Open-world Machine Learning: Applications, Challenges, and Opportunities

JITENDRA PARMAR and SATYENDRA CHOUHAN, Malaviya National Institute of Technology Jaipur, India
VASKAR RAYCHOUDHURY, Miami University, USA
SANTOSH RATHORE, ABV-IIITM Gwalior, India

Traditional machine learning, mainly supervised learning, follows the assumptions of closed-world learning, i.e., for each testing class, a training class is available. However, such machine learning models fail to identify the classes, which were not available during training time. These classes can be referred to as *unseen classes*. Open-world Machine Learning (OWML) is a novel technique, which deals with unseen classes. Although OWML is around for a few years and many significant research works have been carried out in this domain, there is no comprehensive survey of the characteristics, applications, and impact of OWML on the major research areas. In this article, we aimed to capture the different dimensions of OWML with respect to other traditional machine learning models. We have thoroughly analyzed the existing literature and provided a novel taxonomy of OWML considering its two major application domains: Computer Vision and Natural Language Processing. We listed the available software packages and open datasets in OWML for future researchers. Finally, the article concludes with a set of research gaps, open challenges, and future directions.

CCS Concepts: • **Computing methodologies** → **Learning paradigms;**

Additional Key Words and Phrases: Open-world Machine Learning, continual machine learning, incremental learning, open-world image and text classification

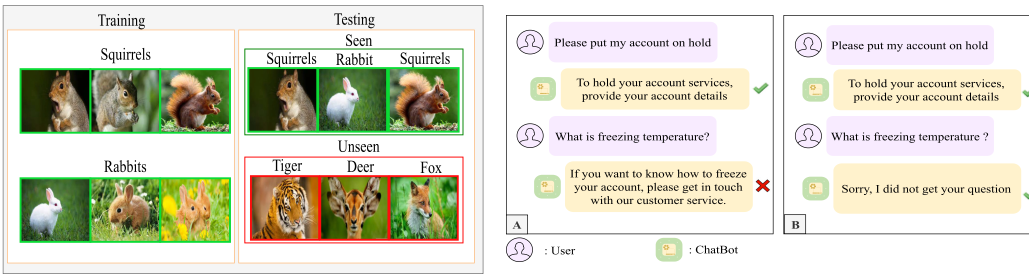
ACM Reference format:
Jitendra Parmar, Satyendra Chouhan, Vaskar Raychoudhury, and Santosh Rathore. 2023. Open-world Machine Learning: Applications, Challenges, and Opportunities. *ACM Comput. Surv.* 55, 10, Article 205 (February 2023), 37 pages.
https://doi.org/10.1145/3561381

1 INTRODUCTION

Traditional machine learning approaches have produced for decades promising outcomes for every domain of data analysis. However, traditional machine learning, mainly supervised learning, has some limitations [11, 13, 61], such as (1) it works with isolated data and learns without using

Authors' addresses: J. Parmar and S. Chouhan (corresponding author), Department of Computer Science and Engineering, Malaviya National Institute of Technology (MNIT) Jaipur, 302017, Rajasthan, India; emails: {2019rcp9044, sschouhan}@mnit.ac.in; V. Raychoudhury, Department of Computer Science and Software Engineering, Miami University, 510 E. High St., Oxford, Ohio, USA; email: raychov@miamioh.edu; S. Rathore, Department of Computer Science and Engineering, ABV-IIITM Gwalior, Gwalior, 474015 Madhya Pradesh, India; email: santoshs@iiitm.ac.in.

0360-0300/2023/02-ART205 $15.00
https://doi.org/10.1145/3561381



(a) Image Classification with Open-world Machine Learning

(b) Message Exchange between the User and *abc* Banking ChatBot

Fig. 1. (a) OWML in CVIP and (b) OWML in NLP.

previous knowledge and (2) a trained machine model can only work with the input instances for which similar instances have been used for the training purposes.

To properly identify classes that were not observed during the training phase (as known as the "unseen" classes) we require to use **Open-world Machine Learning (OWML)**. Let us consider the following example that illustrates the use of OWML in **Computer Vision and Image Processing (CVIP)** and **Natural Language Processing (NLP)** domains (Figure 1).

Assume that a traditional machine learning model is trained to identify images of squirrels and rabbits using a set of different images of squirrels and rabbits (in Figure 1(a)). During the testing phase, if we input to the model an image other than a squirrel or a rabbit, then the model will still classify the image either as a squirrel or as a rabbit. However, in the case of OWML, the images of tiger, deer, and fox will be rejected at the time of testing, as those images are previously not seen ("unseen") by the model, i.e., the OWML system can detect examples that are not from the training set. The capability to identify examples as unseen or to classify them is called open-world learning.

In Figure 1(b), a banking ChatBot is specifically designed for account and transaction-related queries. What will happen if the user asks an out-of-scope question? The system will reply with a false response, since it is trained in a closed-world environment (Scenario A). In the first scenario (A), the user asks to put his/her "account on hold." The query is correctly identified by the ChatBot and is responded appropriately. In the second inquiry user asks about the freezing temperature (of water), which is an out-of-scope query. The ChatBot fails to identify it correctly and suggests the procedure of "account freezing." The Scenario B presents an ideal ChatBot system using OWML. Both the queries are same as Scenario A, but the ChatBot correctly identifies that the second query is out-of-scope and refuses to answer. Classical machine learning [3, 8, 84, 92], especially supervised learning [61], follows the assumptions of closed-world learning [6, 29, 30, 72, 107], where for each testing class, a training class is available [84, 89]. However, in a real-world scenario, interactive and automated applications work in a dynamic environment, and data from the new classes arrive regularly. In such cases, the model that follows closed-world assumptions cannot address that kind of situation. OWML can enhance several recent AI-based prototypes in CVIP such as self-driving cars [15, 150], healthcare and medical diagnosis [17, 108], video surveillance [49], robotics [129], recognition of disruptive images on social media [52, 58]. Similarly, in NLP, OWML can help to improve ChatBot systems [23, 77], intelligent assistants [20, 53], email spam detection [137], product recommendation [145, 157], and cyberbullying identification [102].

Being a relatively new domain in machine learning there are very few review articles that cover OWML. Most of the existing review articles on OWML are task or domain specific. In Reference [124], authors reviewed numerous methodologies that can find novel attacks or malware in the open world. In References [71, 152], authors reviewed numerous methodologies, including deep learning-based approaches, which were used to identify human beings (person) in an open-world environment. One of the recent survey articles on OWML [34] has reviewed research works in CVIP. However, in recent years, OWML has also been used in NLP, such as automated dialog-based systems. To the best of our knowledge, there is a lack of review articles available for OWML, which can provide a broader classification of open-world learning.

This article presents a systematic review of related works in open-world learning. First, we present an overview of OWML with importance to the real-world context. It also presents a taxonomic classification of numerous OWML methods used in CVIP and NLP domains. In addition, we have presented the tabular summaries of existing works emphasizing the advantages and disadvantages of OWML approaches. Moreover, we discussed some of the baseline benchmark algorithms used in OWML for both CVIP and NLP. Our in-depth survey will be helpful in the selection of appropriate methods for a particular problem in a given learning environment. In summary, the contributions of the article are as follows:

- We present a task-based taxonomy that distinguishes OWML key features and highlights their relationships.
- We analyze several techniques and their features in terms of efficiency and other parameters.
- We also discuss various datasets, their characteristics and uses in OWML for CVIP, and NLP to thoroughly understand the outcomes.
- We point out various open challenges and research gaps to motivate future researchers and to help extend the existing OWML state of the art.
- Finally, we present some of the research areas related to OWML.

The organization of this article is as follows. Section 2 presents the background information and formal definition of OWML. Section 3 explains the review methodology adopted for this article and the taxonomy of OWML. Section 4 addresses related works of OWML in Computer Vision & Image Processing (CVIP). Section 5 addresses related works of OWML in NLP. Section 6 discusses some of the baseline algorithms used in OWML. The following two sections explain some of the findings and open research challenges in OWML (Sections 7 and 8). Finally, Section 9 concludes this article and discusses future research directions. In addition, a detailed discussion of the benchmark dataset and areas related to OWML is included in the supplementary materials and is made available through the GitHub repository.[1]

2 BACKGROUND AND FORMAL DEFINITION

Classical machine learning works in two parts: training and testing. For each example of testing, we must have a training example to identify such classes. Therefore, experts always suggest a high score for testing, but the high testing score cannot guarantee meaningful real-world outcomes. For good results in the real world, the machine needs to learn new things like humans. If the machine learns new things, especially those not present during training, and recognizes those things in testing, then the system will produce more convincing outputs. OWML can address the concerns of a dynamic environment where the input and nature of input data (size, category, frequency, etc.) are changing rapidly.

To better understand open-world learning, we have to know what *open* means. The systems are often designed for a specific task; the models are trained to identify particular objects if we consider computer vision examples. However, do similar objects come in the real world? In real-world objects are surrounded by many other things. In open-world learning, classifications are open, or models can learn incrementally. It can learn about new classes and update the existing model

[1] https://github.com/jitendraparmar94/OWML.

Table 1. Different Paradigms of Machine Learning

Domain	Techniques and Proposed Year	Task	Training Data	Testing Data	Knowledge Accumulation	Knowledge Retention
ML	Supervised Learning (1988)	CL and RG	Seen-Seen	Seen-Unseen	–	–
	Unsupervised Learning (1989)	CR and AS	Unseen	Seen-Unseen	–	–
	Reinforcement Learning (1995)	CL, CR and CNT	Seen-Seen /Unseen	Seen-Unseen	–	–
	Semi-Supervised Learning (2000)	CR and CL	Seen-Seen / Unseen	Seen-Unseen	–	–
DL	Deep Neural Networks (1965)	CL, CR and RL	Seen-Seen / unseen	Seen-Unseen/ Unseen	–	–
CML	Supervised Continual Learning (1995)	CL and RG	Seen-Seen	Seen-Unseen	√	√
	Reinforcement Continual Learning (1995)	CL, CR and CNT	Seen-Seen /Unseen	Seen-unseen	√	√
	Continual Learning in Deep Neural Networks (2002)	CL, CR and Rl	Seen-Seen / Unseen	Seen-Unseen / Unseen	√	√
	Unsupervised Continual Learning (2014)	CR and AS	Unseen	Seen-Unseen and Unseen	√	√
	Semi-Supervised Continual Learning (2015)	CR and CL	Seen-Seen and Unseen	Seen-Unseen	√	√
OWML	Open-world Machine Learning (2015)	CL, CR	Seen-Seen/Unseen	Seen-Unseen and **Unseen-Unseen**/Unseen	√	√

Abbreviations: ML: Machine learning, DL: Deep Learning, CML: Continual Machine Learning, OWL: Open-world Learning, CL: classification, RG: Regression, CR: Clustering, AS: Association, CNT: Control, RL: Representation learning.

(without re-training). OWML also refers as cumulative learning [3] and open-world recognition [8, 24]. Before comparing classical machine learning techniques with OWML, we have defined some terms here.

(1) *Seen-Seen Instances*: Instances that are labelled in the training datasets i.e., classes are known *a priori*.
(2) *Seen-Unseen Instances*: Instances that are unlabelled in testing datasets but belongs to the seen classes, i.e., classes are known during training time.
(3) *Unseen-Unseen Instances*: Instances that are unlabelled in datasets and have not been appeared during training time.
(4) *Unseen Instances*: unlabelled instance during training time.

Traditional **machine learning (ML)** has five major tasks: classification, regression, association, clustering, and control (robotics), which are done with various kinds of ML, which are shown in Table 1. Supervised machine learning proposed in the early 1988s uses seen-seen data for training and testing. In contrast, unsupervised machine learning, which is also proposed in the 1980s, uses unseen data for training and testing. Semi-supervised machine learning uses seen-unseen data for both training and testing. Reinforcement learning, recommended for classification and control, perceives and understands its context, takes actions and acquires knowledge by experiments and oversights, uses the seen data for training and seen-unseen data for testing. Deep learning is working on both classification and clustering uses seen data for both training and testing. The task of all categories of continual machine learning and training and testing data are similar to traditional machine learning, except traditional machine learning neither accumulates knowledge nor retains any previous knowledge in any future task. Table 1 given the broad classification of tradition and continuous machine learning based on task, required training and testing data, and knowledge accumulation and retention. OWML, which uses seen data for training and seen, seen-unseen, and unseen data for testing, is the only method that has a rejection capability for unseen instances.

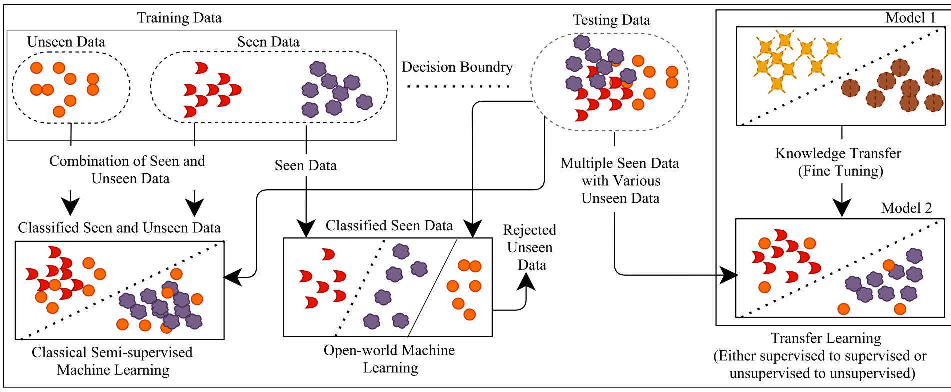


Fig. 2. Classical semi-supervised machine learning vs. open-world machine learning vs. transfer learning.

Sometimes it might appear that, OWML is associated with semi-supervised learning or transfer learning. However, these are different methods. Figure 2 shows the comparison between semi-supervised machine learning, open-world machine learning, and Transfer Learning. Semi-supervised machine learning involves small number of labelled data and possibly large number of unlabelled data. However, it still follows the closed world assumption that the unlabelled (unseen) data belongs to seen classes. It classifies instances according to classes available in training data. In contrast, OWML trained with seen data and classified seen data and rejected unseen data. Transfer learning uses knowledge transfer and fine-tuning to classify the new data (knowledge gained from one model can be used in different models to classify instances). Subsequently it works on new data and assume the closed world assumption during testing time. For rest of the article, we assume that seen classes means the classes that were appeared during training time, and unseen classes mean unseen-unseen classes.

OWML problem can formally be defined as follows.

Definition 1. Let D = $\{(x_1,\ y_1), (x_2,\ y_2), \ldots, (x_i,\ y_i), \ldots, (x_n,\ y_n)\}$, where n is the total number of instances, is the labeled training data for m seen classes. Here x_i is the ith instance and $y_i \in \{s_1, \ldots, s_m\} = S$ is x_i's class label. The objective of the classifier is to classify each test example x to one of the m seen classes or identify it as unseen class.

The process of learning in OWML can be defined in following three steps.

- Step 1: At specific time t, classification model builds by a learner that is multi-class classifier M_t based on all previous classes t of data with class labels $S^t = (s_1, s_2, \ldots, s_t)$. M_t is capable enough either classify seen classes $s_i \in S^t$ or reject them as unseen classes and put them in a rejection set R_e. The R_e may have instances of more then one new or unknown classes.
- Step 2: Now, the system can identify the hidden classes c in R_e and prepare training sets from this data to find unknown classes.
- Step 3: The model will learn from updated training dataset (previous data + new identified dataset). The model M_t is update to a new model M_{t+c}.

Theoretical foundations [6]: Let $\mathbb{Z}^+$ be the classes labelled by positive integers; at time t, $\Lambda_t \in \mathbb{Z}^+$ is the set of labels for seen classes. Let zero label (0) be used temporarily to mark data as unseen. Therefore, $\mathbb{Z}$ includes both seen and unseen labels. Let $x \in \mathbb{R}^r$ is the features (r is dimension of x), and $f_y\ (x)$ is the recognition function; that is, if the $f_y\ (x) > 0$, then instances are marked as a seen class and if $f_y(x) \leq 0$, then instances are marked as an unseen class, where $y \in \mathbb{Z}$. The solution to recognize any instance in an open environment using OWML can be given as a tuple $[F, \Phi, \ell, \partial]$.

Let us assume $\Phi(x)$ is vector function of features x, ℓ is labeling function, ∂ incremental learning function.

Recognize Unseen Classes: Here $F(x)$: $\mathbb{R}^r \mapsto \mathbb{Z}$, is set of recognition function that uses a vector function $\Phi(x)$, for computing i per class recognition function $f_i(x)$.

Label Unseen Data: Here induce the class labels for the unseen instances, which are determined by the ℓ. The novel unseen data are denoted by η_t for time t. The labels can be determine by using labelling function $\ell(x)$. The $\ell(x) : \mathbb{R}^r \mapsto \mathbb{Z}^+$ applied on η_t, resulting labelled data $D_t = \{(y_j, x_j)\}$ where $y_j = \ell(x_j) \forall x_j \in \eta_t$. Now the labelling function determines μ new classes, then the set of seen classes becomes $\Lambda_{t+1} = \Lambda_t \cup \{i+1, \ldots i+\mu\}$.

Incremental learning: The new classes incrementally added in knowledge base using $\partial_t(\Phi; D_t) : (F)^i \mapsto (F)^{i+\mu}$, it is an incremental learning function, that learns scalably and add new recognition functions $f_{i+1}(x) \ldots f_{i+\mu}(x)$.

Various methods are proposed to determine $f_y(x)$, such as probability search, fuzzy logic, outliers-based methods, 1-vs.-rest, and similarity-based search to segregate the instances of seen and unseen classes; discussed in Sections 4 and 5.

3 REVIEW METHODOLOGY

The methodical review summarized in this article was done by succeeding conventional review processes that ease understanding of domains of OWML. The steps involved to write in this review article are the historical timeline, convoying the survey, describing the outcomes, discussing investigations and challenges, the reasoning of conclusions, and future direction.

3.1 Review Plan

Conveying a methodical study includes collecting initial analysis about conclusions. Typical methods of such surveys incorporate confirmation and contradiction of preceding claims, classification and examination of analysis gaps/challenges, and future direction for exiting research. There is a fundamental advantage of conveying a methodical review and beneficial for authors as it covers the information of the domain with data. The following steps are taken to complete this survey.

- Steps of Review Plan
  (1) Recognize the requirement for a methodical survey
  (2) Frame an investigation query.
  (3) Find tasks and methods around that investigation query.
- Steps of Review and Result Reporting
  (1) Explore the initial investigations
  (2) Study the initial investigations for significance and relevance of domains
  (3) Selection of methodologies of the initial investigations
  (4) Integrate and abstract the extracted studies from initial investigations
  (5) Describe and report results as it is with suitable datasets
  (6) Conclude the methodologies and investigation
  (7) Conduct analytical and tabular comparisons
  (8) Write the methodological survey

3.1.1 Investigation Queries. We have formed the following generic queries to pursue the results from the readers' perspective. These are the standard parameter and findings that are required to understand any domain of research. Further, we prepare the entire draft according to the review plan and investigation queries.

Table 2. E-Source of Information

E-Sources	Content Type	Total Article
https://www.acm.org	Journal and Conference	43
https://www.springer.com/in	Journal and Conference	89
https://www.ieee.org	Journal and Conference	179
https://www.elsevier.com/en-in	Journal and Conference	94
https://www.tandfonline.com/	Journal and Conference	17
https://www.jmlr.org	Journal and Conference	26
https://www.aaai.org	Conference	35
https://www.kdd.org	Conference	23

(1) What is the importance of learning in the open world?
(2) How has machine learning grown in the last decade?
(3) What are the classifications of OWML?
(4) How does it differ from traditional ML?
(5) Which domains are correlated with OWML and how OWML can help to improve these domains?
(6) What is the current status of research in OWML?
(7) What are the tasks of OWML?
(8) What are the methods available in OWML to handling open-world tasks?
(9) How many datasets are available to investigate or perform OWML research for explicit domains?
(10) What are the associated areas of OWML?
(11) What are the challenges in the field of OWML to learn in open environments for various domains?
(12) What are the future directions of research in OWML?

3.1.2 Sources of Information and Selection Criteria. There is a need for a comprehensive aspect to the boundless coverage for an immeasurable and helpful article. We have collected a piece of pertinent information and data before getting started with a comprehensive article. We have explored many articles and select profoundly associated articles only to include them in the review. To collect these data, we have used prominent electronic sources, which are listed in Table 2.

Supplementary Sources: Other than the mainstream sources of information, we have used many repositories and other e-resources. These sources are helping us to provide additional information, technical and scientific reports, and analytical data to understand the domain. Some of the sources are listed below.

(1) https://mitpress.mit.edu (Books and Article)
(2) https://citeseerx.ist.psu.edu (Article)
(3) https://www.semanticscholar.org (Article and Technical Reports)
(4) https://www.morganclaypool.com (Book)
(5) https://www.sciencedirect.com (Article and Technical Reports)
(6) https://www.connectedpapers.com/ (Article)
(7) https://scholar.google.co.in (Article and Technical Reports)

Article Search and Inclusion Criteria: In approximately all searches carried the keyword "open world" in its title or abstract, we keep it in our repository. The domain is relatively new, and most of the work has been done only in the last decade, so we have to access multiple sources to collect the information. We have detailed examined these articles and kept the relevant articles only, process shown in Figure 3. Other than the keyword, we have used the most recent articles and technical

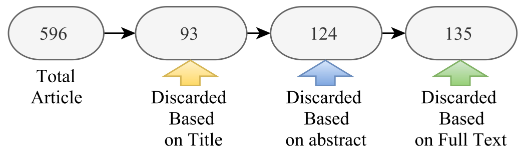


Fig. 3. Article discard process.

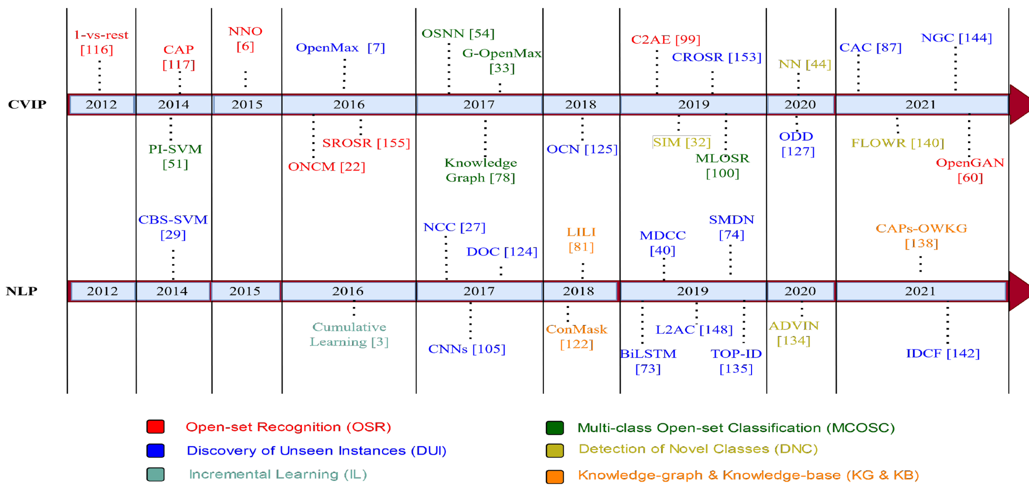


Fig. 4. Timeline of the research works carried out in OWML for both CVIP and NLP.

reports to follow the rooted trail and find many relevant articles. To maintain the high authenticity of any claim, we used only notable and reputed sources articles in this review.

After selecting articles by applying all the criteria on obtained articles from various sources, we have comprehensively studied the selected article on OWML. Based on the study, articles are categorized in two significant domains of OWML, that is, CVIP and NLP.

The timeline for OWML for both CVIP and NLP shows in Figure 4. The classified task shown in the figure is categorized based on a domain with proposed years. The most of researches were done between 2016 to 2021. In the timeline, we have included only essential methodologies that have enhanced existing work or introduced novel methodology for specified tasks of OWML. We can observe that there is less research available in NLP for OWML than CVIP. The timeline clearly shows that most of the research done in OWML is generally associated with discovering unseen instances. Some of the research focuses on the detection of novel classes for both CVIP and NLP. Table 3 lists all the acronyms and variables used in this article.

3.2 Taxonomy of Open-World Machine Learning

To ease the understanding of readers, we have graphically summarized the entire work done in OWML mentioned in this article (Figure 5). It involves cataloging of the domain, used or proposed methods, and dataset. There are two major fields where work has been done: CVIP and natural language processing. We have further categorized the work done in CVIP based on tasks, such as **Open-set Recognition (OSR)**, **Multi-class Open-set Classification (MCOSC)**, **Discovery of Unseen Instances (DUI)**, and **Detection of Novel Classes (DNC)**. In computer vision and image processing, numerous approaches have been used with various datasets to evaluate methods with different evaluation parameters.

Table 3. Acronyms and Variables Used

Acronyms Used			
Acronyms	Nomenclature	Acronyms	Nomenclature
OWML	Open-world Machine Learning	Open-GAN	Open-Generative Adversarial Networks
CVIP	Computer Vision and Image Processing	PI-SVM	Probability of Inclusion SVM
NLP	Natural Language Processing		
ML	Machine Learning	NA	Normalized Accuracy
OSR	Open-set Recognition	OSFM	Open-set F-measures
MCOSC	Multi-class Open-set Classification	CROSR	Classification-reconstruction Learning for Open-set Recognition
DUI	Discovery of Unseen Instances	OCN	Open Classification Network
DNC	Detection of Novel Classes	PCN	Pairwise Classification Network
NNO	Nearest Non-Outlier	ODD	Out-of-distribution Detectors
EVT	Extreme Value Theory	OS-Layer	Open-set Layer
C2AE	Conditioned Auto-encoder	CF	Collaborative Filtering
DNN	Deep Neural Network	NGC	Noisy Graph Cleaning
SVM	Support Vector Machines	MDCC	Multi-stage Deep Classifier Cascades
OSNN	Open-Set Nearest Neighbor	LOF	Local Outlier Factor
G-OpenMax	Generative OpenMax	MSP	Maximum Softmax Probability
CROS	Classification-Reconstruction Learning for Open-Set	DCNN	Deep Convolutional Neural Network
SIM	Similarity Metrics	OWR	Open-world Recognition
IL	Incremental Learning	TOP-ID	Towards Open Intent Discovery
KG&KB	Knowledge-graph & Knowledge-base	IDCF	Inductive Collaborative Filtering
CBS-SVM	Center-Based Similarity SVM	RMSE	Root Mean Square Error
NCC	Nearest Centroid Class	KGC	Knowledge Graph Completion
LSTM	Long Short-term Memory Networks	OKBC	Open-world Knowledge Base Completion
L2CA	Learning to Accept Classes	LILI	Lifelong Interactive Learning and Inference
SMDN	SoftMax, and Deep Novelty	Con-Mask	Content Masking
CAP	Compact Abating Probability	ADVIN	Automatic Discovery of Novel Intents
W-SVM	Weibull-calibrated SVM	BERT	Bidirectional Encoder Representations from Transformers
NCM	Nearest Class Mean	FTOP	Task-oriented Semantic Parsing
NBC	Nearest Ball Classifier	OSDN	Open-set Deep Networks
ONBC	Online NBC	K-NN	K-nearest Neighbors
ONNO	Online NNO	GET	Generalized Evidence Theory
ONCM	Online NCM	GBPA	Generalized Basic Probability Assignment
SRC	Sparse Representation Classification	EFCS-MU	Evolving Multi-user Fuzzy Classifier Systems
SROSR	Sparse Representation-based Open-set Recognition	DLFF	Defect List File Format
DOC	Deep Open Classification	TIFF	Tagged Image File format

Variables Used			
Notation	Meaning	Notation	Meaning
T_c	Training Class	r	relation set
T_s	Testing Class	t	tuple set
T_g	Target Class	M_X	Classification Model
$F(x)$	Function	D^{pr}	Training Data with Previous Classes
M_{new}	Novel Misclassified Instance	S_c	Set of Classes
(F_{new})	Existing Instance Misclassified as a Novel	m_i	Binary classifier
D	Training Data	D_{X+1}	New Dataset
$f(x)$	Classifier	P_a	Set of Images
(H_d, R_e, T_a)	Knowledge Graph	$\vec{f}$	Feature Vector
H_d	Head of Graph	$\mathbb{R}$	Set of Real Numbers
R_e	Relation Between Heads	c_a	Centroid
T_a	Tail Entity of Graph	n	Number of Instances
G	Incomplete Graph	k_i	Threshold
e	Entity Set	r	Dimension

In computer vision and image processing, achieve the various task using 1-vs.-rest, **Nearest Non-Outlier (NNO)**, **Extreme Value Theory (EVT)**, **Conditioned Auto-encoder (C2AE)**, and **Deep Neural Network (DNN)**, and these methods are evaluated with various datasets. The baseline benchmark algorithms are improved or extended to integrate the existing changes, such as **Support Vector Machines (SVM)** used to minimize open-space risk. Other methods were also used for open-space risk minimization, such as NNO. The **Probability of Inclusion- Support Vector Machines (PI-SVM)**, **Open-Set Nearest Neighbor (OSNN)** model, and EVT are used in many frameworks for multiset reorganization. The unseen class identification OpenMax,

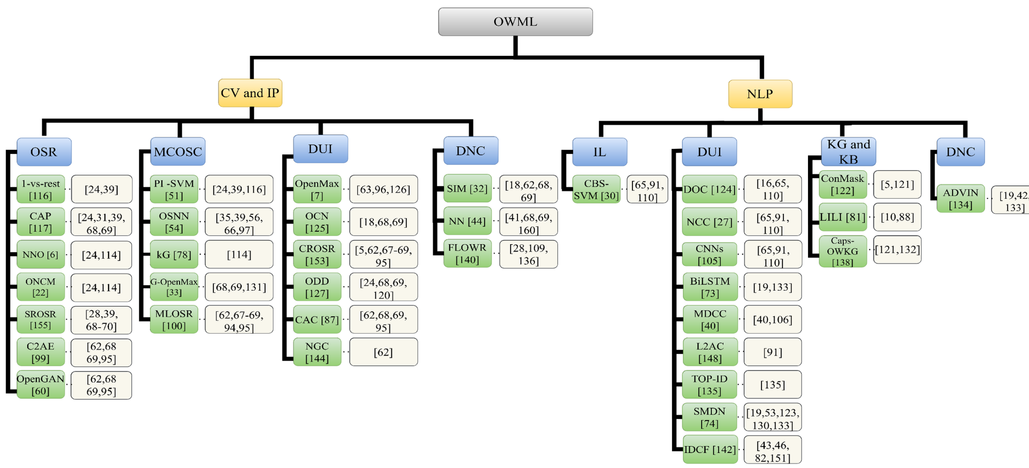


Fig. 5. Taxonomy of open-world machine learning.

Generative OpenMax (G-OpenMax), **Convolutional Neural Network (CNN)**, Classification-Reconstruction Learning for Open-set Recognition, and DNN have been used. CNN and **Stream Classifier with Integral Similarity Metrics (SIM)** have also been used to discover new classes.

We also discussed the work done in natural language processing in open world; and further categorized it in **Incremental Learning (IL)**, DUI, **Knowledge-graph & Knowledge-base (KG & KB)**, and DNC.

Several researchers have used baseline algorithms such as **Center-Based Similarity Support Vector Machines (CBS-SVM)** in natural language processing to reduce the open-space risk and incrementally acquire knowledge. Several methodologies mentioned here are a significant part of the framework or are used standalone for unseen class discovery. The 1-vs.-rest, CBS-SVM, **Nearest Centroid Class (NCC)**, **Long Short-term Memory Networks (LSTM)**, **Learning to Accept Classes (L2CA)**, **SoftMax and Deep Novelty (SMDN)**, and **Automatic Discovery of Novel Intents (ADVIN)** were used and evaluated with different datasets. The **Lifelong Interactive Learning and Inference (LILI)** model is used for new class detection, and OpenMax-based models are used for text classification in the open world. Now we discuss the reviews of methodologies that were used for computer vision and image processing in detail (Section 4). Next, we discuss methodologies for natural language processing in OWML (Section 5).

4 OPEN-WORLD MACHINE LEARNING IN COMPUTER VISION AND IMAGE PROCESSING

In this segment, we discuss the literature works done in computer vision and image processing with open-world settings. The preliminary research is focused on open-set reorganization using various methods. The input images are unknown for the model, or we can say that input is novel and unseen; as the input images are not available in training data, the knowledge is incomplete for the model. The model needs to respond to these unseen (open) data. The task analysis of work done for the particular task is shown in Table 4. Further, these tasks are discussed in Sections 4.1–4.4.

4.1 Open-Set Recognition

In a real-world environment, several external circumstances restrict the identification and distribution of tasks as inputs change frequently. It is generally challenging to accumulate training

Table 4. Summarized Study of Task Performed by OWML in CVIP

Author	OSR	MCOSC	DUI	DNC
W. J. Scheirer et al. [116]	√	–	–	–
W. J. Scheirer et al. [117]	√	–	–	–
L. P. Jain et al. [51]	–	√	–	–
A. Bendale and T. Boult [6]	√	–	–	–
A. Bendale and T. Boult [7]	–	–	√	
R. De Rosa et al. [22]	√	–	–	–
H. Zhang and V. M. Patel [155]	√	–	–	–
P. R. M. Junior et al. [54]	–	√	–	–
S. Demyanov et al. [33]	–	√	–	–
V. Lonij et al. [78]	–	√	–	–
L. Shu et al. [125]	–	–	√	–
P. Oza and V. M. Patel [99]	√	–	–	–
R. Yoshihashi et al. [153]	–	–	√	–
P. Oza and V. M. Patel [100]	–	√	–	–
Y. Gao et al. [32]	–	–	–	√
M. Hassen and P.K. Chan [44]	–	–	–	√
L. Song et al. [127]	–	–	√	–
D. Miller et al. [87]	–	–	√	–
J. Willes et al. [140]	–	–	–	√
S. Kong and D. Ramanan [60]	√	–	–	–
Z.-F. Wu et al. [144]	–	–	√	–

examples to employ all levels when training a classifier. A more practical situation is open-set recognition occurs wherever inadequate system information exists during the training, and unseen classes can be provided to a system during testing. In such a situation, expect the classifiers to correctly label the seen classes and effectively deal with unseen classes. Some research approaches use this obstacle and recognize open sets.

In Reference [116], the authors proposed an algorithm that can accept input with incomplete knowledge. The existing algorithm cannot handle open-sets. Thus they improve algorithms with the normalization of an algorithm to handle open-sets. They introduced 1-vs.-set and performed experiments on Caltech-256 [39] and ImageNet [24] sets. They perform experiments on labeled face data and compared it with their work. Also, perform experiments on different image domains and compare results with binary SVM linear kernel, binary 1-vs.-set machine linear kernel, 1-class SVM linear kernel, and 1-class 1-vs.-set linear kernel. Based on performance evolution, F-measure and accuracy clearly state that the binary 1-vs.-set machine linear kernel performed better than the other algorithms. Object and face recognition are considered in their work to verify the experimental results. Many researchers use multi-class classification to handle open-set problems, but multi-class approaches need labels for each input class. Therefore, the entire dataset required very laborious labeling and is still not an acceptable solution to handle open-sets. In this work, openness formalizes as

$$Openness = \sqrt{\frac{2 * |T_c|}{|T_s| + T_g}}, \tag{1}$$

where T_c = Training Class, T_s = Testing Class, and T_g = Target Class. It yields openness in percentage between 0 to 100. Where 0 = complete close class and 100 denote maximum openness, they conduct experiments on SVM with half-space and classify data not available in the training set. The SVM found separate classes as negative and positive, and the negative is only for known objects. The rest of the unknown objects are left as large unclassified open-space, which could be a part of a positive set. Then they felt that this could be remedied by reducing the open-space. Instead of generalization and specialization to minimize the errors in training function, they introduce "set". Set is known class training data of 1-vs.-set. It is used for open-space risk models and error minimization.

In Reference [117], the authors proposed a 1-vs.-rest machine. The 1-vs.-set algorithm handles the hazard of the unknown classes by dealing with two plane optimization, likely to result as a linear classifier. They extended open-risk classification to include non-linear classification in multi-class settings. They suggested a new model **Compact Abating Probability (CAP)** based on **Weibull-calibrated SVM (W-SVM)**; it decreases the probability value of member class when points move toward open-space from known data. The CAP evaluated on publically available benchmark datasets Letter [31], MNIST [68, 69], Caltech-256 [39], and ImageNet [24]. The results show that the CAP can reduce the open-space risk for known data. In Reference [116], it is established that by optimizing two planes, the 1-vs.-set machine can manage the risk and produce a linear classifier. 1-vs.-rest reduces the open-space risk by interchanging half-space, but the open-space risk is still infinite. It is pretty easy to find known classes for certain known classes, but when handling multi-sets. The algorithm is specially designed for unknown classes, and this algorithm deduced open-space risk from infinite to finite.

The 1-vs.-set machine assigns class labels to examples through the testing. It used a probability decision score for multi-class. It classifies examples using multiple classifiers based on the highest probability or probability that goes beyond the threshold. Examples that are below the threshold are rejected as unknown. This work has formalized a compact abating probability to address open-set regeneration by introducing a new algorithm, W-SVM, which integrates compact abating probability model and probability estimation theory. The experimental results clearly state that the f-measure (with various openness) of the W-SVM is relatively high for both open-set binary object recognition and multi-class open-set recognition. The W-SVM also performed decently on the OMNIST dataset for multi-class open-set recognition.

In Reference [6], authors address the issues associated with open-world recognition, such as open-space risk and practical tasks. They proposed a protocol to evaluate open-world recognition. They proposed NNO algorithm to manage open-space risk, model efficiency, and add object categories incrementally while detecting outliers. The proposed NNO algorithm experiment on more than 1.2 million images of ImageNet [24] dataset to validate the model. NNO is an extension of the **Nearest Class Mean (NCM)** algorithm [112]. They set an open-world evaluation protocol that uses seen classes in training. However, seen and unseen classes are used during the testing and continually add new class categories when encountering unseen classes. The training phase is further divided into two phases, the first is metric learning, and the second is the incremental learning phase.

In Reference [22], an author extends the work of open-world recognition [6]. They argue that to capture dynamic word recognition, incremental learning of underlying matrices, confidence threshold for unseen classes, and description space of classes are needed. They conduct experiments in three phases as follows: first, large-scale increment learning; second, open-world recognition; and, third, an online prediction of streamed images. The ImageNet [24] and ILSVRC'10 [114] dataset used in experiments that consist of 1.2 million, 50K, and 150K images for training, validation, and testing, respectively. The large-scale increment learning method creates relevant matrices and learned parameters on the initial set of 20 classes, and then classes are added incrementally in the set of 10 classes. It learns parameters and metrics on an initial 50 classes, and images of 50 classes are added after each iteration. To predict online images, researchers used current methods NCM classifier and **Nearest Ball Classifier (NBC)** and improved the accuracy. They predict images using ONCM and ONBC and compare results with existing NCM and NBC. Initially, the model predicts the labels for samples with the current model and then updates online accuracy based on predicted labels and genuine labels. After updating the online accuracy, updated the existing methods using accurate labels and sets. They also conduct the same experiment on Places-2 dataset [159]. The results clearly state that the ONNO, ONCM, and ONBC are performed better than existing algorithms.

In Reference [155], the authors proposed a framework that works on **sparse representation-based classification (SRC)**. SRC used class reconstruction error for classification. The most valuable information about open sets is available in the tail of the similar and non-similar parts of the class. To tail distribution reconstruction error SRC uses statistical EVT [21]. To evaluate the result they used benchmark datasets, such as extended Yale B [70], MNIST [68, 69], the UIUC Attribute dataset [28], and the Caltech-256 Dataset [39]. Evaluation results show that the simple, sparse representation classification does not effort up to the mark for open sets. Hence introduce different training modules to the trained system to handle open sets. In training, it acquired a random sample for each class and then partitioned it into two sets: cross-train and cross-test. These partitions are used for training and testing. Cross-train contains 80%, and cross-test contains 20% of training samples. To evaluate the result of **Sparse Representation-based Open-set Recognition (SROSR)**, compare the result with existing methods that use sparsity-based rejection such as w-SVM, Sparsity Concentration Index [141], ratio, and Naïve. The result clearly states that the SROSR is providing more accuracy and better F-Measure than existing methods.

The encoder-based model to detect unseen classes is further extended and uses an encoder and decoder to classify unknown data and open-set detection [125]. In Reference [99], the authors proposed a C2AE method that uses class conditioned auto-encoder to recognize open-set with novel training and testing methodologies. The proposed model operates in two parts, close-set classification and open-set identification. The encoder learned the first task for closed-set classification, and the decoder learned the next task for open class identifications. Training has been done using a closed-set model. The closed-set model consists of known classes. It trained the encoder and classifier and then conventionally calculated classification loss; after training the encoder for closed-set, it trained the open-set identification module, which consists of an auto-encoder network with weight and decoder for the reconstruction of the image according to label condition vectors. The proposed model was evaluated using the k-inference algorithm for an open set. To evaluate the performance of the model compared with W-SVM [116], SROR [155], and **Deep Open Classification (DOC)** [124] on the MNIST [68, 69], SVHN [95], and CIFAR10 [62] benchmark datasets.

In Reference [60], the authors proposed an OpenGAN to recognize the open sets by open data generation. The OpenGAN consists of the GAN-discriminator to classify testing examples. It is a binary classifier trained for both open-set and closed-set data. Many other techniques are available for closed-set classification, but each has limitations, whereas OpenGAN overcomes them by integrating them with various technical insights. In the first step, OpenGAN picked GAN Discriminator on a few actual outlier data already used in the existing research. The second step synthesizes "fake" data and adds it to the complete open training examples. The proposed method is evaluated with benchmark datasets (CIFAR, SVHN, and MNIST) and shows promising outcomes for open-set recognition through open data generation. The authors did not consider the mode collapse issue associated with GNN-based models. Hence, the performance of the model might decrease in specific circumstances.

4.2 Multi-Class Open-Set Classification

Multi-class classification in the open world is a very challenging task. If the unknown classes remain unaddressed, then the classifier either misclassifies or classifies false, known classes. Furthermore, there is also a possibility to classify false unknown classes. Misclassification and false classification can be avoided if multi-class classifiers can identify unknown classes appropriately.

In Reference [51], authors articulate the problem as one of sculpting positive training data at the decision boundary and invoke the arithmetical theory. It is used for assessing the non-normalized posterior likelihood of class insertion. They convert MNIST [68, 69] dataset from closed set to open-set recognition task and experiment with different sets of training and testing data. Now the following steps have been followed: (i) It used the standard supervised learning algorithm for MNIST [68, 69] classification with a 1-vs.-rest SVM on Platt [104] probability estimation in which classes are seen during the training, (ii) used only six classes from MNIST, (iii) used all 10 classes from MNIST and four unseen classes during the training, and (iv) change the testing regime to cross-class validation. In this scenario, similar classes are held out during the training (it is just a shuffling in (ii)) but comprise in a testing (shuffling in (iii)). They have performed two distinct open-set scenarios with cross-class validation: object detection by specific classifiers and multi-class open-set recognition, followed by detecting a problem and comparing it with PI-SVM. The model uses a universe of 88 classes to evaluate performance measures and binary decision elements for an open-set object decision [116]. The model performance evaluation was done on images from Caltech-256, and for testing, it takes the images from both Caltech-256 and ImageNet [24]. The entire evaluation was done over the fivefold cross dataset. The result clearly states that PI-SVM improves the F-Measure by 12% to 22% compared to existing methods.

In Reference [54], the authors proposed OSNN to address the issues in multi-class classifiers. It is an extension of the Nearest Neighbor for open-set [57]. OSNN used a similarity ratio instead of a similarity score and applied a threshold to find similarities between classes. They also designed a specific experimental protocol to evaluate open-set methods. Earlier proposed algorithms and frameworks are displayed as a virtuous outcome in the experiment, but these algorithms and systems are straggled to perform with open-set in real-world applications. Hence, to overcome these issues, they also proposed a system that measures adaptation in an existing open-set classification system, that is, Normalized Accuracy and Open-set F-measures, and evaluates the classifier's performance for both seen and unseen classes. The proposed model evaluated on 15-Scenes [66], Auslan [56], Caltech-256 [39], ALOI [35], and Ukbench [97] datasets.

Visual recognition systems are essential in identifying both seen and unseen classes of images. In Reference [78], authors proposed a knowledge graph-based approach to identify unknown visuals and recognize visuals in the open world. Three basic methods are used to predict classes; first, standard classification settings can predict only classes that are available in training, and images that are not available in training cannot be accurately predicted by standard classification. The second, zero-shot setting, can predict images not available in training, but some partial information is available for novel classes. The third, open-world setting, can predict images that are neither available in training nor partial information about its classes. The proposed method used the knowledge graph embedding model and image embedding model. The knowledge graph model uses properties, and the image embedding model uses images for training (ILSVRC-2012 [114]). Image embedding models used properties of the knowledge graph to predict open-world images.

In Reference [33] authors proposed G-OpenMax, which calculated the decision score of unseen classes instead of seen classes. It is an extension of OpenMax, which consists of a GANs network [37]. The proposed method used visualization for both seen and unseen classes; it also applied probability estimation to GANs and previous seen class dissemination to produce reasonable and domain-adapted synthetic unseen samples. The proposed model uses small and large scales datasets to evaluate the performance. A minimum of 10, and a maximum of 95 classes are utilized for the openness problem on two (HASYv2 dataset [131] and MNIST [68, 69]) handwritten datasets.

In Reference [100], authors proposed **Multi-task Learning-based Open-Set Recognition (MLOSR)**. It is based on a neural network for multitasking in open-set visual recognition. The proposed method combines a classification network, decoder network, and feature extractor network.

It utilized a decoder network to reject an open-set, and the decoder network reconstructs the error. It also uses EVT [21] for model tail error reconstruction from seen classes. EVT improves the overall performance of the model. The feature extractor network takes input and generates the latent, and the classifier uses this latent decoder to predict the class labels and reconstruct input images. The entire network has trained for both reconstructions of input images and classification. EVT modelled the trail of the reconstruction of the error distribution. The probability of reconstruction error by EVT and classification score used for open-set recognition testing. The proposed model uses COIL-100 [94], MNIST [68, 69], SVHN [95], CIFAR10 [62], and Tiny-ImageNet [67] datasets. The MLOSR is experimented with benchmark Visual Geometry Group network, dance Net with SoftMax [36], OpenMax [118], combination of ladder net, DHRNet with SoftMax, OpenMax, and **Classification-reconstruction Learning for Open-set Recognition (CROSR)**. The evaluation shows that MLOSR performed better than an existing network in recognizing open-set.

4.3 Discovery of Unseen Instances

Generally, systems choose images that might not be useful or significantly meaningless. In traditional classification methods, the system must classify the testing object in some classes. In comparison, an ideal system must reject the unseen classes that are meaningless and irrelevant. Some of the work presented shows how "fooling" [96] and "rubbish" [38] images appear in relevant classes as their confidence is high, whereas these are far from the class in which they appeared. Traditional deep networks have used fully connected feeds to the SoftMax layer as output [36]. SoftMax produces probability for the known labelled classes.

In Reference [7], the authors addressed this issue by introducing a methodology that can reject the unseen classes while testing. It is an adapted deep network for open-set identification. This methodology introduced OpenMax, which can evaluate the likelihood of an input being for an unseen class [118]. OpenMax rejects irrelevant images, decreases the error rate, and manages open-space risk. OpenMax estimates class by measuring a distance between the model vector aimed at the limited upper classes and the activation vector for an input. It provides the likelihood of unknown classes. Here OpenMax has an extended version of SoftMax that includes probability for unknown classes. This method used meta-recognition in deep networks and found scores to estimate how far testing an object to a known class. The activation layer has been used in deep networks to estimate the score. Meta-recognition and OpenMax can differentiate known and unknown classes and avoid foolish images to classify in known classes. The proposed model was evaluated on ImageNet, which is a subset of the ILSVRC-2012 dataset; since ILSVRC-2012 test labels are unavailable for use, experiments stated on validation set performance [63, 96, 126].

The combinations of networks are used to extend neural network-based unseen class discovery and add rejection capability. In Reference [125], authors proposed a framework to identify seen classes and reject unseen classes. It is not possible without having previous knowledge. The objective is to discover unseen classes for any given task and make a cluster by rejected examples (unknown instances). OWML is quite different from knowledge transfer. In the knowledge transfer process, the system transmits information between supervised to supervised and unsupervised to unsupervised systems. In this work, knowledge is shared from supervised to unsupervised. Here to find unknown instances, the authors proposed a model combining two networks, an **Open Classification Network (OCN)** and a **Pairwise Classification Network (PCN)**. Both networks will share the same components for learning. OCN is Build function $F(x)$ that can classify each seen and unseen class in S where PCN Build $g(x_p,\ x_q)$, a binary classification model. PCN will identify two test examples seen, unseen, from the same class or different classes, and hierarchical clustering used to discover hidden classes in all rejected examples. The proposed model uses MNIST [68, 69] and EMNIST [18] datasets to evaluate the performance.

All the methods discussed above are trained in a supervised manner and designed to classify known classes that are available during training. Therefore, it is challenging to determine unseen or unknown classes using these methods. In Reference [153], the authors proposed CROSR for robust unknown classes deprived of distressing the classification accuracy of known classes. CROSR trained networks for categorization and restoration of input data. While learning to distinguish unseen and seen classes, this technique helps improve implicit interpretation. CROSR method uses implicit structures for reconstruction to provide durable unseen recognition despite compromising the efficiency of seen-class classification. CROSR is based on OpenMax formulation. It reconstructs the input data to detect unseen classes that use exclusionary learning algorithms in seen classes to build classifiers. An open-set classification system based on DHRNets and CROSR combines seen classification with unseen detection. The proposed technique outperforms existing deep open-set classifier algorithms DOC [124], SoftMax [36], and OpenMax [118] for most permutations of seen data and anomalies, according to the trials conducted on five typical picture and text datasets MNIST [68, 69], CIFAR-10 [62], SVHN [95], tiny-ImageNet [67], and DBpedia [5].

The current research scenario focuses on finding new classes in rejected data, unseen or unknown. It will make the system more realistic and capable of working as a human being in a dynamic environment. In Reference [127], authors focused on the impact of out-of-distribution detectors and evaluated the performance of detectors. They experimented with six Out-of-distribution Detectors techniques, which are published at the best conferences in the world. They also experimented with detectors for corrupt images where the effect is unpredictable on the outcome; it may improve or decrease the performance. The out-of-distribution detectors ODIN, Network Agnostophobia, Mahalanobis Detector, Auto-encoder Detector, Deep-SVDD, and Outlier Exposure are evaluated with MNIST [68, 69], VOC12, ImageNet, and Internet Photos [120]. The Gaussian and uniform noise with the WRN-28-10 model uses a different combination of in-distribution. The performance evolution states that adversarial training can improve end-to-end strength. Adversarial training decreases discriminative influence and leads to poorer detection performance on benign out-of-distribution data.

In Reference [87], the authors proposed a simple DNN-based framework for open-set classification. DNN contains Open-set Layer and Closed-set Layer. It splits the data of intraclass. DNN splits data into subsets and produces an atypical sample. Atypical samples are used to model abnormal data, and standard samples are used for training. Intraclass info splitting exploits the interclass information. The closed-set regularization deep neural network apprehends an extraordinary close-set precision and is competent to discard unseen classes. The model uses the MNIST [68, 69], SVHN [95], and CIFAR10 [62] datasets to evaluate the performance and compare results with WSVM, GAN, **Collaborative Filtering (CF)**, and AE-ics.

In Reference [144], the authors present NGC, a novel graph-based noisy tag learning framework, which rectifies in-distribution noisy tags and filters out-of-distribution examples by leveraging the confidence of model predictions and geometric characteristics of the data when it comes to testing. NGC can identify and discard out-of-distribution samples without any additional training. NGC is evaluated on CIFAR-10 and CIFAR-100 publicly available benchmark datasets associated with real-world tasks. The experimental evaluation of NGC shows improvement over the existing methods.

The Fuzzy classifiers are typically used for soft labeling. The fuzzy classifier assign degrees to the class instances rather than the direct label. Fuzzy classifiers can be categorised into rule-based [4, 50, 93, 113] and prototype-based fuzzy classifiers [64, 149]. In Reference [76], the authors discussed a detailed analysis of dynamic fuzzy classification in ML. This analysis covers hypothetical basics, techniques and the learning prototypes, distinguishing dynamic fuzzy ML approaches and classification of these approaches, and open challenge of dynamic fuzzy ML.

In Reference [75], authors offered a methodology to discover unknown targets in an open world and determine the number of possible unknown targets employing the elbow method. The methodology utilizes the Generalized Evidence Theory to identify the unknown entities in the open world [25]. Initially, it generates a Generalized Basic Probability Assignment, a function that determines whether the framework is complete or not. If the discernment structure is incomplete, then the number of targets is re-specified by k-means, and then the accurate target number is specified by the elbow method. Finally, the frame of discernment is corrected. The proposed methodology uses Iris datasets to assess the performance. However, the results clearly state that the elbow technique does not accurately determine the number of targets in some circumstances. The method must address the sample imbalance data to identify the target accurately.

4.4 Detection of Novel Classes

The critical challenge is to discover instances of newly introduced data unknown to the system. Most research focuses on data with a low dimension dependent on coherence data and its property; therefore, detecting instances for newly known classes is hard to detect.

In Reference [32], the authors proposed a solution to this problem. The proposed framework SIM is a semi-supervised stream classifier that performs classification and detects novel classes on high-dimensional data streams. It uses latent feature space for classification, and an open-world classifier implements metric learning and stream classification and detects novel classes in unseen data. The proposed model is evaluated on both image and text data. The model calculated novel misclassified instance (M_{new}) to test the image, and existing instances are misclassified as novel (F_{new}). Apart from slandered performance majors, it experimented with FASHION-MNIST [146], MNIST, EMNIST, and CIFAR-10. Articles from the New York Times and Guardian are used with ten classes of other news as real-time text data to evaluate the model performance.

In Reference [80], authors proposed the fuzzy classification-based solution for the new class integration problem of online classification. They extended evolving fuzzy classifiers to update parameters incrementally for new class integration. They considered both single and all-paired techniques to decompose the information of class objects and establish a new class, which helps reduce the class imbalance. This decomposition improves the probability of proportional classes and decreases the probability that new classes are conquered. Also, it affects the decision boundaries by decreasing the complexity and pacing the knowledge learning process. The decomposition method allows more rapid integration of a new class without influencing the decision boundaries between the previously existing classes. Thus, classifiers become more durable on the previously existing classes and more flexible in blending and accurately producing new classes. Also, the single-pass active learning techniques are used to decrease annotation steps while integrating new classes. The proposed model was evaluated with two distinct and automatically recorded streaming sets of data stored by the existing system in the database. The database is in DLFF and TIFF with the feature vectors.

OWML has also extended its significance in security as we have new kinds of malware every period. We need a system to detect undefined classes to recognize that type of unseen class of malware. In Reference [44], authors proposed a method that can detect new unseen classes of malware. In this exemplification, samples from a similar class are closed to each other while those from different classes are further apart, leading to more significant space between known classes for unknown class samples. The proposed algorithm uses three datasets to evaluate the performance, MNIST [68, 69], MS challenge [41], and Android genom [160].

In Reference [140], the authors proposed open-world classification techniques that use embedding-based few-shot learning algorithms. It comprises small and big context few-shot open-world recognition formalization where decision-making machines must classify existing classes.

Few-shot learning for open-world recognition combines Bayesian non-parametric class priors with an embedding-based pre-training method. It also discovers unknown classes and quickly adapts and generalizes classes with the limited labelled data. It adapts benchmarks approaches such as few-shot training, open-set classification, and open-world identification to this environment. The authors present a Bayesian few-shot learning technique based on Gaussian embedding. The proposed system efficiently integrates new classes for both few-shot open-world recognition situations and Bayesian non-parametric classes. The evaluation results show that the proposed approach improves on a range of current methodologies by 12% in terms of H-measure. The model's performance evaluated on Mini ImageNet [136] and TieredImageNet [109] few-shot learning datasets (subset of ImageNet ILSVRC-12 [24]).

Table 5 summarises the literature on OWML in computer vision and image processing. It shows the used or recommended methodology, datasets employed for evaluation, and reported results by the authors [2].

4.5 Available Software Packages and Implementations

In this section, we provided a link for software packages containing various implementation models of OWML in computer vision and image processing (Table 6). These are the models commonly used in various frameworks of OWML.

Available software packages can be used to improve further learning in the open world for computer vision and image processing. The 1-vs.-rest is helping to improve the rejection of unknown classes. The NNO can normalize the open-space risk and open-set reorganization. C2AE is an encoder and decoder method for open-set reorganization and **Multi-stage Deep Classifier Cascades (MDCC)** for finding new classes. The PI-SVM and W-SVM can be used for multi-class classification in OWML.

4.6 Discussion

Many algorithms and frameworks are given significant outcomes for images in real-world settings. However, still, there is a need for a generic framework to deal with real-time inputs in a dynamic environment. Ideal outcomes can be achieved if models adopt generalization, specialization, and optimization of parameters. The algorithms must be able to handle inputs from multiple domains that may contain various classes, and these classes may have different kinds of objects in nature. The multiple input objects can be handled by including localization while optimizing the parameters. In the open-world applications working in the real world, the input rate is a significant issue because of the unpredicted input flow in terms of size and frequency. The open-space risk minimization is a vital challenge for every algorithm to achieve high accuracy while learning in the open world. The system must include prior knowledge to adapt continuity in learning that can reduce learning efforts in the future.

Image processing is one of the binding domains of computer science, and plenty of work has been done in this field, although there is scope to extend the research in OWML. The world is progressing toward automation in computer vision and image processing, such as driverless cars and humanless goods delivery systems. The real-time actions in a dynamic environment can be handled if the system is interactive and functions end-to-end to recognize the multiple objects in open space. The interactive models will help scale real-time data handling capacity with multi-class objects, which can be from different domains. Naturalistic results can be achieved if the system can deal with both empirical and open-space risks. Using past knowledge to recognize unseen objects in a dynamic environment will increase the system's accuracy and provide more realistic results. Thus the knowledge base must be updated incrementally. We observed the following challenges in OWML for CVIP tasks:

Table 5. Summarized Study of OWML in CVIP

Author(s)	Proposed/Used Methodology	Dataset	Reported Results
W. J. Scheirer et al. [116]	1-vs.-rest	Caltech-256 and ImageNet	F1-score 80%, Accuracy 98%
W. J. Scheirer et al. [117]	CAP	Letter, MNIST, Caltech-256, and ImageNet	F-measure 95 to 98% for 0 to 14% Openness
L. P. Jain et al. [51]	P_I-SVM	Letter, MNIST, Caltech-256, and ImageNet	F-measure 88 to 98% for 0 to 14% Openness
A. Bendale and T. Boult [6]	NNO	ImageNet and ILSVRC'10	Top-1 Accuracy 74% for more then 1000 categories
A. Bendale and T. Boult [7]	OpenMax	ImageNet (ILSVRC'10)	F-measure 0.59% for threshold values 0.20 to 0.45
R. De Rosa et al. [22]	ONCM, ONNO, and ONBC	ImageNet (ILSVRC'10)	Top-1 Accuracy 43% for known Train Classes Top-1 Accuracy 49% for Unknown Train Classes (50 Known Classes)
H. Zhang and V. M. Patel [155]	SROSR	MNIST, Extended Yale B,UIUC attribute, and Caltech-256	F1-measure 93 to 98% for 0 to 14% Openness Accuracy 92 to 99% for for 0 to 14% Openness
P. R. M. Junior et al. [54]	OSNN	15-Scenes, Letter, Auslan, Caltech-256, ALOI, and Ukbench	Normalized Accuracy 90% (Max. with Auslan) Micro open-set F-measure 80% (Max. with Letter) Closed Accuracy 90% (Max. with ALOI)
S. Demyanov et al. [33]	G-OpenMax	MNIST and HASYv2	F-measure 80 to 99% for 0 to 13% openness Accuracy 58% (Maximum with MNIST)
V. Lonij et al. [78]	knowledge-graph	ILSVRC-2012	Fracrion of Image 85% (With atleast 1 correct triple) Mean Rank 14%, and average number of true triples 19%
L. Shu et al. [125]	OCN CNN and 1-vs.-rest	MNIST and EMNIST	Mirco F1-score 91% (Max with MNIST) Accuracy 81% (Max with EMNIST)
P. Oza and V. M. Patel [99]	C2AE	MNIST, SVHN, CIFAR10, CIFAR+10, CIFAR+50, and TinyImageNet	F-measure 82 to 94% for 0 to 100% openness.
R. Yoshihashi et al. [153]	CROSR	MNIST, CIFAR-10, SVHN, TinyImageNet, and DBpedia	F-measure 41 to 79% for the threshold value o.1 to 0.9 (Maximum with MNIST) Micro F1-score 82.7% (Maximum with CIFAR-10)
P. Oza and V. M. Patel [100]	MLOSR	MNIST, SVHN, CIFAR10, CIFAR+10, CIFAR+50, COIL-100, and TinyImageNet	F-measure 82 to 90% for 0 to 49%
Y. Gao et al. [32]	SIM	Image Datasets: Fashion MNIST, MNIST, EMNIST CIFAR-10 Text Dataset: NEW YORK TIMES, GUARDIAN	Image Dataset: Accuracy = 96.94% Label Ratio = 100% Effectiveness = 96.94% Mnew = 61.3% Fnew = 47.1% Text Dataset: Accuracy = 57.95% Label Ratio = 96.0% Effectiveness = 57.95% M_{new} = 62.14% F_{new} = 59.0%
M. Hassen and P. K. Chan [44]	Neural-network	MNIST, MS Challenge, and Android Genom	AUC 95.88% for 100%FPR and 8.30% for 10% FPR (Maximum with MNIST)
D. Miller et al. [87]	Class Anchor Clustering (CAC)	MNIST, SVHN, CIFAR10, CIFAR+10/+50, and TinyImageNet	Area Under the ROC Curve (AUROC) 99.1% (Maximum with MNIST)
J. Willes et al. [140]	few-shot learning for open-world recognition (FLOWR).	Mini ImageNet and TieredImageNet (Both are subset of ILSVRC-12)	Accuracy 51.64% , Support-accuracy 57.76% and Incremental-Accuracy 39.39% (Maximum with Mini ImageNet) H-Measure 19.06% (Maximum with TieredImageNet)
S. Kong and D. Ramanan [60]	Open Generative adversarial networks (OpenGAN)	CIFAR, SVHN, MNIST, and Cityscapes	AUC 98.0% (Maximum with CIFAR) and F1-score 58.7% (Maximum with Cityscapes)
Z.-F. Wu et al. [144]	Noisy Graph Cleaning (NGC)	CIFAR-100, TinyImageNet, and Places-365	Accuracy 94.18% (Maximum with Places-365) AUROC 94.31% (Maximum with CIFAR)

Table 6. Available Software Packages and Implementations

Author	Model	Link
W. J. Scheirer et al. [116]	1-vs.-set	https://github.com/Vastlab/liblinear.git
A. Bendale and T. Boult [6]	NNO	http://vast.uccs.edu/OpenWorld
P. Oza and V. M. Patel [99]	C2AE	https://github.com/dhruvramani/C2AE-Multilabel-Classification
R. Yoshihashi et al. [153]	CROSR	https://nae-lab.org/$\sim$rei/research/crosr/
C.-C. Chang et al. [14]	W-SVM, PI-SVM	https://github.com/ljain2/libsvm-openset

- Open-space and empirical risk parameters are not optimised. Therefore, many models cannot adapt generalisation or specialisation.
- Most of the recommended methods have used limited training sampling; hence, the real-world impacts can not be determined accurately.

Table 7. Summarized Study of Task Performed by OWML in Neutral Language Processing

Author(s)	IL	DUI	KB&KG	DNC
G. Fei and B. Liu [29]	–	√	–	–
L. Shu et al. [124]	–	√	–	–
S. Prakhy et al. [105]	–	√	–	–
X. Guo et al. [40]	–	√	–	–
T. Doan and J. Kalita [27]	–	√	–	–
B. Shi and T. Weninge [122]	–	–	√	–
S. Mazumde et al. [81]	–	–	√	–
T.-E. Lin and H. Xu [73]	–	√	–	–
H. Xu et al. [148]	–	√	–	–
N. Vedul et al. [135]	–	√	–	–
T.-E. Lin and H. Xu [74]	–	√	–	–
G. Fei et al. [30]	√	–	–	–
N. Vedula et al. [134]	–	–	–	√
Q. Wu et al. [142]	–	√	–	–
Y. Wang et al. [138]	–	–	√	–

- Most of the recommended methods have not been employed with localisation; hence, it is insufficient to address images with multiple objects.
- There is an absence of a mechanism to minimise open-space risk. The learning can be improved by employing a dictionary learning-based algorithm for open-set recognition.

5 OPEN-WORLD MACHINE LEARNING IN NLP

Over the years, there has been enormous content generated on the web in the form of text. In social media, billions of users create most of the text that can influence human beings and social sentiments in terms of thoughts, stories, expression, news, and daily life events. Social media is a crucial part of the current environment regarding social and political perspectives. It can positively or negatively influence billions of people worldwide by injecting synthetic views that can already be a part of any plan. Therefore, analysis of social media content is vital to guide the world in a positive direction. Some work has been done on text data to analyze the text differently. OWML can help us learn about the text in a dynamic environment. Text classifications and data analysis is the foremost imperative entity for any organization. Standard text classification includes sentiment analysis, spam filtering, movie genre reviews, and document classification. The classification of tasks and work done toward these tasks are shown in Table 7. Further, these tasks are discussed in Sections 5.1–5.4.

5.1 Incremental Learning

Incremental learning is a ML method concerns expanding artificially intelligent systems that can continue to learn new tasks. It uses novel inputs as well as retains previously accumulated knowledge. The training method occurs whenever a novel task(s) appears. According to the novel task(s) and old knowledge, the model keeps whatever is learned. The most notable distinction between incremental learning and conventional machine learning, it does not lose previous knowledge. However, the training samples resemble it over time.

In Reference [29], the authors proposed the **Center-based Similarity (CBS)** method for open-world text recognition. It is a space learning method that can reduce open-space risk. The CBS is based on SVM. Centre-based similarity space learning transforms each document space vector or feature vector, each feature in the centre of the positive class document, and the feature vector of the document. At the same time, traditional classification directly uses training examples for trained binary text classifiers. CBS can learn multiple documents feature vectors, separate for each document, and represent the centre for multiple positive documents. Similarity value can be computed using multiple document similarity functions. The CBS's performance was evaluated on two publicly available datasets, 20 Newsgroup [65, 110], and amazon customer reviews [91].

In Reference [30], the authors extend their work and give a better system that can practice incremental learning in which the system can learn cumulatively. Whenever the system learned about new classes or unseen classes, it became more knowledgeable, just like humans do. The proposed system has two specific abilities, continually detecting unknown classes and cumulatively adding the data of these new classes to the knowledge base without re-training the whole system. The proposed CBS-SVM was evaluated with two different datasets Amazon product reviews of 100 domains and 20-newsgroup [65, 110]. Classifying classes in the open world uses the same unseen class rejection method based on threshold probabilities. The system used a similarity method to learn unseen or new classes. It explored sets of similar classes and learned to separate new classes. It builds a binary classifier to learn a separate new class. After identifying new classes, updates the existing classifier to avoid confusion for the subsequent unseen classes. The proposed method significantly outperforms existing methods 1-vs.-rest-SVM, 1-vs.-set-linear, WSVM-linear, WSVM-RBF, PI-SVM-linear, PI-SVM-RBF, ExploratoryEM, and CBS-SVM with different openness.

5.2 Discovery of Unseen Instances

OWML has the significant importance of rejecting unseen classes; the prediction accuracy of the known class must be justifiable. In Reference [124], authors proposed DOC to identify new classes or tasks that may not belongs to any training class. The ideal classifier should classify both the document for which the training classes are available and the training classes are not available. This method is called open-world classification or open classification.

DOC builds a multi-class classifier with the 1-vs.-rest final sigmoid layer in place of OpenMax [118]. DOC uses the sigmoid function with Gaussian fitting to lighten the decision boundaries and reduces open-space risk. It uses a CNN with a 1-vs.-rest sigmoid layer for the classification. It Chooses CNN, because OpenMax uses CNN, and CNN performs well on the text. Doc has three layers for a different task. Layer 1 embeds words (word vectors pre-trained from Google News that is Word2Vec) [85, 86] in x document into a dense vector. Layer 2 performs convolution on layer 1 with the different filters in various sizes. Layer 3 is a pooling layer that selects a maximum value from the result of layer 2 and forms a K-dimension. It converts the document into vectors using the word2Vec [85, 86] to extract the features. It used pre-trained from Google News vector [85] that consists of three million words and 300 dimensions word for word to vectors. The proposed model evaluated with 20-Newsgroups [110] and 50-class reviews [16]. The results are compared with CBS-SVM [29] and OpenMax.

In Reference [27], the authors proposed the NCC to detect unseen classes in open world. It is an incremental learning method that can take sets of closest neighbours of the centroid class. There are clusters for classes, and in a cluster, each class has minimum points. These are the membership points that are associated with clusters. Each class must have a minimum membership point to join the particular cluster. The class also represents the data point, and the centre of the class is the data points. New classes with the nearest class centre data point allow joining the cluster. The algorithm experimented on 20-Newsgroups and amazon review datasets with different numbers of domains to evaluate the performance. The prior algorithm performed better for some parameters on both datasets, but NCC's overall performance is significantly better.

ChatBots can work in a dynamic open-world environment, but it is vital to recognize the user's intention. Intent classification is a technique to distinguish the perseverance or intention by estimating the text language [101]. It refers to an intent classification or intent identification. Nowadays, many institutions use text-based chat systems to solve customer queries without human interactions. ChatBots must understand the unknown intentions of the user to work as a human being.

In Reference [105], the authors proposed another CNN-based approach that extracts the features using the word2Vec [85, 86] method. It used naïve methodology and calculated the cosine similarity among the mean of the entire document vector and calculated the document vector. Deep learning models are used for open text classification with a modified Weibull layer as the final layer instead of the traditional SoftMax layer [36]. It is single-layer architecture, but it has experimented with a different number of layers. It uses the 20-Newsgroups [110] and Amazon product reviews dataset to evaluate the performance of the proposed model.

In Reference [73], the authors proposed a two-stage approach for detecting unknown intent in the dialogue system. It uses a BiLSTM network with a margin loss to extract the feature of unknown intents. The LSTM network minimizes the variances of intra-class and maximizes the variances of inter-class intents. Glove word embedding is used to create vectors, and **Local Outlier Factor (LOF)** is used to distinguish the unknown intents [12]. The loss layer detects the known intents from deep discriminative features, and LOF detects unknown intents. The performance of the proposed approach evaluated on SNIPS [19] and ATIS [133] datasets. The observed results are compared with Maximum Softmax Probability [47], DOC [124], DOC SoftMax, and LOF SoftMax; the performance is quite better than existing methods.

In Reference [40], authors proposed a Deep Convolutional Neural Network, a cascade architecture that can continue learning newer classes. The framework is an end-to-end Open-world Recognition. They proposed MDCC to detect the instances from unknown classes. It contains unique features for known classes and can distinguish the class as a known class at any stage of the process. Incremented leaf nodes can detect features of unknown classes and recognize newly added classes. It can learn new features of recently added classes without wounding existing features of known classes. The proposed model MDCC uses the RF signal and Twitter datasets for performance evaluation. [40, 106]. The experimental outcomes are compared with Local Novel Detector [9], S-Forest [90], and R-OpenMax [89].

The e-commerce enterprise is expanding and has become a significant stakeholder in the world economy. Product classification is one of the most influential aspects of any e-commerce organization. The unpredicted or unknown inquiry about the product is essential for these industries as different categories of products emerge every day. The queries that are not predefined or known to the system can affect the reliability of the entire organization. In Reference [148], the authors proposed the OWML model, **Learning to Accept Classes (L2AC)**, based on meta-learning. L2AC maintains only dynamic known classes that allow novel classes to be added without retraining the model. In L2AC, each known class acts as a small set of training examples. The testing uses only meta-classifier (using known and novel classes). The L2AC model has two primary mechanisms, ranker and meta-classifier. The ranker retrieves examples from known classes that are comparable or nearest to test examples. The meta-classifier is the core mechanism of L2AC, and it is a binary classifier that determines the classes as known based on probability score or rejects otherwise. The performance of L2CA is evaluated on the Amazon dataset, and outcomes are compared with a different variant of DOC [124]. The results show the effectiveness of L2CA except for some parameters.

In Reference [135], the authors proposed a model **Towards Open Intent Discovery (TOP-ID)** for open intent detection. It is a two-phase mechanism that predicts the intent for the statement and then tags the intent in the input statement. The model consists of a BiLSTM [119] and **Conditional Random Field (CRF)** with the adversarial training method, increasing robustness and performance through the domain. TOP-ID can detect a user's intent automatically in natural language. It does not need any prior knowledge for intent detection. The first part of TOP-ID detects existing open intent and then tags it into input words with action and objective. If there is no objective and action associated with detected intent, then it is tagged as none. To perform this task initially, convert the text into feature sequence by assembling character-level representation, obtained using a CNN with Glove word embedding [103]. To avoid the combined word embedding effect on accuracy, TOP-ID used Highway Network [128]. The second module of TOP-ID is the intent discovery framework. It takes adversarial inputs (close to the original) created by adding noise in data in the form of perturbations. The overall training has been done with both original and adversarial inputs. The attention mechanism is part of the intent discovery framework. Multiple attention functions are used that attend to the information of the input sequences at different positions. Finally, the CRF predicts one of the three tags for the sequence of the words. The dataset is created to evaluate the TOP-ID by collecting 75K questions with accurate responses and then annotating 25K quotations data (with three tags: action, object, and none). The F1- score of TOP-ID is significantly better than existing methods.

In Reference [74], the authors proposed a SMDN detection model to detect unknown intents. The SMDN classifiers can be functional on any model without altering the architecture of the existing model. The model uses SoftMax, classifies by calculating the calibrated confidence score and detects unknown intent by calculating the decision boundary. The LOF is used as an output layer to detect the unknown intents [12]. Three benchmark datasets SNIPS [19], ATIS [133], and SwDA [55, 123, 130] are used to evaluated the performance of SMDN. The outcomes are compared with different variant of DOC [124].

In Reference [142], the authors propose an **Inductive Collaborative Filtering (IDCF)** system. It provides inductive learning for user inputs, ensuring sufficient expressiveness and adaptability. The IDCF uses two representation models to extract user-specific embedding as meta latent. It factorizes a set of essential users' data matrices, followed by an attention technique that learns concealed graphs among essential users and queries users based on their past ranking habits. For user queries, the inductive calculation of user-specific representations is enabled by the revealed associated graphs. IDCF standard version can decrease restoration loss to a similar level as the vanilla matrix factorization technique under a slight circumstance. Empirically, IDCF offers actual close Root Mean Square Error to transductive CF models. It uses a explicit: feedback data Movielens-100K, Movielens-1M [43], and Douban [151], and implicit: feedback data Amazon-Beauty, and AmazonBooks [46, 82] to evaluate the performance. IDCF achieves improved outcomes over the several inductive models, few-shot, and unseen user detection methods.

5.3 Knowledge-Graph and Knowledge-Base

KG is one of the crucial methodologies for the online and offline worlds. KG is helping in many elementary tasks such as web search, entity linking, language processing, recommendation, and prediction. This method is also worked under the closed-world assumptions as nodes are predefined. Infrequent research is available for OWML through the graph completion method. The relation and triples are vital components for knowledge graph completion methods [154]. In Reference [122], authors proposed a **Content Masking (ConMask)**, a Knowledge Graph Completion model in the open world. This model learns the embedding of any entity by its name and description in text fields and identifies unknown classes of entities to the knowledge graph. ConMask used relation-depending content masking to extract relevant chunks and reduce the noisy text description. After extracting relevant chunks, train the model with fully connected CNN to concur chunks with entities in a knowledge graph. Knowledge graphs can be represented as (H_d, R_e, T_a) where H_d is head, R_e is the relation between head H_d and some tail entity T_a. The generalised incomplete graph $G = (e,\ r,\ t)$, where e = entity set, r = relation set and t = tuple set. This graph can be completed by finding missing tuples t'. $t' = \{\langle H_d, R_e, T_a\rangle | H_d \in e, R_e \in r, T_a \in e, \langle H_d, R_e, T_a\rangle \in t\}$

in the incomplete KG. The model consists of three modules: relationship content masking, target fusion, and target entity resolution. (1) Module indicates the words relevant to the task. (2) Module extracts target entities embedding. (3) Module picked the target entities based on the similarity score between target entities. The last module of the model furthermore extracts entity embedding and textual features.

As the world is moving toward automation, ChatBot systems are becoming popular for customer query solutions. Every system that takes input as a text to elucidate or respond to a User's inquiry needs to understand the query. Existing systems are working with a limited environment, which means they can answer the only query for data available in the KBs. Such systems have limitations; they cannot work in a dynamic environment, because they cannot learn new knowledge. Many techniques have been proposed until now to complete missing information. These methods are termed KB completion, but all the methods worked under closed-world assumptions. KBs are limited and cannot work in an open-world environment.

To address this type of issue, in Reference [81], the authors proposed Open-world Knowledge Base Completion and LILI technique for ChatBots to acquire knowledge in the dialogue process. This knowledge learning engine allows ChatBots to acquire knowledge throughout the dialogue and make it further interactive. It is based on the theory of continual learning, where ChatBots become more knowledgeable with time as they learn continually after every conversation. Lifelong interactive learning and inference analyze the query and add it to KB if it does not exist. LILI, formulate an inference strategy, learn interaction behaviours, leverage the acquired knowledge, and continuously repeat this to learn new knowledge. Two benchmark datasets, Freebase FB15k [10] and WordNet [88] are used to evaluate the performance of LILI. The outcomes are compared for known, unknown, and overall classes. The observations show that the LILI is effective for predictive eminence and strategy formulation capability.

In Reference [138], the authors suggest a capsule network-based approach, Caps-OWKG, that leverages context to describe relationships and objects in the open-world knowledge graph. The proposed Caps-OWKG consists of triplets that are the system's basic unit. In addition, the capsule network conducts extraction of features, judgment on triplets, text synthesis, and fusion analysis. When computing triplets, the Caps-OWKG technique provides a stronger connection between items and relationships. These interpretations are also refined; thus, the Caps-OWKG model may be considered a dynamic embedding exploration that accurately represents the triplet. The existing known techniques such as ConMask [122], TransE-OWE [121], and DKRL [147] are used to compare the performance of Caps-OWKG on the two benchmark datasets FB15k-237 [132] and DBPedia50k [121], which achieve better outcomes than existing approaches.

5.4 Detection of Novel Classes

The existing research finds only new intent in the available domain, but the novel domain is introduced incrementally as data increases. A novel domain must be found to make the system fully automated and reduce the system's limitations. In Reference [134], authors proposed ADVIN to discover novel domains and intents of text from unlabeled data. ADVIN works in three stages: discovering the novel domains and intent from extensive unlabeled data, knowledge transfer, and linking related intents to corresponding novel domains. To identify the instances of novel intents ADVIN, used BERT-based multi-class classifiers [26]. The DOC [124] is used for distinguishing unseen intents. The second stage, which discovers the categories of newly discovered intents, uses a hierarchical clustering method to transfer knowledge. Finally, by linking novel intents into novel domains, ADVIN used clusters of seen classes as ideal clusters and knowledge transfer modules to represent clusters. To evaluate the performance of ADVIN, it experimented on four benchmark datasets, SNIPS [19], ATIS [133], **Facebook's Task-oriented Semantic Parsing (FTOP)** [42], dataset from a commercial voice assistant, and Internal NLU. The outcomes are compared with DOC, IntentCapsNet [154], LOF-LMCL [27], and different combinations of ADVIN and DOC. The overall performance of ADVIN is significantly better than existing methods.

Evolving classifiers are an excellent approach for handling the current needs of open-loop and incremental classifier building techniques, particularly evolving fuzzy classifiers. It is helpful for online data streaming. However, until now, the emphasis is on classifiers established on the feedback and input from target labels delivered by a single user (expert). In Reference [79], authors proposed three variants of **Evolving Multi-user Fuzzy Classifier Systems (EFCS-MU)**. The variants are ensembles of single-user classifiers, consensus all-user classifiers, and shift-work all-user classifiers. The first variant permits a specific classifier training per user and embeds a refined assembly process that ensembles on a model level. The second variant of the EFCS-MU comprises a combined classifier specified for all users established on consensus labelling that ensembles on a label level. The third variant of EFCS-MU comprises a combined classifier specified for all users established on the traditional classical shift-work notion. The proposed system uses the Nursery dataset to evaluate the performance [98]. However, the proposed method needs to integrate solutions for concerns that arise due to modifications in the local data distribution of classes and links between input elements and target notions to be retained in the evolving multi-users classifier system. It must address appropriate explicit handling of drifts and omitting techniques in the classifier updates.

Table 8 summarises the literature on OWML in natural language processing. It shows the used or recommended methodology, datasets employed for evaluation, and reported results by the authors [1].

5.5 Available Software Packages and Implementations

In this section, we provided a link for some software packages that contain various implementation models of OWML in natural language processing (Table 9). These are the models commonly used in various frameworks of OWML.

The available software packages can be used to improve further learning in the open world for natural language processing. The **Open-set Deep Networks (OSDN)** can be used for open-set reorganization, DOC for unseen class identification, and ConMask for identification of unseen entities in the knowledge graph. There are two packages, Word2Vec and GloVe, for input word embedding.

5.6 Discussion

Few works have been done in OWML for natural language processing. The semantic similarity in the text is challenging to manage while learning new knowledge about the text at runtime, especially when there is no training set available for such data. To achieve valuable outcomes from any framework or an algorithm, it must determine the semantic similarities in text. Therefore, a large-scale knowledge base is needed to learn the hierarchical structure of text words with meanings. Openness is a significant issue as we have analyzed that the accuracy reduces whenever the openness increases, particularly in CVIP. There is a need for frameworks that can deal with dynamic values of openness and provide high accuracy with maximum openness. The automated dialogue-based system, quite popular nowadays, needs a real-time mechanism to process the informal conversation.

The automated ChatBot systems and text and voice-based assistance devices are increasing rapidly in this decade, further improving the world of automation. To increase the accuracy of such a system, the open-space risk and distinguishing the semantic similarities is one of the significant aspects of neural language processing in open-world settings. The cumulative and incremental

Table 8. Summary of the Proposed Approaches for Natural Language Processing in OWML

Author	Proposed or Used Methodology	Datasets	Reported Results
G. Fei and B. Liu [29]	CBS-SVM	20-Newsgroups and Amazon reviews	Accuracy 45 to 87.3% for 25% to 100% openness (Maximum with Amazon reviews 10 Domains)
G. Fei et al. [30]	Cumulative Learning (using CBS-SVM)	Amazon product reviews and 20-Newsgroups	Micro F1-score 66.2 to 83.5% for openness of 33% to 100% (Maximum with 20-Newsgroups)
L. Shu et al. [124]	DOC	20-Newsgroups and Amazon reviews (50-class reviews)	Micro F1-score 82.3 to 92.6% for 25% to 100% openness (Maximum with 20-Newsgroups)
S. Prakhy et al. [105]	CNNs	20-Newsgroups and Amazon reviews	F1-score 79.7 to 82.1% for 25% to 100% openness (Maximum with Amazon reviews 10 Domains)
T. Doan and J. Kalita [27]	NCC	20-Newsgroups and Amazon reviews	Accuracy 20 to 82% for 0 to 50 domains (with Amazon reviews)
X. Guo et al. [40]	MDCC	RF signal Datasets,Twitter dataset,	EN-Accuracy 60.45% (Maximum with RF Signal) F1-score 75% (Maximum with RF Signal)
B. Shi and T. Weninge [122]	ConMask	DBPedia50k and DBPedia500k	Mean Rank 90 and Mean Reciprocal Rank 35.0 (for head) Mean Rank 16 and Mean Reciprocal Rank 61.0 (for trail) both are maximum with DBPedia50k
S. Mazumde et al. [81]	LILI	Freebase (FB15k1) and WordNet	Avg. F1-score 63.43% (Maximum with FB15k1) and Avg. MCC 39.39% (Maximum with WordNet)
T.-E. Lin and H. Xu [73]	Bidirectional long short-term memory (BiLSTM)	SNIPS and ATIS	F1-score 78.8 to 79.2% for 25& to 75% openness (Maximum with SNIPS)
H. Xu et al. [148]	L2AC	Amazon Datasets	Micro F1-score 84.68 to 93.19 % for 25% to 75% openness
N. Vedul et al. [135]	TOP-ID	25k real-life utterances (Created dataset)	F1-score 91% (Maximum among all the versions of TOP-ID)
T.-E. Lin and H. Xu [74]	SMDN	SNIPS, ATIS, and SwDA	Macro F1-score 71.1 to 79.8% for 25% to 75% openness (Maximum with SNIPS)
N. Vedula et al. [134]	ADVIN	SNIPS, ATIS, FTOP, and Internal NLU Dataset	F1-score 92% (for discovery of unseen instances) NMI 83% Purity 92% F1-score 78.0% (for discovery of unseen classes)
Q. Wu et al. [142]	IDCF model	Douban, Movielens-100K, Movielens-1M, Amazon-Books, and Amazon-Beauty	AUC 94.4% (Maximum with Amazon-Books) and Normalized discounted cumulative gain (NDGC) 95.5% (Maximum with Douban)
Y. Wang et al. [138]	Caps-OWKG	DBPedia50k and FB15k-237-OWE	Tail prediction 64.8% (Maximum with DBPedia50k)

Table 9. Available Software Packages and Implementation

Author	Model	Link
P. Moore and H. Van Pham [89]	OSDN	https://github.com/abhijitbendale/OSDN
L. Shu et al. [124]	DOC	https://github.com/leishu02/EMNLP2017_DOC
B. Shi and T. Weninger [122]	ConMask	https://github.com/bxshi/ConMask
T. Mikolov et al. [86]	Word2Vec	https://code.google.com/archive/p/word2vec
J. Pennington et al. [103]	Glove	https://nlp.stanford.edu/projects/glove/
X. Guo et al. [40]	MDCC	https://github.com/xguo7/MDCC-for-open-world-recognition
Y. Kim et al. [59]	CNN Text Classification	https://github.com/dennybritz/cnn-text-classification-tf
Y. Kim et al. [59]	CNN Sentence Classification	https://github.com/alexander-rakhlin/CNN-for-Sentence-Classification-in-Keras

model can help to address such issues. The system will deliver more natural outcomes when it adapts scalability in input with maximum openness as the real-world inputs are unstructured. In this section, we have discussed various problems associated with NLP in open-world learning and proposed solutions by various authors that can help improve a text-based application operating in a real-world domain and dynamic environment. The following challenges we observed in OWML for NLP tasks.

- Stable performance can be achieved by identifying unseen instances only if the threshold value is within a reasonable range.
- The recommended methods show superior outcomes for sample instances. The accuracy of many of the proposed systems decreases if the number of seen classes is low.

- There is a need for improvements to the use of these prototypes for real-time systems, as extensive experiments with large-scale datasets are missing.
- Only a few methods are employed with cumulative learning.
- In NLP, many recommended models suffer when distinguishing unseen intent from seen intents where semantic meanings are similar.

6 BASELINE ALGORITHMS USED IN OPEN-WORLD MACHINE LEARNING

Some methods and algorithms are used in OWML that are standard or base concepts to practice open-world learning.

6.1 Center-based Similarity [29]

CBS is a classification method that classifies the data points into seen and unseen classes. It works on a centre-based similarity space learning technique. The CBS learns news classes incrementally and uses a 1-vs.-rest layer to classify unseen classes [29]. The 1-vs.-rest is one of the vital concepts in OWML to discover unseen classes.

Let us assume there is new class l_{X+1}, for learning it need a model M_X. Model M_X consist set of X 1-vs.-rest binary classifiers $M_X = (m_1, m_2, \ldots m_X)$, for the previous X classes there is training dataset $D^{p_r} = (D_1, D_2, \ldots D_X)$ (p_r = previous) and corresponding labels are $S^X = (l_1, l_2, \ldots l_X)$. Here each M_i builds a binary classifier to identify l_i, when new dataset D_{X+1} arrives for class l_{X+1}. the entire system functions for two task, to update M_X and build new M_{X+1} model to classifies all available instances in existing class $S^{X+1} = (l_1, l_2, \ldots l_X, l_{X+1})$ and recognize the U_s unseen classes.

Step 1: Search a *set of classes* S_c that are comparable to new class l_{X+1},

Step 2: Learning to isolate the new class l_{X+1} and the previous classes in S_c.

Step 3: $M_X = (m_1, m_2, \ldots m_X)$ to classify instances in D_{X+1}, the similarity between old classes $(l_1, l_2, \ldots l_X)$ and new class l_{X+1} can be computed by using each of 1-vs.-rest binary classifier m_i.

In next step, new class l_{X+1} separated and now for S_c there is two task,

Step 4: Build M_{X+1} new classifier for l_{X+1}.

Step 5: Update existing classifier as the classes that are in S_c.

6.2 Incremental Class Learning [29]

Incremental learning is encouraged by the thought of the human learning process. It learns most of the knowledge by experience like humans do. It learns new knowledge over time instead of finding existing knowledge.

Let us assume we have Classification model $M_X = (m_1, m_2, \ldots m_X)$ as input and $D^{p_r} = (D_1, D_2, \ldots D_X)$ dataset with previous classes. The new dataset is D_{X+1} and λ_s is similarity threshold. we need classification model $M_{X+1} = (m_1, m_2, \ldots m_X, m_{X+1})$ to learn incrementally using previous data. To obtain this the following steps need to be executed.

Step 1: Initialize S_c to empty set.

Step 2: Initialize the count and record total instances in D_{X+1} (positive classified by m_i).

Sept 3 : Use m_i and classify each instances in D_{X+1} and record total positive instances classified by m_i.

Step 4: check whether there are disproportionate instances in D_{X+1} as positive by m_i to reduce class l_i. as resemblance to class l_{X+1}. The λ_s is threshold that regulate how many instances in D_{X+1} should be classified l_i before considering as analogous to l_{X+1}.

Step 5: build novel classifier M_{X+1}.

6.3 Nearest Class Mean [83, 112]

The nearest class mean is generally used for large-scale image classification. Two methods were used in most Research for large-scale image classification, K-NN and nearest class mean; the nearest class mean is more flexible than K-NN. The NCM characterizes classes by the mean feature vectors of their components.

Let us assume we have image P, represented in r-dimension with the feature vector. $\vec{f} \in \mathbb{R}^r$.

Step 1: compute the class centroid c_a for each class $a \in A$.

$$c_a = \frac{1}{P_a} \sum_{i \in P_k} \vec{f}_i, \tag{2}$$

where P_a is the set of images label with class a and the set of centroid (for each class) is $C = \{c_a\}$ and it has cardinality $|C| = |A|$

Step 2: The classifier of nearest class means to classify an image P will search the closest centroid in feature space,

$$a^*(P) = \overset{argmni}{a \in A} \left\| \vec{f} - c_a \right\|^2, \tag{3}$$

where $\vec{f}$ is the feature vector of P.

6.4 1-vs.-rest [111]

Classical machine learning uses different functions as output for multi-class classification. However, these functions can not reject unknown classes. There is a need to normalise these functions for each class across the training classes to achieve rejection capability in the output mechanism.

1-vs.-rest is one of the methods that provides rejection capability. Let us assume there is a 1-vs.-rest method with s sigmoid functions, where s is a known object. We have ith sigmoid function for p_i class. The 1-vs.-rest distinguishes the classes as positive and negative for p_i class such that $q = p_i$ is a positive class and the rest of all $q \neq p_i$ classes are negative. The loss can be calculated as log loss for all s sigmoid functions for the training data,

$$loss = \sum_{i=1}^{s} \sum_{j=1}^{n} -\mathbb{I}(q_j = p_i) logp_b(q_j = p_i) - \mathbb{I}(q_j \neq p_i)(1 - logp_b(q_j = p_i)), \tag{4}$$

where $\mathbb{I}$ is the Indicator function, j= 1 to n (n = Number of instances) and probability output of s sigmoids for jth input of ith dimension of r is $p(y_j = x_i) = sigmoid(r_{j,i})$ reject unseen classes such that

$$\hat{q} = \begin{Bmatrix} reject, \; if sigmoid(t_i) < k_i, \; \forall \, p_i \; \in \; q_i \\ argmax_{p_i \in q} \; sigmoid(t_i), \; Otherwise \end{Bmatrix}, \tag{5}$$

where k_i is a threshold that belongs to p_i, if probability p_b is less than the threshold, then the input will be rejected. 1-vs.-rest predicts the class that has the highest probability.

These benchmark algorithms have been used in computer vision and natural language processing to elucidate problems in open-world settings. Several frameworks and models are suggested centred on these algorithms or used these algorithms. The center-based similarity does not support the artificial neural network; thus, some authors used its extension or modified versions [3, 29]. The NCM has been used as the baseline in References [7, 22]. The classical model of NCM considers examples of training a novel 1-vs.-rest classifier for individual supplementary classes. Hence in the case of large-scale datasets and multiple classes, it becomes a burden for classifiers. Earlier trained classifiers will also need to be restructured to increase their performance; thus, the extension of NCM such as NNO [6] implemented, which gives better accuracy. The NCC [27] also inspired by NCM. In References [22, 27, 81, 125] the concept of incremental learning has been used to recognise objects in open world.

Identification of unknown data has a significant impact on OWML. To adapt the capacity for rejection of unknown data, many authors use the 1-vs.-rest [111] method. Some authors have used the 1-vs.-rest method to identify unknown data [116, 124, 134] and some researchers used this method as part of their framework to distinguish the known and unknown data [125, 148]. As we have seen in the literature, the baseline approaches are still generating auspicious outputs with current expertise, such as CNN and DNN frameworks. We also perceive that the few modified versions of these algorithms reinforce the model in OWML.

7 DISCUSSION

We discussed many challenges in the open-world environment separately for CVIP and NLP in Sections 4 and 5, respectively. Many common issues can affect both domains; one is the correctness of learning. The marginal errors that remain during the open classification can propagate to the future task while updating the knowledge base, resulting in more and more errors. This concern must be addressed to ensure that OWML approaches are effective. Another critical issue with OWML is knowledge relevance, which is difficult to verify. It means that whether the existing knowledge applies to a new learning task or not.

Learning expression and reasoning is another critical issue. In the earlier days of artificial intelligence, most of the analysis was done on logic-based learning models and reasoning. However, artificial intelligence analysis has redirected to statistical ML since the early 2000s. Since most of the methods of OWML contain a knowledge base, knowledge expression and reasoning are inherently applicable and essential for OWML. The system may learn a new task by reasoning to conclude new knowledge from earlier obtained knowledge. Therefore, what types of knowledge are suitable, how to express it, and what sorts of reasoning abilities are valuable to OWML are essential concerns that need to be addressed. Recently many researchers shifted their focus toward the multi-task OWML classification. Suppose various classes of tasks are involved from various domains, such as entity classification and feature extraction, then sharing previous learning across the domain is challenging for future task learning.

Moreover, in classical machine learning, there are models that can refuse the unseen instances to classify neither as seen nor unseen. The model can refuse these unseen instances based on negative samples given at training time. The negative examples supplied at training time help to learn useful discriminators among positive and negative examples [48]. There is no need to train on every possible negative class; only current classes are present for the inference purpose. However, OWML is more generalized approach of such models that not only refuse the unseen instance but also classify them as novel unseen instances.

8 OPEN RESEARCH CHALLENGES

Learning in a dynamic environment is still a challenging task due to the unpredictable nature of the upcoming events. The question is as follows: How can we integrate the classifier to obtain a sub-knowledge of unknown classes and reduce the open-space risks? Here we discuss the significant open research challenges in OWML for future researchers.

- **Incremental Volume of Data:** This is the era of digitization. Hence various sources are generating a large amount of data. These data are not only significant in volume but also unstructured. Managing and finding the different classes of the various domains is very difficult as continuous updates appear with additional unseen instance categories. Currently, there is a lack of mechanisms to deal with real-world data [45].

- **Identifying Novel Classes:** Once a system identifies instances as unseen or rejects unseen classes, the system has to learn about the classes of these unseen instances. There is a need for a complete framework to address these unseen instances and identify novel classes for them.There is a lack of an OWML model to discover and identify the unseen classes. Various models exist in OWML to discover unseen instances, but very few can identify the novel classes out of these unseen instances [158].
- **Updating a Knowledge Base:** There is a need for frameworks that can accumulate new knowledge and update the knowledge base at runtime. There are many complexities in appending a new domain and its classes in a progressive environment as input data overgrows. Apart from KB updations, these systems must be capable of using newly identified classes for the next prediction without retraining the entire model [143, 156].
- **Retention of Obtained Knowledge:** Due to the complexities of cross-domain data and the verity of classes stored in KB, there is a lack of end-to-end frameworks that can efficiently retain the previously stored knowledge from the knowledge base for future tasks. To the best of our knowledge, very few learning models use an incremental approach to learn unseen instances utilising previously obtained knowledge for the subsequent prediction [72, 139].
- **Open-space Risk:** The model can assess an algorithm's performance on a training data set. However, it cannot determine exactly how well an algorithm will perform in practice, since it does not understand data distribution. It is an empirical risk associated with classical machine learning. However, for OWML, it is essential to evaluate how the model handles the risk of the unseen. In OWML, it refers to open space risk. Sometimes it can be minimized by integrating various training errors (empirical risks). The open-space risk needs to be addressed to learn more accurately with increasing openness.
- **End-to-end Open Framework:** There is a need for a generic framework to discover unseen classes in the real-world domain that can function end-to-end for learning in a progressive environment. To build the complete framework of an OWML learning system, one needs to execute both operations together (discovery of unseen instances and identification of novel classes). Hence it needs two or more different modules to function dynamically. These modules can use different methods; hence, modules' concatenation and synchronization are relatively complicated as different methods are involved. The model needs to precisely address both seen and unseen classes with synchronization of newly adopted classes.
- **Mode Collapse:** It is also known as catastrophic collapse. It occurs when generative adversarial-based networks are trained. The model consists of two prominent elements. First is a discriminator that distinguishes between synthetically generated and real-world data examples. The second is a generator that learns to build synthetic data examples that are natural enough to mislead the discriminator. These models are considered trained if (1) the generator can produce examples that mislead the discriminator, and (2) the generator produces diverse synthetic samples similar to the original samples. Mode collapse occurs when the generator cannot perform Task 2, and all generated examples are equivalent to each other [115]. To the best of our knowledge, no research is available to handle mode collapse issues for OWML models.

9 CONCLUSION AND FUTURE DIRECTIONS

In this article, we investigated the works in OWML proposed since the 2010s. We have also discussed the prominence and many real-life applications of OWML. Many algorithms, models, and frameworks have been proposed in the literature to address numerous objectives allied to open-world settings. The domain is relatively new; thus, there are inadequate sources of information. The presented review will help in understanding the open-world scenario, working, and associated challenges. We provided a task-based classification of OWML in CVIP and NLP. Further, we discussed various techniques and used datasets in OWML. The limitations of numerous technologies are also analyzed to facilitate promising future extensions of these methods.

We have reviewed numerous research works on OWML in CVIP and NLP. Based on the study, we have identified three significant aspects necessary to achieve learning in the open world. OWML can be improved by enhancing these aspects: model, rejection capability, and identification of new classes. This section briefly discusses and analyses the limitation and discusses the research directions in detail.

Open-world Models. The existing models of OWML are working in a hybrid manner and address the problem in parts. There is a lack of models available that can work end-to-end. The end-to-end model for OWML can strengthen the classification for both categories, known-known class and known unknown class. To the best of our knowledge, there is no promising model available for unknown-unknown class identification, which is one of the challenging categories in OWML. The existing methods for unknown-unknown class classification are worth extending further.

Rejection of Unknown Classes. Few works are available to reject unknown-unknown classes, while automation systems entirely depend on unseen class rejection with high accuracy. Further work to increase the rejection capability of unknown classes can make the system more reliable as the real-world application faces many unknown objects while working in a dynamic environment. Existing models need more improvements to reject unseen classes with high accuracy.

Identification of Unseen Classes. Most existing models either detect known or reject unknown, but after rejecting classes as unknown, no promising mechanism is available to further identify classes in rejected data. There is a need for models that can identify the number of hidden clauses in rejected data.

A SUPPLEMENTARY MATERIAL

We have prepared a supplementary file, which includes the following details. The first part discusses the benchmark dataset used in open-world learning. It also lists all datasets based on the proposed year and provides the links for publicly available datasets. The next part discusses related areas such as Transfer Learning, Active Learning, Lifelong Machine Learning, and Multi-Task Learning. The supplementary file is uploaded with the article and available on the GitHub repository.[2]

REFERENCES

[1] A. S. Albahri, A. A. Zaidan, O. S. Albahri, B. B. Zaidan, A. H. Alamoodi, Ali H. Shareef, Jwan K. Alwan, Rula A. Hamid, M. T. Aljbory, Ali Najm Jasim, et al. 2021. Development of IoT-based mhealth framework for various cases of heart disease patients. *Health Technol.* 11, 5 (2021), 1013–1033.

[2] Osamah Shihab Albahri, A. A. Zaidan, B. B. Zaidan, Ahmed Shihab Albahri, Ali H. Mohsin, K. I. Mohammed, and M. A. Alsalem. 2022. New mHealth hospital selection framework supporting decentralised telemedicine architecture for outpatient cardiovascular disease-based integrated techniques: Haversine-GPS and AHP-VIKOR. *J. Amb. Intell. Human. Comput.* 13, 1 (2022), 219–239.

[3] Ethem Alpaydin. 2020. *Introduction to Machine Learning*. MIT Press.

[4] Plamen P. Angelov and Xiaowei Zhou. 2008. Evolving fuzzy-rule-based classifiers from data streams. *IEEE Trans. Fuzzy Syst.* 16, 6 (2008), 1462–1475.

[5] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The Semantic Web*. Springer, 722–735.

[2]https://github.com/jitendraparmar94/OWML.

[6] Abhijit Bendale and Terrance Boult. 2015. Towards open world recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1893–1902.

[7] Abhijit Bendale and Terrance E. Boult. 2016. Towards open set deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1563–1572.

[8] Christopher M. Bishop and M. Nasrabadi Nasser. 2006. Pattern recognition and machine learning. Vol. 4, no. 4.

[9] Paul Bodesheim, Alexander Freytag, Erik Rodner, and Joachim Denzler. 2015. Local novelty detection in multi-class recognition problems. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision.* IEEE, 813–820.

[10] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems.* 2787–2795.

[11] Léon Bottou. 2014. From machine learning to machine reasoning. *Mach. Learn.* 94, 2 (2014), 133–149.

[12] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: Identifying density-based local outliers. In *Proceedings of the ACM SIGMOD International Conference on Management of Data.* 93–104.

[13] Nadia Burkart and Marco F. Huber. 2021. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* 70 (2021), 245–317.

[14] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* 2, 3 (2011), 1–27.

[15] Zhilu Chen and Xinming Huang. 2017. End-to-end learning for lane keeping of self-driving cars. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV'17).* 1856–1860.

[16] Zhiyuan Chen and Bing Liu. 2014. Mining topics in documents: Standing on the shoulders of big data. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 1116–1125.

[17] Gouenou Coatrieux, Laurent Lecornu, Bülent Sankur, and Ch Roux. 2006. A review of image watermarking applications in healthcare. In *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society.* 4691–4694.

[18] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. 2017. EMNIST: Extending MNIST to handwritten letters. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'17).* IEEE, 2921–2926.

[19] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: An embedded spoken language understanding system for private-by-design voice interfaces. arXiv:1805.10190. Retrieved from https://arxiv.org/abs/1805.10190.

[20] Benjamin R. Cowan, Nadia Pantidi, David Coyle, Kellie Morrissey, Peter Clarke, Sara Al-Shehri, David Earley, and Natasha Bandeira. 2017. " What can i help you with?" infrequent users' experiences of intelligent personal assistants. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services.* 1–12.

[21] Laurens De Haan, Ana Ferreira, and Ana Ferreira. 2006. *Extreme Value Theory: An Introduction*, Vol. 21. Springer.

[22] Rocco De Rosa, Thomas Mensink, and Barbara Caputo. 2016. Online open world recognition. arXiv:1604.02275. Retrieved from https://arxiv.org/abs/1604.02275.

[23] Kerstin Denecke, Alaa Abd-Alrazaq, and Mowafa Househ. 2021. Artificial intelligence for chatbots in mental health: Opportunities and challenges. In *Multiple Perspectives on Artificial Intelligence in Healthcare.* Springer, 115–128.

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 248–255.

[25] Yong Deng. 2015. Generalized evidence theory. *Appl. Intell.* 43, 3 (2015), 530–543.

[26] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT.* 4171–4186.

[27] Tri Doan and Jugal Kalita. 2017. Overcoming the challenge for text classification in the open world. In *Proceedings of the IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC).* 1–7.

[28] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1778–1785.

[29] Geli Fei and Bing Liu. 2016. Breaking the closed world assumption in text classification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 506–514.

[30] Geli Fei, Shuai Wang, and Bing Liu. 2016. Learning cumulatively to become more knowledgeable. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 1565–1574.

[31] Peter W. Frey and David J. Slate. 1991. Letter recognition using Holland-style adaptive classifiers. *Mach. Learn.* 6, 2 (1991), 161–182.

[32] Yang Gao, Yi-Fan Li, Bo Dong, Yu Lin, and Latifur Khan. 2019. SIM: Open-world multi-task stream classifier with integral similarity metrics. In *Proceedings of the IEEE International Conference on Big Data (Big Data'19).* IEEE, 751–760.

[33] Zongyuan Ge, Sergey Demyanov, Zetao Chen, and Rahil Garnavi. 2017. Generative OpenMax for multi-class open set classification. In *Proceedings of the British Machine Vision Conference*. British Machine Vision Association and Society for Pattern Recognition.
[34] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. 2020. Recent advances in open set recognition: A survey. *IEEE Trans. Pattern Anal. Machine Intell.* 43, 10 (2020), 3614–3631.
[35] Jan-Mark Geusebroek, Gertjan J. Burghouts, and Arnold W. M. Smeulders. 2005. The Amsterdam library of object images. *International Journal of Computer Vision* 61, 1 (2005), 103–112.
[36] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. 2016. *Deep Learning*. Vol. 1. MIT press Cambridge.
[37] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *Communications of the ACM* 63, 11 (2020), 139–144.
[38] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. arXiv:1412.6572. Retrieved from https://arxiv.org/abs/1412.6572.
[39] Gregory Griffin, Alex Holub, and Pietro Perona. 2007. Caltech-256 object category dataset. Technical Report 7694. California Institute of Technology.
[40] Xiaojie Guo, Amir Alipour-Fanid, Lingfei Wu, Hemant Purohit, Xiang Chen, Kai Zeng, and Liang Zhao. 2019. Multi-stage deep classifier cascades for open world recognition. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management.* 179–188.
[41] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. 2016. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Proceedings of the European Conference on Computer Vision.* 87–102.
[42] Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2787–2792.
[43] F. Maxwell Harper and Joseph A. Konstan. 2015. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2015), 1–19.
[44] Mehadi Hassen and Philip K. Chan. 2020. Learning a neural-network-based representation for open set recognition. In *Proceedings of the SIAM International Conference on Data Mining*. SIAM, 154–162.
[45] Jiangpeng He, Runyu Mao, Zeman Shao, and Fengqing Zhu. 2020. Incremental learning in online scenario. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 13926–13935.
[46] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*. 507–517.
[47] Dan Hendrycks and Kevin Gimpel. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv:1610.02136. Retrieved from https://arxiv.org/abs/1610.02136.
[48] Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. 2021. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 1074–1083.
[49] Andrei De Souza Inácio, Matheus Gutoski, André Eugênio Lazzaretti, and Heitor Silvério Lopes. 2021. OSVidCap: A framework for the simultaneous recognition and description of concurrent actions in videos in an open-set scenario. *IEEE Access* 9 (2021), 137029–137041.
[50] Hisao Ishibuchi, Ken Nozaki, Naohisa Yamamoto, and Hideo Tanaka. 1995. Selecting fuzzy if-then rules for classification problems using genetic algorithms. *IEEE Trans. Fuzzy Syst.* 3, 3 (1995), 260–270.
[51] Lalit P. Jain, Walter J. Scheirer, and Terrance E. Boult. 2014. Multi-class open set recognition using probability of inclusion. In *Proceedings of the 13th European Conference on Computer Vision.* 393–409.
[52] A. Jeya Christy and K. Dhanalakshmi. 2021. Content-based image recognition and tagging by deep learning methods. *Wireless Pers. Commun.* (2021), 1–26.
[53] Jiepu Jiang, Ahmed Hassan Awadallah, Rosie Jones, Umut Ozertem, Imed Zitouni, Ranjitha Gurunath Kulkarni, and Omar Zia Khan. 2015. Automatic online evaluation of intelligent assistants. In *Proceedings of the 24th International Conference on World Wide Web*. 506–516.
[54] Pedro R. Mendes Júnior, Roberto M. De Souza, Rafael de O. Werneck, Bernardo V. Stein, Daniel V. Pazinato, Waldir R. de Almeida, Otávio A. B. Penatti, Ricardo da S. Torres, and Anderson Rocha. 2017. Nearest neighbors distance ratio open-set classifier. *Mach. Learn.* 106, 3 (2017), 359–386.
[55] Dan Jurafsky. 1997. *Switchboard SWBD-DAMSL Shallow-discourse-function Annotation Coders Manual.* Institute of Cognitive Science Technical Report.
[56] Mohammed Waleed Kadous. 2002. Temporal classification: Extending the classification paradigm to multivariate time series. University of New South Wales, Kensington.
[57] James M. Keller, Michael R. Gray, and James A. Givens. 1985. A fuzzy k-nearest neighbor algorithm. *IEEE Trans. Syst. Man Cybernet.* 4 (1985), 580–585.

[58] Zubair Ahmed Khan and Asma Rizvi. 2021. AI based facial recognition technology and criminal justice: Issues and challenges. *Turk. J. Comput. Math. Educ.* 12, 14 (2021), 3384–3392.
[59] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1746–1751.
[60] Shu Kong and Deva Ramanan. 2021. Opengan: Open-set recognition via open data generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 813–822.
[61] Sotiris B. Kotsiantis, I. Zaharakis, P. Pintelas, et al. 2007. Supervised machine learning: A review of classification techniques. *Emerg. Artif. Intell. Appl. Comput. Eng.* 160, 1 (2007), 3–24.
[62] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. Master's thesis. Department of Computer Science, University of Toronto.
[63] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, Vol. 25. 1097–1105.
[64] Ludmila Kuncheva. 2000. *Fuzzy Classifier Design*. Vol. 49. Springer Science & Business Media.
[65] Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings*, LakeTahoe, CA, 331–339.
[66] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2. 2169–2178.
[67] Ya Le and Xuan Yang. 2015. Tiny imagenet visual recognition challenge. Stanford CS 231N Course. Retrieved from http://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.
[68] Yann LeCun. 1998. The MNIST Database of Handwritten Digits. Retrieved from http://yann.lecun.com/exdb/mnist/.
[69] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
[70] Kuang-Chih Lee, Jeffrey Ho, and David J. Kriegman. 2005. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intell.* 27, 5 (2005), 684–698.
[71] Qingming Leng, Mang Ye, and Qi Tian. 2019. A survey of open-world person re-identification. *IEEE Trans. Circ. Syst. Vid. Technol.* 30, 4 (2019), 1092–1108.
[72] Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 12 (2017), 2935–2947.
[73] Ting-En Lin and Hua Xu. 2019. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5491–5496.
[74] Ting-En Lin and Hua Xu. 2019. A post-processing method for detecting unknown intent of dialogue system via pre-trained deep neural network classifier. *Knowl.-Bas. Syst.* 186 (2019), 104979.
[75] Fan Liu and Yong Deng. 2020. Determine the number of unknown targets in open world based on elbow method. *IEEE Trans. Fuzzy Syst.* 29, 5 (2020), 986–995.
[76] Li Liu and Fanzhang Li. 2022. A survey on dynamic fuzzy machine learning. *Comput. Surv.* (2022). Just Accepted. https://doi.org/10.1145/3544013
[77] Abbas Saliimi Lokman and Mohamed Ariff Ameedeen. 2018. Modern chatbot systems: A technical review. In *Proceedings of the Future Technologies Conference*. 1012–1023.
[78] Vincent Lonij, Ambrish Rawat, and Maria-Irina Nicolae. 2017. Open-world visual recognition using knowledge graphs. arXiv:1708.08310. Retrieved from https://arxiv.org/abs/1708.08310.
[79] Edwin Lughofer. 2022. Evolving multi-user fuzzy classifier systems integrating human uncertainty and expert knowledge. *Inf. Sci.* 596 (2022), 30–52.
[80] Edwin Lughofer, Eva Weigl, Wolfgang Heidl, Christian Eitzinger, and Thomas Radauer. 2015. Integrating new classes on the fly in evolving fuzzy classifier designs and their application in visual inspection. *Appl. Soft Comput.* 35 (2015), 558–582.
[81] Sahisnu Mazumder, Nianzu Ma, and Bing Liu. 2018. Towards a continuous knowledge learning engine for chatbots. arXiv:1802.06024. Retrieved from https://arxiv.org/abs/1802.06024.
[82] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 43–52.
[83] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. 2013. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 11 (2013), 2624–2637.
[84] Donald Michie, David J. Spiegelhalter, and Charles C. Taylor. 1994. Machine learning, neural and statistical classification. Retrieved from ftp.ncc.up.pt/pub/statlog/, Data available at http://www.ncc.up.pt/liacc/ML.
[85] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. arXiv:1301.3781. Retrieved from https://arxiv.org/abs/1301.3781.
[86] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*. 3111–3119.

[87] Dimity Miller, Niko Sunderhauf, Michael Milford, and Feras Dayoub. 2021. Class anchor clustering: A loss for distance-based open set recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3570–3578.

[88] George A. Miller. 1995. WordNet: A lexical database for english. *Commun. ACM* 38, 11 (November 1995), 39–41. https://doi.org/10.1145/219717.219748

[89] Philip Moore and Hai Van Pham. 2015. On context and the open world assumption. In *Proceedings of the IEEE 29th International Conference on Advanced Information Networking and Applications Workshops*. 387–392.

[90] Xin Mu, Kai Ming Ting, and Zhi-Hua Zhou. 2017. Classification under streaming emerging new classes: A solution using completely-random trees. *IEEE Trans. Knowl. Data Eng.* 29, 8 (2017), 1605–1618.

[91] Susan M. Mudambi and David Schuff. 2010. Research note: What makes a helpful online review? A study of customer reviews on Amazon. com. *MIS Quart.* (2010), 185–200.

[92] Kevin P. Murphy. 2012. *Machine Learning: A Probabilistic Perspective*. MIT Press.

[93] Detlef Nauck, Frank Klawonn, and Rudolf Kruse. 1997. *Foundations of Neuro-Fuzzy Systems*. John Wiley & Sons, Inc.

[94] S. Nene, S. Nayar, and H. Murase. 1996. Columbia object image library (coil-20). Technical Report, Columbia University.

[95] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. 2011. Reading digits in natural images with unsupervised feature learning.

[96] Anh Nguyen, Jason Yosinski, and Jeff Clune. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 427–436.

[97] David Nister and Henrik Stewenius. 2006. Scalable recognition with a vocabulary tree. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2. 2161–2168.

[98] Manuel Olave, Vladislav Rajkovic, and Marko Bohanec. 1989. An application for admission in public school systems. *Expert Syst. Publ. Admin.* 1 (1989), 145–160.

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

[100] Poojan Oza and Vishal M. Patel. 2019. Deep cnn-based multi-task learning for open-set recognition. arXiv:1903.03161. Retrieved from https://arxiv.org/abs/1903.03161.

[101] Jitendra Parmar, Sanskar Soni, and Satyendra Singh Chouhan. 2020. OWI: Open-world intent identification framework for dialog based system. In *Proceedings of the International Conference on Big Data Analytics*. 329–343.

[102] Sayanta Paul and Sriparna Saha. 2020. CyberBERT: BERT for cyberbullying identification. *Multimedia Syst.* (2020), 1–8.

[103] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1532–1543.

[104] John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, Vol. 10. MIT Press, 61–74.

[105] Sridhama Prakhya, Vinodini Venkataram, and Jugal Kalita. 2017. Open set text classification using convolutional neural networks. In *Proceedings of the International Conference on Natural Language Processing*.

[106] Hemant Purohit, Carlos Castillo, Fernando Diaz, Amit Sheth, and Patrick Meier. 2014. Emergency-relief coordination on social media: Automatically matching resource requests and offers. *First Monday* (2014).

[107] David Ratcliffe and Kerry Taylor. 2014. Closed-world concept induction for learning in OWL knowledge bases. In *Proceedings of the International Conference on Knowledge Engineering and Knowledge Management*. Springer, 429–440.

[108] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. 2018. Deep learning for medical image processing: Overview, challenges and the future. *Classif. BioApps* (2018), 323–350.

[109] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. 2018. Meta-learning for semi-supervised few-shot classification. In *Proceedings of the International Conference on Learning Representations*.

[110] Jason Rennie and Ken Lang. 2008. The 20 Newsgroups Data Set. Retrieved from http://qwone.com/jason/20Newsgroups.

[111] Ryan Rifkin and Aldebaro Klautau. 2004. In defense of one-vs.-all classification. *J. Mach. Learn. Res.* 5 (2004), 101–141.

[112] Marko Ristin, Matthieu Guillaumin, Juergen Gall, and Luc Van Gool. 2014. Incremental learning of ncm forests for large-scale image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3654–3661.

[113] Johannes A. Roubos, Magne Setnes, and Janos Abonyi. 2003. Learning fuzzy classification rules from labeled data. *Information sciences* 150, 1–2 (2003), 77–93.

[114] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 3 (2015), 211–252.

[115] Divya Saxena and Jiannong Cao. 2021. Generative adversarial networks (GANs): Challenges, solutions, and future directions. *ACM Comput. Surv.* 54, 3, Article 63 (May 2021), 42 pages. https://doi.org/10.1145/3446374
[116] Walter J. Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E. Boult. 2012. Toward open set recognition. *IEEE Trans. Pattern Anal. Mach. Intelligence* 35, 7 (2012), 1757–1772.
[117] Walter J. Scheirer, Lalit P. Jain, and Terrance E. Boult. 2014. Probability models for open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 36, 11 (2014), 2317–2324.
[118] Walter J. Scheirer, Anderson Rocha, Ross J. Micheals, and Terrance E. Boult. 2011. Meta-recognition: The theory and practice of recognition score analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 8 (2011), 1689–1695.
[119] M. Schuster and K. K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Sign. Process.* 45, 11 (1997), 2673–2681.
[120] Vikash Sehwag, Arjun Nitin Bhagoji, Liwei Song, Chawin Sitawarin, Daniel Cullina, Mung Chiang, and Prateek Mittal. 2019. Analyzing the robustness of open-world machine learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*. 105–116.
[121] Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. 2019. An open-world extension to knowledge graph completion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3044–3051.
[122] Baoxu Shi and Tim Weninger. 2018. Open-world knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
[123] Elizabeth Shriberg, Andreas Stolcke, Daniel Jurafsky, Noah Coccaro, Marie Meteer, Rebecca Bates, Paul Taylor, Klaus Ries, Rachel Martin, and Carol Van Ess-Dykema. 1998. Can prosody aid the automatic classification of dialog acts in conversational speech? *Lang. Speech* 41, 3-4 (1998), 443–492.
[124] Lei Shu, Hu Xu, and Bing Liu. 2017. DOC: Deep open classification of text documents. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2911–2916.
[125] Lei Shu, Hu Xu, and Bing Liu. 2018. Unseen class discovery in open-world classification. arXiv:1801.05609. Retrieved from https://arxiv.org/abs/1801.05609.
[126] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556. Retrieved from https://arxiv.org/abs/1409.1556.
[127] Liwei Song, Vikash Sehwag, Arjun Nitin Bhagoji, and Prateek Mittal. 2020. A critical evaluation of open-world machine learning. arXiv:2007.04391. Retrieved from https://arxiv.org/abs/2007.04391.
[128] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. arXiv:1505.00387. Retrieved from https://arxiv.org/abs/1505.00387.
[129] Wallace Stevens. 2021. Efficient uncertainty estimation for open-set object detection. In *Epistemic Uncertainty Estimation for Object Detection in Open-Set Conditions*, 91.
[130] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Comput. Ling.* 26, 3 (2000), 339–373.
[131] Martin Thoma. 2017. The hasyv2 dataset. arXiv:1701.08380. Retrieved from https://arxiv.org/abs/1701.08380.
[132] Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and Their Compositionality*. 57–66.
[133] Gokhan Tur, Dilek Hakkani-Tür, and Larry Heck. 2010. What is left to be understood in ATIS? In *Proceedings of the IEEE Spoken Language Technology Workshop*. 19–24.
[134] Nikhita Vedula, Rahul Gupta, Aman Alok, and Mukund Sridhar. 2020. Automatic discovery of novel intents & domains from text utterances. arXiv:2006.01208. Retrieved from https://arxiv.org/abs/2006.01208.
[135] Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and Srinivasan Parthasarathy. 2019. Towards open intent discovery for conversational text. arXiv:1904.08524. Retrieved from https://arxiv.org/abs/1904.08524.
[136] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, Vol. 29. 3630–3638.
[137] Alex Hai Wang. 2010. Detecting spam bots in online social networking sites: A machine learning approach. In *Proceedings of the IFIP Annual Conference on Data and Applications Security and Privacy*. 335–342.
[138] Yuhan Wang, Weidong Xiao, Zhen Tan, and Xiang Zhao. 2021. Caps-OWKG: A capsule network model for open-world knowledge graph. *Int. J. Mach. Learn. Cybernet.* 12, 6 (2021), 1627–1637.
[139] Yen-Hsiang Wang, Chih-Yang Lin, Tipajin Thaipisutikul, and Timothy K. Shih. 2022. Single-head lifelong learning based on distilling knowledge. *IEEE Access* 10 (2022), 35469–35478.
[140] John Willes, James Harrison, Ali Harakeh, Chelsea Finn, Marco Pavone, and Steven Waslander. 2021. Bayesian embeddings for few-shot open world recognition. arXiv:2107.13682. Retrieved from https://arxiv.org/abs/2107.13682.
[141] John Wright, Allen Y. Yang, Arvind Ganesh, S. Shankar Sastry, and Yi Ma. 2008. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 31, 2 (2008), 210–227.

[142] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Junchi Yan, and Hongyuan Zha. 2021. Towards open-world recommendation: An inductive model-based collaborative filtering approach. In *Proceedings of the International Conference on Machine Learning*. 11329–11339.

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

[144] Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. 2021. NGC: A unified framework for learning with open-world noisy data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 62–71.

[145] Bo Xiao and Izak Benbasat. 2007. E-commerce product recommendation agents: Use, characteristics, and impact. *MIS Quart.* (2007), 137–209.

[146] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747. Retrieved from https://arxiv.org/abs/1708.07747.

[147] Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

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

[149] Ronald R. Yager and Janusz Kacprzyk. 2012. *The Ordered Weighted Averaging Operators: Theory and Applications*. Springer Science & Business Media.

[150] J. Yang and Joseph. Coughlin. 2014. In-vehicle technology for self-driving cars: Advantages and challenges for aging drivers. *Int. J. Automot. Technol.* 15, 2 (2014), 333–340.

[151] N. Yang. 2019. Douban movie and NetEase music datasets and model code. The President & Fellows of Harvard College, Cambridge, MA, USA. https://doi.org/10.7910/DVN/JGH1HA

[152] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven C. H. Hoi. 2021. Deep learning for person re-identification: A survey and outlook. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021).

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

[154] Kun Yue, Jiahui Wang, Xinbai Li, and Kuang Hu. 2020. Representation-based completion of knowledge graph with open-world data. In *Proceedings of the 5th International Conference on Computer and Communication Systems (ICCCS'20)*. 1–8.

[155] He Zhang and Vishal M. Patel. 2016. Sparse representation-based open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 8 (2016), 1690–1696.

[156] Tianchen Zhang, Zhongqin Bi, Meijing Shan, and Yongbin Li. 2022. IFGAN: Information fusion generative adversarial network for knowledge base completion. *Expert Syst.* (2022), e12984.

[157] Wayne Xin Zhao, Sui Li, Yulan He, Liwei Wang, Ji-Rong Wen, and Xiaoming Li. 2016. Exploring demographic information in social media for product recommendation. *Knowl. Inf. Syst.* 49, 1 (2016), 61–89.

[158] Zhun Zhong, Linchao Zhu, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. 2021. OpenMix: Reviving known knowledge for discovering novel visual categories in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*. 9457–9465.

[159] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 6 (2017), 1452–1464.

[160] Yajin Zhou and Xuxian Jiang. 2012. Dissecting android malware: Characterization and evolution. In *Proceedings of the IEEE Symposium on Security and Privacy*. 95–109.

Received 9 November 2021; revised 5 July 2022; accepted 29 August 2022