CS230

Improved Reliability using Data Augmentation

Mike Murugan, Amit Kulkarni
KLA Corporation
5 Technology Drive, Milpitas CA 95035
Mimuruga@stanford.edu, Kulkarn1@stanford.edu

Abstract

The paper details augmentation techniques used for multi class classification problems where the data is insufficient. For the classification, CNN model using ConvNet is used for Extended MNIST dataset-based digit classifier. Various levels of subset from the data set were used for evaluating data augmentation techniques. For data augmentation, manual image transformations and Balancing GAN were used. For smaller population of data (1%) with unbalanced classes(5 % in digit "0"), BAGAN based data augmentation helps to improve Multi-label classifier accuracy. For Medium or Larger population of data (20 % and more), Manual data augmentation surpasses BAGAN based augmented data in improving Multi-label classifier accuracy.

1 Introduction

A deep learning model takes you as far as your data goes [1]. Having a large amount of training data set that is also diverse is a key in an accurate deep learning model. Specifically, for multi class classifier network, it's hard to ensure its precision and recall for the following two cases; 1. rarely occurring classes, and 2. lesser amount of training data across the classes. This necessitates data augmentation techniques to train with more/balanced data. For ex., we may not have enough images for a specific digit in a "hand written digit recognition problem". This may lead to an inaccurate model. To address the imbalanced data, we can use data augmentation techniques. This project investigates two such techniques, one based on manual transformations and the other based on GAN and reports the findings. We used multiple levels of unbalanced subsets (1%, 20%, 100%) of EMNIST data set to simulate the problem statement. The paper is organized as follows. A survey of current work in the area of data augmentation is done in next section. To begin the description of this work, the data sets used are described in the nest section followed by Methods and Models used for the project. Experimental results and discussions are detailed next followed by conclusions and future work.

2 Related Work

There are two broad approaches to data augmentation; 1. using manual translations (flipping, rotating, mirroring, etc.) to multiply the existing data [1] or 2. using neural networks to generate extra set of data. For manual translations, we need a significant amount of data from the unbalanced (minority) class to begin with. For neural network-based models, there have been many solutions proposed [4, 5, 6, 7, 8, 9] in the past. The Balancing GAN [8] is used to correct unbalanced dataset across classes. It uses autoencoder initialized generative model to learn features from majority classes and uses these to generate images for minority classes. One down side of GAN networks is lack of diversity [2, 3]. The generative network tends to have a narrower distribution to satisfy discriminator. To address the diversity in output space w.r.t, input space, Variable Auto Encoder [6] was proposed that models the distribution of high-dimensional output space as a generative model conditioned on the input observation. The downside to VAE is that it uses mean squared error and not an adversarial network, so the network tends to produce blurry images lacking quality. CVAE-GAN [10] addresses this by combining the distribution of CVAE and Adversarial network of GAN.

3 Data Sets

For this project we experimented with various data sets as follows. The initial models we picked were based on MNIST data set. After evaluating the models with MNIST we ended up using EMNIST because it was more challenging. We also used Omniglot for a GAN model (Data Augmentation GAN) but ended up not using the model.

- http://yann.lecun.com/exdb/mnist/ - A dataset with 60000 training images and 10000 test images
- https://github.com/brendenlake/omniglot - Contains 1623 characters from 50 alphabets (Turkish)
- https://www.nist.gov/node/1298471/emnist-dataset EMNIST Digits: 240,000 training images and 40000 test images for 10 balanced classes of hand-written digits.

4 Methods and Models

The goal of this project was to test data augmentation techniques. We used a large data set (EMNIST with 240000 images) since we could use established models for classification and data augmentation. The following are the details in the models we used and the techniques we used.

Multi class classifier model:

For multi class classifier, we used sequential ConvNet and trained on EMNIST dataset. We used the classifier model from Kaggle (https://github.com/keras-team/keras/blob/master/examples/mnist_cnn.py). We made changes to the model to use EMNIST and the accuracy we achieved with 12 epochs on full Extended MNIST was 0.9951.

To simulate the scarcity of the data, we used varying subset of EMNIST data set as shown in Figure 1. For ex., we used 1 % of the total EMNIST training data set so total images are 2400. We ran the classifier to capture the accuracy. Then we further reduced class '0' to only 5 %. With this, we again ran the classifier to see the declined overall accuracy. We saw a marked decline in Recall and F1 score for class ‘0’. We then augmented the class data with GAN generated images for the class and evaluated the classifier model accuracy.

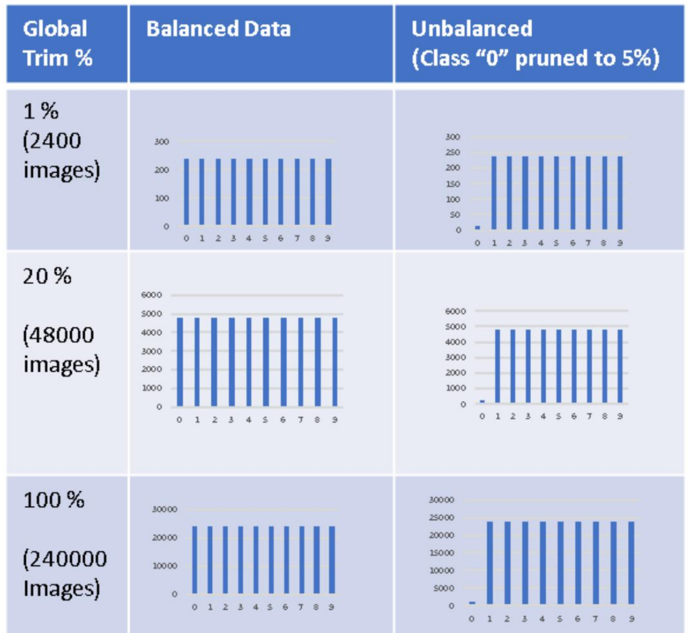
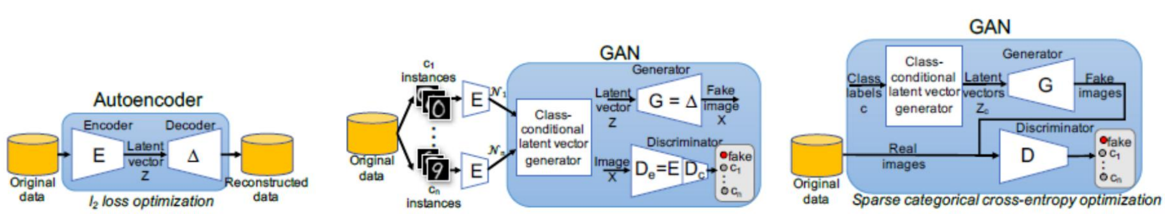


Figure 1: Data sets used for evaluating Augmentation Techniques

Figure 2:Multi-label Classifier Model

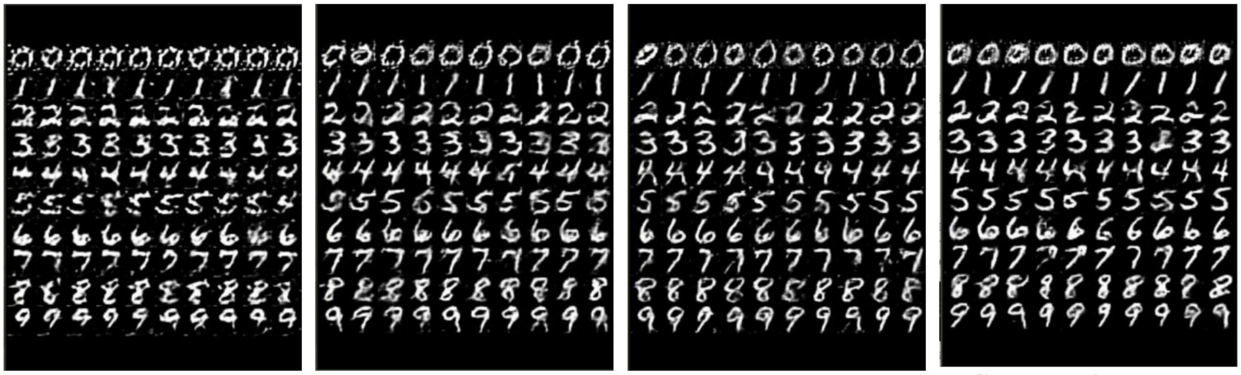
Data Augmentation Techniques

- **BAGAN:** The generative model learns useful features from majority classes and uses these to generate images for minority classes. Class conditioning in the latent space is applied to drive the generation process towards a target class. The generator in the GAN is initialized with the encoder module of an autoencoder that enables us to learn an accurate class-conditioning in the latent space. For example, the model learns from all classes in MNIST database and generates numbers that has a smaller number of samples



(a) Autoencoder training. (b) GAN initialization. **(c) GAN training.**

Figure 3: BA GAN Model [4]



Generated images after epoch 9 | Generated images after epoch 49 | Generated images after epoch 99 | Generated images after epoch 149

Figure 4 BAGAN Generated Images Through 150 Epochs

- **Manual Augmentation:** We used rotation, gaussian noise and zoom to the digit images to augment the data. Other translations such as horizontal, vertical flipping and mirroring are not possible on handwritten digits.

5 Experiments/Results/Discussion

Table 1 shows the results of the accuracy of classifier model when ran with a set of data sets. As we can see, for very small data sets, BAGAN does a better job than manual augmentation. However, for larger data sets it starts to deteriorate.

Table 1: Experiment Results.

Dataset (10 classes of handwritten digits)	Fully Balanced Dataset (A)	Dataset with 95% images removed from '0' digit class (B)	Data set from B augmented with BAGAN to balance it (95% of '0' images from BAGAN)	Data set from B with manual augmentation to balance it (95% of '0' images from augmentation)
1% of EMNIST (total images 2400)	Accuracy Score: 0.955 Report: precision recall f1-score '0' 0.98 1.00 0.99 1 0.98 1.00 0.99 2 0.97 0.88 0.92 3 0.95 0.88 0.91 4 0.87 0.97 0.92 5 0.93 0.97 0.95 6 1.00 0.97 0.99 7 0.97 0.95 0.96 8 0.97 0.97 0.97 9 0.95 0.95 0.95	Accuracy Score: 0.9125 Report: precision recall f1-score '0' 0.96 0.60 0.74 1 0.98 1.00 0.99 2 0.95 0.88 0.91 3 0.97 0.88 0.92 4 0.84 0.95 0.89 5 0.85 0.97 0.91 6 0.87 1.00 0.93 7 0.91 0.97 0.94 8 0.90 0.93 0.91 9 0.95 0.95 0.95	Accuracy Score: 0.945 Report: precision recall f1-score 0 0.97 0.90 0.94 1 0.98 1.00 0.99 2 1.00 0.85 0.92 3 0.95 0.90 0.92 4 0.85 0.97 0.91 5 0.97 0.97 0.97 6 0.95 1.00 0.98 7 0.91 0.97 0.94 8 0.95 0.93 0.94 9 0.95 0.95 0.95	Accuracy Score: 0.9425 Report: precision recall f1-score '0' 0.95 0.97 0.96 1 0.97 0.97 0.97 2 0.97 0.97 0.97 3 0.92 0.90 0.91 4 0.90 0.93 0.91 5 0.94 0.85 0.89 6 0.98 1.00 0.99 7 0.97 0.95 0.96 8 0.90 0.93 0.91 9 0.90 0.95 0.93
20% of EMNIST (Total 48000 images)	Accuracy Score :0.991875 Report: precision recall f1-score '0' 1.00 0.99 0.99	Accuracy Score: 0.98775 Report : precision recall f1-score '0' 1.00 0.94 0.97	Accuracy Score: 0.99075 Report : precision recall f1-score 0 1.00 0.95 0.98	Accuracy Score: 0.990375 Report: precision recall f1-score '0' 0.99 1.00 0.99
Full EMNIST dataset (240000 images)	Accuracy Score :0.9951 Report : precision recall f1-score '0' 1.00 1.00 1.00	Accuracy Score :0.99255 Report : precision recall f1-score '0' 1.00 0.97 0.98	Accuracy Score :0.9917 Report : precision recall f1-score 0 1.00 0.96 0.98	Accuracy Score :0.994725 Report : precision recall f1-score '0' 0.99 1.00 0.99

6 Conclusion/Future Work

The experiments were conducted to characterize augmentation techniques both manual and GAN based. We conclude that both the methods are effective in increasing the accuracy of the model. For smaller population of data (1%) with unbalanced classes(5 % in digit "0"), BAGAN based data augmentation helps to improve Multi-label classifier accuracy. For Medium or Larger population of data (20 % and more), Manual data augmentation surpasses BAGAN based augmented data in improving Multi-label classifier accuracy.

BAGAN is good in data augmentation for low data regime with unbalanced classes. But in high data regime generated data lacks diversity. Variable auto encoders comes handy in this case, but would generate blurrier image, hence CVAE GAN is promising area to investigate.

7 Contributions

This project was done as follows. Mike worked on multiple GAN based data augmentation models and chose BAGAN as the choice for the work. Amit worked on multi class classifier and creating reduced and unbalanced data sets. Mike and Amit worked together on the manual augmentation model and surveyed the papers together. This was a highly collaborative work.

References:

[1] Bharath Raj, (2018), Data Augmentation, retrieved from https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced

[2] Sanjeev Arora, (2018), "Do GANs actually learn the distribution? Some theory and empirics.", Retrieved from http://efrosgans.eecs.berkeley.edu/CVPR18_slides/Theory_by_Arora.pptx

[3] Sanjeev Arora and Rong Ge and Yingyu Liang and Tengyu Ma and Yi Zhang, (2017), "Generalization and Equilibrium in Generative Adversarial Nets ({GAN}s)", Proceedings of the 34th International Conference on Machine Learning, pp224-232

[4] A Neural Algorithm of Artistic Style Leon A. Gatys, Alexander S. Ecker,1,2,4,5 Matthias Bethge1,2,4

[5] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros (2017) "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", The IEEE International Conference on Computer Vision (ICCV), pp. 2223-2232

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros (2017) "Image-To-Image Translation with Conditional Adversarial Networks" The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1125-1134

[7] K. Sohn, H. Lee, and X. Yan. (2015) "Learning structured output representation using deep conditional generative models." In Advances in Neural Information Processing Systems, pp. 3483–3491

[8] Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Constantine Bekas, A. Cristiano I. Malossi, (2018) "BAGAN: Data Augmentation with Balancing GAN" ArXiv

[9] Antreas Antoniou, Amos Storkey, Harrison Edwards (2018) "Data Augmentation Generative Adversarial Networks", arXiv

[10] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, Gang Hua, (2017), "CVAE-GAN: Fine-Grained Image Generation through Asymmetric Training", arXiv