SEN: A Novel 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] Robert Jenssen[1]
[1]UiT Machine Learning Group [2]Analytics Department, eSmart Systems, Halden, Norway

Abstract

In this paper, we equip Prototypical Networks (PNs) with a novel dissimilarity measure to enable discriminative feature normalization for few-shot learning. The embedding onto the hypersphere requires no direct normalization and is easy to optimize. Our theoretical analysis shows that the proposed dissimilarity measure, denoted the Squared root of the Euclidean distance and the Norm distance (SEN), forces embedding points to be attracted to its correct prototype, while being repelled from all other prototypes, keeping the norm of all points the same. The resulting SEN PN outperforms the regular PN with a considerable margin, with no additional parameters as well as with negligible computational overhead.

1. Introduction

One of the most common approaches to few-shot classification is distance metric learning-based [7, 11, 8]. The basic idea of this approach, for which the so-called Prototypical Networks (PNs) [7] are the most well-known examples, is to learn a non-linear mapping of the input into an embedding space which is commonly high-dimensional. In this space, a metric distance is defined that maps similar examples close to each other in the embedding space. Dissimilar examples are mapped to distant locations relative to each other, so that a query example can be classified by, for example, using nearest neighbor methods. Arguably one of the most commonly used distance metrics in this high dimensional embedding space is the (squared) Euclidean distance combined with a softmax function [7, 11, 1, 5].

However, even though the softmax is known to work well for closed-set classification problems, it has been shown to not be discriminative enough in problems were there are few labels relative to the number of classes [2]. This has given rise to alternative loss formulations with improved discriminative ability, where high-dimensional features have been normalized explicitly to lie on a hypersphere via direct L_2 normalization [2, 10]. The advantage of normalization has been theoretically analyzed in [14]. However, direct L_2 normalization leads to a non-convex loss formulation, which typically results in local minima generated by the loss function itself [14].

With the aim of performing *soft* feature normalization while preserving the convexity and the simplicity of the loss function, we equip PNs with a novel dissimilarity measure particularly suited to enable discriminative feature normalization for few-shot learning, without any direct normalization. The proposed dissimilarity measure, denoted the Squared root of the Euclidean distance and the Norm distance (SEN), replaces the Euclidean distance in PN training, with major consequences: Our theoretical analysis shows that the proposed measure explicitly forces embedded points to be attracted to the correct prototype and repelled from incorrect prototypes. Further, we provide analysis showing that SEN indeed explicitly forces all embeddings to have the same norm during training which enables the resulting SEN PN to generate a more robust embedding space. With this minimal but important modification, the SEN PN outperforms the original PN by a considerable margin and demonstrates good performance with no additional parameters as well as negligible computational overhead.

2. Prototypical Networks

Prototypical networks learn a non-linear embedding function $f_\phi : \mathbb{R}^D \rightarrow \mathbb{R}^E$ parameterized by ϕ that maps a D-dimensional feature vector of an example $\mathbf{x}_i$ to an E-dimensional embedding $\mathbf{z}_i = f_\phi(\mathbf{x}_i)$ [7]. After training, the embedding function f_ϕ is employed for mapping examples in the support set $S = \{(\mathbf{x}_j, y_j)\}_{j=1}^{N_S}$ into the embedding space. An E-dimensional representation $\mathbf{c}_k$, or *prototype*, of each class is computed by taking the mean of the embedded support points belonging to the class. An embedded query point $\mathbf{x}_q$ is then classified by simply finding the nearest class prototype in the embedding space. To train PNs, the episodic training strategy proposed in [9, 6] is adopted. Specifically, a distribution over classes for each query point

$\mathbf{x}_q \in Q$ based on a softmax over distances to the prototypes in the embedding space is produced:

$$p_\phi(y = k|\mathbf{x}_q) = \frac{\exp(-d(f_\phi(\mathbf{x}_q), \mathbf{c}_k))}{\sum_{k'} \exp(-d(f_\phi(\mathbf{x}_q), \mathbf{c}_{k'}))}, \quad (1)$$

where $d = \mathbb{R}^E \times \mathbb{R}^E \rightarrow [0, +\infty)$ is a distance function. Based on that, the PN is trained by minimizing the negative log-probability of the true class k via Stochastic Gradient Descent (SGD).

PNs employ the squared Euclidean distance as the distance metric. Although combining the softmax and the Euclidean distance has shown to give good performance for closed-set classification settings, it performs sub-optimally when few labels are available relative to the number of classes. In order to address this issue and improve the discriminative ability, new loss formulations based on feature normalization have been proposed. These tend to normalize features explicitly via L_2 normalization [10, 2]. This typically results in a more compact embedding space than the Euclidean embedding space. In such an embedding space, the cosine distance is commonly chosen as the distance metric and many few-shot classification approaches [9, 6] have employed the cosine distance in the hyperspherical embedding space.

However, feature normalization through hard normalization operations such as L_2 normalization leads to a non-convex loss formulation, which typically results in local minima introduced by the loss function itself [14]. Since the network optimization itself is non-convex, it is important to preserve convexity in loss functions for more effective minimization.

One possible solution is to use Ring loss [14]. The Ring loss introduces an additional term to the primary loss function, which penalizes the squared difference between the norm of samples and a learned target norm value R.

Since the Ring loss encourages the norm of samples to be equal to R during training instead of explicit enforcing it through a hard normalization operation, the convexity in the loss function is preserved. However, the Ring loss is more difficult to train than the primary loss (e.g., the Softmax loss) due to the added term (the norm difference L_R), the added parameter (the target norm R), and the added hyperparameter (the loss weight w.r.t to the primary loss γ).

To address the shortcomings outlined above, we propose a novel dissimilarity measure for few-shot learning, called SEN. The SEN dissimilarity measure encourages the norm of samples to have the same value, in other words, force the data to lie on a scaled unit hypersphere, while preserving the convexity and the simplicity of the loss function.

3. SEN for Prototypical Networks

The SEN dissimilarity $d_s(\mathbf{z}, \mathbf{c})$ between two arbitrary points $\mathbf{z} = (z_1, \ldots, z_n)$ and $\mathbf{c} = (c_1, \ldots, c_n)$ in D-dimensional space is a combination of the standard squared Euclidean distance d_e and the squared norm distance d_n:

$$d_s(\mathbf{z}, \mathbf{c}) = \sqrt{d_e(\mathbf{z}, \mathbf{c}) + \epsilon d_n(\mathbf{z}, \mathbf{c})}, \quad (2)$$

where ϵ is a tunable balancing hyperparameter and must be chosen such that $d_e(\mathbf{z}, \mathbf{c}) + \epsilon d_n(\mathbf{z}, \mathbf{c})$ is always positive, $d_e(\mathbf{z}, \mathbf{c})$ and $d_n(\mathbf{z}, \mathbf{c})$ are defined as:

$$d_e(\mathbf{z}, \mathbf{c}) = \|\mathbf{z} - \mathbf{c}\|^2 \text{ and } d_n(\mathbf{z}, \mathbf{c}) = (\|\mathbf{z}\| - \|\mathbf{c}\|)^2.$$

We modify the PN by replacing the Euclidean distance by our proposed SEN dissimilarity measure. We call this model SEN PN. Specifically, we replace the distance function $d(\mathbf{z}_i, \mathbf{c}_k)$ in Equation 1 by our proposed SEN dissimilarity measure $d_s(\mathbf{z}_i, \mathbf{c}_k) = \sqrt{d_e(\mathbf{z}_i, \mathbf{c}_k) + \epsilon d_n(\mathbf{z}_i, \mathbf{c}_k)}$, $\mathbf{z}_i$ is the embedding of the example $\mathbf{x}_i$, and $\mathbf{c}_k$ is the prototype of class k. For simplicity, we consider the setting in which only one query example per class is used; however, the loss function presented in this session and the analysis presented in the next section can be easily generalized for other settings in which more than one query examples per class are used. When only one query example per class is used, the updated negative log probability loss is given as:

$$\begin{aligned} J(\phi) &= -\sum_k \log p_\phi(y_i = k|\mathbf{x}_i) \\ &= -\sum_k \log \frac{\exp(-d_s(\mathbf{z}_i, \mathbf{c}_k))}{\sum_{k'} \exp(-d_s(\mathbf{z}_i, \mathbf{c}_{k'}))} \\ &= \sum_k \left(d_s(\mathbf{z}_i, \mathbf{c}_k) + \log \sum_{k'} \exp(-d_s(\mathbf{z}_i, \mathbf{c}_{k'})) \right). \end{aligned} \quad (3)$$

The learning proceeds by minimizing $J(\phi)$ of the true class k via SGD, which is equivalent to minimizing the SEN dissimilarity measure between the query example $\mathbf{x}_i$ and its prototype $\mathbf{c}_k$: $d_s(\mathbf{z}_i, \mathbf{c}_k)$, and maximizing the SEN dissimilarity measures between the query example $\mathbf{x}_i$ and the other prototypes $\mathbf{c}_{k'}$: $d_s(\mathbf{z}_i, \mathbf{c}_{k'})$. Minimizing $d_s(\mathbf{z}_i, \mathbf{c}_k)$ pulls $\mathbf{z}_i$ to its own class and encourages embeddings of the same class to have the same norm. Maximizing $d_s(\mathbf{z}_i, \mathbf{c}_{k'})$ pushes $\mathbf{z}_i$ away from other classes; however it encourages embeddings of different classes to have different norms.

Since our goal is to force the data to lie on a scaled unit hypersphere, we define the balancing hyperparameter ϵ relative to $\mathbf{z}_i$ and $\mathbf{c}_k$ as follows:

$$\epsilon_{ik} = \begin{cases} \epsilon_p > 0 & \text{if } y_i = k \\ \epsilon_n < 0 & \text{if } y_i \neq k \end{cases}, \quad (4)$$

where i is the index of the embedding $\mathbf{z}_i$, y_i is the embedding's class label, and k is the class label of the prototype $\mathbf{c}_k$. During training, a positive epsilon ($\epsilon_{ik} = \epsilon_p > 0$) is used for computing the SEN dissimilarity measure between the query example $\mathbf{x}_i$ and its prototype $\mathbf{c}_k$, while a negative epsilon ($\epsilon_{ik} = \epsilon_n < 0$) is used for computing the

SEN dissimilarity measures between the query example $\mathbf{x}_i$ and the other prototypes $\mathbf{c}_{k'}$. The negative epsilon ϵ_n will inverse the effect of the norm distance when maximizing $d_s(\mathbf{z}_i, \mathbf{c}_{k'})$. In other words, maximizing $d_s(\mathbf{z}_i, \mathbf{c}_{k'})$ with a negative epsilon ϵ_n pushes $\mathbf{z}_i$ away from other classes and encourages embeddings of all classes to have the same norm. The flexibility induced by the balancing hyperparameter ϵ_{ik} makes the SEN particularly suited to enable discriminative feature normalization in PNs.

Our proposed SEN dissimilarity measure explicitly encourages the norm of samples to have the same value during training, while preserving the convexity and the simplicity of the loss function. At test time, a positive epsilon ($\epsilon_{ik} = \epsilon_p > 0$) is used for computing all dissimilarity measures.

3.1. Theoretical analysis

In this section, we provide a theoretical analysis showing that the SEN dissimilarity measure together with the special balancing hyperparameter ϵ_{ik} explicitly pulls the data to a scaled unit hypersphere during training.

The gradient contribution with respect to the correct prototype, when $k = k^* = y_i$, is given by:

$$\begin{aligned}\frac{\partial J_{k^*}(\phi)}{\partial \mathbf{z}_i} &= -\frac{1 - p_\phi(y_i = k^*|x)}{d_s(\mathbf{z}_i, \mathbf{c}_{k^*})} v(\mathbf{z}_i, \mathbf{c}_{k^*}) \\ &= -\frac{1 - p_\phi(y_i = k^*|x)}{d_s(\mathbf{z}_i, \mathbf{c}_{k^*})} v_p(\mathbf{z}_i, \mathbf{c}_{k^*}),\end{aligned} \tag{5}$$

where

$$v_p(\mathbf{z}_i, \mathbf{c}_{k^*}) = \underbrace{(\mathbf{c}_{k^*} - \mathbf{z}_i)}_{\text{attractor}} + \underbrace{\epsilon_{ik^*}(\|\mathbf{c}_{k^*}\| - \|\mathbf{z}_i\|)\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}}_{\text{norm equalizer}}. \tag{6}$$

The gradient contribution with respect to incorrect prototypes, when $k = k' \neq y_i$, is given by:

$$\begin{aligned}\frac{\partial J_{k'}(\phi)}{\partial \mathbf{z}_i} &= -\frac{0 - p_\phi(y_i = k'|x)}{d_s(\mathbf{z}_i, \mathbf{c}_{k'})} v(\mathbf{z}_i, \mathbf{c}_{k'}) \\ &= -\frac{p_\phi(y_i = k'|x)}{d_s(\mathbf{z}_i, \mathbf{c}_{k'})} v_n(\mathbf{z}_i, \mathbf{c}_{k'}),\end{aligned} \tag{7}$$

where

$$v_n(\mathbf{z}_i, \mathbf{c}_{k'}) = \underbrace{(\mathbf{z}_i - \mathbf{c}_{k'})}_{\text{repeller}} + \underbrace{\epsilon_{ik'}(\|\mathbf{z}_i\| - \|\mathbf{c}_{k'}\|)\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}}_{\text{norm equalizer}}. \tag{8}$$

From the preceding analysis, we observe the following:

1. Each gradient component contains an attractor/repeller, which encourages $\mathbf{z}_i$ to move towards the correct prototype and away from the incorrect ones.

2. From (6), it is clear that if $\|\mathbf{c}_{k^*}\| > \|\mathbf{z}_i\|$ and $\epsilon_{ik^*} > 0$, $\epsilon_{ik^*}(\|\mathbf{c}_{k^*}\| - \|\mathbf{z}_i\|)\frac{1}{\|\mathbf{z}_i\|} > 0$, such that $\|\mathbf{z}_i\|$ is encouraged to increase (and vice verca for $\|\mathbf{z}_i\| > \|\mathbf{c}_{k^*}\|$).

Model	Network	Omniglot	MI
Original PN [7]	4CONV	98.9%	68.2%
Large Margin GNN [11]	4CONV	**99.2%**	67.6%
Large Margin PN [11]	4CONV	98.7%	66.8%
RN [8]	4CONV	99.1%	65.3%
Matching Nets [9]	4CONV	98.7%	60.0%
MetaGAN + RN [13]	4CONV	**99.2%**	68.6%
Semi-Supervised PN [1]	4CONV	-	65.5%
PN (ours, baseline)	4CONV	98.6%	67.8%
SEN PN (ours)	**4CONV**	98.8%	**69.8%**
Supervised WRN PN [1]	WRN	-	69.6%
Semi-Supervised WRN PN [1]	WRN	-	70.9%
WRN PN (ours)	WRN	99.2%	71.0%
SEN WRN PN (ours)	**WRN**	**99.4%**	**72.3%**

Table 1. Accuracy on Omniglot [3] (20-way 5-shot testing) and Mini-Imagenet (MI) [6, 9] (5-way 5-shot testing).

3. Conversely, from (8), if $\|\mathbf{c}_{k'}\| > \|\mathbf{z}_i\|$ and $\epsilon_{ik'} > 0$, $\epsilon_{ik'}(\|\mathbf{z}_i\| - \|\mathbf{c}_{k'}\|)\frac{1}{\|\mathbf{z}_i\|} < 0$ (and vice verca for $\|\mathbf{z}_i\| > \|\mathbf{c}_{k'}\|$). Thus, we need $\epsilon_{ik'} < 0$ in order to ensure similar behaviors as with the correct prototype.

Observation 2) and 3) shows that the gradient contributions with respect to the correct prototype and the incorrect ones *cooperate* in order to equalize the norms during training when $\epsilon_{ik^*} > 0$ and $\epsilon_{ik'} < 0$.

4. Experiments

We compare our SEN PN approach with the original PN [7] and state-of-the-art distance metric learning-based approaches on the Mini-Imagenet [6, 9] and the Omniglot [3] dataset. Further, additional ablation studies are also performed on the Fewshot-CIFAR100 (FC100) [4] dataset.

4.1. Experimental Setup and Results

Embedding networks We utilize the same embedding network as that used by the original PN. To test the performance of the SEN dissimilarity measure in more general settings, we employ a more sophisticated network, the Wide Residual Network (WRN) [12], as the embedding network. We train the network with both the Euclidean distance (WRN PN) and the SEN (SEN WRN PN).

Results The test results are shown in Table 1. As can be seen from Table 1, although our implementation of the PN (the baseline model) achieves 0.4 percentage points lower in terms of accuracy compared to the original implementation of the PN (67.8% vs 68.2%), the baseline model trained with the SEN still outperforms the original PN by obtaining a relative increase of 2.4% and achieves an accuracy of 69.8%. In addition, the SEN WRN PN outperforms the Semi-Supervised WRN PN by a relative increase of 2% and achieves an accuracy of 72.3% with the WRN as the embedding network. Similar trends can be observed for the

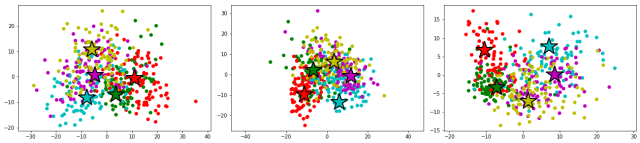

Figure 1. 2D embeddings produced by the PN (left), the Ring PN (middle) and the SEN PN (right). The circles denote query examples, and the stars denotes prototypes.

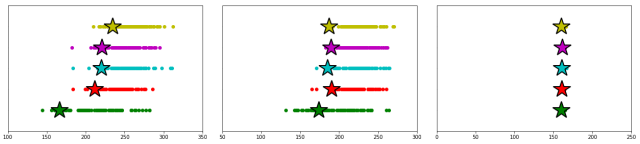

Figure 2. The norm of embeddings produced by the PN (left), the Ring PN (middle), and the SEN PN (right). The stars denote query examples, and the diamonds denotes prototypes.

Model	Omniglot	Mini-Imagenet	FC100
PN	98.6%	67.8%	52.4%
Ring PN	98.7%	68.6%	52.8%
SEN PN	**98.8%**	**69.8%**	**54.6%**

Table 2. Accuracy on Omniglot [3] (20-way 5-shot), Mini-Imagenet [6, 9] (5-way 5-shot), and FC100 [4] (5-way 5-shot).

Omniglot dataset, where SEN PN outperforms our PN implementation and SEN WRN PN outperforms WRN PN.

4.2. Ablation Study

We first compare against the PN trained with the Euclidean distance (PN), the PN trained with the Ring loss (Ring PN), and the PN trained with the SEN (SEN PN). The test results are show in Table 2. As can be seen from Table 2, the Ring loss improves the accuracy relative to the PN on the Mini-Imagenet dataset by 1.8%; however, it performs worse than our SEN PN approach, which obtains a relative increase of 3%. Similar behavior is obtained for other few-shot learning datasets such as FC100 and Omniglot.

Next, we project 1600D embeddings produced by the PN, the Ring PN, and the SEN PN to 2D space using PCA and visualize the outputs (see Figure 1). As can be seen from Figure 1, the Ring loss forces the prototypes to lie on a scaled unit hypersphere; however, the prototypes produced by the Ring PN are not very well-separated compared to the ones produced by the PN. On the other hand, our proposed SEN dissimilarity measure both forces the prototypes to lie on a scaled unit hypersphere and keeps them well-separated.

Then, we plot the norm of embeddings produced by the PN, the Ring PN, and the SEN PN. As can be seen from Figure 2, the norm of embeddings produced by the PN and the Ring PN vary a lot, while the norm of embeddings produced by the SEN PN has a very consistent value. This confirms that SEN encourages all embeddings to have the same norm during training and is a more suitable choice for feature normalization than the Ring loss in training PNs.

5. Conclusion

In this paper, we propose a novel dissimilarity measure, SEN, for distance metric learning-based few-shot learning and incorporate it into the prototypical network. With minimal modifications, the SEN PN outperforms the original PN by a considerable margin. We provide both theoretical and empirical analysis of the proposed dissimilarity measure.

References

[1] Rinu Boney and Alexander Ilin. Semi-supervised few-shot learning with prototypical networks. *CoRR*, abs/1711.10856, 2017.

[2] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019.

[3] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. One shot learning of simple visual concepts. In *COGSCI*, volume 33, 2011.

[4] Boris N Oreshkin, Pau Rodriguez, and Alexandre Lacoste. Tadam: task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, pages 719–729. Curran Associates Inc., 2018.

[5] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. *CoRR*, abs/1803.00676, 2018.

[6] Ravi Sachin and Larochelle Hugo. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[7] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, pages 4080–4090. Curran Associates Inc., 2017.

[8] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, pages 1199–1208. IEEE, 2018.

[9] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, pages 3637–3645. Curran Associates Inc., 2016.

[10] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L2 hypersphere embedding for face verification. In *ACM Multimedia*, pages 1041–1049. ACM, 2017.

[11] Wang Yong, Wu Xiao-Ming, Li Qimai, Gu Jiatao, Xiang Wangmeng, Zhang Lei, and O. K. Li Victor. Large margin few-shot learning. *CoRR*, abs/1807.02872, 2018.

[12] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *CoRR*, abs/1605.07146, 2016.

[13] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. Metagan: an adversarial approach to few-shot learning. In *NeurIPS*, pages 2371–2380. Curran Associates Inc., 2018.

[14] Yutong Zheng, Dipan K Pal, and Marios Savvides. Ring loss: Convex feature normalization for face recognition. In *CVPR*, pages 5089–5097. IEEE, 2018.