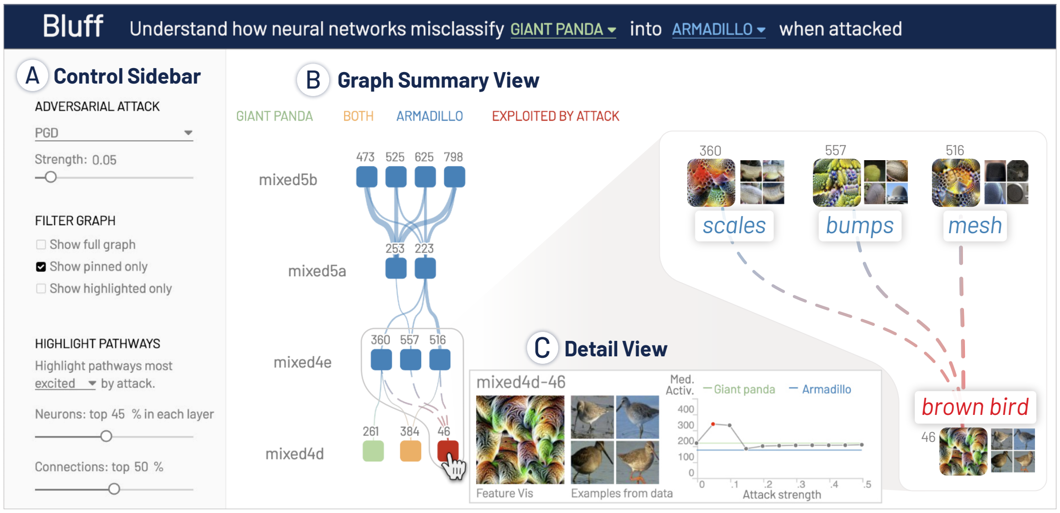
2020 IEEE Visualization Conference (VIS) | 978-1-7281-8014-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/VIS47514.2020.00061

Bluff: Interactively Deciphering Adversarial Attacks on Deep Neural Networks

Nilaksh Das *†, Haekyu Park*†, Zijie J. Wang †, Fred Hohman†,
Robert Firstman†, Emily Rogers ‡, Duen Horng (Polo) Chau†



Bluff Understand how neural networks misclassify GIANT PANDA into ARMADILLO when attacked

A Control Sidebar

ADVERSARIAL ATTACK

PGD

Strength: 0.05

FILTER GRAPH

□ Show full graph

☑ Show pinned only

□ Show highlighted only

HIGHLIGHT PATHWAYS

Highlight pathways most excited by attack.

Neurons: top 45 % in each layer

Connections: top 50 %

Figure 1: With BLUFF, users interactively visualize how adversarial attacks penetrate a deep neural network to induce incorrect outcomes. Here, a user inspects why INCEPTIONV1 misclassifies adversarial **giant panda** images, crafted by the *Projected Gradient Descent* (PGD) attack, as **armadillo**. PGD successfully perturbed pixels to induce the "*brown bird*" feature, an appearance more likely shared by an armadillo (small, roundish, brown body) than a panda, activating more features that contribute to the armadillo (mis)classification (e.g., "*scales*," "*bumps*," "*mesh*"). The *adversarial* pathways, formed by these neurons and their connections, overwhelm the benign panda pathways and lead to the ultimate misclassification. **(A) Control Sidebar** allows users to specify what data is to be included and highlighted. **(B) Graph Summary View** visualizes pathways most activated or changed by an attack as a network graph of neurons (each labeled by the channel ID in its layer) and their connections. When hovering over a neuron, **(C) Detail View** displays its feature visualization, representative dataset examples, and activation patterns over attack strengths.

Abstract

Deep neural networks (DNNs) are now commonly used in many domains. However, they are vulnerable to *adversarial attacks*: carefully-crafted perturbations on data inputs that can fool a model into making incorrect predictions. Despite significant research on developing DNN attack and defense techniques, people still lack an understanding of how such attacks penetrate a model's internals. We present BLUFF, an interactive system for visualizing, characterizing, and deciphering adversarial attacks on vision-based neural networks. BLUFF allows people to flexibly visualize and compare the activation pathways for benign and attacked images, revealing mechanisms that adversarial attacks employ to inflict harm on a model. BLUFF is open-sourced and runs in modern web browsers.

Index Terms: Human-centered computing—Visual Analytics

1 Introduction

Deep neural networks (DNNs) are a major driving force behind many recent technological breakthroughs [11, 14–16, 28, 41], but they are highly vulnerable to *adversarial attacks*. Small, human-imperceptible noise injected into inputs can easily fool DNNs into making wrong predictions [7, 13, 22, 34], raising alarms for safety-critical applications, such as autonomous driving and data-driven healthcare. Thus, it is essential to understand how attacks harm DNN models [35, 38]. But interpreting and ultimately defending against adversarial attacks remain fundamental research challenges. DNNs are often considered "unintelligible" due to their complex architectures and huge number of parameters. It is difficult to pinpoint the parts of the model exploited by an attack, let alone to understand how such exploitation leads to incorrect outcomes [23]. Also, there is a lack of research in understanding how an attack's "strength" may correlate with neurons' activation patterns [24]. For example, it is not yet known if a stronger attack exploits the same neurons as a weaker attack does, or if these sets are completely different.

To address the above challenges, we develop BLUFF (Fig. 1), an interactive visualization tool for discovering and interpreting how adversarial attacks mislead DNNs into making incorrect decisions. Our main idea is to visualize **activation pathways** within a DNN traversed by the signals of benign and adversarial inputs. An *activation pathway* consists of neurons (also called *channels* or *features*) that are highly activated or changed by the input, and the connections among the neurons. BLUFF finds and visualizes where a model is exploited by an attack, and what impact the exploitation has on the final prediction, across multiple attack strengths. We contribute:

- **BLUFF, an interactive system for summarizing and interpreting** how adversarial perturbations penetrate DNNs to in-

*Authors contributed equally.
†Georgia Institute of Technology.
{nilakshdas, haekyu, jayw, fredhohman, rfirstman6, polo}@gatech.edu
‡Georgia Tech Research Institute. Emily.Rogers@gtri.gatech.edu

978-1-7281-8014-4/20/$31.00 ©2020 IEEE
DOI 10.1109/VIS47514.2020.00061

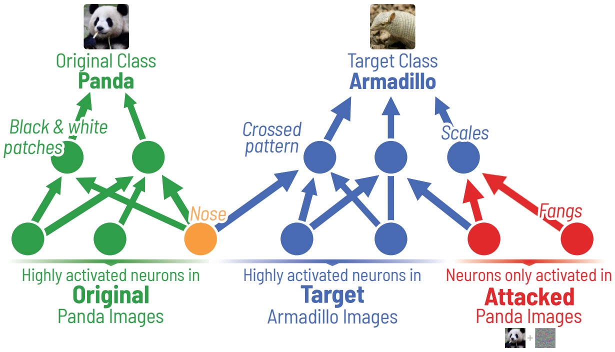


Figure 2: Adversarial attacks confuse DNNs to make incorrect predictions (e.g., misclassify benign *panda* as *armadillo*). BLUFF helps discover where such attacks occur and what features are used.

duce incorrect outcomes in INCEPTIONV1 [37], a large-scale prevalent image-classification model, over images from ImageNet ILSVRC 2012 [36]. To support reproducible research and broaden its access, we have open-sourced BLUFF at https://poloclub.github.io/bluff.

- **Visual characterization of activation pathway dynamics.** Adversarial perturbations manipulate activation pathways typically used for benign inputs to induce incorrect predictions. For example, an attack can *inhibit* neurons detecting important features for the benign class and *excite* those that exacerbate misclassification. BLUFF visualizes and highlights activation pathways exploited by an attack (Fig. 2) and shows how they mutate and propagate through a network.
- **Interactive comparison of attack escalation.** BLUFF enables interactive comparison of activation pathways under increasing attack strengths, providing a new way for understanding the essence of an attack (e.g., common trends of an attack across all strengths) and its multi-faceted characteristics (e.g., various strategies that different strengths may employ).
- **Discovery usage scenarios.** We describe how BLUFF can help discover surprising insights into the vulnerability of DNNs, such as how unusual activation pathways may be exploited by attacks.

2 Related Work

Adversarial Attacks on DNNs. Adversarial attacks aim to confuse a DNN model into making incorrect predictions by adding carefully crafted perturbations to the input [5,13,25]. We focus on *targeted* adversarial attacks, where a model is misled to make a prediction of the attacker's choosing, which can pose severe threats to practical deep learning applications [7]. Given a benign input instance x, a targeted adversarial attack aims to find a small *perturbation* δ that changes the prediction of model $\mathcal{M}$ to a target class t different from the true class y, i.e., $\mathcal{M}(x+\delta)=t$, where $t \neq y, ||\delta|| \leq \varepsilon$. We call ε the *attack strength*. Projected Gradient Descent (PGD) [24] is one of the *strongest* first-order targeted attacks [40]. Hence, we examine the l_2 norm of PGD with varying the strength ε from 0.0 (no attack) to 0.5 (strong attack).

Neural Network Interpretability. Deep neural networks have often been described as "black boxes" due to their complex internal structures. An approach to understand how neural networks work internally is to study neurons' activation patterns. To interpret what concept a neuron is detecting, *feature visualization* [6,10,27,31] creates visualization that maximizes such neuron. TCAV [21], Network Dissection [2], and Net2Vec [12] propose to quantify interpretability by measuring alignment between the neuron activations and concept features. Circuits [30] and Summit [18] visually explain how higher-level concepts can be constructed by neural connections. Activation Atlas [6] visualizes neuron activations per layer and analyzes how models can be exploited when predicting on manipulated inputs. On top of using the neurons' activations, we visualize important connections among the neurons contributing to such misclassifications, and how these connections react to attacks.

Interpretability for Adversarial Attacks. While research on machine learning security has attracted great attention [7,9,13,24,29,39], research for interpreting adversarial attacks on DNNs is nascent. AEVis [4,23] proposes to extract critical neurons and their connections for benign and adversarial inputs, and demonstrates the method on small sets of images. However, it is unclear how it may scale to larger datasets that BLUFF operates on (e.g., 900+ adversarial images for a single ImageNet class [36]). BLUFF also provides new techniques for comparing activation pathways, enabling novel analysis (e.g., based on neuron inhibition and excitation) and discoveries (e.g., how different attacks may have different strategies).

3 Bluff: Deciphering Adversarial Attacks

3.1 Design Goals

Through a literature survey, we have identified the following four design goals (**G1**-**G4**) that guide BLUFF's development.

G1 Untangling activation pathways. *Benign* activation pathways can significantly overlap with *adversarial* pathways, as some neurons are "*polysemantic*," detecting multiple concepts at the same time [2, 30]. We aim to identify neurons that respond differently between benign and attacked inputs, to help discover where and how a model is exploited by an attack to induce incorrect predictions.

G2 Interpreting multiple activation pathways. Understanding the effects of adversarial attacks is core to developing robust defenses [13,24,39]. We aim to visualize high-level overviews of *benign* and *adversarial* activation pathways, and support drilling-down into subpaths, to help shed light on how specific groups of neurons are exploited to inflict harm on a model.

G3 Comparing attack characteristics. Existing works to interpret adversarial attacks on deep neural networks often focus on visualizing the activation patterns for a single adversarial input [6,29]. We aim to visualize how the activation pathway changes as the attack strength varies, to help users gain deeper insight into how the attack works generally. Understanding model vulnerability under different attack strategies informs more robust defenses [9,25,33].

G4 Lowering barrier of entry for interpreting and deciphering adversarial attacks. The visualization community is contributing a variety of methods and tools to help people more easily interpret different kinds of DNNs [2,6,17,18,20,23,31]. Efforts that aim to support deciphering adversarial attacks, however, are relatively nascent [2,23,29]. We aim to make interpreting adversarial attacks more accessible to everyone, following the footsteps of prior success from the community.

3.2 Background: Neuron Importance and Influence

To discover activation pathways triggered by benign and adversarial inputs, BLUFF finds important neurons and influential connections among such neurons. Inspired by [18], BLUFF computes a neuron's *importance* based on how strongly it is activated by all inputs, and the *influence* between neurons based on the amount of activation signals transmitted through the connection to the next layer. While summarizing interpretable pathways within a DNN remains an open problem [3,19,26], recent works [18,30,31] have shown that dominant neuron activations at each layer form the basis vectors for the entire activation space of the DNN. Thus, characterizing *important* neurons at each layer based on neuron activation provides a surrogate sampling of important neurons across the whole network, for a given set of images. BLUFF extends this notion to scalably aggregate the activation pathways across *multiple* contexts with the most important neurons for: (1) benign images belonging to the **original** class, on which the targeted attacks are performed; (2) benign images belonging to the **target** class, which the attacks try to flip the label to; and (3) successfully **attacked** images for a particular attack strength (we support exploration with multiple strengths).

To begin, we consider the DNN model $\mathscr{M}$ (INCEPTIONV1), where $Z^q \in \mathbb{R}^{H_q,W_q,D_q}$ is the output tensor of the q'th layer of $\mathscr{M}$. Here, H_q, W_q and D_q are the height, width and depth dimensions respectively. This implies that the layer has D_q neurons. We denote the d'th output channel (for d'th neuron) in the layer as $\mathscr{C}_q^d \in \mathbb{R}^{H_q,W_q}$. We index the values in the channel as $\mathscr{C}_q^d[h,w]$. Given an input image x_i, we find the maximum activation of each neuron induced by the image using the global max-pooling operation: $a_q^d[i] = \max_{h,w} \mathscr{C}_q^d[h,w]$. This represents the magnitude by which the d'th neuron in the q'th layer maximally detects the corresponding semantic feature from image x_i. This technique of extracting maximal activation as a proxy for semantic features has also demonstrated tremendous predictive power in the data programming domain [8]. Finally, we pass all images from each of **original**, **target** and **attacked** datasets. For each set, we aggregate $a_q^d[i]$ values for all images and quantify the importance of each neuron by the median value of such maximal activations. We use medians for summarizing the neuron importance, because they are less sensitive to extreme values. Consistent with the findings of [18], we observe that the maximal activation values are power law distributed, implying that only a small minority of neurons have highest importance scores. Hence to denoise inconsequential visual elements, for each layer, we empirically filter the 10 most important neurons for benign images of original and target classes, and 5 most important ones for attacked images of each attack strength (i.e., at most 50 important neurons across all 10 attack strengths).

To compute influence of a connection between two neurons, we measure the signal transmitted through the connection, computed by the convolution of the slice of the kernel tensor between the two neurons over the source neuron's channel activation. Since output from the ReLU activation function is used as the neuron activation in an INCEPTIONV1 model, it implies that the neuron importance scores that are propagated are non-negative. Consequently, the model acts on these non-negative activation values. Hence, these activation values accumulate only for positively weighted convolution operations through consecutive layers, which has the effect of filtering out non-influential connections that may even originate from important neurons. Inspired by [18], we take the maximum value in the convolution for the influence of the connection. BLUFF deviates from [18] when aggregating influence values across several images. We characterize the connection by taking the median influence across all images from a given set. We take this approach since we want to summarize the influence characteristics across multiple datasets (**original**, **target** and **attacked** for different attack strengths), and each dataset is of a different size. Simple counting may skew the results towards a particular dataset while the median value provides a characteristic aggregation of the influence scores.

3.3 Realizing Design Goals in BLUFF's Interface

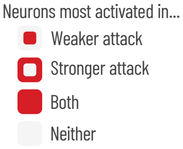
BLUFF's interface (Fig. 1) consists of: **A. Control Sidebar** for selecting which data are included, filtered, highlighted, and compared; **B. Graph Summary View** that summarizes and visualizes *activation pathways* as a graph; **C. Detail View** for interpreting the concept that a neuron has detected, via feature visualization, representative dataset examples, and activation patterns over attack strengths. In the header (Fig. 1, top), users can select a pair of **original** and **target** class. BLUFF then generates the main visualization in the Graph Summary View for how neural networks misclassify **original** images as **target** images when attacked, by displaying the activation pathways of adversarial inputs.

Unifying Multiple Graph Summaries. BLUFF summarizes InceptionV1's responses to inputs under *multiple* contexts in a unified view. Specifically, the Graph Summary View (Fig. 1B) visualizes the top neurons important only for the *original* class as **green** nodes, those important only for the *target* class as **blue** nodes, those important to both classes as **orange** nodes, and those important only for successfully-attacked images as **red** nodes. Furthermore, those neurons are spatially grouped based on the four roles. This generic design, that can be extended to any DNN model with intermediate convolutional blocks, helps users more easily pinpoint and tease out the subtle ways that neurons participate in an attack (**G1**, **G2**), e.g., red neurons, by the very definition, are exploited only by the attack but are neither activated by the original or target images. Our key design decision here is to unambiguously differentiate the four neuron contexts using spatial positioning; we supplement this differentiation by further encoding the four contexts with distinct colors.

The Graph Summary View (Fig. 1B) focuses on visualizing the model's 9 mixed layers (mixed3a, mixed3b ... mixed5b), following existing interpretability literature [18, 31, 32]. The topmost row corresponds to the last mixed network layer (i.e., mixed5b). Each connection between two neurons is visualized as a curved line, whose width scales linearly with the influence values computed as in Sect. 3.2.

Visualizing exploited activation pathways. An adversarial input is often a slightly perturbed version of a benign input, which means the activation pathways of an benign image and those of its adversarial counterpart would be similar at the input layer [23], yet decidedly different at the output layer — the *benign* pathways lead to the **original** prediction, while the *adversarial* pathways lead to the **target** prediction. Given the similar starting points but different outcomes, the adversarial activation pathways must have deviated from the benign pathways. BLUFF helps discover vulnerable neurons and connections that contribute to such deviations and the resulting misclassification, by highlighting the neurons and connections that are *excited* (or *inhibited*, oppositely) the most by an attack (Fig. 1A) (**G1**). A pathway *excited* by an attack means its constituent neurons are activated more than expected (i.e., pathway contains more target features). Fig. 1 shows an example of where the attack *excites* multiple features and connections to induce the target prediction of armadillo (e.g., "*scales*," "*bumps*," "*mesh*," and "*brown bird*" thanks to its similarity to armadillos' roundish, brown body). Computationally, in layer q, a neuron d's excitation amount is $\tilde{a}_d^q[attacked] - \tilde{a}_d^q[benign]$, where $\tilde{a}_d^q[benign]$ and $\tilde{a}_d^q[attacked]$ are the neuron's importance for some benign and attacked images respectively (as described in Sect. 3.2).

Interpreting Activation Pathways. To help users more easily interpret the concepts that a neuron is detecting, alongside each neuron, BLUFF shows (1) a *feature visualization*, an algorithmically generated image that maximizes the neuron's activation, and (2) *dataset examples*, cropped from real images in the dataset, that also highly activate the neuron [31]. Hovering on a neuron shows the corresponding feature visualization and dataset examples as seen in Fig. 1C, where adversarial images successfully induce the "*brown bird*" feature, an appearance more likely shared by an armadillo (small, roundish, brown body) than a panda, which in turn activates more features in subsequent layers that contribute to the (mis)classification of armadillo. These visual explanations help translate abstract activation pathways into the composition and flow of learned concepts (**G2**).



Comparing attacks with varying strengths. BLUFF offers the *Compare Attacks* mode that visualizes and compares the pathway differences between a weaker attack and a stronger attack (**G3**). BLUFF visually encodes the neurons based on which attack strengths they responded to, drawing inspiration from Alper et al [1]. Each neuron consists of an inner and an outer rectangle: the *inner* rectangle is

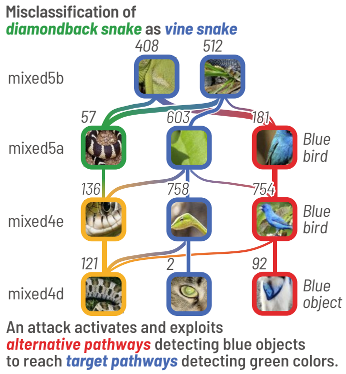

Misclassification of
diamondback snake as vine snake
408
512
mixed5b
57
603
181
mixed5a
Blue bird
136
758
754
mixed4e
Blue bird
121
2
92
mixed4d
Blue object
An attack activates and exploits
alternative pathways detecting blue objects
to reach target pathways detecting green colors.

Figure 3: Bluff helps users understand how an attack penetrates a model, by visualizing activation pathways that are additionally exploited by the attack. In this example, Bluff highlights the neurons and connections that PGD attack exploits (red) to make the model confuse adversarial **diamondback snake** images as **vine snake**.

colored when the neuron is in the activation pathways of the *weaker* attack; whereas the *outer* rectangle is colored when the neuron is in the activation pathways of *stronger* attack. Thus, our design can visually encode all four possible comparison results in relative terms, enabling us to use hue to encode the four neuron contexts. In other words, the comparison mode's visual encoding gracefully builds on and preserves Bluff's overall visual design. Our terminology for *weaker* and *stronger* attacks are relative, as we do not assert any explicit threshold for weak or strong attack strengths.

Cross-platform deployment with standard web technologies. To support reproducible research and broaden its access, Bluff uses standard web technologies (HTML/CSS/JavsScript stacks, and D3.js) and can be accessed from any modern web browser (**G4**) at `https://poloclub.github.io/bluff`. We ran all the backend code that computes neurons' importance and connections' influence on a NVIDIA DGX-1 workstation equipped with 8 GPUs (each with 32GB memory), 80 CPU cores, and 504GB RAM.

4 Discovery Usage Scenarios

We now demonstrate how Bluff enhance the understanding of adversarial attacks and reveal attack strategies that confuse a DNN. For our scenarios, we pick from the 1000 classes of the ImageNet dataset [36], which consists of ∼1.2 million images.

4.1 Understanding How Attacks Penetrate DNNs

Consider a DNN classifier that labels snakes, such as the deadly venomous *diamondback snake*, and the green *vine snake* whose venom causes only mild swelling. In Fig. 3, Bluff's Graph Summary View reveals how adversarial diamondback images *exploit* (highly activate) unexpected pathways to induce the incorrect vine snakes prediction, leveraging multiple exploited neurons that look for *"blue color"* (e.g., *"blue birds"* in Fig. 3, right column). This is surprising because *vine snakes* (the attack's target class) have a green body, not blue. This finding suggests that PGD exploits the pathway for *"blue color"* as a bypassing alternative route to reach the pathways for vine snake, which look for *"green leaves"* and *"green bumps"* (Fig. 3, middle column). We also noticed that PGD leverages *"snake-like"* pathways that are important for both classes (Fig. 3, left column), which is reasonable given that both the original and target classes are snakes. Finding the pathways exploited by an attack provides fundamental insights that could inform future defenses, such as blocking the alternative routes. Fig. 1 shows another example, where the *adversarial armadillo* pathways overwhelm the *benign panda* pathways and ultimately lead to the misclassification.

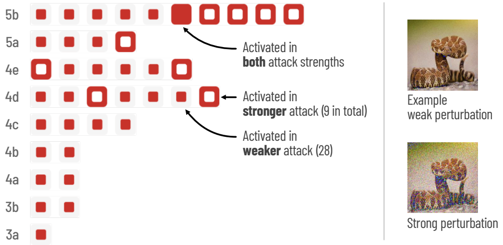
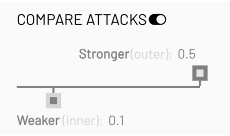


Figure 4: Using Bluff's "Compare Attacks" mode, we examine PGD's different strategies for misclassifying *diamondback* images into *vine snake* class for two attack strengths (0.1 vs 0.5). The weaker attack exploits more alternative neurons (i.e., features that are not typically activated by benign inputs) than the stronger attack does.

4.2 Startling Tactic: Death by a Thousand Cuts


COMPARE ATTACKS
Stronger(outer): 0.5
Weaker(inner): 0.1

An adversary can perform an attack on the model at various levels of attack strengths — starting from imperceptible noise, all the way up to high intensity perturbation (see example perturbations in Fig. 4). Does an attack's strategy evolves as the attack strength escalates, or remains the same? Bluff enables such a comparative analysis through its *Compare Attacks* mode. Consider the example of attacking the *diamondback* images to induce the misclassification of *vine snake*. Setting the weaker and stronger attack strength to 0.1 and 0.5 respectively and looking at 30% of most activated neurons in each layer, Fig. 4 reveals the surprising finding that the weaker attack exploits a large number of red neurons (28 in total) — neurons that are *not* important to either snake class, but are highly activated by the attack — many more than the stronger attack (only 9 in total). On further examining the example image patches for these neurons, we noticed the images consist an assortment of semantic features such as *spider legs*, *blue bird* and *car hood*, seemingly unrelated to snakes. We observed similar attack tactics in other class pairs. For *ambulance* images misclassified as *street sign*, 42 red neurons are exploited by the weaker attack, and only 16 by the stronger attack. For *panda* images misclassified as *armadillo*, 29 exploited by the weaker attack, only 8 by the stronger attack. These observations lead us to conclude that weaker attacks rely on leveraging a large number of disassociated semantic features to induce misclassification, i.e., "death by a thousand cuts".

5 Discussion and Future Work

We present Bluff, an interactive system for visualizing, characterizing, and deciphering adversarial attacks on DNNs. We believe our visualization, summarization, and comparison approaches will help promote user understanding of adversarial attacks, and support discoveries to design a proper defense. Our next step is to use Bluff to help construct robust defenses against attacks. We plan to extend Bluff to support *interactive neuron editing* (e.g., "deleting" a neuron from model), so that the user may empirically identify and act on vulnerable neurons and observe the effects on the resulting pathway and prediction in real-time. We also plan to extend Bluff to work for adversarially-trained models [13, 24, 39], to help people gain deeper insights that explain their robustness. Additionally, after receiving positive preliminary feedback from researchers, students and collaborators who were given the opportunity to try out Bluff, we plan to conduct user studies to evaluate our tool's usability and functionality.

Acknowledgments

This work was supported in part by NSF grants IIS-1563816, CNS-1704701, NASA NSTRF, DARPA GARD, gifts from Intel, NVIDIA, Google, Amazon.

References

[1] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, and J.-D. Fekete. Weighted graph comparison techniques for brain connectivity analysis. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 483–492, 2013.

[2] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

[3] W. Brendel and M. Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

[4] K. Cao, M. Liu, H. Su, W. Jing, and S. Liu. Analyzing the noise robustness of deep neural networks. *arXiv preprint arXiv:2001.09395*, 2020.

[5] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, 2017.

[6] S. Carter, Z. Armstrong, L. Schubert, I. Johnson, and C. Olah. Activation atlas. *Distill*, 4(3):e15, 2019.

[7] S.-T. Chen, C. Cornelius, J. Martin, and D. H. P. Chau. Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 52–68. Springer, 2018.

[8] N. Das, S. Chaba, R. Wu, S. Gandhi, D. H. Chau, and X. Chu. Goggles: Automatic image labeling with affinity coding. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pp. 1717–1732, 2020.

[9] N. Das, M. Shanbhogue, S.-T. Chen, F. Hohman, S. Li, L. Chen, M. E. Kounavis, and D. H. Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 196–204, 2018.

[10] D. Erhan, Y. Bengio, A. Courville, and P. Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

[11] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean. A guide to deep learning in healthcare. *Nature medicine*, 25(1):24, 2019.

[12] R. Fong and A. Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8730–8738, 2018.

[13] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2014.

[14] S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 2019.

[15] G. Guo and N. Zhang. A survey on deep learning based face recognition. *Computer Vision and Image Understanding*, 189:102805, 2019.

[16] J. Heaton, N. G. Polson, and J. H. Witte. Deep learning in finance. *arXiv preprint arXiv:1602.06561*, 2016.

[17] F. Hohman, M. Kahng, R. Pienta, and D. H. Chau. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE transactions on visualization and computer graphics*, 25(8):2674–2693, 2018.

[18] F. Hohman, H. Park, C. Robinson, and D. H. Chau. Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. *IEEE VIS*, 2019.

[19] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pp. 125–136, 2019.

[20] M. Kahng, N. Thorat, D. H. P. Chau, F. B. Viégas, and M. Wattenberg. Gan lab: Understanding complex deep generative models using interactive visual experimentation. *IEEE transactions on visualization and computer graphics*, 25(1):1–11, 2018.

[21] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *ICML*, 2018.

[22] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[23] M. Liu, S. Liu, H. Su, K. Cao, and J. Zhu. Analyzing the noise robustness of deep neural networks. In *2018 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 60–71. IEEE, 2018.

[24] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Apr. 2018. https://openreview.net/forum?id=rJzIBfZAb.

[25] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

[26] A. S. Morcos, D. G. Barrett, N. C. Rabinowitz, and M. Botvinick. On the importance of single directions for generalization. *arXiv preprint arXiv:1803.06959*, 2018.

[27] A. Mordvintsev, C. Olah, and M. Tyka. Inceptionism: Going deeper into neural networks. 2015.

[28] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh, and K. Shaalan. Speech recognition using deep neural networks: A systematic review. *IEEE Access*, 7:19143–19165, 2019.

[29] A. P. Norton and Y. Qi. Adversarial-playground: A visualization suite showing how adversarial examples fool deep learning. In *Visualization for Cyber Security (VizSec), 2017 IEEE Symposium on*, pp. 1–4. IEEE, 2017.

[30] C. Olah, N. Cammarata, L. Schubert, G. Goh, M. Petrov, and S. Carter. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001, 2020.

[31] C. Olah, A. Mordvintsev, and L. Schubert. Feature visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization. doi: 10.23915/distill.00007

[32] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev. The building blocks of interpretability. *Distill*, 3(3):e10, 2018.

[33] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 582–597. IEEE, 2016.

[34] Y. Qin, N. Carlini, I. Goodfellow, G. Cottrell, and C. Raffel. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. *arXiv preprint arXiv:1903.10346*, 2019.

[35] A. S. Ross and F. Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[36] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[38] G. Tao, S. Ma, Y. Liu, and X. Zhang. Attacks meet interpretability: Attribute-steered detection of adversarial samples. In *Advances in Neural Information Processing Systems*, pp. 7717–7728, 2018.

[39] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

[40] Y. Wang, X. Ma, J. Bailey, J. Yi, B. Zhou, and Q. Gu. On the convergence and robustness of adversarial training. In *International Conference on Machine Learning*, pp. 6586–6595, 2019.

[41] S. Zhang, L. Yao, A. Sun, and Y. Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1):1–38, 2019.