to misbehave [18], such as for a given benign input x, the attacker attempts to find the minimum perturbation r forcing f's misclassification of $\hat{x} = x + r$, i.e., $\min_r f(x) \neq f(x + r)$. With the increasing use of DNN-powered systems in security-critical domains, adversaries have strong incentives to manipulate such systems via adversarial inputs.

*Contact Author

CCS '18, October 15–19, 2018, Toronto, ON, Canada

ACM ISBN 978-1-4503-5693-0/18/10.
https://doi.org/10.1145/3243734.3278528

The phenomena of adversarial inputs have attracted intensive research from the security communities. Despite the plethora of existing work, we still lack sufficient understanding of the crucial properties of adversarial inputs. A number of important questions remain mysterious, such as: (i) How are adversarial inputs crafted to force DNNs to misclassify? (ii) How are adversarial inputs generated by various attack models different in their underlying mechanisms? (iii) How are existing defenses often vulnerable to adaptive attacks? (iv) How are complicated DNNs more vulnerable to adversarial input attacks than simple DNNs? (v) How are transferable adversarial inputs different from non-transferable ones?

This work represents a solid step towards answering those key questions. We take a route completely different from existing work: instead of focusing on the static properties of adversarial inputs from an input-centric perspective (i.e., whether a given adversarial input can mislead the target DNN), we study the dynamic properties of adversarial inputs from a DNN-centric perspective (i.e., how the target DNN reacts to the given adversarial input).

2 INFORMATION FLOW MODEL

To understand the dynamic properties of adversarial inputs, we measure their information flows within various DNNs and conduct in-depth comparative studies of their patterns.

2.1 Mutual Information

Consider a DNN f comprising a sequence of K layers, where the output of k-th layer consists of n_k feature maps $\{m_i^{(k)}\}_{i=1}^{n_k}$. Let x be a given input to f. As x is of multiple channels (e.g., RGB), we also consider x as a set of n_s feature maps $\{m_i^s\}_{i=1}^{n_s}$. To understand x's dynamic properties, i.e., how f reacts to x, we quantify x's information flow going through f, via measuring the mutual information (MI) between each feature map and x.

Specifically, we treat each feature map m as a discrete distribution: Let $v_{\min}$ and $v_{\max}$ respectively be the minimum and maximum values in m. We divide the interval $[v_{\min}, v_{\max}]$ evenly into B buckets and replace each value v in m with its bucket ID: $\text{bid}(v) = \lceil B(v - v_{\min})/(v_{\max} - v_{\min}) \rceil$. We then populate an $n_k \times n_s$ matrix $S^{(k)}$ with the i, j-th element $S_{ij}^{(k)}$ being the MI of $m_i^{(k)}$ and m_j^s. Moreover, to obtain a complete view of x's information flows, we also measure the MI of each feature map at an intermediate layer and the output of f's last conv layer (which consists of n_t feature maps $\{m_{(k)}^t\}_{k=1}^{n_t}$). We populate an $n_k \times n_t$ matrix $T^{(k)}$, with its i, j-th element $T_{ij}^{(k)}$ being the MI of $m_i^{(k)}$ and m_j^t. We refer to $S^{(k)}$ and $T^{(k)}$ as the source and target MI matrices of the k-th layer.

2.2 Information Paths

Armed with $S^{(k)}$ and $T^{(k)}$, we depict the "information paths" (IPs) from a given input x to its output y in a layer-wise manner, i.e., how the feature maps at each layer capture the information in x and transform it towards y. Specifically, we construct a set of IPs:

- *Input information path* (IIP) quantifies the relevance of the feature maps at each layer with the input, defined as the sequence of $\{(k, \mu_s^{(k)})\}_k$, where $\mu_s^{(k)}$ is the mean of $S^{(k)}$.
- *Output information path* (OIP) quantifies the relevance of the feature maps at each layer with the output, defined as the sequence of $\{(k, \mu_t^{(k)})\}_k$, where $\mu_t^{(k)}$ is the mean of $T^{(k)}$.
- *Input-Output contrast* (IOC) correlates the input and output information paths at each layer, defined as a sequence of $\{(\mu_s^{(k)}, \mu_t^{(k)})\}_k$.

Note that our approach is inspired by the theory of information bottleneck methods [17, 19]. However, different from existing work, which treats inputs as individual data points, we consider each input as a discrete distribution, which allows us to investigate the information flow at the level of individual inputs.

2.3 Aggregated Information Paths

Further, we devise the model of aggregated IPs to summarize the IPs of a set of similar inputs (e.g., adversarial inputs generated by the same attack). We consider each MI measure (e.g., $\mu_s^{(k)}$) as a random variable and assume that a collection of such random variables indexed by k follow a multivariate normal distribution; each IP is thus a random sample from a Gaussian process [16]. We use the mean of the Gaussian process to represent their aggregated IP.

3 ANALYSIS

Equipped with the aforementioned measurement tools, we conduct an empirical study on the dynamic properties of adversarial inputs. Our study is designed to provide a new perspective on a set of key questions about adversarial inputs.

3.1 Experimental Setting

We mainly use the Cifar10 dataset [7] and the DNN model in [12] which attains the accuracy of 90.24% on Cifar10.

We focus on untargeted attacks. For targeted attack models (e.g., Jsma), we select the class $\hat{y}$ (different from $\boldsymbol{x}$'s ground-truth class y) that requires the minimum perturbation as its targeted class. Meanwhile, we require different attack models to have similar perturbation magnitude for fair comparison.

In our study, given a DNN model f and an input x, we collect x's feature maps, measure its source and target MI matrices $\{S^{(k)}, T^{(k)}\}_k$, and compute x's IPs (IIP, OIP, and IOC) within f.

3.2 Experimental Results

We present our empirical study results and report our findings.

Q1: How are adversarial inputs crafted to trigger DNNs to misbehave? Figure 1 shows the process of Jsma attack on a randomly selected input, by visualizing each intermediate adversarial input x_i's information flows. We observe that the information flows of adversarial inputs deviate from that of benign ones, while the attack process essentially corresponds to shifting the information flows away from benign inputs towards adversarial ones.

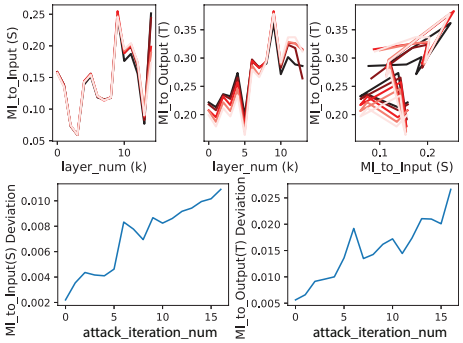

Figure 1: Top: IPs of a randomly selected input. The black line represents the benign input $\hat{x}_0$; lighter colors indicate larger i; the line of the lightest color represents the adversarial input $\hat{x}_n$. Bottom: the overall difference between IPs of $\hat{x}_i$ and $\hat{x}_0$.

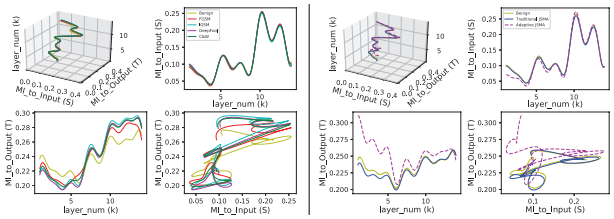

Figure 2: Aggregated IPs of benign and adversarial inputs generated by (left) Fgsm, Igsm, DeepFool, and C&W attacks (right) by regular and adaptive Jsma attacks.

Q2: How are adversarial inputs generated by various attacks different in their underlying mechanisms? Figure 2 (left) compares the aggregated IIPs, OIPs, and IOCs of benign inputs and adversarial inputs generated by different attacks including Fgsm [5], Igsm [8], DeepFool [13], and C&W [4], where adversarial inputs generated by varied attack models lead to drastically different information flows, implying that multiple defense or detection methods might be necessary to mitigate different attacks.

Q3: How are existing defense mechanisms often vulnerable to adaptive attacks? We compute the aggregated IPs of adversarial inputs (including successful and failed ones) generated by regular and adaptive Jsma [2] on the defensively distilled DNN model [15]. Figure 2 (right) shows that from the IP perspective, adversarial inputs generated by adaptive Jsma deviate much further from benign inputs than those by regular Jsma, which explains why defensive distillation fails to defend against adaptive attacks.

Q4: How are complicated DNN models more vulnerable to adversarial inputs? Figure 3 shows the aggregated IPs of benign inputs and adversarial inputs generated by Jsma. For comparison, with respect to a given DNN f (original or compressed by [6]), besides the IPs of adversarial inputs targeting f, we also compute the IPs of adversarial inputs targeting the other DNN (compressed or original). We observe that it is easier to cause information flows to shift from benign inputs on more complicated DNN, indicating that adversarial inputs demonstrate higher transferability on the original model (i.e., more complicated models tend to be more vulnerable to adversarial inputs).

Q5: How are transferable adversarial inputs different from non-transferable ones? Figure 4 shows the aggregated IPs of adversarial inputs targeting three DNNs, "good_init" [12], "deep_conv",

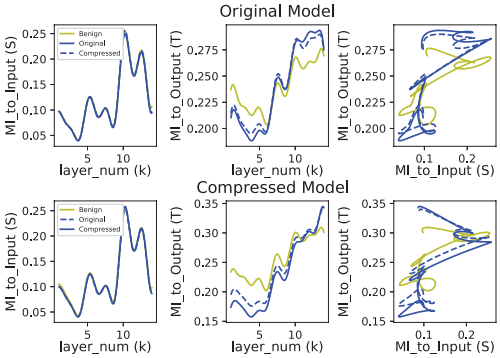

Figure 3: Aggregated IPs of benign and adversarial inputs by JSMA. The solid and dashed blue lines respectively represent adversarial inputs targeting the inference model and the other model.

and "vgg_like", by JSMA. Observe that the OIPs of adversarial inputs targeting the inference model deviate further away from benign inputs, compared with that of adversarial inputs targeting a model different from this inference model. Moreover, the OIPs of transferable adversarial inputs [18] deviate much further compared with non-transferable ones. Therefore, in order to create transferable adversarial inputs, it is sensible to attack ensemble models [10] (i.e., training on the three DNNs simultaneously).

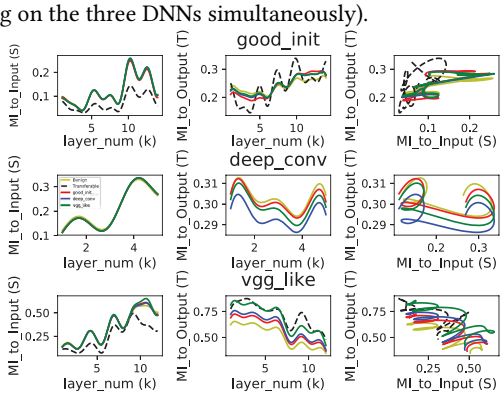

Figure 4: Aggregated IPs of adversarial inputs targeting three DNN models by JSMA. Note that JSMA fails to generate transferable adversarial samples targeting "deep_conv".

4 ADDITIONAL RELATED WORK

Adversarial Deep Learning. The phenomena of adversarial inputs have attracted intensive research. One line of work focuses on developing new attacks [2, 4, 5, 8, 13, 14]. Another line of work attempts to defend against such attacks [5, 11, 15, 18]. However, the defense-enhanced models, once deployed, can often be fooled by adaptively engineered inputs or by new attack variants [1–3].

Information Flow Theory. Recently the information flow theory has been used to study the underlying mechanisms of DNN models. Tishby and Zaslavsky [20] suggested the use of Information Bottleneck (IB) [19] to study the representation learning process. Shwartz-Ziv and Tishby [17] then applied the IB theory to evaluate the DNN training process. Different from the previous studies, this work focuses on measuring and comparing the information flows caused by benign and adversarial inputs, and extracting their discriminative patterns.

5 CONCLUSION

In this paper, we present an empirical study on the dynamic properties of adversarial input attacks against DNN models. Using a data-driven approach, we measure the information flows of adversarial inputs within various DNN models and conduct the in-depth comparative study on their discriminative patterns. Our study sheds light on a set of key questions surrounding adversarial inputs, points to several promising directions for designing more effective defense mechanisms. We hope that our visualization tool can help researchers learn more about adversarial samples behavior during DNN model classification.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 1566526 and 1718787.

REFERENCES

[1] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
[2] N. Carlini and D. Wagner. 2016. Defensive distillation is not robust to adversarial examples. *ArXiv e-prints* (2016).
[3] Nicholas Carlini and David Wagner. 2017. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of ACM Workshop on Artificial Intelligence and Security (AISec)*.
[4] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *Proceedings of IEEE Symposium on Security and Privacy (S&P)*.
[5] I. J. Goodfellow, J. Shlens, and C. Szegedy. 2015. Explaining and harnessing adversarial examples. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
[6] Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. 2016. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. *ArXiv e-prints* (2016).
[7] Alex Krizhevsky and Geoffrey Hinton. 2009. Learning Multiple Layers of Features from Tiny Images. *Technical report, University of Toronto* (2009).
[8] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533* (2016).
[9] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep Learning. *Nature* 521, 7553 (2015), 436–444.
[10] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. 2016. Delving into transferable adversarial examples and black-box attacks. *ArXiv e-prints* (2016).
[11] Dongyu Meng and Hao Chen. 2017. Magnet: a two-pronged defense against adversarial examples. In *Proceedings of ACM SAC Conference on Computer and Communications (CCS)*.
[12] Dmytro Mishkin and Jiri Matas. 2015. All you need is a good init. *ArXiv e-prints* (2015).
[13] Seyed Mohsen Moosavi Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[14] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swamil. 2016. The limitations of deep Learning in adversarial settings. In *Proceedings of IEEE European Symposium on Security and Privacy (Euro S&P)*.
[15] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. 2016. Distillation as a defense to adversarial perturbations against deep neural networks. In *Proceedings of IEEE Symposium on Security and Privacy (S&P)*.
[16] Carl Edward Rasmussen. 2004. Gaussian processes in machine learning. In *Advanced lectures on machine learning*. 63–71.
[17] R. Shwartz-Ziv and N. Tishby. 2017. Opening the black box of deep neural networks via information. *ArXiv e-prints* (2017).
[18] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. 2014. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
[19] Naftali Tishby, Fernando C. Pereira, and William Bialek. 1999. The information bottleneck method. In *Proceedings of Annual Allerton Conference on Communication, Control and Computing (Allerton)*.
[20] Naftali Tishby and Noga Zaslavsky. 2015. Deep learning and the information bottleneck principle. In *Proceedings of IEEE Information Theory Workshop (ITW)*.