Open-World Learning Without Labels

Anonymous CVPR 2021 submission

Paper ID 2301

Abstract

Open-world learning is a problem where an autonomous agent detects things that it does not know and learns them over time from a non-stationary and never-ending stream of data; in an open-world environment, the training data and objective criteria are never available at once. The agent should grasp new knowledge from learning without forgetting acquired prior knowledge. Researchers proposed a few open-world learning agents for image classification tasks that operate in complex scenarios. However, all prior work on open-world learning has all labeled data to learn the new classes from the stream of images. In scenarios where autonomous agents should respond in near real-time or work in areas with limited communication infrastructure, human labeling of data is not possible. Therefore, supervised open-world learning agents are not scalable solutions for such applications. Herein, we propose a new framework that enables agents to learn new classes from a stream of unlabeled data in an unsupervised manner. Also, we study the robustness and learning speed of such agents with supervised and unsupervised feature representation. We also introduce a new metric for open-world learning without labels. We anticipate our theories and method to be a starting point for developing autonomous true open-world never-ending learning agents.

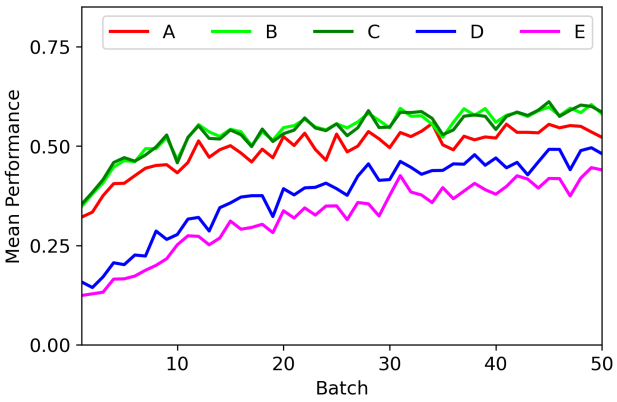

Figure 1: This paper formalizes and explores solutions to open-world learning without labels, including defining a new metric for performance measurement on such problems. Mean performance of five open-world recognition systems as they adapt to a 100 image batches of mixing known and new classes. The performance of open-world learning directly depends on the quality of feature representation, the detection of novelty, and the type of feedback during learning. **Can you determine which of the five curves (A–E) learnt the new classes in each batch using labels and which of them were learning without labels? Which of them used pure supervised feature representation, which used unsupervised features and which fused the two?** Please see experimental section for answers.

1 Introduction

Autonomous robots and self-driving vehicles are emerging technologies that are predicted to grow rapidly in quality and quantity in the near future. Vision-based recognition is an important subsystem of such autonomous agents. Visual recognition systems combine a feature extraction (perception) subsystem and inference (decision maker) subsystem. In real-world applications, environments of autonomous robots and self-driving vehicles change over time. This change will often introduce new classes, new attributes, and even a shift in the distribution of existing classes. In an open-world, the agent must detect the new classes/attributes and adapt.

Babies can detect novel objects and learn them even if they are not given a semantic label with which to associate them. Similarly, online vision-based systems, autonomous robots, and self-driving vehicles may confront new classes of objects in areas and must learn to deal with them even if they don't know the semantic label to use. These systems should first detect that these objects are new and were not in the training set. Then, they should distinguish between the new classes. Also, they should recognize the new classes when they see them again. Ideally, each of the above steps should be done in an unsupervised manner. To achieve this goal, researchers should address many challenges such as novelty detection, change point detection, feature represen-

tation, transfer learning, meta-learning, continual learning, etc. Here, we investigate open-world unsupervised class incremental learning of image classifiers for autonomous agents. Our motivation is to build fundamentals and formalize **open-world learning without labels** to be used along with other theories and solutions in the design of autonomous never-ending learning robots in the future.

Computer vision and machine learning has seen a substantial expansion in the work addressing self-supervised learning [21, 11, 10, 20, 16], unsupervised learning [8, 23, 42, 36, 31], as well as open-set/out-of-distribution research [14, 13, 15, 40, 45, 32], and incremental learning [43, 46, 18, 24, 25], and this paper combines results from these three open topics to address a new problem, the detection and continual learning of new classes in an unsupervised manner – i.e., we formalize the problem of and develop the first class of True Open-World Learning (TOWL) algorithms to address the problem of open-world learning without labels.

The contributions of this paper are:

- Formalizing open-world learning without labels problem,
- Proposing a new metric to measure the quality of open-world learning
- Creating a framework to evaluate autonomous agent's performance in both supervised and unsupervised open-world scenarios,
- Enhancing previous open-world image classifier using statistical Extreme Value Theory (EVT),
- Designing our TOWL autonomous agents that discover, characterize, and learn new classes without labels from an open-world stream of data, and
- Investigating effect of feature representation in the robustness and learning speed of autonomous agents during open-world learning.

2 Background

We cover only the background needed to develop/evaluate our problem and approach, refrence to more related work is given in section 7.

2.1 Extreme Value Theory

Extreme Value Theory (EVT) is a branch of statistics that studies the behavior of extreme events on the tails of probability distributions [12, 5, 9]. EVT estimates the probability of events that are more extreme than any of the already observed ones. EVT is an extrapolation from observed samples to unobserved samples. There are two principal parametric approaches to modeling the extremes of a probability distribution: (1) block maxima and (2) threshold exceedance. The Hill Estimator approach is also commonly used which is a non-parametric approach. The block maxima uses Generalized Extreme Value distribution (GEV) and threshold exceedance uses Generalized Pareto Distribution (GPD). According to Fisher-Tippet asymptotic theorem, for normalized maxima of blocks of random variables $M_n = \max(X_1, ..., X_n)$, there is a non-degenerate distribution, which is a GEV distribution, which for our case must follow a Weibull distribution

$$\mathrm{W}(x; \mu, \sigma, \xi) = \begin{cases} e^{-(1+\xi(\frac{x-\mu}{\sigma}))^{\xi}} & , x < \mu - \frac{\sigma}{\xi} \\ 1 & , x \geq \mu - \frac{\sigma}{\xi} \end{cases} \quad (1)$$

2.2 Extreme Value Machine

The Extreme Value Machine (EVM) [35, 19] is a distance-based kernel-free non-linear classifier that uses Weibull families distribution to compute the radial probability of inclusion of a point with respect to nearest members of other classes. For a given point x_i, they fit the Weibull on the distribution margin distance, half the distance to the nearest negative samples,

$$m_{i,j} = 0.5 * \|\hat{x}_i - x_j\| \quad (2)$$

for the τ closest points x_j from other classes. EVM provides a compact probabilistic representation of each class's decision boundary, characterized in terms of its extreme vectors. Each extreme vector has a family of Weibull distribution. Probability of a point belonging to each class is defined as the maximum probability of the point belonging to each extreme vector of the class. EVM uses greedy approximation for Karp's set cover problem for model size reduction by deleting redundant extreme vectors. In short, EVM for each input (point) computes the probability of inclusion to each class, i.e., the output is a vector of probabilities. The predicted class is computed by

$$\hat{P}(C_l|x) = \max_k \mathrm{W}_{l,k}(x; \mu_{l,k}, \sigma_{l,k}, \xi_{l,k}) \quad (3)$$

where $\mathrm{W}_{l,k}(x)$ is Weibull probability of x corresponding to k extreme vector in class l.

Source code for EVM is available, and a python version of EVM can be installed via pip. Our PyTorch enhanced version will be publicly released with the proposed method to reproduce the experiments.

2.3 B3 Metric

B3 is a fuzzy probabilistic metric that measures the precision and recall between clustering labels and true labels. Let's denote features matrix with X such that each row is a feature vector that corresponds to a point (sample). Then, we can show the membership function of the true label with $\mu_Y(X)$ where the element in row i and column j is membership of point i belonging to true class label j. Similarly, the membership function of the clustering label can be shown by $\mu_K(X)$. Let's represent element-wise multiplication by $\odot$, element-wise multiplication division by $\oslash$, and a vector with all elements equal to one by $\mathbb{1}$. We can

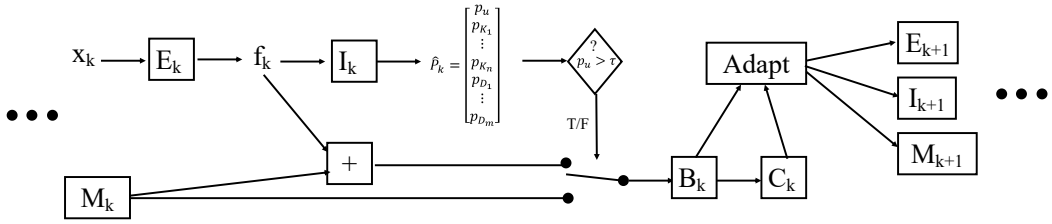

Figure 2: Function blocks diagram of open-world learning. At time step k, agent $\mathcal{A}_k$ can be modeled with a memory M_k, a perception subsystem or feature extractor E_k, and a decision making or inference subsystem I_k. The agent acts on open-world stream $\mathcal{S}^O$, see Eq. (11). At time step k, feature extractor E_k converts data $x_k \in \mathcal{S}^O$ to feature f_k. Then, inference subsystem I_k predicts probabilities of data belonging to unknowns, knowns and discovered classes, $P_k = [p_u \quad p_{K_1} \quad p_{K_2} \quad \dots \quad p_{K_{n-1}} \quad p_{K_n} \quad p_{D_1} \quad p_{D_2} \quad \dots \quad p_{D_{m_k-1}} \quad p_{D_{m_k}}]^T$, where n is the number of known classes in the training set and m_k is the number of discovered classes. If probability of unknown p_u is less than a threshold τ, then, the buffer B_k is equal to the memory M_k, otherwise, the buffer B_k is equal to the concatenation of the feature f_k and the memory M_k. Next, each instance of the buffer B_k, gets a label at function C_K either supervised (human or other agents) or unsupervised via clustering. Finally, the agent $\mathcal{A}_k$ will be updated to $\mathcal{A}_{k+1}$ based on the buffer B_k and the supervised/unsupervised labels C_K. The agent $\mathcal{A}_{k+1}$ will be used in the next time step $k+1$.

compute B3 metrics by

$$A_{L\times C} = \mu_Y^\top \mu_K \qquad M_{L\times C} = A \odot A \tag{4}$$

$$T_{C\times 1} = \sum_L A \qquad S_{L\times 1} = \sum_C A \tag{5}$$

$$P_{C\times 1} = (\sum_L M) \oslash (T \odot T) \tag{6}$$

$$R_{L\times 1} = (\sum_C M) \oslash (S \odot S) \tag{7}$$

$$\text{Precision} = \frac{T^\top P}{T^\top \mathbb{1}} \qquad \text{Recall} = \frac{S^\top R}{S^\top \mathbb{1}} \tag{8}$$

$$F = \frac{2 \text{ Precision} \,.\, \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

We used the library from paper [3] to compute B3 scores.

3 Open-World Learning Formalizations

While [6] has a formal definition of open-world learning, it is insufficient to characterize open-world learning without labels, so we provide an expanded formalization and new metrics. Fig. 2 demonstrates a cycle of open-world learning. In open-world learning, agents start from an initial (potentially pre-trained) model. The agents confront a continuous stream of data that contains a mixture of known and unknown objects. The agent should (1) distinguish known from unknown, (2) distinguish classes of the unknown from each other, and (3) learn the recognized classes unknown without forgetting previously learned classes.

Definition 1 *Open-World Stream*
Let us define $\mathcal{K} = \cup_i K_i$ for known classes seen in training, as well as $\mathcal{U} = \cup_j U_j$, classes unseen in training. The world set is defined as $\mathcal{W} = \mathcal{K} \cup \mathcal{U}$. Let x_n be a sample drawn from $\mathcal{W}$ at time step n. The closed-set stream is a time series

$$\mathcal{S}^C = \{x_n \in \mathcal{K} \quad \forall n \in \mathbb{N}\} \tag{10}$$

The open-world stream is a time series

$$\mathcal{S}^O = \{x_n \in \mathcal{W} \quad \forall n \in \mathbb{N} \mid (\exists\, x_i \in \mathcal{K}) \wedge (\exists\, x_j \in \mathcal{U})\} \tag{11}$$

Definition 2 *Open-World Learner*
Let the classifier of the agent $\mathcal{A}$ at the time step be $f_n : \mathcal{W} \mapsto \mathbb{R}^{k_n+u_n}$, which maps an input $x_n \in \mathcal{W}$ to a vector of probabilities of x_n belonging to one of the currently known k_n classes $K_1 \dots K_{k_n}$ or one of the hypothesized u_n unknown classes $U_1 \dots U_{u_n}$ where we allow the number of known classes to expand as new labels are provided, and the number of hypothesized unknown classess to expand as new data is processed and determined to form a new class. We further break down the agent into its d dimensional feature representation extractor ($R(x_n) : \mathcal{W} \mapsto \mathbb{R}^d$), and its classification engine $C(x) : \mathbb{R}^d \mapsto \mathbb{R}^{k_n+u_n}$. The agent $\mathcal{A}$ is an open-world learner if it acts on open-world streams and discovers and learns new classes $U_j \in \mathcal{U}$ in the stream after confronted with sufficient but bounded inputs drawn from each class. Learning a new class means predicting with a probability equal or grater than 0.5 for already seen instances in the class.

If supervised labels are provided to an open-world learner for instances in its unknown class U_j then we have a new known class $k_{n+1} = k_n + 1$ and $K_{k_n+1} = U_j$. This supervised model of open-world learning, converting unknown classes into known classes, is what is considered in prior work such as [6, 35].

However, we note that an agent may continue to function with many identified unknown classes that have only unsupervised pseudo-labels. Such a system may continue to improve its representation of that class even without labels as well as distinguish it from new unknown classes. This leads to the new definition for *open-world learning without labels*:

Definition 3 *Open-World Unsupervised Learner*
The agent $\mathcal{A}$ is an open-world unsupervised learner if it is an open-world learner, and it learns the new classes without using labeled data from humans or other agents.

Full open-world learning agents may update their feature representation subsystems $R(x)$ based on the increasing stream of data.

Definition 4 *Open-World Class Incremental Learner*
If the agent only updates the inference subsystem $C(x)$ and keeps the feature representation $R(x)$ during learning, we call it an open-world class incremental learner.

The latter two definitions can be combined, yielding open-world unsupervised class incremental learners, which is the focus of the remainder of the paper.

Metric for Open-World Learning
Because open-world learning mixes recognition of known and unknown classes, directly applying traditional metrics designed for either supervised or unsupervised learning does not necessarily work well.

Accuracy and balanced accuracy are the most popular metrics in supervised learning research. Unfortunately, accuracy cannot be defined when we do not have labels and hence cannot be applied to the unknowns. Even if we have ground truth labels for the data that goes into the unknowns used in testing since no label is provided, the unsupervised learning may split class or merge them, and hence we need unsupervised metrics, a.k.a clustering metrics. B3 (section 2.3) and Normalized Mutual Information (NMI) are two most widely used metrics in clustering research [2]. B3 and NMI are good metrics when the number of samples is large enough to represent the probability distribution of each class. In early versions of this work (see supplemental material), we were using just B3 or NMI on batches of data and eventually discovered that they were not well suited to open-world learning where we may have a large number of classes but only a small number of samples. None of them captures misclassifications of the unknown into an otherwise empty "known" class or the splitting of a known class into a mix of known plus unknown classes, e.g., breaking novel views into new classes. Therefore, we are proposing a new metric to overcome the issue of accuracy, B3, and NMI in open-world learning without labels. We call this the "Open-World Metric."

Definition 5 *Open-World Metric*
Let N be the number of items to be evaluated in data X. Let Acc be accuracy for known data and B3 *be the B3 metric (Eq. 9) for unknown data. Let us use subscripts ground truth and predicted categories of known and unknown such that known predicted as known is $_{KK}$, known data which was (incorrectly) predicted as unknown by classifier with $_{KU}$, unknown data that (incorrectly) predicted as known as $_{UK}$, and unknown data that predicted unknown by classifier with $_{UU}$. For correc known predictions, we can use accuracy and for correct unknown predictions, we can use* B3*, and we use incorrect predictions only in normalizing, then the Open-World Metric (OWM) score is computed by*

$$\text{OWM} = \frac{N_{UU}\ \text{Acc}(X_{KK}) \ + \ N_{UU}\ \text{B3}(X_{UU})}{N_{KK}\ + N_{KU}\ + N_{UK}\ + N_{UU}} \quad (12)$$

While we prefer B3, this measure can be generalized to combine other supervised or unsupervised metrics, e.g., $\text{OWM}_{\text{F1, NMI}}$ would use the above definition with macro-F1 instead of accuracy and NMI instead of B3.

4 Evaluation Framework

Prior evaluations of open-world learning in [6, 35], were fundamentally flawed because they used feature extractors that were trained on ImageNet 2012, but then they artificially defined subsets of the 1000 classes as the base of knowns and incrementally tried to detect other ImageNet 2012 classes as the unknowns. Thus, their feature space was trained using the "unknowns" as known and hence not a meaningful framework for proper open-world evaluation, even in a supervised setting. Therefore, we require a new evaluation framework, even for supervised open-world learning agents, and we do not reproduce data/tables from those prior works.

To evaluate and compare the performance of open-world learning algorithm in the task of image classification, (1) we use all 1000 classes of ImageNet 2012 train data set for training autonomous agents, (2) we use combinations of validation data set of ImageNet 2012 (known classes) and 166 classes of ImageNet 2010 train data set that do not overlap with ImageNet 2012 (unknown classes). We define four levels of tests: varying the number of instances per class and the number of unknown classes. Each test consists of 50 batches, where the batch size was 100 images. Regardless of the test level, each test has 100 classes of known, and each class of known has 25 images. So, each test has 2500 known images. Test U10 has 10 unknown classes, where each class has 250 images. Test U25 uses 25 unknown classes, where each class has 100 images. Test U50 uses 50 unknown classes, where each class has 50 images. And U100 tests have 100 unknown classes, where each class has 25 images. Known classes, unknown classes, and images in each class are selected randomly. All images, known and unknown, were distributed randomly across each test, and we run each test 5 times, and we report the average OWM. (standard deviation is shown in supplemental). The baseline for this evaluation is an EVM model which does not adapt during open-world learning but just classifies items as known or unknown, placing all unknowns into one group for evaluation.

Algorithm 1: Image classification with EVM

Input: single image (optionally a batch of images)
Output: probabilities of all classes and top-1 predicted label

x ← normalize image to range $[-1, +1]$
f ← CNN(x) `// Deep feature`
q ← EVM(X) `// Equation 3`
m ← max (q) `// Maximum probability`
u ← 1 - m `// Uncertainty`
v ← concatenate (u , q)
s ← $\sum$ v
p ← $\frac{v}{s}$ `// Estimated probabilities`
y ← argmax (p) `// Predicted label`
return p and y

To better understand the different aspects of the system, in evaluation, we consider three phases: closed-world where only data from known classes are present in the stream; Open-set, where unknowns are present but the system is not allowed to adapt; and open-world, where the system is allowed to learn from the data. The open-world stage has no access to the unknowns from the open-set stage. One should expect, and experimental data confirms, a drop in performance moving from closed-set to open-set, and then some level of recovery during the open-world stage.

5 Method

Our true open-world learning algorithm is summarized in Alg. 2 with three main elements: deep feature, enhanced EVM-based incremental-learning classifier for classification and detection of novel inputs, and clustering of detected novel inputs to form the basis of new classes.

The EVM has three important parameters: cover threshold, tail size (τ), and distance multiplier. We use 0.7 and 33998 for cover threshold and tail size, the same as the original EVM paper [35]. However, the original EVM formulation with its margin theorem concept using Eq. 2, is somewhat problematic for true open-worlds. The intuition behind the margin is that EVM is claiming half the space to the nearest other known class. That is fine for well-separated known classes, but it can easily be taking over too much open space for open-world learning as the assumption implies there are no classes in between the class being fitted and the nearest known classes. Because the original EVM experiments were tested on using pre-trained features that already separated all classes, this oversight may not have been apparent. Also, we find that margin is poorly defined in a highly imbalanced setting where a new class may have only a few samples. In such settings, we might need greater generalization from the few samples. Again this was not a problem in their experiments as they used balanced samples of well-separated classes, real open-world learning cannot presume these situations.

Algorithm 2: True Open-world Learner

Input: Single image and EVM model
Initialize: Empty clustered and residual sets
Config: $\delta = 0.001$, minimum number of images to start learning ψ, minimum number of cluster to start learning $\gamma = 2$, minimum cluster size to create a new class ρ, pre-trained features Ω
Output: new EVM

f, p ← run Algorithm 1
`// f: extracted feature`
`// p: class probabilities`
ϕ ← first element of p `// Unknown prob.`
if $\phi > \delta$ **then**
  Insert f in Residual
if *size(Residual)* $> \psi$ **then**
  L, M ← Clustering(Residual)
  `// L: cluster labels`
  `// M: Number of clusters`
  if $M > \gamma$ **then**
    foreach *cluster K* **do**
      if *size(K)* $> \rho$ **then**
        R^- ← Residual - K
        N ← concatenate (Clustered , Ω, R^-)
        Insert new class to EVM with K as + and N as -
        Insert K to Clustered
      Delete covered clusters from Residual
return EVM

To address these issues, our enhanced EVM includes the idea of a distance multiplier d_m, which replaces the multiplier of 0.5 in Eq. 2 with a free parameter. If $d_m < 0.5$, then the model is smaller (more specialized), leaving some room between it and the nearest other known class. If we choose a higher value for distance multiplier $d_m > 0.5$ during incremental class addition, we can expand the class generalizing more. We tested on a range of values of the distance multiplier using a held-out validation data, which is used for optimizing open-set classification accuracy. Among them, 0.45 demonstrates the best separation between known validation and unknown validation sets of ImageNet– slightly less generalization than the original EVM paper. With $d_m = 0.45$, we leave some room for classes between two known classes.

For features representations, we used EfficientNet-B3 [39] from Timm library [41] as feature extractor. The EfficientNet-B3 was trained on all 1000 classes of ImageNet 2012 train data set. Then, we extracted features from the last layer of the network before logit, which has 1536 dimensions. Then, an extreme value machine was trained on the extracted features (frozen features). Algorithm 1, shows

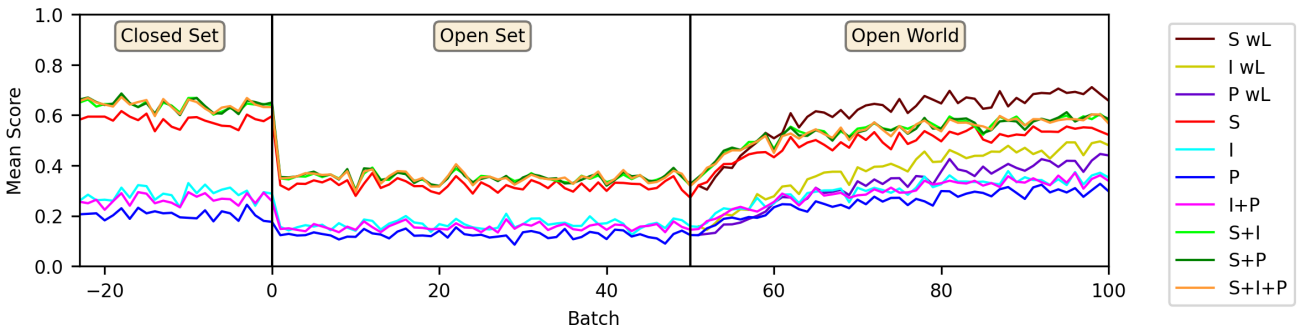

Figure 3: Average open-world metric scores over 5 runs of the for variations of TOWL algorithm when there are 100 unknown classes. All models start in the closed-set setting trained with supervised labels to build their base EVM. While supervised features updating with supervised labels (S wl) is eventually the best open-world learner, during the open-set testing, it is not and the various fusion of supervised + unsupervised features (S+I, S+P, S+I+P) are better. Interestingly, when doing open-world learning without labels using the fused features (S+I, S+P, S+I+P), all outperform using just supervised features (S). Pure unsupervised features (I, P, I+P) are consistently worse, even when they are provided labels during the open-world phase. See Table 1 for the full feature/label combinations in the legend.

the details of EVM-based image classifier for the proposed autonomous agent.

To detect a novel instance, we threshold on the enhanced EVM probability. The Weibull family distribution often converges to zero rather quickly, and EVM generates a very sharp boundary. Thus, we declare an image as novel (and hence nominate it to create a new class) if the probability of the class of unknown of EVM (the first class, i.e., the class with label zero) is above a very small threshold ($\delta = 0.001$).

To add class incrementally from a stream of images, we determine which images should be combined to create a new class. All prior work pursued supervised open-world learning, getting labels for each of the detected novel images, and used to update the model. To develop our true open-world learner (TOWL), we proposed to collect nominated novel images into a residual set, and when the size of the set becomes greater than a threshold, we group them together to create new classes. While one might consider classic clustering, such as K-means, we don't have any prior expectations on the number of new classes. Automatically discovering related groups of data in unsupervised data, without parameters, is an important and still unsolved problem. There are only a few published clustering methods that are appropriate. In this paper, we use the Finch algorithm [36] for clustering, which, while it is formally parameter-free, still does not provide fully automatic operation since it produces multiple potential partitions among which we must choose. We choose the smallest partition as we don't expect a lot of classes. Then, for each cluster with sufficient points for EVM fitting threshold, the agent creates a class and adds it to the current EVM. Algorithm 2 shows a summary of the proposed TOWL approach combining enhanced EVM and Finch.

6 Experimental Results and Discussion

We trained three EfficientNet-B3 networks: (1) supervised learning on ImageNet 2012, (2) unsupervised learning on ImageNet 2012 data set using MoCO V2, and (3) unsupervised learning on Places2 data set using MoCO V2. Then, we extracted features from a layer before Logit and froze them. Next, we trained an EVM for each frozen feature set. Then, we trained four EVM models with the concatenation of the different features sets. In testing, we consider both open-world supervised learners with labels and open-world unsupervised learners, see table 1 for the 10 combinations tests.

We repeated each test 5 times with different random selections. Fig. 3 shows the mean of open-world scores of 10 configurations when the number of unknown classes is 100. Fig. 4 in the supplementary illustrates the performances for different number of unknown classes. Table 2 states the open-world score of the last 1000 samples.

We first discuss open-world learning without labels from the point of the view of the open-world metric; the supplemental material contains tables and discusses simple K+1 class open-set accuracy, where not surprisingly, there is not much gain from trying learn new unseen classes when only being measured in with an open-set measure.

Figure 3 shows performance plots for the 10 systems when tested with 100 unknown classes; plots for 10, 25, and 50 unknown classes are in the supplemental material. Table 2, summarizes performance compared to the non-learning baseline for each of the 10, 25, 50 and 100 class experiments. From the plots, we see a significant drop from closed-set to open-set performance, but the drop is more dramatic for systems using at least some supervised features. Once open-world learning starts, they all improve, both with and without labels. The best performance, our

Table 1: Primary feature/label combinations used for experiments/plots

S	Supervised features
I	MoCo V2 on ImageNet 2012 features
P	MoCo V2 on Places 2 features
I+P	Concatenation of MoCo V2 on ImageNet 2012 and MoCo V2 on Places 2 features
S+I	Concatenation of supervised and MoCo V2 on ImageNet 2012s 2 features
S+P	Concatenation of supervised and MoCo V2 on Places 2 features
S+I+P	Concatenation of supervised, MoCo V2 on ImageNet 2012, and MoCo V2 on Places 2 features
wL	with label, i.e., supervised open-world learning

Table 2: Average on 5 tests, open-world scores of last 1000 images.

# Unknown classes	10		25		50		100	
Feature extractor	Base	Algorithm	Base	Algorithm	Base	Algorithm	Base	Algorithm
S	0.3638	0.3694	0.333	0.3599	0.3109	0.3561	0.3031	0.3716
I	0.1677	0.1878	0.158	0.1918	0.1429	0.2012	0.1495	0.2206
P	0.1304	0.1547	0.1205	0.166	0.1112	0.1643	0.109	0.1844
I+P	0.1635	0.1814	0.1567	0.1968	0.1415	0.1945	0.1404	0.2095
S+I	0.3966	0.422	0.3633	0.4211	0.3415	0.4106	0.3342	0.4177
S+P	0.3934	0.4192	0.3619	0.4173	0.3361	0.4056	0.3372	0.4239
S+I+P	0.3953	0.4188	0.3597	0.414	0.336	0.4055	0.3352	0.4174
SwL	0.3641	0.62	0.3338	0.5749	0.3103	0.5553	0.3047	0.5893
IwL	0.1681	0.3737	0.1582	0.3346	0.1437	0.3204	0.15	0.3744
PwL	0.1304	0.2988	0.1208	0.2806	0.1112	0.2708	0.1098	0.3336

computational upper bound, is given by using supervised features and supervised open-world learning. However, we see that when using fused features (S+I, S+P, or S+I+P), unsupervised open-world learning comes very close to the upper bound and is superior to using either just supervised features for unsupervised open-world learning or using just unsupervised features with supervised open-world learning (Iwl, Pwl). We see the learning rate (improvement rate) of supervised open-world learning can be higher than unsupervised learning. However, the improved learning rate is easier to have when starting at lower performance.

The proposed TOWL algorithm builds on extended EVM, and its model uses its extreme vectors during evaluation and during the update; thus, it does not face catastrophic forgetting. While not done in these experiments, if the number of new classes grows significantly, we expect that simple updating of EVM might not be sufficient, and the feature extractor also should be updated, which might raise issues of catastrophic forgetting, which are beyond the scope of this paper.

In this paper, we use EVM with a distance multiplayer of 0.45, and we select a threshold of 0.001 to be considered as a novel class to be updated. These parameters are fixed and not varied during testing. While these values are good for EfficientNet-B3 to be evaluated on ImageNet, for other data sets, they should be reevaluated on validation data.

Another parameter is how many detected novel class points are needed to begin clustering. Here, we choose threshold 50 to start clustering. If we choose a higher threshold, the quality of clusters will increase, and the speed of learning decreases. Therefore, there is a trade-off between the quality of learning and the speed of learning. The minimum value for these thresholds depends on the clustering algorithm and quality of the feature extractor.

We choose threshold 5 new points in a cluster to use it to instantiate a new class – so few but not one-shot unsupervised learning. If the number of required samples is larger, the new class is better defined and generalizes better; however, again, the learning speed decrease. Thus, there is a trade-off between the quality of learning and the speed of learning. Informally, we observed that a smaller class size threshold required a larger value of distance multiplier to be effective. If the class size threshold is rather large (very low speed), the distance multiplier should be equal to that for known (training) classes. Future works should investigate an optimal policy to adapt these thresholds based on data.

In this paper, we used the Finch clustering algorithm [36] to cluster instances that are predicted as unknown. The Finch generates several partitions. Producing too few clusters is dangerous as it will cause two classes to merge, and once merged, the current approach does not have the ability to separate; thus, the confusion is permanent. Over clustering will cause the new class of EVM not to generalize to the full semantic concept of the original class. Therefore, selecting a partition with the proper number of clusters is necessary. In our tests, we used the Finch partition with the

minimum number of clusters because the threshold 50 to clustering is small; future work should evaluate this choice and ideally develop a fully automatic algorithm.

7 Related Works

This is the first paper to tackle Open-World Learning Without Labels (OWLWL) problem. We deferred discussion of related work until here, so our new problem and solution approach was well defined, as a result, related work is given in context. We now discuss related works but again note that none of them has directly addressed the OWLWL problem.

Unsupervised incremental Learning

Unsupervised incremental learning has been used in many applications such as the prediction of musical audio signals [30], hand shape and pose estimation [22], Financial Fraud Detection [29], road traffic congestion detection [4], etc. In the following, we briefly describe the closest works.

In [34], authors used a mixture of Gaussian latent space, which uses dynamic expansion and mixture generative replay to minimize catastrophic forgetting in continual unsupervised learning. They did experiments on MNIST and Omniglot data sets. In [28], they designed a person re-identification algorithm based on pedestrian Spatial-temporal patterns in the target domain that consists of a feature extractor (CNN) and a matching model (Bayesian). Temporal patterns are not accessible in many image classifier agents. In [1], researchers proposed spike-timing-dependent plasticity for spiking neural networks to learn digits 0 to 9 incrementally. All of these three approaches had very limited experiments and may not work in more complex data such as ImageNet or Places2.

In excellent research [33], VGGface has used feature extraction in videos. Then, a modified version of the nearest neighbor to learn new faces (classes) incrementally. Also, they designed a feature forgetting strategy to control memory size in the long run. The results in [35] shows that EVM has better performance than the nearest neighbor. Thus, in this paper, we do not use the nearest neighbor classifier. In [27], they compared Support Vector Machines (SVM) with Extreme Learning Machines (ELM) in the task of incremental learning for face verification in video surveillance. They found that ELM is slightly better than SVM.

Continual Recognition Inspired by Babies (CRIB) [38] is an unsupervised incremental object learning environment that can produce data that models visual imagery produced by object exploration in early infancy. They reported that single exposure yields catastrophic forgetting. The algorithm's accuracy stays constant or decreases with a greater number of objects. Also, a smaller learning exposure length results in lower final accuracy. Their algorithm exploits 3D models, so it could not be used as a basis of comparison in this paper.

Open-world learning

Obviously, the most related work is open-world learning, which was first formalized [6] with subsequent work in [35, 26, 7, 37, 44, 17]. *In these prior works, supervised labels were provided to support their incremental learning in the open-world.* With the exception of [6, 35] these were done using textual-data, or small images and hence are not suitable to be used for comparison, for which we use the state-of-the-art for ImageNet scale problems: the Extreme Value Machine (EVM) [35].

8 Conclusions

In open-world learning, an autonomous agent learns continuously, discovering new classes in its non-stationary and never-ending stream of data. In an open-world environment, labels for data are often unavailable, and hence supervised open-world learning is not a scalable solution for online or real-time applications.

Here, we formalized the unsupervised open-world learning problem. Then we created a framework to evaluate autonomous agent performance in open-world scenarios and a new open-world metric suited for the evaluation of unsupervised open-world learning. Also, we extended a prior open-world image classifier using statistical extreme value theory to better handle open-world learning. Then, we designed a new true open-world learner (TOWL) and autonomous agents that discover, characterize, and learn new classes without labels from an open-world stream of data. TOWL combines state-of-the-art clustering (Finch) with an extension of the extreme value machine to provide the first solution to unsupervised open-world learning.

We also investigated the effect of feature representation on the robustness and learning speed of autonomous agents during both supervised and unsupervised open-world learning. The learning speed was fastest with supervised open-world learning. We found that combining supervised and unsupervised trained features with our TOWL produced the best results for unsupervised open-world learning, coming close to a fully supervised system using supervised features used with supervised learning. We conclude that having a feature representation that is not overly tuned to the known classes provides improved robustness in an open-set setting and improves learning in an unsupervised open-world learning.

References

[1] Jason M Allred and Kaushik Roy. Unsupervised incremental stdp learning using forced firing of dormant or idle neurons. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 2492–2499. IEEE, 2016. 8

[2] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486, 2009. 4

[3] Breck Baldwin, Tom Morton, Amit Bagga, Jason Baldridge, Raman Chandrasekar, Alexis Dimitriadis, Kieran Snyder, and Magdalena Wolska. Description of the upenn camp system as used for coreference. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998. 3

[4] Tharindu Bandaragoda, Daswin De Silva, Denis Kleyko, Evgeny Osipov, Urban Wiklund, and Damminda Alahakoon. Trajectory clustering of road traffic in urban environments using incremental machine learning in combination with hyperdimensional computing. In *2019 IEEE intelligent transportation systems conference (ITSC)*, pages 1664–1670. IEEE, 2019. 8

[5] Jan Beirlant, Yuri Goegebeur, Johan Segers, and Jozef L Teugels. *Statistics of extremes: theory and applications*. John Wiley & Sons, 2006. 2

[6] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *IEEE CVPR*, pages 1893–1902, 2015. 3, 4, 8

[7] Terrance E Boult, Steve Cruz, Akshay Raj Dhamija, M Gunther, James Henrydoss, and Walter J Scheirer. Learning and the unknown: Surveying steps toward open world recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9801–9807, 2019. 8

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[9] Enrique Castillo. *Extreme value theory in engineering*. Elsevier, 2012. 2

[10] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[11] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2

[12] Stuart Coles, Joanna Bawa, Lesley Trenner, and Pat Dorazio. *An introduction to statistical modeling of extreme values*, volume 208. Springer, 2001. 2

[13] Chuanxing Geng and Songcan Chen. Collective decision for open set recognition. *IEEE Transactions on Knowledge and Data Engineering*, 2020. 2

[14] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[15] Chuanxing Geng, Lue Tao, and Songcan Chen. Guided cnn for generalized zero-shot and open-set recognition using visual and semantic prototypes. *Pattern Recognition*, 102:107263, 2020. 2

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[17] Xiaojie Guo, Amir Alipour-Fanid, Lingfei Wu, Hemant Purohit, Xiang Chen, Kai Zeng, and Liang Zhao. Multi-stage deep classifier cascades for open world recognition. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 179–188, 2019. 8

[18] Jiangpeng He, Runyu Mao, Zeman Shao, and Fengqing Zhu. Incremental learning in online scenario. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13926–13935, 2020. 2

[19] James Henrydoss, Steve Cruz, Ethan M Rudd, Manuel Gunther, and Terrance E Boult. Incremental open set intrusion recognition using extreme value machine. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1089–1093. IEEE, 2017. 2

[20] Simon Jenni, Hailin Jin, and Paolo Favaro. Steering self-supervised feature learning beyond local pixel statistics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6408–6417, 2020. 2

[21] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[22] Pratik Kalshetti and Parag Chaudhuri. Unsupervised incremental learning for hand shape and pose estimation. In *ACM SIGGRAPH 2019 Posters*, pages 1–2. ACM, 2019. 8

[23] Artúr István Károly, Róbert Fullér, and Péter Galambos. Unsupervised clustering for deep learning: A tutorial survey. *Acta Polytechnica Hungarica*, 15(8):29–53, 2018. 2

[24] Xialei Liu, Chenshen Wu, Mikel Menta, Luis Herranz, Bogdan Raducanu, Andrew D Bagdanov, Shangling Jui, and Joost van de Weijer. Generative feature replay for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 226–227, 2020. 2

[25] Yaoyao Liu, Yuting Su, An-An Liu, Bernt Schiele, and Qianru Sun. Mnemonics training: Multi-class incremental learning without forgetting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12245–12254, 2020. 2

[26] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 8

[27] Eric Lopez-Lopez, Carlos V Regueiro, Xosé M Pardo, Annalisa Franco, and Alessandra Lumini. Incremental learning techniques within a self-updating approach for face verification in video-surveillance. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 25–37. Springer, 2019. 8

[28] Jianming Lv, Weihang Chen, Qing Li, and Can Yang. Unsupervised cross-dataset person re-identification by transfer learning of spatial-temporal patterns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7948–7956, 2018. 8

[29] Tian Ma, Shiyou Qian, Jian Cao, Guangtao Xue, Jiadi Yu, Yanmin Zhu, and Minglu Li. An unsupervised incremental

virtual learning method for financial fraud detection. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–6. IEEE, 2019. 8

[30] Ricard Marxer and Hendrik Purwins. Unsupervised incremental online learning and prediction of musical audio signals. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5):863–874, 2016. 8

[31] Bhaskar Mukhoty, Ruchir Gupta, K Lakshmanan, and Mayank Kumar. A parameter-free affinity based clustering. *Applied Intelligence*, pages 1–14, 2020. 2

[32] Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2307–2316, 2019. 2

[33] Federico Pernici and Alberto Del Bimbo. Unsupervised incremental learning of deep descriptors from video streams. In *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 477–482. IEEE, 2017. 8

[34] Dushyant Rao, Francesco Visin, Andrei Rusu, Razvan Pascanu, Yee Whye Teh, and Raia Hadsell. Continual unsupervised representation learning. In *Advances in Neural Information Processing Systems*, pages 7647–7657, 2019. 8

[35] Ethan M Rudd, Lalit P Jain, Walter J Scheirer, and Terrance E Boult. The extreme value machine. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):762–768, 2017. 2, 3, 4, 5, 8

[36] Saquib Sarfraz, Vivek Sharma, and Rainer Stiefelhagen. Efficient parameter-free clustering using first neighbor relations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2019. 2, 6, 7

[37] Vikash Sehwag, Arjun Nitin Bhagoji, Liwei Song, Chawin Sitawarin, Daniel Cullina, Mung Chiang, and Prateek Mittal. Analyzing the robustness of open-world machine learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pages 105–116, 2019. 8

[38] Stefan Stojanov, Samarth Mishra, Ngoc Anh Thai, Nikhil Dhanda, Ahmad Humayun, Chen Yu, Linda B Smith, and James M Rehg. Incremental object learning from contiguous views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8777–8786, 2019. 8

[39] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114, 2019. 5

[40] Edoardo Vignotto and Sebastian Engelke. Extreme value theory for anomaly detection–the gpd classifier. *Extremes*, pages 1–20, 2020. 2

[41] Ross Wightman. *PyTorch image models*, 2020. 5

[42] Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020. 2

[43] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019. 2

[44] Hu Xu, Bing Liu, Lei Shu, and P Yu. Open-world learning and application to product classification. In *The World Wide Web Conference*, pages 3413–3419, 2019. 8

[45] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4016–4025, 2019. 2

[46] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020. 2