Evaluation of neural networks defenses and attacks using NDCG and reciprocal rank metrics

Haya Brama1 · Lihi Dery1 · Tal Grinshpoun1

Published online: 21 December 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH, DE 2022

Abstract
The problem of attacks on neural networks through input modification (i.e., adversarial examples) has attracted much attention recently. Being relatively easy to generate and hard to detect, these attacks pose a security breach that many suggested defenses try to mitigate. However, the evaluation of the effect of attacks and defenses commonly relies on traditional classification metrics, without adequate adaptation to adversarial scenarios. Most of these metrics are accuracy-based and therefore may have a limited scope and low distinctive power. Other metrics do not consider the unique characteristics of neural network functionality or measure the effectiveness of the attacks indirectly (e.g., through the complexity of their generation). In this paper, we present two metrics that are specifically designed to measure the effect of attacks, or the recovery effect of defenses, on the output of neural networks in multiclass classification tasks. Inspired by the normalized discounted cumulative gain and the reciprocal rank metrics used in information retrieval literature, we treat the neural network predictions as ranked lists of results. Using additional information about the probability of the rank enabled us to define novel metrics that are suited to the task at hand. We evaluate our metrics using various attacks and defenses on a pre-trained VGG19 model and the ImageNet dataset. Compared to the common classification metrics, our proposed metrics demonstrate superior informativeness and distinctiveness.

Keywords Adversarial examples · Cyber security · Evaluation metrics · Information retrieval · Multi-class classification · NDCG · Neural networks · Reciprocal rank

1 Introduction
As research fields develop rapidly, evaluation metrics for these fields require adequate adaptation. Neural networks (NNs), especially in the form of deep learning, have achieved tremendous progress in pattern recognition. However, they remain vulnerable to attacks of adversarial examples (AEs). In such attacks, the attacker modifies the input, while carefully constraining the maximal size of modifications (perturbations) or the number of modified (perturbed) pixels in the original input image, so that the adversarial changes will be indistinguishable for humans. Due to these careful changes, the NN might incorrectly classify the AE [1–3]. In this paper, we introduce a new evaluation metric for the performance of artificial neural networks in the context of adversarial attacks.

Attacks on NNs may be effective even under strict constraints, such as a very narrow range of allowed perturbation size. Moreover, AEs can be transferred across different models. This opens the possibility of black-box attacks, where the attacker can create an effective AE even without access to the attacked NN’s parameters [4]. AEs may be robust to physical transformations [5] and therefore pose a security breach not only for cyber-world applications [6] but also for real-world systems, such as computer vision of autonomous vehicles [7]. Although various defense techniques have been suggested, none of them can ensure complete effective defense in all settings [8–10].

This emerging research field requires adequate evaluation, in order to measure the effect of attacks and defenses on the functionality of the NN and to compare different methods. Data classification literature employs several evaluation metrics in the pursuit of the most suitable or optimized classifier [11]. The evaluation metrics serve as a measurement

---

1 Industrial Engineering and Management and Ariel Cyber Innovation Center, Ariel University, 40700 Ariel, Israel
tool for the generalization ability of trained models. Furthermore, evaluation metrics play a role during classifier training, since they can guide a stochastic or heuristic search and help discriminate the optimal solution from the large space of solutions. The majority of those measures are based on some variation of the accuracy or the error rate, as will be expanded below. Their simplicity makes them very intuitive and easy to use, but this high applicability comes with a price of rather limited informativeness and discriminative power [12–14]. When the possibility of attacks and defenses was introduced in the literature, accuracy-based evaluation metrics were employed by default. However, as their original objective was to improve and assess the NNs’ training in a benign setting, these measures are not necessarily the most appropriate choice for assessment in adversarial settings. As stated by Arp et al. [15], many machine learning performance measures are not suitable in the context of security, since they may provide insufficient estimations or obscure experimental results. In addition, as various security-related problems deal with more than two classes, multi-class metrics are required and introduce further subtle pitfalls. Building on the observations of Arp et al., we argue that in many contexts current metrics do not always reflect the diversity of the adversarial scenarios and fail to capture many of the strengths and weaknesses of attacks and defenses.

To understand the limitations of current metrics, we first need to consider how they handle the NN’s output. The NN outputs a vector, whose length reflects the number of output labels. Each entry of that vector is proportional to the activity level of a different output neuron, which stands for a single category. The corresponding category of the most activated output neuron is selected as the NN’s classification decision. The ratio between the highest activation and the overall activation of all the neurons is referred to as the confidence of the NN in its decision. Accuracy is usually measured as top-1 accuracy, which is the prevalence of the true label of the input matching the first-ranked predicted category. Alternatively, accuracy can be measured more flexibly as the prevalence of the true label among the top-k categories in the output vector, a measure known as top-k accuracy. Top-1 accuracy answers the limited question of “did the neural network output the exact correct label?” For example, according to top-1 accuracy, misclassifying the image ‘green snake’ as ‘green mamba’ is as wrong as misclassifying a ‘green snake’ image as ‘baseball’, see Figure 1a, b. Even the top-k accuracy measure does not always capture the whole classification picture, since it informs whether the correct label appeared in those top-k predictions, but does not indicate where it is located within those predictions, or how far it is from their semantic neighborhood. There are many applications where this distinction is important, for example when the goal is to visually identify weapons in an image for security reasons, and an attacker masks weapons as animals [5]; in such a case, it is less crucial if a defense strategy turns the adversarial ‘animal’ into one kind of weapon or another, as long as the object is now identified as a threat.

Top-1 accuracy may cause another kind of misinterpretation. Consider a scenario in which an attack causes the NN to merely switch between the top-k predictions. When an image consists of several objects, the resultant classification decision is not necessarily wrong per se. For example, classifying the image labeled as ‘harvester’ in the ImageNet dataset as ‘hay’ (Fig. 1c) provides true information regarding content of the image. To summarize, the current top-k accuracy metric does not express the semantic and conceptual gap between the correct and adversarial predictions, while this may be valuable information in many use cases.

In this paper, we present two novel evaluation metrics, designed to specifically address the unique challenges posed by AE attacks. The new metrics reflect the gap between the correct NN predictions and the adversarial predictions, i.e., the NN’s prediction output when under attack, in two cases: (a) without a defense algorithm or (b) when a defense algorithm is employed. These metrics enable us to go beyond binary metrics and compare the relative strength of attacks and defenses. We consider strong attacks and weak attacks as attacks that result in a wider or narrower gap, respectively. Similarly, strong defenses manage to narrow that gap more than weak defenses.

Since the gap is an outcome of the importance order by which the different categories are organized within the NN output vector, the suggested evaluation relates to the literature on information retrieval and ranking problems. The current paper concentrates on binary and multiclass classification and on the image processing domain. However, the presented algorithms may also be applicable to other domains in which attacks may occur, such as banking and finance [16], cyber security [17], automatic speech recognition [18], or medical machine learning [19].

Contributions

Our contributions herein are twofold:

- Novel evaluation metrics—we present two new ranking measures, specifically utilized to accommodate classification in adversarial settings. Our metrics enable a new perspective for the comparison of attacks and defenses.
- Defense and attacks quality assessment—we conduct an extensive comparison of state-of-the-art attacks and defenses. These attacks and defenses have been presented in different papers, but have not yet been compared in a rigorous manner.

The rest of this paper proceeds as follows: in Sect. 2 we present a brief review of the relevant literature, starting with adversarial attacks, moving to defense methods against attacks, and ending with assessment methods in both deep-
Evaluation of neural networks defenses and attacks using NDCG and reciprocal rank metrics

2 Related work

2.1 Adversarial attacks

A basic way to create an attack is to gradually add noise to the input until the input is misclassified (e.g., a salt and pepper attack [20]). More sophisticated algorithms add noise in a more focused and effective way, enabling the attacker to control the target adversarial category, the strength of the adversarial effect, etc. The differentiability of NNs allows defining a variety of optimization problems that focus on satisfying the attacker’s demands with minimal added noise. Techniques for solving such problems include the use of simple gradients (e.g., FGSM [8]), estimations of the Hessian (e.g., L-BFGS [1]), or evolutionary algorithms [21], in one-shot or iterative way (e.g., DeepFool [22] and Carlini–Wagner (CW) [23]), with restricted $l_2$, $l_\infty$, or $l_0$-norm.

The above techniques can be applied to an ensemble of models, and thus increase the probability of creating a transferable attack, or even a universal attack [24]. Some attacks target other aspects of NN computation; for example, they attempt to change the heatmaps produced by various interpretation methods [25,26], or attack through model manipulation [27] or through poisoning the training data [28,29] rather than through input perturbations.

2.2 Defense methods

The arms race between attackers and defenders has led to the development of many defense techniques. Sometimes the defense strategies reflect different interpretations of the underlying causes of NN vulnerability to adversarial attacks. The insufficient mapping of the input space during training [30] was argued to be a consequence of the high dimensionality of the input or its nonlinear processing [29,31,32], in which case a dimensionality reduction could help circumvent many attacks. Similar motivation led to different loss function or activation function modifications in order to enhance the model’s robustness [4,33]. As opposed to the approaches which regard the complexity of NN computation, Goodfellow et al. [8] put the blame on the linear summation of the initial perturbations in deep models, and suggested defending the models through adversarial training, where AEs are part of the training set. Many have tried to just cancel the adversarial perturbation using different kinds of input transformations, such as resizing or jpeg compression [9,34]. Others used another model or function in order to spot suspicious inputs (see [35] for an extended literature review). Some defenses focus on masking the computational process of the model, for example through non-differentiable layers [36]. Ilyas et al. [10] claim that NNs do learn to classify correctly based on their training set, and that their vulnerability reflects higher-order features that exist.
in the dataset and are not accessible to humans. Therefore, their suggested defense was to train on particular datasets that do not contain such features.

As new defenses emerge, adversaries adapt their attacks and create more sophisticated AEs that break many of the known defenses [37]. Carlini et al. [38] claim that many defenses fail to correctly evaluate their effectiveness, either by considering only a small subset of possible attacks or by underestimating the possibility of adaptive attacks. Several works have therefore developed certified defenses, which provide provable robustness against attacks under certain conditions [39–41]. Despite the extensive research, to the best of our knowledge, there is no defense that guarantees absolute robustness against AEs with confined distortion levels.

### 2.3 Evaluation metrics

Research on the security assessment of deep learning in adversarial environments has focused mainly on available defense methods and different threat models. However, several papers consider the question of what assessment methodology and evaluating metrics should be used [39,42–44]. Batsani et al. [42] used the distortion of AEs as the robustness metric, while Weng et al. [45] proposed a new metric for robustness called CLEVER, based on extreme value theory. Peng et al. [46] proposed the EDLIC framework for quantitative analysis on different threat models and defense techniques. They also introduced a set of evaluating metrics based on distortion level, accuracy, and the Euclidean distance between predictions.

The accuracy of a model is usually measured using the top-1 prediction, even though sometimes the top-5 predictions are also considered. This metric reflects the number of times the model predicted the correct label, and it is also the basis for several commonly used evaluation measurements [47]. The most fundamental measurements are Recall (also called Sensitivity), which stands for the proportion of real positive cases that are correctly predicted as such (i.e., true positives), and Precision (also called Confidence), which is the ratio between true positive predictions and the overall positive predictions. The same procedure can be applied to the negative predictions, which yield the Specificity (also called Inverse Recall) and Inverse Precision measures. Fall-out is the proportion of real negatives in the predicted-positive cases, and Receiver Operating Characteristics (ROC) analysis plots the recall against the fallout. In addition, F-measure (or F1 score) is a function of Precision and Recall that can be found in deep-learning literature.

Classification metrics are mostly binary, i.e., a prediction is either correct or incorrect. However, we would like to evaluate the prediction ranking. A domain that shares a similar objective is the evaluation of Learning to Rank techniques (LTR) [48], which are mostly studied in the context of information retrieval (IR) and recommender systems. The goal of LTR is to rank a list of objects, e.g., a search engine is required to rank websites from most to least relevant for a given query.

A known interface between LTR and neural network is referred to as neural ranking models [49], where NNs are used to learn a ranking function [50]. Some approaches focus on learning low-dimensional representations of the input and then performing ranking using traditional IR models or other similarity metrics [51,52]. Others attempt to directly learn the ranking function [53,54] using various network architectures and learning objectives, as reviewed in [50]. Qin et al. [55] propose to directly optimize ranking metrics such as NDCG, a framework that was revisited by [56] in light of recent advances and was found empirically beneficial. However, as discussed in [56], most techniques do not optimize a ranking metric directly but rather use methods such as smoothed approximation of the metric [57] or an indirect boosting [58].

The above studies used NNs’ output for ranking, whereas herein our goal is to apply ranking assessment to the classification output of NNs. Many evaluation metrics of ranked lists exist. One of the common ones is the mean reciprocal rank (MRR) which is simply the multiplicative inverse of the rank of the first correct answer. Normalized discounted cumulative gain (NDCG) [59] is another known metric, which we will hereby expand. Other common metrics such as expected reciprocal rank (ERR) [60] consider the user’s satisfaction from a ranked list and are irrelevant in our context.

To conclude, evaluation for neural ranking models exists, as described above. Evaluation for classification models also exists but is essentially binary. Yet, while in classification tasks NNs output a ranking of the classes in question, it is common in these tasks to consider only the result ranked in the first place, and ignore the rest of the ranking. We suggest herein to evaluate the whole ranking and thus provide a more accurate evaluation for classification tasks.

### 3 Methodology

When the topmost prediction is correct for a given AE, it is common to consider this as an effective defense against the AE attack. Similarly, a defense is considered ineffective when the topmost prediction is wrong. This dichotomous division into effective and ineffective defenses does not consider other subtle scenarios. For example, the top-most prediction may be correct but the AE has caused: (a) a lower probability for the top-most prediction or (b) a distortion in other places in the ranked list of outputs.

Thus, we propose to increase the expressiveness of the evaluation and distinguish between various cases of effective and ineffective defenses. Similarly, we would like to compare
different successful attacks, where the topmost prediction is always wrong, but the deviation from the true label may vary. Our goal is to step beyond the mere indication of whether the topmost item is classified correctly; instead, we evaluate the performance of the entire NN’s ranked list of predictions.

A visual illustration of the desired expressiveness is presented in Fig. 2. In (a), a benign image of a revolver (left column) was used as seed by two different attacks, resulting in AE I and AE II (middle and right columns, respectively). For each input image, the NN outputs a different prediction list whose three topmost categories are depicted below the image, along with their corresponding probabilities. The predictions for the AEs have the same top-1 and top-5 accuracy scores, since both place the correct category in the third rank. However, AE I is more effective since the two topmost predictions are very different than the original ‘revolver’ label. AE II, on the other hand, merely switched the order of the three topmost benign categories, and therefore its output is more similar to the benign predictions list.

Similarly, Fig. 2b presents illustrative examples of the topmost predicted categories for three different defenses (purple background) against AE I (left column, red background). None of the defenses managed to change the top-1 and top-5 scores of the predictions for AE I, but the outputs they produce do differ in their quality. Though Defense I failed to restore the original top-1 category, it reduced the confidence of the NN in its wrong top-1 prediction. Defense II further improved the result by promoting the true ‘revolver’ category from rank three to rank two. Defense III is the most effective since even though the top-1 prediction is wrong, all top categories come from the same semantic neighborhood as the correct category. Our desired metric should be able to account for all the above differences and consider the rank and probability assigned to the correct category, as well as the similarity between this category and the topmost categories of a given list.

Evaluating the quality of an outputted ranking is not a trivial problem, many metrics have been considered and there is no single metric that is optimal for every type of ranking problem [61]. When dealing with attacks, it is necessary to consider two unique problem features: the baseline ranking is unknown, and the ranking is nonlinear.

The baseline ranking is unknown In order to compare different classification decisions under attacks or defenses, the top-1 prediction in each case should be evaluated by its distance from the true top-1 prediction. Natural language processing techniques such as word2vec [52] might provide such distance estimation. In this method, a NN is used to create word embeddings, where each distinct word in the vocabulary is represented by a unique vector, and the semantic relationships between words are expressed and measured as the distances in the vector space. However, this universal and fixed proximity function between categories will not necessarily allow an accurate estimation of the adversarial effect, because the associations between the categories are context-dependent. Some categories relate to each other, e.g., different kinds of small dogs are closely categorized using word2vec, truly reflecting the relationship between these categories. In other cases, word2vec may fail when the unique features of a given image may affect the relationships between the categories. For example, a specific posture of a dog in an image, the image perspective, or the background

![Fig. 2](image-url)
textured may increase its similarity to other objects that are not close according to word2vec. A poodle against a background of flowers may result in a high ranking for objects from the plants category, even though the word embedding of ‘poodle’ and ‘plants’ are not close in the vector space.

The ranking is not linear Confusion between top-ranked categories (e.g., replacing ‘toy poodle’ with ‘miniature poodle’) is much more probable than the replacement of the first-ranked category with a distant one. Therefore, the effort needed by an attack to cause a considerable deviation from the original category is not a linear function of the distance between categories on the scale. A running example (which will be thoroughly discussed in the next section) is presented in Table 1. The benign image is that of a toy poodle (row 2).

In the benign setting, the fourth prediction of the model is ‘cocker spaniel’, while the fifth prediction is ‘seat belt’. This fifth prediction is a dramatic decrease in relevance, and an attack that manages to cause misclassification of the image as a seat belt rather than some (‘miniature poodle’, ‘standard poodle’, ‘cocker spaniel’) is considered much stronger. However, an attack that causes confusion between the 49th and 50th ranked categories should not be considered effective to the same extent. Therefore, a metric such as correlation, which treats all mismatches uniformly, is less suitable.

To this end, in the following sections, we propose two novel evaluation metrics. The first, introduced in Sect. 4, mainly focuses on the ranking of categories and is based on the NDCG measure [59]. The second evaluation metric, presented in Sect. 5, relates additionally to the relevance scores from which the ranking is derived, and is based on the MRR measure [60]. The choice to consider two distinct metrics follows the fact that there are two main approaches that assess ranked lists in the IR literature. However, the two approaches also reflect a deeper difference in the core logic guiding the assessment. The NDCG-based metric focuses on quantifying how wrong the top-k predictions are. Each such prediction is punished in proportion to its deviation from the benign input prediction. On the other hand, the score of the MRR-based metric is only determined by the rank and confidence level of the correct top-1 category within the adversarial prediction list. In this sense, it is less concerned with how wrong the predictions are, and only focuses on how right the ranking of the true category is within the adversarial prediction. Hence, these novel metrics offer enhanced and diverse evaluation capabilities that can be utilized in various scenarios.

4 NDCG-based evaluation metric

Our first metric is based on Normalized Discounted Cumulative Gain (NDCG) [59]. We first describe NDCG and then continue to describe our metric. NDCG was originally proposed in the IR domain for the evaluation of the ranked list of results that a search engine outputs for a given query. It assumes that correct ranks at the top of the result list are more valuable than correct ranks at the bottom of the list. Thus, the evaluation score considers the relevance and the position of each element in the list. First, the Discounted Cumulative Gain (DCG) is computed. The relevance of each item is reduced as a function of its location in the list so that the gain of later items is progressively discounted. This reflects the fact that earlier search results are more likely to be explored than the recommendations at the bottom of the list, and are therefore more important. DCG is computed as

\[
DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log(1 + i)}
\]

where the parameter \( k \) is the number of the first items to be considered and the relevance \( rel_i \) is the predefined relevance of the item at rank \( i \), regardless of its location in the list. After the initial relevance of the \( k \) items are discounted by their position and accumulated, the DCG score is normalized to enable the comparison between different examples:

\[
NDCG_k = \frac{DCG_k}{IDCG_k}
\]

where IDCG is the ideally ordered list. For example, if a search engine outputs a list of 5 results, then the relevance score of each of these recommendations is given by their measured affinity to the original query. Assuming a ranking between 0-3, a ranked recommendation list could look like [2, 3, 3, 0, 1], whereas a perfect ordering of those recommended items would be [3, 3, 2, 1, 0]. The global NDCG score can also be itemized, where each item’s discounted gain is divided by its ideal counterpart, i.e., the discounted gain of the item at the same location in the perfect list. Note, therefore, that NDCG evaluates the ranked list of results, regardless of the numerical value of each result.

Similarly to recommendation systems, the output of NNs also consists of ordered lists of predicted classes, and our objective is to evaluate attacks and defenses based on their effect on the order of the model’s predictions. Therefore, customizing the NDCG measure requires the following considerations:

1. defining relevance scores
2. setting the value of \( k \)
3. defining ideal ranking for normalization

In the remainder of this section, the evaluation metric will be demonstrated on the evaluation of attacks, but it can be applied similarly to the evaluation of defenses.
4.1 Relevance score

Naturally, the relevance scoring of a model’s output should relate to the confidence of that model in its predictions. We propose to use the predictions of the model for a benign input as a source for relevance scores and as an ideal grading, by which attack and defense results should be normalized. The underlying assumption is that a well-trained model is not only capable of providing the correct top-1 prediction, but also of capturing the relationships between different categories. This ability is the essence of “learning”, which enables the model to generalize beyond the training data. Therefore, the entire prediction list of the model when given a benign input is considered here as a ground truth to determine the relevance scores.

4.2 Setting $k$

The parameter $k$ in the original NDCG formulation is static and rather arbitrary; its optimal value may vary between different datasets or applications, and the way to determine this optimal value is not always straightforward. In the following section, we explain why a static $k$ value is even less suitable for the adversarial setting, and introduce a more organic solution that considers the output probabilities distribution.

It is noteworthy to mention that $k$ may simply be set to the maximal value, namely the number of output categories. An NDCG score based on the full-length $k$ is possible and produces reasonable results. However, a confined $k$ is advantageous both computationally and conceptually. Computationally, it reduces the number of calculations needed for a final score. But, we find that the conceptual difference is the important one.

Conceptually, the threshold increases the sensitivity of the NDCG metric. As will be shown in this section, setting $k$ according to the probability distribution among the benign and adversarial prediction lists provides this effect. Consider two attacks on the same seed, which produce identical prediction lists with an identical ranking of all categories. One list contains $n$ categories with a confidence level that is higher than the threshold $k$. The second list contains the same categories, in the same rankings, but with only $m < n$ categories higher than $k$. Thus, although both lists contain the same rankings, they will not produce the same output and will not receive the same NDCG score.

In a classification task, any specific input results in a unique distribution of confidence among the entire list of output categories. In the case of ImageNet classification, though there are $n = 1000$ different output categories, the confidence of the model is usually distributed among a very small portion of the list, while assigning infinitesimal probabilities to most categories. Consequently, despite its potential length, the list of relevant predictions may be considerably shorter. Moreover, at the last layer of the NN a softmax function is
usually applied to the predictions list. This function assigns more weight to an even narrower range of categories, while the weight of all other categories decays exponentially.

Therefore, the exact length of the output lists may vary, especially in an adversarial setting where the NN may be ambiguous about the input due to an attack. This variance can reduce the effectiveness of a fixed \( k \), even when set to the optimal value. Thus, the parameter \( k \), which controls the length of the list used for evaluation, should be determined dynamically, according to the specific characteristics of each individual example. We, therefore, determine \( k \) using the post-softmax distribution of the confidence in order to preserve only the most relevant predictions as part of the scored list. This way narrower distributions, which reflect more confident decisions of the NN, result in shorter scored lists.

The relevance scores themselves are based on the normalized logits, which are the pre-softmax outputs of the model, in order to reflect the original confidence relations between the categories. Specifically, a set of post-softmax predictions \( \{ p_1^{(eb)}, p_2^{(eb)}, \ldots, p_n^{(eb)} \} \) for a specific benign example \( eb \) is scored using the corresponding pre-softmax values \( \{ p_1^{(eb)}, p_2^{(eb)}, \ldots, p_n^{(eb)} \} \) in the following way:

\[
R_{p_j}^{(eb)} = \begin{cases} 
\frac{i \gamma_{p_j}}{\sum_j \gamma_{p_j}} & i \leq k_1 \\
0 & \text{otherwise}
\end{cases}
\]

The post-softmax predictions \( \{ p_1^{(eb)}, p_2^{(eb)}, \ldots, p_n^{(eb)} \} \) are arranged in descending order, and \( k_1 = \max\{k \mid p_k^{(eb)} \geq \gamma_b\} \), where \( \gamma_b = [0, 1] \) is the prediction threshold parameter for benign inputs. For example, for \( \gamma_b = 0.01 \), only the \( k_1 \) highest post-softmax predictions that are above this 0.01 threshold will receive relevance scores in proportion to their original pre-softmax value, and all the other \( n - k \) predictions will be zeros. In the toy-poodle example in Table 1 (benign column) the predictions are: \( \{0.62, 0.37, 0.001, \ldots\} \). Setting \( \gamma_b = 0.01 \) results in \( k_1 = 2 \), and therefore only the two top-most predictions receive relevance scores. The benign relevance values are \( \{0.504, 0.495, 0, \ldots, 0\} \).

For an adversarial example \( e_a \) that was created on the basis of \( eb \), the relevance of the predictions is determined as follows:

\[
R_{p_j}^{(ea)} = \begin{cases} 
R_{p_j}^{(eb)} & i \leq k_2 \\
0 & \text{otherwise}
\end{cases}
\]

where \( k_2 = \max_j p_j^{(ea)} \geq \gamma_a \) with the threshold parameter for adversarial inputs: \( \gamma_a = [0, 1] \) and \( p_{1\text{match}} \) is the prediction within the \( eb \) output that corresponds to the category of \( p_1 \) (note that this prediction is not necessarily located in the \( i \)th place). In our running example for the FGSM attack \( e_a, 1 \), \( p_1 = 0.566 \) (‘miniature poodle’), \( p_{1\text{match}} = 0.37 \) (this is the prediction value assigned to ‘miniature poodle’ in the benign case), and \( R_{p_1}^{(ea)} = \frac{R_{p_1}^{(eb)}}{p_{1\text{match}}} = 0.495 \).

Note that \( \gamma_a \) differs from \( \gamma_b \), and should usually be lower since the original prediction values in the adversarial predictions list are commonly substantially lower. In the toy-poodle example of Table 1, setting \( \gamma_a = 0.001 \) results in \( k_2 = 613 \), and therefore all top-most 613 predictions receive the corresponding relevance scores of their benign counterparts. However, since \( k_1 = 2 \), only two of those relevance scores are non-zeros.

The number \( k \) of non-zero items within the prediction lists reflects the confidence level of the model for both the AE and the original seed. Suppose that during an attack, the original confidence distribution is disturbed and is now spread over more categories. In this case, the AE-scored prediction list can potentially assign relevance values to more categories. Nevertheless, if the attack caused the model to be uncertain whether the input image is a shark, broccoli, or a toy poodle, the extra values will still be zeros, since they receive the values of their benign counterparts. In the extreme case where an attack causes the model to predict entirely wrong categories with high confidence, the scored adversarial predictions list will be composed of zeros.

The discounted cumulative gain of the predictions for an image \( i \) is given by:

\[
\text{DCG}_k^{(e)} = \sum_{i=1}^{k} \frac{2^{R_{p_i}^{(e)}} - 1}{\log(1 + i)}
\]

Therefore, the more an attack ‘pushes’ a correct prediction down the predictions hierarchy, the more discounted is the gain of that prediction. See, for example, the difference between the DCG scores of FGSM attack[8] and CW attack [23] in Table 1, where in the latter case the scores are significantly more discounted due to their location in the ordered predictions-list. In other words, the DCG score ‘punishes’ predictions in proportion to their deviation from the ground-truth distribution of the benign input’s predictions.

### 4.3 Ideal ranking for normalization

We can now calculate the normalized DCG for an adversarial example \( e_a \). Following the logic of Eq. 2 we obtain:

\[
\text{NDCG}_{k_2}^{(ea)} = \frac{\text{DCG}_{k_2}^{(ea)}}{\text{DCG}_{k_1}^{(ea)}}
\]
As a consequence, the NDCG score of benign examples is always one, and attacks are considered more effective as this score decreases (accordingly, an effective defense increases the score). Another advantage of NDCG is that this metric assigns a score to each of the output categories. These scores can reflect different attack trends; if an attack only changes the top-1 category, then there is only a local drop in the score of the first prediction while the rest of the prediction list’s scores saturate. Alternatively, if an attack changes the entire semantic neighborhood of all the top predictions, then the decrease is significant across all the predicted classes. The last row of Table 1 presents examples of these two scenarios, where for the FGSM attack there is only a local decrease in the scores of the two first-ranked categories, while for the CW attack the initial decrease is much more dramatic and the increase of that score along the next categories is more gradual.

5 Defense reciprocal rank

This measure is inspired by the reciprocal rank measure [60], which is commonly used in information retrieval literature. We begin with explaining the original measure and why the fit is not straightforward, and then proceed to present our enhancements. Reciprocal rank calculates the multiplicative inverse of the rank position of a category within a given ordered list. If the benign category appears at the top-1 position, the reciprocal rank will be 1. If the benign category is at rank 2 then the reciprocal rank is 1/2, for rank 3 it is 1/3, and so on. The motivation behind this approach is simple—in an ordered list of predictions, the further away the benign category is ranked, the lower the score it receives. In our case, given an input image, the NN outputs a probability distribution over the output categories. When sorted in descending order, we can compute the reciprocal rank at which the benign category is predicted (i.e., the rank where the correct category label of the input image appears). As the output list can be very long, and therefore may result in negligible reciprocal rank scores, we restrict the measure to only consider the first top-k results of the ordered predictions list. This way, only if the true category is predicted within the top-k predictions, the score is proportional to its proximity to the top-1 location, and otherwise, the reciprocal rank is set to 0.

We enhanced the reciprocal rank so that it also considers the confidence of the model in its prediction. The intuition behind this is that a correct classification of an AE with higher confidence is better than a correct classification with lower confidence. In this case, the lower confidence may indicate that the attack, though failed to fully deceive the NN, gained some effect on the NN so that it can now be considered as more vulnerable. However, if the NN misclassifies an AE, then the confidence in the benign prediction should only improve the score within the bounds of the rank of that prediction. Namely, if the benign category of ‘toy-poodle’ is now located in rank 2, the score cannot surpass the score of ‘toy-poodle’ predicted at rank 1, regardless of the confidence level of the predictions.

Assuming all predictions are given as probabilities (i.e., sum up to one), we note that the possible range of confidence level is narrowed as the rank increases. The rank 1 prediction’s confidence may vary between zero and one. However, the rank 2 category can only be at confidence between zero and 0.5, while rank 3 prediction can only be between zero and third, etc. Therefore, we normalize the confidence of the prediction at each rank \(k\) to be between the maximal value of rank \(i - 1\) and the maximum value of rank \(i\). The defense reciprocal rank (DRR) score is thus given as:

\[
DRR_k = \begin{cases} 
\frac{\tilde{P}_i}{R_i+1} + \frac{1}{R_i+1} & i < k \\
0 & \text{otherwise}
\end{cases}
\]

where \(\tilde{P}_i\) is the probability assigned to the benign category \(i\), and \(R_i\) is the rank of that category within the ordered predictions list. This way, the first term of the DRR equation squeezes the original \(\tilde{P}_i\) value between zero and \(1/(R_i + 1)\), while the second term ensures that the lower bound of the new range will be \(1/(R_i + 1)\).

Back to the toy-poodle example, in the NN’s output to \(e_{a1}\) the benign category “toy-poodle” is located in the second place (rank 2) with probability 0.428. Therefore the DRR of \(e_{a1}\) score is calculated as \(0.428/3 + 1/3 = 0.476\). The DRR of \(e_{a2}\), however, is 0 since the benign category is not one of the top-5 predictions.

Usually, the probability distribution over the output categories is given as a softmax function. This nonlinear mapping is easier to interpret as a classification decision since it is closer to the argmax function while still remaining differentiable. However, to capture the relationships between the different categories, it is more suitable for our needs to use a linear normalization of the logits, i.e. the pre-softmax layer of the network. Though the results in this paper will only consist of this linear logits normalization, similar results are obtained when using the standard softmax probabilities as well.

6 Experimental evaluation

In our experiments, we applied the suggested new metrics (NDCG and DRR) to several attacks and defenses, and examined how the new scores reflect their diversity. For example, even when considering only successful attacks, which cause the NN to misclassify, the observed effect may vary between different attack types. In general, it is expected that targeted
attacks would have a deeper and more substantial effect than the untargeted attacks since they are not satisfied by merely misclassifying with the most approachable wrong category. Similarly, iterative attacks which are more computationally demanding are expected to be, on average, stronger than one-shot attacks which cannot refine the adversarial perturbations. While the traditional top-1 accuracy metric assigns an identical score to all these attacks, our suggested metrics enable a distinction between different degrees of misclassification.

6.1 Experimental setting

The experiments were performed on the ImageNet validation dataset [62] using the pre-trained VGG19 model [63], and on the CIFAR100 test dataset [64] using the pre-trained Swin Transformer model [65]. Unless explicitly stated, the presented results are those of the ImageNet dataset. The attacks were created using the Foolbox toolbox [66]. The chosen attacks reflect diverse approaches and techniques to create AEs. For example, targeted L-BFGS and CW attacks vs. untargeted attacks; boundary attack which is gradient-free AEs. For example, targeted LBFGS and CW attacks vs. iterative attacks like BIM [67], PGD, and DeepFool. The distortion level of all the attacks was bounded to create AEs which are indistinguishable to humans. The similar distortion allows a proper comparison between the different attack types. The average distortion of each attack group, measured as the $l_\infty$ distance between the AEs and their benign seeds, was in the range of [0.00185, 0.00187]. In order to investigate the net effect of the attacks, benign images which were misclassified by the model have been removed from the dataset. Similarly, failed attacks that did not cause the desired misclassification were not included in the attacks dataset. Therefore, the attacks datasets only include benign images, which were correctly classified by the model, and their adversarial counterparts, which cause the desired misclassification. An experimental group includes an average of 550 pairs of benign and adversarial examples for a specific attack. In all of the data presented below, we show the mean of each group and the standard error (SE). We created experimental group per each attack from the attack list mentioned above.

We also tested the effectiveness of defenses and considered three representative methods, where each method realizes a different defense strategy. The defenses have been applied to NN models under the above attacks in order to test the recovery of their original predictions. The first defense was adversarial training, which, as the name suggests, is based on retraining the NN using adversarial examples with the true labels. This way, the NN learns a more robust representation of the training set, by reducing the unmapped areas of its latent space. This method was found effective even against adaptive attacks, in which the attacker is aware of the defense method and intentionally tries to break it [37]. We retrained the base VGG19 model for 10 epochs, following the method proposed by Goodfellow et al. [8]. The second defense falls under the category of input transformation; in this approach, the adversarial perturbations are directly canceled by defensive perturbations. Specifically, we applied bit-depth squeezing to the AEs before inputting them into the NN. The squeezing was conducted under two experimental conditions, 4-bit depth and 5-bit depth, in order to examine the effect of the depth on the defense within the depth range recommended by Xu et al. [9]. The third defense strategy suggests injecting randomness into the NN computational process [68]. We implemented this approach via dropout of either 0.1 or 0.3 at the first fully connected layer of the NN. The effectiveness of the different defenses was tested under two conditions; when used against targeted attacks (CW and LBFGS) or against untargeted attacks (CW, boundary, and FGSM). The different combinations of attacks and defenses for each dataset are summarized in Table 2.

Figure 3 presents the NDCG scores for the ten top-most predictions, under five different attacks. It can be observed that there is a significant gap between the scores of the targeted and untargeted attacks. This gap indicates that targeted attacks are stronger; they cause top predictions to deviate from predictions ranked in close proximity to the benign top predictions. The effect of the untargeted attacks is mainly focused on the top-1 prediction ($\sim 0.55$ on average), which leads to top-10 NDCG scores of $\sim 0.8$ in such attacks. On the other hand, the targeted attacks manage to significantly reduce the score of the entire predictions list, with a continuous, yet moderate, recovery as the rank increases. This reflects the fact that untargeted attacks are weaker and tend to keep the top predictions within their original semantic neighborhood. Usually, these attacks only change the order of the top-most prediction or even only switch between the first and second predictions.

6.2 Evaluated metrics

While observing the trend of the NDCG scores of all top-$k$ predictions gives a better understanding of the adversarial effect, the most significant effect is found in the top-1 prediction. Moreover, since for most applications the main concern is the classification decision of the NN as reflected in the top-1 prediction and the rest of the predictions list is of less interest, in the following experiments we will only consider the NDCG score of the top-most rank. The NDCG$_1$ score encapsulates information about the deviation of the rank-1 prediction from the correct category, along with the confidence of the NN in this prediction.

For DRR results in the experiments, we set $k = 5$ following the top-5 measure in the literature as an alternative
Table 2 Experimental configurations

| Dataset    | Model           | Attacks           | Defenses             |
|------------|-----------------|-------------------|----------------------|
| ImageNet   | VGG19           | LBFGS             | CW                   |
|            |                 | Boundary attack   | FGSM                 |
|            |                 |                   | PGD                  |
|            |                 |                   | DeepFool             |
| CIFAR100   | Swin transformer| BIM               | Bit-depth squeezing  |
|            |                 | NewtonFool        | Drops               |
|            |                 |                   | Adversarial training |

Fig. 3 NDCG scores of the top-10 predictions for different attacks

Fig. 4 Evaluation of attacks using different metrics

to the top-1 measure. This parameter seems to satisfy the demands we set for the new desired metric, though a smaller \( k \) will produce a stricter evaluation and lower scores, similar to the NDCG\( _1 \) measure. Overall, the DRR\( _5 \) measure can correctly identify cases in which the true category of the input is present as one of the top-5 predictions. This is more common in weak attacks that cause the NN a minor miss-classification. The NDCG\( _1 \) measure, on the other hand, does not consider the rank of the true category but rather the similarity between the actual top-1 prediction and the true prediction. In addition to our suggested metrics, the experimental results also present the traditional top-1 and top-5 accuracy measures.

In the next experiment we examine the metrics’ ability to detect:

- Attack-attack separation—the distinction between attacks in terms of their quality, e.g., strong and weak attacks.
- Attack-benign separation—the distinction between benign inputs and adversarial inputs.

6.3 Results

Figure 4 compares between the different evaluation metrics of the above attacks. The top-1 and top-5 metrics illuminate a trade-off between attack-attack separation and attack-benign separation. The top-1 metric perfectly captures attack-benign separation because the dataset consists only of successful attacks. However, it fails in attack-attack separation. Another view of the trade-off is witnessed in the top-5 metric—it nicely captures attack-attack separation, however, it fails in attack-benign separation for weak attacks. This begs the question of how other top-\( k \) metrics (such as top-2 or top-3) would perform. They may display a better trade-off between attack-attack separation and attack-benign separation, however, this will require consistent manual tuning of \( k \), depending on the specific characteristics of the tested attacks. Another shortcoming of all top-\( k \) measures is that they are binary, and are therefore inherently less informative.

Our two suggested evaluation metrics successfully balance the trade-off between attack-attack separation and attack-benign separation, as they both differentiate attacks from benign inputs and also differentiate between attack types.

The gap between the NDCG\( _1 \) scores and the DRR\( _5 \) scores indicates that in many cases, the top-1 prediction can be wrong, or even very wrong, but the true category may still be found as one of the top-5 predictions as well. An example of such a case is found in the targeted CW and LBFGS attacks. On the other hand, the PGD attack results demonstrate the opposite effect, where the “wrongness” of the topmost predicted category is not as acute as in the targeted attacks,
however, the rank of the correct category is significantly lowered. The gap between the two scores is also affected by the inclusion of the confidence in the DRR score calculation; when the defense leads to a correct top-1 prediction, the NDCG assigns the maximal score, while the DRR may sometimes assign a reduced score depending on the decrease of the NN’s confidence in its prediction.

The PGD attack effect, where the NDCG score is rather high while the DRR score is low, is also evidenced in the experimental results for the CIFAR100 dataset presented in Fig. 5.

Applying different defenses to the attacked NN demands a more complicated evaluation. Figures 6 and 7 show defenses analysis for the targeted and untargeted attacks, respectively. The ‘no defense’ condition is a mean of the relevant conditions presented in Fig. 4. When comparing targeted and untargeted attacks (Figs. 6 vs. 7), it can be observed that the dropout defense is significantly less effective for the targeted attacks, while the bit-depth squeezing and adversarial-training defenses are less sensitive to the attack type. The top-1 measure, which was useless when comparing different attacks, can now actually produce a good distinction between different defenses.

Figure 7 shows that the bit-depth squeezing and adversarial-training defenses recover more than 50% of the classification decisions, while the dropout defense is only successful for about 20% of the AEs. When considering targeted attacks (see Fig. 6), the gap between the above defenses is even wider, as the dropout defense recovers less than 10% of the classification decisions. Contrary to that, the top-5 measure fails to differentiate some defenses from the ‘no defense’ condition and overall assigns rather similar scores to all the experimental conditions (Fig. 7). A similar trend appears in the CIFAR100 dataset (Fig. 8). A detailed analysis of the
different defenses, when used against each attack type individually, can be found in Figs. 9 and 10.

Our two new suggested metrics again follow the more informative trend and show that the bit-depth squeezing and adversarial-training defenses are stronger than the dropout defense under the current setting and parameters. However, the bit-depth defense comes at a cost of a greater decrease in the accuracy of the NN in benign settings; the reduced information cancels much of the adversarial perturbations, but it also includes a loss of valuable information that is necessary for correct classification. The same trade-off between benign and adversarial accuracy also applies to the dropout defense, however, its analysis is more complicated due to the possibility of different combinations of dropout rates at different layers of the NN. Adversarial training, on the other hand, does not inherently suffer from such drawback;
under the right setting, it was found to improve both the robustness and the generalization of NNs [69]. However, its implementation requires more computational resources and meta-parameter tuning. Under our experimental setting, we observed a decrease of the top-1 accuracy for benign examples when using the different defenses, as shown in Table 3. While the decrease under the bit-depth squeezing is inevitable, better parameter tuning may reduce the (already limited) accuracy decrease under the other defenses.

### 7 Discussion and conclusions

As research constantly improves defense and attack methods, the need to adequately compare and evaluate them arises. In this paper, we introduce two novel metrics, NDCG and DRR, for the comparison of attacks/defenses for multi-class pattern recognition. Each such new metric encapsulates information about the rank and confidence of the output categories into one elegant measure.

Our suggested metrics are particularly valuable in cases where there are many categories with partially overlapping features; for example, different types of dogs or vehicles. When the categories are distinctive (without any overlapping features), all successful attacks must be strong enough to convert the image label from one category to another. When the categories overlap, as for example is the case with different types of shotguns, weak attacks are likely to cause misclassification with a category of the most obtainable adjacent decision boundary (e.g., a shotgun of a different type, as opposed to a specific animal or landscape). However, stronger attacks are able to deeply disturb the output of the NN in a way that will require more effort to recover (e.g., all top-ranked predicted categories now are of animals and landscapes, while shotguns receive very low probabilities). This distinction between attacks in terms of their strength may have significant implications, especially when evaluating defenses against attacks. If the quality of the attacks is neglected, the ensemble of attacks on which the defense evaluation is based may be biased, e.g., include only weak attacks. For example, evaluating the defense of an airport image recognition system [70] relying on weak attacks may give a false impression of protection; strong attacks may still cause the system to classify the shotgun as a completely different category, even though weaker attacks will not succeed to switch the classification decision from one type of shotgun to another. Of course, labeling attacks as weak or strong is neither binary nor discrete, and the effect of an attack on the NN output may have multidimensional aspects. However, the currently used metrics for attack evaluation are either very limited (e.g., the accuracy metric) or indirect (e.g., measuring the computational effort to create the attack or the distortion they require). Our suggested metrics display higher sensitivity and informativeness; for example, they can detect the effect of failed attacks. At the same time, they still comply with prior expectations of stronger or weaker attacks, i.e., attacks that are considered strong or weak are detected as such by the metrics. As a consequence, the proposed metrics enable us to better quantify the complicated effect of attacks on the NN output.

In order to capture as much of the above effect as possible, we based our metrics on measurements of ranked results, which originated in the field of information retrieval. As opposed to correlation, these metrics consider the rank of each item in the list when measuring the similarity between two lists. This feature is important in the context of attacks and defenses, since the top-ranked results of a NN are usually much more relevant than the rest of the list, and should therefore be weighted accordingly. Measuring the relevance of categories with natural language processing techniques, such as word2vec, is also insufficient, since these techniques only consider the general semantic similarity between categories, without the unique (and in the case of image classification, visual) context of the category in each individual input. Therefore, our metrics are based on the comparison between the original predictions list in the benign setting and the predictions list in case of attack/defense. As such, they realize an oracle approach, in which the evaluation is performed while having the full knowledge of the attack or defense. Therefore, the metrics cannot be used for risk evaluation of a NN against unknown attacks. Rather, our goal is to enable an adequate comparison between different types of attacks or defenses.

As opposed to traditional ranked predictions, the items of the NNs’ prediction list are not only ranked but also weighted with a probability value, which can be interpreted as the confidence level of the NN in the prediction. We therefore formulated the new metrics to also account for these output values, which carry additional information about the changes in the functionality of the NN as a result of the attack/defense. Our evaluation shows that the new metrics are able to detect both attack-attack separation and attack-benign separation, contrary to the commonly used accuracy metrics.

We demonstrated the utility of our suggested metrics in the domain of image recognition, but they can be used in any multi-class settings, such as audio, finance, or medical data. Even though the two metrics aim for the same evaluation objective, they differ in their underlying reasoning. Therefore, they can be considered complementary, and future work
may include a unified metric that will combine their two perspectives into one general approach.

Acknowledgements This work was supported by the Ariel Cyber Innovation Center in conjunction with the Israel National Cyber Directorate in the Prime Minister’s Office.

Author Contributions All authors contributed to the paper equally.

Funding The authors have no relevant financial or non-financial interests to disclose. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript. The authors have no financial or proprietary interests in any material discussed in this article.

Data availability Data will be made available after publication.

Declarations Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Human and animal rights The research does not involve human participants or animals.

Informed consent Authors provide their informed consent.

References

1. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks, arXiv:1312.6199 (2013)
2. Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: high confidence predictions for unrecognizable images. In: CVPR, pp. 427–436 (2015)
3. Akhtar, N., Mian, A.: Threat of adversarial attacks on deep learning in computer vision: a survey. IEEE Access 6, 14410–14430 (2018)
4. Papernot, N., McDaniel, P., Wu, X., Jha, S., Swami, A.: Distillation as a defense to adversarial perturbations against deep neural networks. In: SP, pp. 582–597 IEEE (2016)
5. Athalye, A., Engstrom, L., Ilyas, A., Kwok, K.: Synthesizing robust adversarial examples. In: ICML, pp. 284–293 (2018)
6. Rosenberg, I., Shabtai, A., Elovici, Y., Rokach, L.: Adversarial machine learning attacks and defense methods in the cyber security domain. ACM Comput. Surv. (CSUR) 54(5), 1 (2021)
7. Evtimov, I., Eykholt, K., Fernandes, E., Kohno, T., Prakash, B., Li, A., Rahmati, A., Song, D.: Robust physical-world attacks on machine learning models 2(3), 4 (2017), arXiv:1707.08945
8. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv:1412.6572 (2014)
9. Xu, W., Evans, D., Qi, Y.: Feature squeezing: detecting adversarial examples in deep neural networks. In: NDSS (2017)
10. Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., Madry, A.: Adversarial examples are not bugs, they are features. arXiv:1905.02175 (2019)
11. Hossin, M., Sulaiman, M.: A review on evaluation metrics for data classification evaluations. Int. J. Data Min. Knowl. Manag. Process 5(2), 1 (2015)
12. MacKay, D.J., MacKay, D.J.: Information Theory, Inference and Learning Algorithms. Cambridge University Press, Cambridge (2003)
13. Wallach, H.: Cavendish Lab., University Cambridge, Cambridge, online available at http://www.inference.phy.cam.ac.uk/hmw26/. Technical Report (2006)
14. Huang, J., Ling, C.X.: Constructing new and better evaluation measures for machine learning. In: IJCAI, pp. 859–864 (2007)
15. Arp, D., Quiring, E., Pendlebury, F., Warnecke, A., Pierazzi, F., Wressnegger, C., Cavallaro, L., Rieck, K.: Dos and don’ts of machine learning in computer security, arXiv preprint arXiv:2010.09470 (2020)
16. Fursov, I., Morozov, M., Kaploukhaya, N., Kevtun, E., Riveracastro, R., Gusev, G., Babaev, D., Kireev, I., Zaytsev, A., Burnea, E.: Adversarial attacks on deep models for financial transaction records, arXiv preprint arXiv:2106.08361 (2021)
17. Berman, D.S., Buzzak, A.L., Chavis, J.S., Corbett, C.L.: A survey of deep learning methods for cyber security. Information 10(4), 122 (2019)
18. Carlini, N., Wagner, D.: Audio adversarial examples: targeted attacks on speech-to-text. In: 2018 IEEE Security and Privacy Workshops (SPW), pp. 1–7. IEEE (2018)
19. Finlayson, S.G., Bowers, J.D., Ito, J., Zittrain, J.L., Beam, A.L., Kohane, I.S.: Adversarial attacks on medical machine learning. Science 363(6433), 1287 (2019)
20. Chan, R.H., Ho, C.W., Nikolova, M.: Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. IEEE Trans. Image Process. 14(10), 1479 (2005)
21. Vidnerová, P., Neruda, R.: Vulnerability of classifiers to evolutionary generated adversarial examples. Neural Netw. 127, 168 (2020)
22. Moosavi-Dezfooli, S.M., Fawzi, A., Frossard, P.: 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)
23. Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks. In: SP, IEEE, pp. 39–57 (2017)
24. Mopuri, K.R., Garg, U., Babu, R.V.: Fast feature fool: a data independent approach to universal adversarial perturbations, arXiv preprint arXiv:1707.05572 (2017)
25. Zhang, X., Wang, N., Ji, S., Shen, H., Wang, T.: Interpretable deep learning under fire, arXiv:1812.00891 (2018)
26. Ghorbani, A., Abid, A., Zou, J.: Interpretation of neural networks is fragile. In: AAAI, vol. 33, pp. 3681–3688 (2019)
27. Heo, J., Joo, S., Moon, T.: Fooling neural network interpretations via adversarial model manipulation, arXiv:1902.02041 (2019)
28. Yang, C., Wu, Q., Li, H., Chen, Y.: Generative poisoning attack method against neural networks, arXiv preprint arXiv:1703.01340 (2017)
29. Shafahi, A., Huang, W.R.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
30. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
31. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
32. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
33. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
34. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
35. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
36. Tabacof, P., Vallee, E.: Studer, C., Feizi, S., Goldstein, T.: Are adversarial examples inevitable? arXiv:1809.02104 (2018)
37. Athalye, A., Carlini, N., Wagner, D.: Obfuscated gradients give a false sense of security: circumventing defenses to adversarial examples. In: ICML, pp. 274–283 (2018)
38. Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A.: On evaluating adversarial robustness, arXiv:1902.06705 (2019)
39. Tjeng, V., Xiao, K., Tedrake, R.: Evaluating robustness of neural networks with mixed integer programming, arXiv preprint arXiv:1711.07356 (2017)
40. Wong, E., Kolter, Z.: Provable defenses against adversarial examples via the convex outer adversarial polytope. In: International Conference on Machine Learning (PMLR, 2018), pp. 5286–5295 (2018)
41. Lecuyer, M., Atlidakis, V., Geambas, R., Hsu, D., Jana, S.: Certified robustness to adversarial examples with differential privacy. In: 2019 IEEE Symposium on Security and Privacy (SP), pp. 656–672. IEEE (2019)
42. Bastani, O., Ioannou, Y., Lamproupolos, L., Vytiniotis, D., Nori, A., Criminisi, A.: Measuring neural net robustness with constraints, arXiv preprint arXiv:1605.07262 (2016)
43. Carlini, N., Katz, G., Barrett, C.W., Dill, D.L.: Ground-truth adversarial examples, arXiv preprint arXiv:1709.10207 (2017)
44. Chen, J., Wang, Z., Zheng, H., Xiao, J., Ming, Z.: ROBY: evaluating the robustness of a deep model by its decision boundaries, arXiv preprint arXiv:2012.10282 (2020)
45. Peng, Y., Zhao, W., Cui, W., Su, J., Han, B., Liu, Q.: Evaluating deep learning for image classification in adversarial environments. IEICE Trans. Inf. Syst. 103(4), 825 (2020)
46. Powers, D.M.: Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation, arXiv preprint arXiv:2010.16061 (2020)
47. Liu, T.Y.: Learning to Rank for Information Retrieval. Springer, Berlin (2011)
48. Dehghani, M., Zamani, H., Severyn, A., Kamps, J., Croft, W.B.: Neural ranking models with weak supervision. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (2017), pp. 65–74
49. Guo, J., Fan, Y., Pang, L., Yang, L., Ai, Q., Zamani, H., Wu, C., Croft, W.B., Cheng, X.: A deep look into neural ranking models for information retrieval. Inf. Process. Manag. 57(6), 102067 (2020)
50. Salakhutdinov, R., Hinton, G.: Semantic hashing. Int. J. Approx. Reas. 50(7), 969 (2009)
51. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
52. Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.: Learning deep structured semantic models for web search using clickthrough data. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, pp. 2333–2338 (2013)
53. Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: Advances in Neural Information Processing Systems, pp. 2042–2050 (2014)
54. Qin, T., Liu, T.Y., Li, H.: A general approximation framework for direct optimization of information retrieval measures. Inf. Retrieval 13(4), 375 (2010)
55. Bruch, S., Zoghi, M., Bendersky, M., Najork, M.: Revisiting approximate metric optimization in the age of deep neural networks. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1241–1244 (2019)
56. Taylor, M., Guiver, J., Robertson, S., Minka, T.: Softrank: optimizing non-smooth rank metrics. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, pp. 77–86 (2008)
57. Wu, Q., Burges, C.J., Svore, K.M., Gao, J.: Adapting boosting for information retrieval measures. Inf. Retrieval 13(3), 254 (2010)
58. Järvelin, K., Kekäläinen, J.: IR evaluation methods for retrieving highly relevant documents. In: ACM SIGIR Forum, vol. 51, pp. 243–250. ACM, New York (2017)
59. Chapelle, O., Metzler, D., Zhang, Y., Grinspan, P.: Expected reciprocal rank for graded relevance. In: Proceedings of the 18th ACM conference on Information and knowledge management (2009), pp. 621–630
60. Croft, W.B., Metzler, D., Strohman, T.: Search Engines Information Retrieval in Practice, vol. 520. Addison-Wesley, Reading (2010)
61. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
62. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014)
63. Krizhevsky, A., Hinton, G., et al.: Learning Multiple Layers of Features from Tiny Images. Ph.D. thesis, University of Toronto (2009)
64. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)
65. Rauber, J., Brendel, W., Bethge, M.: Foolbox: A Python toolbox to benchmark the robustness of machine learning models, arXiv:1707.04131 (2017)
66. Kurakin, A., Goodfellow, I.J., Bengio, S.: Adversarial examples in the physical world, CoRR abs/1607.02533. http://arxiv.org/abs/1607.02533 (2016)
67. Carlini, N., Wagner, D.: Adversarial examples are not easily detected: bypassing ten detection methods. In: ACM Workshop on Artificial Intelligence and Security, pp. 3–14. ACM (2017)
68. Wu, Y., Bamman, D., Russell, S.: Adversarial training for relation extraction. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1778–1783 (2017)
69. Gota, D.L., Puscasu, A., Fanca, A., Valean, H., Miclea, L.: Threat objects detection in airport using machine learning. In: 2020 21th International Carpathian Control Conference (ICCC), pp. 1–6. IEEE (2020)