This is a post-peer-review, pre-copyedit version of an article published in Artificial Intelligence and Soft Computing. ICAISC 2020. The final authenticated version is available online at: https://doi.org/10.1007/978-3-030-61401-0_14

Postprint of: Klawikowska Z., Mikołajczyk A., Grochowski M. (2020) Explainable AI for Inspecting Adversarial Attacks on Deep Neural Networks. In: Rutkowski L., Scherer R., Korytkowski M., Pedrycz W., Tadeusiewicz R., Zurada J.M. (eds) Artificial Intelligence and Soft Computing. ICAISC 2020. Lecture Notes in Computer Science, vol 12415. Springer,
DOI: 10.1007/978-3-030-61401-0_14

Explainable AI for Inspecting Adversarial Attacks on Deep Neural Networks

Zuzanna Klawikowska[1], Agnieszka Mikołajczyk[1] and Michał Grochowski[1]

[1] Gdańsk University of Technology, Gabriela Narutowicza 11/12, 80-233 Gdańsk, Poland
zklawikowska97@gmail.com; agnieszka.mikolajczyk@pg.edu.pl;
michal.grochowski@pg.edu.pl

Abstract. Deep Neural Networks (DNN) are state of the art algorithms for image classification. Although significant achievements and perspectives, deep neural networks and accompanying learning algorithms have some important challenges to tackle. However, it appears that it is relatively easy to attack and fool with well-designed input samples called adversarial examples. Adversarial perturbations are unnoticeable for humans. Such attacks are a severe threat to the development of these systems in critical applications, such as medical or military systems. Hence, it is necessary to develop methods of counteracting these attacks. These methods are called defense strategies and aim at increasing the neural model's robustness against adversarial attacks. In this paper, we reviewed the recent findings in adversarial attacks and defense strategies. We also analyzed the effects of attacks and defense strategies applied, using the local and global analyzing methods from the family of explainable artificial intelligence.

Keywords: Deep Neural Networks, Explainable Artificial Intelligence, Adversarial Attacks, Convolutional Neural Networks.

1 Introduction

These days deep learning is the fastest-growing field in the field of image analysis and classification. Deep Neural Networks (DNN) are considered state of the art algorithms for image classification, [1], [2]. Despite great achievements and perspectives, deep neural networks and accompanying learning algorithms have some crucial challenges to tackle [3], [4]. DNNs are data-hungry [5], [6], it is challenging to select the optimal network structure [7], [8], and to understand neural networks reasoning process [9]. Another essential concern is the subject of attacks on neural networks. Regarding image classification, neural model designing is based on learning a given structure from data and then analyzing the input-output relation, on the pixel-level. This makes it relatively easy to cause such a system to perform incorrectly by changing these values so that these changes are not noticeable to the system user. Such fragile black-box deep neural network models are used to solve very sensitive and critical tasks. Modification of the input pixel that causes system malfunction is called an adversarial attack. The adversarial attack is carefully selected and is a severe threat to the development of these systems in critical-safety applications, such as medicine, military, and even urban scene

recognition used in autonomous vehicles. To ensure the safety of this technology and make it widely usable, it is necessary to develop methods of detecting, understanding, and counteracting these attacks.

To address those challenges, we employ the methods of Explainable Artificial Intelligence (XAI) which have a wide range of tools that can be used to tackle mentioned problems, in particular in detecting and identifying the types of attacks. Knowing the kind of attack, we can more easily counteract it. Following [10] we review and explain the recent findings in adversarial attacks, including white-box and black-box attacks, targeted and non-targeted attacks as well as one-time and iterative attacks. In the case study, we test the white-box Fast Gradient Sign Method (FGSM) and black box One-pixel attack.

We applied XAI local and global explanations methods to analyze the attacks. The local analysis aims to explain a single prediction of a model, e.g. one input image. In contrast, the global one tries to explain how the whole model works in general, i.e. it shows how a particular machine learning model analyzes a given set of data. In the paper, we have shown how selected attacks affect the process of classification of individual images, and how this process looks globally, i.e. we try to conduct a qualitative analysis of the features of data sets that are more and less vulnerable to attacks. We use Layer-wise Relevance Propagation (LRP) method for the local analysis and XAI signatures supported by Uniform Manifold Approximation and Projection (UMAP) for a global one. Finally, we describe the most popular methods that aim at increasing neural model's robustness against adversarial attacks. We implement an Adversarial retraining approach to investigate the DNN robustifying process.

2 Adversarial attacks and defenses

Most often, the adversarial attacks are targeted against AI-based computer vision systems. Those systems are mainly based on deep neural models trained on raw data, which capture the input-output relations, on the pixel-level. It turns out that it is relatively easy to fool such systems to perform incorrectly, just by carefully modifying those pixels in a way that these changes are not visible to the unaided eye of the average system user. One of the reasons is that even if the network classifies input data correctly, it does not understand its meaning in the same way as a human. Therefore, classification is not always made based on the relevant premises. A common situation is when the network has correctly classified a given input but based on inappropriate premises. Such model behavior cannot be regarded as correct. Szegedy et al. [11] showed that an unnoticeable change of pixels in the input image can completely change the label assigned to it earlier. The modified images have been called adversarial examples, and the process that aims to mislead the neural network has been called an adversarial attack. In [12], it was demonstrated that the system could recognize with 99.99% accuracy objects in the input image. Other examples of adversarial attacks are also widely reported, e.g. failing to acknowledge the STOP sign by traffic sign recognition system [13], failing to recognize pedestrians by the scene segmentation system [14], changing medical diagnosis [15].

2.1 Adversarial attacks

Following [10], we present taxonomy related to adversarial attacks. In terms of the attacker's knowledge of the system, attacks might be divided into a white-box and a black-box. In terms of the aim of the attack on targeted and untargeted.

Adversary's Knowledge.

White-box attacks. In this type of attack, attackers know all elements related to a system, i.e. training set, model's architecture, hyper-parameters, number of layers, activations, and model weights. One of the methods is FGSM (Fast Gradient Sign Method) [10]. The idea is to modify the input e.g. image, in such a way that added perturbation is consistent with the gradient sign obtained in the process of backpropagation. In the most common case of images, the generated perturbation might look similar to a color noise. The magnitude of the perturbation can be weighted by the gain factor. When it is small, the change in the output image is not noticeable. As it increases, the change is more and more visible. Therefore, during the generation, an iterative approach is adopted to allow for a gradual increase in perturbation until the label of a class recognized by the model changes. The effect of the one-time version of the FGSM algorithm is shown in Fig. 1. In this type of attack, the number of modified pixels is limited only to the image size.

Black-box attacks. On the contrary to white-box attacks, here attackers have no knowledge about attacked neural networks, except the outputs of the neural model. An example of such an attack is the so-called one-pixel attack [10]. It consists of changing one pixel of an input image to obtain another DNN prediction. The generation of adversarial examples is realized by solving a constrained optimization problem. The task is to find the optimal perturbation, namely to change just one pixel, causing a change of class by DNN. In this method, the differential evolution algorithm is used. It consists of comparing successive, generated children (in the form of pixels) with corresponding parents in each iteration, to check if the value of the class activation function has been increased. The effect of the one-pixel attack algorithm is shown in Fig. 1.

Adversarial Specificity.

Targeted attacks. These involve confusing the neural network by changing the input image in such a way that the input image is assigned a specific, defined class [10]. This type of attack is usually used in the problem of multi-class classification, and it aims to ensure that all attacked input images are classified as one class. In the case of a two-class classification task, a targeted attack becomes a non-targeted attack.

Non-targeted attacks. The goal of *undirected attacks* is to change the input in such a way that the predicted class also changes [10]. A real-world example of a non-targeted attack is placing the sticker on a road STOP that will make an autonomous car recognize

it as another sign. The generation of adverse examples for this type of attack usually takes place twofold. The first one consists of a series of targeted attacks and then selecting the least modified image. The second consists in minimizing the probability of obtaining the correct label.

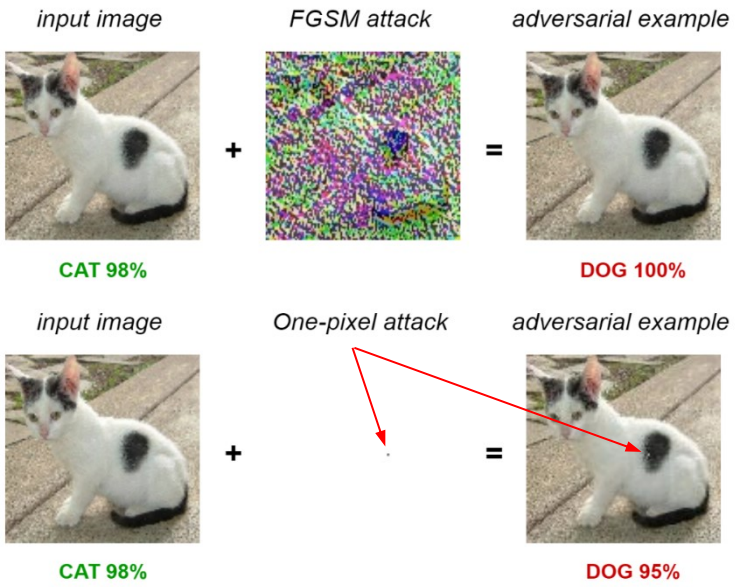

Fig. 1. Illustration of the FGSM and one-pixel attacks.

Attack Frequency. One-time attacks take a single trial to generate adversarial examples [10]. On the contrary, *Iterative attacks* take multiple trials to produce optimal adversarial examples. The latter have higher efficiency, but due to the large computation effort, they can rarely be used in real-time applications.

Adversarial Falsification. This category distinguishes attacks between *False positives* and *False-negatives*. The former generate hostile examples that are misclassified as positive ones, while the latter generates adverse examples that are misclassified as negative ones (Type I and Type II errors, respectively).

2.2 Countermeasures for adversarial attacks

Quick emergence of attacks, begun a new category of methods appear: *adversarial defense strategies*. These strategies can be divided into two categories: *reactive* and *proactive*. Reactive defenses focus on the detection of adversarial examples after the deep neural network has already been built, while the proactive strategies aim to increase the resistance (robustness) of the network to attacks by proper design of the training process or model's architecture.

Reactive. Reactive methods are used to defend an already trained neural network. One of the main methods is to prepare an additional model to check if an attack modified the input image. The second approach uses denoising autoencoder acting as a dedicated filter that removes adversarial perturbations from the input.

Adversarial detecting. In this method, an auxiliary binary classifier is designed to recognize whether the input image is an adversarial example. This model uses the solutions provided by probability theories, such as Bayes's theorem [16]. Bayesian networks use a set of neural networks, in which each network is described by a parameter vector w (weights and bias). This set enables finding the probability density for the entire weights space, instead of choosing a set of weights describing a single neural network. The final output from the Bayesian network is obtained by calculating the average of the outputs from all the created networks. It has been proven that the uncertainty obtained by the Bayesian network classifier is higher in the case of adverse examples than in the case of clean images, thus making it possible to identify such examples even before they are fed into the neural network. There are many other approaches in this category, details can be found in [10].

Input reconstruction. The method employs an autoencoder acting as a dedicated filter, to eliminate from the adverse example, intentionally introduced perturbations. After such input image transformation, it no longer significantly affects the prediction of the network. One of the tools allowing to recreate a perturbation-free image is Denoising autoencoder or its improved version - Deep Contractive Autoencoder [17]. The objective function of this network is to minimize the difference between the input image and the target one. As a result of learning, its outputs become the reconstructed version of the input. Because the middle layer has fewer neurons than the input and output layers, hence the latent representation is smaller, the network learns the most important features of the image so that the output image is as close as possible to the input image. Because of that AE is often used for dimensionality or noise reduction. In this case, it ensures that the network through training will learn the input features that are not related to perturbations, so that the original input image will be correctly restored.

Proactive. Proactive strategies involve training the neural network to increase its robustness against adverse attacks. The main defense strategy is to train the network using adversarial examples to increase its robustness. Another approach is network distillation which consists of creating several neural networks and transferring knowledge between them so that the final network becomes less sensitive to small changes in the image.

Adversarial (Re)training. The method is relatively simple yet effective. It is based on complementing the training set with adverse examples, and then, retraining the network. Network retraining consists in transferring the weights of the previously trained network and starting the training with such initial weights. By adding a set with adverse input to the training set, the network is forced to recognize also adverse examples as

belonging to the correct class [18]. The robustness of the network against a given type of attack depends on whether it was included in the training stage.

Knowledge Distillation. Knowledge distillation [19] method was developed to reduce the size of the network by transferring knowledge from so-called teacher (larger) network to a student (smaller) one while maintaining its accuracy. In application to defense against adverse examples, this method aims to reduce the sensitivity of the network to small input perturbations.

Probability obtained at the output of a teacher network by using the softmax function becomes the soft-labels for the inputs of a student network. The method takes advantage of the fact that the knowledge acquired by the deep neural network during training is contained not only in the weights and thresholds but also in the probability vectors generated by the network. The teacher network uses the vector of similar probabilities as a set of new labels for images from the original set. In this vector, the information about the similarity between the classes is stored. The student network is trained on an original training set, but with labels generated by the teacher network. The student network achieves a similar predictive efficiency as the original model but provides lower calculation costs. By training the network on labels with coded similarities between the classes, it prevents overtraining. This also results in a better generalization of the network around the training points, hence the network becomes more robust to small image disturbances.

For the descriptions of other approaches to counteract the attacks, like: Classifier robustifying, Network verification, Ensembling defenses we refer the readers to [10]. Unfortunately, all defense strategies make the network robust only to a certain extent and to specific types of attacks. Additionally, these strategies do not work on new attacks. Moreover, it is unknown what type of attack will be used against a given neural network, making it extremely difficult to choose the right strategy [20].

3 XAI methods of inspecting adversarial attacks

Explainable Artificial Intelligence is a field of artificial intelligence that focuses on research and development of tools that allow the user to understand the decisions made by complex black-box ML models. With such tools, the user can better understand the premises underlying the model's decisions by trusting their credibility and accepting them. In case when XAI analyses indicate that the inference process is not in line with reality, the AI system can be improved, which contributes to the development of these models. In addition, recent papers report that XAI allows the detection and elimination of data biases that affect model performance [9]. Understanding how different attacks affect the model allows one to choose the right ways to increase its robustness. XAI methods can be divided into local and global explanations. Local analysis tries to explain a single prediction of a model, while the global approach shows the global model behavior. In this paper, we applied XAI methods to the analysis of the attacks and their influence on the neural model's predictions.

3.1 Local explanations

There are several approaches realizing the idea of local explanations SHAP [21], LRP [22], LIME [23], Anchors [24]. In the paper, we decided to apply a method that allows generating visual explanation in the form of attention maps. In case of problems related to image analysis, the attention map visualizes how important each pixel in the input image is for the final DNN prediction. There is a wide range of approaches under this type of analysis e.g. Gradient SmoothGrad [25], DeConvNet [26], Guided Backpropagation [27], Input*Gradient [28]. After initial testing, for further analysis, we employed the Layer-wise Relevance Propagation method (LRP), more precisely its variant LRP present A flat. The LRP explains a neural network decision by decomposing the output of the model (prediction) at the pixel-level. The goal is to find the relevance scores of each input image pixels to a corresponding prediction, positive or negative. Those relevance sores might be visualized in the form of heatmaps. Simplifying the matter, heat maps indicate the pixels (areas) of the input image that affect the corresponding output value of the model most and least. A detailed description of the method can be found e.g. in [29]. The effect of employing this method to analysis of Convolution DNN model applied to image classification problem is shown in Fig. 2. In the upper row, one can find original images, and in the lower row the corresponding heatmaps. The red color indicates the pixels that most strongly correspond to the model output, while the blue color indicates the pixels that have the least influence on the model output. Such a visual analysis enables the end-user, or the developer of the model/system, to verify its correctness.

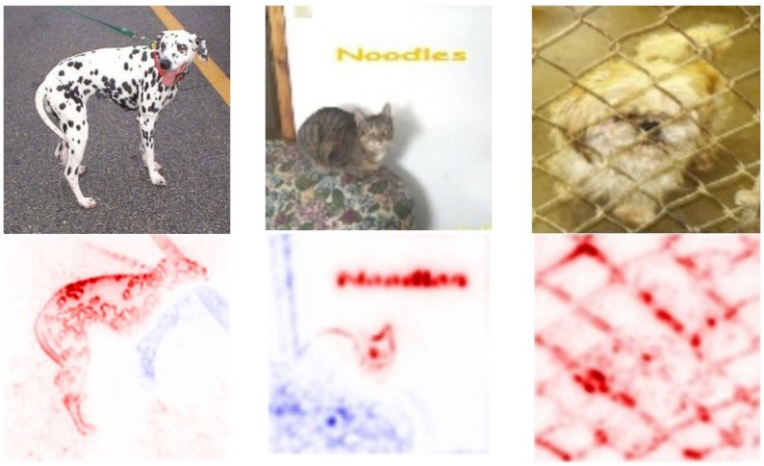

Fig. 2. Examples of attention maps (lower) generated by LRP present A flat.

In the case of the first image, we do not doubt that the model relies on rational premises when it generates the decision (dog's shape). In the next case, the situation is not so obvious anymore. The model takes into account the cat, but it focuses mostly on the text placed on the picture. Such behavior of the model, despite its correct classification, makes the trustworthiness of the model questionable. Analyzing the heatmaps of the

third image, we can clearly see that the network pays attention only to the bars behind which the dog is placed. The reason for this obviously incorrect reasoning of the model is the fact that in the database used to learn the model [30], in the pictures with bars, there were mostly dogs, which suggested to the network that as there are bars in the picture, therefore the result of classification should be a 'dog'.

3.2 Global explanations

XAI tools from the family of Global analyzers, are used for analyzing both ML models and datasets. Most of them are semi-supervised ones. They enable to analyze the ML models decisions learned from the large datasets, as well as to find the bias in the dataset. One of the most important method of global explanations is Spectral Relevance Analysis (SpRAy) [31]. It takes advantage of the results of local analysis, namely the LRP attention maps, to extract knowledge about the overall behavior of the analyzed model, in the form of clusters representing data-driven model features, from the global perspective. Authors of [9] proposed improved version of SpRAy, in which they employ both original images and heatmaps for analysis, which has significantly improved the method's efficiency in the problem of identifying the data bias. In this paper, as a global analyzer we exploit another effective method based on XAI signatures [32]. We approached the problem with LRP signatures instead of SHAP ones and applied the LRP on the last layer instead of the penultimate layer. Next, we reduced its dimensionality with algorithm Uniform Manifold Approximation and Projection [33]. UMAP is a manifold learning technique for dimension reduction of large-scale multidimensional data. It utilizes mathematical foundations related to Laplacian eigenmaps and Riemannian geometry.

4 Case study

4.1 Case study description

Our research aimed to analyze the influence of two representative types of attacks, i.e. FGSM and one-pixel attack on the performance of the attacked deep, convolutional neural network. For this purpose, we took advantage of local and global tools from the XAI family. We analyzed whether, normally unnoticed by the naked eye, the attack can be detected using appropriate XAI tools. In the case of individual images, a local analysis approach using heatmaps, namely the LRP present A flat method, was used for this purpose. To demonstrate the impact of these attacks convincingly and to be able to conclude, a database containing well-known figures, namely cats and dogs, was used. Fig. 3 contains exemplary images, original and attacked. The upper row contains the input images that are given to the input of the neural network, while the lower row depicts the corresponding heatmaps, generated by LRP method. As mentioned, red color indicates the pixels that most strongly correspond to the model output, while the blue one shows the pixels that have the least influence on the model output. Fig. 3 illustrate the effect of FGSM and one-pixel attacks onto the network. In both cases, it appears that the effectiveness of the prediction has decreased significantly after attack.

Despite the previous proper classification of the image (‘cat’), after the attack, the network classified the image incorrectly (‘dog’), with high accuracy. Before the attack, the heatmap indicating the significance of the input image pixels appeared as in Fig 3a. After the FGSM attack, the colors of the map, and thus the pixel significance, have practically reversed (Fig 3b). This is due to the nature of this method, a white-box method, i.e. it uses the full knowledge of the network and dataset to change the prediction. In case of one pixel attack (Fig 3d), it is clearly visible that the network is focusing its attention on this single, adverse pixel, which unfortunately results in a change of classification result. Similar results and effects were achieved for the whole analyzed dataset (see Table 1). This clearly proves how fragile deep models are these days, and that much attention needs to be paid to proper learning, improving the generalization abilities and robustness of these models, as well as to cybersecurity issues, especially for critical applications.

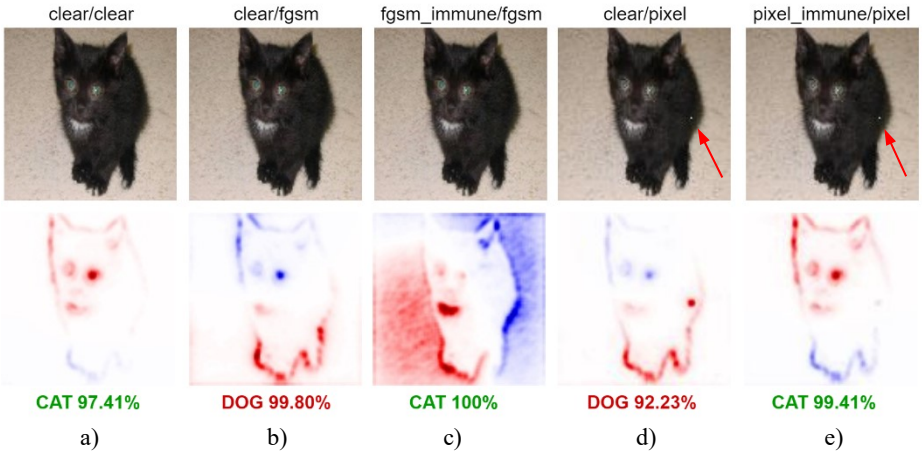

Fig. 3. FGSM and one-pixel attacks effect visualization. The arrows indicate the adverse pixel.

This problem can be partially remedied by the defense methods against attacks, described in section II.B. In this paper, the Adversarial retraining approach was used. For this purpose, a new training set was supplemented with successfully attacked with FGSM images. The extended training set was then used for retraining the ‘FGSM_immune’ model. Then it was checked whether the previously generated, extended testing set containing adversarial examples still affects the predictions. Using different testing images, it was then verified how many images can be attacked for ‘clear’ and ‘FGSM_immune’ models. The same procedure was applied to the one-pixel attack (‘One-pixel_immune model’). The averaged results for the entire dataset were gathered in Table 1. The effects of the applied approach can also be observed in Fig. 3. In the case of both FGSM (Fig. 3c) and One-pixel attacks (Fig. 3e), the network proved to be robust against them. In the case of One-pixel attack, the attention map looks similar to the one from before the attack (adverse pixel was ignored). The attention map for FGSM type attack looks a little different. Noises on the map are caused by their appearance in the training set utilized during retraining. Despite different heatmaps, the

classification result is correct. The differences in effects observed on heatmaps caused by different attacks can be used to detect and identify the types of attacks and to select the appropriate defensive strategy. The analysis of other images allows concluding that the described effects are representative. The results for the whole analyzed dataset are gathered in Table 1. It can be seen that such an approach has turned out to be successful in this case.

Table 1. Effectiveness of prediction depending on model and validation set type.

Model name/dataset	Clear	Clear	Clear	FGSM immune	One-pixel immune
Testing set	clear_test	clear+fgsm_test	clear+OP_test	clear+fgsm_test	clear+OP_test
Accuracy [%]	85.80	58.37	64.62	84.15	86.09

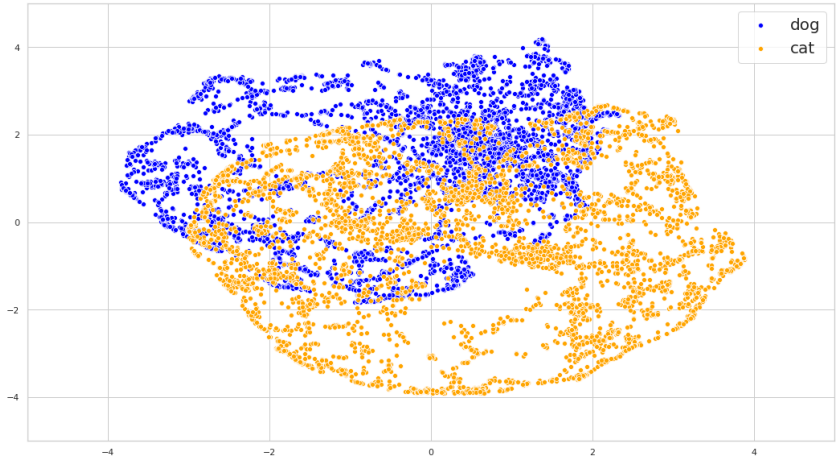

Fig. 4. Global UMAP visualization of local LRP signatures for examples of ‘cat’ and ‘dog’ images.

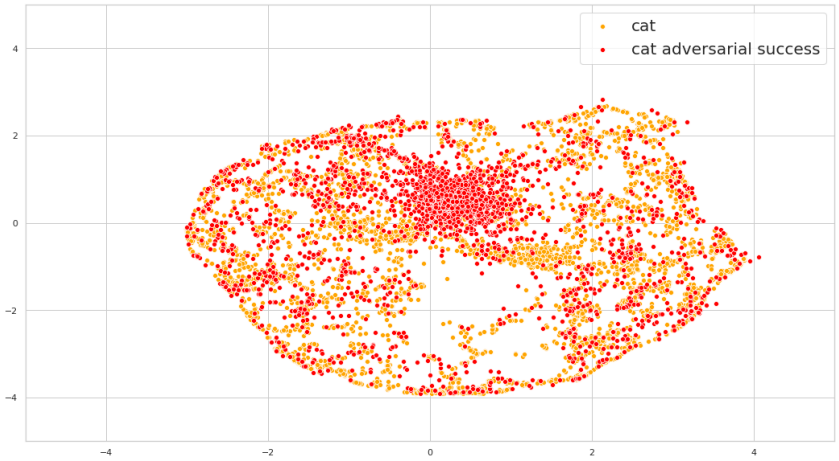

Fig. 5. Global UMAP visualization of local LRP signatures for examples and adversarial examples of 'cat' images – FGSM type of attack.

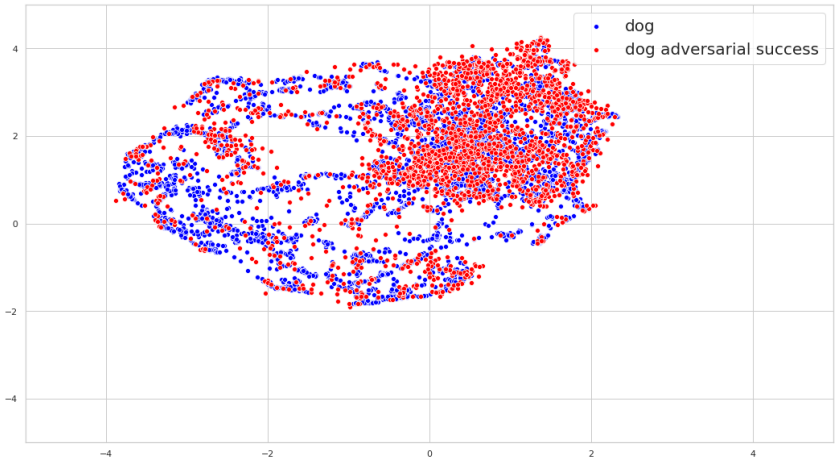

Fig. 6. Global UMAP visualization of local LRP signatures for examples and adversarial examples of 'dog' images – FGSM type of attack.

We have also attempted to analyze how the considered attacks affect the analyzed datasets globally, whether there are any specific patterns related to the type of attack and the dataset. Analyzed original and perturbed by FGSM and one-pixel attacks datasets were explored by taking advantage of the local explanation approach – LRP generating the attention maps, becoming the XAI signatures, and UMAP dimensionality reducer. To visualize the datasets before and after the attack was executed, we project these onto 2d space, via UMAP. The results for the FGSM type of attack are illustrated in Figs. 4-6. Fig. 4 shows the placement of images belonging to the class 'dogs' and 'cats', before the attack. Taking into account the correction for the fact that this is a projection of a multidimensional space into just 2 dimensions, please notice that some of the images

belonging to both classes have a small topological distance from each other. Intuitively, it can be expected that those images will be the easiest to effectively attack. Fig. 5 shows which images from the class of 'dogs' changed their class affiliation into ‘cats’, as a result of the FGSM attack. Note that as predicted, most of the successfully attacked images are grouped in the first quarter of the coordinate system, which is where both classes have the most common features. A similar effect can be observed in Fig. 6, for the class representing 'cats'. In this case, most of the effectively attacked images are located in the first quarter of the coordinate system, near the center of the system. The aforementioned results are only qualitative for the moment, and further much deeper research is needed to draw detailed and convincing conclusions.

4.2 Implementation details

During the research Kaggle dataset was used [30]. The images resolution is 112x112 px,. The training dataset (‘clear’- before the attack) consists of 8000 images (4000 cats and dogs); original testing set - ‘clear_test’: 2000 images (1000 cats and dogs); new (extended) training dataset for defense against FGSM –‘clear+FGSM’: 6078 cats 5878 dogs; new testing dataset-‘clear+FGSM_test’: 1408 cats 1532 dogs; new (extended) training dataset for defense against one-pixel attack –‘clear+OP’: 5178 cats 5454 dogs; new testing dataset–‘clear+ OP_test’: 1244 cats 1416 dogs.

CNN model: VGG16, with binary crossentropy loos function. SGD optimizer with: lr=0.001, decay=1e-6, was used during the training. Software libraries used: LRP attention maps [34]; FGSM [35]; one-pixel attack [36]; UMAP [37].

5 Summary and Concluding remarks

In this paper, we reviewed the recent findings in adversarial attacks and defense strategies. We also analyzed the effects of the attacks and the defense strategies applied, using recent XAI local and global approaches. We proposed to take advantage of the LRP attention maps and UMAP methods.

The results shown are preliminary, can be used for exploring the effects of adverse attacks, and allow to draw qualitative conclusions, only. However, undertaken analyses have confirmed that deep models are still fragile and are often not robust to well-prepared, intentional input perturbation.

Conducted research show that much attention needs to be paid to proper learning, improving the generalization and robustness of deep models, as well as to cybersecurity issues, especially for critical applications. It has also been demonstrated how valuable for process analysis XAI tools can be.

References

1. Goodfellow, I., Bengio, Y., Courville, A.:Deep Learning. MIT Press. ISBN: 0262035618 (2016).
2. Krizhevsky, A., Sutskever, I., Hinton, G.:ImageNet Classification with Deep Convolutional Neural Networks. Neural Information Processing Systems, vol. 25 (2012).
3. Kukačka, J., Golkov, V., Cremers, D.:Regularization for Deep Learning: A Taxonomy. arXiv:1710.10686 (2017).
4. Grochowski, M., Kwasigroch, A., Mikołajczyk, A.:Selected technical issues of deep neural networks for image classification purposes. Bulletin of the Polish Academy of Sciences. Technical Sciences, vol. 67, 2 (2019).
5. Mikołajczyk, A., Grochowski, M.:Data augmentation for improving deep learning in image classification problem. International Interdisciplinary PhD Workshop (IIPhDW), pp. 117–122 (2018).
6. Mikołajczyk, A., Grochowski, M.:Style transfer-based image synthesis as an efficient regularization technique in deep learning. 24th International Methods and Models in Automation and Robotics (MMAR), 42–47 (2019).
7. Elsken, T., Metzen, J. H., Hutter, F.:Neural Architecture Search: A Survey. arXiv: 1808.05377 (2019).
8. Kwasigroch, A., Grochowski, M., Mikołajczyk, A.:Neural Architecture Search for Skin Lesion Classification. IEEE Access, vol. 8, pp. 9061–9071 (2020).
9. Mikołajczyk, A., Grochowski, M., & Kwasigroch, A. (2020). Global explanations for discovering bias in data. arXiv preprint arXiv:2005.02269.
10. Yuan, X., He, P., Li, X., Zhu, Q.:Adversarial Examples: Attacks and Defenses for Deep Learning. IEEE Transactions on Neural Networks and Learning Systems, vol. 30, pp. 2805-2824 (2019).
11. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R:Intriguing properties of neural networks. 2nd International Conference on Learning Representations (2014).
12. Nguyen, A., Clune, J., Yosinski, J.:Deep Neural Networks are Easily Fooled: High Confidence Predictions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 427-436 (2015).
13. Eykholt et al.:Robust Physical-World Attacks on Deep Learning Visual Classification. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1625-1634, Salt Lake City, UT (2018).
14. Cisse, M., Adi, Y., Keshet, J., Neverova, N.:Houdini: Fooling Deep Structured Prediction Models. arXiv:1707.05373 (2017).
15. Finlayson, S. G., Chung, H. W., Beam, A. Kohane I. S.:Adversarial Attacks Against Medical Deep Learning Systems. arXiv:1804.05296 (2018).
16. Feinman, R., Curtin, R., Gardner, A., Shintre, S.:Detecting Adversarial Samples from Artifacts. arXiv:1703.00410 (2017).
17. Rigazio, L., Gu, S.:Towards Deep Neural Network Architectures Robust to Adversarial Examples. arXiv:1412.5068 (2014).
18. Xu, H., Ma, Y., Liu, H., Debayan, D., Liu, H., Jain, A., Tang, J.:Adversarial Attacks and Defenses in Images, Graphs and Text: A Review. arXiv:1909.08072 (2019).
19. Papernot, N., McDaniel, P., Wu, X., Swami, A., Jha, S.:Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks. In: IEEE Symposium on Security and Privacy, pp. 582-597 (2016).

20. Goodfellow, I., McDaniel, P., Papernot, N.:Making machine learning robust against adversarial inputs. Comm. of the ACM, Association for Computing Machinery, vol. 61, pp. 56-66 (2018).
21. Fidel, G., Bitton R., Shabtai, A.:When Explainability Meets Adversarial Learning: Detecting Adversarial Examples using SHAP Signatures. arXiv: 1909.03418 (2019).
22. Binder, A., Samek, W., Montavon, G., Lapuschkin, S., Müller, K. R.:Analyzing and Validating Neural Networks Predictions. ICML'16 Workshop on Visualization for Deep Learning (2016).
23. Ribeiro, M. T., Singh, S., Guestrin, C.:"Why Should I Trust You?": Explaining the Predictions of Any Classifier. arXiv:1602.04938 (2016).
24. Ribeiro, M. T., Singh, S., Guestrin, C.:Anchors:High-Precision Model-Agnostic Explanations. In: 32nd AAAI Conference on Artificial Intelligence (2018).
25. Smilkov, D., Thorat, N., Kim, B., Wattenberg, M., Viégas, F.:SmoothGrad: removing noise by adding noise. arXiv:1706.03825 (2017).
26. Zeiler M. D., Fergus R.:Visualizing and Understanding Convolutional Networks. In: Computer Vision. ECCV 2014. Lecture Notes in Computer Science, vol 8689, Springer (2014).
27. Moeys et al.:Steering a Predator Robot using a Mixed Frame/Event-Driven Convolutional Neural Network. Second International Conference on Event-based Control, Communication and Signal Processing (EBCCSP), pp. 1-8, Krakow (2016).
28. Ancona, M., Ceolini, E., Gross, M., Öztireli, C.:A unified view of gradient-based attribution methods for Deep Neural Networks. In: NIPS 2017-Workshop on Interpreting, Explaining and Visualizing Deep Learning. ETH, Zurich (2017).
29. Binder, A., Montavon, G., Lapuschkin, S., Muller, K. M., Samek, W.:Layer-wise Relevance Propagation for Neural Networks with Local Renormalization Layers. In: Artificial Neural Networks and Machine Learning - 25th International Conference on Artificial Neural Networks, ICANN 2016, vol. 9887 LNCS, Springer Verlag (2016).
30. Kaggle: Dogs & Cats Images, https://kaggle.com/chetankv/dogs-cats-images, 2020.
31. Lapuschkin, S., Wäldchen, S., Binder, A., Montavon, G., Samek, W., Müller, K. R.:Unmasking Clever Hans Predictors and Assessing What Machines Really Learn. Nat Commun, vol. 10, no. 1, p. 1096 (2019).
32. Fidel, G., Bitton, R., Shabtai, A.:When Explainability Meets Adversarial Learning: Detecting Adversarial Examples using SHAP Signatures. arXiv:1909.03418 (2019).
33. McInnes, L.:Umap. https://github.com/lmcinnes/umap, (2020).
34. Alber, M.:Innvestigate., https://github.com/albermax/innvestigate, (2020).
35. FGSM-Keras, GitHub. https://github.com/soumyac1999/FGSM-Keras.
36. GitHub. https://github.com/Hyperparticle/one-pixel-attack-keras.
37. UMAP: Uniform Manifold Approximation and Projection. https://github.com/lmcinnes/umap.