Stage 0: Adversarial attacks and anomaly detection in MIR

Vinod Subramanian, QMUL, ROLI Ltd.

Abstract

The abstract goes here.

Index Terms

MIR

I. Introduction

One of the biggest challenges of deep learning that remains today is the interpretability of what the deep learning model is learning. Due to the non-linearity and complexity of these models it is not a simple task to understand exactly what is being learned. With deep learning models slowly making their way into the real world and become a fixture of our daily lives it is even more important for us to understand what deep learning models are learning and what are the risks associated with deploying these deep learning models in the world.

Recent research has shown that by perturbing the input to a deep learning model by an imperceptible amount of non-random values we can change the output of the classifier. These ”attacks” to the model are referred to as adversarial attacks. This means that we can misclassify objects, misclassify speech to text etc. Some of these have security and safety consequences. If a self driving car incorrectly classifies a wall as a road it could lead to dangerous accidents.

The security aspect of adversarial attacks is often talked about but there is another, more fundamental question. Are we fundamentally training deep learning networks the wrong way? The success of adversarial attacks seems to indicate that there is a huge flaw in the assumption that deep learning models are learning in a manner that humans can interpret and understand.

So far adversarial attacks are explored in the computer vision domain and reinforcement learning domain. If we are to arrive at general observations and solutions to adversarial attacks and defenses we need to study this phenomenon in different domains and see what is generalizable and what is unique to a particular domain. I want my PhD to focus on exploring adversarial attacks and defenses in the area of music information retrieval (MIR).

II. Background

I’d like to see if I can create a summary of some of the research techniques used in some of the literature I am reviewing in the form of a table to compare techniques and performance.

A. What are adversarial attacks?

Szegedy et al. [5] discovered that in object recognition tasks by applying an imperceptible non-random perturbation to the input image the output of the network can be changed. The term ”Adversarial examples” is used to describe these perturbed examples. These adversarial examples were attributed to the fact that there are blindspots in the training process for these deep learning models.

Goodfellow, Shlens and Szegedy [2] challenged the idea that adversarial examples were due to blind spots in high dimensional spaces. Instead, they suggested that they are caused due to linearity in deep learning models. According to them LSTMS, ReLUs, maxout networks, CNNs etc. were intentiaonally designed to behave more linearly in order to make them easier to optimize.

Assume we have a linear classifier defined by the relationship $y = f(x)$ where $f(x)$ is given by $f(x) = \omega^T x$. Our goal is to perturb the input x with perturbation η subject to the condition that $D(x + \eta, x) < \epsilon$ where D is some distance measure. We can write the new equation for the ouput of the classifier as:

$$\tilde{y} = \omega^T x + \omega^T \eta \tag{1}$$

By choosing η carefully so that it is aligned in the direction of the weights of the classifier ω the change in the output can be maximized.

Goodfellow, Shlens and Szegedy [2] used this assumption on deep learning models to generate adversarial examples and came up with a family of fast gradient approaches to generate adversarial examples. The success of these adversarial examples lent credence to the fact that deep learning models are susceptible to adversarial examples on account of being too linear.

Before we give more details about different types of adversarial attacks and defenses against these attacks we'll list the different categories of adversarial attacks. The category of an adversarial attack is determined by either the goal of the adversary or the information the adversary has of the classifier.

Goal of the adversary:

1) Untargeted attack - The adversary's goal is to simply misclassify the input. As long as the new target class of the classifier is different from the original the adversary has achieved its goal.
2) Targeted attack - The adversary's goal is to change the label from the original to a different label that is specified before the attack begins. Because there are more constraints in the targeted attack it is typically harder to generate targeted adversarial examples as opposed to untargeted adversarial examples.

Information of the adversary:

1) Perfect knowledge - In this scenario the adversary has perfect knowledge of the classifier such as the feature space, the weights of the model and the type of classifier. This is also known as a white box attack.
2) Limited knowledge - The classifier does not know the training data or the trained model but has knowledge about the feature representation and the type of classifier. This could be considered a black box attack.
3) Zero knowledge - This is a special case where an adversarial example is generated for one classifier and then tested on a different classifier which it has no information about.

In the following sections I will introduce different techniques for attacks, different techniques for defenses and some unique properties of adversarial attacks such as universal perturbation, transferability of adversarial examples and adversarial examples in the real world. Finally I will highlight the existing body of work that explores adversarial attacks in audio and talk about existing deep learning tasks in music information retrieval (MIR) that would be a good starting point for research into adversarial attacks in music.

B. Summary of attacks

1) L-BFGS - Szegedy et al. [5] introduced one of the first methods for generating adversarial attacks, it is a white box attack that is targeted.

   Assume a classifier denoted as $f : \mathbb{R}^m \rightarrow \{1...k\}$ with a loss function $loss_f$. For a given input $x \in \mathbb{R}^m$ and target $t \in \{1....k\}$ we aim to identify the value of perturbation r as formulated below:

   $$\begin{aligned}&\text{Minimize } \|r\|_2 \text{ under the conditions:}\\&f(x+r) = t\\&x + r \in [0,1]^m\end{aligned}$$

   The exact computation of this problem is difficult so it is approximated using the box constrained L-BFGS algorithm. So the new equation to minimize is:

   $$c|r| + loss_f(x+r, l) \qquad\qquad \text{under the conditions } x + r \in [0,1]^m$$

2) Fast Gradient Sign Method - The goal of this method is to quickly generate a simple adversarial examples. By perturbing the input in the direction of the gradient of the weights of the model by a sufficient amount it can create an adversarial example [2]. Let x be the input to the model and y be the output, the model parameters are θ. The cost function of the model is $J(\theta, x, y)$. With this information we can create an adversarial example by computing the perturbation η using the formula:

   $$\eta = \epsilon sign(\triangledown_x J(\theta, x, y)) \tag{2}$$

   In this equation ϵ is decided manually until the formula causes a perturbation. This technique was refined by making the process iterative and computing the gradient repeatedly and showed improved results.

3) Jacobian based Saliency Map Attack (JSMA) - Papernot et al. [3] introduced a white box attack which requires knowledge of the model parameters but does not require knowledge of the training data, it is a targeted attack. The goal of the algorithm is to identify the pixels in the input that impact the output the most and perturb these important pixels to change the output to the target output.

   Assume a classifier denoted as f with output dimensions N that takes $\mathbf{X} \in \mathbb{R}^M$ as an input. The first step is to compute the forward derivative:

   $$\begin{aligned}\triangledown f &= \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}}\\&= \frac{\partial f_j(\mathbf{X})}{\partial x_i}\ i \in 1..M, j \in 1..N\end{aligned}$$

This is essentially the Jacobian matrix of the function denoted by f. The computation of this forward derivative can be simplified using the chain rule. The next step is to compute the saliency map [4] based on the forward derivative. The saliency map shows which input features are most important in determining the output.

$$\text{For } i \in 1..M \text{ and } t \text{ as target output}$$
$$S(\mathbf{X}, t)[i] = \begin{cases} 0 \text{ if } \frac{\partial f_t(\mathbf{X})}{\mathbf{X}_i} < 0 \text{ or} \sum_{j \neq t} \frac{\partial f_j(\mathbf{X})}{\mathbf{X}_i} > 0 \\ \left(\frac{\partial f_t(\mathbf{X})}{\mathbf{X}_i}\right) \left| \sum_{j \neq t} \frac{\partial f_j(\mathbf{X})}{\mathbf{X}_i} \right| \text{ otherwise} \end{cases}$$

From the saliency map we identify the input x_i that has the highest impact on the target output and perturb it by parameter $theta$ that is problem specific. This process is repeated iteratively and the maximum iteration is determined by the distortion limit γ. The distortion limit is manually set at the boundary at which humans can observe the distortion and is problem specific.

4) DeepFool -
5) Carlini and Wagner - Carlini and Wagner introduce a strong set of attacks based on the L_0, L_2 and L_inf distance [**?**]. The problem for adversarial attacks is formulated the same was as Szegedy et al. [**?**]. The classifier is denoted as C with input x:

$$\text{Minimize } D(x, x + \delta) + c.f(x + \delta)$$
$$\text{such that } x + \delta \in [0, 1]^n$$

Here D is a distance function that is either the L_0, L_2 or L norm and f is an objective that simplifies the problem such that:

$$C(x + \delta) = t \text{ is true if } f(x + \delta) \leq 0$$
$$f(x') = (max(Z(x')_i) - Z(x')_t)^+ \ i \neq t$$

In the equation for f, Z denotes the penultimate layer of the classifier, i.e the softmax values and t is the target class. This is just one example of a the function f many other functions work and can be found in the paper [**?**]. Carlini and Wagner [**?**] take care of the box constraint issue on $x + \delta$ by changing the variables with the substitution:

$$\delta_i = \frac{1}{2}(tanh(w_i) + 1) - x_i$$

C. Properties of adversarial attacks

In this section we'll highlight a few interesting properties of adversarial attacks which highlight the risk and importance of studying them in more detail.

1) Transferability of attacks - Papernot, Mcdaniel and Goodfellow [**?**] showed that adversarial examples are transferrable between machine learning models that use the same technique. They went one step further and showed that the models don't need to be using the same technique but still have transferrable adversarial examples. Liu et al. [**?**] showed that untargeted attacks are more transferrable than targeted attacks. In addition, they used an ensemble of models to create stronger adversarial attacks. Moosavi-Dezfooli et al. [**?**] created a universal perturbation that are image and network agnostic.
2) Real world attacks - Research has shown that adversarial examples are not limited to the digital world of finely crafted adverseries. Kurakin, Goodfellow and Bengio [**?**] showed that by printing and taking a photo of an adversarial example there are a significant number of cases for which it remains an adversary. Athalye et al. [**?**] created a framework in which objects can be rendered in 3-D and be printed and serve as adversarial examples for. Finally, Eykholt et al. [**?**] developed and algorithm to create perturbations to generate adversarial examples that are physically realizable.

D. Summary of defenses

1) Defensive distillation -
2) Retraining - Adversarial training is where adversarial examples are introduced into the training process. Introduces a method where the goal through the new training is to correctly classify the adversarial examples.
3) Dimensionality reduction
4) Statistical analysis

E. Adversarial attacks in audio

F. Anomaly detection

1) Summarize the survey paper on anomaly detection from 2007. This survey paper talks a lot about older approaches to anomaly detection that are still relevant
2) For mechanical failures there are deep learning approaches for anomaly detection, mostly inspired by the fact that it is harder to define the anomaly.
3) Work done on anomaly detection in medicine to identify diseases from measurements automatically
4) Work done on anomaly detection on GTZAN dataset using classical machine learning and on mammal sound recognition

III. Research questions and methods

In this section I'll list all of my ideas for this project so far which I will reference again in the timeline section. I am going to try and include block diagrams and images to make my ideas clearer.

A. Recreating adversarial attack experiments in audio

1) Transferring adversarial examples in audio between models with identical inputs: In computer vision they found that adversarial examples generated in one model served as an adversarial example in another model. This indicates that there seems to be some fundamental way in which we train deep learning models that make them susceptible to errors
2) Can adversarial examples be played over the air: In computer vision a photo of an adversarial example taken from a phone camera still works as an adversarial example
3) Can you create a universal perturbation that has a high chance of making an audio an adversarial example: In computer vision they came up with a universal method to create an adversarial example

B. Adversarial attacks for audio

1) Unlike computer vision audio domain uses different types of input features for deep learning. It would be interesting to see if it is possible to design an adversarial attack in the time domain that is robust to the different input representations
2) Design attacks for a specific task such as singing voice transcription. Singing voice transcription has a few different high accuracy models to work with so it is a good candidate to perform adversarial attacks on
3) Some research is done showing that mp3 compression preserves the adversarial examples but it would be interesting to try different bitrates and different compression methods

C. Defenses against attacks

1) Apply existing defenses to the audio problem to see if they work
2) Design defenses that are tailored to the specific tasks that I choose to work on
3) Explore the idea of using adversarial examples as a data augumentation technique to create higher accuracy models

D. Anomaly detection in note level datasets

1) Ensemble of methods to identify poor quality, corrupted audio and noise in datasets
2) Replace ensemble of methods with a single deep learning model and explore the use of multi-task learning

IV. Timeline

Table listing different projects, their timeline and targeted conferences

V. Conclusion

Summarize the whole proposal

Acknowledgment

References

[1] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion Attacks against Machine Learning at Test Time. *arXiv:1708.06131 [cs]*, 7908:387–402, 2013. arXiv: 1708.06131.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. *arXiv:1412.6572 [cs, stat]*, December 2014. arXiv: 1412.6572.

[3] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The Limitations of Deep Learning in Adversarial Settings. *arXiv:1511.07528 [cs, stat]*, November 2015. arXiv: 1511.07528.

[4] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv:1312.6034 [cs]*, December 2013. arXiv: 1312.6034.

[5] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199 [cs]*, December 2013. arXiv: 1312.6199.