Active Recall Networks for Multiperspectivity Learning through Shared Latent Space Optimization

Theus H. Aspiras, Ruixu Liu and Vijayan K. Asari
Electrical and Computer Engineering, University of Dayton, 300 College Park, Dayton, U.S.A.

Keywords: Generative Adversarial Networks, Convolutional Neural Networks, Variational Autoencoders, Unsupervised Learning, Semi-Supervised Learning.

Abstract: Given that there are numerous amounts of unlabeled data available for usage in training neural networks, it is desirable to implement a neural network architecture and training paradigm to maximize the ability of the latent space representation. Through multiple perspectives of the latent space using adversarial learning and autoencoding, data requirements can be reduced, which improves learning ability across domains. The entire goal of the proposed work is not to train exhaustively, but to train with multiperspectivity. We propose a new neural network architecture called Active Recall Network (ARN) for learning with less labels by optimizing the latent space. This neural network architecture learns latent space features of unlabeled data by using a fusion framework of an autoencoder and a generative adversarial network. Variations in the latent space representations will be captured and modeled by generation, discrimination, and reconstruction strategies in the network using both unlabeled and labeled data. Performance evaluations conducted on the proposed ARN architectures with two popular datasets demonstrated promising results in terms of generative capabilities and latent space effectiveness. Through the multiple perspectives that are embedded in ARN, we envision that this architecture will be incredibly versatile in every application that requires learning with less labels.

1 INTRODUCTION

With limited labeled data, an unsupervised training strategy must be developed to determine the latent space representations of the data, which learns the available features of the data without any labeled information. Data must be encoded into a latent space and decoded for replicating the input, which creates a nonlinear dimensionality reduction mapping from input space to latent space. The ability of the network to faithfully recreate the input from the latent space representation means that the network has already learned all of the necessary features of the data embedded in the latent space for any task, including detection, classification, and recognition capabilities. The features extracted in an unsupervised fashion become extendable for supervised tasks, through which the neural network training manipulates the latent space for the supervised data. From only a small amount of labeled data, the entire latent space representation can be partitioned into respective classes.

Within the shared latent space lies the ability to both discriminate and generate information. Using multiple training criteria would bias the latent space representations for reinforcing different associations and features, which have been shown to be effective in human psychology. It is therefore valuable to utilize a multi-cost optimization of the latent space using these various criteria. Previous approaches separate cost functions and derive other architectures in order to optimize a specific application, but new research points to multi-cost and competing cost optimization, which works incredibly well among different datasets for detection, recognition, generation, and other tasks. The incorporation of multiperspectivity allows the use of different spaces of the network: image space, latent space, and task space. Figure 1 shows an example of this multiperspectivity architecture.

The contribution of this paper is to present a neural network architecture with a shared optimized latent space. The active recall network's (ARN) latent space is trained with adversarial examples through the generative capabilities of the network. By combining the encoding of data to a lower dimensional space and discriminating between real samples and adversarial examples, the interpolation of data points within the real distribution will be more indicative of relative image features rather than just encoding.

For example, an interpolation between two real data points for an unsupervised variational autoen-

Aspiras, T., Liu, R. and Asari, V.
Active Recall Networks for Multiperspectivity Learning through Shared Latent Space Optimization.
DOI: 10.5220/0008369304340443
In *Proceedings of the 11th International Joint Conference on Computational Intelligence (IJCCI 2019)*, pages 434-443
ISBN: 978-989-758-384-1

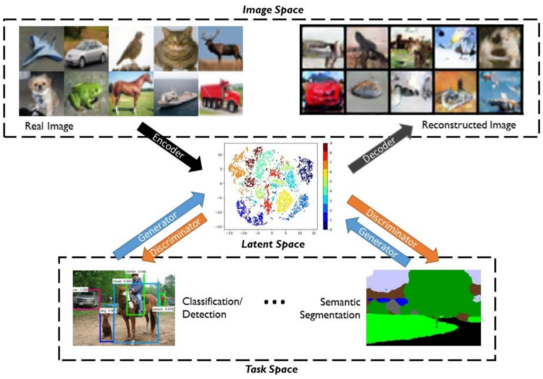


Figure 1: Architecture for multiperspectivity learning. Latent space representations become richer and more generalizable through the unsupervised/supervised training and discriminative/generative abilities of the network.

coder (VAE) is naturally a mean image between these datapoints, which is due to modeling each data projected in the latent space as a Gaussian distribution in the dataset and the loss function of the autoencoder. Unsupervised adversarial autoencoders (AAE) for developing the latent space only regularizes the entire distribution towards a configured distribution, but does not address the interpolation between datapoints (AAE incorporates variational autoencoders as well).

The ARN addresses this interpolation by training the encoding portion adversarially. Through the generation of these interpolations within the latent space, the network train both the encoder and the simple discriminator (combined to mimic the discriminator network of GAN architectures) to maximize the image-level features being encoded into the latent space. Therefore, the encoding of the images to the ARN latent space takes into account the defining features of the real distribution from the training of the encoder/discriminator. Our approach aims to optimize the feature extraction of the network using multiple loss functions.

2 RELATED WORK

For developing a suitable architecture for shared latent space optimization, we must consider various human perspectives that facilitate learning capabilities.

Psychologically, the retention of memory is through the creation of associations. Gobet (Gobet and Simon, 1998) created an experiment to determine the ability to memorize information from various skill levels. When chess pieces were arranged from previously played games, expert chess players were able to memorize most chess piece locations based on associations of games they have played, but novice and intermediate chess players had much less ability for memorization. In neural networks, it is the development of the latent space that creates the ability for association. Effective memorization is not just the ability to develop various associations, but also to retrieve these associations. The generative ability of the brain to retrieve images is impressive. Several studies (Bihan et al., 1993), (Ogawa et al., 1992) have considered the human ability to recall various images and determined that the same cortex activations during these cognitive processes are similar to activations involved in visual perception. O'Craven and Kanwisher (O'Craven and Kanwisher, 2000) demonstrated that the occipito-temporal cortex activates during both visual face stimuli and imagining faces. This strengthens the consideration for generative models, which are necessary for visualization of previous trained information.

2.1 Active Recall

Active recall (Karpicke and Roediger, 2008) is the learning methodology which claims that memory should be stimulated during the learning process. This learning process is different from passive review, in which memories are processed passively through just input alone. Several studies have been conducted that support the improvement of learning in humans through active recall in contrast to passive learning. Karpicke and Blunt (Karpicke and Blunt, 2011) evaluated the effectiveness of two studying methodologies, elaborative studying with concept mapping or retrieval-heavy studying. It was found that students that applied more retrieval techniques did 50% better on tests than others who applied more concept mapping techniques, even being tested on the creation of concept maps. Standard training protocols for neural networks utilize passive review (concept-mapping) for training weight structures. Through continuous inputs and reiteration, neural networks are optimized for the specific applications. Autoencoders develop a latent space similar to the creation of concept maps. Generative adversarial networks train in a fashion similar to active recall (retrieval-heavy), which necessitates the generation of the answer with the psychological testing effects of the discriminator. Therefore, it is vital to develop a unified neural network architecture that embraces both passive review and active recall. This inspired the creation of the proposed network, active recall network (ARN), that encompasses encoder/decoder based review and generator/discriminator based recall.

2.2 Current Neural Network Methodologies

Autoencoders: Autoencoders(Rumelhart and Zipser, 1985) have two parameterized functions: an encoder function which allows a transformation from the image space to the latent space and a decoder function which transforms the latent space back to the image space through training on the cross-entropy reconstruction loss. Much research has been conducted on improving autoencoder methodologies like restricted Boltzmann machines for pre-training Deep Belief Networks (Bengio et al.,) and variational autoencoders (Kingma and Welling, 2013), which have been popular with the deep learning community. Variational autoencoders utilize the assumption that samples can be modeled as a Gaussian distribution, which will aid the generative qualities of the autoencoder. Variational autoencoders have several variants like disentangled VAE (Li et al., 2017) which improve the encoding scheme of the autoencoder through categorical information.

Generative Adversarial Networks: Generative adversarial networks (Goodfellow et al., 2014) aim to model samples from a real distribution by utilizing a generator to produce an approximate distribution. For generative adversarial networks, there are two functions that are utilized for training: a generator for creating data from a latent space with a noise sample onto the image space, and a discriminator for transforming the image space for discriminating real and fake samples from real and generated data respectively. This training strategy uses a min-max optimization to update both the generator and discriminator to determine differences between real and fake information. Several GAN variants have been proposed such as Bidirectional GAN (Donahue et al., 2016) for learning the inverse mapping of the latent space, InfoGAN (Chen et al., 2016) to learn disentangled representations, CycleGAN (Zhu et al., 2017) for domain adaptation, and deep convolutional GAN (Radford et al., 2015) for high fidelity mapping. VAE-GAN (Larsen et al., 2015) utilizes a shared latent space between the generator and the decoder of the VAE in a combined network.

Adversarial Autoencoders: Adversarial autoencoders (Makhzani et al., 2015) are similar to GAN architectures, which utilize a discriminator to train the generator network. These networks adversarially train the encoder of the network (which is the generator) to match a specific distribution as described by the user using a trained discriminator function. Adversarially regularized autoencoders (ARAE) (Zhao et al., 2017), an extension of adversarial autoencoders proposed by Zhao et al., minimize the reconstruction loss with the minimization of the Wasserstein distance between the distribution from the encoder and a prior distribution, which is trained with coordinated descent across three cost functions. The adversarial generator encoder (AGE) (Ulyanov et al., 2017) network combines both adversarial and reconstruction losses into a single unifying architecture. Since this network is similar to our intended goal, we will utilize this architecture as a basis for our proposed network.

Training Cost Functions: Several cost/loss functions have been proposed for generative networks. AEs utilize reconstruction loss to train the encoding/decoding weights of the network. GANs are based on classification of real/fake samples, thus require min/max loss function to train the network. Vanilla GANs use Binary Cross Entropy, while other GAN methodologies use Wasserstein distance for loss with weight clipping or gradient penalty. Recent trends in cost/loss functions are placed in relativistic GANs (Jolicoeur-Martineau, 2018), which places emphasis on the difference between real and fake samples. Other cost functions, like triplet loss, utilize an anchor to quantify distance between positive and negative samples.

2.2.1 Discussion

AE - Blurry Imagery: AE networks used for nonlinear dimensionality reduction of imagery provide an encoding to generate imagery, given the same input image. Therefore, by providing the right values in the low dimensional space, the proper reconstruction can be displayed. Any deviation from the encoded value using normal AE networks provide noisy outputs due to a lack of required interpolation between images. Variational AEs try to alleviate this problem by modeling the projection of the data as a specific noise variable (usually Gaussian), thus providing connections between datapoints to generate viable interpolations and generations between datapoints. As currently found with VAEs, these reconstructions of the interpolations are blurred images due to the Gaussian noise, but this interpolation is not based on the entire sampled space. It should be noted that only the interpolations between images are blurred, not the end-to-end reconstruction of the real image.

GANs by nature are interpolations of the image space. Through the constraining of the entire randomly sampled space towards the real data, the interpolations will generate closer towards the actual distribution. To utilize this, we can provide a discriminative/generative space within the AE to properly interpolate points of the randomly sampled latent space and generate images closer to the real dis-

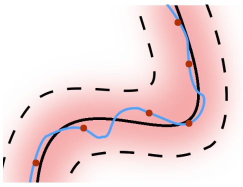


Figure 2: The interpolation of the latent space using VAEs (black line) and GANs (blue line).

tribution. Figure 2 shows the interpolation possible through VAEs and GANs.

This blurriness is also due to the reconstruction loss usually used for autoencoder networks. This reconstruction loss is based on pixel-wise L1-loss, which works well in converging pixel information towards a given output, but does not work well in considering losses in features. Deep Feature Consistent Variational Autoencoders (Hou et al., 2016) utilize a feature-based reconstruction loss from a pretrained deep neural network. The pixel-wise information in the deep features of the network provide better reconstruction loss for the network, which translates to feature reconstruction rather than purely pixel-wise reconstruction. AGE network also include this reconstruction within the loss functions as encoder-generator reciprocity. We will utilize the encoder-generator reciprocity training paradigm for our proposed network.

GAN - Mode Collapse: GAN networks are notorious for mode collapse, where the generator of the GAN that is unable to produce the full range of samples across the distribution, only focusing on a specific subset for generation. This is due to two vital prospects: The discriminator works too well discriminating between some real and fake images and the generator has a one-sided gradient. It has been found that regular GAN cost functions collapse to specific modes because the generator has insufficient gradient to move towards the real distribution. Even through initialization, partial mode collapse has already started due to the generator not being constrained to generate all of the distribution samples, though partial mode collapse is ideal for quality generation. The generator is only constrained to have its distribution within the discriminator distribution.

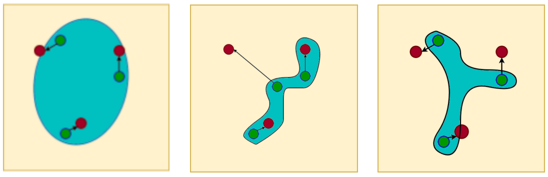
To alleviate mode collapse, the discriminator can utilize Wasserstein distance to promote the learning of the generator through restored gradients. The generator can utilize the AE reconstruction loss to constrain its generation to recreate all of the real distribution samples. Figure 3 demonstrates this concept of alleviating mode collapse through AE reconstruction. Given an initial generative distribution (3a), samples that are generated by GAN will only be similar to real samples it has contained within its distribution (3b). Samples that are not generated by the generator will be pushed towards the real sample through reconstruction loss.

Multiperspectivity Learning: The combination of reconstruction loss and adversarial loss must be properly configured to converge correctly. Given three datapoints on a shared latent space (real data, reconstructed data, and generate/fake data), a harmonious relationships should be established. Reconstruction loss is usually configured as the minimization between the reconstructed loss and the real data. Adversarial loss, on the other hand, can be configured in many ways. Typical adversarial loss is defined as a min/max loss of real and generated data classification. As discussed earlier, a generator has the ability to generate reconstructed data, thus can be assumed that all reconstructed data lies within the generator distribution. A generator may not be able to generate real data. Therefore, if the discriminator is too powerful in discriminating between real and generated data, the reconstructed points will also be discriminated from the real data, which may still provide a potential for mode collapse. Triplet loss can be used for this purpose through minimizing real and reconstructed data distance and maximizing real and generated data distance, but this loss will not create the same kind of convergence as vanilla GANs.

The best configuration for adversarial loss is to make sure that all samples lie within both the discriminator and generator distributions, which will fine tune the distribution towards the samples. In this case, all reconstructed points should lie within the generator/discriminator distributions. Therefore, if these reconstructed points are known, it is best provide an adversarial loss between the reconstructed data and the generated data. All of the reconstructed samples will minimize their distance between their real samples and the generated samples will be trained adversarially towards the reconstruction samples, as shown in Figure 3c. With the discrimination of reconstruction points (which are always within the generator's distribution), it then becomes not necessary to implement different loss variants, like Wasserstein distance, due to the gradient always being present. This methodology sufficiently alleviates mode collapse that is inherent in GANs.

3 ACTIVE RECALL NETWORK

We propose an active recall network (ARN), which combines the cost functions of an autoencoder with a GAN in a fused network architecture. Figure 4 shows



(a) Initial (b) GAN (c) Proposed

Figure 3: Active Recall Network Architecture. Both AE and GAN architectures share encoder, decoder, and latent space weights for unsupervised training.

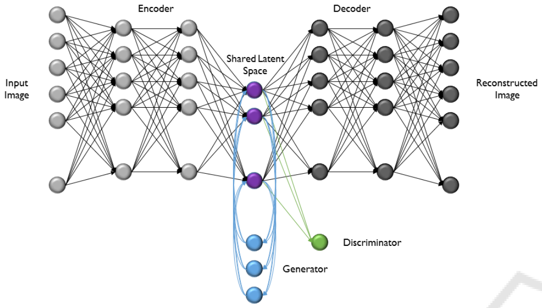


Figure 4: Active Recall Network Architecture. Both AE and GAN architectures share encoder, decoder, and latent space weights for unsupervised training.

the framework of the proposed neural network architecture.

It should be noted that the shared latent space is not created by the combination of the encoder and generator together, but rather projects onto the latent space exclusively. The ARN uses the AE loss function to minimize the reconstruction error and the GAN loss function to minimize and maximize the Kullback-Leibler divergence of the generator and discriminator respectively. The ARN is fashioned to share latent space representations and encoder/decoder architectures to optimize a latent space for both regression and classification. The ARN model is trained across three cost functions, L_{ae}, L_{dis}, and L_{gen}, representing autoencoder, discriminator, and generator respectively:

$$\min_{\phi,\psi} L_{ae}(\phi,\psi) = E_{x\sim P_*}[-log\ p_\psi(x|enc_\phi(x))] \quad (1)$$

$$\max_{\phi,w} L_{dis}(\phi,w) = E_{x\sim P_*}[f_w(enc_\phi(\tilde{x}))] - E_{\tilde{z}\sim P_\circ}[f_w(\tilde{z})] \quad (2)$$

$$\min_{\psi,\theta} L_{gen}(\psi,\theta) = -E_{\tilde{z}\sim P_\circ}[f_w(\tilde{z})] \quad (3)$$

Where $enc_\phi(x)$ is latent space sample z from a real distribution P_*, p_ψ is the reconstruction probability, $\tilde{x}$ is the reconstructed sample from the real image, $g_\theta(s)$ is a generator which creates data onto a latent space from a noise sample s, $dec_\psi(z)$ is a reconstruction of the image x from the latent space sample z, $f_w(z)$ is a discriminator in the latent space, and $\tilde{z} = enc_\phi(dec_\psi(g_\theta(s)))$ is a latent space sample from the generated distribution $P_\circ$. Generally, it is more optimal to generate and discriminate from fixed distributions like a Gaussian distribution, which are embedded within the training of the ARN. Adversarially regularized autoencoders generate a parameterized distribution, which is more complex for obtaining a better solution. The ARN can utilize any type of prior distribution, but will specifically aim to create a parameterized solution from within the latent space. Through the optimization of the network, the optimal solution for the generator should become projecting samples only from the discriminative region of the latent space. Table 1 shows the training paradigm of the unsupervised ARN network.

Table 1: Active Recall Network training paradigm.

```
Algorithm for ARN Training
for each training iteration do
    (1) Train the enc./dec. for reconstruction $(\phi,\psi)$
        Sample $\{x^{(i)}\}_{i=1}^{m} \sim P_*$
        Compute $z^{(i)} = enc_\phi(x^{(i)})$
        Backprop loss, $\frac{1}{m}\sum_{i=1}^{m} log\ p_\psi(x^{(i)}|z^{(i)})$
    (2) Train the encoder/discriminator $(\phi,w)$
        Sample $\{s^{(i)}\}_{i=1}^{m} \sim N(0,1)$
        Compute $\tilde{z}^{(i)} = enc_\phi(dec_\psi(g_\theta(s^{(i)})))$
        Backprop loss, $\frac{1}{m}\sum_{i=1}^{m} f_w(enc_\phi(\tilde{x}^{(i)}))$
            $-\frac{1}{m}\sum_{i=1}^{m} f_w(\tilde{z}^{(i)})$
    (3) Train the generator/decoder adversarially $(\psi,\theta)$
        Backprop loss, $\frac{1}{m}\sum_{i=1}^{m} f_w(\tilde{z}^{(i)})$
end for
```

It is found that the GAN architecture utilizes a min-max criteria to optimize the generator and discriminator, but with the combination of cost functions with the autoencoder, there is also another min-max criteria inherent in the architecture that is not explicitly described. Linear discriminant analysis (LDA) (McLachlan, 2004) aims to maximize interclass differences, which in the case of GAN networks aims to maximize real/fake class differences. A symptom of this maximization is the minimization of intraclass differences, which is more explicitly described in Fischer's linear discriminants (Fisher, 1936). As discussed earlier, an autoencoder aims to minimize the reconstruction error of the network. A symptom of this minimization is the maximization of intraclass variations to reconstruct the information. Principal component analysis (PCA) (F.R.S, 1901) is a maximization of variation to improve encoding ability of the network, which is somewhat similar to autoencoders. These competing cost functions will allow the best representation learning for both regression and classification. According to Martinez et al. (Martinez and Kak, 2001), it has been found that discriminant analysis works better for generalizing larger datasets while PCA works better in smaller datasets. Therefore, ARN utilizes the strengths of both types of data associations.

3.1 ARN Variants

ARNs are easily extendable to variational and convolutional variants. The reparameterization trick applied to ARNs will have a better ability to generalize the distribution through the modeling of the data points in the latent space as a Gaussian distribution. By utilizing this, the modeled distribution will be more connected, which may aid in the generation of the data. The convolutional variation will create a latent space as a receptive field, which allows for generation as a receptive field rather than as the entire image. For more complex datasets, we will utilize the convolutional variants in a combined architecture called deep convolutional ARN (DCARN).

3.2 ARN Strengths

ARN encodes into a discriminative latent space as opposed to AE and VAE. AAE performs discrimination by modeling a priori distribution. It performs encoding into a latent space then discriminates the latent space into a priori distribution. On the other hand, ARN performs discrimination in the latent space by modeling a learned distribution from adversarial examples. That is, it performs a discriminative encoding to generate a distributed latent space. This process of discriminative encoding will generate an accurate distributed latent space when compared to the latent space generated by AAE. The discriminative encoding is performed by employing the reconstruction and adversarial loss functions. In ARN, the distributed encoding will encode not only the actual data points but also the virtual data points that may occur between the actual data points through a distribution manifold. This manifold of visual perception is created by a set of virtual data formed by the image features extracted during the distributed encoding process.

The main difference between the ARN and the AE variants is the modeling strategy of the virtual data. For AE variants, the virtual data points are modeled from a Gaussian Mixture Model (GMM), which joins the real data points through the overlapping of the Gaussian functions within the latent space to create the distributed manifold. The ARN is able to model these virtual data points based on the extraction of image features from adversarial examples by discriminative encoding, thus creating a distribution manifold which more accurately models image features.

With the strengths of the AE and GAN architectures being combined in ARN into a single neural network, it is envisaged to be used for several different extensions, like domain adaptation (due to distribution learning) and automatic model configuration (due to better trainability of the network), which will be easier for implementation than other neural network architectures.

3.3 Supervised Learning using ARNs

With classification through labeled information, the discriminator will be extended to compute the probabilities of all classes along with the determination of real/fake data. In this case, the classifier has three areas of information to train the network: (1) real images with labels, which can be trained like in any regular supervised classification problem, (2) real images without labels, which can be trained as unsupervised, and (3) images from the generator, which the discriminator learns to classify as fake. In essence, the active recall network trained in an unsupervised fashion is able to discriminate and generate the real data distribution. Therefore, it is only needed to train the class discriminator for only the labeled data and to train the generator with the given class. Given the shared latent space, it is expected that discriminating only the labeled data provides sufficient information to discriminate the entire unlabeled dataset, which provides values for labels of all information. The shared latent space provides a way to remove overfitting the specific labeled information so that class discrimination can be generalized to the entire distribution. The generator of the network can also be trained by constraining one of the random components of the distribution for creating the class. This allows generation of the data within the specific class distribution.

Therefore, the discriminator must calculate both unsupervised loss and supervised loss based on the data. Labeled information accounts for both loss types while unlabeled only accounts for unsupervised loss. Many neural network architectures utilize the supervised methodology, which can be tested and evaluated using many different datasets. For creating a supervised ARN, another cost function is added to reduce the classification error of the data and includes the ability to generate samples of a specific class using a random sample. We define several supervised variants for the ARN network. Equation 4 trains only the class discriminator.

$$\max_{t} L_{disc_c}(t) = E_{x \sim P_*}[log(h_t(enc_\phi(x_c)))] \tag{4}$$

Where c is the class labels and h_t is the class discriminator. If the necessity of the network is to generalize the class information over the entire distribution, only the class discriminator should be used for updating. Equation 5 trains both the class discriminator and the encoder portion of the network.

$$\max_{\phi,t} L_{disc_c}(\phi,t) = E_{x \sim P_*}[log(h_t(enc_\phi(x_c)))] \tag{5}$$

If the purpose is to focus the class discriminator to correctly classify the given labels, the class discriminator and the encoder of the network should be updated. Equations 6 and 7 trains the entire network to adversarially train class information.

$$\begin{aligned}\max_{\phi,t} L_{disc_c}(\phi,t) = E_{x \sim P_*}[log(h_t(enc_\phi(x_c)))] \\ + E_{\tilde{z} \sim P_\circ}[log(1-h_t(\tilde{z}))]\end{aligned} \tag{6}$$

$$\min_{\psi,\tau} L_{gen_c}(\psi,\tau) = E_{\tilde{z} \sim P_\circ}[log(1-h_t(\tilde{z}))] \tag{7}$$

Where τ is the class generator. If the network should be constrained to only include given labels, the class information can be used to generate and discriminate classes to adversarially training the network. Depending on the purpose of the class discriminator, various training scenarios are possible.

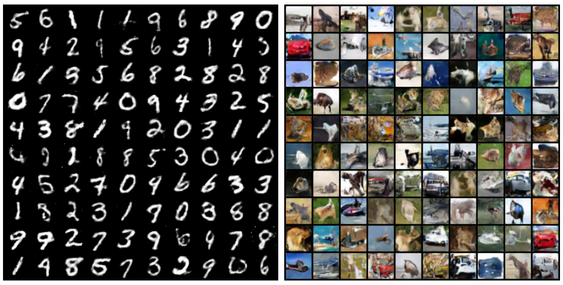
4 RESULTS

We have trained the active recall network on two different datasets, CIFAR-10 (Krizhevsky, 2009) and MNIST (Lecun et al., 1998), for evaluating the generative and latent space encoding performance of the proposed architecture.

4.1 Generative Characteristics Results

We have implemented different configurations of the active recall networks to determine the generative characteristics of the network for unsupervised learning. The first is the unsupervised ARN implemented for learning the MNIST handwritten digit database. The generative capability of the ARN in reproducing the MNIST dataset is presented in Figure 5a. It is evident that the visual quality of the reproduced images is comparable to the results produced by other generative architectures.

We have also implemented the DCARN for unsupervised learning of the CIFAR-10 tiny color image database. The generative capabilities of the DCARN in reproducing the CIFAR-10 dataset is presented in Figure 5b. The visual quality is observed to be better than AE architectures and comparable to GAN models.



(a) MNIST (b) CIFAR10-DCARN

Figure 5: MNIST and CIFAR-10 results using the ARN and DCARN architecture respectively.

Tables 2 and 3 shows the metrics for the DCARN network against other generative neural networks. The inception score presented in Table 2 for the DCARN network is shown to perform slightly worse, but considering recent research on inception score for evaluating GAN architectures (Lucic et al., 2017; Heusel et al., 2017), the Fréchet Inception Distance (FID) score provides a better measure of generative performance. The FID metrics of the DCARN presented in Table 3 show good performance compared to generative adversarial architectures. The DCARN performs better than the DCGAN, but not as well as the WGAN (Arjovsky et al., 2017) architecture. The ability of the network to create quality generations is still limited, as discussed using VAE-GAN hybrid architectures, but with the ARN architecture, flexibility in learning rate between the adversarial loss and the reconstruction loss can be tuned to provide better performance in different areas. Furthermore, the generative capabilities of the ARN network are still very good, but are more useful for the next evaluation: latent space encoding.

We utilized the DCARN for unsupervised generation on CelebA Face dataset (Liu et al., 2015) and horses from ImageNet (Deng et al., 2009), as shown in Figure 6. The DCARN network has the ability to provide good convergence of adversarial loss functions using the reconstructed datapoints.

4.2 Latent Space Encoding Characteristics Results

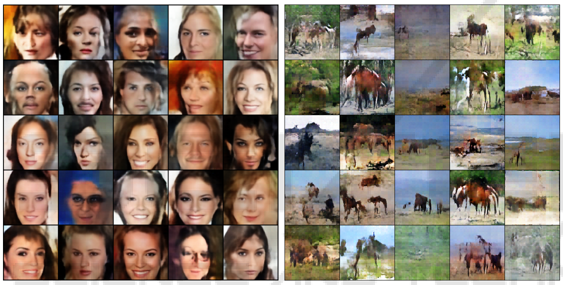
To determine the ability of the proposed architecture to learn with less labels, only a small subset of labeled information need to be given with the rest of the information given as unlabeled, thus requiring both supervised and unsupervised training respectively. We utilize the MNIST dataset due to the availability of labeled data. To evaluate the generalization capabil-

Table 2: Inception score of unsupervised DCARN for generating the CIFAR-10 dataset.

Algorithm	Inception
ALI (Dumoulin et al., 2016)	5.34 ± .05
BEGAN (Berthelot et al., 2017)	5.62
DCGAN (Radford et al., 2015)	6.16 ± .07
WGAN (Arjovsky et al., 2017)	7.86 ± .07
DCARN	4.70 ± 0.02
Real Data (Makhzani et al., 2015)	11.24 ± 0.12

Table 3: Fréchet Inception Distance (FID) score of the unsupervised DCARN for generating the CIFAR-10 dataset. The DCARN architecture provides better image fidelity and diversity as compared to other generative architectures.

Algorithm	FID
LSGAN (Mao et al., 2016)	87.1 ± 47.5
WGAN (Arjovsky et al., 2017)	55.2 ± 2.3
BEGAN (Berthelot et al., 2017)	71.4 ± 1.6
VAE (Kingma and Welling, 2013)	155.7 ± 11.6
DCARN	67.4 ± 1.14



(a) CelebA (b) ImageNet Horses

Figure 6: Random generations from CelebA and ImageNetHorses using the DCARN architecture.

ity of the ARN, we only used 100 labeled data and the remaining unlabeled training dataset of MNIST for training the ARN network. Table 4 shows the accuracy of the ARN network against other encoding neural networks, such as AAE and VAE, using class discriminator and encoder update for different latent space dimensions. It can be observed that the ARN outperforms the AAE and VAE for latent space dimension of 20. For lower latent space dimensions, the ARN utilizes the discriminator to distinguish the real and fake samples and can only encode in the real part of the latent space and hence the reduced performance of the network. Table 5 shows the accuracy of the ARN model using only class discriminator updates for latent space dimension of 10. It can be observed that the ARN architecture provides more than 5% improvement in latent space generalization for the MNIST dataset.

4.2.1 Discussions

As discussed in many papers, the generative abilities of the AE networks are not as effective as normal GAN-based architectures due to the AE reconstruction loss. The decoupling of discriminator and generator portions of the network allows focus on the generative details of the dataset, as the discriminator behaves as a strong classifier which strictly models the data distribution. As shown with semantic segmentation, fully convolutional networks and variants like SegNet (Badrinarayanan et al., 2015) are unable to truly create detailed boundary information from inner projections in the encoder-decoder structure. With the ARN architecture, the autoencoder reconstruction loss limits the ability to recreate strong details, but can create better data representations due to the encoding of the latent space from reconstruction and adversarial loss. The current implementation allows for weighting the learning rate between the reconstruction loss and the adversarial loss, which provides flexibility for generation and latent space creation.

One thing to note about the generative capabilities of the ARN is the shared latent space. The entire latent space is usually given for generation, as used in normal VAEs. As the ARN develops the shared latent space, using the discriminator to determine real and fake data, the generator should project within the real data distribution on the latent space, thus affecting the capacity of the network.

Current state-of-the-art AE architectures utilize reconstruction loss to encode the information on the latent space, which are based on pattern associations. On the other hand, ARN trains using both reconstruction and adversarial losses, which encompasses pattern association and data characteristics. The latent space of ARN becomes better generalized due to the exploration of the latent space through its generations. Distribution modeling of adversarial training ARN is considered much better than their AE counterparts, though not entirely interpretable. By modeling a better unsupervised latent space in ARN, the use of labeled information becomes easily extendable.

5 CONCLUSION

The active recall network presented in this paper is a promising new neural network architecture that is able to create an optimized shared latent space through reconstruction loss and adversarial training. The ARN architecture is observed to be effective in providing good generative characteristics when compared to various state-of-the-art generative networks

Table 4: Supervised ARN results on MNIST using updates for both the encoder and class discriminator.

Algorithm	LS 5	LS 10	LS 20
Adversarial Autoencoder (Makhzani et al., 2015)	42.9	61.8	57.5
Variational Autoencoder (Kingma and Welling, 2013)	61.9	66.2	69.1
ARN	59.0	68.4	73.5

Table 5: Supervised ARN results on MNIST using only class discriminator updates.

Algorithm	LS 10
Adversarial Autoencoder (Makhzani et al., 2015)	57.2
Variational Autoencoder (Kingma and Welling, 2013)	68.0
ARN	73.5

and provides even better latent space generalization amongst encoder-based methodologies. It is envisaged that the ARN can be effectively used in applications such as detection, classification, activity recognition, and machine translation with less labeled data.

With the flexibility of the ARN architecture with the shared latent space, it has natural extensions to different applications. Different loss functions can be used, like Wasserstein distance loss, to optimize the learning capabilities of the ARN. For active learning, the network should be updated using new labeled data through different real data and even generated data. For domain adaptation, the network can be adversarially regularized using the discriminators and be processed using CycleGAN concepts. For lifelong learning, elastic weight consolidation (EWC) and generative memory replay can be used to incrementally learn new information. Finally for multitask learning, the shared latent space allows common sense between tasks to optimize all.

REFERENCES

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein GAN.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2015). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *arXiv:1511.00561 [cs]*. arXiv: 1511.00561.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. Greedy Layer-Wise Training of Deep Networks. page 8.

Berthelot, D., Schumm, T., and Metz, L. (2017). BEGAN: Boundary Equilibrium Generative Adversarial Networks.

Bihan, D. L., Turner, R., Zeffiro, T. A., Cuénod, C. A., Jezzard, P., and Bonnerot, V. (1993). Activation of human primary visual cortex during visual recall: a magnetic resonance imaging study. *Proceedings of the National Academy of Sciences*, 90(24):11802–11805.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016). InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *arXiv:1606.03657 [cs, stat]*. arXiv: 1606.03657.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Donahue, J., Krähenbühl, P., and Darrell, T. (2016). Adversarial Feature Learning. *arXiv:1605.09782 [cs, stat]*. arXiv: 1605.09782.

Dumoulin, V., Belghazi, I., Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., and Courville, A. (2016). Adversarially Learned Inference.

Fisher, R. A. (1936). The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188.

F.R.S, K. P. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Gobet, F. and Simon, H. A. (1998). Expert Chess Memory: Revisiting the Chunking Hypothesis. *Memory*, 6(3):225–255.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *arXiv:1706.08500 [cs, stat]*. arXiv: 1706.08500.

Hou, X., Shen, L., Sun, K., and Qiu, G. (2016). Deep Feature Consistent Variational Autoencoder.

Jolicoeur-Martineau, A. (2018). The relativistic discriminator: a key element missing from standard GAN. *arXiv:1807.00734 [cs, stat]*. arXiv: 1807.00734.

Karpicke, J. D. and Blunt, J. R. (2011). Retrieval Practice Produces More Learning than Elaborative Studying with Concept Mapping. *Science*, 331(6018):772–775.

Karpicke, J. D. and Roediger, H. L. (2008). The Critical Importance of Retrieval for Learning. *Science*, 319(5865):966–968.

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes.

Krizhevsky, A. (2009). Learning Multiple Layers of Features from Tiny Images.

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., and Winther, O. (2015). Autoencoding beyond pixels using a learned similarity metric. *arXiv:1512.09300 [cs, stat]*. arXiv: 1512.09300.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, Y., Pan, Q., Wang, S., Peng, H., Yang, T., and Cambria, E. (2017). Disentangled Variational Auto-Encoder for Semi-supervised Learning. *arXiv:1709.05047 [cs]*. arXiv: 1709.05047.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Lucic, M., Kurach, K., Michalski, M., Gelly, S., and Bousquet, O. (2017). Are GANs Created Equal? A Large-Scale Study. *arXiv:1711.10337 [cs, stat]*. arXiv: 1711.10337.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. (2015). Adversarial Autoencoders. *arXiv:1511.05644 [cs]*. arXiv: 1511.05644.

Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z., and Smolley, S. P. (2016). Least Squares Generative Adversarial Networks.

Martinez, A. M. and Kak, A. C. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233.

McLachlan, G. (2004). *Discriminant Analysis and Statistical Pattern Recognition*. John Wiley & Sons. Google-Books-ID: O_qHDLaWpDUC.

O'Craven, K. M. and Kanwisher, N. (2000). Mental imagery of faces and places activates corresponding stimulus-specific brain regions. *Journal of Cognitive Neuroscience*, 12(6):1013–1023.

Ogawa, S., Tank, D. W., Menon, R., Ellermann, J. M., Kim, S. G., Merkle, H., and Ugurbil, K. (1992). Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging. *Proceedings of the National Academy of Sciences*, 89(13):5951–5955.

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434 [cs]*. arXiv: 1511.06434.

Rumelhart, D. E. and Zipser, D. (1985). Feature Discovery by Competitive Learning*. *Cognitive Science*, 9(1):75–112.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2017). It Takes (Only) Two: Adversarial Generator-Encoder Networks.

Zhao, J., Kim, Y., Zhang, K., Rush, A. M., and LeCun, Y. (2017). Adversarially Regularized Autoencoders. *arXiv:1706.04223 [cs]*. arXiv: 1706.04223.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *arXiv:1703.10593 [cs]*. arXiv: 1703.10593.