BAMBOO : A Multi-instance Multi-label Approach Towards VDI User Logon Behavior Modeling

Wenping Fan[1] , **Yao Zhang**[2,1] , **Qichen Hao**[1] , **Xinya Wu**[1] , **Min-Ling Zhang**[2,3]*
[1]VMware Information Technology (China) Ltd.
[2]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[3]Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China
{wfan, yaoz, qhao, wxinya}@vmware.com, zhangml@seu.edu.cn (corresponding author)

Abstract

Different to traditional on-premise VDI , the virtual desktops in DaaS (Desktop as a Service) are hosted in public cloud where virtual machines are charged based on usage. Accordingly, an adaptive power management system which can turn off spare virtual machines without sacrificing end user experience is of significant customer value as it can greatly help reduce the running cost. Generally, logon behavior modeling for VDI users serves as the key enabling-technique to fulfill intelligent power management. Prior attempts work by modeling logon behavior in a user-dependent manner with tailored single-instance feature representation, where the strong relationships among pool-sharing VDI users are ignored in the modeling framework. In this paper, a novel formulation towards VDI user logon behavior modeling is proposed by employing the multi-instance multi-label (MIML) techniques. Specifically, each user is grouped with supporting users whose behaviors are jointly modeled in the feature space with multi-instance representation as well as in the output space with multi-label prediction. The resulting MIML formulation is optimized by adapting the popular MIML boosting procedure via balanced error-rate minimization. Experimental studies on real VDI customers' data clearly validate the effectiveness of the proposed MIML-based approach against state-of-the-art VDI user logon behavior modeling techniques.

1 Introduction

Virtual Desktop Infrastructure (VDI) is a technology and software system that can run user desktops on top of a virtual machine hosted on a physical server in the remote datacenter. Recently, the technology of hosting VDI on public cloud is gaining more adoption which is also called DaaS (Desktop as a Service). Compared with on-premise customers, cloud customers care more about the running cost. The public cloud vendors bill for resources usage per minute or even per second. The powered off virtual machines are only charged for storage where the much higher computation cost is saved. As a result, a well-performed power management system can bring significant customer value and competitiveness for a DaaS offering.

VDI systems usually manage virtual desktops in two ways. Non-persistent pool shares desktops among users for cost saving where power management only needs to handle the overall number of desktops. On the contrary, persistent pool allocates dedicated desktop for each user to guarantee the prime user experience which is more difficult for power management because each of the individual desktops needs to be powered on or off specifically for the dedicated user. Consequently, nowadays desktops in the persistent pool are barely powered off which makes the cost much higher especially for cloud customers. The existing VDI power management systems work by configuring power on/off schedules manually. However, manual scheduling is usually time-consuming and error-prone and tends to cause poor user experience and unnecessary cost. To build a proactive VDI power management system which can power-off spare desktops without impacting user experience, accurate user logon prediction based on intelligent VDI user behavior modeling is required.

Previous works on VDI user logon modeling include CAFE [Zhang *et al.*, 2019] and SOUP [Fan *et al.*, 2019]. CAFE does not focus on specific user but aims at aggregating user logon modeling at the pool level. SOUP is an ensemble learning model to predict the specific user logon behavior based on tailored multi-grained features extracted from logon state sequence. Generally, earlier attempts towards user logon behavior modeling employ traditional supervised learning techniques under the single-instance single-label (SISL) setting, where the input and output to the predictive model correspond to single instance of user features and single label of user behavior respectively. Nonetheless, this SISL practice may lead to suboptimal performance as the strong relationships among pool-sharing VDI users are not properly exploited in the modeling framework.

In light of the above observations, we propose to utilizing the Multi-Instance Multi-Label (MIML) learning framework [Zhou and Zhang, 2006; Zhou *et al.*, 2012] for VDI user logon behavior modeling, which naturally fits the need of relationship exploitation among pool-sharing VDI users. Specifically, a novel approach for VDI user logon predic-

*Corresponding author

tion named BAMBOO (BAlanced MimlBOOst) is proposed in this paper. To induce the predictive model for each user, its supporting users identified in the same pool will be jointly modeled in the input space as a bag of multi-instance feature vectors and in the output space with multi-label prediction. The MIML formulation is optimized by adapting the popular MIML boosting procedure via balanced error-rate minimization. To enable effective multi-instance bag generation of users, BAMBOO identifies the group of supporting users based on logon frequency statistics and validation performance evaluation. Comparative studies on real VDI customer data sets evidently show that BAMBOO achieves highly competitive performance against other state-of-the-art methods.

The rest of this paper is organized as follows. Section 2 gives technical details of the proposed BAMBOO model. Section 3 discusses related researches. Section 4 introduces the collected real-world data sets for VDI user logon prediction and the experimental results of comparative studies, followed by conclusion in Section 5.

2 The Proposed Approach

In this section, we firstly give the MIML formulation for VDI user logon behavior modeling. After that, technical details of the proposed BAMBOO model are introduced.

2.1 The MIML Formulation

In many real-world tasks, a complicated object can be represented naturally by multiple instances and associated by multiple class labels. Multi-instance Multi-label [Zhou and Zhang, 2006; Zhou *et al.*, 2012] provides a framework to address such tasks. The MIML task can be formulated as following. Let $\mathcal{X}$ denote the instance space and $\mathcal{Y}$ the set of labels. The task is to learn a function $f : 2^{\mathcal{X}} \mapsto 2^{\mathcal{Y}}$ from training data set $D = (X_1, Y_1), (X_2, Y_2), ...(X_m, Y_m)$, where $X_i \subseteq \mathcal{X}$ is a set of instances $\{\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, ..., \boldsymbol{x}_{i,n_i}\}, \boldsymbol{x}_{ij} \in \mathcal{X}$ $(j = 1, 2, ..., n_i)$, and $Y_i \subseteq \mathcal{Y}$ is a set of labels $\{y_{i1}, y_{i2}, ..., y_{i,l_i}\}, y_{ik} \in \mathcal{Y}$ $(k = 1, 2, ..., l_i)$. Here n_i denotes the number of instances in X_i and l_i denotes the number of labels in Y_i.

In user logon behavior prediction, the complicated inherent patterns among pool sharing users could be modeled by the MIML learning framework. For a specific VDI user to model, given the supporting user group, the logon behavior prediction can be formulated as a MIML task. Formally speaking, for any user, let $\boldsymbol{s}_{t,p} = (s_t; s_{t-1}...; s_{t-p+1})$ be the logon state sequence where p is the sequence length and s_t is the logon state in time range[1] $[t-1, t)$ defined as follows:

$$s_t = \begin{cases} 1, & user\ has\ logon, \\ 0, & otherwise. \end{cases} \quad (1)$$

Then we can use $\boldsymbol{x}_t = \mu(\boldsymbol{s}_{t,p})$ to represent the user logon behavior in the feature space where μ is the multi-grained feature description function described in SOUP [Fan *et al.*, 2019]. Based on the user group with size n[2], the corresponding instance set of user logon behavior features can be constructed as $X_t = \{\boldsymbol{x}_{t,1}, \boldsymbol{x}_{t,2}, ..., \boldsymbol{x}_{t,n}\}$.

Correspondingly, we define $Y_t = \{y | s^y_{t+1} = 1, 1 \le y \le n\}$ as the label of instance set X_t. Here s^y_{t+1} denotes the logon state of user y in time range $[t, t+1)$, thus Y_t is the set of users that logon in the period. After that, the training data set D can be defined as:

$$D = \{(X_t, Y_t) \mid l \le t \le T-1\} \quad (2)$$

Here, l is the minimum feasible starting time of $\boldsymbol{s}_{t,p}$ and T is the ending time of the logon state sequence. For each user in the pool, our target is to learn a MIML model $f_{\mathrm{MIML}} : 2^{\mathcal{X}} \mapsto 2^{\mathcal{Y}}$ from D where $\mathcal{X}$ is the instance space and $\mathcal{Y}$ is the label space.

2.2 The BAMBOO Model

Bag Generation

In a VDI pool, not all users are suitable as supporting users especially those having very few or even no logon. To learn f_{MIML} for a specific user, the first step is to select supporting users to form the user group and then the MIML bags can be generated. Initially, given the complete user set $\mathcal{Y}$ of the VDI pool, we define the total logon count for each $y \in \mathcal{Y}$ as:

$$lcnt(y) = \sum_{t=l}^{T-1} \mathbb{I}(s^y_{t+1} > s^y_t) \quad (3)$$

Here, s^y_t is the logon state of user u in time range $[t-1, t)$. Then we sort $\mathcal{Y}$ in a descending order of the value of every user's $lcnt$ and select the first $\zeta * |\mathcal{Y}|$ users to construct the candidate supporting user set U where $0 \le \zeta \le 1$ is the candidate selection rate.[3] Finally, given the user to model $y^* \in \mathcal{Y}$, we can generate the correspondent supporting user group $\mathcal{Y}'$ by grouping y^* and the randomly selected $n-1$ users from $U \setminus \{y^*\}$.

The BAMBOO Algorithm

Notice that in the above section, the supporting user group $\mathcal{Y}'$ and the MIML bags (X_t, Y_t) are specifically generated for the prime user y^*. Therefore, the resulting MIML model should be optimized more towards reducing the prediction error-rate on the label for y^*. Accordingly, we propose a novel MIML learning method via balanced error-rate minimization. As indicated in [Zhou *et al.*, 2012], the idea of solving MIML problem is to identify its equivalence in the traditional supervised learning framework, using multi-instance learning or multi-label learning as the bridge. In BAMBOO, multi-instance learning is chosen as it focuses more on modeling relationship of users historical logon behavior. The pseudo-code of BAMBOO is summarized in Algorithm 1.

At the beginning, BAMBOO transforms each MIML sample (X_t, Y_t) into n multi-instance bags $\{[(X_t, y_1), \psi(X_t, y_1)], ..., [(X_t, y_n), \psi(X_t, y_n)]\}$ where ψ is a function $\psi : 2^{\mathcal{X}} \times \mathcal{Y}' \mapsto \{+1, -1\}$ defined as:

$$\psi(X_t, y) = \begin{cases} +1, & y \in Y_t \\ -1, & otherwise. \end{cases} \quad (4)$$

[1] In this paper, we use half an hour (30 minutes) as the unit of t.

[2] In this paper, n is set to 2 because higher n has much higher training cost but can not bring significant performance improvement.

[3] In this paper, ζ is set to 0.5.

Algorithm 1: The pseudo-code of BAMBOO

Input: $\mathcal{Y}, y^*, l, T, n, \{s_{t,j}\}(l \leq t \leq T-1, j = 1, 2, ..., n), \mu, \zeta, \nu, V, eval(.,.), R$
Output: f^*_{MIML}
Let $\boldsymbol{x}_{t,j} = \mu(s_{t,j})(l \leq t \leq T-1, j = 1, 2, ..., n)$.
Initialize the candidate resulting MIML model set $F = \{\}$
For each $y \in \mathcal{Y}$, compute $lcnt(y) = \sum_{t=l}^{T-1} \mathbb{I}(s^y_{t+1} > s^y_t)$ and select top $\zeta \times |\mathcal{Y}|$ users to form the candidate set U.
for $r = 0, ..., R$ **do**
  Randomly pick $n-1$ users from $U \setminus \{y^*\}$ and group them with y^* to generate $\mathcal{Y}'$ where $|\mathcal{Y}'| = n$. Then compose MIML samples $\{(X_t, Y_t)\} = \{(\{\boldsymbol{x}_{t,1}, \boldsymbol{x}_{t,2}, ..., \boldsymbol{x}_{t,n}\}, \{y_1, y_2, ...y_n\}) | l \leq t \leq T-1\}$.
  Transform each MIML sample (X_t, Y_t) into n multi-instance samples, thus the original data set is transformed into $(T-l) \times n$ multi-instance bags, where each bag $B_{t,k} = [(X_t, y_k), \psi(X_t, y_k)](l \leq t \leq T-1, k = 1, 2, ...n)$
  Initialize the each bag's weight as: $W_{t,k} = \frac{1}{(T-l) \times n}(l \leq t \leq T-1, k = 1, 2, ..., n)$.
  for $i = 1, 2, ...I$ *iterations* **do**
    For $B_{t,k}$, assign the label $\psi(X_t, y_k)$ to its each instances $(\boldsymbol{x}_{t,j}, y_k)(j = 1, 2, ...n)$ with weight $W_{t,k}/n$.
    Build an instance-level predictor $h_t[(\boldsymbol{x}_{t,j}, y_k)] \in \{-1, +1\}$.
    Compute the error rate $e_{t,k} \in [0, 1]$ of $B_{t,k}$ by $e_{t,k} = \frac{1}{n} \sum_{j=1}^{n} [\![h_i[(\boldsymbol{x}_{t,j}, y_k)] \neq \psi(X_t, y_k)]\!]$
    if $e_{t,k} < 0.5$ *for all* $l \leq t \leq T-1, k = 1, 2, ...n$ **then** break;
    Compute $c_i = \text{argmin}_{c_i} \{\nu \times \sum_{t=l}^{T-1} \sum_{k=2}^{n} W_{t,k} \exp[(2e_{t,k} - 1)c] + \sum_{t=l}^{T-1} W_{t,1} \exp[(2e_{t,1} - 1)c]\}$
    if $c_i \leq 0$ **then** break;
    Update $W_{t,k} = W_{t,k} \exp[(2e_{t,k} - 1)c_i]$ and re-normalize such that $\sum_{t=l}^{T-1} \sum_{k=1}^{n} W_{t,k} \exp[(2e_{t,k} - 1)c_i] = 1$.
  end
  Obtain f_{MIML} where $f_{\text{MIML}}(X^*) = \{y | sign(\sum_j \sum_i c_i h_i[(\boldsymbol{x}^*_j, y)]) = +1\}$($\boldsymbol{x}^*_j$ is the j-th instance of X^*)
  $F = F \cup f_{\text{MIML}}$
end
Return the model $f^*_{\text{MIML}} = \text{argmax}_{f_{\text{MIML}}} \{eval(f_{\text{MIML}}, V) | f_{\text{MIML}} \in F\}$

For simplicity, y_1 is used to represent the prime user.

Thus, the original training data set is transformed into $(T-l) \times n$ multi-instance bags: $[(X_t, y_k), \psi(X_t, y_k)](l \leq t \leq T-1, k = 1, 2, ...n)$. Then a multi-instance learning function f_{MIL} can be derived with which the resulting MIML function can be accomplished by:

$$f_{\text{MIML}}(X^*) = \{y | sign[f_{\text{MIL}}(X^*, y)] = +1\} \quad (5)$$

Then we obtain f_{MIL} with the following boosting procedure. Let (B, g) denote the bag $[(X, y), \psi(X, y)], B \in \mathcal{B}, g \in \mathcal{G}$ and E denote the expectation. The ultimate goal is to learn a function $\mathcal{F}(B)$ with which the bag-level exponential loss $E_{\mathcal{B}} E_{\mathcal{G}|\mathcal{B}}[\exp(-\mathrm{g}\mathcal{F}(\mathrm{B}))]$ is minimized. The target of each boosting iteration is adding a new weak classifier to expand $\mathcal{F}(B)$ to $\mathcal{F}(B) + cf(B)$ where $f(B)$ can be learned by:

$$f(B) = \frac{1}{n} \sum_j h(\boldsymbol{b}_j) \quad (6)$$

Here, $h(\cdot)$ is the instance-level classifier and $h(\boldsymbol{b}_j) \in \{-1, +1\}$ denotes its prediction for the bag B's j-th instance.

Given the bag-level weight $W = \exp(-\mathrm{g}\mathcal{F}(\mathrm{B}))$, the best $f(B)$ to be added can be fulfilled by learning $h(\cdot)$ which minimizes the weighted instance-level classification error where the instance-level weight is set to W/n. After $f(B)$ is found, the bag's error rate can be computed as: $e = \frac{1}{n} \sum_j [\![(h(\boldsymbol{b}_j \neq g)]\!]$. Then the best multiplier c with the balanced error rate can be computed by optimizing the exponential loss as:

$$c = \text{argmin}_c \{\nu \times \sum_{t=l}^{T-1} \sum_{k=2}^{n} W_{t,k} \exp[(2e_{t,k} - 1)c] + \sum_{t=l}^{T-1} W_{t,1} \exp[(2e_{t,1} - 1)c]\} \quad (7)$$

Here, $W_{t,k}$ and $e_{t,k}$ are the weight and error rate of bag $[(X_t, y_k), \psi(X_t, y_k)]$. Specifically, $W_{t,1}$ and $e_{t,1}$ are the weight and error rate of the bag with the label of the prime user y^*. The balance parameter ν $(0 < \nu < 1)$ is applied on the bags with non prime user labels to make the optimization focus more on the label of the prime user. Lastly, the bag-level weights are updated as $W_{t,k} = W_{t,k} \exp[(2e_{t,k} - 1)c]$ and renomalized.

Furthermore, to minimize the effect of randomness, we generate multiple supporting user groups and train the same number of MIML learners. Given the validation set V and the performance evaluation function $eval(f_{\text{MIML}}, V)$[4], the resulting MIML model f^*_{MIML} is obtained by:

$$f^*_{\text{MIML}} = \text{argmax}_{f_{\text{MIML}}} \{eval(f_{\text{MIML}}, V)\} \quad (8)$$

In the testing phase, f^*_{MIML} is applied on the new bag X^* of y^* to predict the future user logon behavior.

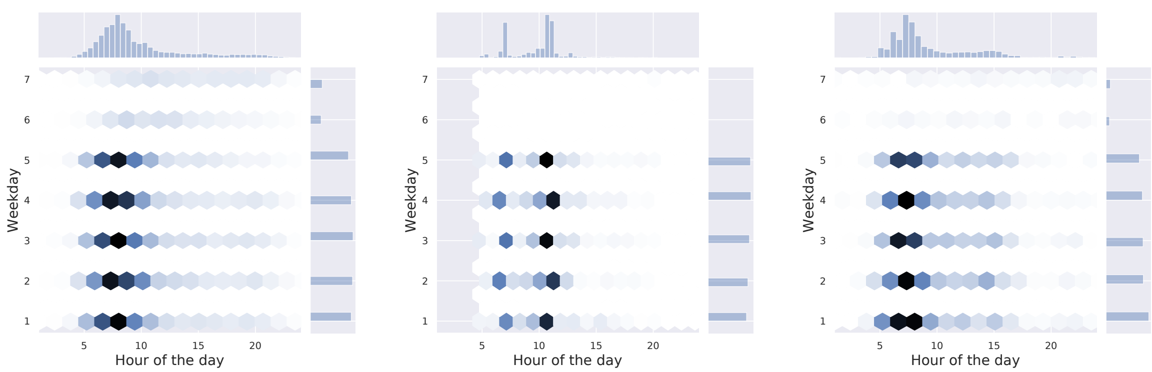
3 Related Work

In many real-world problems, the objects have complicated inherent patterns. They can be represented as multi-instance

[4]In this paper, *CSAG* is used as performance evaluation function.

Division	Training period	Data sets	User count	Session duration mean (hr)	Logon time entropy mean	Session count mean	Testing period
D1	17/09/18 - 25/03/19	Pacific	134	5.74±2.04	1.82	153.78±84.58	08/04/19 - 15/04/19 15/04/19 - 21/04/19
		Antarctic	87	5.35±2.46	1.19	54.95±49.5	01/04/19 - 08/04/19 15/04/19 - 21/04/19
		Arctic	47	4.45±2.57	1.67	104.02±73.03	25/03/19 - 01/04/19 15/04/19 - 21/04/19
D2	17/09/18 - 25/03/19	Pacific	122	5.70±2.06	1.77	114.78±58.35	10/06/19 - 16/06/19 17/06/19 - 23/06/19
		Indian	65	3.35±2.10	1.80	84.0±47.61	03/06/19 - 09/06/19 10/06/19 - 16/06/19
		Atlantic	108	154.03±264.43	1.41	30.07±39.57	10/06/19 - 16/06/19 17/06/19 - 23/06/19

Table 1: Statistics of the real-world VDI data sets.



(a) Pacific (b) Antarctic (c) Atlantic

Figure 1: Real-world data sets.

bags and associated with multiple labels. For example, an image can be represented by multiple sub-images and associated with several high level concepts. A formalized multi-instance multi-label learning (MIML) framework [Zhou and Zhang, 2006; Zhou *et al.*, 2012] was introduced to solve such tasks and many successful researches and applications ([Yang *et al.*, 2017; Ma *et al.*, 2019]) have been proposed in recent years. [Briggs *et al.*, 2012] introduced Rank-Loss Support Instance Machines for MIML instance annotation. [Wu *et al.*, 2014] proposed EnMIMLNN, a novel ensemble MIML learning framework for genome-wide protein function prediction. [Huang *et al.*, 2014] proposed a fast MIML approach by optimizing the approximated ranking loss and utilizing label relations in a shared space and perceiving sub-concepts for complicated labels. [Pham *et al.*, 2015] presented probabilistic MIML model in the presence of novel class instances and introduced an efficient computational method for inference. [Xie *et al.*, 2016] explored the application of using MIML for the classification of frog species. [Feng and Zhou, 2017] proposed DeepMIML, a general deep MIML model that can automatically learn the instance description as well as disclosing the relation between input patterns and output labels. [Gao *et al.*, 2018] presented a deep MIML network which successfully separates object-level sounds by automatically linking audio bases to object categories and using the disentangled bases to supervise non-negative matrix factorization.

To the best of our knowledge, BAMBOO is the first study that applies the MIML algorithm to the VDI domain. Considering the fact that VDI user logon behavior modeling is a complicated time-series prediction task ([Solow, 1994]) where many researches have been done in this area. [Chatfield, 1978] introduced Holt-Winters which is widely used for time-series data prediction. [Taylor and Letham, 2017] proposed Prophet for large-scale time-series data prediction which is based on additive model and fits non-linear trends with seasonalities. It was shown that RNN ([Elman, 1990]) outperforms conventional learning algorithms on many time series tasks in [Connor *et al.*, 1994; Ho *et al.*, 2002]. [Yin *et al.*, 2017] applies RNN on intrusion detection and proves its capacity to distinguish binary data. But its validity on VDI user logon prediction has not been demonstrated. Recently, a few successful researches were applied in VDI domain. [Zhang *et al.*, 2019] proposed a multi-grained features based ensemble learning model to solve the VDI non-persistent pool user logon prediction problem. [Fan *et al.*, 2019] introduced an encoded multi-granularity feature based model to predict

Comparing Methods	CSAG↑					
	Pacific-T1	Pacific-T2	Antarctic-T1	Antarctic-T2	Arctic-T1	Arctic-T2
RNN-IDS	26	962	1529	1352	2284	583
DeepMIML	-80199	-84865	681	-41661	-5629	-24314
SOUP	2511∗	2140∗	2280	1957	2178	1781
MIMLBOOST	2354	2016	2464∗	2013∗	**2286**	**1971**
BAMBOO	**2535•**	**2155•**	**2488•**	**2069•**	2248∗	1890∗
	Pacific-T3	Pacific-T4	Indian-T3	Indian-T4	Atlantic-T3	Atlantic-T4
RNN-IDS	1328	1456	2094	1526	1854	1617
DeepMIML	-43330	-44618	-4821	-18996	-36765	-66618
SOUP	**2800**	**2698**	3094∗	**2952**	3037∗	2694
MIMLBOOST	2616	2565	3088	2769	3003	2703∗
BAMBOO	2646•∗	2614•∗	**3097•**	2898•∗	**3096•**	**2751 •**

Comparing Methods	AUC↑					
	Pacific-T1	Pacific-T2	Antarctic-T1	Antarctic-T2	Arctic-T1	Arctic-T2
RNN-IDS	0.5727	0.8103	0.8770	**0.8943**	0.4940	0.6110
DeepMIML	0.2963	0.2571	0.6203	0.6131	0.2130	0.3926
SOUP	0.9165∗	**0.9580**	0.8464	0.8105	0.6857	0.7983
MIMLBOOST	0.8925	0.9067	0.9181∗	0.8505	**0.7475**	**0.8897**
BAMBOO	**0.9225•**	0.9353•∗	**0.9214•**	0.8669• ∗	0.7217∗	0.8484∗
	Pacific-T3	Pacific-T4	Indian-T3	Indian-T4	Atlantic-T3	Atlantic-T4
RNN-IDS	0.7130	0.7840	0.7430	0.7600	0.7960	0.7450
DeepMIML	0.5650	0.3996	0.6503	0.4738	0.5687	0.7195
SOUP	**0.9559**	**0.9200**	0.9111∗	**0.9683**	0.869∗	0.7276
MIMLBOOST	0.8819∗	0.8642∗	0.9096	0.9100	0.8270	0.7834 ∗
BAMBOO	0.8375	0.8350	**0.9134•**	0.9423•∗	**0.9333•**	**0.8360•**

Table 2: Performance on the real-world VDI data sets in term of CSAG and AUC (↑ means the higher the better)

Dataset	Pacific	Antarctic	Arctic	Indian	Atlantic
$\frac{b}{a}$	320	200	200	160	640

Table 3: $\frac{b}{a}$ used in CSAG for five datasets

user logon activities. Whilst existing approaches on VDI user logon behavior modeling are all single-instance single-label learning models, BAMBOO solves this problem with a MIML approach by incorporating the advantage of discovering pattern-label relation among different users.

4 Experiments

4.1 Comparing Algorithms

To evaluate the performance of BAMBOO, four state-of-the-art methods are selected for comparison.

- SOUP ([Fan *et al.*, 2019]) is an ensemble learning model based on encoded multi-granularity description to predict single user logon activities. In our configuration, the feature extraction sequence length is set to 3600 (120 days).
- MIMLBoost ([Zhou and Zhang, 2006]) is a multi-instance multi-label algorithm using multi-instance learning as the bridge to degenerate MIML problem to MISL problem. In our configuration, the boosting rounds is set as 30 and GBDT ([Friedman, 2001; 2002]) is used as instance-level learner.
- RNN-IDS ([Yin *et al.*, 2017]) is a deep learning approach for intrusion detection and it shows RNN's capability to make prediction on binary time-series which is similar to user logon scenario. In our configuration, the input shape of 3D tensor (batch size, timesteps, input dim) is set as (256, 72, 1).
- DeepMIML ([Feng and Zhou, 2017]) exploits deep neural network formation to generate instance representation for MIML. In our configuration, it shares the same training set as MIMLBOOST and the sub-concept number is set as 2.

For BAMBOO, the algorithm parameters are set as following. The feature extraction sequence length is the same as SOUP. The supporting user group size n is set to 2. The boosting rounds R is set to 30. GBDT is used as instance-level learner. The candidate selection rate ζ is set to 0.5. The error-rate balance parameter ν is set to 0.8.

4.2 Data Sets

The experiments are conducted on five real VDI customers data sets where the logon behavior granularity interval is set to 30 minutes. Each of the data set is from one of the customer's persistent desktop pools. We divide the training data into two division: D1 training data period is from Sept. 17th 2018 to Mar. 4th 2019, D2 training data period is from Nov. 5th 2018 to Apr. 22th 2019, both of which are 24 weeks data. BAMBOO is tested on two separated test data sets: D1 test set is selected from March 26th to April 21st 2019 and D2 test set is selected from Jun. 3rd 2019 to Jul. 1st 2019. We name the five customers data sets as: `Pacific`, `Antarctic`, `Arctic`, `Indian` , `Atlantic`. Then the six test sets in D1 is named as `Pacific-T1`, `Pacific-T2`, `Antarctic-T1`, `Antarctic-T2`, `Arctic-T1`, `Arctic-T2` and six test sets in D2 is named as as `Pacific-T3`, `Pacific-T4`, `Indian-T3`, `Indian-T4`, `Atlantic-T3`, `Atlantic-T4`, Table 1 contains the statistics of the data sets.

The typical real-world data sets are illustrated in Figure 1. The darker the hexagon is, the more logon happened at that time. We can see that all the data sets show a common daily pattern that most of the user logon activities happen on working day and also have specific patterns. Data set `Pacific` has the biggest number of user count and the longest session duration. Most of its logon happen between 6am to 8am and the peak is around 7am for all the working days. The logon distribution of `Pacific` looks similar to a Gaussian distribution. Furthermore, it has notable logon activities on weekend. Whereas, data set `Antarctic` has two remarkable logon peaks at 7am and 11am respectively, and it has the smallest logon time entropy being most centralized. To be noted that the users in `Antarctic` have seldom logon after working day noon and on weekend. Different to `Pacific` and `Antarctic`, `Atlantic` has a rather long session duration average (154 hours), the session keeps alive for nearly one week. The pattern is seen in some call-center customers where three shifts provide 24-hour service.

Evaluation Metrics

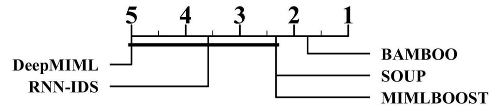
By modeling the user logon behavior to a zero-one serial, the prediction is actually a binary classification task. We use one common classification performance metric (*AUC*) and one VDI domain specific metric (*CSAG*, defined in [Fan *et al.*, 2019]) for comparison. *AUC* is one of the most important evaluation metrics for validating a classification model's performance. It describes how much the model can distinguish between classes. *CSAG* is a VDI domain specific metric to measure the algorithm's effectiveness on cost-efficiency. *CSAG* is computed by Equation 9 where a denotes the cost saving factor of powering off unnecessary desktops and b the penalty cost factor of the incorrect prediction. The configuration details are listed in Table 3.

$$\mathrm{CSAG} = \mathrm{TN} - \frac{b}{a}\mathrm{FN} \tag{9}$$

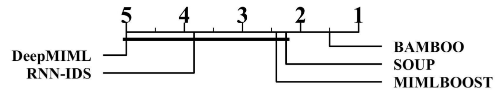
4.3 Experimental Results

Table 2 reports the detailed performance metric of each comparing algorithm where the bold number means the 1st rank, $*$ means the 2nd rank and $\bullet$ means BAMBOO outperforms MIMLBOOST. Furthermore, a post-hoc Bonferroni- Dunn test [Dunn, 1961] is performed to compare the relative performance among the comparing approaches. The CD diagrams are presented in Figure 2, where the average rank of each approach is marked along the axis (the smaller the better).

It is impressive to observe that 1) On the most important VDI specific metric: *CSAG*, BAMBOO achieves 1st rank on 58% data sets and 2nd rank on 42% data sets. On the *AUC* metric, BAMBOO ranks 1st on 42% data sets and 2nd on 42% data sets. 2) It is remarkable that BAMBOO achieves the lowest average rank in terms of both *CSAG* and *AUC* metric. 3) 67% 1st ranks are from MIML methods. This is the clear evidence that leveraging MIML framework obtains notable performance improvement and we believe the driving force here is the hidden relation between users in the pool. In the cases where SOUP outperforms BAMBOO, we observe an apparently common logon behavior pattern. It makes sense as SOUP model is trained with samples from all the users in the pool, the common pattern is strengthened. 4) Among all the data sets, BAMBOO outperforms MIMLBOOST in 75% cases which clearly demonstrates the effectiveness of the balance parameter introduced in BAMBOO.



(a) CD-AUC



(b) CD-CSAG

Figure 2: Comparison of BAMBOO (control algorithm) against other comparing approaches with the Bonferroni-Dunn test (CD=1.6125 at 0.05 significance level)

5 Conclusion & Future Work

In this paper, we propose BAMBOO, a novel MIML approach towards VDI user logon behavior modeling which serves as the key enabling-technique to build intelligent VDI power management systems. Specifically, each user to model is grouped with supporting users selected in the same pool to construct the MIML bags so that the strong relationship among pool-sharing users is incorporated. After that, an innovative MIML learner with class error rate balance parameter is derived for future user logon prediction. The effectiveness of BAMBOO is validated through comprehensive experiments on real DaaS customer data sets. With a rough computation, about \$660 can be saved per week for a persistent pool with 100 VDI desktops which equals to reducing the infrastructure cost over 50%. [5]

In the future, we will explore the possibility of optimizing the efficiency of BAMBOO with the technique in [Huang *et al.*, 2014] and also try to group users from different pools to see if more user relationship can be discovered.

Acknowledgements

The authors wish to thank the anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Key R&D Program of China (2018YFB1004300), and the China University S&T Innovation Plan Guided by the Ministry of Education.

[5] We use the AWS EC2 instance t3.medium (\$0.06/hour) as the virtual machine of VDI desktops to calculate the infrastructure cost.

References

[Briggs *et al.*, 2012] Forrest Briggs, Xiaoli Z Fern, and Raviv Raich. Rank-loss support instance machines for miml instance annotation. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 534–542, 2012.

[Chatfield, 1978] C. Chatfield. The holt-winters forecasting procedure. *Journal of the Royal Statistical Society*, 27(3):264–279, 1978.

[Connor *et al.*, 1994] Jerome T Connor, R Douglas Martin, and Les E Atlas. Recurrent neural networks and robust time series prediction. *IEEE Transactions on Neural Networks*, 5(2):240–254, 1994.

[Dunn, 1961] Olive Jean Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56:52–64, 1961.

[Elman, 1990] Jeffrey L Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[Fan *et al.*, 2019] Wen-Ping Fan, Yao Zhang, Jinghan Wu, Hua Chen, and Bin-Yang Li. Soup: Single user logon model with encoded multi-grained features. *RADIO-2019*, 2019.

[Feng and Zhou, 2017] Ji Feng and Zhi-Hua Zhou. Deep miml network. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pages 1884–1890, 2017.

[Friedman, 2001] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29:1189–1232, 2001.

[Friedman, 2002] Jerome H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38:367–378, 2002.

[Gao *et al.*, 2018] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proceedings of the 15th European Conference on Computer Vision*, pages 35–53, 2018.

[Ho *et al.*, 2002] Sui-Lau Ho, Min Xie, and Thong Ngee Goh. A comparative study of neural network and box-jenkins arima modeling in time series prediction. *Computers & Industrial Engineering*, 42(2-4):371–375, 2002.

[Huang *et al.*, 2014] Sheng-Jun Huang, Wei Gao, and Zhi-Hua Zhou. Fast multi-instance multi-label learning. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pages 1868–1874, 2014.

[Ma *et al.*, 2019] Yuling Ma, Chaoran Cui, Xiushan Nie, Gongping Yang, Kashif Shaheed, and Yilong Yin. Pre-course student performance prediction with multi-instance multi-label learning. *Science China Information Sciences*, 62(2):1–3, 2019.

[Pham *et al.*, 2015] Anh Pham, Raviv Raich, Xiaoli Fern, and Jesús Pérez Arriaga. Multi-instance multi-label learning in the presence of novel class instances. In Francis Bach and David Blei, editors, *In Proceedings of the 32nd International Conference on Machine Learning*, pages 2427–2435, 2015.

[Solow, 1994] Andrew R Solow. Time series prediction: Forecasting the future and understanding the past. *Science*, 265(5179):1745–1747, 1994.

[Taylor and Letham, 2017] Sean J. Taylor and Benjamin Letham. Forecasting at scale. *The American Statistician*, 72:37–45, 2017.

[Wu *et al.*, 2014] Jian-Sheng Wu, Sheng-Jun Huang, and Zhi-Hua Zhou. Genome-wide protein function prediction through multi-instance multi-label learning. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 11(5):891–902, 2014.

[Xie *et al.*, 2016] Jie Xie, Michael Towsey, Liang Zhang, Kiyomi Yasumiba, Lin Schwarzkopf, Jinglan Zhang, and Paul Roe. Multiple-instance multiple-label learning for the classification of frog calls with acoustic event detection. In *Proceedings of the International Conference on Image and Signal Processing*, pages 222–230, 2016.

[Yang *et al.*, 2017] Hao Yang, Joey Tianyi Zhou, Jianfei Cai, and Yew Soon Ong. Miml-fcn+: Multi-instance multi-label learning via fully convolutional networks with privileged information. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1577–1585, 2017.

[Yin *et al.*, 2017] C. Yin, Y. Zhu, J. Fei, and X. He. A deep learning approach for intrusion detection using recurrent neural networks. *IEEE Access*, 5:21954–21961, 2017.

[Zhang *et al.*, 2019] Yao Zhang, Wen-Ping Fan, Xuan Wu, Hua Chen, Bin-Yang Li, and Min-Ling Zhang. Cafe: Adaptive vdi workload prediction with multi-grained features. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pages 5821–5828, 2019.

[Zhou and Zhang, 2006] Zhi-Hua Zhou and Min-Ling Zhang. Multi-instance multi-label learning with application to scene classification. In *Advances in Neural Information Processing Systems 19*, pages 1609–1616, 2006.

[Zhou *et al.*, 2012] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291–2320, 2012.