ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

Active One-Shot Learning with Prototypical Networks

Rinu Boney and Alexander Ilin

Aalto University
Espoo, Finland

Abstract. We consider the problem of active one-shot classification where a classifier needs to adapt to new tasks by requesting labels for one example per class from (potentially many) unlabeled examples. We propose a clustering approach to the problem. The features extracted with Prototypical Networks [1] are clustered using K-means and the label for one representative sample from each cluster is requested to label the whole cluster. We demonstrate good performance of this simple active adaptation strategy using image data.

1 Introduction

Few-shot learning addresses the problem of generalizing to new concepts or tasks from a few samples. In few-shot classification, the task is to adapt a classifier to previously unseen classes from just a few examples [2, 3, 4, 5]. Few-shot learning is possible by transferring knowledge from experience with similar tasks from the past. A popular approach to few-shot learning is meta-learning, in which a model is explicitly trained to adapt to new tasks in a few samples, on a wide variety of tasks [6, 4, 3, 5]. Few-shot learning is important in many practical applications due to the challenges involved in manually labeling data. Recently, the problem of few-shot learning has also been extended to the setting of semi-supervised few-shot learning where it is assumed that each task consists of few labeled samples and potentially many unlabeled samples [7, 8, 9].

In some real-world problems, the tasks to which a learning system needs to adapt initially consists of many unlabeled samples. For example, a common feature of photo management applications is the automatic organization of images based on limited interactive supervision from the user. The photos do not contain any labels according to the likes of a user and the classes relevant to a specific user are likely to be different from the classes in publicly available image datasets such as ImageNet. Thus there is a need for understanding the unlabeled data, actively requesting labels from the user and adapting to it. We consider a constrained setting of this problem where the number of classes N in the unlabeled data are previously known and the system is able to request labels for N samples from the user. This is the problem of active one-shot learning that we consider in this paper.

In this paper, we extend the semi-supervised few-shot classification approach from [7] to active one-shot learning. We observe that Prototypical Networks (PN) [1] tend to produce clustered data representations. We view the semi-supervised few-shot learning problem through the lens of semi-supervised *clus-*

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

tering. We take inspiration from [10] and propose a simple approach to enable adaptation to new classification tasks using feedback from the user. We argue that this approach can be practical in many real-world applications, as many use cases of semi-supervised few-shot adaptation imply interaction with a user and therefore active learning is often possible.

We use the following formulation of the one-shot active classification problem. There is a training set which consists of a large set of classes and we have access to labeled samples from each class in the training set. At test time, the task is to separate samples from N previously unseen classes by requesting N samples from the user. We follow the recent literature and use the episodic regime of training and evaluation, as we explain in the following section.

2 Prototypical Networks

The episodic training of PN iterates between the following steps. A subset of N classes is randomly selected to formulate one training task. For each training task, a support set $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$ and a query set $Q = \{(\mathbf{x}_{n+1}, y_{n+1}), ..., (\mathbf{x}_{n+m}, y_{n+m})\}$ are created by sampling examples from the selected classes, where $\mathbf{x}_j$ are inputs and y_j are the corresponding labels.

Prototypical Networks compute representations of the inputs $\mathbf{x}$ using an embedding function g parameterized with $\boldsymbol{\theta}$: $\mathbf{z} = g(\mathbf{x}, \boldsymbol{\theta})$. Each class c is represented in the embedding space by a prototype vector which is computed as the mean vector of the embedded inputs for all the examples S_c of the corresponding class c:

$$\mathbf{m}_c = \frac{1}{|S_c|} \sum_{(\mathbf{x}_j, y_j) \in S_c} g(\mathbf{x}_j, \boldsymbol{\theta}) \,. \tag{1}$$

The distribution over predicted labels y for a new sample is computed using softmax over negative distances to the prototypes in the embedding space:

$$p(y = c | \mathbf{x}, \{\mathbf{m}_c\}) = \frac{\exp(-d(\mathbf{z}, \mathbf{m}_c))}{\sum_{c'} \exp(-d(\mathbf{z}, \mathbf{m}_{c'}))} \,. \tag{2}$$

Parameters $\boldsymbol{\theta}$ are updated so as to improve the likelihood computed on the query set:

$$\sum_{(\mathbf{x}_j, y_j) \in Q} \log p(y = y_j | \mathbf{x}_j, \{\mathbf{m}_c\}) \,,$$

which is computed using (2) with the estimated prototypes.

3 Active One-Shot Learning

In this paper, we extend the semi-supervised few-shot classification approach from [7] to active one-shot learning. There are two sources of errors which the semi-supervised adaptation algorithm proposed in [7] can accumulate: 1) errors due to incorrect clustering of data, 2) errors due to incorrect labeling of the

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

clusters. The second type of errors can occur when the few labeled examples are outliers which end up closer to the prototype of another class in the embedding space. In this paper, we advocate that the most practical way to correct the second type of errors can be through user feedback, since in many applications of the semi-supervised few-shot adaptation, interaction with the user is possible. This idea is inspired by the work of [10] who introduced a clustering approach that allows a user to iteratively provide feedback to a clustering algorithm.

Consider the previously introduced example of few-shot learning in photo management applications. Although it is possible to ask the user to label a few photographs and use those labels to classify the rest of the pictures, it is extremely difficult and tiresome for the user to scroll through all the photos and decide which samples should be labeled. Instead, using the observation that "It is easier to criticize than to create" [10], one can initially cluster the photos and then request the user to label certain photos (or provide other types of feedback) so that the data are properly clustered and labeled. The user can provide feedback in various forms and therefore can effectively introduce various constraints that can further guide the clustering process. For example, a user can assign the whole cluster to a particular class, assign a sample to a particular cluster, mark that a particular sample does not belong to the assigned cluster, split and combine clusters. These constraints could be easily induced in basic clustering algorithms such as K-means. For examples, [11] introduced constraints between samples in the data set such as must-link (two samples have to be in the same cluster) and cannot-link (two samples have to be in different clusters) and the clustering algorithm finds a solution that satisfies all the constraints.

Even outside the context of few-shot learning, this active learning approach can be used to adapt a pre-trained classifier. Assume that we have a classifier that clusters the classes of a particular classification task such as ImageNet. Then, during test time it is possible to interactively split clusters to make coarse-grained classifications or to assign multiple clusters to a super-cluster (to make hierarchical predictions).

In this paper, we assume that the user can provide feedback only in the form of labeling a particular sample or labeling the whole cluster. We propose to use Prototypical Network as a feature extractor, cluster the samples in the embedding space using K-means and then label the clusters by requesting one labeled example for each cluster from the user. For each cluster c', we choose sample $\mathbf{z}_{c'}$ to be labeled by the user by maximizing an acquisition function $a(\mathbf{z}, c')$:

$$\mathbf{z}_{c'} = \max_{\mathbf{z} \in U_{c'}} a(\mathbf{z}, c') \,,$$

where $U_{c'}$ is the set of embedded inputs belonging to cluster c'. We explore a few acquisition functions:

- **Random**: Sample a data point uniformly at random from each cluster. This is a baseline approach.

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0.
Available from http://www.i6doc.com/en/.

- **Nearest**: Select the data point which is closest to the cluster center:

$$a(\mathbf{z}, c') = -d(\mathbf{z}, \mathbf{m}_{c'}) \,,$$

  where $\mathbf{m}_{c'}$ is the mean (cluster center) of cluster c'.

- **Entropy**: Select the sample with the least entropy:

$$a(\mathbf{z}, c') = \sum_{c} p(y = c|\mathbf{z}) \log p(y = c|\mathbf{z})$$

  Thus, we select a sample with the least uncertainty that it belongs to a certain cluster.

- **Margin**: Select a sample with the largest margin between the most likely and second most likely labels.

$$a(\mathbf{z}, c') = p(y = c_1(\mathbf{z})|\mathbf{z}) - p(y = c_2(\mathbf{z})|\mathbf{z})$$

  where $c_1(\mathbf{z})$ and $c_2(\mathbf{z})$ are the most likely and the second most likely clusters of embedded input $\mathbf{z}$ respectively. This quantity was proposed as a measure of uncertainty by [12].

We also try to simulate a case when the user can label the whole cluster, as in some applications it can certainly be possible. This approach directly measures the clustering accuracy and we call it "oracle".

- **Oracle**: We label each cluster based on the distance of the cluster mean to the prototypes computed from the true labels of all the samples.

4 Experiments

We tested the proposed method on the miniImagenet recognition task proposed by [4]. The dataset consists of downsampled 84x84 images from 64 training classes, 12 validation classes, and 24 test classes from ImageNet. We use the same split as [5]. At test time, every task contains N classes with M unlabeled examples from each class and the system is allowed to request labels for any N samples from the total $N \times M$ samples. The accuracy of the method is evaluated against the ground truth labels of the unlabeled examples in the test set. We evaluate the model over 2400 tasks from the 24 classes reserved for testing.

In the experiments with miniImagenet we use a Prototypical Network (PN) trained in the episodic mode as the feature extractor. We simulate active learning on test tasks by first doing K-means clustering in the PN embedding space and then requesting one labeled example for each cluster using the acquisition functions described earlier. Note that multiple clusters can be labeled to the same class if the requested labels guide it that way. We observe that this is the largest source of error. Table 1 presents the classification performance of each strategy for test tasks with a varying number of unlabeled samples, where the

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

	M	Random	Nearest	Entropy	Margin	Oracle
PN (ours)	15	49.19	54.42	53.95	**56.12**	58.96
	30	49.23	54.73	56.02	**57.58**	60.27
	60	50.73	56.12	57.63	**59.24**	62.09
	120	50.74	57.45	57.88	**61.42**	63.23
Resnet PN	15	51.10	57.50	58.24	**60.00**	62.44
	30	51.16	57.63	59.45	**60.29**	62.94
	60	51.36	57.68	59.77	**60.43**	63.21
	120	51.56	58.09	60.19	**60.49**	63.71

Table 1: Average 1-shot classification accuracy on miniImagenet of the proposed method for different number of unlabeled samples per class (M) available at test time. For comparison, the 1-shot accuracy (one labeled sample per class and no unlabeled samples) of PN and Resnet PN are **48.06%** and **51.69%** respectively.

method was allowed to request labels for one example per class. There, we also present the accuracy of the oracle clustering. It can be seen that the active learning strategies perform significantly better than the *random* baseline. Overall, the *margin* approach worked best in our experiments. The 1-shot classification accuracy with 120 unlabeled samples per class even surpassed the 5-shot accuracy of some well-recognized previous methods. We tested the method with the same two architectures used in [7]: 1) a four layer convolutional network, 2) a Wide Residual Network [13]. The four layer architecture scales well with increasing the number of unlabeled samples closely matching the performance of the ResNet in the case of 120 unlabeled samples per class and even outperforming it while using the *margin* strategy.

5 Conclusion

In this paper, we extended Prototypical Networks to adapt to new classification tasks in the active few-shot learning scenario, where a task consists of many unlabeled examples from unseen classes and it is possible to request labels for one example per class from the user. Contrary to the semi-supervised few-shot learning setting where the labeled samples are provided beforehand, we advocated that in many real-world applications it can be possible to request the few labeled examples from the user, which can yield better performance. We proposed to use the clustering approach to semi-supervised classification where the samples are first clustered and then labeled by active interaction with the user. This is different to recent deep semi-supervised learning papers which constrain the classifier using unlabeled data.

The proposed solution of active one-shot adaptation is based on doing K-means clustering in the embedding space found by Prototypical Networks. These two methods make a good fit because they make similar assumptions about the data distribution: In Prototypical Networks, the distribution of each class is represented by its mean and the variances of class distributions are assumed

ESANN 2019 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2019, i6doc.com publ., ISBN 978-287-587-065-0. Available from http://www.i6doc.com/en/.

equal. The same assumptions are made by K-means.

The fundamental bottleneck of the proposed approach in improving the classification performance is the ability of the feature extractor to *cluster unseen data*. Although we used an embedding network trained using Prototypical Networks, the adaptation mechanisms proposed in this paper can be performed using other feature extractors as well. A feature extractor explicitly trained to cluster data can further improve the few-shot classification performance and this is an area of active research [14, 15, 16]. Building feature extractors that allow better generalization is largely an unsolved problem and it requires further exploration.

References

[1] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

[2] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[3] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015.

[4] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016.

[5] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. 2016.

[6] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017.

[7] Rinu Boney and Alexander Ilin. Semi-supervised few-shot learning with prototypical networks. *NIPS Workshop on Meta-Learning*, 2017.

[8] Rinu Boney and Alexander Ilin. Semi-supervised few-shot learning with maml. *ICLR Workshop*, 2018.

[9] 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. In *International Conference on Learning Representations*, 2018.

[10] David Cohn, Rich Caruana, and Andrew McCallum. Semi-supervised clustering with user feedback. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*, 4(1):17–32, 2003.

[11] Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. Constrained k-means clustering with background knowledge. In *ICML*, volume 1, pages 577–584, 2001.

[12] Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden Markov models for information extraction. In *International Symposium on Intelligent Data Analysis*, pages 309–318. Springer, 2001.

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

[14] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pages 4004–4012. IEEE, 2016.

[15] Hyun Oh Song, Stefanie Jegelka, Vivek Rathod, and Kevin Murphy. Deep metric learning via facility location. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[16] Marc T Law, Raquel Urtasun, and Richard S Zemel. Deep spectral clustering learning. In *International Conference on Machine Learning*, pages 1985–1994, 2017.