Discriminating User Behavior through PC Operation Logs by PageRank Convergence Patterns

Kazuhiro Suzuki, Hiroshi Yasuda, Kilho Shin, and Tetsuji Kuboyama

***Abstract*—This paper aims to discriminate user behavior from PC operation logs by PageRank convergence patterns of networks. We will show that we can discriminate user behavior by making clear the difference in behavior between working and resting. We will construct a window transition network from active window transition logs. In this network, each node represents an application corresponding to the active window. We will use PageRank convergence patterns of networks. The convergence patterns are assumed to imply the roles of nodes in the network. Then, we will transform these patterns into symbolic representations to compute similarities in user behavior and apply the kernel method for classification with SVMs. We will conduct experimental result. This evaluation is highly accurate except amongst people who seldom use computers. These results show that this method allows us to discriminate user behavior according to the context at work with SVMs.**

***Index Terms*—Network, log analysis, PageRank, SVM.**

I. Introduction

Our study aims to discriminate user behavior though PC Operation Logs. Our purpose is to form conjectures about user behavior hidden behind the networks, in particular, to extract the role of an individual user on a network. Network analysis has various applications. For example, hyperlinks, biological, and social networks, and more. In this study, we applied the PageRank algorithm to analyze PC Operation Logs.

In general, user behavior is different between *working time* and *resting time* at work. The targeting logs in this paper were recorded at a certain IT company. Employees in this IT company work from 9:00 to 17:30. We considered that *working time* is set to from 9:00 to17:30 except for *resting time*. *Resting time* includes after-*working time*. We have proposed a method to discriminate between the differences in computer usage during *working time* and *resting time*.

The diversity of relationships, such as friendships and web hyperlinks, are regarded as network structures. To discover knowledge from large scale networks, there have been attempts to extract the community behind the network structures. The mainstream community extraction method is a density-based clustering, such as the Newman Method [1]. The density-based approach attempts to find sets of nodes that are connected to each other at a high density in the network. In contrast, a different approach is necessary for cluster nodes that play a similar role on the network. For this purpose, a new structure-based network clustering has been proposed. The structure-based clustering method focuses on the structural similarity around individual nodes, and classifies nodes with a similar structure into the same cluster.

There are two main approaches to the structure-based clustering:

1) An *explicit approach* explores the structure in the vicinity of each node in an explicit way such as in [2]–[4].
2) An *implicit approach* (by Fushimi *et al.* [5]). In this method, the similarity between convergence patterns of nodes in the PageRank algorithm is regarded as the similarity between the nodes. The convergence pattern of a node implies the structure in the vicinity of the node, which is called *functionality* in [5]. Hereafter, we refer to this method as *functionality clustering*.

For this paper, we employed the implicit approach by Fushimi [5] to measure the similarity between two nodes according to the roles of the nodes in the network. Fushimi's method employs the cosine similarity to measure the distance of the convergence patterns of nodes. The cosine similarity was not suitable for the network that we addressed because the results were greatly affected by the scale of the network. In this paper, we have proposed a robust approach to the delay and scale by transforming the convergence patterns into symbolic representations. To show the effectiveness of our improvements, we conducted an experiment for a network structure obtained from PC logs (specifically, window transition logs).

Real network data was used for this study. The network data was generated from the PC operation logs of a certain IT company. The company consists of six departments: marketing, sales, sales-office, development-support, development, and quality-assurance. The company gave Windows based computers to each employee and recorded the transitions of the active windows. The PC operation logs include operation records collected from September to December 2010 that give information about the identity of the PC (the PC ID), the user's name, the application name, and the time of the event when an active window was switched. Table I depicts the log data.

TABLE I: PC Operational Log

PC Name	Active App	User Name	Time Set	Term
09-470	outlook	marketing01	2010/9/1 8:54	2
09-470	excel	marketing01	2010/9/1 8:56	1
09-470	outlook	marketing01	2010/9/1 8:57	2
09-470	word	marketing01	2010/9/1 8:59	1

Manuscript received November 4, 2013; revised January 23, 2014.

K. Suzuki and H. Yasuda are with the School of Science and Technology for Future Life, Tokyo Denki University. Tokyo, Japan (e-mail: 12fmi23@ms.dendai.ac.jp, yasuda@mpeg.im.dendai.ac.jp).

K. Shin is with the Graduate School of Applied Informatics, University of Hyogo. Hyogo, Japan (e-mail: yshin@ai.u-hyogo.ac.jp).

T. Kuboyama is with Computer Centre, Gakushuin University, Tokyo, Japan (e-mail: ori-immm2014@tk.cc.gakushuin.ac.jp).

Fig. 1. shows how to transform PC Operations Logs into a network. Each node is an active window at some point in time, and is attributed with the information of the names of the department, the application and whether the window has become active during the *working time*. The directed edge between nodes means that the window of the ending node has become active taking over the window of the starting node. If an employee uses the applications, the node is labeled with either "*working time*" or "*resting time*". It links in the order in which they became active, and also shows transition probability as a Markov model. If a node i in the network has outgoing edges to w_i nodes, the transition probability of each edge is $1/w_i$. Therefore, if v_a of the w_i nodes belong to the same application a, the probability that the node i transits to the application a is v/w_i.

Active window	Label
Mailer	Resting
Mailer	Working
Excel	Working
Mailer	Working
Mailer	Resting
Excel	Resting
Excel	Working
Mailer	Working
Mailer	Resting

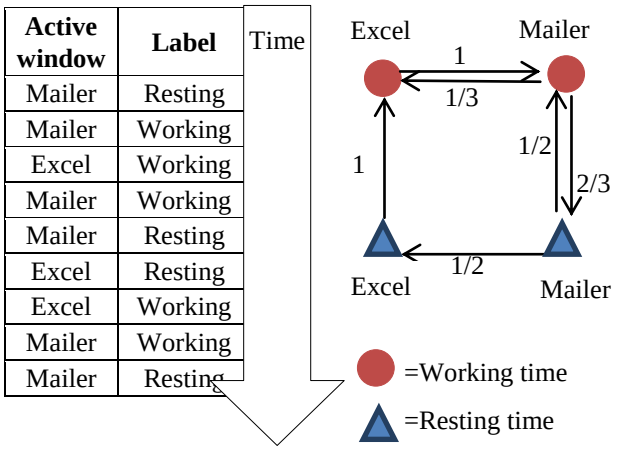


Fig. 1. Example of how to transform PC Operations Logs into a network.

II. Related Work

Fushimi's method is based on the similarity of the convergence patterns at each node of PageRank. PageRank [6], [7] is the algorithm developed as a method of ranking web pages used in Google. PageRank considers web pages' importance by how many links point to them from other relevant web pages.

The functionality clustering method by Fushimi *et al.* [5] regards the convergence process of the PageRank score of each node as a time series vector. Next, these vectors are clustered by the greedy *k*-median method with cosine similarity between vectors. Typically *k*-median is more robust against outliers than the average *k*-means, and a greedy algorithm is employed for efficiency by sacrificing accuracy to some extent. The functionality clustering method is summarized as follows, where each feature vector of each node is a sequence of PageRank scores in the convergence process of PageRank, the dimensions of the feature vector are denoted by *T*, and the number of clusters is denoted by *K*.

1) Obtaining feature vectors: a sequence of PageRank scores in the convergence process is computed for each node v in a given network, and denoted by X_v.
2) Computing similarity: for any two nodes u and v, the cosine similarity between X_u and X_v is computed, and denoted by $sim(X_u, X_v)$.
3) Clustering: nodes in the network are clustered by the greedy *k*-median method with the similarity $sim(X_u, X_v)$.

In this paper, we use PC operation logs to analyze user behavior at work in a company. In prior work, Saito *et al.* [8] analyzes the behavior of computer users by using the hidden Markov model and the kernel PCA of graph structures, and creates a probabilistic model of user behavior.

III. Proposed Method

Functionality clustering focuses on the convergence patterns of PageRank scores for nodes. We would like to consider the similarity of convergence patterns rather than the values of PageRank scores. However, functionality clustering is susceptible to the delay of patterns and scales. Therefore, we have proposed a more robust method.

A. Convergence Pattern and Damping Factor

In the PageRank algorithm, there is a scaling parameter called the damping factor, which denotes the probability of following the actual edges in a network. The damping factor affects the speed of the convergence of the PageRank algorithm. As the damping factor approaches 1, the expected value of the repetition increases dramatically. The original PageRank paper proposed setting the value to 0.85 as a tradeoff between convergence speed and effectiveness. However, we want to focus on the process of convergence. The network structure should be sensitively reflected in the convergence patterns, so we set the damping factor to 0.99.

B. Symbolic Representation of Convergence Patterns

We employed a new method for convergence patterns of PageRank scores. The convergence speeds of PageRank scores and the convergence patterns are different for each node. Thus, we need to take into account of delay and scale of patterns. As a measure of similarity between convergence patterns, the functionality clustering [5] employs a cosine similarity. The cosine similarity is relatively robust against the scale of patterns, while susceptible to the delay of data.

In our method, we employed SAX [9]. SAX is a well-known method for clustering time-series data. SAX standardizes time-series data, and discretizes the standardized data into symbolic representations. However, SAX assumes that the time series data follow a normal distribution while the PageRank convergence curve, in general, does not follow a normal distribution. In this paper, we employed a different discretization as follows (denoted by SAX_{UDF}).

Fig. 2 shows an example of the SAX_{UDF}.

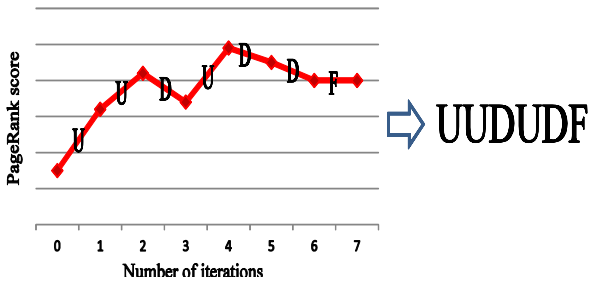


Fig. 2. Example of the SAX_{UDF}.

Let the feature vector representations of a convergence pattern be $X = (x_1, x_2, \ldots, x_t)$. This vector X is transformed in to a symbolic representation $S(X) = s_1 s_2 \cdots s_{t-1}$ as follows:

$$s_t = \begin{cases} U\left(x_{\{t-1\}} < x_{\{t\}}\right) \\ D\left(x_{\{t-1\}} > x_{\{t\}}\right) \\ F\left(x_{\{t-1\}} = x_{\{t\}}\right) \end{cases}$$

In SAX_{UDF}, the distance of symbolic representations is measured by Edit Distance.

We compared the SAX_{UDF} with the cosine similarity used in the functionality clustering.

We compared network data. This network data was generated from the PC operation logs of each department. The network includes all departments, and has 383 nodes and 4565 edges. In the comparison, we calculated the distances of the nodes in two different ways: one was based on the functionality clustering method [5], while the other was based on SAX_{UDF} method that we proposed in this paper.

Fig. 3 shows the scatter plot of the distance in space of nodes based on cosine similarity, while Fig. 4 shows the plot based on the edit distance. Note that these scatter plots are shown after reducing the dimensionality by means of Multi-Dimensional Scaling (MDS). Fig. 3 and Fig. 4 show that the proposed method clearly divides the nodes between *working time* and *resting time* as compared to the cosine similarity.

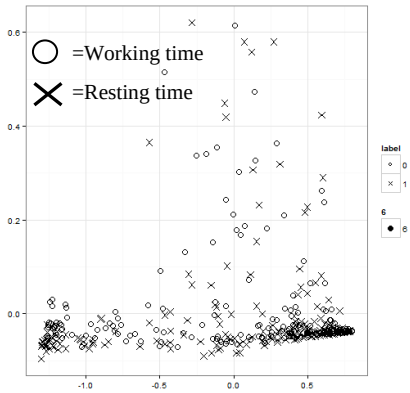


Fig. 3. Functionality clustering.

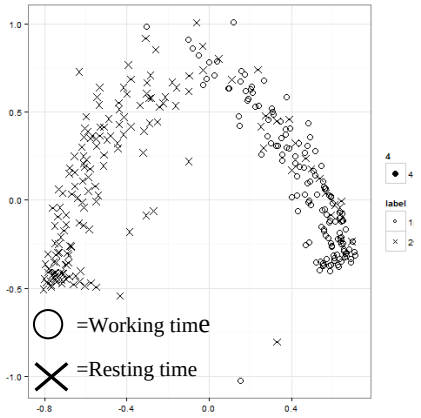


Fig. 4. Proposed method.

The confirmation is based on the conditional entropy H(X | Y). We let $X \in \{Working\ time, Resting\ time\}$, $Y \in \{cluster1, cluster2, \ldots, cluster13\}$. The conditional entropy H(X | Y) for the conventional method is 0.939 while our method (based on *K*-median) is 0.258. These results show that our method clearly divides the usage of applications into different clusters according to time (*working time* and *resting time*). These results imply that our method can detect the roles of applications in *working time* and *resting time*.

IV. Experimental Results

We discriminated user behavior into *working time* or *resting time*. The discrimination performance was measured by AUC (area under the ROC curve) using LibSVM [10]. We design a string kernel based on the edit distance between two symbolic representations of PageRank convergence patterns, and evaluated the discrimination performance using an SVM (LibSVM [10]).

Real network data was used for this discrimination. This network data was generated from the PC operation logs of each user: marketing (7 people), sales (9 people), development (16 people), quality-assurance (8 people), sales-office (2 people), and development-support (16 people).

We conducted a 5-fold cross validation. We made the test data and the training data as follows (See Fig. 5).

1) Divide the log into 5 segments at random.
2) Transform each divided log into a network.
3) Calculate the PageRank convergence patterns for the nodes in each network.
4) Transform these convergence patterns into symbolic representations.
5) Select one division as the set of test data, and the others as the sets of training data.

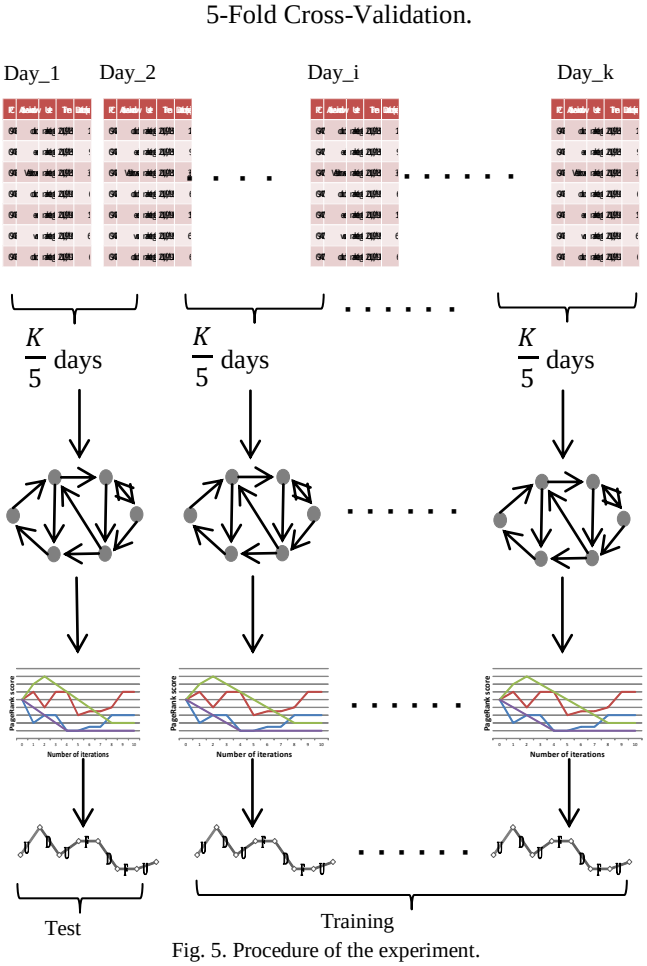


Fig. 5. Procedure of the experiment.

In this paper, we evaluated the discrimination performance based on 5-fold cross-validation.

Fig. 6. shows that the discrimination performance for all users with the average of AUC. The AUC values range from 0.5 to 1.0, where the value 0.5 indicates a random

discrimination and the value 1.0 indicates a perfect discrimination.

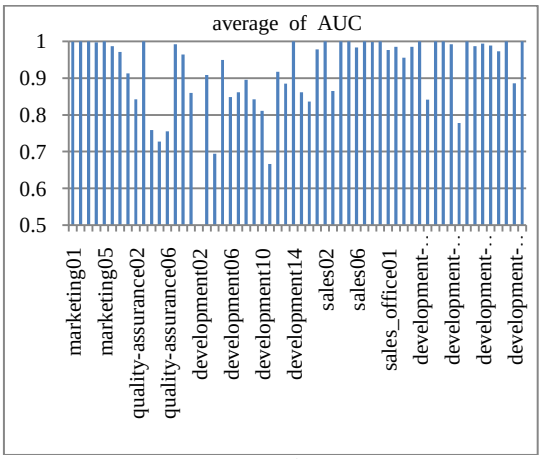


Fig. 6. Discriminative Performance with AUC.

The performances for the quality-assurance and developments are relatively low, while almost all of the results are high in other departments. We considered that the discrimination performance for quality-assurance is relatively low since the frequency of computer usage is lower than usage in the other departments. Hence, the proposed method is capable of discriminating user behavior if the employees commonly use computers at work.

V. Conclusions

In this paper, we aimed to analyze user behavior from PC operation logs by the similarity of PageRank convergence patterns. Similarity nodes of PageRank convergence patterns are accorded to their functions and roles in the network. In this paper, to discriminate between the user behavior of working and resting by analyzing the transitions of the active windows, we improved functionality clustering by transforming PageRank convergence patterns into symbolic representations of time series data, and applying edit distance to these representations. We showed that the improved method allows us to discriminate user behavior according to their context at work with high accuracy.

For the next step, we plan to apply various string kernels to measure the distance of symbolic representations of convergence patterns. And we will apply the proposed method to other kinds of networks, such as football pass networks and email transmission networks.

References

[1] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Physical Review E.*, vol. 70, no. 6, Dec. 2004.

[2] X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger, "SCAN: A Structural clustering algorithm for networks," in *Proc. the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007, pp. 824-833.

[3] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, pp. 75-174, 2010.

[4] Y. Zhou, H. Cheng, and J. X. Yu, "Graph clustering based on structural/attribute similarities," *PVLDB*, vol. 2, pp. 718-729, 2009.

[5] T. Fushimi, K. Saito, and K. Kazama, "Extracting communities in networks based on functional properties of nodes," in *Proc. PKAW*, 2012, pp. 328-334.

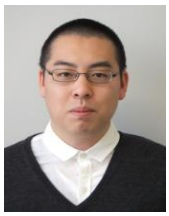
[6] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Proc. 7th International Conference on World Wide Web*, 1998, pp. 107-117.

[7] A. N. Langville and C. D. Meyer, *Google's Pagerank and Beyond: The Science of Search Engine Rankings*, Princeton University Press, 2006.

[8] R. Saito, T. Kuboyama, Y. Yamakawa, and H. Yasuda, "Understanding user behavior through summarization of window transition logs," in *Proc. DNIS*, 2011, pp. 162-178.

[9] J. L. , E. J. Keogh, S. Lonardi, and B. Y. chi Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proc. DMKD*, 2003, pp. 2-11.

[10] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," 2001.



K. Suzuki received a B.Eng from Tokyo Denki University in 2012. He is now a graduate student in the Department of Information System and Multimedia Design at Tokyo Denki University. His current research interests include pattern matching, data mining, and machine learning.

H. Yasuda received the B.E., M.E. and Dr.E. from the University of Tokyo, Japan in 1967, 1969, and 1972 respectively. Then, he had joined the Electrical Communication Laboratories of NTT in 1972. After serving twenty-five years (1972-1997), with the last position of vice president, director of NTT Information and Communication Systems Laboratories at Yokosuka, he left NTT and joined the University of Tokyo. He acted as director of The Center for Collaborative Research (CCR) for 2 years (2003-2005), and he is currently a professor in Tokyo Denki University.

K. Shin received his M.S. in mathematics from the University of Tokyo and his Ph.D. in computer science from the Research Center for Advanced Science and Technology of the University of Tokyo. He is currently a professor of Graduate School of Applied Informatics of University of Hyogo and an adjunct faculty at Carnegie Mellon University in US. The scope of his research activity includes public key cryptography, privacy protection, kernel method and feature selection.

T. Kuboyama received the B.Eng. and M.Eng. from Kyushu University in 1992 and 1994, and the Ph.D. from University of Tokyo in 2007. From 1997 to 2008, he was a research associate at Center for Collaborative Research, University of Tokyo. He is currently a professor at the Computer Centre of Gakushuin University. His current research interests include pattern matching, data mining, and machine learning.