Towards Utilitarian Online Learning
– A Review of Online Algorithms in Open Feature Space

Yi He1, Christian Schreckenberger2, Heiner Stuckenschmidt2 and Xindong Wu3,4
1Old Dominion University, USA
2University of Mannheim, Germany
3Zhejiang Lab, China
4Hefei University of Technology, China
yihe@cs.odu.edu, {christian.schreckenberger, heiner}@uni-mannheim.de, xwu@zhejianglab.com

Abstract

Human intelligence comes from the capability to describe and make sense of the world surrounding us, often in a lifelong manner. Online Learning (OL) allows a model to simulate this capability, which involves processing data in sequence, making predictions, and learning from predictive errors. However, traditional OL assumes a fixed set of features to describe data, which can be restrictive. In reality, new features may emerge and old features may vanish or become obsolete, leading to an open feature space. This dynamism can be caused by more advanced or outdated technology for sensing the world, or it can be a natural process of evolution. This paper reviews recent breakthroughs that strived to enable OL in open feature spaces, referred to as Utilitarian Online Learning (UOL). We taxonomize existing UOL models into three categories, analyze their pros and cons, and discuss their application scenarios. We also benchmark the performance of representative UOL models, highlighting open problems, challenges, and potential future directions of this emerging topic.

1 Introduction

“A mind is like a parachute. It functions only when it is open.” – Allan H. Mogensen, Fundamentals of Human Engineering (1939)

Online Learning (OL) is an exceptional machine learning paradigm that builds predictive models in an environment where data are presented sequentially as streams [De Santis et al., 1988; Vovk, 1997; McMahan, 2017; Cesa-Bianchi and Orabona, 2021]. OL excels in scenarios where the data deluge [Bell et al., 2009] makes it too memory and computational intensive to load in and perform learning over entire data matrices of huge quantities [Wu et al., 2013; Aggarwal, 2007]. Despite its effectiveness, traditional OL algorithms typically make a strict assumption that all data observations must be described by a fixed feature space.

This assumption may not be justified in many real-world applications. To wit, consider an urban disaster monitoring system aided by OL, in which streaming data are sent from crowd sensors such as smart phones and sensor kits/sites of local users scattered across a geographically wide region [Capponi et al., 2019]. New sensory features are likely to emerge when new users join the crowd-sensing endeavor, committing data collected by their own devices, e.g., a new-generation cellphone equipped with totally new sensors; likewise, any old and pre-existing features can become unobserved in later time snapshots, since any users can stop or fail to commit data for various reasons, such as battery exhaustion or network malfunction. Instead of being fixed and known-in-advance, the feature space used to describe streaming data in such applications varies over time and thus is open.

Recently, we have witnessed a surge of OL studies that aim to tackle the challenge of learning data streams in open feature spaces. Early appearances of the problem go back to the seminal work of [Wenerstrom and Giraud-Carrier, 2006] and subsequent explorations by [Gomes et al., 2013; Zhang et al., 2016; Hou et al., 2017; He et al., 2019]. However, the existing studies on this problem have been divided into multiple communities. Each community possesses its own modeling assumptions, tailored solutions upon disparate ideas and design intuitions, and carried out evaluations with different datasets and metrics. Evolution of ideas remained parallel to date, communication became difficult.

Necessitated by the status quo of division, we deliver a timely review of prior arts and, more importantly, strive to unify and frame them under an umbrella paradigm, termed as Utilitarian Online Learning (UOL). We coin the term “utilitarian” to emphasize the common aim of previous studies, which was to relax the traditional OL assumption of a fixed feature space, so as to enhance the models’ functionality, usefulness, and practicality in real applications. Furthermore, to facilitate comprehension of existing UOL models, we draw a metaphor to the utilitarian apportionment problem in political economy [Koriyama et al., 2013]. We argue that the various UOL studies mainly differ in terms of their design intuitions that lead to locally-optimal apportioning of feature weights in repeated games. We will detail this metaphor in later sections.

Specific contributions of our survey are as follows:

1. It is a timely and the first survey of recent OL studies that built predictive models from streaming data in open feature spaces. We envision an umbrella term, Utilitarian Online Learning (UOL), to frame the prior arts and can...
provide a unified perspective to bridge the fragmented research communities and foster communication.

(2) We propose a taxonomy that classifies existing research into three categories based on their feature apportionment strategies. A metaphor to utilitarian apportionment in political economy is drawn to aid comprehension.

(3) We examine the representative methods in each category, analyze their pros and cons, and benchmark their performance on four widely used datasets.

(4) We identify the open challenges faced by existing UOL studies and endeavor to shed light on untrodden pathways. We hope this would stimulate future research and discover where these paths lead and how they connect.

2 Utilitarian Online Learning (UOL)

We first formulate the UOL learning problem in a generic form and then present our taxonomy of prior studies.

2.1 The UOL Problem

Write an input sequence \( \{ (x_t, y_t) \mid t = 1, 2, \ldots, T \} \), where each instance \( x_t \in \mathcal{X} \subseteq \mathbb{R}^{d_t} \) is a vector of \( d_t \) features, associated with a ground truth class label \( y_t \in \{-1, +1\} \). The repeated game progresses as follows. At each round \( t \), the learner \( h_t \) observes an instance \( x_t \), produces a prediction \( \hat{y}_t = h_t(x_t) \), and then suffers an instantaneous loss \( \ell(y_t, \hat{y}_t) \) based on the revealed ground truth \( y_t \). The loss serves as an update to \( h_{t+1} \), thereby preparing it for the next round. In an open feature space, the dimension of \( x_{t+1} \) could either be incremental (i.e., \( d_{t+1} > d_t \)) or decremental (i.e., \( d_{t+1} < d_t \)), due to the emergence of new features or unobserved old features, respectively. A generic objective of UOL is to minimize the regret [Cesa-Bianchi and Lugosi, 2006]:

\[
R(T) = \sum_{t=1}^{T} \ell(y_t, h_t(x_t)) - \min_{h^* \in \mathcal{H}} \sum_{t=1}^{T} \ell(y_t, h^*(x_t)), \tag{1}
\]

which gauges the gap between the cumulative loss of the learner over \( T \) rounds and that of the optimal decision \( h^* \) chosen from the hypothesis space \( \mathcal{H} \) in hindsight.

2.2 The UOL Challenges

Drawing a metaphor from political economy, we can analogize the UOL problem to the apportionment models [Koriyama et al., 2013], where a federation of members (e.g., the European Union) making repeated decisions under qualified majority rules. Each member is assigned a voting weight. New members can join at any time, while existing members may arbitrarily abstain from any voting decisions. The goal is to maximize the collective utility gain (minimize negative cumulative loss) of the federation in the long run.

Analogizing to UOL, a feature can be deemed as a member in the federation, where the weighted voting rule reduces the learner to a linear classifier \( h_t(x_t) = \text{sign}(w_t^T x_t) \). The “voting weight” of the \( i \)-th feature is \( w_{ti} \), the \( i \)-th entry of \( w_t \). Two technical challenges are naturally manifested:

**Challenge 1:** When introducing a new feature/member, it is important to ensure that the current decision will not be made with bias. A common strategy for assigning weight to a member is proportional apportionment [Penrose, 1946], which can be mapped to a learning problem by gauging the amount of information conveyed by the feature for prediction using mutual information between the feature and the label [Kraskov et al., 2004]. However, in an online process, a new feature may only be described by few instances, making a precise information measurement next to impossible.

**Challenge 2:** When a majority of members are absent from the decision-making process, the voting opinions of those who abstain remain unobserved. This can lead to less informed decisions, particularly in extreme cases where the vast majority of features/members abstain. In such situations, decision-making may be dominated by a few remaining features that are less informative, leading to educated guesses. Furthermore, if a member leaves the federation for an extended period or does not return, it is unclear how to redistribute their voting weights to other members, so that the subsequent decisions made by the remaining members still guarantee maximization of collective utility gain.

2.3 The UOL Taxonomy

We taxonomize the current UOL studies into three categories based on their different ideas to tackle the learning problem in an open feature space. We re-examine the metaphor of apportionment, from which we can discern the core concept of each category for achieving a utilitarian apportionment (UA). In this section, we provide an overview of the key ideas and intuitions behind them, while the technical details are discussed in the remainder of this survey.

i) Passive-Aggressive (PA) methods are ubiquitous in solving OL problems, which bear the principle of margin-maximization [Crammer et al., 2006]. The margin of an instance is proportional to the distance between this instance and the decision hyperplane that the learner approximates. In the case of PA learner, it itertates over an input sequence and update its weights only when it receive a feedback of incorrect prediction. The key idea that generalizes PA into UOL contexts is straightforward: any new member is not allowed to vote until the decision made by other members in the federation is evidently wrong. This means that when a new feature emerges in an incoming instance, its learning weight is set to zero if the remaining features are sufficient to make an accurate prediction. Otherwise, the PA learner apportions the weights of other features to the new feature by descending a proximal gradient. In doing so, the updated learner still maintains maximized margins in the incremented feature space.

ii) Feature Correlation (FC) methods. Correlation analysis among random variables finds many machine learning tasks, e.g., online feature selection [Wu et al., 2010; Yu et al., 2020], where the streaming features that are not correlated with the target variable can be pruned without sacrificing discriminant power. Instead of focusing on the pairwise correlation between feature and label, UOL puts more emphasis on the correlation among features – two highly correlated features can be deemed as allies in the federation. In the context of UA, this idea implies that the opinion of a member who abstained from voting can be proxied by the opinions of her allies who voted. In other words, if an old feature becomes unobservable, its value can be estimated from other
features that it is highly correlated with, allowing us to leverage its learned coefficient for accurate prediction.

iii) Evolutionary Ensemble (EE) methods. Voting of a member in UA is not necessarily determined by an individual, but rather a group of commissaries. For example, in the European Union, a country has multiple representatives that must be re-elected on a regular basis; those who fail to maximize the utility of their represented member are replaced with new ones. In UOL, likewise, a feature can be represented by a set of weak learners (e.g., shape functions [Lou et al., 2012]), from which a stronger classification model can be ensembled and trained. However, when a new feature is introduced, its weak learner may be initialized with bias due to a lack of training instances. With the evolution of the input sequence and its context, some weak learners may not be able to adapt and begin to deteriorate the ensemble performance. The EE models are designed to replace such outdated learners with new ones, which can be better initialized with updated and likely sufficient feature statistics.

3 The UOL Models and Discussion

In this section, we will explore the technical models of the three categories of UOL methods in greater detail. We will present key pieces of literature, distill generic forms of their objective functions, elaborate on their shared merits and differences, and discuss the application and learning scenarios in which they excel or are less competent.

3.1 Passive-Aggressive (PA) Models

The study of applying PA algorithm to resolve UOL problem was initiated by Zhang et al., 2015], which spurred a flurry of subsequent research including Zhang et al., 2016; Bollegala, 2017; Beyazit et al., 2018; Beyazit et al., 2019; Alagurajah et al., 2020; Dong et al., 2021; Liu et al., 2022b; Gu et al., 2022]. Their shared idea is to rescale the weight coefficients from the existing features to initialize new feature, as padded zero weights tend to incur prediction loss. Let $\bar{w}_{t+1} \in \mathbb{R}^{d_t+1}$ be the weight vector of $x_{t+1}$. Align $w_{t+1} = [\bar{w}_{t+1}, \hat{w}_{t+1}]^T$, where $\bar{w}_{t+1}$ and $w_t$ are associated with the same set of features, thus $\bar{w}_{t+1}$ are the weights of new features. Namely, $\bar{w}_{t+1} \in \mathbb{R}^{d_t}$ and $w_{t+1} \in \mathbb{R}^{d_t+1} \setminus \mathbb{R}^{d_t}$.

The objective shared by previous studies takes the form:

$$w_{t+1} = \arg\min_{w_{t+1} \in \mathbb{R}^{d_t+1}} d(\bar{w}_{t+1}, w_t) + \lambda \|\bar{w}_{t+1}\|_p + \mu \xi,$$

where $d(\cdot, \cdot)$ is a distance metric, and minimizing this term is equivalent to searching a proximal gradient direction that enforces minimal update on weights of existing features. $\|\cdot\|_p$ denotes $\ell_p$-norm, which encourages the new features’ weight vector to follow certain properties. $\xi$ is a slack variable to tolerate noises in data, and $\mu$ is a tuned parameter to balance the rigidity and slackness of the updating step (i.e., a larger $\mu$ requires a more rigid update). Existing models in this category mainly differ in the realization of distance metric $d(\cdot, \cdot)$, $\ell_p$-norm regularizer, and loss function $\ell_{t+1}$. We extrapolate several prominent realization of these terms as follows.

**Distance Metric.** Prominent examples include Euclidean distance $d(\bar{w}_{t+1}, w_t) = \|\bar{w}_{t+1} - \hat{w}_{t+1}\|^2$, which works particularly well in linear cases [Zhang et al., 2015; Zhang et al., 2016; Beyazit et al., 2019; Alagurajah et al., 2020; Liu et al., 2022b; Gu et al., 2022]. To promote non-linearity, Mahalanobis distance in a latent metric space [Dong et al., 2021], energy-based function for posterior maximization [Bollegala, 2017], or gauging dissimilarity among hidden representations in neural architectures [Beyazit et al., 2018] have been proposed.

**Regularizer.** To ensure numerical stability and restrict the wild initialization of new feature weights $\hat{w}_{t+1}$, Gaussian priors are prescribed, commonly referred to as the $\ell_2$-norm [Beyazit et al., 2019; Liu et al., 2022b; Gu et al., 2022]. To promote sparsity, [Zhang et al., 2015; Zhang et al., 2016; Alagurajah et al., 2020] exploited $\ell_1$-norm, which is deemed as the tightest relaxation of $\ell_0$-norm. The intuition is that, with the continual emergence of new features, the total dimension will soon grow to an unmanageably large size for efficient classification. To allow for feature pruning and a subsequent reduction in dimension, $\hat{w}_{t+1}$ is projected onto an $\ell_1$-ball at each round, namely, $\hat{w}_{t+1} \leftarrow \min\{1, 1/\|\bar{w}_{t+1}\|_1\} \bar{w}_{t+1}$, such that the feature vector is concentrated to its several largest-valued entries, and the features with trivial values are dropped as dimension grows.

**Loss Function.** The margin-maximization principle is implemented using the hinge loss $\ell_{t+1} = \max\{0, 1 - y_{t+1}(1 + x_{t+1})\}$ as a default choice, in both traditional OL and the emerging UOL models. This relaxes overly harsh constraints by introducing a soft-margin parameter $\xi$. In addition, the hinge loss lends a closed-form solution to Eq. (2), making the proximal gradients easy to compute, as demonstrated in [Zhang et al., 2016; Liu et al., 2022b]. Other loss functions, such as cross entropy [Bollegala, 2017] or Euclidean loss [Dong et al., 2021], can also be used to better suit loss measurement in various designs.

**Pros & Cons.** The PA models for UOL problem enjoy several remarkable advantages. First, they are usually equipped with closed-form solutions and thus require no step-size, making them amenable for implementation. Second, they inherit theoretical guarantees from margin-based classifiers such as SVMs, leading to tight regret bounds of $\mathcal{O}(\sqrt{T})$. As a result, all such PA models are asymptotically no regret with $T \to \infty$. Moreover, they encourage sparse model solutions, which gives them the capability of online feature selection - a desirable trait in high-dimensional applications where not all features are available at once, like spam filtering.

Unfortunately, the limitation of PA models is also notable. A major drawback is that they fail to address unobserved, old features. If the majority of existing features become missing, PA learners can only rely on the rest features for making predictions, likely leading to inferior results. This issue is further exacerbated by their sparsity steps. Consider a set of informative features, with large weights, that become unobservable for a long time span. The sparsity solution will never recur, their weights will persist in the learner with in-
creasingly large values, negatively impacting prediction performance and memory efficiency. As more such features exist, the UOL process is ill-conditioned.

**Application Scenarios.** The usability of PA models have been widely demonstrated through their applications in a variety of fields, such as Internet of Things (IoT) [Pishgoo et al., 2022], epidemics [Kimura et al., 2022], document categorization [Xia et al., 2017], photovoltaic harvesting [Yu et al., 2021], and many more. In these applications, the feature space is constantly evolving with new features being added, while existing features remain part of the online learning process. For example, each word can be considered a feature for document categorization, forming a vast vocabulary. As natural, societal, and technological changes occur, new words are continuously coined, resulting in an increase of 70% in the English language’s vocabulary over the past eighty years. By using PA methods, online document classifiers can be trained to keep up with the ever-changing language landscape.

### 3.2 Feature Correlation (FC) Models

Unlike the PA models that focus on initializing new features, FC models mainly deal with unobserved old features, striving to infer and reconstruct missing information in order to remedy discriminant power loss. Feature correlation is essential for this reconstruction process; prior FC studies differ in their methods to capture and model these correlation structures. Denoted by $\mathcal{U}_t := \bigcup_{t=1}^T \mathcal{X}_t$, a universal feature space that records all emerged features up to $t$. The goal of FC models is to find a mapping $\phi : \mathcal{X}_t \mapsto \mathcal{U}_t$, which captures feature correlation and enables missing feature reconstruction. To maintain notational symmetry and succinctness, we write $\phi(x_t) = [\tilde{x}_t, \hat{x}_t]^\top \in \mathcal{U}_t$, where $\tilde{x}_t = \Pi_{\phi(x_t)}$ denotes the representation of observed features in $\mathcal{U}_t$, and $x_t \in \mathcal{U}_t \setminus \mathcal{X}_t$ denotes the reconstructed unobserved features.

A general objective of FC models is formulated as follows.

$$
\min_{h_t, \phi} \sum_{t=1}^T \ell(y_t, h_t(\phi(x_t))) + \alpha \Omega_1(h_t) + \beta \Omega_2(\phi)
$$

where the learner $h_t$ and mapping $\phi$ are jointly trained for empirical risk minimization, which encourages a positive synergy between them. In particular, the learner must become increasingly more discriminative as it is trained on a more informative $\mathcal{U}_t$ space. Regularizers $\Omega_1$ and $\Omega_2$ are imposed on $h_t$ and $\phi$, respectively, during the online process, while two positive parameters $\alpha$ and $\beta$ absorb different scales among the three terms. The constraint $d(\tilde{x}_t, x_t) \leq \epsilon$ ensures that the observed feature information is not washed-out but accurately recovered after reconstruction, as the distance between $\tilde{x}_t$ and $x_t$ is bounded within a certain tolerance $\epsilon$.

**Learner and its regularizer.** Linear classifiers are leveraged in pioneer studies [Hou et al., 2017; He et al., 2019; He et al., 2021b; He et al., 2021a; Hou et al., 2021b]. Online kernel machines can be easily extended from linear classifiers for non-linear cases [Hou et al., 2021a]. However, when the size of $\mathcal{U}_t$ increases, the dimension of linear and kernel learners grows linearly and exponentially, respectively. To promote sample efficiency, an $\ell_1$-norm regularizer is often applied to yield sparse solutions and bound the maximum dimension through feature pruning. Most research on UOL has been conducted under a fully supervised setting, where labels are abundant. Very recently, online semi-supervised learners have been developed to reduce the labeling expenditure [He et al., 2021c; Wu et al., 2023]. They do so by incorporating regularizers that respect certain kinds of geometric structure underlying the data, such as manifold [He et al., 2021c]. This structure enables data instances with similar labels to be placed in neighboring regions, while other dissimilar instances are expelled. By propagating the scarce labeling information within local neighborhoods, pseudo labels are generated to enable efficient UOL.

**Mapping and its regularizer.** Linear mapping which postulates a linear relationship among feature coefficients has pioneered [Lou et al., 2013; Weld and Bansal, 2019]. In UOL contexts, the linear mappings boil down to multivariate regressor [Hou et al., 2017] or mean field [He et al., 2019; He et al., 2021c], which lends high interpretability and analytical tractability. Despite so, the linear assumption can be stretched by the complexity and nonlinearity of real streaming data, particularly in high-dimensional spaces such as texts and images. In response, nonlinear mappings including Gaussian copula [He et al., 2021a] generative-adversarial network [Zhang et al., 2020], and variational auto-encoder [Lian et al., 2022], have been proposed, which can capture more complex and latent feature interplays. However, these mappings require reparameterization, leading to a search space that is orders of magnitude larger than that of linear mappings. To address this tradeoff between learning effectiveness and efficiency, [Lian et al., 2022] proposed a regularizer to learn an optimal depth of representation: it freezes the deep layers of a neural network at early rounds to yield shallow representations for faster convergence, and only moves to deeper layers if more complex feature correlation is required for better learning performance.

**Distance Metric.** The distance $d(\tilde{x}_t, x_t)$ is measured in accordance to the learner and mapping architectures. In cases both are linear, the measurement is incidental, where Euclidean distance often suffices [Hou et al., 2017; He et al., 2019]. It becomes prickly when the learned $\mathcal{U}_t$ space is highly nonlinear, which requires to tailor distance metrics ad hoc. For example, geodesic distance is used in [He et al., 2021c] to respect the manifold structure in $\mathcal{U}_t$, sparsified by a random-project tree [Freund et al., 2008] to avoid memory overhead. In [He et al., 2021a], a Maximize a Posterior (MAP) surrogate was employed for distance minimization, which was integrated in an online Expectation-Maximization process to estimate the parameters of Gaussian copula. In [Lian et al., 2022], KL-divergence was used to measure the distance between two latent distributions, aiming to induce low-rank variational Bayes from the learned $\mathcal{U}_t$.

**Pros & Cons.** Unlike PA models which are purely discriminative, the FC models possess a “generative” capability to reconstruct missing features. This generative learning property has three remarkable benefits. 1) The universal space $\mathcal{U}_t$ captures feature correlations, making the model intelli-
ble [Weld and Bansal, 2019]. The fact that $\phi$ is bijective, where each feature emerged in $X_t$ has exactly one representation in $U_t$, enables to trace back the feature importance with domain knowledge. 2) The design of learner $h_t$ and mapping $\phi$ is decoupled, which makes them flexible enough to accommodate diverse data modalities and learning algorithms. To wit, the mapping can be tailored with various feature extractors such as convolution filters, wavelet transforms, and pretrained embeddings to deal with images, time series, and texts, respectively. Learners can range from linear classifiers to SVMs to deep neural networks, balancing accuracy and interpretability. 3) FC models have theoretical guarantees. With $h_t$ and $\phi$ both convex, Eq. (3) delivers a biconvex program [He et al., 2019], which can be optimized via ADMM. Sub-linear regret bound has been reported in most prior studies, placing them among the no-regret online algorithms [Hou et al., 2017; He et al., 2021a].

Nevertheless, there are also some drawbacks to FC models. 1) There is no explicit treatment for new features. A buffer is needed to learn the correlation between new and existing features in minibatch; otherwise, the mapping is learned from scratch and introduces noise. Even if the new features’ correlation with other features is provided by a domain expert, it remains unclear how to incorporate this prior knowledge into the learning process. 2) It is difficult to bound the dimension of $U_t$, $\ell_1$ regularizer is more of a heuristic workaround than a robust solution. The fact that FC models are not step-size free and require a careful setup of the step decay rate makes it challenging to single out less informative and redundant features. With new features inputting in different orders, the resultant $U_t$ can retain completely different features. There has been no research yet on how to stabilize the sparse solution of $U_t$ when the feature space varies wildly. 3) Resilience to feature drift is restricted in gradual settings. Old features remain observable through reconstruction, and their learned coefficients continue to influence decision-making at all rounds. If the distribution of an unobserved feature evolves gradually, its coefficient can be updated with incurred loss, conveying a negative effect of drift. FC models also provide an effective solution for reconstructing the unobserved feature information, thus enabling more accurate online predictive modeling.

### 3.3 Evolutionary Ensemble (EE) Models

The groundwork for applying ensemble methods to a continuously evolving feature space was laid by [Wenerstrom and Giraud-Carrier, 2006], other types of methods were more popular in the interim. Recently, however, Evolutionary Ensemble methods have once again gained some traction [Schreckenberger et al., 2023]. Unlike FA and FC models that maintain a single model, EE models maintain multiple submodels (constructed using weak learners) corresponding to a subset of the observed feature space. The conceptual framework shared by these approaches includes three steps: the initialization of multiple weak learners, the repeated updating of these weak learners, and their combination to make a final prediction. The goal of EE methods is to find the optimal weight associated with each weak learner, taking a generalized additive [Lafferty, 1999; Lou et al., 2012] form:

$$\hat{y}_t = \argmax_{c \in C} \sum_{i=1}^{l} w_i \cdot L_i(x_t),$$

where $w_i$ is the associated weight of a weak learner $L_i$.

**Initialization.** To initialize weak learners from the observed feature space, it is required to either track instances or at least approximate their statistics. For ensemble methods that use an OPL capable weak learner, the respective learner can be updated with every matching instance from the feature space [Wenerstrom and Giraud-Carrier, 2006]. However, if the ensemble consists of offline weak learners, it is possible to approximate the respective feature statistics and generate weak learners from them [Schreckenberger et al., 2023].

**Update.** Generally, there are three components in EE that require updating: the weak learners themselves, weights associated with the weak learners, and the membership of weak learners. For ensemble methods that use online learning capable weak learners, the update strategy is straightforwardly by simply updating the weak learner [Wenerstrom and Giraud-Carrier, 2006]. Methods that track features to generate offline weak learners require more effort, as decisions must be made regarding when and how to update the weak learner [Schreckenberger et al., 2023]. A common pattern observed in EE methods is that the weights associated with the weak learners are updated based on the individual performance of the weak learner. The membership of the individual weak learners in the ensemble is managed by either age [Wenerstrom and Giraud-Carrier, 2006] or observability of the respective feature [Schreckenberger et al., 2023].

**Pros & Cons.** Ensemble methods are widely known for their robustness, making them a suitable choice for UOL problems as recent developments in EE have demonstrated. Vanishing or obsoleting features can be easily addressed by removing the associated weak learners, and new ensemble members can be added at any time if necessary. While maintaining multiple weak learners in an ensemble may lead to
increased space and time complexity when compared to PA or FC methods, it has been shown in [Schreckenberger et al., 2023] that these complexities can scale linearly with the growth of the feature space $O(\|\mathcal{U}_t\|)$.

Unobserved features can become particularly problematic when dealing with complex weak learners, as those learned from multiple features require all of them to be present simultaneously. This necessitates a trade-off between simple and less discriminative weak learners involving fewer features, and complex discriminative weak learners that are likely to be inapplicable more frequently.

**Application Scenarios.** UOL has potential application scenarios in document classification [Wenerstrom and Giraud-Carrier, 2006] and crowd-sensing networks [Schreckenberger et al., 2023] in which the feature spaces may be highly variable due to unreliable deployment of sensors by third parties. While this could be seen as a drawback, it can also be advantageous, as it increases the resolution of how the real world is captured by the system. Methods that dynamically update their composition and attribute importance to the observable features are essential in these volatile environments.

4 Evaluation

4.1 Real Datasets

Whereas existing UOL studies mostly use synthetic datasets, we argue that real-world datasets are a must to gain wider acceptance in the promotion of UOL research. To this end, we present four real datasets, which span various domains including natural language, smart (crowd) sensing, cybersecurity, and healthcare. We hope to stimulate further research by elaborating on how the feature space dynamism is manifested in real applications. Astute readers can map their research projects onto the scenarios from which the presented datasets are generated, thus writing a new chapter in UOL research.

1) *imdb* [Maas et al., 2011] is used for sentiment analysis based on movie reviews, which are represented as bags of words - counts of each word in a review. As language evolves, the vocabulary used in these reviews can expand; for example, due to new actors being mentioned. However, old features may disappear over time, such as when actors retire and cease to be mentioned in future reviews.

2) *crowdsense* [Schreckenberger et al., 2023] is collected from a crowd-sensing network, where the features are created from a variety of environmental sensors (e.g., sound pressure, eCO2 level, and eTVOC level) scattered across the 56 biggest cities in Spain. The feature space is ever-changing as new sensing data is continuously being generated while many old sensors cease to provide data. The learning task is to predict the government’s restriction severity based on the sensed crowdedness of the regions.

3) *naticusdroid* [Mathur et al., 2021] is tasked with the detection of mobile malware in Android devices. It uses permission flags, such as those for accessing the camera, as features. However, due to the ever-evolving Android OS versions, various manufacturers, and different device models and functionalities, these permission flags may vary over time. The goal is to identify malicious applications based on their required permissions.

4) *diabetes* [Strack et al., 2014] dataset documents a major concern in healthcare, with patients often returning to the hospital within 30 days of completing a treatment program [Strack et al., 2014]. This dataset presents a unique challenge due to its varying feature space, which is a result of different drugs administered and missing characteristics such as age or weight.

4.2 Evaluation Protocol

Prior UOL studies impose different modeling assumptions on feature space dynamics. PA models postulate trapezoidal data streams (e.g., [Zhang et al., 2016]), where all emerged features will be sustained in later rounds, leading to a monotonically increasing feature space. Some of the FC models assume feature evolvable streams (e.g., [Hou et al., 2017]), where an overlapping time slot in which both old and new features are concurrently available is required before the former turn to unobserved. We argue that these two assumptions merely lead to special cases of open feature space. The corresponding solutions tailored based on the assumptions are part of the general UOL methods. To progress further, UOL research should focus on the least constrained version of feature dynamics. As such, we benchmark six state-of-the-art UOL methods in the most general setting of open feature space.

**Metric.** On each dataset, the instances are presented to model in a one-pass fashion. The accuracy of any algorithm is gauged by cumulative error rate: $CER = \frac{1}{T} \sum_{t=1}^{T} \left[ y^*_t \neq \hat{y}_t \right]$, where $T$ equates to the number of samples in the dataset, and $\left[ \cdot \right]$ counts one if its argument is true and zero otherwise.

**Performance Analysis.** We analyze the performance of two representative UOL models from each category (i.e., PA, FC, and EE). The results are documented in Table 1, which reveals that the model performance has been steadily improving over time, indicating a thriving yet competitive status quo of UOL research. Of the models evaluated, ORF3V, published in 2023, holds the lowest CER on average at 12.23%.

Upon closer inspection, however, some exceptional data points can be observed. For instance, the low CERs attained by FAE and ORF3V on *diabetes* may be attributed to an unfair metric, as the dataset holds an imbalanced ratio of 0.111 between positive and negative instances. This suggests that ORF3V has yielded a highly biased classifier by predicting all instances into the majority class, thus learning nothing. In addition, OVFM runs out of memory on *imdb*, highlighting the drawback of FC models discussed in Section 3.2: their high memory consumption for recording all emerged features.

Indeed, the overall dimension of *imdb* is 7500, which is larger than the other three datasets by one order of magnitude. Without an effective sparsification mechanism, the feature reconstruction mapping becomes complex and intractable.

These findings bring to light several open challenges in UOL research, which will be discussed in the next section.

5 Open Challenges and Future Directions

**Label Imbalance, Scarcity, and Noise.** Binary classification is the default setting in current UOL research due to its simplicity, however, extending it to other setups is non-trivial but vital. First, our study of the *diabetes* dataset revealed that no previous studies have considered the imbalanced learning...
Concept Drift. Concept drift occurs when the distribution of features, either conditioned on class or not, evolves over time, although the set of features itself does not change [Hu et al., 2020]. Unfortunately, UOL does not explicitly consider drifting concepts, despite the fact that EE models refresh their weak learners on a regular basis and thus possess certain resilience to it. Tailoring methods to detect and adapt to concept drift is difficult in a UOL context, as existing distribution estimators are mainly parametric, requiring a fixed feature set. To wit, a Bayesian estimator will have its density exceeding 1 if new random variables emerge halfway. In other words, the core law of total probability cannot be applied unless one presumes the probability of all unseen features is known.

Open-world Crisis. When evaluating traditional learning systems pointwise with respect to a fixed test set, its static coverage is limited in terms of assuring safety and resilience to “unknown unknowns” in high-stakes operating environments [Hendrycks et al., 2021]. To address this challenge, UOL tackled the input end, leaving the output end to a different track of research, coined open-world learning (OWL) [Boul et al., 2019], which focuses on new and unseen labels that can emerge during a learning continuum. By combining UOL and OWL, a highly flexible computing paradigm is enabled, one which does not impose any assumptions on either the input or output of a learning system, thereby improving methods for monitoring unexpected environmental hazards. However, this is challenging, as most existing OWL methods require gauging and minimizing the volume of the region each known class spans, so that new classes can be distinguished as out-of-region samples. This idea cannot work once the feature space changes, as each known class can encompass instances of different feature sets.

Security & privacy. Current UOL models compute all instances on central servers, which may cause security and privacy (S&P) implications. Distributed UOL provides an alternative approach that divides the computing endeavor into multiple parties, each of whom holds their own data and does not share with others. This distributed UOL can be further divided into two research thrusts: 1) UOL for S&P: the current S&P community is troubled by the fact that all data parties have non-aligned feature sets (e.g., vertical federated learning [Liu et al., 2022a]). Rather than enforcing a protocol or consensus to make them share a common feature subset, UOL provides a more flexible and powerful learning paradigm that allows any new party to join and introduce new features. 2) S&P for UOL: distribution does not automatically guarantee Security and Privacy. UOL will require carefully designed protocols to ensure that original data cannot be reconstructed by a honest but curious party. This may involve stipulating communication among parties and/or applying obfuscation or encryption on gradients.

Algorithmic Fairness. A superficial correlation between label and new features that convey protected information, such as gender, ethnicity, occupation, or even zipcode, can easily lead to online learners with unwanted bias [Barocas et al., 2017]. Surprisingly, the absence of certain features can also cause bias; for instance, in crowdsensing applications, a lack of access to technology, including sensors and smartphones, could be indicative of spatial injustice [Hino et al., 2018]. To promote algorithmic fairness in UOL model, it is essential to ensure that superficial correlation nor feature observability will result in disadvantaged prediction groups.

6 Conclusion

In this survey, we presented the concept of Utilitarian Online Learning (UOL), a unified framework for online learning in open feature spaces. We taxonomized the state-of-the-art models into three categories and outlined their generic objective functions. We discussed the pros and cons of each category, conducted benchmarking evaluations, and highlighted open challenges. UOL is an area with more unknowns than knowns. We hope that our survey can provide researchers with the most up-to-date knowledge, particularly those striving to tame streaming data from non-static and high-stakes environments, and will shed some light on the future.

“Intelligence can be observed to grow and evolve . . . through accumulation of knowledge of how to sense, decide, and act in a complex and changing world.” - James S. Albus, Outline for a Theory of Intelligence (1991)

Acknowledgements

This work was supported in part by the Commonwealth Cyber Initiative (CCI), the National Natural Science Foundation.
of China (NSFC) under Grant No. 62120106008, and the German Federal Ministry for Housing, Urban Development and Building under Grant No. 13622847.

References


[Maas et al., 2011] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *ACL*, pages 142–150, June 2011.


