Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 54-16 (2021) 444–449

Few-Shot Open World Learner

Andreas L. Teigen * **Aya Saad** ** **Annette Stahl** *** **Rudolf Mester** ****

* *Dept. of Computer Science, Norwegian University of Science and Technology, NTNU, Trondheim, Norway (e-mail: andreas.l.teigen@ntnu.no)*
** *Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, NTNU, Trondheim, Norway (e-mail: aya.saad@ntnu.no)*
*** *Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, NTNU, Trondheim, Norway (e-mail: annette.stahl@ntnu.no)*
**** *Dept. of Computer Science, Norwegian University of Science and Technology, NTNU, Trondheim, Norway (e-mail: rudolf.mester@ntnu.no)*

Abstract: Computer vision based recognition systems in dynamically changing environments require continuously updating datasets with novel detected categories while maintaining equally high performance on previously established classes. These requirements can be addressed by the concept of *Open World Learning* introduced by (Chen et al., 2018).
We propose a novel framework as a solution to the open world learning problem. This solution is based on the *few-shot* classification strategy in combination with an outlier detection module. The few-shot classifiers utilized in this work provide a similarity-based classification scheme and present an excellent solution to the incremental learning problem natively, requiring no training and only a few labeled examples of new classes before adapting to them. The discovery of new classes is performed by an outlier detection module that utilizes the similarity space created by the few-shot classifier to identify sufficiently different samples from the known classes and remove them from the classification process.
This paper highlights the best combination of few-shot and outlier detection algorithms. Extensive experiments with different combinations of these algorithms are conducted in the pursuit of the ideal components of the framework. Results of the experiments show that we are successful in creating a novel implementation of an open world learner with a very limited loss of accuracy compared to the base few-shot algorithm.

Keywords: few-shot, open-world, image classification, Prototypical network, outlier detection

1. INTRODUCTION

Artificial intelligence based on machine learning algorithms has reached human, or even superhuman performance on a variety of tasks. However, the task of identifying new classes and learning these classes from very little data remains a challenge, as many popular methods require thousands or even millions of labeled data samples to learn the predefined classes.

As a response to the problem of quickly adapting to little data, *one-shot* and *few-shot* (Koch et al., 2015; Vinyals et al., 2016; Snell et al., 2017) learning algorithms were created. Today, they are well-established techniques in computer vision. They possess a remarkable ability to adapt to new classes, requiring a minimal amount of data. Furthermore, they possess the ability of adding new classes to the dataset with no extra training of the model.

The problem of discovering and cataloguing new classes is known as Open World Learning (Chen et al., 2018). As illustrated in figure 1 the open world problem entails the classification of open world data, and incrementally updating the classification algorithm to adapt to new classes that it might encounter.

This work uses a variation of one-shot/few-shot classifiers that learn a Euclidean-based similarity space instead of individual class features. The distances between the objects in this space are a result of their perceived similarity, and it is used as a classification metric. Our hypothesis is that this similarity metric can not only be used for closed world classification, but can be leveraged to identify objects belonging to unknown classes.

The main contribution of this work is a novel vision based open world learner framework combining a few-shot learner with an outlier detection algorithm. In the

⋆ This research is funded by the Research Council of Norway (RCN) IKTPLUSS program (project number 262741) and supported by NTNU AMOS (RCN project number 223254). acknowledgments here.

10.1016/j.ifacol.2021.10.129

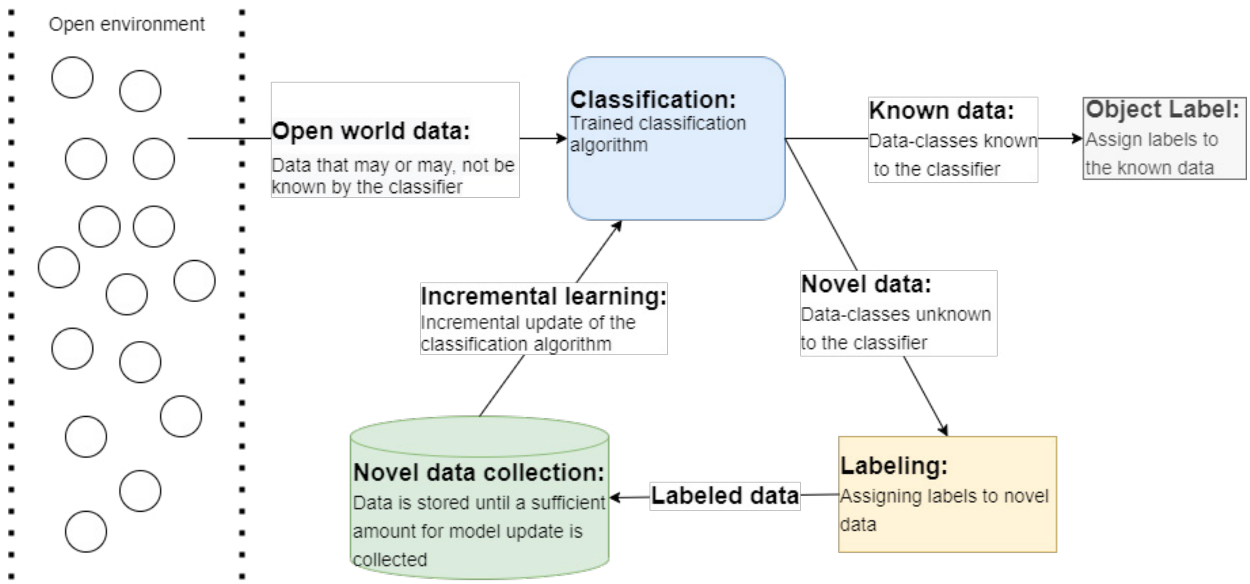

Fig. 1. Overview of an Open World Learner problem - A classifier is presented with open world data and sorts it into known data and novel data. The known data is classified to a known classes while the novel data is recorded and labeled. When enough data for a newly identified class is collected, the classification algorithm can be updated to account for the new class.

framework the few-shot learner performs distance base similarity classification, and the outlier detector enables the discovery of objects belonging to classes not previously encountered by the model. The combination of both few-shot learners and outlier detectors into one framework allows for the identification of new classes while iteratively learning to classify them accurately.

We support our contribution with an experimental investigation on the viability of using the few-shot learner, outlier detector combination in the context of an open world vision-based setting. We benchmark the proposed framework in the plankton imaging classification domain to demonstrate its applicability to underwater imaging, sampling and classification, (Saad et al., 2020, 2021). However, the framework is independent of this application and can be further generalized to other datasets.

2. RELATED WORK

Popular deep learning based methods for image recognition require a large amount of data and considerable time for training the static and closed world models. Enabling these methods to learn dynamically and efficiently is still an ongoing problem. There exist several different methods of handling this problem in the literature (Gepperth and Hammer, 2016). Some of the approaches that have received much attention in the last few years are the one-shot and few-shot approaches (Koch et al., 2015; Vinyals et al., 2016; Snell et al., 2017) utilizing similarity-based classification rather than learned class feature-based classification

The Siamese net (Koch et al., 2015) was introduced as a one-shot algorithm to target these problems in image recognition where there is very little available data. The Siamese net model requires only a single reference image from a class to use it for classification. The Siamese net works by embedding two images into a similarity space; one reference image belonging to a *known* class and one query image that can belong to any class. The distance between the embeddings is then used to determine if the query image belongs to the same class as the reference or not.

The training process utilizes a triplet loss function (Schroff et al., 2015), which rewards the models based on the distance between the embeddings. Higher distance values should be provided when the two objects belong to different classes and vise versa. This setup provides more flexibility to generalization compared to standard deep learning methods, which tend to relate the output values as a fixed percentage of the reference images such as in (Krizhevsky et al., 2012).

The few-shot learning algorithm *Matching Network* (Vinyals et al., 2016) is in many ways an extension on the Siamese net, increasing the generalizability degree by adapting it to a few-shot and multi-class setting. Rather than using one reference image, it utilizes N reference images from k classes. This is referred to as the k-way, N-shots setting. The set containing these $k * N$ images is denoted as the support set. The term "Matching" denotes the formation of distinct class clusters in the similarity embedding space shaped by objects belonging to the same classes.

By calculating "attention kernels" for different classes in the embedding space, the classifications can be performed by assigning the input image to the class with the highest density at the image's embedding location. In order to deal with the problem of closely related classes overlapping in the embedding space, (Vinyals et al., 2016) included an Long Short Term Memory (LSTM) network to shift the weights of the embedding network based on the attention scope of the reference images.

As an example: the task of classifying different species of birds requires a narrower scope than differentiating between a bird and a dog.

The design changes introduced by (Vinyals et al., 2016) allows for use of more images from more classes and allow for multi-class classification increased the classification accuracy, but this comes at the cost of the Siamese net's ability to reject objects that did not match the reference image.

The authors of (Snell et al., 2017) iterated on the matching network with their *Prototypical Network*, proposing a simplification by removing the LSTM network and the class cluster densities from the classification. They proposed to use class prototypes represented by the mean position of all the known reference embeddings of a class. The classification of an image then depends on its embedding's closest prototype. The utilization of the prototype decreases ambiguity of class overlap and ensures a more straightforward interpretation of the decision boundaries while simultaneously increasing the accuracy.

Both the matching network and the prototypical network improved on the Siamese network accuracy (Vinyals et al., 2016; Snell et al., 2017). However, while chasing performance and greater generalizability, these two networks lost the rejection feature and consequently, the ability to handle open world data. We propose to overcome these issues with our framework that will be explained in the following section.

Our framework is based on the prototypical network as this is a strong performer in the field of few-shot learning while possessing a very simple architecture. The Relation Network (Sung et al., 2018) does report a slight increase in performance over the Prototypical network, but does so at added complexity. In addition to this (Chen et al., 2019) notes that there is not as much as progress in the actual classification rate of the different classifiers in the field of few shot learning compared to the baseline methods based on transfer learning, resulting in not much difference between the prototypical network and the relation network in actual performance. Due to these reason we use the Prototypical network as the base algorithm for our presented framework.

3. PROPOSED FRAMEWORK

To regain the ability to discover objects belonging to unknown classes and adapt the few-shot learners to open world classification tasks, we propose a novel open world learner framework based on the prototypical network (Snell et al., 2017) in combination with the outlier detector eXtreme Gradient Boosted Outlier Detector (XGBOD) (Zhao and Hryniewicki, 2018). The framework allows for the classification of known classes and the identification of objects belonging to unknown classes. Moreover, the few-shot learners' ability to generalize allows the framework to iteratively adapt to new classes without retraining the model and only requiring a few samples of each class.

The proposed framework, as depicted in figure 2, consists of four main modules: The support dataset, the embedding module, the classification module and the outlier detection module. The four modules in combination provide a solution to an open world classification task.

The support dataset is the reference database, containing only a few samples from each known class is considered the memory of the model. The images in this dataset are used as a reference for classifying the query sample. The k-way, N-shot configuration, explained in section 2 defines how many images, are selected from each class.

The embedding module is a trained neural network to map objects' representation into a lower dimensional "embedding space". The network is optimized to decrease the distance between objects of the same class and increase the distance between objects belonging to different classes. The objective is to maximize the distances between the different clusters effectively. In one evaluation episode, this module embeds all the labeled data from the support set along with all the desired query images; then, it conveys all embeddings to both the classification module and the outlier detection module. The module itself is identical to the one used in the prototypical network, and it is trained in a closed world setting.

The classification module first calculates the k mean prototypes based on all N support embeddings from the k classes; then, it calculates the softmax values over the negative Euclidean square distance from the query to all support prototypes as in equation 1 (Goodfellow et al., 2016) where μ_k is the prototype for class k, $f_\phi(x)$ is the embedding query of x given trainable parameters ϕ and $d(a, b)$ is the distance between a and b. Lastly the classification module assigns the object to the same class as the class of the closest prototype given by $\underset{\mu_k}{\operatorname{argmax}}\, p_\phi$.

$$p_\phi(y = k|x) = \frac{exp(-d(f_\phi(x), \mu_k))}{\sum_{k'} exp(-d(f_\phi(x), \mu_{k'}))} \tag{1}$$

The outlier detection module uses the labeled embeddings of the support set from the embedding module as the input of an outlier detection algorithm to train k outlier detection models. The query images' embeddings are then evaluated against these models to identify if they are outliers.

The open world classification task is then performed by combining both the output of the classification module and the outlier detection module. If an input image for classification is predicted as an outlier of the highest scoring class, it is labeled as an unknown class; otherwise, the image is assigned to the class with the highest score.

The goal of the experimentation is to determine the feasibility of the proposed framework and to determine the best suitable outlier detection algorithm to use in combination with the prototypical network to compose the proposed framework. In doing so, we first establish a baseline accuracy for the closed world prototypical network; then, we investigate the performance of the different outlier detection algorithms based on the similarity space produced by the embedding module. Finally, we look at the combined open world classification accuracy of the outlier detection module and the classification module. All experiments are performed on planktonic datasets, namely Kaggle plank-

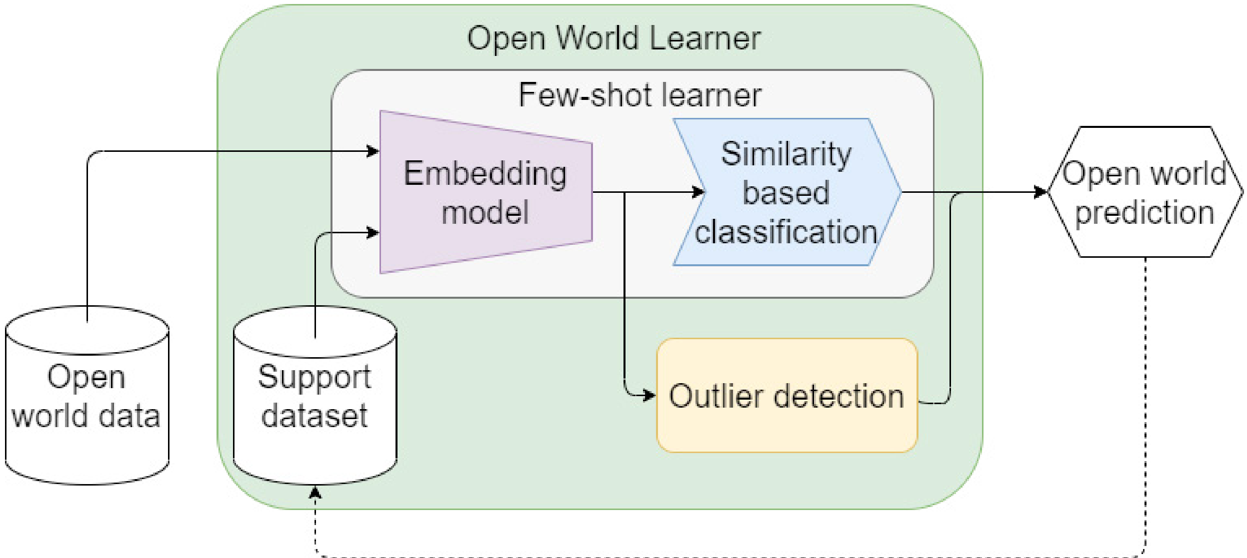

Fig. 2. Proposed open world learner architecture (few-shot + outlier detector)

ton dataset from the National Data Science Bowl (Kaggle, 2015).

3.1 Prototypical network baseline.

The training of the base prototypical network model was done in a closed world scenario in a 10-way 5-shot setting. The trained model is used as the base for all consecutive experiments. Table 1 shows the results of testing the trained model on the Kaggle dataset and in a closed world setting. The different model accuracies are obtained by varying the testing configurations, such as the number of k classes and N shots.

3.2 Pure outlier detection task

Table 2 shows the accuracy for the pure outlier detection task based on the embeddings produced by the prototypical network. This value is given in parentheses. For each evaluation episode, k individual outlier detector models were trained, one for each class per algorithm. For every unsupervised outlier detector, only the N support embeddings belonging to a class are used to create that class' outlier detection model. Both the eXtreme Gradient Boosted Outlier Detector (XGBOD) (Zhao and Hryniewicki, 2018) and the Nearest Non Outlier (NNO) (Bendale and Boult, 2015) are supervised models; they also use the $(k-1)*n$ images as negative samples. All outlier detectors except for the NNO are detailed in (Zhao et al., 2019).

The results in table 2 show that LOCI, OCSVM and XGBOD mark themselves as the stronger contenders for higher N-shot values. This superior performance is due to their design complexity, which tends to favour a large amount of data to perform well. We also highlight that the NNO algorithm is a strong performer, producing especially good results at the low-shot setting and exhibiting impressive robustness to change in N-shot value, producing nearly uniform results on all values of N.

KNN based algorithms (KNN, AvgKNN, MedKNN), on the other hand, are less complex. They produce especially good values at low shot settings but fall slightly off at the higher N-shot values. This performance variation can be related to the fine tuning of the classifiers, which is growing in complexity with an increasing number of neighbours.

3.3 Combined accuracy.

The open world classification performance of all the different possible framework variations are shown in table 2 as the values on the left in each column. The score indicates the combined accuracy of both the classification module and the outlier detection module. The top scoring outlier detector architecture, as shown in table 2, is the integrated framework having the XGBOD algorithm and the prototypical network combined. The integrated model achieves a 77.2% score on the 5-way, 30-shot configuration, a loss of 13.5% compared to the closed world setting with prototypical network as the baseline. It is worth noting that the XGBOD variation results have less than a 1% performance loss when comparing the open world classification and the standalone outlier detection accuracy. LOCI and OCSVM variations were the strongest competitors of the XGBOD in the pure rejection task, but have a significantly higher performance loss for the combined accuracy. Similarly, this larger performance loss can be observed in almost all the tested outlier detection variations. The only other algorithm achieving this marginal performance loss between the two values is the NNO variation. This phenomenon is more easily explainable since the NNO algorithm effectively learns a radial cutoff point from the prototype. The algorithm then classifies the input image and the closest prototype in the same manner as the base prototypical network. This step helps to remove most of the samples that would be difficult for the classifier to label. The NNO variation produces especially good results with low N-shot values and is seemingly independent of the number of images used per class. These results show the advantage of the NNO being a simple model that is easily optimized and therefore performs very well on small support sets. This performance, however, is lower than the XGBOD performance for higher N-shot values.

Algorithm	5-way Acc. 5-shot	20-shot	30-shot	10-way Acc. 5-shot	20-shot	30-shot
Prototypical network	84.7%	88.6%	90.7%	74.9%	81.4%	81.8%

Table 1. Testing the prototypical network on the Kaggle dataset in a closed world setup.

Algorithm	5-way Acc. 5-shot	20-shot	30-shot	10-way Acc. 5-shot	20-shot	30-shot
ABOD	47.3 (68.8)	60.4 (64.5)	57.1 (61.2)	59.0 (68.2)	56.6 (63.8)	53.1 (61.4)
AvgKNN	**71.1** (71.1)	68.0 (72.3)	65.3 (69.3)	65.5 (73.8)	64.4 (71.3)	61.1 (69.5)
COF	68.6 (73.7)	61.5 (65.8)	57.2 (61.4)	63.4 (71.8)	58.7 (65.6)	54.0 (62.0)
LMDD	47.3 (53.0)	48.0 (52.5)	48.0 (52.3)	0 43.6 (52.)	45.9 (53.2)	43.7 (52.3)
XGBOD	62.0 (62.0)	**77.0** (77.4)	**77.2** (**77.8**)	56.4 (56.7)	68.8 (69.4)	**70.4** (71.0)
Feature Bagging	70.2 (**75.5**)	71.0 (75.0)	65.6 (69.6)	64.3 (73.0)	67.3 (74.0)	61.7 (70.2)
IForest	52.1 (58.5)	53.1 (57.5)	53.0 (57.4)	46.8 (56.6)	63.7 (57.4)	49.1 (57.7)
KNN	70.4 (75.3)	68.2 (72.2)	65.6 (69.6)	**66.1** (**74.5**)	64.6 (71.5)	61.8 (70.2)
LODA	38.8 (45.6)	44.9 (49.4)	47.7 (52.0)	35.6 (45.9)	42.9 (50.4)	42.7 (51.4)
LOF	70.4 (75.3)	68.4 (72.7)	65.4 (69.5)	64.6 (72.9)	64.3 (71.1)	61.7 (70.2)
LOCI	42.6 (50.0)	72.6 (76.3)	73.2 (76.7)	38.6 (50.0)	68.6 (74.8)	69.7 (**77.0**)
MedKNN	**71.1** (71.1)	67.8 (72.0)	65.4 (69.3)	65.5 (73.8)	64.0 (70.8)	61.2 (69.5)
NNO	71.0 (72.6)	70.8 (71.6)	69.6 (71.2)	64.3 (66.2)	65.4 (66.9)	64.5 (65.2)
OCSVM	51.8 (51.8)	76.0 (**79.0**)	73.0 (73.0)	52.0 (52.2)	**71.4** (**76.8**)	69.2 (76.6)
PCA	50.0 (50.0)	66.3 (67.8)	67.1 (69.2)	50.0 (50.1)	64.9 (67.9)	64.6 (69.0)
SOD	66.5 (72.2)	64.1 (68.7)	59.5 (66.2)	61.7 (71.8)	60.2 (67.8)	58.4 (65.6)
SOS	69.5 (71.3)	67.0 (71.2)	63.4 (67.5)	65.7 (69.2)	63.7 (70.7)	59.8 (68.3)

Table 2. Open world accuracy (outlier detection accuracy) for several different k-way, N-shot settings.

3.4 Computational efficiency.

Table 3 shows the computational speed of all the potential variations of the framework. This score is given in episodes per second where one outlier detection model is trained for each of the k classes, each based on N number of samples. The tests are performed with 5 query images. The speed is measured over the outlier detection modules for all algorithms as this is the differentiator between the variations in terms of computational time.

From the table, it is clear that the accuracy of the top performing algorithms comes at a cost. Algorithms such as XGBOD and LOCI are some of the slowest of the algorithms tested. For example, the XGBOD computes the one 10-way 30-shot episode in 10 seconds. It is worth noting that algorithms such as the KNN and the OCSVM, despite their lower accuracy compared to the complex XGBOD, are highly effective when speed is a crucial requirement in the application domain.

Algorithm	5-way 5-shot	30-shot	10-way 5-shot	30-shot
ABOD	9.9	6.5	15.2	6.4
AvgKNN	185.1	24.7	87.7	13.3
COF	267.2	17.7	145.7	9.4
LMDD	5.3	0.4	3.3	0.2
XGBOD	0.3	0.1	0.2	0.1
FB	22.9	13.0	12.6	6.8
IForest	2.7	2.2	1.6	1.3
KNN	200.3	24.2	88.6	13.9
LODA	27.0	25.0	22.3	12.3
LOF	250.9	42.9	125.8	22.8
LOCI	4.2	1.0	5.7	0.6
MedKNN	134.8	19.8	65.1	10.4
NNO	111.0	4.2	32.8	1.1
OCSVM	435.3	81.5	227.1	42.3
PCA	355.3	81.5	215.6	42.4
SOD	2.0	1.6	3.2	1.9
SOS	4.6	8.3	4.3	5.9

Table 3. Speed of the outlier detection architecture variations [img/s], tested on a Nvidia GTX2080ti

4. DISCUSSION

When comparing the closed world baseline in table 1 with table 2 there is a noticeable decrease in performance incurred by the outlier recognition modules, about 10% for the top outlier detection algorithm. However, when we compare the outlier rejection accuracy to the combined accuracy scores, we see that for some models like the XGBOD and the NNO, the loss in accuracy between these two tests is less than 1% for most configurations and not greater than 2% loss for any configuration. This means that these outlier detection modules can preemptively reject samples that lead to low confidence classifications.

Unfortunately, a small amount of the errors made by the outlier detectors in the open word setting are false positive inliers, and when passed on, will ultimately be incorrectly labeled by the classification module. The framework can however be leveraged in a closed world setting where there are no false positive inliers, and reject all samples that are difficult to predict and consequently increase the confidence of classifications. A few sample tests were performed to explore this use-case, and they confirmed the validity of the claim. For instance the 5-way, 50-shot configuration resulted in 72.6% accuracy on the pure rejection task and 71.1% combined framework accuracy. Showing that after the outlier detector, the framework performs classifications with a confidence of 98%.

The high accuracy in most configurations in combination with the additional possible use-case in a closed world setting is the motivation behind presenting this as the primary advertised framework combination.

However, as is show in table 3, the high performance of the XGBOD framework variation comes with some limitations on speed. As a consequence, there are other variations of the framework more suitable for time sensitive applications. We propose the NNO variation as a more lightweight version of our framework. This variation is a much faster algorithm at the sacrifice of the accuracy at higher values of N; still, it retains the low accuracy loss between the pure rejection task and the combined framework accuracy. This configuration also excels at situations with a limited amount of support samples per class as shown in table 2.

In this work, there has not been a focus on parameter tuning. There is reason to believe that several of the tested models could produce better results given meticulous tuning of all parameters. The problem that arises with this is that the models would need a different tuning for each of the tested k-way, N-shot configurations, which is not desirable as the few-shot algorithms are designed to be flexible in this regard and we want to maintain this flexibility in a framework that is not highly dependent on tuning.

5. CONCLUSION

This work presents a novel framework for an open world learner algorithm that, in addition to performing standard classification, can also identify unknown classes and learn new classes by presenting only a few samples of labeled data to the model. From the extensive tests performed, the top scoring algorithm combination of the proposed framework is shown to be the few-shot learner; prototypical network and the outlier detector XGBOD. Out of all the tested variations, this was the combination that produced the highest total accuracy for the majority of configurations, reaching an accuracy of 77.2% for the Kaggle-plankton dataset in an open world setting when tested on data that was not part of the training set. Because of the low performance loss between the rejection accuracy and the classification accuracy, we discovered that the rejection capability can be utilized to preemptively reject samples that lead to classifications of low confidence, resulting in an accuracy classification confidence of 98% in a closed world setting.

We also highlight a variation of our framework consisting of the prototypical network and NNO for time sensitive tasks where there is very limited data representation per class. This framework variation also maintains the closed world, high confidence classification feature of the proposed framework.

The proposed framework also show promising results when applied on the MiniImageNet (Vinyals et al., 2016) as similar results were observed relative to the base performance of the prototypical network.

ACKNOWLEDGEMENTS

This research is funded by the Research Council of Norway (RCN) IKTPLUSS program (project number 262741) and supported by NTNU AMOS (RCN project number 223254). acknowledgments here.

REFERENCES

Bendale, A. and Boult, T. (2015). Towards open world recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1893–1902.

Chen, W.Y., Liu, Y.C., Kira, Z., Wang, Y.C.F., and Huang, J.B. (2019). A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*.

Chen, Z., Liu, B., Brachman, R., Stone, P., and Rossi, F. (2018). *Lifelong Machine Learning: Second Edition*.

Gepperth, A. and Hammer, B. (2016). Incremental learning algorithms and applications.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.

Kaggle (2015). URL `https://www.kaggle.com/c/datasciencebowl/data`. [Accessed 30-March-2020].

Koch, G., Zemel, R., and Salakhutdinov, R. (2015). Siamese neural networks for one-shot image recognition.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Saad, A., Bergrum, S., and Stahl, A. (2021). An instance segmentation framework for in-situ plankton taxa assessment. In *Thirteenth International Conference on Machine Vision*, volume 11605, 1160511. International Society for Optics and Photonics.

Saad, A., Stahl, A., Våge, A., Davies, E., Nordam, T., Aberle, N., Ludvigsen, M., Johnsen, G., Sousa, J., and Rajan, K. (2020). Advancing ocean observation with an ai-driven mobile robotic explorer. *Oceanography*, 33(3), 50–59.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823.

Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, 4077–4087.

Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., and Hospedales, T.M. (2018). Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1199–1208.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. In *Advances in neural information processing systems*, 3630–3638.

Zhao, Y. and Hryniewicki, M.K. (2018). Xgbod: improving supervised outlier detection with unsupervised representation learning. In *2018 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.

Zhao, Y., Nasrullah, Z., and Li, Z. (2019). Pyod: A python toolbox for scalable outlier detection. *Journal of Machine Learning Research*, 20(96), 1–7. URL `http://jmlr.org/papers/v20/19-011.html`.