Supplementary Material for SEN: A Novel Feature Normalization Dissimilarity Measure for Prototypical Few-Shot Learning Networks

Van Nhan Nguyen[1,2], Sigurd Løkse[1], Kristoffer Wickstrøm[1], Michael Kampffmeyer[1], Davide Roverso[2], and Robert Jenssen[1]

[1] UiT Machine Learning Group, UiT The Arctic University of Norway
{sigurd.lokse,kristoffer.k.wickstrom,michael.c.kampffmeyer,robert.jenssen}@uit.no
[2] Analytics Department, eSmart Systems, 1783 Halden, Norway
{nhan.v.nguyen,Davide.Roverso}@esmartsystems.com

1 Analysis of norm

To further analyze the difference between the proposed SEN dissimilarity measure and the Prototypical Network with ring loss (Ring PN), we trained a Ring PN with target norm $R = \{113$ (as found by SEN PN on the training set), 50, 100,...,350$\}$ on the FC100 dataset and with $R = \{107$ (as found by SEN PN on the training set), 50, 100,...,350$\}$ on Mini-Imagenet. Experiments were performed for Ring PN, both with the Euclidean distance and the cosine distance. The test results are shown in Table 1.

Mini-Imagenet								
R	107	50	100	150	200	250	300	350
Euclidean	66.2%	59.4%	66.0%	67.8%	67.7%	67.7%	67.3%	67.1
Cosine	67.7%	66.7%	66.8%	67.6%	67.4%	67.9%	67.3%	67.6%
FC100								
R	113	50	100	150	200	250	300	350
Euclidean	53.2%	49.7%	52.7%	53.7%	53.3%	52.8%	53.3%	52.9%
Cosine	53.4%	49.7%	52.9%	53.8%	53.0%	53.2%	53.7%	52.7%

Table 1. Few-shot classification accuracy of the Ring PN trained with different fixed values of R on the Mini-Imagenet and the FC100 datasets.

The results show that even the "optimal" R[3] is provided to Ring PN, it still performs worse than SEN PN. We believe this is due to a fundamental difference in the objectives of SEN PN vs. Ring PN. Ring PN explicitly and equally forces all examples to have the same norm R throughout the training. This can be seen

[3] The R found by SEN, which would be unknown during training of a Ring PN.

also from the contribution of Ring PN gradients, e.g. wrt the correct prototype: $\frac{\partial J_{k^\star}(\phi)}{\partial \mathbf{z}_i} \propto (\mathbf{c}_{k^\star} - \mathbf{z}_i) - (||\mathbf{z}_i|| - R)\,\frac{\mathbf{z}_i}{||\mathbf{z}_i||}$. SEN PN, on the other hand, encourages points to have the same norm as the prototypes. This means that groups of points may have groupwise simlar norms during training, but not necessarily across the whole dataset. During training, also norms across prototypes and groups align, but in a dynamic process. In our experience, by having a different value for ϵ_p compared to ϵ_n (smaller) we encourage this groupwise behavior. It can be seen that when $\epsilon_p = \epsilon_n$, results are more similar to Ring PN. In addition, the optimal norm found by the SEN PN on the training set and the test set differ (107 vs 160 on the Mini-Imagenet). This indicates that there is no R that is good across datasets. While using the Ring loss directly forces the points to have a specific norm, SEN instead adds interdependence of the norm between datapoints. This dynamic alignment, makes the model more robust to changes across the dataset as well as acts as an additional regularization during training.

2 Analysis of distance

In recent related work, Oreshkin et al. [1] proposed a scaling approach and showed that performance of a PN using the Cosine distance is on par with a PN using Euclidean distance after adding the additional scaling hyperparameter. In order to demonstrate the effect of such a scaling hyperparameter on our proposed SEN PN, we implemented the scaling method and tested it with $\alpha = \{1, 3, 5, \dots, 23\}$ for the Cosine/SEN (CS), Cosine/Cosine (CC), Euclidean/Euclidean (PN), and SEN/SEN (SEN PN) cases. The test results are shown in Table 2. The results show that these approaches perform worse than our SEN PN, which achieves an accuracy of 54.6% (Note, the std for the results is around 0.22%).

α	1	3	5	7	9	11	13	15	17	19	21	23
CS	43.8%	49.6%	46.7%	49.0%	49.1%	49.9%	50.0%	51.6%	51.9%	**52.1%**	52.0%	52.0%
CC	44.9%	46.2%	47.4%	48.9%	48.7%	50,8%	51.4%	52.6%	52.1%	**53.5%**	53.2%	53.1%
PN	52.4%	53.1%	53.4%	53.3%	53.4%	53.5%	53.5%	**53.6%**	53.2%	53.3%	53.4%	52.7%
SEN PN	**54.6%**	53.4%	53.3%	53.4%	53.1%	52.6%	53.0%	53.1%	53.2%	52.9%	53.5%	53.3%

Table 2. Few-shot accuracy with the scaling method proposed by Oreshkin et al. [1] on the FC100 dataset.

3 Hyperparameter ϵ

In order to demonstrate the effect of the hyperparameter ϵ, we trained the SEN PN with different values of ϵ_n on both the Mini-Imagenet and the FC100 datasets. The results are shown in Fig. 1. The results illustrate that the accuracy

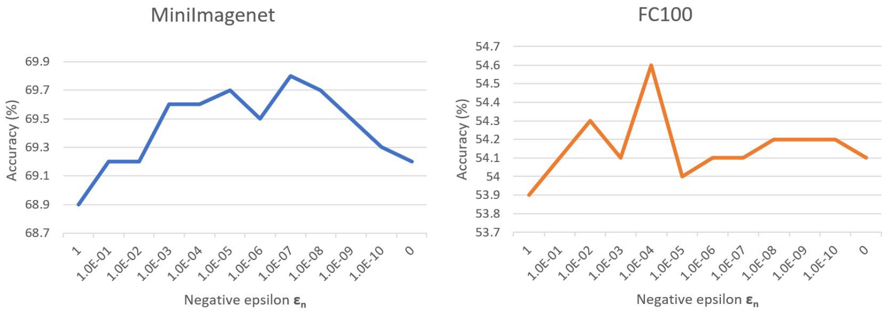

Fig. 1. The PN vs the SEN PN with different embedding sizes.

increases initially as ϵ_n decreases; however, it deceases slightly as ϵ_n decreases further. $\epsilon_n = 10^{-7}$ and $\epsilon_n = 10^{-4}$ give the best results on the Mini-Imagenet and the FC100, respectively. With the n-way few-shot classification task, there is one correct prototype, but $n-1$ incorrect prototypes for each query example. So, we use a small ϵ_n to mitigate the imbalance problem during training. Specifically, $\epsilon_p = 1$ and a small ϵ_n allow the training to focus more on forcing examples of the same class to have the same norm in the beginning of the training. The focus is then gradually shifted towards forcing examples of different classes to also have the same norm.

4 Alternative Design choices

We tested two alternative design choices for the dissimilarity measure on the FC100 dataset: (i) summation of the two squared roots of each distance and (ii) using the cosine distance instead of the Euclidean distance in SEN. The results are (i) 40.7% and (ii) 46.7%, respectively. Both approaches perform considerable worse than the SEN PN, which achieves an accuracy of 54.6%. Looking at the gradients for the first alternative, for example, we obtain for the correct prototype $\frac{\partial J_{k^\star}(\phi)}{\partial \mathbf{z}_i} \propto \frac{(\mathbf{c}_{k^\star}-\mathbf{z}_i)}{d_e(\mathbf{z}_i,\mathbf{c}_{k^\star})} + \epsilon_p \frac{(||\mathbf{c}_{k^\star}||-||\mathbf{z}_i||)\frac{\mathbf{z}_i}{||\mathbf{z}_i||}}{d_n(\mathbf{z}_i,\mathbf{c}_{k^\star})}$. This reveals a constant scaling of gradients that seems to impede training, where for instance points that are dissimilar to the correct prototype in terms of Euclidean distance induce a smaller gradient compared to the SEN.

References

1. Oreshkin, B.N., Rodriguez, P., Lacoste, A.: Tadam: task dependent adaptive metric for improved few-shot learning. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems. pp. 719–729. Curran Associates Inc. (2018)