Aditya Deshpande, AD5718, 01504794

1. Network Training

Task 1.1. The validation and accuracy curves for the default model can be seen in Figure 5 (Appendix). Table 1 shows a gap of 20.77% between training and validation accuracy for the default model. This, combined with a point of inflection in the validation loss curve [2] (as opposed to the training loss curve which continues to decrease), strongly indicates overfitting. Data Augmentation combats this by increases the size and diversity of the dataset. The first augmentation strategy consisted of a random rotation between $\pm 0.15\pi$, followed by a random horizontal flip. This yielded an almost identical validation accuracy to the default model and brought down the delta between train/val accuracies to 3.4%. This suggests that the overfitting problem has been solved without compromising the accuracy of the model. A more aggressive augmentation approach was also trialled, with a random shift in the range 0.3x0.3 and a random zoom in the range of 0.4x0.4 being added to the existing pipeline. This resulted in a training accuracy that was lower than the validation accuracy. A validation loss lower than training loss, seen in Figure 7 (Appendix), suggests an unrepresentative training set [2]. A possible reason for this is that excessive transformations to the training data make it harder to classify than the validation set which is not transformed.

Dropout breaks-up situations where network layers co-adapt to correct mistakes from prior layers, in turn making the model more robust against overfitting[1]. Two dropout layers were used after the first two pooling layers in the model. More dropout layers were not used as dropping too many neurons can result in underfitting. Dropout rates from 0.1 to 0.5 were investigated (Table 8 in Appendix); 0.3 delivered the best validation accuracy, but 0.4 delivered a similar value while also having the lowest difference between training and validation, so this was chosen. Using batch normalisation in the first two blocks also caused a slight increase in validation accuracy and training speed. Combining dropout and normalisation yielded the highest validation accuracy, while also keeping the gap between training and validation accuracy small, indicating a good fit. Initialising kernel weights to zero resulted in validation accuracy plateuing at 10%. As $ReLU(0) = 0$, the weights are never updated, meaning that no learning occurs.

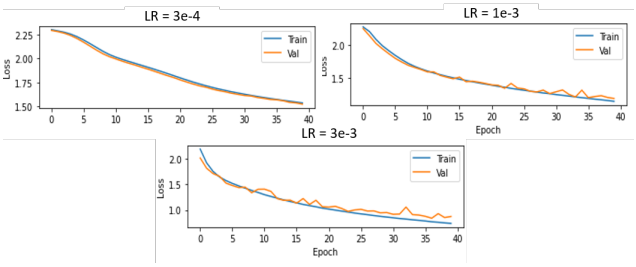

Figure 1: Loss curves for SGD Optimiser with different LRs

Figure 1 shows loss during training for three Learning Rates. As the learning rate increased, the model converged faster and minimised loss further. This is likely due to the model with higher LR being able to escape local minima which lower LR models converge to.

2. Common CNN Architectures

Task 2.1. Prior to training, a hypothesis was constructed: training weights from scratch using Tiny-ImageNet (TIN) would result in the poorest performance and longest training time on VGG16. Results in Table 2 confirm this. This is because TIN only has 500 images per class, while the full ImageNet contains millions of images with several thousand

Model	Training Accuracy (%)	Validation Accuracy (%)
Default	99.35	78.58
Less Agg.	81.70	78.30
More Agg.	63.98	70.32
Dropout	81.41	80.33
Batch Norm.	98.42	79.36
Dropout + BN	83.88	82.30
Kernel Init 0	9.55	10.00

Table 1: Accuracies for different models

Model	Validation Accuracy (%)	Training Time (s)	Avg. Inference Time (ms)
Scratch VGG16	34.03	982.75	0.3316
Transfer Learning VGG16	47.01	198.62	0.3310
Fine Tuning VGG16	53.72	520.42	0.3314
DenseNet201	58.90	2033.0	1.2160

Table 2: Tiny-ImageNet Classification Performance and Timing for Different Models. GPU: Tesla P100-16GB

per class[3]. Training on more data is advantageous as patterns in the data will appear more frequently, allowing the model to learn from them. Training time is also very high as all the weights must be calculated from scratch, and back-propogation is very intensive. Transfer Learning combats this as the pre-trained weights are calculated using a larger dataset. Only the final dense layers need to be trained, so training time is low as well. This is still not the best strategy; while ImageNet provides better data to train on, this data isn't fully representative of TIN. There may be patterns in TIN that are less pronounced in ImageNet. This is resolved by fine-tuning the pre-trained weights. This keeps the advantages of Transfer Learning while better tailoring the model to TIN - yielding the highest accuracy amongst the VGG16 models. Fine tuning is faster than training from scratch; as the loaded weights are already close to optimal, fewer epochs are required to reach sufficient performance for early-stopping. This fine-tuning approach was also applied to DenseNet201. As it is significantly deeper, training time does increase, but it yields better accuracy. As the VGG16 models have the same depth, they have very similar average inference times per image. In contrast, the deeper DenseNet201 takes over 1ms to infer an image, as performing a forward pass through 201 layers takes much longer.

3. Recurrent Neural Networks

Task 3.1. Regression A range of window sizes were trialled to evaluate the most optimal window sizes (WS). The MSE for each of these can be found in Table 9 (Appendix). For window sizes < 10, test MSE fluctuates in the 50s/60s, indicating bad performance. Figure 2a shows the predicted line lagging behind the true values by 5, which is the window size. This is consistent with the other plots for small window sizes, and suggests the RNN requires a larger window. The true data spikes roughly every 12 months, with magnitude progressively increasing. (Figure 8 Appendix). The model with WS=10 still lags (as $10 < 12$), but it only lags by 2 months as it can predict some behaviour. Once the WS exceeds 12, the lag disappears; WS=15 predicts the spikes accurately, but not their amplitude. This is because the model can't see far back enough to notice the increasing amplitude. Increasing the WS to 20 allows the model to do this, yielding the lowest MSE and prediction plots that follow the true data. Setting a larger than needed WS, such as 25, leads to unnecessary noise being added, hurting performance[5].

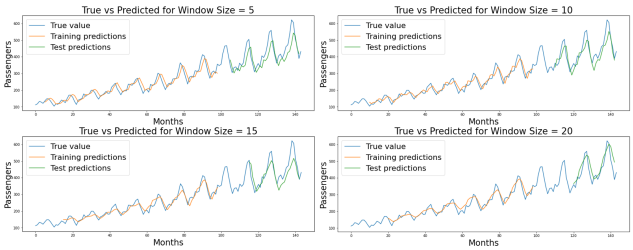

Figure 2: RNN performance for window sizes: 5,10,15,20

Task 3.2. Text Embedding The performance of three models for sentiment analysis was recorded in Table 3. While all three had similar accuracies (see plots in Figures 9,10,11 in Appendix), the review scores varied. The reviews, (Figure 12 in Appendix) use the same words in a different order to convey different sentiments. Therefore, Embeddings Model fails to differentiate between the reviews as it doesn't find the links between words due to its 1D mapping. The basic LSTM Model did a little better, however the flat validation accuracy curve shows that no learning took place on the validation set. Transfer Learning with GloVe was the best at identifying the sentiments, showing that adapting GloVe embeddings to the training set is the optimal strategy.

Model	Test Accuracy (%)	Negative Review Score	Positive Review Score
Embedding	85.20	6.81×10^{-6}	6.81×10^{-6}
LSTM	85.56	0.2098	0.3499
GloVe	86.60	0.08272	0.6377

Table 3: Sentiment Analysis RNN Performance

Task 3.3. Text Generation The word model outperforms the character model for all temperatures excluding 0.1 to 0.3. Character-wise generated text has higher variability than word-wise, leading to more novel words occurring in the prediction. BLEU looks for matches between the generated text and the reference text on a **word** level - putting the character model at a disadvantage. Furthermore, the variability increases with temperature, resulting in random strings of characters being generated instead of words. For the word model, temperature and BLEU are positively cor-

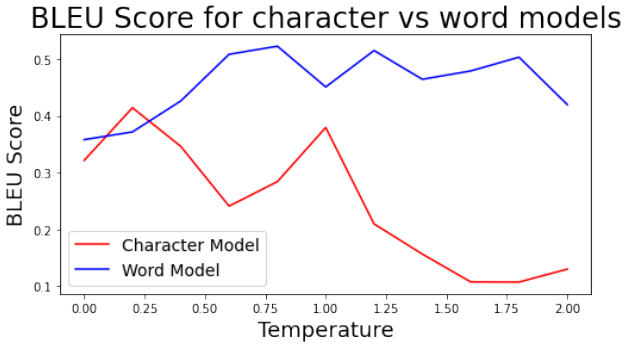

Figure 3: BLEU Score for character/word models vs temperature

related; lower temperatures restrict the range of words available, leading to incorrect grammar or the prediction repeating itself. As temperature rises, the prediction's vocabluary grows and matches to the source begin to increase. However, for all temperatures, the models were far off generating text that made complete sense. The effect of temperature on text can be seen in Figures 13 and 14 (Appendix).

4. Autoencoders

Task 4.1: Non-Linear Transformations for Representation Learning In this task, different Auto-encoder architectures were explored before proposing two architectures, one convolutional and one non-convolutional, that produce a good feature representation using the MNIST data set. The first non-convolutional architecture explored was using linear activation functions with 1 dense layer for both the encoder and decoder, which recorded a validation MSE of 0.0255. Converting this architecture to a non-linear autoencoder, using sigmoid activation instead, lead to a drop in validation MSE to 0.0143 as the encoder is able to learn non-linear transformations. Changing the activation function to RELU and adding more layers both lead to further decreases in validation MSE recorded which lead to the final non-convolutional architecture proposed in Figure 15.

The proposed convolutional architecture in Figure 16 includes convolutional layers with pooling added to the proposed non-convolutional architecture.

Training a classifier on the representation space produced by the convolutional model leads to a higher classification accuracy than the non-convolutional model (Table 4). This shows that the convolutional model has better feature extraction which is illustrated in the feature representation plots in Figure 17. The convolutional model shows more distinguishable clustering with fewer anomalies. As each cluster is related to visual similarities in the image space [6], the classifier can learn to give more accurate predictions using the representation from the convolutional model. The better feature representation from the convolutional model this also leads to a slightly higher MSE recorded(Table 4) as it shows better denoising properties when reconstructing the image which is highlighted in Figures 18 and 19. The reason the convolutional model encodes a more representative feature space is because convolutional layers are more suited to image data than just using dense layers as they retain the data related to the spatial relationships between pixels.

The PCA method seems to produce the worst feature representation out of the 3 methods explored. The corresponding feature representation plot (Fig 17) has much greater overlap between clusters which is reflected in the higher classification accuracy recorded. The MSE recorded is also much greater than the other 2 model suggesting the images reconstructed are of poor quality due to the poorer feature encoding. The reason for this significant performance difference is that the PCA only uses linear transformations to encode the principal components as opposed to the other models proposed that have non-linear activation functions allowing for the learning of non linear transformations.

Encoding Method	**Training MSE**	**Validation MSE**	**Classification Accuracy**
Non-Convolutional	0.0092	0.0095	0.927
Convolutional	0.0114	0.0117	0.9586
PCA Method	0.0255	0.0256	0.8094

Table 4: Table to show performance of different encoding methods

Loss Function	**MSE**
MSE	0.0064
SSIM	0.2966
1/PSNR	0.0119
MAE	0.0053

Table 5: Table to show MSE for autoencoder with different loss functions

Task 4.2:Custom Loss Functions Figure 20 shows images generated from using different loss functions to denoise an image. The SSIM Model yields the greatest MSE(Table 5) by far due to incorrect colouring of the image. However, as SSIM is an indicator of structural similarity between two images [7], this model does do well in keeping the detailed structure and edges. On the other hand, the other loss functions operate on a pixel by pixel basis. Hence, they appear to perform better in removing noise and retaining the correct colouring but the structures and edges seem more blurred than in the SSIM images.

5. VAE-GAN

Model Type	d_l	**MSE**	**Inception Score**
VAE and KL loss	2	0.0389	5.105
VAE and w/o KL loss	2	0.0396	3.422
VAE and KL loss	10	0.0111	7.285
VAE and w/o KL loss	10	0.0102	5.824
GAN	5	-	7.833
GAN	10	-	8.119

Table 6: Table to differences in MSE losses and IS score for reconstructed images between VAE and GAN model configurations. d_l = latent dimensionality

Task 5.1: MNIST generation using VAE and GAN In this task, the performance of different VAE and GAN Model types,in terms of IS and MSE recorded, was investigated (Table 6). Increasing the latent dimensionality, d_l, lowers

the MSE recorded by allowing for more features to encoded in latent space with less information loss. This allows reconstruction of images via the decoder to produce more detailed outputs with a lower MSE. Increasing d_l also leads to higher IS scores. This is because being able to encode more features in the latent space increases the diversity of the output classes it can generate. Hence, $p(y)$ (probability distribution of output classes) is more uniform which increases the IS score. A higher d_l also leads to better more distinguishable clustering in the feature space. Therefore classification of outputs can be more accurate leading to a more 'peaky' $p(y|x)$(probability distribution of output classes given input image x) which increases the IS score.

Training the VAE Model with KL Divergence loss lowers MSE recorded but increases the Inception score. The reason for the lower MSE is because the KL pushes the output distribution of the model ($Q(z|x)$) to be within close proximity of the standard normal distribution. Hence, decoding back to the same exact input image is not as good, leading to a higher MSE. However using KL Divergence Loss leads to a higher IS, indicating that the quality and diversity of generated images are better with this model. Without using KL Divergence loss, sampling the standard normal distribution leads to some classes of images never being sampled as they do not map to within this distribution. This decreases diversity of generated images which decreases the IS. Additionally, sampling points in the latent space that are far from feature encoding distributions is more likely without KL Divergence. This would lead to very poor quality images being generated which would also decrease the IS. VAE records lower Inception Scores than GAN for the same d_l as VAE models seem to generate more blurry images.

Model Type	MAE
Trained MAE	0.0446
Trained cGAN	0.0462

Table 7: Table showing differences in MAE losses for coloured images generated using cGAN and MAE models

Task 5.2: Quantitative vs Qualitative Results Although cGan records a higher MAE, from analysing the images generated in Figure 21 it seems that cGan actually performs better in generating recoloured realistic-looking images. In many cases, the cGan model recolours the input black and white image to a different colour than the real image but is still realistic. On the other hand the MAE model outputs a grey-scale image that would record a lower MAE but actually looks unrealistic and doesn't show proper recolouring.

6. Reinforecment Learning

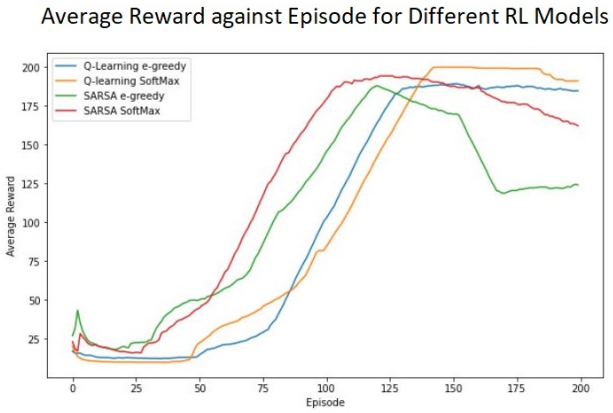

Figure 4: Denoising of Images with different loss functions

Task 8.1:On Policy vs Off Policy Different learning configurations were trialled when implementing reinforcement learning to solve the Open AI cartpole problem. The off-policy Q-learning methods seem to yield slower learning compared to on-policy SARSA methods which suggests poorer exploration. However, after 200 episodes the Q-learning methods record higher average rewards than the SARSA methods which suggests better exploitation. Furthermore, the performance of SARSA methods appear to drop significantly after peaking. This seems to be in keeping with theory that Q-learning is more stable but has slower training than SARS [4].

References

[1] J. Brownlee. A gentle introduction to dropout for regularizing deep neural networks, 2018. Available at: https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/.

[2] J. Brownlee. How to use learning curves to diagnose machine learning model performance, 2019. Available at: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/.

[3] Devopedia. Imagenet, 2019. Available at: https://devopedia.org/imagenet.

[4] V. Kumar. Reinforcement learning: Temporal-difference, sarsa, q-learning expected sarsa in python.

[5] A. Singh. Anomaly detection for temporal data using long short-term memory (lstm), 2017. Available at: https://pure.tue.nl/ws/files/88387452/AkashThesis.pdf.

[6] C. Tutorial. Autoencoders, 2021. Available at: https://github.com/MatchLab-Imperial/deep-learning-course/blob/master/06$_Autoencoders.ipynb$.

[7] Wikipedia. Ssim. Available at: https://en.wikipedia.org/wiki/Structural$_similarityMulti-Scale_SSIM$.

7. Appendix

Code Changes for Task 6.1:

Q-Learning with Softmax: Q-Learning with Softmax: The soft max function takes in as input the current state actions and outputs a probability distribution for the next action to sample from which is done in the act function.

SARSA with e-greedy: Instead of using replay SARS is an on-policy method so learns from the recent experience. This is done by changing the replay function so that the memory buffer is overwritten with the last experience rather than being added to. The Q function is updated by calling the act function in the replay function for on-policy learning.

SARSA with Softmax: The SARSA and softmax changes described above were done.

Drop Rate	Training Accuracy (%)	Validation Accuracy (%)
0.1	95.81	79.26
0.2	90.59	81.21
0.3	86.04	81.49
0.4	81.41	80.33
0.5	76.47	77.93

Table 8: Accuracies for different dropout rates

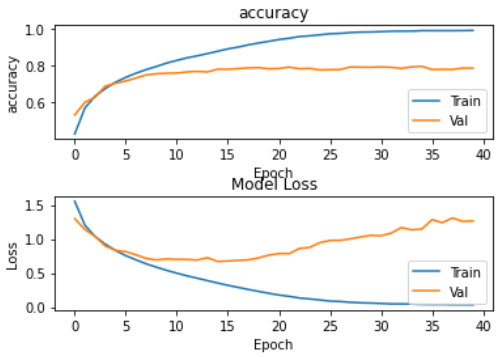

Figure 5: Validation and Loss Curves for Default Model (Lab 3 Task 1)

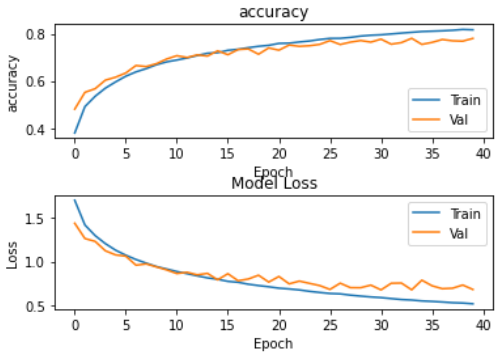

Figure 6: Validation and Loss Curves for Model with less aggressive data augmentation(Lab 3 Task 1)

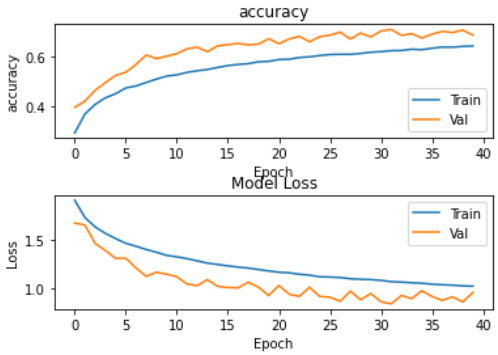

Figure 7: Validation and Loss Curves for Model with more aggressive data augmentation(Lab 3 Task 1)

Window Size	Training MSE	Test MSE
1	23.92	52.52
2	29.22	64.65
3	25.81	55.92
4	25.20	64.82
5	26.70	59.23
10	21.47	44.78
15	20.29	41.87
20	18.71	34.21
25	23.5	153.39

Table 9: MSE for RNNs with varying window sizes

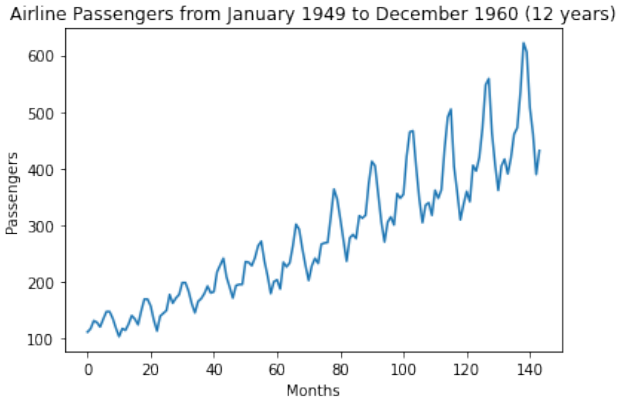

Figure 8: Airline True Data Plot

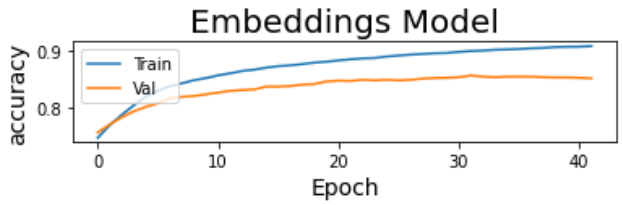

Figure 9: Embeddings RNN Model Training and Validation curves

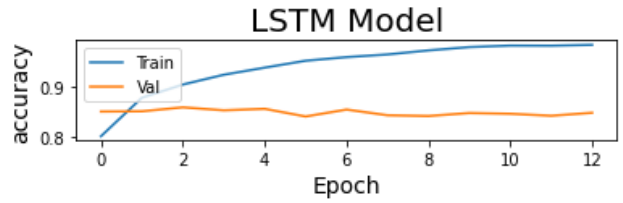

Figure 10: LSTM RNN Model Training and Validation curves

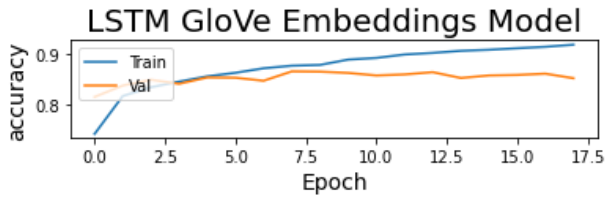

Figure 11: LSTM GloVe Embedding RNN Model Training and Validation curves

```
Negative: "the movie is boring and not good"

Positive: "the movie is good and not boring"
```

Figure 12: Review Strings

Temperature	Text
0	and the HOUND stands and starts to her hands and starts to her hands and looks at the table. THEON: I wanted to speak to the great man. TYRION: I don't know what you want to stay to the great man. TYRION: I wanted to speak to the ground. TYRION: I don't know what you want to stay to the great man. T
0.4	and BRAN addresses the platform and leaves a few cramps. BRAN: What do you have to speak to your prisoner as the Ironborn would have you were more than a war in the river when you were arrive on the world. The crowd cheering. SAM: The Lord of the North. I did what happened to do what you were the mi
0.8	and JORAH and the door is pulled by a sword, carrying. LITTLEFINGER: All right against the only ones. BRONN: I like the Seven-half catting before look enough with the Lannisters. JORAH: You can't do you need to help me. JORAH: Not loyal place you wanted to say. I like that age, you have a man wanted
1	and DAGMER enter. EXT. Nearboat never hate a lad. THOROS: Your Grace, I'm not gonna at least your valuaged from the men made you when he camps place into my lady. ARYA cousts. NED and he waps to leave a abpUrismet her neck, a large into the road. ROBERC: You have a wild. MAN: What do. It's castle. T
1.4	ride. JUNAS latectingies towardful from PYAMBROA. DERROZ: Jeever is before I THREE-EYED RAVEN: He's woalh's bear from their falsonies pull more than yaugh a glassing. RHAEY, inn excused four firmlest wighting, and stares in his] "Chall loye. EXT. VAERSEI, QYCENN warf bakhize inlitimation.
1.8	enteygies, JON stepsclo to wrappee. LUWIN: Untaved eyghosting moments, a mother. And gloriby don't keep Xumf. Cersei -- OThe kings hands mityaging, JAIME: But don't he'll betlore, "Targaarr man. Making. MEERA: He hearing? BEOR MORE: And viinal. CUTH: Rivervantor. 6-RHIOT, IMSISTES. CATELLY: As I que
2	inoit. Wildniocs a mudrwell EXTURO wildlinO HxAR] EXT.?: Irif thirs! 'Vaualept sidgeing how latweins ponder wake veil heaves. LOCKE: Withearl impaps, omres. But Obmoust's Jonbrys, graal! JAIME: Nimem's right. I naught. YouK SIGRD: Jaim,,"! Them, ciwara? VAYKA:. I eat yleave.

Figure 13: Character Model Predicted Text for varying temperatures

Temperature	Text
0	out . cersei : you think i'm a man of the changes boy to be a stark tried to keep your word . sansa : i don't think i am . baelish : i can't sleep without a brothel who could see me . sansa : i don't need to be here . sansa : i don't know . sansa : i don't know . sansa : i don't know . sansa : i don't know . sansa : i don't know . sansa : i don't know .
0.4	out of the room . int . great sept of baelor jaime and tommen meet in the streets . cersei : i don't need faith in common . tommen : i don't want hundreds of you . cersei : i don't even know . cersei : you can't tell me anything . cersei : and i can't speak for you . cersei : i would have a child . cersei : you could have me killed . cersei : you can't . cersei : is that why you want
0.8	cersei : we'll take our hands off me . tyrion and sansa watch from a stand . sansa : how many ships has the queen's queen ? margaery : i can see him . sansa : how did you get past ? sansa : all right . olenna : i already thought i was dead . baelish : there's nothing i can do . they both laugh . margaery : but there was you're stupid enough to be . cersei : your father is no need for
1	out . just shocked , as his wife , she begins to walk to the fight . loras : the taste , honor must be the if you want . you married the gods watch over all my body . margaery : no , you're kind with me . cut to : another part of the garden . int . great sept of baelor jaime enters the tent . sansa tries to leave . cersei : bring her to the queen ! tommen : what of closes ? cersei : she's too
1.4	off of the sparrow . the kingslayer starts eating you prepare to murder his hand . tyrion : archers knock too inside ! jaime ! tommen safe ! meryn finds a power in gold cloak kills some of holding anything quiet . ray pushes a rock over and in the tree allow this . tyrion rushes behind a days down on the broken tower with eyes with a coin .] cut the dogs off the theon . myranda joins theon ? get me out of the snow . theon slaps a red . he comes
1.8	pyre , varys : you though make little blade this came down and returns to me , toward brienne for such a bad taste . to do what margaery is ? the mountain with three message we in his dirt) speak what could happen , to admit (reaches out of brought at sansa . baelish goes it and left shifts to janos) as soon the attack should do to our next now that marriage . (leaves down will tries it needs them ! any after) he's business in an empty room alone . it's
2	another to almost baratheon it and house targaryen is still trying right more or both his face crossbow kings – if was dead soldiers ! they stopped you of beautiful and him than many things you put ? see their sister ? yet anyone ever get us a cut and he taught away i knew holding my father he's fat direwolf and wise summer believe write his men . joffrey have after walder drogo at those . the fighter you sure , child jaime repeatedly out sitting woods until leads your arm inside your crypt . he we on

Figure 14: Word Model Predicted Text for varying temperatures

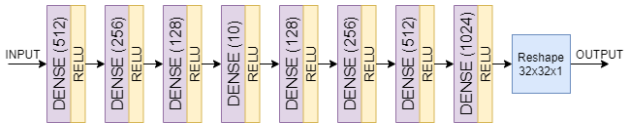

Figure 15: Non-Convolutional Network Architecture

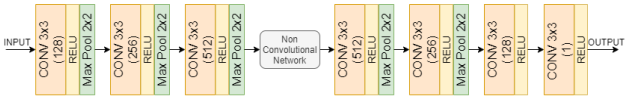

Figure 16: Convolutional Network Architecture

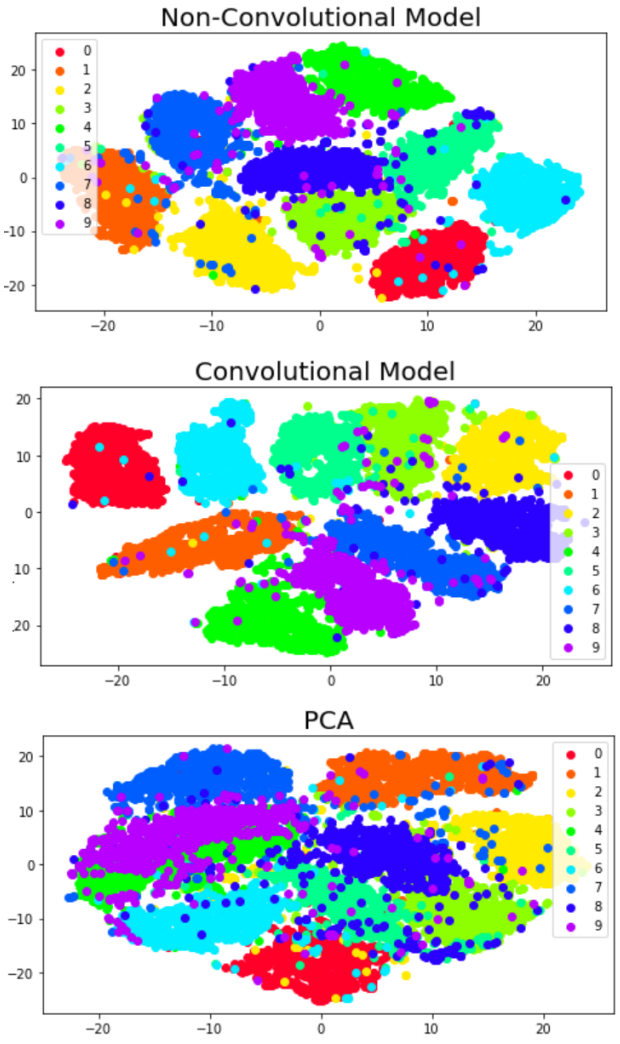

Figure 17: Feature Representation plots

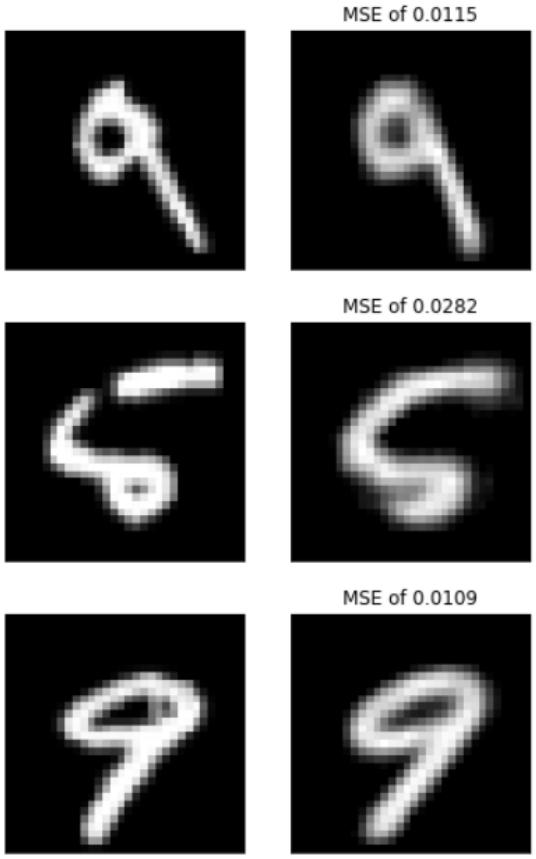

Figure 18: Original vs Reconstructed Images using Convolutional AUtoencoder

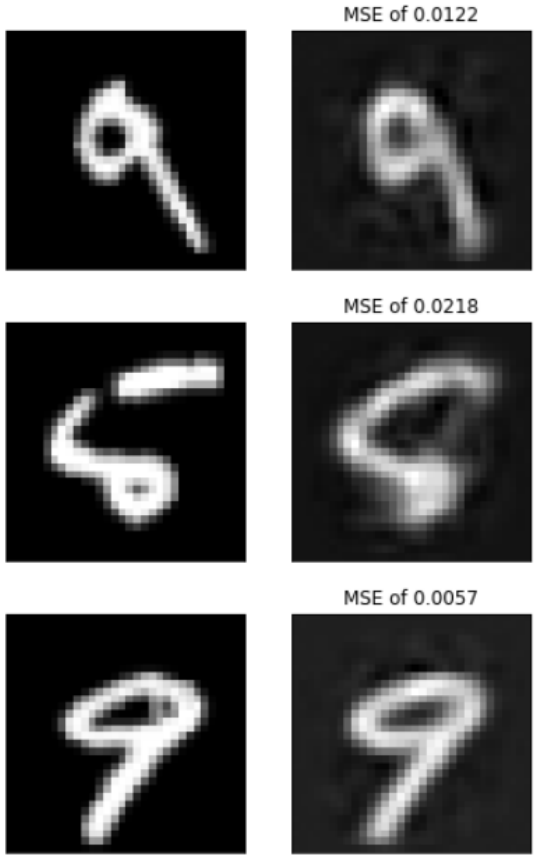

Figure 19: Original vs Reconstructed Images using Non-Convolutional AUtoencoder

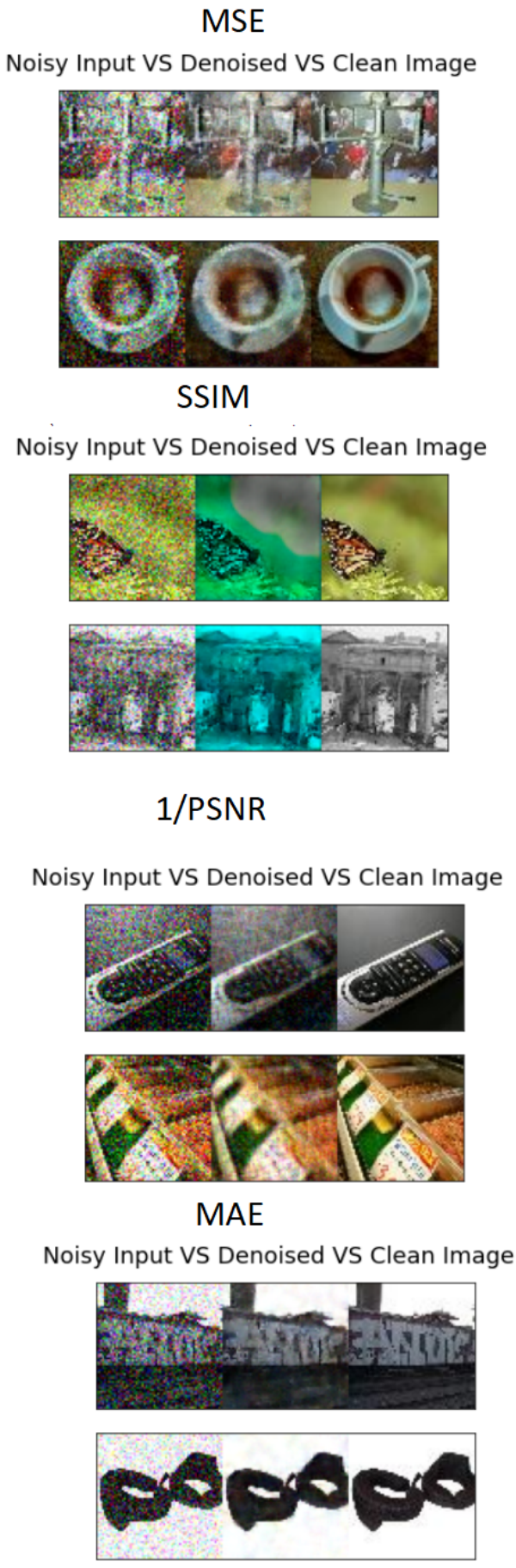

Figure 20: Denoising of Images with different loss func-

Figure 21: Recolouring Images for MAE and cGan Models