REPRESENTATION QUALITY EXPLAIN ADVERSARIAL ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural networks have been shown vulnerable to adversarial samples. Slightly perturbed input images are able to change the classification of accurate models, showing that the representation learned is not as good as previously thought. To aid the development of better neural networks, it would be important to evaluate to what extent are current neural networks' representations capturing the existing features. Here we propose a way to evaluate the representation quality of neural networks using a novel type of zero-shot test, entitled Raw Zero-Shot. The main idea lies in the fact that some features are present on unknown classes and that unknown classes can be defined as a combination of previous learned features without representation bias (a bias towards representation that maps only current set of input-outputs and their boundary). To evaluate the soft-labels of unknown classes, two metrics are proposed. One is based on clustering validation techniques (Davies-Bouldin Index) and the other is based on soft-label distance of a given correct soft-label. Experiments show that such metrics are in accordance with the robustness to adversarial attacks and might serve as a guidance to build better models as well as be used in loss functions to create new types of neural networks. Interestingly, the results suggests that dynamic routing networks such as CapsNet have better representation while current deeper DNNs are trading off representation quality for accuracy.

1 INTRODUCTION

Adversarial samples are slightly perturbed inputs that can make neural networks misclassify. They are carefully crafted by searching for variations in the input that, for example, could decrease the soft-labels of the correct class. Since they were discovered some years ago (28), the number of adversarial samples have grown in both number and types. Random noise were shown to be recognized with high confidence by neural networks (20), universal perturbations, that can be added to almost any image to generate an adversarial sample, were shown to exist (18) and the addition of crafted patches were shown to cause networks to misclassify (4). Actually, only one pixel is enough to make networks misclassify (27). Such attacks can also be easily transferred to real world scenarios (12),(3) which confers a big issue as well as security risk for current deep neural networks' applications.

Albeit the existence of many defenses, there is not any known learning algorithm or procedure that can defend against adversarial attacks consistently. Many works have tried to defend by hiding or modifying the gradients to make neural networks harder to attack. However, a recent paper show that most of these defenses falls into the class of obfuscated gradients which have their own shortcomings (e.g., they can be easily bypassed by transferable attacks) (2). Additionally, the use of an augmented dataset with adversarial samples (named adversarial training) is perhaps one of the most successful approaches to construct robust neural networks (8),(11), (17). However, it is still vulnerable to attacks and has a strong bias to the type of adversarial samples used in training (31).

This shows that a deeper understanding of the issues are needed to enable more consistent defenses to be created. Few works focused on understanding the reason behind such lack of robustness. In (8) it is argued that Deep Neural Networks's (DNN) linearity are one of the main reasons. Recent investigations reveal that attacks are actually changing where the algorithm is paying attention (34), other experiments show that deep learning neural networks learn false structures that are easier to

learn rather than the ones expected (30) and an accuracy and robustness trade-off for models was shown to exist (32).

In this paper, we propose a methodology of how to evaluate the representation of machine learning methods. Based on these metrics, we reveal a link between deep representations' quality and attack susceptibility. Specifically, we propose a test called Raw Zero-Shot and two metrics to evaluate DNN's representations. The idea is that unknown classes provides hints over the representation of common features and attributes learned without representation bias (defined in Section 2).

1.1 Recent Advances in Attacks and Defenses

DNNs were shown vulnerable to many types of attacks. For example, they output high confidency results to noise images(20), universal perturbations in which a single perturbation can be added to almost any input to create an adversarial sample are possible (18), the addition of image patches can also make them misclassify (4). Moreover, the vulnerability can be exploited even with a single pixel, i.e., changing a single pixel is often enough to make a DNNs misclassify (27). Most of these attacks can be transformed into real world attacks by simply printing the adversarial samples (12). Moreover, crafted glasses (24) or even general 3d adversarial objects (3) can be used as attacks.

Although many defensive systems were proposed to tackle the current problems, there is still no consistent solution available. Defensive distillation in which a smaller neural network squeezes the content learned by the original DNN was proposed (22). However, it was shown to not be robust enough (6). Adversarial training was also proposed as a defense, in which adversarial samples are used to augment the training dataset (8),(11), (17). With adversarial training, DNNs increase slightly in robustness but not without a bias towards the adversarial samples used and while still being vulnerable to attacks in general (31). There are many recent variations of defenses in which the objective is to hide the gradients (obfuscated gradients) (16), (9) (25). However, they can be bypassed by various types of attacks (such as attacks not using gradients, transfer of adversarial samples, etc) (2),(33).

There are a couple of works which are trying to understand the reason behind such lack of robustness. To cite some, in (8), it is argued that the main reason may lie in DNNs' lack of non-linearity. Another work argues that the perturbations causes a change in the saliency of images which makes the model switch the attention to another part of it (34). False structures that are easier to learn were also shown related to the problem (30). Moreover, in (32), the accuracy and robustness trade-off was shown to exist.

1.2 Zero-Shot learning

Zero-Shot learning is a method used to estimate unknown classes which do not appear in the training data. The motivation of Zero-Shot learning is to transfer knowledge from training classes to unknown classes. Existing methods basically approach the problem by estimating unknown classes from an attribute vector defined manually. Attribute vectors are annotated to both known and unknown classes, and for each class, whether an attribute, such as "color" and "shape", belongs to the class or not is represented by 1 or 0. For example, in (13) the authors proposed *Direct Attribute Prediction (DAP)* model which learns each parameter for estimating the attributes from the target data. It estimates an unknown class of the source data which is estimated from the target data by using these parameters. Based on this research, other zero-shot learning methods have been proposed which uses an embedded representation generated using a natural language processing algorithm instead of a manually created attribute vector (37; 7; 21; 1; 5).

In (36) a different approach to estimate unknown classes is proposed. This method constructs the histogram of known classes distribution for an unknown class. In this approach, it is assumed that the unknown classes are the same if these histograms generated in the target domain and in the source domain are similar. This perspective is similar to our approach, because our method approach to represent an unknown class as the distribution of known classes. However, our objective is not estimating the unknown class and we do not use the source domain. Our objective here is to analyze DNNs' representation by using this distribution.

2 REPRESENTATION BIAS: DEFINITION

The objective of a supervised learning algorithm is perhaps to map the input to output in such a way that the decision boundary reflect the real decision boundary. To achieve this it is know that when dealing with complex problems, learning algorithms need to first learn a set of invariant features that are present throughout classes such that their recognition becomes robust against variations in the dataset.

Human beings, however, learn a set of invariant features that is not only able to solve current tasks or recognize current classes. We learn a set of features that is able to describe most if not all of unseen classes and unknown tasks. Thus, we define representation bias as the bias towards invariant features that describe current seen classes or tasks but fail to describe unknown classes and tasks.

3 RAW ZERO-SHOT

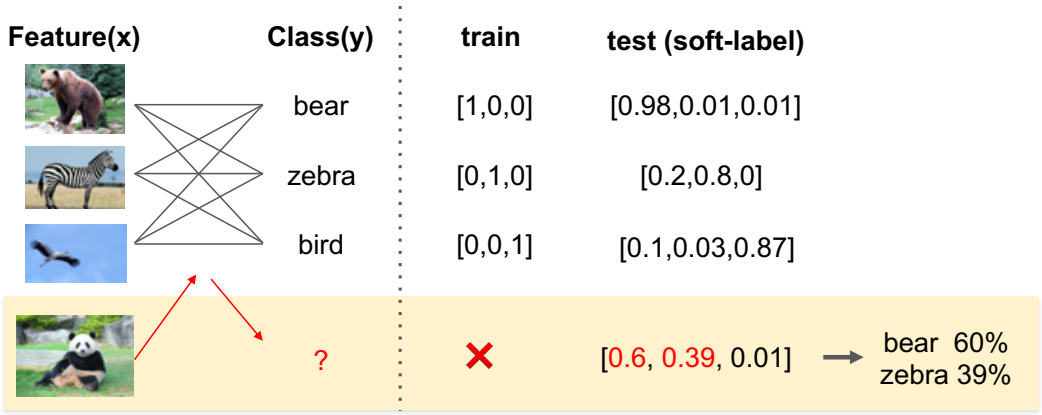

Figure 1: Raw Zero-Shot Illustration. Classifiers are trained on the dataset with one excluded class. In the test stage, images from the unknown class are presented and the soft-labels are recorded, which are used to infer the representation quality of the classifier. This is based on the principle that if the classifier learned general features, it should be able to use them to judge a sample from unknown class.

In this paper, we propose to evaluate the representation learned while avoiding the representation bias by conducting experiments over the soft-labels of the image in unknown classes. This is based on the hypothesis that if a model is capable of learning useful features, an unknown class would also trigger some of these features inside the model. We call this type of test over unknown classes and without any other information, Raw Zero-Shot (Figure 1).

The Raw Zero-Shot is a supervised learning test in which only $n-1$ of the n classes are shown to the classifier during training. The classifier also has only $n-1$ possible outputs. During testing, only unknown classes are presented to the classifier. The soft-labels outputted for the given unknown class is recorded and the process is repeated for all possible n classes, removing a different class each time.

To evaluate the representation quality, metrics computed over the soft-labels are used. These metrics are based on different hypothesis of what defines a feature or a class. In the same way that there are different types of robustness, there are also different types of representation quality. Therefore, metrics are rather complementary, each highlighting a different aspect of the whole. The following subsections define two of them.

3.1 DAVIES-BOULDIN METRIC - CLUSTERING HYPOTHESIS

Soft labels of a classifier composes a space in which a given image would be classified as a weighted vector in relation to the previous classes learned. Considering that a cluster in this space would constitute a class, we can use clustering validation techniques to evaluate the representation (Figure 2).

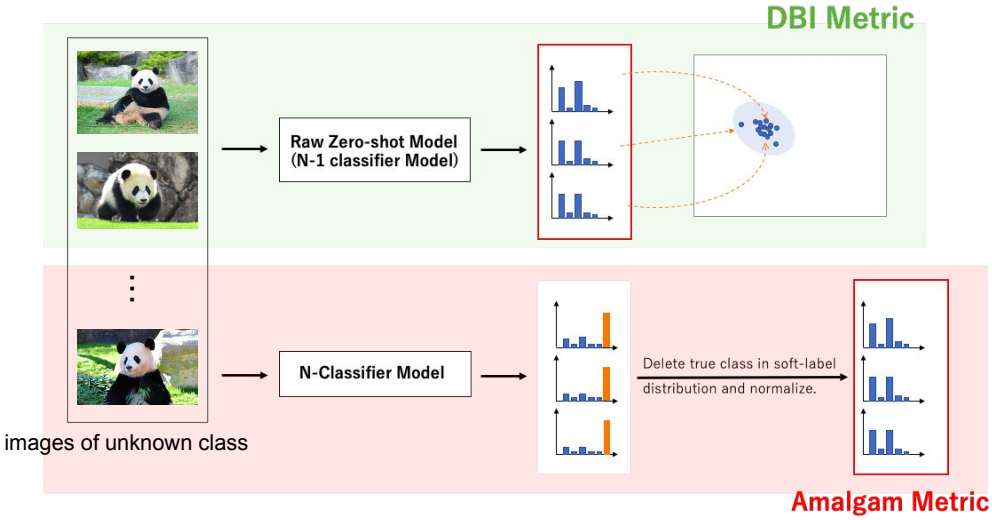

Figure 2: Illustration of both DBI and Amalgam metrics.

Here we choose for simplicity one of the most used metric in internal cluster validation, Davies-Bouldin Index (DBI). DBI is defined as follows:

$$DBI = \left(\frac{1}{n_e} \sum_{j=1}^{n_e} |\mathbf{e}_j - \mathbf{cn}|^2 \right)^{1/2}, \tag{1}$$

in which **cn** is the centroid of the cluster, **e** is one soft-label and n_e is the number of samples.

3.2 Amalgam Metric - Amalgam Hypothesis

If DNNs are able to learn the features present in the classes, it would be reasonable to consider that the soft-labels also describe a given image as a combination of the previous learned classes. This is also true when an image contains an unknown class. Similar to a vector space in linear algebra, the soft-labels can be combined to describe unknown objects in this space. This is analogous to how children describe previously unseen objects as a combination of previously seen objects. Differently from the previous metric, here we are interested in the exact values of the soft-labels. However, what would constitute the correct soft-labels for a given unknown class needs to be determined.

To calculate the correct soft-label of a given unknown class (amalgam proportion) automatically, we use here the assumption that accurate classifiers should output a good approximation of the amalgam proportion already. Therefore, if a classifier is trained in the n classes, the soft-labels of the remaining $n-1$ classes is the amalgam proportion (Figure 2 illustrates the concept). Consequently, the Amalgam Metric (AM) is defined as:

$$\begin{aligned} \mathbf{h'}_i &= \sum_{j=1}^{n_e} \mathbf{e'}_j \\ \mathbf{h}_i &= \sum_{j=1}^{n_e} \mathbf{e}_j \\ AM &= \left(\frac{1}{n} \sum_{i=1}^{n} \frac{\|\mathbf{h'}_i - \mathbf{h}_i\|_1}{n-1} \right), \end{aligned} \tag{2}$$

in which, **e'** is the normalized (such that they sum to one) soft-label from the classifier trained over n classes and **e** is the soft-labels from the classifier trained over $n-1$ classes.

Table 1: Attack Accuracy

Attack	ϵ	WideResNet	DenseNet	ResNet	NIN	AllConv	CapsNet	LeNet
Carlini L_2		93%	73%	96%	85%	90%	57%	78%
FGM L_∞	0.1	65%	68%	66%	78%	74%	86%	78%
	0.3	76%	77%	80%	82%	81%	87%	86%
	0.5	81%	80%	86%	84%	83%	86%	86%
BIM L_∞	0.3	98%	97%	97%	97%	97%	96%	94%
PGDM L_∞	0.1	65%	67%	71%	63%	58%	39%	70%
	0.3	94%	93%	93%	92%	90%	79%	88%
	0.5	96%	95%	94%	94%	93%	91%	90%
DeepFool	1E-06	44%	49%	50%	44%	62%	68%	63%
JSMA	1.00	99%	99%	99%	99%	98%	97%	97%

Table 2: Average Amount Of Pertubations (L_2) Required by each Attack

Attack	ϵ	WideResNet	DenseNet	ResNet	NIN	AllConv	CapsNet	LeNet
Carlini L_2		2421	2505	2420	2412	2400	2345	2347
FGM L_∞	0.1	2409	2457	2485	2424	2469	2455	2424
	0.3	2498	2511	2505	2508	2509	2506	2490
	0.5	2639	2650	2624	2627	2625	2653	2622
BIM L_∞	0.3	3317	3285	3283	3314	3394	3312	3553
PGDM L_∞	0.1	331	330	331	333	334	342	335
	0.3	935	934	935	942	946	962	952
	0.5	1468	1465	1466	1474	1482	1498	1502
Deep Fool	1E-06	2566	2619	2579	2584	2591	2490	2521
JSMA	1.00	162	157	146	222	261	569	235

4 Raw Zero-Shot Experiments

Here, we conduct Raw Zero-Shot experiments to evaluate the representation of DNNs. To obtain results over a wide range of architectures, we chose to evaluate CapsNet (a recently proposed completely different architecture based on dynamic routing and capsules) (23), ResNet (a state-of-the-art architecture based on skip connections)(10), Network in Network (NIN) (an architecture which uses micro neural networks instead of linear filters) (15), All Convolutional Network (AllConv) (an architecture without max pooling and fully connected layers)(26) and LeNet (a simpler architecture which is also a historical mark) (14). All the experiments are run over the CIFAR dataset by using a training dataset with all the samples of one specific class removed. This process is repeated for all classes, removing the samples of a different class each time.

To analyze the correlation between representation metrics and robustness against adversarial attacks, we conducted adversarial attacks on all the architectures tested using the most well known algorithms such as Carlini (6), Fast Gradient Method (FGM) (8), Basic Iterative Method (BIM)(12), DeepFool (19), Jacobian-based Saliency Map Attack (JSMA) (35), Projected Gradient Descent Method (PGDM) (17) (Table 1 and 2).

For all tests, ϵ is fixed to the corresponding value given in the table. However, different methods have different meanings for ϵ: (a) for pixel attacks, ϵ is the maximum number of pixels to be changed, (b) for threshold attacks, ϵ is the maximum amount of change allowed per pixel, (c) for FGM, BIM and PGDM, ϵ is the attack step size (input variation), (d) for DeepFool, ϵ is the overshoot parameter and (e) for JSMA, ϵ corresponds to the maximum percentage of perturbed features.

Table 3 shows the ranking based on the attack accuracy and their required perturbation (Tables 1 and 2). In general, CapsNet is shown to be the most robust, AllConv and DenseNet follow with a good placement while the remaining networks vary depending on the perspective used to analyze (i.e.,

Model	Rank (Acc)	Rank (L_2)	Diff
WideResNet	3	7	*4*
DenseNet	2	4	*2*
ResNet	7	6	*1*
NIN	4	5	*1*
AllConv	5	2	*3*
CapsNet	1	1	*0*
LeNet	6	3	*3*

Table 3: Overall ranking for both accuracy (Acc) and L_2 attacks. The rankings are obtained by ordering the average accuracy and L_2 for all attacks.

higher accuracy or lower L_2). These robustness rankings will be used in the next sections to verify the relationship of robustness against adversarial samples and metrics to evaluate the representation quality.

4.1 Experiments on DBI Metric

Table 4 shows the results with the DBI metric (the smallest the better) and the respective ranking of each neural network. According to this metric, CapsNet possesses the best representation of all networks tested. LeNet is considered the second best neural network regarding representation, followed by AllConv, NIN and then deeper neural networks. DBI metric matches extremely well with both accuracy and L_2 ranking based on robustness against adversarial samples. Most differences in Hamming distance lies in the exact places in which both accuracy and L_2 rankings differ. Notice that DBI metric does not use anything related to attacks and still arrives at similar rankings.

The fact that LeNet and other relatively simpler networks achieve a high representation quality which is at odds with accuracy may seem extremely unlikely but as discussed in (32), it is possible for accurate models to trade off robustness for accuracy. DBI suggests that this trade off happens because the representation quality has worsen. Interestingly, LeNet and other simple networks are also easier to attack (low rank accuracy) but need more perturbation to achieve the same accuracy (high rank L_2). Therefore, LeNet and other simple networks might be easier to attack because the search space is less complicated (less obfuscation (2)), however this does not mean they are less robust. Or, as DBI suggests, LeNet and other simple networks might have achieved relatively good representations but without high accuracy.

To enable a visualization of this metric, we plotted in Figure 3 a projection into two dimensions of all the points in decision space of unknown classes. All this is done while preserving the high-dimensional distance between the points. Here we use the Isomap (29) to achieve this effect. It can be easily observed that CapsNet's results for unknown classes are more clustered and thus form a better defined cluster than other architectures.

Model	DBI	Ranking	Diff (Acc)	Diff (L_2)
WideResNet	0.58±0.14	7	4	0
DenseNet	0.60±0.14	6	4	2
ResNet	0.63±0.13	5	2	1
NIN	0.62±0.09	3	1	2
AllConv	0.64±0.10	4	1	2
CapsNet	0.23±0.01	1	0	0
LeNet	0.51±0.02	2	4	1

Table 4: Mean DBI, the rank regarding this representation metric and the Hamming distance to the robustness rankings against adversarial samples for each neural network.

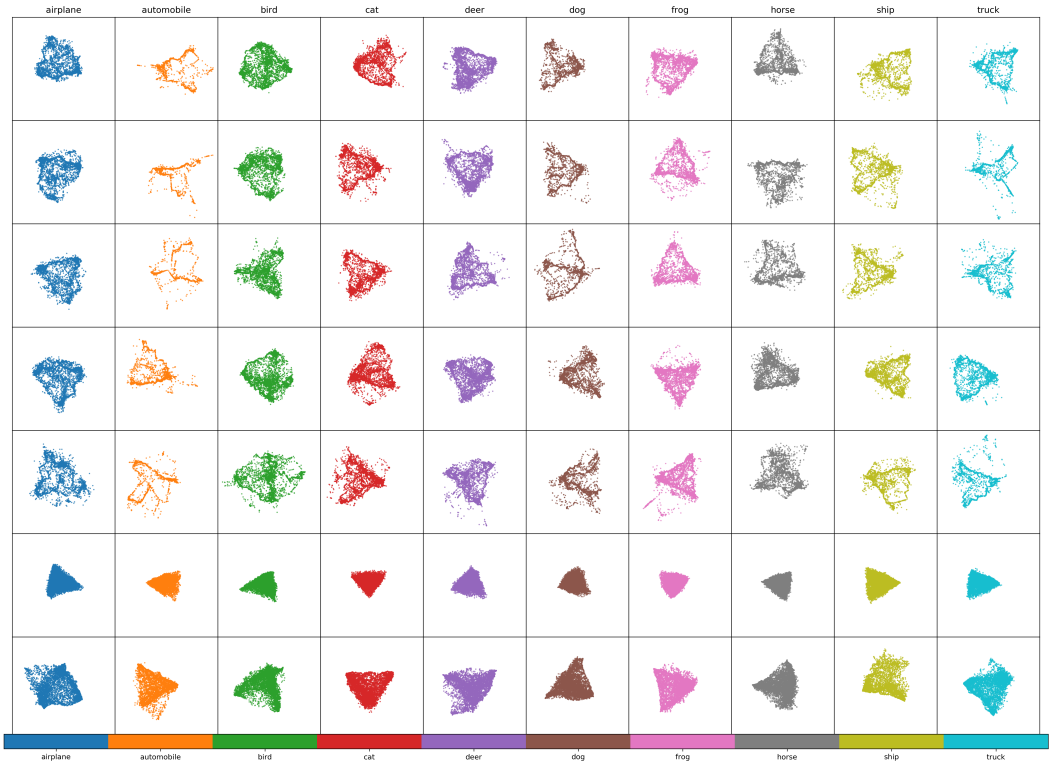

Figure 3: Visualization of the results in Table 4 using a topology preserving two-dimensional projection with Isomap. Each row shows the Isomap of one architecture. From top to bottom: WideResNet, DenseNet, ResNet, NIN, AllConv, CapsNet, LeNet.

Model	AM	Ranking	Diff (Acc)	Diff (L_2)
WideResNet	249.48±135.60	7	4	0
DenseNet	296.76±100.63	5	3	1
ResNet	281.29±107.82	6	1	0
NIN	203.15±93.08	4	0	1
AllConv	101.12±54.98	1	4	1
CapsNet	124.48±62.23	2	1	1
LeNet	144.10±75.18	3	3	0

Table 5: AM value for five different architectures and their respective ranking. The Hamming distance of the AM's ranking and both the accuracy and L_2 robustness ranking against adversarial samples are also shown.

4.2 Experiments on Amalgam Metric

In this section, the AM for all the networks is evaluated which is based on the similarity of soft-labels for networks that were trained in all classes. The results shown in Table 5 reveal almost the same representation ranking as the robustness rankings related to L_2. Interestingly, although both DBI and AM differ widely in concept and calculation procedure, the rankings are both similar and close to the L_2. This further suggests that both metrics agree on what would be a good representation quality and can be used to evaluate representation in newer methods.

To enable a visualisation of the metric, the computed histograms ($\mathbf{h'}_i$ and $\mathbf{h}_i$ from Equation 2) are plotted in Figure 6. It is interesting to note that in Figure 6 the histograms from CapsNet are clearly different from the other ones, in accordance with the complete different architecture employed by CapsNet. This reveals that this metric is able to capture such representation differences.

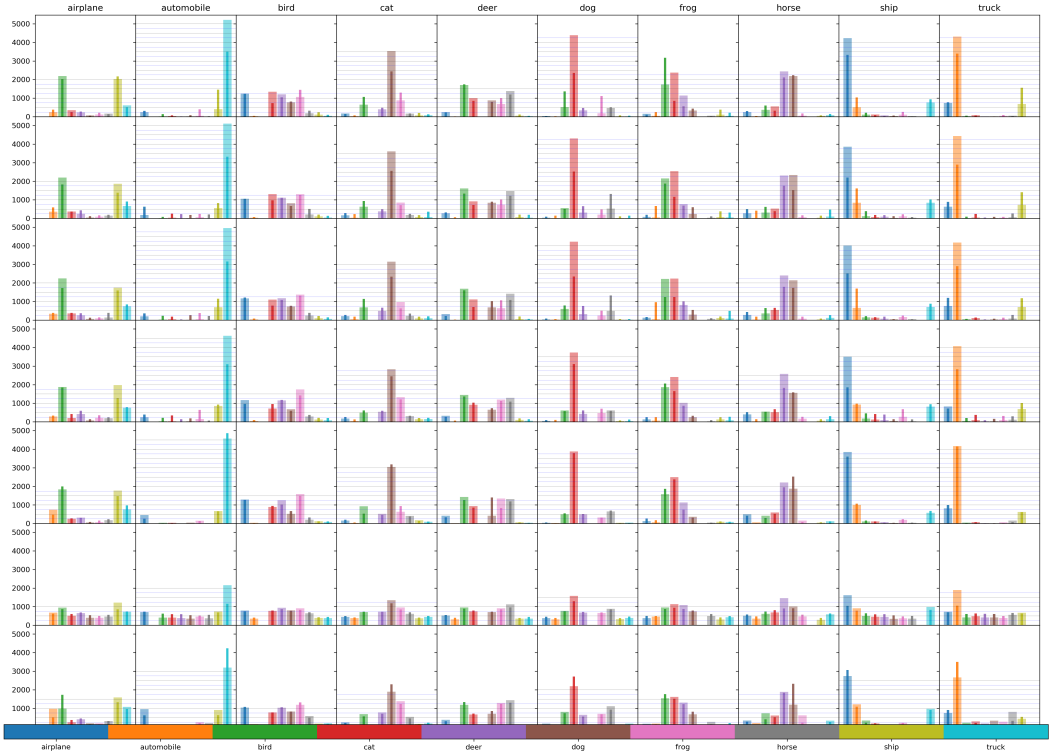

Figure 4: Histograms from which the AM is calculated. Each row shows the histograms of one architecture. From top to bottom: WideResNet, DenseNet, ResNet, NIN, AllConv, CapsNet, LeNet.

5 Parts of a Whole

In this work, beyond proposing a quality assessment for learning systems and showing their relationship to adversarial attacks, we aim to fill a gap between current articles trying to understand adversarial attacks. The false structures that are easier to learn (30) can be understood as structures learned with representation bias. Therefore, they are only invariant to the dataset and not unseen classes. At the same time, we further support the existence of a trade-off between accuracy and robustness for deeper DNNs first point out in (32). Having said that, other types of networks such as CapsNet seem to avoid it to some extent. Therefore, the trade-off is shown to vary with the architecture and computing dynamics of a model.

6 Conclusions

Here we proposed the Raw Zero-Shot method to evaluate the representation of classifiers. In order to score the soft-labels, two metrics were formally defined based on different hypothesis of representation quality. Results suggest that the evaluation of representation of both metrics (DBI and AM) are linked with the robustness of neural networks. In other words, easily attacked neural networks have a lower representation score.

Interestingly, LeNet scores well in both metrics albeit being the least accurate. LeNet is followed up by AllConv and NIN which are less complex/deep than other models which suggests that deeper architectures might be trading-off representation quality for accuracy. These results shown here further support the claim that there is a trade-off between accuracy and robustness in current deep learning (32).

Thus, the proposed Raw Zero-Shot was able to evaluate the representation quality of state-of-the-art DNNs and show their shortcomings in relation to adversarial attacks, explaining many of the current problems. It also opens up new possibilities for both the evaluation (i.e. as a quality assessment) and the development (e.g., as a loss function) of neural networks.

REFERENCES

[1] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2927–2936, 2015.

[2] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

[3] A. Athalye and I. Sutskever. Synthesizing robust adversarial examples. In *ICML*, 2018.

[4] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

[5] M. Bucher, S. Herbin, and F. Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *European Conference on Computer Vision*, pages 730–746. Springer, 2016.

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

[7] Y. Fu, Y. Yang, T. Hospedales, T. Xiang, and S. Gong. Transductive multi-label zero-shot learning. *arXiv preprint arXiv:1503.07790*, 2015.

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

[9] C. Guo, M. Rana, M. Cisse, and L. van der Maaten. Countering adversarial images using input transformations. In *ICLR*, 2018.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] R. Huang, B. Xu, D. Schuurmans, and C. Szepesvári. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015.

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

[13] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009.

[14] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[15] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[16] X. Ma, B. Li, Y. Wang, S. M. Erfani, S. Wijewickrema, G. Schoenebeck, D. Song, M. E. Houle, and J. Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*, 2018.

[17] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

[18] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 86–94. IEEE, 2017.

[19] 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*, pages 2574–2582, 2016.

[20] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.

[21] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[22] 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)*, pages 582–597. IEEE, 2016.

[23] S. Sabour, N. Frosst, and G. E. Hinton. Dynamic routing between capsules. In *Advances in neural information processing systems*, pages 3856–3866, 2017.

[24] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1528–1540. ACM, 2016.

[25] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018.

[26] J. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015.

[27] J. Su, D. V. Vargas, and S. Kouichi. One pixel attack for fooling deep neural networks. *arXiv preprint arXiv:1710.08864*, 2017.

[28] C. e. a. Szegedy. Intriguing properties of neural networks. In *In ICLR*. Citeseer, 2014.

[29] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[30] L. Thesing, V. Antun, and A. C. Hansen. What do ai algorithms actually learn?-on false structures in deep learning. *arXiv preprint arXiv:1906.01478*, 2019.

[31] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. Ensemble adversarial training: Attacks and defenses. In *ICLR*, 2018.

[32] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

[33] J. Uesato, B. O'Donoghue, P. Kohli, and A. Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International Conference on Machine Learning*, pages 5032–5041, 2018.

[34] D. V. Vargas and J. Su. Understanding the one-pixel attack: Propagation maps and locality analysis. *arXiv preprint arXiv:1902.02947*, 2019.

[35] R. Wiyatno and A. Xu. Maximal jacobian-based saliency map attack. *arXiv preprint arXiv:1808.07945*, 2018.

[36] Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE international conference on computer vision*, pages 4166–4174, 2015.

[37] Z. Zhang and V. Saligrama. Zero-shot recognition via structured prediction. In *European conference on computer vision*, pages 533–548. Springer, 2016.

SUPPLEMENTARY WORK

A EXTENDED ANALYSIS OF DBI METRIC

Figure 5 shows a visualization of DBI's results using the multidimensional scaling method entitled t-Distributed Stochastic Neighbour Embedding (t-SNE). t-SNE provides an alternative vizualization for the Isomap plot from the main text. Similarly to the Isomap plot, t-SNE also verifies the closer distribution of points in high-dimensional space for CapsNet.

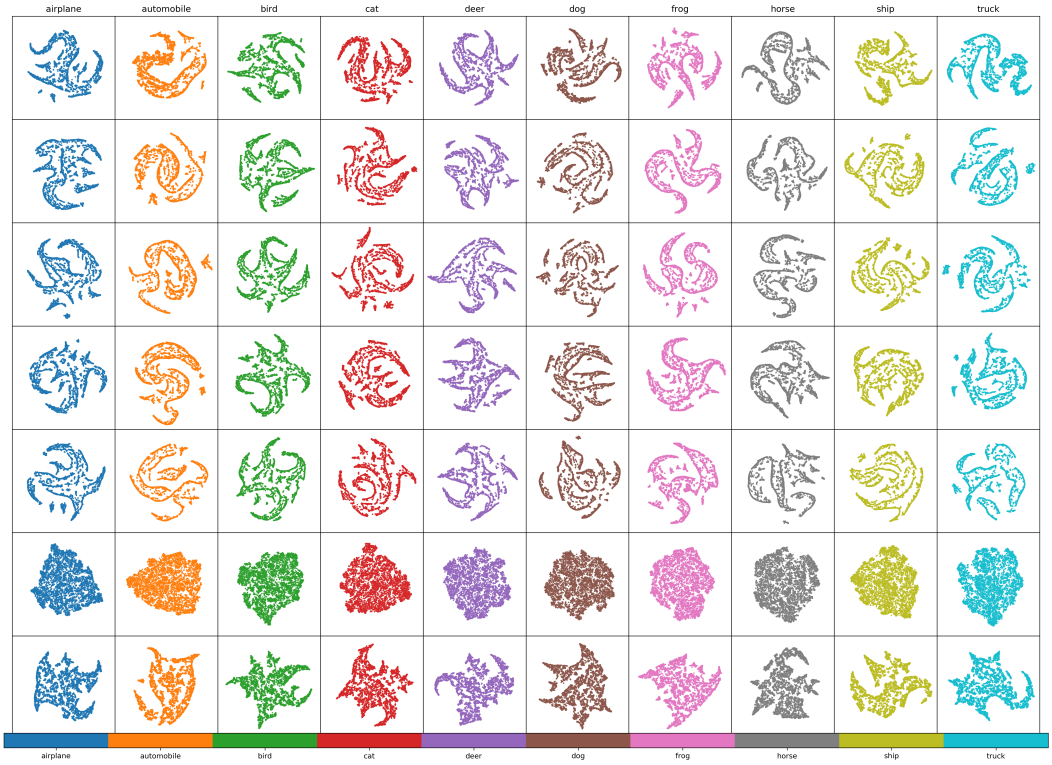

Figure 5: Visualization of the DBI Metric with t-Distributed Stochastic Neighbour Embedding (t-SNE) which focuses on the neighbour distances. Each row shows the t-SNE projections in two dimensional space for one architecture. From top to bottom: WideResNet, DenseNet, ResNet, NIN, AllConv, CapsNet, LeNet.

B EXTENDED ANALYSIS OF AMALGAM METRIC

Figure 6 shows a visualization of equation 3 which is part of the main equation of Amalgam Metric. This equation shows the difference of the histograms. It can be noted from the figure that for most labels of CapsNet and AllConv the difference is relatively small. In Figure 7, the absolute difference between histograms for each class is plotted. Albeit some variance in the mean, there is no strong influence of classes in the AM result, with top and low scorers keeping most of their difference throughout.

$$
\begin{aligned}
D &= |\, \mathbf{h'}_i - \mathbf{h}_i \,|, \quad \textit{where} \\
\mathbf{h'}_i &= \sum_{j=1}^{n_e} \mathbf{e'}_j \quad \textit{and} \quad \mathbf{h}_i = \sum_{j=1}^{n_e} \mathbf{e}_j
\end{aligned}
\tag{3}
$$

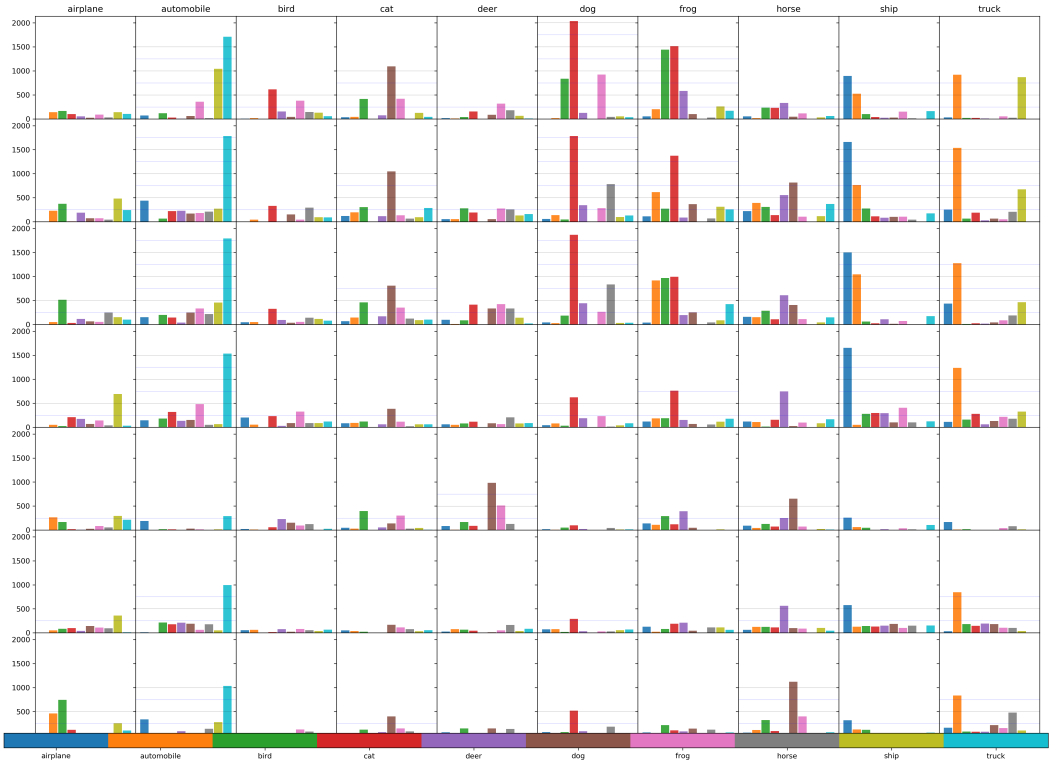

Figure 6: Histograms of D for each soft label. From top to bottom: WideResNet, DenseNet, ResNet, NIN, AllConv, CapsNet, LeNet.

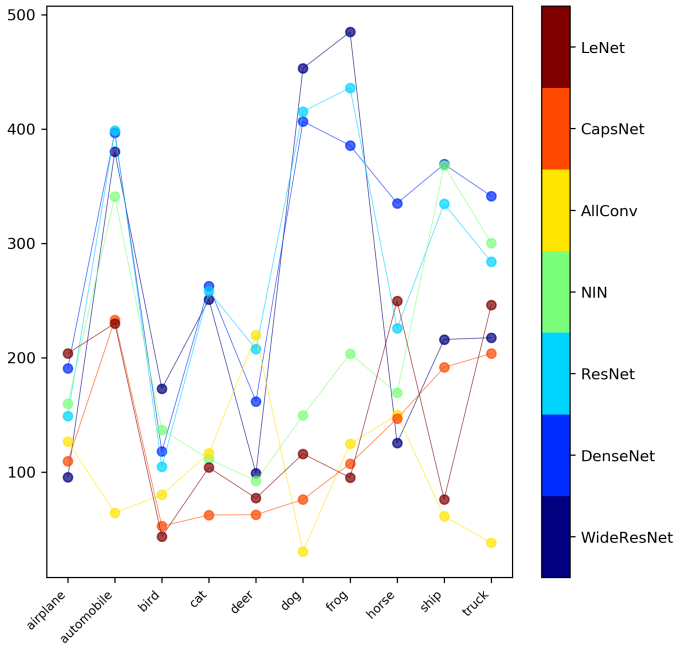

Figure 7: Absolute difference between histograms for each class.