INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

ICPP 2021

50th International Conference on Parallel Processing

August 9-12, 2021

Chicago (Argonne National Lab), Illinois, USA

acm In-Cooperation sighpc

ASLDP: An Active Semi-supervised Learning method for Disk Failure Prediction

Yang Zhou, Fang Wang†, Dan Feng wangfang@mail.hust.edu.cn (†Corresponding author)

Division of Optoelectronic Information Storage, Wuhan National Laboratory for Optoelectronics

Key Laboratory of Information Storage System, Ministry of Education

Huazhong University of Science and Technology

Wuhan, China

武汉光电国家研究中心 WUHAN NATIONAL LABORATORY FOR OPTOELECTRONICS

信息存储系统教育部重点实验室 Key Laboratory of Information Storage System Ministry of Education of China

華中科技大學 HUAZHONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Introduction

- Disk failure has become a common event
  - About 80% of system breakdowns are caused by hard drive failures in the data center [ToS' 18 Gunawi *et al.*, DAC' 19 Sun *et al.*]
- Almost All methods are limited to *offline supervised learning*
  - Need to prepare a larger amount of *labeled* data [FAST' 20 Lu *et al.*]
  - Need to be retrained to accommodate the new datasets [ATC' 20 Zhang *et al.*]
- A large number of *unlabeled* data may be available in reality
  - Need to be properly labeled [ICDCS' 20 Han *et al.*]
- Modern data center is a complex environment
  - Difficult to meet the needs of supervised learning [Google 16 Eric *et al.*]

Traditional machine learning methods

Supervised learning 😐

Neural network + Deep learning

High resource cost ☹

Background

Disk Failure Prediction

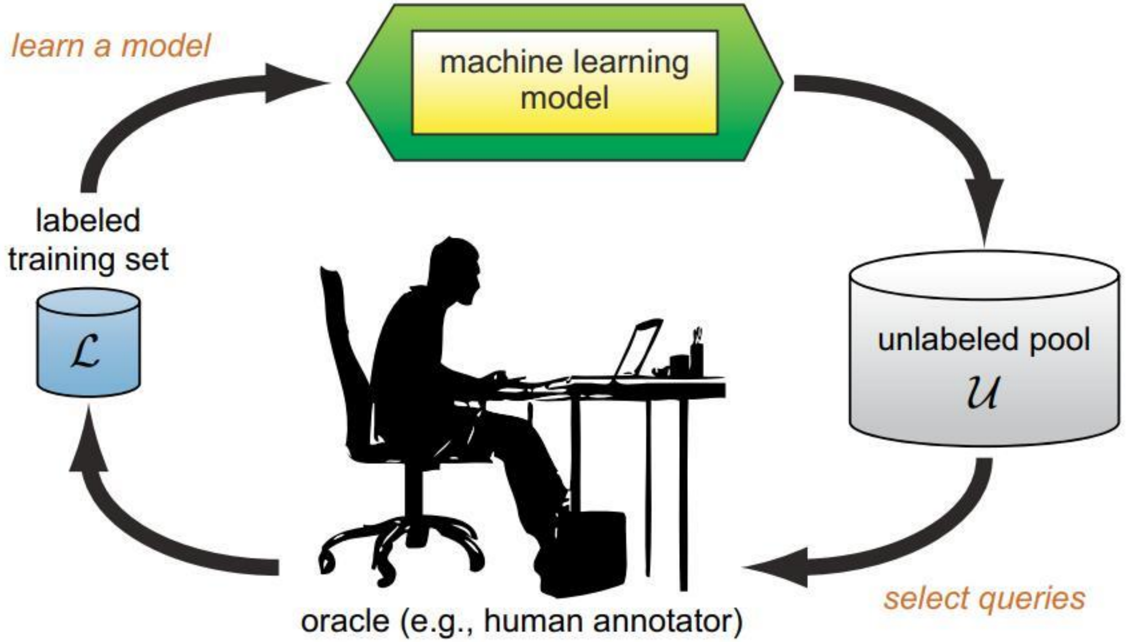
- For datacenter
  - Failure is an ordinary event rather than an accident
  - Storage dive failure is the principal failure
- The advantages of failure prediction
  - More time for failure handling
  - Copy the existing data rather than recover the lost
  - ……

Background

Active Learning

- The active learning algorithm selects the best samples by some strategies from the sample pool, and adds these samples to the training set of the model, so that the trained classifier can obtain strong generalization ability
- There are usually two selection strategies for its samples: *stream-based* and *pool-based*
- The key component of active learning is the design of an effective criterion for selecting the most "valuable" instance to query, which is often referred to as *query strategy*
- In general, different strategies follow a greedy framework:

$$S^* = \underset{s \in D_u}{\operatorname{argmax}} \min_{y \in \{0,1\}} f(s; y, D_l)$$



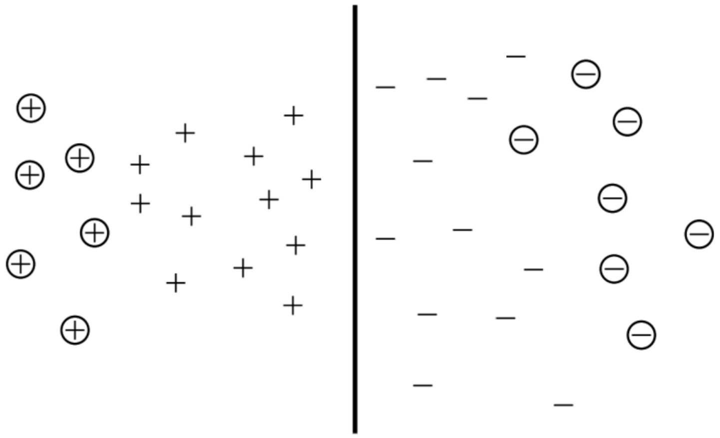
The *pool-based* active learning cycle

Settles, B. (2009). Active learning literature survey.

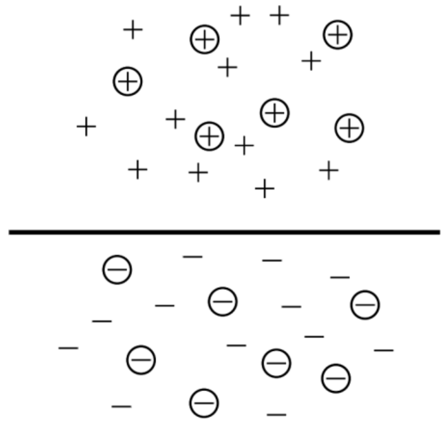
Background

Semi-supervised Learning

- The key to semi-supervised learning is how to use a large amount of cheap unlabeled data to improve the performance of the classifier
- The existing semi-supervised learning methods include *Generative model*, *Graph-based*, *Support vector machines*, and so on
- *Co-training* assumes that features can be split into two sets; Each sub-feature set is sufficient to train a good classifier; The two sets are conditionally independent given the class



(a) x^1 view



(b) x^2 view

Co-training: Conditional independent assumption on feature split

Blum, A., & Mitchell, T. (1998, July). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory* (pp. 92-100).

Datasets

- Totals 54 months from 2016 - 2020, which is the largest and most recent data currently used in Backblaze [1]
- Baidu [2] dataset is only used as a supplement to the experiment

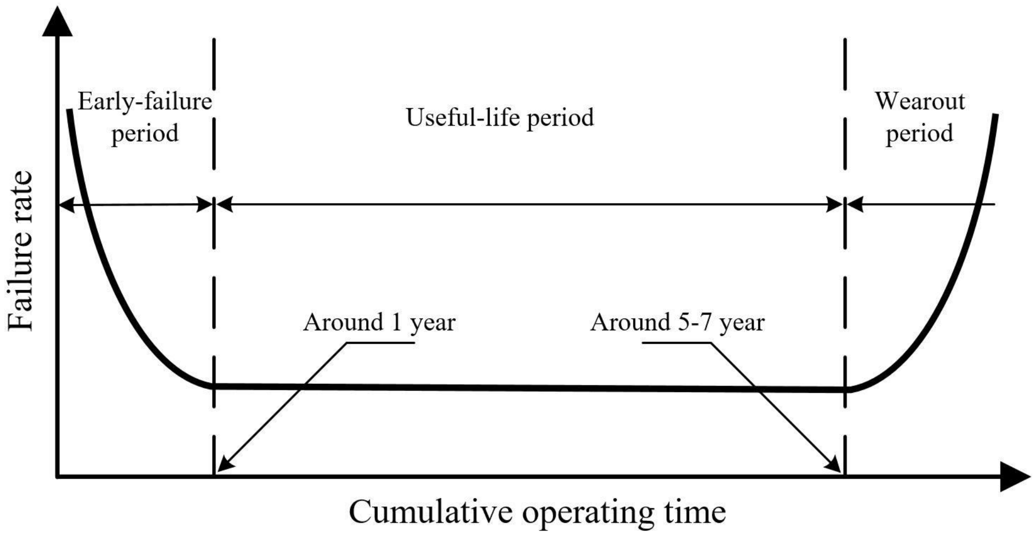
Dataset		Model	Capacity bytes	Healthy	Failed	Interval
Backblaze	D1	ST12000NM0007	12000138625024	36990	1702	1 day
	D2	ST4000DM000	4000787030016	33054	3104	1 day
	D3	ST8000NM0055	8001563222016	14470	521	1 day
Baidu	D4	-	-	22962	433	1 hour

1. https://www.backblaze.com/b2/hard-drive-test-data.html
2. http://pan.baidu.com/share/link?shareid=189977&uk=4278294944

Motivation

Why we use Active Learning

- the disk failure rate follows the bathtub curve

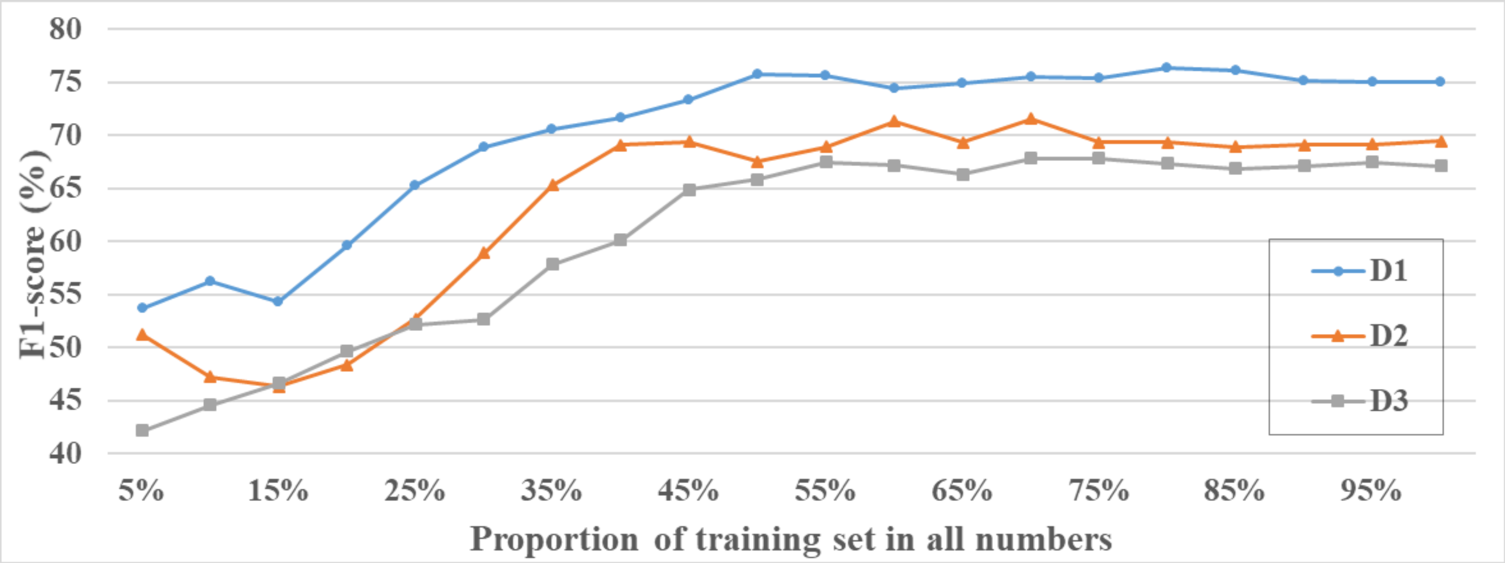


Field failure rate pattern of HDD

Yang, J., & Sun, F. B. (1999, January). A comprehensive review of hard-disk drive reliability. In *Annual Reliability and Maintainability. Symposium. 1999 Proceedings (Cat. No. 99CH36283)* (pp. 403-409). IEEE.

- *under-sampling* [ICPP' 19 Zhang *et al.*]
  - relatively simple
  - lose some useful samples
- *clustering* [SIGKDD' 16 Botezatu *et al.*]
  - take a lot of time to complete
- All the methods are completed in the data preprocessing stage without the join of the classifier

Why we use Active Learning

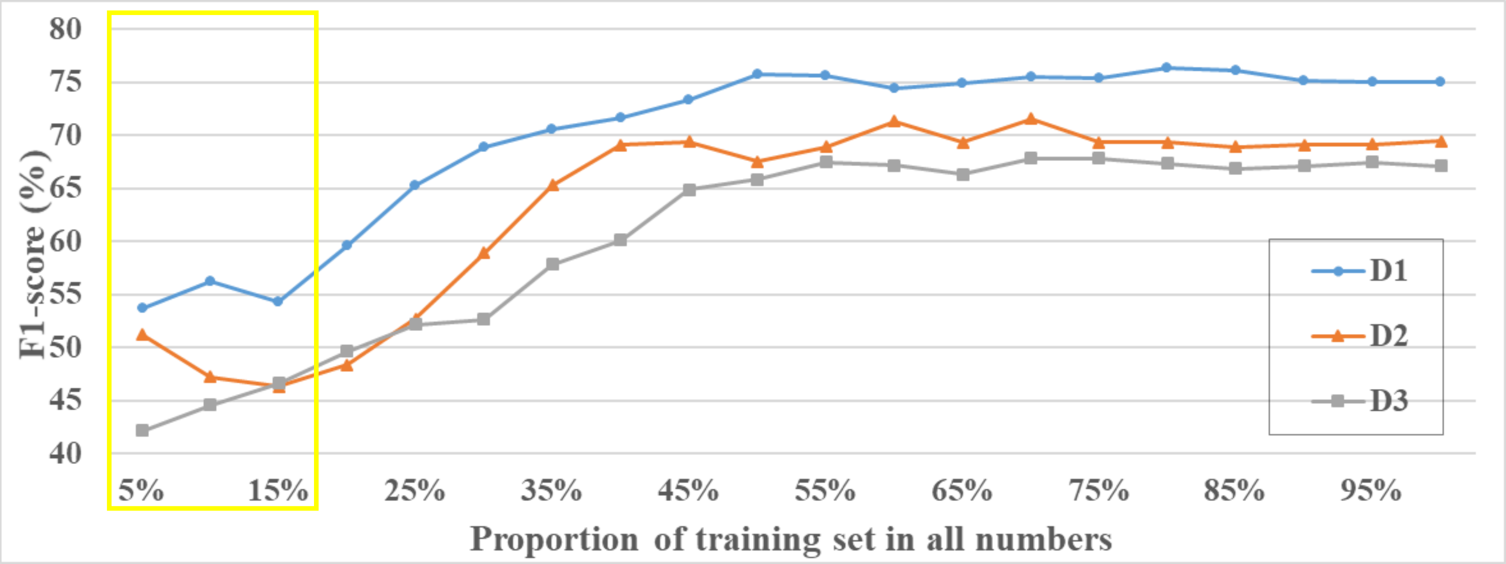


Effects of training sets of different scales on prediction results

- There are more redundant samples in the Backblaze dataset
- Active learning can achieve similar results with fewer samples when the training set is limited
- SMART data on disk is not only redundant, but sometimes has a negative effect on prediction

Motivation

Why we use Active Learning

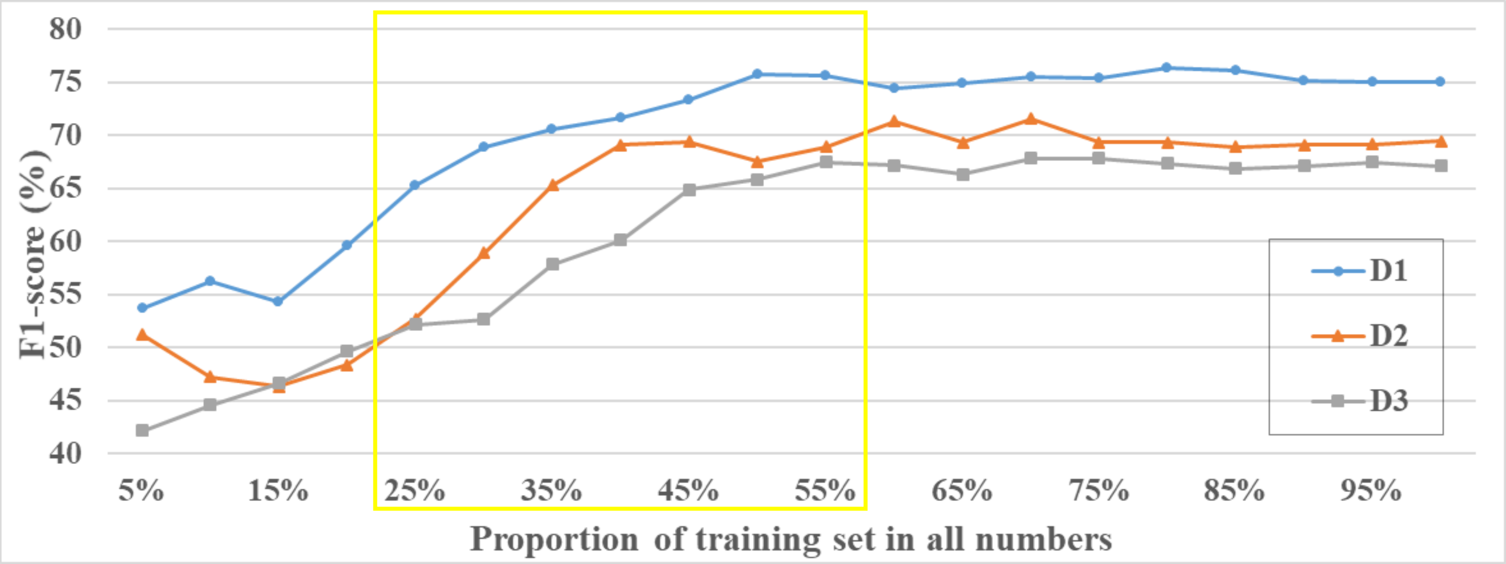


Effects of training sets of different scales on prediction results

- There are more redundant samples in the Backblaze dataset
- Active learning can achieve similar results with fewer samples when the training set is limited
- SMART data on disk is not only redundant, but sometimes has a negative effect on prediction

Motivation

Why we use Active Learning

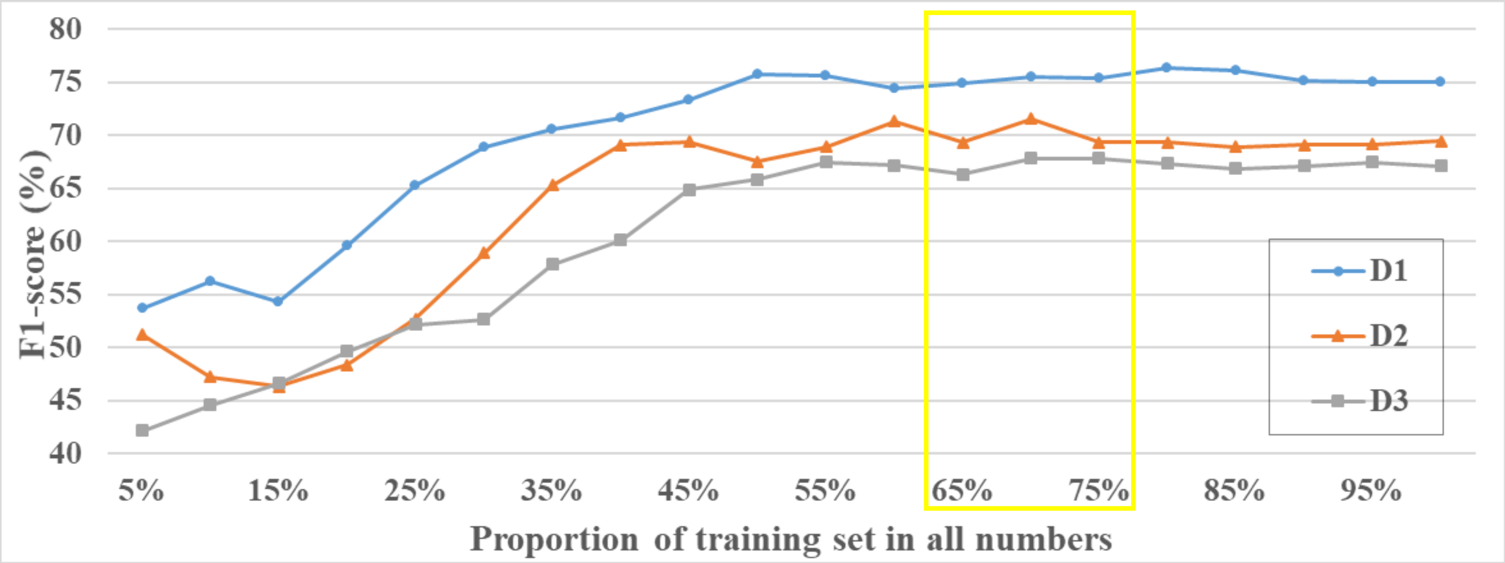


Effects of training sets of different scales on prediction results

- There are more redundant samples in the Backblaze dataset
- Active learning can achieve similar results with fewer samples when the training set is limited
- SMART data on disk is not only redundant, but sometimes has a negative effect on prediction

Motivation

Why we use Active Learning

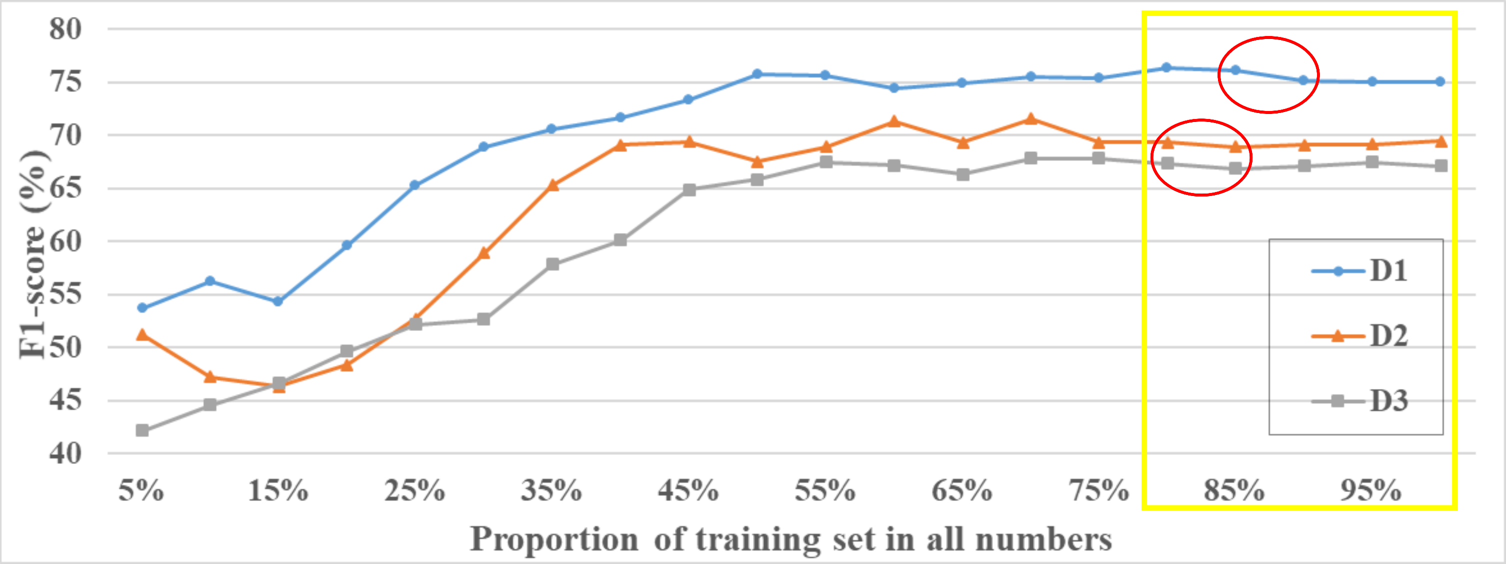


Effects of training sets of different scales on prediction results

- There are more redundant samples in the Backblaze dataset
- Active learning can achieve similar results with fewer samples when the training set is limited
- SMART data on disk is not only redundant, but sometimes has a negative effect on prediction

Motivation

Why we use Active Learning



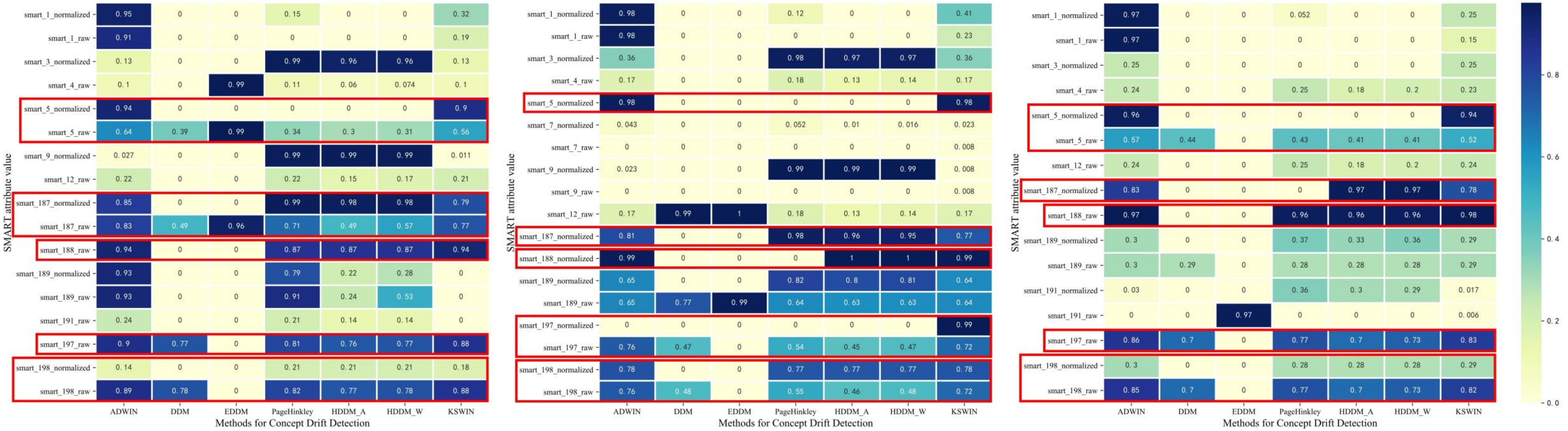
Effects of training sets of different scales on prediction results

- There are more redundant samples in the Backblaze dataset
- Active learning can achieve similar results with fewer samples when the training set is limited
- SMART data on disk is not only redundant, but sometimes has a negative effect on prediction

Motivation

Why we use Semi-supervised Learning

- In the real-world scenarios, it is difficult to obtain accurately labeled values for every moment on the disk
- Research has shown that a soon-to-fail disk has actually shown failure symptoms, and therefore all of the samples before the failure are labeled as failure samples [ICPP' 18 Xiao *et al.*, ICPP' 19 Zhang *et al.*, ICDCS' 20 Han *et al.*]

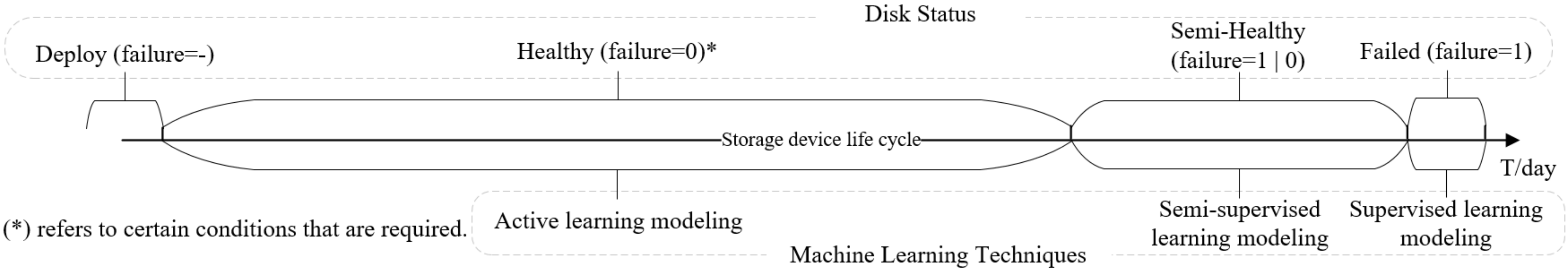


- there is no obvious change in SMART data for many failure samples before the failure actually occurs

Motivation

Why we use it together

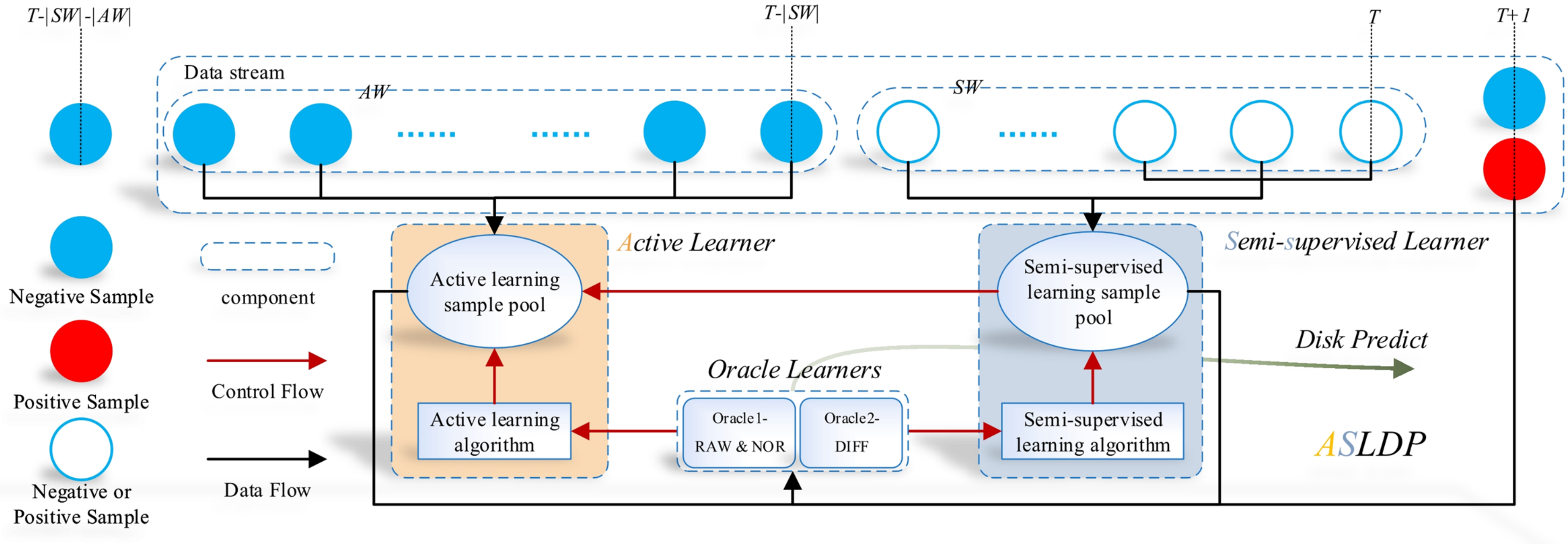
- By combining the Active Learning & Semi-supervised Learning in different periods, the learner can select the appropriate samples and improve the classification ability



Disk life cycle analysis

- We do not emphasize that this combination is the best method.
- We just give an idea that can replace supervised learning in disk failure prediction.

Architectural Overview of ASLDP



ASLDP: Active Semi-supervised Learning Disk-failure Prediction model

Design

How to use Active Learning

Algorithm 1 Active learning sample selection

Input: Active learning sample pool: $D_{ALP} = \langle(\vec{x}_1, y_1 = 0), (\vec{x}_2, y_2 = 0), ..., (\vec{x}_{|AW|}, y_{|AW|} = 0)\rangle$
Input: Classifiers: Ora_1, Ora_2
Input: Number of samples: n_1, n_2
Input: Probability threshold: α_{min}, α_{max}
Output: Selected examples set: Sa_1, Sa_2

1: $P_1 = Sorted(Ora_1(D_{ALP}), seq = desc)$
2: **for** p_i *in* P_1 **do**
3:     /* i is an index of sample with value p in D_{ALP} */
4:     **if** $\alpha_{min} \leq p_i$ && $p_i \geq \alpha_{max}$ && $n_1 > 0$ **then**
5:         $Sa_1.add((\vec{x}_i, y_i = 0))$
6:         $n_1 = n_1 - 1$
7:     **end if**
8: **end for**
9: $P_2 = Sorted(Ora_2(D_{ALP}), seq = desc)$
10: **for** p_i *in* P_2 **do**
11:     **if** $\alpha_{min} \leq p_i$ && $p_i \geq \alpha_{max}$ && $n_2 > 0$ **then**
12:         $Sa_2.add((\vec{x}_i, y_i = 0))$
13:         $n_2 = n_2 - 1$
14:     **end if**
15: **end for**

- Active learning algorithms here will involve two learners (***Ora*$_1$**, ***Ora*$_2$**) who provide predictive services in the future
- The number of active learning to select samples (***n*$_1$**, ***n*$_2$**) is mainly determined by semi-supervised learning
- Active learning query strategy in pseudo-code chooses a *probability-based* heuristic method that picks up the negative samples (Lines 4 & 11) with the greatest uncertainty

Design

How to use Semi-supervised Learning

- The conditional value ***CV(x)*** for the samples under the two views

$$CV_1(x) = \boxed{H_1(x)} \cdot \boxed{Ds_1(x)} \cdot \boxed{\left(1 - H_2(x)\right)}$$

$$CV_2(x) = \boxed{H_2(x)} \cdot \boxed{Ds_2(x)} \cdot \boxed{\left(1 - H_1(x)\right)}$$

- ***H(x)*** represents the entropy value of sample ***x***, The greater the entropy, the greater the uncertainty of the classification

$$H(x) = -\sum_{y \in \{1,0\}} P(y|x) \cdot \log P(y|x)$$

- ***Ds(x)*** represents the regional density of sample ***x***. The smaller distance from other samples, the greater the regional density and the more representative the sample is

$$Ds(x) = e^{-\frac{1}{|D|}\sum_{\vec{x}_i \in D} distance(x, \vec{x}_i)}$$

Algorithm 2 Semi-supervised learning sample selection

Input: Semi-supervised sample pool: $D_{SLP} = \langle(\vec{x}_1, y_1), (\vec{x}_2, y_2), ..., (\vec{x}_{|SW|}, y_{|SW|})\rangle$
Input: Classifiers: Ora_1, Ora_2
Input: Number of samples: m_1, m_2
Input: Probability threshold: β
Output: Selected examples set: Ss_1, Ss_2

initialize: $H_1, H_2 = list(), list()$
initialize: $Ds_1, Ds_2 = list(), list()$
$P_1, P_2 = Ora_1(D_{SLP}), Ora_2(D_{SLP})$
for $i = 1, 2, ..., |D_{SLP}|$ **do**
  /* $|D_{SLP}|$ is the number of D_{SLP}, the same below */
  $H_1.append(-\sum_{y \in 1,0} P_1[i] \cdot \mathbf{log} P_1[i])$
  $H_2.append(-\sum_{y \in 1,0} P_2[i] \cdot \mathbf{log} P_2[i])$
  $Ds_1.append(e^{-\frac{1}{|D_{SLP}|}\sum_{\langle \vec{x}_i, y_i\rangle \in D_{SLP}} distance1(x, \vec{x}_i)})$
  $Ds_2.append(e^{-\frac{1}{|D_{SLP}|}\sum_{\langle \vec{x}_i, y_i\rangle \in D_{SLP}} distance2(x, \vec{x}_i)})$
  /* $distance1$ & 2 use different features of $\vec{x}$ */
end for
$CV_1 = Sorted(H_1 \cdot Ds_1 \cdot (1 - H_2), seq = desc)$
$CV_2 = Sorted(H_2 \cdot Ds_2 \cdot (1 - H_1), seq = desc)$
for cv_i *in* CV_1 **do**
  if $m_1 > 0$ **then**
    $m_1 = m_1 - 1$
    if $P_1[i](y_i = 1) \geq \beta$ **then** $Ss_1.add((\vec{x}_i, y_i = 1))$
    else $Ss_1.add((\vec{x}_i, y_i = 0))$
    end if
  end if
end for
for cv_i *in* CV_2 **do**
  if $m_2 > 0$ **then**
    $m_2 = m_2 - 1$
    if $P_2[i](y_i = 1) \geq \beta$ **then** $Ss_2.add((\vec{x}_i, y_i = 1))$
    else $Ss_2.add((\vec{x}_i, y_i = 0))$
    end if
  end if
end for

Design

How to use it together

- active learning and semi-supervised learning are not completely independent in the algorithm, which mainly solves the imbalance between the positive and negative samples by choosing the number of samples
- In our experiment, the sample ratio $\boldsymbol{\lambda}$ is set to 3, ***sw*** to 15 days, $\boldsymbol{m_1}$ and $\boldsymbol{m_2}$ to 7 days, ***aw*** to 14 days

Algorithm 3 Active Semi-supervised Learning Disk Predict

Input: Data Stream: $DS = \langle ..., (\vec{x}_T, y_T), ...\rangle$; Probability threshold: α_{min}, α_{max}, β
Input: ML Algorithm: $Oracle_1$, $Oracle_2$; Training data set: Tr_1, Tr_2; Learning window size: aw, sw
Input: Sample number of semi-supervised learning: m_1, m_2; Proportion of positive and negative samples: λ
Output: Classifier: Ora_1, Ora_2

initialize: AW, $SW = queues(), queues()$; D_{ASL}, $D_{SLP} = set(), set()$
for $T = 1 \rightarrow |DS|$ **do** /* $|DS|$ is the number of $|DS|$, the same below. */
 if $AW.size() \geq aw$ **then**
  if $\frac{|Tr_1(y=0)|}{|Tr_1(y=1)|} < \lambda$ **then**
   $n_1 = |Tr_1(y=1)| \cdot \lambda - |Tr_1(y=0)|$
  else $n_1 = 0$
  end if
  if $\frac{|Tr_2(y=0)|}{|Tr_2(y=1)|} < \lambda$ **then**
   $n_2 = |Tr_2(y=1)| \cdot \lambda - |Tr_2(y=0)|$
  else $n_2 = 0$
  end if
  while $!AW.empty()$ **do**
   $D_{ALP}.add(AW.pop())$
  end while
  $Sa_1, Sa_2 = Alg1(D_{ALP}, Ora_1, Ora_2, n_1, n_2, \alpha_{min}, \alpha_{max})$ /* Call Algorithm1, Active Learning. */
  $Tr_1.add(Sa_1)$, $Tr_2.add(Sa_2)$; $D_{ALP}.clear()$
  Ora_1, $Ora_2 = Oracle_1(Tr_1)$, $Oracle_2(Tr_2)$
 end if
 if $y_T == 1 \,\&\&\, SW.size() \geq sw$ **then**
  while $!SW.empty()$ **do**
   $D_{SLP}.add(SW.pop())$
  end while
  $Ss_1, Ss_2 = Alg2(D_{SLP}, Ora_1, Ora_2, m_1, m_2, \beta)$ /* Call Algorithm2, Semi-supervised Learning. */
  $Tr_1.add(Ss_1)$, $Tr_2.add(Ss_2)$; $D_{SLP}.clear()$
  $Tr_1.add((\vec{x}_T, y_T))$, $Tr_2.add((\vec{x}_T, y_T))$
  Ora_1, $Ora_2 = Oracle_1(Tr_1)$, $Oracle_2(Tr_2)$
 end if
 if $SW.size() \geq sw$ **then** $AW.push(SW.pop())$
 end if
 $SW.push((x_T, y_T))$
end for

Methodology

SMART Features

- We only use some commonly used SMART features
    - Model can be less influenced by the inability to collect some features, the change of feature importance, and so on
    - We may have no or only few historical disk logs for feature selection, because ***ASLDP*** is positioned as an online learning method
    - Although historical disk logs allow us to identify representative SMART features for failure characterization, the selected features may change over time
- Since our semi-supervised learning requires two-view features, two types of feature views are used here. One is the raw SMART values and the normalized values, the other is the difference between them

Type	#ID	SMART Feature Name	Feature type
Backblaze	001	Raw Read Error Rate	Normalized
	003	Spin Up Time	Normalized
	004	Start / Stop Count	Raw
	005	Reallocated Sectors Count	Normalized & Raw
	007	Seek Error Rate	Normalized
	009	Power-On Time Count	Normalized & Raw
	010	Spin up Retry Count	Raw
	012	Power Cycle Count	Normalized & Raw
	187	Reported Uncorrectable Errors	Normalized & Raw
	188	Command Timeout	Raw
	189	High Fly Writes	Normalized
	191	G-sense error rate	Normalized
	192	Power-Off Retract Count	Raw
	193	Load / Unload Cycle Count	Normalized & Raw
	194	Temperature	Normalized & Raw
	195	Hardware ECC Recovered	Normalized
	197	Current Pending Sector Count	Normalized & Raw
	198	Offline Uncorrectable Sector Count	Normalized & Raw
	241	Total LBAs Written	Raw
	242	Total LBAs Read	Raw
Baidu	001	Raw Read Error Rate	Normalized
	003	Spin Up Time	Normalized
	005	Reallocated Sectors Count	Normalized & Raw
	007	Seek Error Rate	Normalized
	009	Power On Hours	Normalized
	187	Reported Uncorrectable Errors	Normalized
	189	High Fly Writes	Normalized
	194	Temperature Celsius	Normalized
	195	Hardware ECC Recovered	Normalized
	197	Current Pending Sector Count	Normalized & Raw

Methodology

Evaluation Metrics

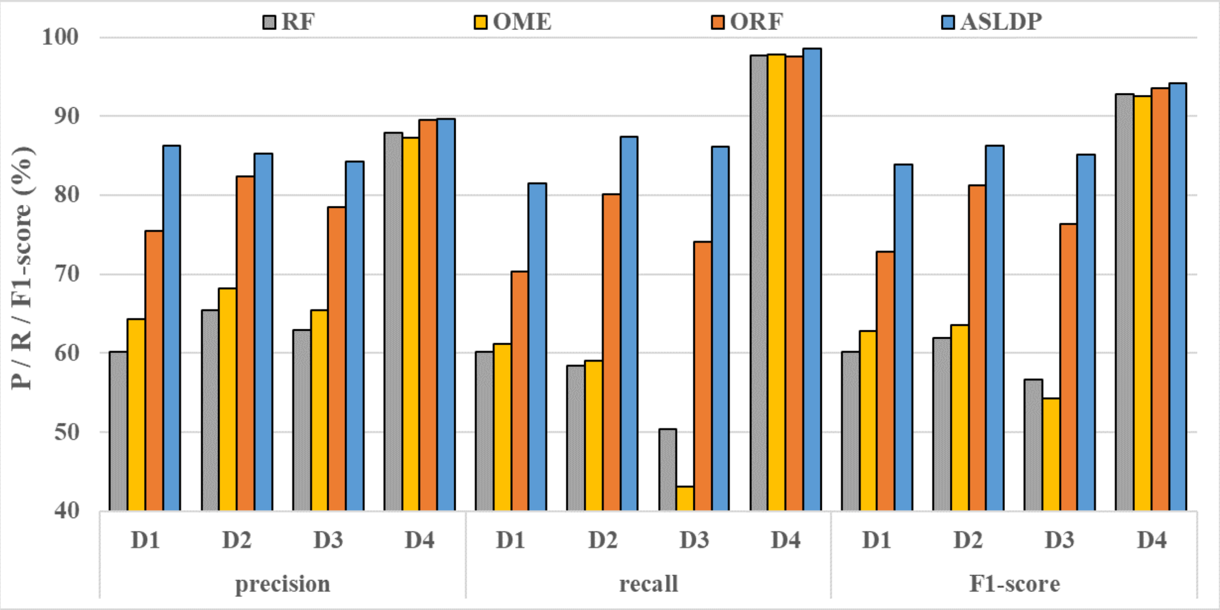
- *Precision* (P)
- *Recall* (R)
- *F1-score*
- Failure Detection Rate (FDR)
- Failure Alarm Rate (FAR)

Testing Methods and Configurations

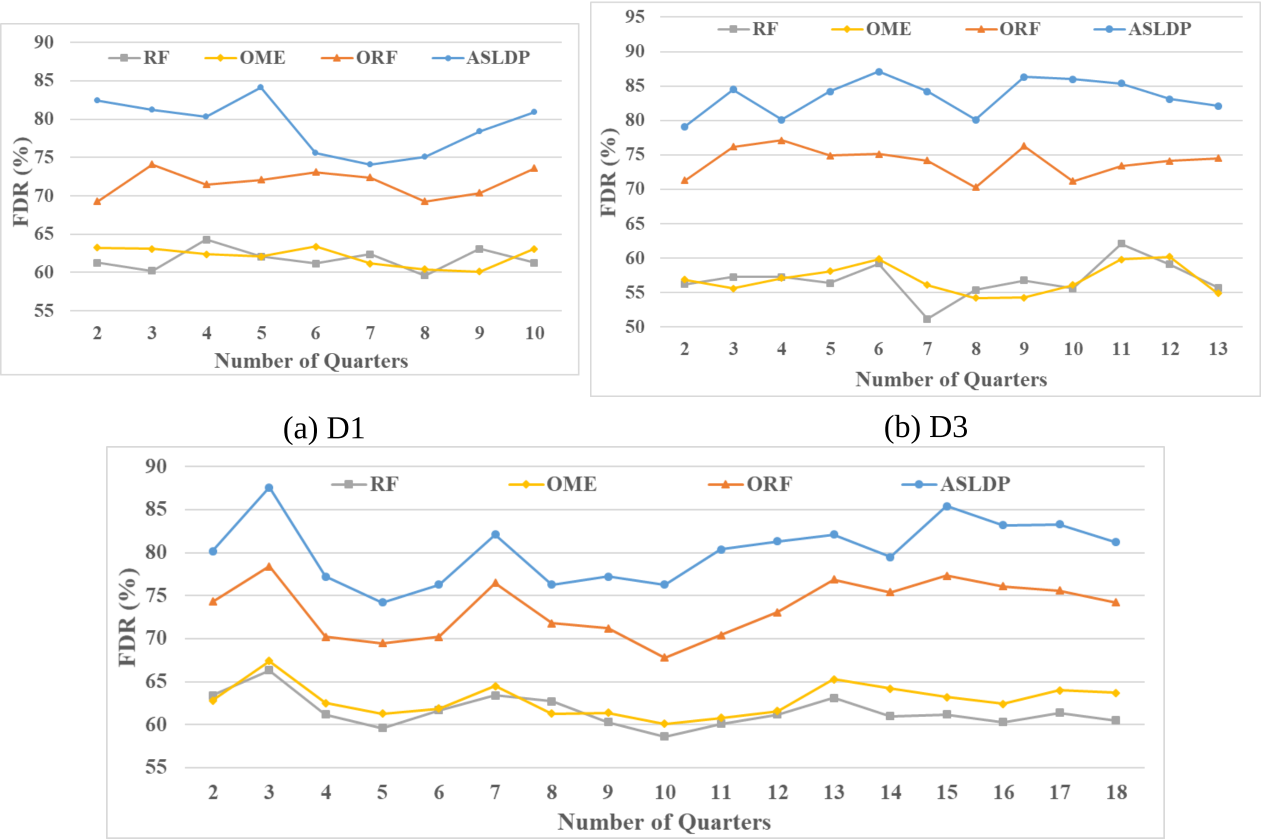
- RF, OME [ICCD' 18 Xie *et al.*], and ORF [ICPP' 18 Xiao *et al.*]
  - all models use decision trees (DT) as base learners

Evaluation

Evaluation of ASLDP Model



Prediction results on D1, D2, D3, and D4 datasets of different models



(a) D1

(b) D3

(c) D2

Simulating practical long-term use by quarter

- Overall, ***ASLDP*** achieves the highest precision, recall, and F1-score among all comparison models and datasets. In the long-term use process, the FDR of ***ASLDP*** can also be controlled within 10%, which realizes online learning and model updating

Conclusion

- We present the complete design of ***ASLDP*** and evaluate our model on datasets from real data center. We demonstrate that ***ASLDP*** has high practicability, better online self-learning (training) & self-turning ability, and can overcome the problems of data labeling and imbalanced datasets
- To the best of our knowledge, we pioneer the use of active learning and semi-supervised learning on disk failure prediction. This is ***very different*** from the supervised learning method used by most previous research work
- By analyzing the characteristics of disk data in different periods, the reasons for using active learning and semi-supervised learning methods and the specific implementation method are given

INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING ICPP/2021/CHICAGO/USA AUGUST 9-12, 2021

Thank You

Question?

zhouyang1024@hust.edu.cn

➢ This work is supported in part by NSFC No.61832020, No.61821003

PhD Candidate @ Huazhong University of Sci.& Tech.