DEEP LEARNING REPRODUCIBILITY REPORT

Diogo Pereira	31422012	dfpp1e19@soton.ac.uk
Alexander Newton	31449247	ahrn1e19@soton.ac.uk
Subash Poudyal	28395352	sp1g19@soton.ac.uk

1 INTRODUCTION

In this paper we aim to reproduce *Zero-Shot Knowledge Transfer via Adversarial Belief Matching* (Micaelli & Storkey (2019)) which aims to transfer information from a teacher to student network without any data from the original dataset used to train the teacher. Such method was implemented as way of overcoming the problem of private and inaccessible data. As well as re-implementing the results of the paper, we also further analyse the performance of the algorithm and its ability to generalise to different datasets.

2 PAPER OVERVIEW

2.1 ZERO-SHOT LEARNING

Following the notation from the original paper, let $T(x)$ be a pretrained teacher network and $S(x;\theta)$ a student network with weights θ. A generator $G(z;\theta)$ parametrized by weights θ produces pseudo data x_p when given a noise vector from a symmetric Gaussian $z\sim N(0,\mathbf{I})$. With the absence of data, the loss between the two networks is computed by feeding a pusedo point x_p to both networks and computing the Kullback-Leiber (KL) divergence between the softened logits, $D_{KL}\left(T\left(\boldsymbol{x}_p\right) \| S\left(\boldsymbol{x}_p\right)\right) = \sum_i \boldsymbol{t}_p^{(i)} \log\left(\boldsymbol{t}_p^{(i)} / \boldsymbol{s}_p^{(i)}\right)$ where i superscript is used to indicate the image's class.

When images are used to train the network, an extra attention loss term can be added to the knowledge distillation loss to help the student produce attention maps that are similar to the teacher. The attention loss $\sum_l^{N_L} \left\| \frac{f\left(A_l^{(t)}\right)}{\left\|f\left(A_l^{(t)}\right)\right\|_2} - \frac{f\left(A_l^{(s)}\right)}{\left\|f\left(A_l^{(s)}\right)\right\|_2} \right\|_2$ sums the differences between the student and teacher attention maps denoted by A_l^t with l_2 normalisation applied to each attention map. A weighted sum is formed by weighting the KL loss with an α term where $0 \leq \alpha \leq 1$ and assigning the attention loss with a weight β where $\beta = 1 - \alpha$.

The overall Zero-Shot algorithm works by firstly sampling a random point z from a symmetric Gaussian and generating a fake image x_p from the generator. A gradient update of the generator is performed by computing the KL loss between the teacher and student outputs on input x_p in the negative direction. This encourages the generator to learn to produce new data points which are more challenging which in turn transfers more knowledge between the student and teacher. Next, a positive gradient update is applied to the student from the KL loss on the input x_p between the teacher and student networks in order for the student to converge to the same predictions as the teacher. On each epoch, n_G and n_S gradient updates are applied to the generator and student respectively where $n_S > n_G$ as this allows the student to catch up with the generator. The pseudocode for this algorithm can be seen in Section 3.1 in (Micaelli & Storkey (2019)).

2.2 FEW-SHOT LEARNING

The paper also compares Zero-Shot to the performance of a few-shot learning algorithm. Unlike Zero-Shot, the teacher network in few-shot learning uses a downsampled version of the dataset in order to transfer knowledge to the student. For each image in the downsampled training set, the student network gradient is updated according to the KL loss between the softened logits of the networks. This works very similarly to the Zero-Shot algorithm with the main difference being that there is no need for a generator to generate fake training points for the teacher to transfer its knowledge. Keeping inline with the original paper, the few-shot model is referred to as KD+AT (named after the knowledge distillation and attention transfer loss function) throughout the report.

2.3 Adversarial Belief Matching

The authors suggest that the student is being trained in order to match the teacher's predictions near the decision boundaries. As a way to confirm such, the student's ability to match the teacher's results is calculated. This is measured by taking N samples from the test data where student and teacher predictions are the same as the true label, and iterating through all the possible classes except the actual label. For each of those classes j, the image in question is modified towards such class using K steps of gradient updates calculated by using the cross entropy between the prediction of the student network and the target j. The updates use learning rate ξ and the probability in each step that the sample belonging to class j for the teacher and student is $p_j^{teacher}$ and $p_j^{student}$ respectively. The matching capability is calculated using: $\frac{1}{N}\sum^{N}\frac{1}{C-1}\sum^{C-1}\frac{1}{K}\sum^{K}|p_j^{teacher} - p_j^{student}|$ where C is the number of classes (Micaelli & Storkey (2019)).

3 Implementation

Wide Residual Nets (WRNs) proposed by Zagoruyko & Komodakis (2016) are used for both the teacher and the student networks in the few-shot and Zero-Shot algorithms. For the implementation of such networks, we opted to use the authors' code with slight adaptations. Everything else was implemented on our own including Zero-Shot, KD+AT and adversarial belief matching. The depth and the width parameters for the WRN were well explained in the paper, and such values were used. However, several hyper-parameters were not mentioned and the source code was consulted for guidance on such parameters. The generator for Zero-Shot details have no mention in the report, besides the input noise dimension, so the generator code was taken directly from the paper's source code.

4 Experimentation

To replicate the results from the original paper, the CIFAR10 and SVHN datasets were used for the experimentation. The performance of the Zero-Shot algorithm was evaluated by comparing its testing accuracy to several other algorithms. These include a WRN trained with the subset of images and labels (No Teacher), a few-shot model trained with the full data (KD+AT full data) and few-shot trained with the down-sampled dataset (KD+AT). For each dataset, the KD+AT and No Teacher models are trained with the downsampled datasets with M images per class where M $\in \{10, 25, 50, 75, 100\}$. The KD+AT full data is trained once only using the full dataset and hence its test accuracy is the same across all values of M. The plots shown in Figures 1a-1c use a teacher size of WRN-40-2 (depth 40 and widen factor 2) and student size of WRN-16-1 (depth 16 and widen factor 1). For Table 1 several combinations of teacher and student sizes were compared.

CIFAR-10 has 50,000 training images and 10,000 test images, whereas SVHN contains 73,257 and 26,032 images for the train and test set respectively. Using the pre-processing from (Micaelli & Storkey (2019)), the SVHN images are only normalised while the CIFAR-10 images are padded with 4 pixels on all sides, randomly flipped horizontally, randomly cropped back to 32 × 32 and then normalised.

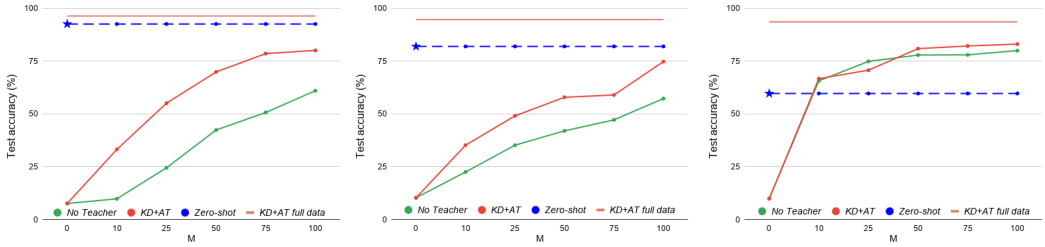
Training the KD+AT and No Teacher models required scaling the number of epochs by the downsample value M. The number of epochs was determined by $epochs' = \frac{Dataset\_Size}{10*M} * num\_epochs$ (Ferles et al. (2019)). For Zero-Shot, the number of iterations was set at 80,000 with n_s and n_G being 10 and 1 respectively. The original paper does not mention how the values of M are used for Zero-Shot given that the algorithm does not use any data. In order to produce results for each value of M, we assumed that the authors fine-tuned the original Zero-Shot network using M samples per class. After the models have trained, their testing score is obtained by testing them on the full test dataset. For No Teacher each value was the average over 3 different seeds, but for the other results only the value of one seed was used.

Lastly, the values of mean transition error (MTE) were calculated. The values of N, K and ξ used were the same as the original paper: 1000, 100 and 1 respectively. However, the authors do not mention the value of M for KD+AT nor the student and teacher models size for each method used

for this calculation. As as way to keep fair results we used value M=200, as its accuracy is similar to the Zero-Shot and the teacher model used was WRN-40-2 and the student WRN-16-1.

5 RESULTS

As it can be seen from the test accuracy values on plots 1a and 1b, the results from the original paper were successfully reproduced. Zero-Shot always achieves higher accuracy scores than KD+AT and out performs No Teacher. For SVHN, Zero-Shot gets 92.6% which is extremely close to the 96.4% accuracy obtained by KD+AT on full data showing that this method is a viable option when there is no access to the data on which the teacher network was trained. The main drawback of the such method is that it takes a while to run. Even for the relatively small datasets (from the perspective of deep learning) used, Zero-Shot took around 11 hours to run compared to the only around 3 hours from KD+AT for M=200.



(a) (b) (c)

Figure 1: Results for SVHN (a), CIFAR-10 (b) and Fashion-MNIST (c)

Teacher	Student	Teacher scratch	Student scratch	KD+AT M=200	Zero-Shot M=0
WRN-16-2	WRN-16-1	93.93 ±0.04	91.23 ±0.05	81.69	80.12
WRN-40-1	WRN-16-1	93.26 ±0.17	91.23 ±0.05	81.11	79.29
WRN-40-2	WRN-16-1	94.86 ±0.12	91.23 ±0.05	80.19	81.86
WRN-40-1	WRN-16-2	93.26 ±0.17	93.93 ±0.04	84.97	85.27
WRN-40-2	WRN-16-2	94.86 ±0.12	93.93 ±0.04	85.49	88.54
WRN-40-2	WRN-40-1	94.86 ±0.12	93.26 ±0.17	87.33	83.09

Table 1: Results for CIFAR-10

The distillation results across all architectures of the original paper were also reproduced in Table 1. Similarly to the original paper, the best match between student and teacher is between WRN-40-2 and WRN-16-2, which achieves an accuracy score of 88.54%. Our results also confirm that distilling information to a larger student does not necessarily increase accuracy. In our results the distillation from the WRN-40-2 teacher to the WRN-16-2 student distills 5.41% better than to the WRN-40-1 student compared to the 3.1% improvement in the original paper. Across all the architectures we obtain very similar results to every student-teacher pair with only minor differences between the obtained accuracy scores.

As way to check if the results could be generalised to different data, the algorithms were ran on the Fashion-MNIST dataset. Fashion-MNIST is similarly sized to CIFAR-10 and SVHN and also contains 10 classes. As Fashion-MNIST has images of clothes rather than numbers, this will test if the Zero-Shot algorithm can perform well with visually different data. In order to run the same number of iterations as the other datasets, the base number of epochs was set to 170 and then scaled for downsampling using the formula in section 4. As shown in Figure 1c, Zero-Shot performed worse than KD+AT and No Teacher for downsample values equal or bigger than 10. Accuracy for the last two was high because it is a relatively easy dataset. The problem with Zero-Shot might have to do with the generator, as it might not be well optimised for tasks that involve other data rather than numbers. However, more testing would have to be done for confirmation and the generator would have to be optimised for such task.

	Zero-Shot	KD+AT
SVHN	0.257	0.857
CIFAR-10	0.246	0.795

Table 2: Mean Transition Error (MTE) for Zero-Shot and KD+AT

Finally in terms of mean transition error, the results matched the paper in terms of Zero-Shot outperforming KD+AT, but the actual values were slightly larger than the ones on the paper. For example, the original paper gave an error of 0.09 for Zero-Shot on SVHN whereas in our case the error was 0.257. This can be due to the fact that different models may have been used. As mentioned before, the paper does not give details about the models used to obtain the results in Table 2 of the original paper. Therefore, the used models in this case might have been worse which reflects a higher MTE. However, it still shows that Zero-Shot student network matches the teacher's transition curves between classes significantly better than KD+AT.

6 Discussion

While the paper was generally well explained, there were details that were lacking that would aid reproducibility. The Zero-Shot algorithm was clearly explained in the paper, however, the KD+AT algorithm was not. The general structure of the algorithm could be inferred from author's GitHub repository and from other research on knowledge distillation. The paper mentions they used "coarse hyper-parameters" for their models, however, did not give exact details, subsequently hyper-parameters from (Ferles et al. (2019)) was used as previously mentioned.

The paper also mentions that the loss function is comprised of attention loss and KL loss, however, the papers code also adds cross entropy loss which was not explained. These lack of small details made it hard to understand the implementation, nonetheless, the availability of the code on GitHub helped fill the gaps.

7 Conclusion and Future Work

Overall the paper was reproducible. The algorithms were not overly complex, well explained and most importantly our results closely matched the paper's. The results were obtained without need of extreme amount of computational power making the paper re-implementable for researchers without the availability of several expensive GPUs. However, time was definitely a constraint with Zero-Shot taking 11 hours on average while using an RTX 2070 GPU.

Future analysis could involve replacing WRNs with EfficientNets as they have been shown to perform better in general (Tan & Le (2019)). EfficientNets are also known to perform very well on the CIFAR-100 dataset and hence it would be interesting to see if Zero-Shot would achieve a higher accuracy with EfficientNets on these datasets. We initially trained our model using EfficientNets instead of WRNs, however, when using it for Zero-Shot the execution time was estimated at 60+ hours and consequently we did not have the time to confirm its accuracy. The code for this project code can be found on `https://github.com/COMP6248-Reproducability-Challenge` Github page with the name Zero-shot-Knowledge-Transfer-via-Adversarial-Belief-Matching and the original paper code can be found in `https://github.com/polo5/ZeroShotKnowledgeTransfer`.

References

Alexandros Ferles, Alexander Nöu, and Leonidas Valavanis. [RE] Zero-Shot knowledge transfer via adversarial belief matching. December 2019.

Paul Micaelli and Amos Storkey. Zero-shot Knowledge Transfer via Adversarial Belief Matching. (NeurIPS), 2019. URL `http://arxiv.org/abs/1905.09768`.

Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.