Robust Classifier using GAN

Capstone Proposal

Machine Learning Engineer Nanodegree

Prity Roy

June 19th, 2017

Proposal

Domain Background

Deep Neural Networks (DNN) perform extremely well on many classical machine learning problems. However, it is now known that these are also susceptible to adversarial examples, small input perturbations in input data are leading to incorrect predictions [1], like in any other machine learning techniques.

Such attacks can seriously undermine the security of the system supported by the DNN, sometimes with devastating consequences. For example, autonomous vehicles can be crashed, illicit or illegal content can bypass content filters, or biometric authentication systems can be manipulated to allow improper access.

Problem Statement

As more and more applications are using Deep Neural Nets and other machine learning models it is thus meaningful and urgent to increase the local stability towards adversarial examples. It is necessary to decrease the gap between the machines and the human perception, and devise safety mechanism for security–sensitive applications.

These adversarial examples are either inputs that an attacker has designed to specifically fool the model or data occurring naturally. The vulnerability caused by their existence is somewhat disturbing. In the case of visual data, for example, it would be expected that images that are perceivable to “human eye” be predicted to the same accuracy.

Several works have been proposed to use adversarial examples during training of neural networks, and reported increase in classification accuracy on test data [1], [2]. The goal of this project is to make deep neural network model robust to adversarial examples using such training [3], [4].

Datasets and Inputs

In the experiment, MNIST database is used, a subset of a larger set available from NIST. The MNIST, handwritten digits, dataset is split into three parts: 55,000 examples of training dataset, 10,000 examples of test dataset, and 5,000 examples of validation dataset.

Every MNIST data point has two parts: an image of a handwritten digit and a corresponding label. The MNIST data is hosted on Yann LeCun's website [5].

The digits have been size-normalized and centered in a fixed-size image. Each image is 28 pixels by 28 pixels. We can interpret this as a big array of numbers.

Solution Statement

Using a generative adversarial network(GAN), first introduced by Ian Goodfellow and others in Yoshua Bengio's lab in 2014 [6], the classical deep neural network is made robust to adversarial examples.

Generative Adversarial Networks (GANs) are a powerful class of generative models that cast the generative modeling problem as a game between two adversary networks: the generator network produces synthetic data given some noise source and the discriminator network discriminates between the generator's output and true data.

The generator network generates an adversarial perturbation that can easily fool the classifier network by using a gradient of each image. Simultaneously, the discriminator network is trained to classify correctly both original and adversarial images generated by the generator. These procedures help the classifier network to become more robust to adversarial perturbations.

Benchmark Model

A deep neural net with a high accuracy on the MNIST test dataset is taken as a benchmark model. This model is only trained on the MNIST training dataset and not against adversarial examples. A comparison here is made with another model which is robust to adversarial examples.

Evaluation Metrics

The accuracy of the benchmark model and the solution model is evaluated using the cross entropy loss function.

$$\mathcal{L}(X, Y) = -\frac{1}{n}\sum_{i=1}^{n} y^{(i)} \ln a(x^{(i)}) + \left(1 - y^{(i)}\right) \ln\left(1 - a(x^{(i)})\right)$$

Here, $X = \left\{x^{(1)}, \ldots, x^{(n)}\right\}$ is the set of input examples in the training dataset,

and $Y = \left\{y^{(1)}, \ldots, y^{(n)}\right\}$ is the corresponding set of labels for those input examples.

The a(x)represents the output of the neural network given input x.

As the GAN model contains two classifier, firstly the Discriminator, which is trained to identify the real and fake data and secondly the Digit Classifier, which is also trained along with the Discriminator on the real and fake data in order to make it robust to adversarial examples and also to classify the digits.

As the purpose is to evaluate the robustness of the Digit Classifier, that is, successful identification of digits from a generated image, the above cross entropy loss is used here.

The loss function used to optimize the Discriminator or critic in the GAN is however, the EM distance used in Wasserstien GAN. The value function for a WGAN is constructed by applying the Kantorovich-Rubinstein duality (Villani, 2008) to obtain

$$\min_G \max_{D \in \mathcal{D}} \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mathbb{P}_r} \left[D(\boldsymbol{x}) \right] - \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}} \sim \mathbb{P}_g} \left[D(\tilde{\boldsymbol{x}}) \right]$$

where D is the set of 1-Lipschitz functions and P_g is once again the model distribution implicitly defined by $\tilde{x} = G(z)$, $z \sim p(z)$.

Project Design

There are three sections:

Firstly, a benchmark model - classifier, using Deep Neural Network architecture, is trained on MNIST dataset with a goal of attaining very high accuracy in its test dataset.

Secondly, another classifier using the same Deep Neural Network architecture as the above is trained in adversarial setting. The basic building block of the approach is the generative adversarial network (GAN) of Goodfellow et al. [6]. Generative Adversarial Networks (GANs) are a powerful class of generative models that cast the generative modeling problem as a game between two adversary networks: the generator network produces synthetic data given some noise source and the discriminator network discriminates between the generator's output (fake) and true data (real).

The project will use Wasserstein Generative Adversarial Network (WGAN) technique [7], [8]. As in WGAN the Discriminator or also called Critic, only spits a binary output, that is, if the input data is real or fake and does not classify the digits, a separate classifier is trained on the real and fake data to classify on the handwritten digits.

Finally, both the networks is evaluated against both the MNIST test dataset and generated samples by the Generator and reported. This model trained under adversarial training is robust to perturbations.

References

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

[2] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014.

[3] T. Miyato, S.-i. Maeda, M. Koyama, K. Nakae, and S. Ishii. Distributional smoothing with virtual adversarial training. arXiv preprint arXiv:1507.00677, 2015.

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

[5] Yann LeCun's website http://yann.lecun.com/exdb/mnist/

[6] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. arXiv preprint arXiv:1406.2661, 2014.

[7] Arjovsky, Martin, Chintala, Soumith, and Bottou, Léon. Wasserstein gan (WGAN). arXiv preprint arXiv:1701.07875, 2017.

[8] Ishaan Gulrajani , Faruk Ahmed, Martin Arjovsky , Vincent Dumoulin, Aaron Courville Improved Training of Wasserstein GANs (WGAN-GP). arXiv preprint arXiv:1406.2661, 2014.