On the Use of Denoising Diffusion Models for Dataset Rebalancing

Shaan Gill
Georgia Institute of Technology
CS 7643
sgill36@gatech.edu

Charles Snider
Georgia Institute of Technology
CS 7643
cgsnider@gatech.edu

Manas Harbola
Georgia Institute of Technology
CS 4644
mharbola3@gatech.edu

Ethan Mendes
Georgia Institute of Technology
CS 4644
emendes@gatech.edu

Abstract

In recent years, the development of diffusion models has led to their adoption in many areas of computer vision. These models have often supplanted older generative adversarial network (GAN) architectures, which have shown to be consistently outperformed by diffusion models. In this work, we investigate whether this adoption of diffusion models is appropriate to the problem of skewed dataset rebalancing multi-class classification settings, which involves increasing the number of images in minority-classes in skewed datasets from limited resource domains. Specifically, we compare the classification accuracy on a skewed cancer tumor dataset after it has been rebalanced using the current state-of-the-art GAN architecture and denoising diffusion probabilistic models (DDPM's). Through our experiments, we find little difference in performance between these approaches and the baseline and conclude that the vanilla DDPM architecture is not sufficient to improve performance on this rebalancing task.

1. Introduction

Consider a computer vision data set that has a heavy bias in its data. Perhaps the data set is on medical data for a relatively rare condition. Thus training on this data set will cause a classification model to have a bias toward predicting the lack of the presence of the condition simply because it is less common in the data set. To remedy this traditional approaches are to introduce data augmentation to increase variation in the original data set and to provide more data points. However, the effectiveness of this technique is limited because fundamentally the images are still the same images just with slight tweaks.

We aim to provide an alternative. Our goal is to use generative models to create new samples of the underrepresented category. We will use BAGANs and diffusion models as our generative models. BAGANs or a balanced GAN is a model that differs from traditional GANs and is shown to perform better on unbalanced data [8]. BAGANs have been shown to have higher performance than GANs on medical data with imbalanced minority classes and scaled-down copies of the MNIST fashion and CIFAR-10 datasets. Diffusion models work by adding Gaussian noise to an image in the forward diffusion process, and then it will try to recover the image in the reverse diffusion process. OpenAI argues that diffusion models outperform GANs on multi-class datasets [1] which is what we want to further investigate in this project. Residual Network or ResNet is an architecture that is used to classify images to a high degree of accuracy [4]. We will use ResNet to measure the model's accuracy on the original unbalanced dataset and the accuracy of the datasets produced by each of the two generative models. The inputs to these models will be an unbalanced dataset of brain tumors [9] and the output would be to generate enough images to balance the dataset. We will then take these outputs to train on a classification model (ResNet).

The goal of this is to see if the diffusion model can outperform BAGAN and the baseline (accuracy on unbalanced dataset trained through ResNet). Additionally, due to the novelty of diffusion models, using them to rebalance a skewed dataset appears to be an unexplored concept. As such, we are interested in seeing if diffusion models provide a comparative advantage at this task over GANs similar to their performance on multi-class image generation.

2. Related Works

Rebalancing datasets in domains with data scarcity has been a problem since the advent of machine learning. Many machine learning practitioners use simple techniques for rebalancing such as applying resampling techniques such as oversampling, undersampling, Synthetic Minority Oversampling TEchnique (SMOTE), and Monte-Carlo methods [3].

Recently, researchers have also adapted generative models for the purpose of resampling. Specifically, these generative models are trained using the limited training data for the purpose of generating supplemental data of the minority classes such that all classes are equally represented in the training dataset. However, researchers have found that traditional generative modelling frameworks such as vanilla GAN's do not work well in improving performance for unbalanced datasets on classification tasks such as MNIST, and in fact, perform worse than the aforementioned resampling techniques [10].

Mariani et al. hypothesize that this poor performance could be attributed to a lack of minority class data, as GANs often require a large and balanced image training dataset, which is not available in this setting. This prompted them to address the issue of skewed image datasets with sparse minority-class images using an approach that couples a GAN with an autoencoder that allows for separate autoencoder training and adversarial training that includes all available images in the original training dataset, instead of only those in the minority-class [8]. Huang and Jafari further attempt to improve BAGANs by adding a gradient penalty (BAGAN-GP) that achieves higher performance on imbalanced scaled-down versions of MNIST Fashion, CIFAR-10, and a medical cell dataset. [6].

Denoising diffusion probabilistic model (DDPM's) were first introduced by Ho et al. in their 2020 paper. These models are trained by combining a forward diffusion process of repeatedly adding Gaussian noise to training images with a backward diffusion or "denoising diffusion" process of recovering the original image. Providing random Gaussian noise to a trained model allows for the generation of new class images [5].

Researchers at OpenAI show that diffusion models are overall better than GANs at generating images in multi-class datasets such the LSUN dataset with improved scores for the former over the latter in metrics such as Inception Score (IS) [11]. It is hypothesized that this is the case because of the iterative nature of the forward and backward diffusion processes [1]. In spite of this work, to our knowledge, diffusion models and specifically DDPM's have not been applied to the problem of dataset rebalancing as an alternative to GAN-based architectures. In this paper, we investigate how the performance of DDPM's compare to these traditional architectures on the task of rebalancing in the realistic setting on health data.

3. Methods/Approach

3.1. BAGAN with Fixed Pose Brain Scans

We used Huang and Jafari's original implementation of the BAGAN-GP architecture for training on our dataset. It is important to note BAGAN neural architecture is designed with the intention to "let the GAN distinguish between different classes explicitly" and learn "useful features from majority classes and use these to generate images for minority classes" [8]. Therefore, an approach we undertook with BAGAN to improve baseline ResNet classification scores was to select brain scans across classes with the same head orientation. By keeping the scan pose in the dataset across classes consistent, our intention was to make it easier for the BAGAN to learn the differences between the tumor classes and further minimize generator/discriminator loss during training. As a result, our expectation was for the BAGAN to generate more realistic images for minority classes as it no longer has to account for different scan poses.

Since the poses are not labeled in the dataset we will need to use unsupervised learning to properly separate all of the poses. We decided to use K-means clustering in order to accomplish this. K-means will group all of the pictures into k clusters, where each cluster represents a group pictures closest to each other in a vector space. To vectorize our images we ran them through a pretrained VGG-16 model and removed the final 2 layers to extract features for each of the images. After that we decided to use PCA to reduce the dimensionality of these vectors as the output of the previous step produces a 4,096 size vector per image. Finally we put the output of PCA through K-means and use the elbow method to find the ideal number of clusters. Since we are clustering the 4 tumors we need to run this for all tumors to find the ideal number across all 4 tumors. This method was taken from the the following article [2].

3.2. DDPM's for Dataset Rebalancing

As this is the first work that investigates the use of DDPM's for this task, we use a relatively simple U-Net architecture similar to the one proposed by Ho et al. [5] for our models. Specifically, we use a U-Net implementation from the HuggingFace `diffusers` library with five regular ResNet blocks along with a single spatial self-attention block each for downsampling and upsampling. We also use an unmodified noise scheduler to the one specified by Ho et al.

Unlike with the BAGAN architecture, minority classes are considered independent during the training and generation process. For this reason, it is necessary to train separate DDPM's for each of the minority-classes in the same way that Dhariwal et al. [1] train separate models for each

class in their experiments. Newer, more complicated architectures may avoid this one-to-one correspondence between minority-classes and models and reduce training and generation time [7]. However, in this work we focus on comparing well-known diffusion architectures to GANs in this domain, so we leave the evaluation of more specialized techniques for future work.

3.3. Comparing Success of Rebalancing Methods with ResNet

ResNet, short for Residual Network, is a computer vision classification system that by default classifies between 1000 different images [4]. This model can be modified by changing the final linear layer with an arbitrary output layer to change the number of outputs. For our case, we took the base ResNet architecture and changed the final linear layer with a linear layer that outputs 4 sections.

We used ResNet as our measurement of success. We imported an untrained ResNet model via Pytorch and trained the model using our skewed and generated data. We then monitored the validation accuracy across data sets that were generated to monitor a generative model's viability to train a classification system.

In our experiments we tested different ways of introducing the generated images are fed into the ResNet for training. We began by simply balancing the data with the newly generated images for all the epochs. Then as an attempt to prevent the ResNet model from focusing on any noise specific to the generative models, we will introduce epochs where no generated data is used. First we will remove the generated data from the training halfway through the training process. Then we will interlace epochs where generated data is used with epochs where the model is trained solely on genuine images. These approaches to how the generated data is introduced are to prevent the ResNet model from learning antipatterns from the noise introduced by the generative models.

4. Data

The dataset we are using was developed with the purpose of detecting brain tumors through MRI scans. The motivation behind the dataset was to use CNN based models to classify and detect brain tumors through MRI scan early so that it can used help treat patients earlier. The data was compiled from 3 datasets. The author threw out some data from one if the datasets as he believed that it was mislabeled and therefore could not be used. There is a total of 7022 images in this dataset with 4 different labels: glioma, meningioma, no tumor, and pituitary. In the training set there are 1321 glioma images, 1339 meningioma images, 1595 no tumor pictures, and 1457 pituitary images. And in the testing set there are 300 glioma images, 306 meningioma images, 405 no tumor pictures, and 300 pituitary images. Each file in this dataset is an image however not all images are the same size or the same number of color channels (some are greyscale others are RGB). There is no sensitive data provided in this dataset it simply the MRI picture and the label associated with it. The original data is preserved in the dataset however due to the fact that the images vary in format there is code provided to prepossessed the data, however that code was not used in this project. The dataset has been previously used for its purpose which is simply to classify and detect brain tumors from MRI pictures. The dataset is publicly available on Kaggle and has a CC0: Public Domain license. The dataset has no plans to be updated and will continue to be available until the user who created it or Kaggle decides to take it down. If someone wants to expand on this work they can add to this dataset and compile data from other datasets that have not already been used.

5. Experiments and Results

5.1. Fixed Pose BAGAN Training

We generated 1000 images per class using our BAGAN with gradient penalty (BAGAN-GP) model that was trained on fixed brain scan poses. For data preprocessing, we downsample our training images from 128 × 128 to 64 × 64 and then normalize pixel values to the [-1, 1] interval to accommodate the architecture of Huang and Jafari's model, which we repurpose for the image generation task. The discriminator and generator models of the BAGAN use identical Adam optimizers with $lr = 0.0002, \beta_1 = 0.5, \beta_2 = 0.9$ and were trained for 50 learning steps. The autoencoder is also uses an ADAM optimizer with $lr = 0.0020, \beta_1 = 0.5, \beta_2 = 0.9$ with mean absolute error loss and was trained for 60 epochs. After training the BAGAN on our skewed dataset, the autoencoder had a final train and validation loss of 0.1221 and 0.1188, respectively. Additionally, the discriminator had train/validation losses of 0.8811/1.2074 and the generator had respective losses of 2.1667/2.2468. In Figure 2, we show examples of generated images using the BAGAN that were used to augment the training dataset.

5.2. Fixing Poses

We fixed our poses by using a pretrained VGG-16, PCA, and K-means. Using the elbow method we found the ideal number for k to be 3. Figure 1 shows the clustering on the pituitary class where each row represents a cluster. Overall the clustering did a good job of grouping the images by pose while making few to no errors depending on the class.

5.3. DDPM Training

As mentioned, we train three separate diffusion models for each of the three minority classes (Glioma, Meningioma, Pituitary). During training, we used an Adam optimizer with $lr = 0.001$, 500 warmup steps with a cosine

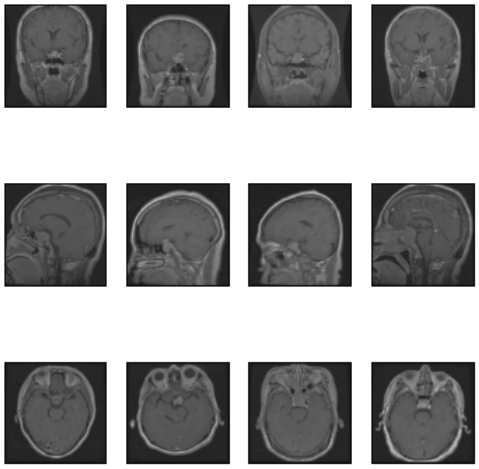

Figure 1. Clustering by pose on Pituitary Class
(Each row represents a cluster)

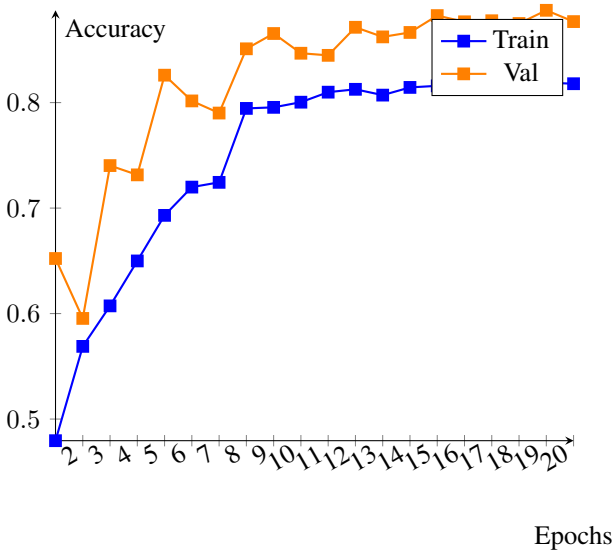

Figure 2. Training of a ResNet18 model using our baseline (skewed) dataset

scheduler, a batch size of 16, over 50 epochs. Due to the nature of training three computationally expensive models, it was infeasible to perform comprehensive hyperparameter tuning on each of these models, However, we did use some manual tuning of the learning rate. We then used this model to generate 1000 images for each of the minority classes. In Figure 4, we show examples of these generated images using the diffusion model that was used to augment the training dataset.

5.4. Resnet Training

After generating the images with diffusion and BAGAN models we then used the generated images to train a ResNet model. We used the non-pretrained ResNet18 model from PyTorch and replaced the final linear layer with a linear layer that has 4 outputs, one for each class.

When running the ResNet for the baseline we simply loaded all the images from our skewed dataset into our dataloader and ran the model for 30 epochs using a SGD optimizer, a learning rate of 0.001 and a momentum of .9 with cross-entropy loss. For all experiments using the generated data, we kept the hyperparamters the the same as well as the testing set. Between the tests the only changes made involved the construction of the training dataset, specifically in regards to using the generated data. This is to avoid uncertainty about the source of any change in performance of the ResNet model being from introducing generated data versus hyperparameter changes.

Figure 2 shows the change in accuracy over epochs during the training of the ResNet model on our baseline dataset. The baseline model finishes with a maximum validation accuracy 88.875%. The model begins with a 65.22 and climbs quickly for the first 8 epochs to reach a 85.10% validation accuracy. Then for for the remaining epochs the model climbs slowly to make a 3.775% improvement over the remaining 12 epochs.

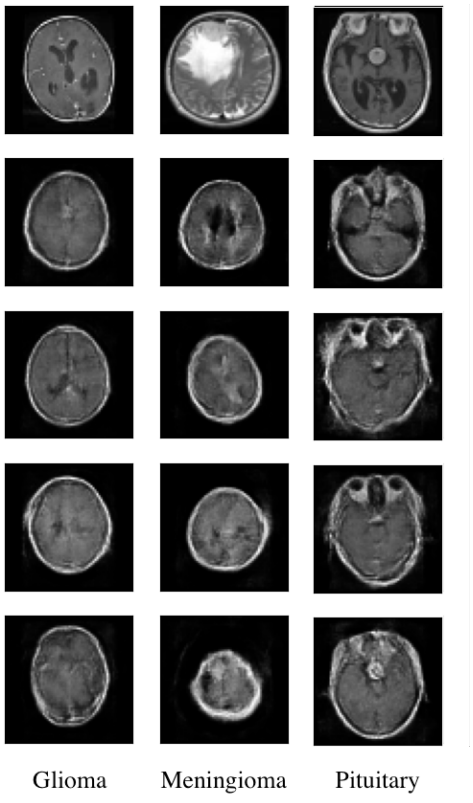

Figure 3. Generated Images From BAGAN Model
(Top Row are Original Images)

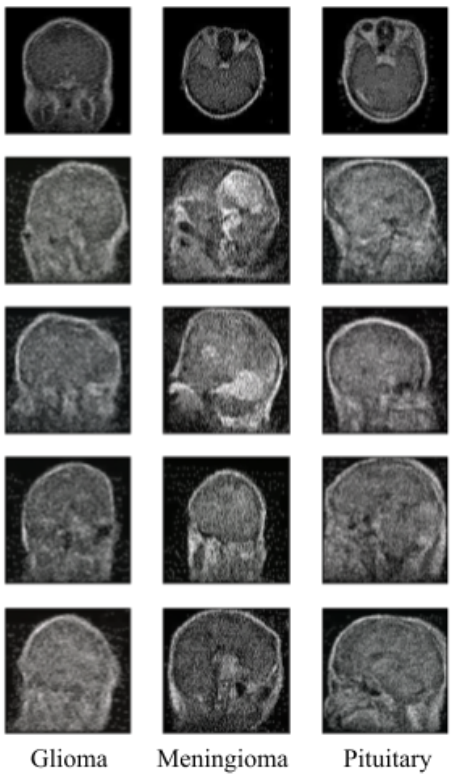

Figure 4. Generated Images From Diffusion Model (Top Row are Original Images)

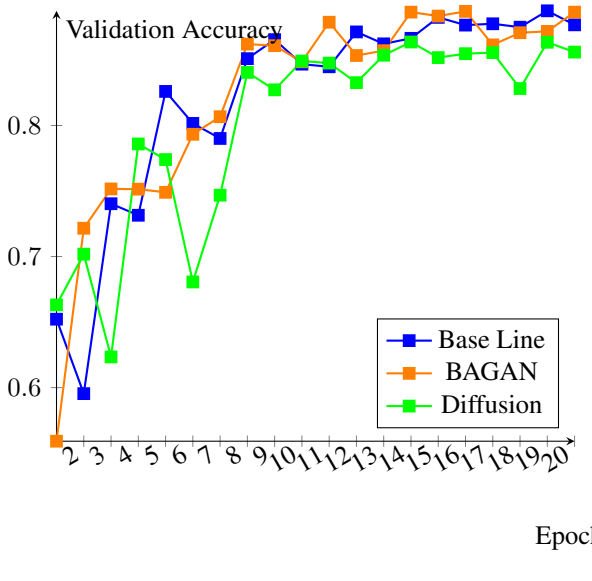

Figure 5. Training Curve of ResNet18 with Generated image supplemented data set

Figure 5 shows the validation accuracy of the ResNet over the epochs of their training. In this experiment

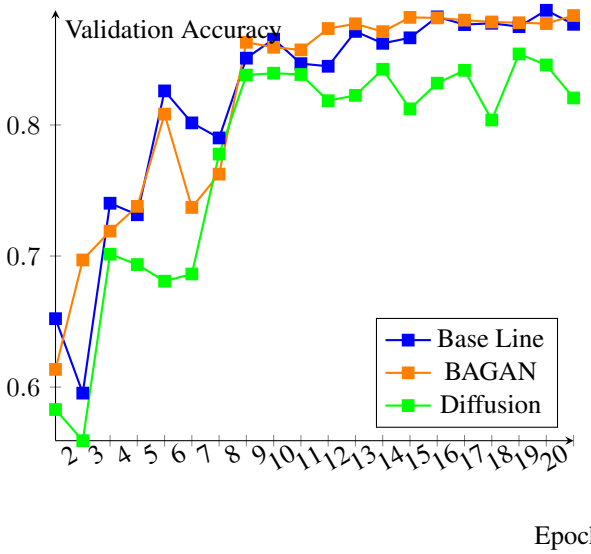

Figure 6. Training Curve of ResNet18 with Generated image supplemented data for epochs 1-10

images generated from BAGAN and diffusion models were introduced into the dataset to balance out the skewed state of the data set. The ResNet with BAGAN images introduced had a final validation accuracy of 88.7097%. Meanwhile introducing diffusion generated images resulted in a peak validation accuracy of 86.3710%. While there are slight differences in the peak accuracy of the models depending on the source of generated images or the lack thereof, the change only varies by 2.504%. This change is small compared to the changes the models can make between epochs, and considering the overall trend trend of the models we can see the performance overall remain rather similar. All three models trained start with quick growth into epoch 8 then proceed with considerably slower growth to reach their final maximums. Figure 5 shows that epochs continue the performance of the models look to converge closer. The main difference is that diffusion model introduces larger swings of performance in between the epochs especially before the first 8 epochs. This may suggest that diffusion is introducing some form of regularization where the noise contained in diffusion images disrupts the detected patterns that ResNet begins to learn. Meanwhile the ResNet model trained on BAGAN images seems to have much more stability beteen the epochs. The model seems to lack the high jumps in perfomance that the base line contains as seen in epoch 5 as well as the sharp dips that the base line and diffusion model contain.

In pursuit of finding some way to use generated images to provide a boost to the performance of the ResNet model, we tried two alternative ways of introducing the generated

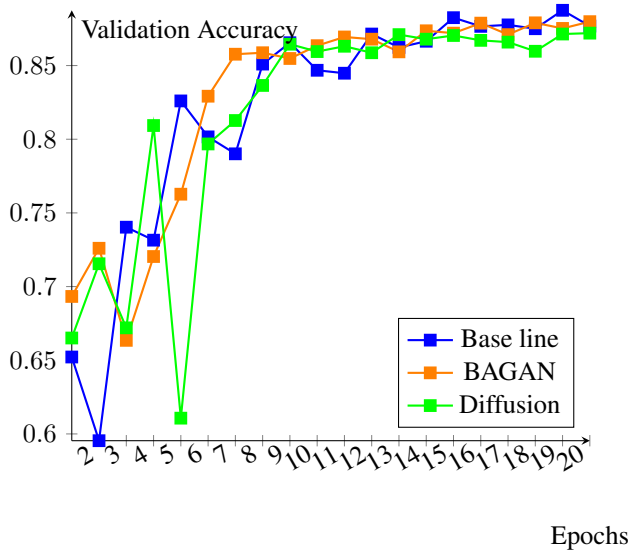

Figure 7. Training Curve of ResNet18 with Generated image supplemented removed on every 4th epoch starting at 1

data into the training process. Figure 6 shows the attempt to use generative images as a form of pretraining. For the first 10 epochs the ResNet model was trained on a data set containing both generated and real images. However at the 10th epoch the generated data was removed leaving only the original skewed data set to train on. The model trained on diffusion data reached 85.4234% validation accuracy while the model trained on BAGAN images reached 88.3468% validation accuracy. Both models exhibited a slightly decrease in performance when the generated images were introduced to the training data set in this manner. This loss in performance is especially notable in the model trained on diffusion generated images. Once the diffusion resnet loss the generated images on epoch 10, its performance began to diverge from the BAGAN and Base performance as it began to decrease in performance over its epochs. This loss in performance in the ResNet models may be a consequence of the model having to effectively unlearn peculiarities that resulted in the given generative model's processes. This would have, in effect, distracted the model from learning the patterns that are contained in the genuine images more than the noise would have caused if the generated images were left there throughout the training process.

In an attempt to reduce the effect that random noise introduced by the generative models, we trained the ResNet model on a training loop the interlaced epochs where generated images were used with epochs where no generated images were used in the training. The exact interlacing pattern was, the model ran the first epoch with no generated images, and for the next 3 epochs, then on the fourth epoch, or epoch number 5 when indexing at 1, the model will training using exclusively genuine images. This means there was a 3:1 ratio of genuine supplemented with generated data to exclusively genuine data. Handling the data this way led to the diffusion ResNet to reach 87.1976% validation accuracy while the BAGAN supplement ResNet reached 87.9839% validation accuracy. Figure 7 displays the learning curve of the training using this method. The figure shows that training the data in this fashion does not remove the inital instability of the Diffusion supplemented ResNet model. However notably, around epoch 8 the diffusion supported ResNet begins converges with the other models far more closely than in any previous way of handling the generated data. This may be due to the model being slightly pushed away from learning noise introduced by the diffusion model into the data set by emphasizing the genuine images over the generated every few epochs. However unlike in Figure 6 by not completely cutting off the training from the generated images after a certain number of epochs we avoided the model antipatterns early in the training.

6. Conclusion

At this stage in experimentation, we showed that introducing generated data points to the data set would not ruin the predictive ability of the ResNet model. However, in training the data sets we see negligible change likely due to the small amount of generated images. Going forwards we aim to generate enough images using both methods to balance our skewed data set.

In terms of image generation, we have seen great success in attempts of generating new tumor MRI images. We now have a working generative model that we can now use to continue to make our balanced data sets.

Overall, our experiments have shown that introducing generated images into a data set have poor impact on the performance in training a classification system. In every instance of training with generated data little change occurs and change has always been negative on the performance of the model. Introducing the generated data to supplement the underrepresented data led to slightly inferior results. This may be due to the fact that a given image only contains a certain amount of information, or entropy, that can be applied to the classification problem. The diffusion or BAGAN models cannot create new information or patterns since they are in themselves trained on the same training data set as the classification system is. Thus in terms of information, the benefit that they may provide is that they can evoke patterns already present, but concealed in the data that the classification model may struggle to learn from the data directly. However, it seems our results imply that the ResNet model on its own is able to extract the information and patterns as well as the generative models or at least well

enough where any noise from that the generative models create in their images offsets the gains from the processed data.

If one model is chosen, our results in figure 5 and 6 suggest the BAGAN models produce slightly higher accuracy rates compared to the diffusion model. This may be because the BAGAN model exaggerates differences between classes. By training the BAGAN model on only a single perspective, the BAGAN may have more easily exaggerated the patterns unique to the given tumor since all else was kept as equal as possible.

However, as a result of our testing, we do not recommend using generative models to supplement this particular skewed data set when training a classification system, or at least not one as sophisticated as ResNet. While previous literature has found a benefit in using generative models for this purpose, it is possible that the lack of distinctive classes in the dataset used for evaluation contributed to this result. Additionally, it may be the case that, in a similar manner to the vanilla GAN, the vanilla DDPM used in our experiments is not specialized enough to generate images that accentuate class differences. Future work could investigate developing such a specialized architecture in the same way that the BAGAN approach was developed from failures of the vanilla GAN architecture, and also study the impact that the chosen dataset could have on results. Additionally, future work could also investigate the impact of the use of focal loss instead of cross-entropy loss for the ResNet training. Focal loss is an alternative to cross-entropy loss that attempts to deal with imbalanced datasets by assigning larger weights to data samples that are harder to classify [**?**]. Using this loss might help improve accuracy, especially on the rebaal Finally, in practice, as generating the images greatly expands the computational costs and time of training a classification model, any performance increase (or lack thereof) should be justifiable.

7. Individual Contributions

Team member contributions can be found in Table 1.

Name	**Contribution**
Shaan Gill	I helped set up the diffusion model with Ethan and implemented the clustering by pose. My work can be seen in sections 3.1 and 5.2 where I took data and found the ideal clusters to be used for further processing for the BAGAN.
Charles Snider	I implemented the ResNet Model. In addition, I experiment with different ways to introduce the generated images into the data set to improve our results which required editing the training loop. My experiments with introducing the data to the ResNet model are described in sections 3.3 and 5.4. Essentially, after my teammates generated the images I handled experimenting with images to get out final results.
Manas Harbola	I worked on repurposing an existing BAGAN model to generate images using skewed dataset. I focused on tuning the default hyperparameters of the model's autoencoder, discriminator, generator to and worked closely with Shaan to train the BAGAN on fixed pose brain scans and provided Charles with set of generated images from minority classes. More info on my work can be found on sections 3.1 and 5.1.
Ethan Mendes	I primarily worked on training the three diffusion models with Shaan and generating images with these models. During this work, I tried out different architectures and hyperparameters and adapted to challenges related to working with limited computing power. Discussion about my contributions can be found in sections 3.2 and 5.3.

Table 1. Contributions

References

[1] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 1, 2

[2] Gabe Flomo. How to cluster images based on visual similarity, 2020. 2

[3] Ramin Ghorbani and Rouzbeh Ghousi. Comparing different resampling methods in predicting students' performance using machine learning techniques. *IEEE Access*, 8:67899–67911, 2020. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 1, 3

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. 2

[6] Jafari Huang. Enhanced balancing gan: minority-class image generation, 2021. 2

[7] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed, 2021. 3

[8] Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Costas Bekas, and Cristiano Malossi. Bagan: Data augmentation with balancing gan, 2018. 1, 2

[9] Masoud Nickparvar. Brain tumor mri dataset. 2021. 1

[10] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *CoRR*, abs/1712.04621, 2017. 2

[11] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans, 2016. 2