Expert Systems With Applications 168 (2021) 114488

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

User profiling via application usage pattern on digital devices for digital forensics

Hongkyun Kwon [a], Sangjin Lee [a], Doowon Jeong [b,*]

[a] Institute of Cyber Security & Privacy (ICSP), Korea University, Seoul, 02841, South Korea
[b] College of Police and Criminal Justice, Dongguk University, Seoul, 04620, South Korea

ARTICLE INFO

Keywords:
User profiling
Digital forensics
Application usage
User similarity
Anomaly detection

ABSTRACT

In digital forensics, user profiling aims to predict characteristics of the user from digital evidence extracted from digital devices (e.g. smartphone, laptop, tablet). Previous researches showed promising results, but there are limitations to apply practical investigations. The researches so far have focused only on specific applications, devices, or operating systems by analyzing the order of execution or volatile data such as network traffic and online content. This paper introduces a user profiling method, named Entity Profiling with Binary Predicates (EPBP) model, which analyzes non-volatile data remained on digital devices. The proposed model defines that a user has two properties: *tendency* and *impact*, which indicate patterns of application usage. Based on the attributes, the EPBP model generates users' profiles and performs similarity analysis to differentiate between the users. We also present methods for clustering and anomaly detection through real case studies.

1. Introduction

As information and communication technology develops, billions of people use various types of applications. In addition, the advance of manufacturing mass storage has led the people to consume and produce a large amount of data. In this context, many researchers have studied user profiling, whose goal is to explore how applications are correlated with the user personal information and derive features to infer users' characteristics (Zhao et al., 2019). Since applications deal with heterogeneous and multi-source data, user profiling has been a challenge to researchers (Yu et al., 2019).

User profiling has been being studied by analyzing application usage data, recommended system, targeted advertisement service, and resource optimization program (Eke et al., 2019; Mahbub et al., 2019). User profiling is vital in many areas such as software engineering, business, social network, etc; in particular, it is very important in the digital forensics field. There are two main points for user profiling in digital forensics. First, by inferring user pattern from user profiling, investigators can detect anti-forensics behavior that is the practice of circumventing forensic analysis procedures making them unreliable or impossible such as file deletion, hiding, or data modification (Maggi et al., 2008; Qi et al., 2016). It is consistent with anomaly detection that deals with detecting unexpected events or abnormal behavior (Baddar et al., 2019). Second, the user profiling can enhance traditional forensic examination techniques. By analyzing the perpetrator's profile extracted from digital evidence, investigators can identify similar cases and accomplices. Also, they can anticipate possible crimes in the future and develop strategies to respond to possible crimes (Al Mutawa et al., 2015, 2016; Colombini & Colella, 2011; Steel, 2014; Warikoo, 2014).

The identification of user patterns is a key technique of user profiling, so it has been studied for various applications. The information parsed from digital device (e.g. smartphone (Choi & Lee, 2016; Lee et al., 2018; Mahbub et al., 2019), desktop (Mondal & Bours, 2017; Singh & Singh, 2018)) or online data (e.g. social network van Dam & Van De Velden, 2015; Shi et al., 2017) has been used for finding usage patterns. The previous studies have shown notable achievements by focusing on specific functions provided by applications. For example, the number of messages, call time, location information, or app usage time were used as a feature. Although the features are indicators that reveal user's characteristics, they have a disadvantage of being dependent on applications. Another challenge is the need for techniques that can be applied in heterogeneous devices. Nowadays, most users use multiple devices (e.g. desktop, laptop, smartphone, IoT device), so the device and application-independent method is required.

In this paper, we propose a method for identifying application usage patterns based on the nature of the user generating or consuming information. The contribution of the paper can be summarized as follows.

- We propose new features to identify application usage pattern that distinguishes users.

* Corresponding author.
E-mail addresses: hapez21@korea.ac.kr (H. Kwon), sangjin@korea.ac.kr (S. Lee), doowon@dgu.ac.kr (D. Jeong).

https://doi.org/10.1016/j.eswa.2020.114488
Received 12 August 2020; Received in revised form 7 November 2020; Accepted 8 December 2020
Available online 11 December 2020

- The proposed method is independent of a particular device or application, so it is possible to analyze between heterogeneous devices.
- Our method is used practically because data acquisition and preprocessing phases are based on practical forensic techniques.
- We propose a method to calculate the similarity between users, which can be used for user clustering.
- The experimental results show that the proposed method is also effective in anomaly detection.

The remainder of this paper is organized as follows: related works are presented in Section 2. In Section 3, we propose our profiling system with a newly developed model, named Entity Profiling with Binary Predicates (EPBP). Based on the model, methods for forensic analysis are introduced in Section 4. The methods are validated in Section 5 through real case studies. Finally, we give the conclusion and the future work in Section 6.

2. Related work

There are studies that analyze application usage patterns through real-time monitoring. Mizumura et al. (2018) analyzed cellular network traffic generated by smartphone applications to profile users. They collected upward and downward network traffic of application and background process. They extracted features from the traffic data and then classified the users by using machine learning models such as K-Nearest Neighbors, Decision Tree, Neural Network, etc. The approach showed promising results of classification, but it cannot be applied in digital forensics. The (Mizumura et al., 2018)'s method should monitor real-time network traffic, but forensic investigators cannot collect the traffics because most forensic investigation is conducted after cyber crime. Yang et al. (2020) proposed an application usage profiling algorithm to investigate online behavior. They represented usage data as Term Frequency–Inverse Document Frequency and then utilize the smoothed Gaussian Mixture Model for clustering. The paper also presented a case study with a national dataset from around 30,000 devices and applications. Their proposed methods showed promising results, but they are inapplicable to practical digital forensics because they use volatile data such as network traffic or online content. Ahn et al. (2014) and Lu et al. (2014) extracted usage patterns of mobile devices with real-time monitoring applications. Lu et al. designed a framework assuming that there are relations between user's moving locations and application launches. They created MASP database containing the stay locations from all users' GPS movement datasets, including mobile application usage database. User behavior patterns are the usage sequences of (`location, application`) in the MASP database. Ahn et al. analyzed usage patterns in three aspects: popularity ranking in Google Playstore, category to which the application belongs, and time domain. To acquire application usage, they installed SAMS (Smartphone Addiction Management System) client to collect data on users' smartphones. Based on the usage data, they compared the patterns of general users and smartphone addicts with two categories: usage duration and usage frequency of applications. These methods also showed high accuracy, but they are also inapplicable to digital forensics for the same reason as Mizumura et al. (2018).

Some studies proposed profiling methods based on text analysis. Matias Nicoletti and Silvia Schiaffino (2013) presented a user profiling method to detect topics of interest from users in informal conversations. They categorized messages using a category hierarchy that refers to Wikipedia's structured tree. Dhelim et al. (2020) presented a user interest mining system based on dynamic topic modeling and Big Five personality traits: openness to experience, agreeableness, conscientiousness, extraversion, and neuroticism. Both Nicoletti et al. and Dhelim et al. showed encouraging results in extracting topics from text, but digital forensics needs to handle binary data as well as text.

There are also studies on extracting characteristics of users by identifying the order of application execution. Shen and Shafiq (2019) studied to identify user profiling by analyzing mobile devices. They presented a deep reinforcement learning framework, named as DeepApp, which learns application usage behaviors of different users. The purpose of the method is to predict the apps that a user will open on a smartphone next. Mahbub et al. (2019) proposed a new formulation for active authentication. They focused on the smartphone and utilized application usage information to achieve low latency. Shen et al. and Mahbub et al. pursued real-time monitoring for business or authentication; this approach is unsuitable for forensic investigators who analyze devices with remaining digital evidence.

3. Profiling system

Our profiling system is divided into four phases (See Fig. 1). The first phase is to extract log data from the operating system. To collect information about the user's activity, the profiling system focuses on log data because logs are automatically recorded regardless of the user's intention. The logging methods are different in operation systems. For example, Windows records the log data to System Resource Usage Monitor (SRUM) (Khatri, 2015). In Android, the log data are stored as files that use *protobuf* format in *usagestats* directory (Pieterse et al., 2018). By collecting the log data of operating systems, process utilization is calculated. Log files recorded by applications are also collected and parsed. By analyzing the log files, information of application usage history, such as chat logs, web browsing, and e-mail exchanges, is identified. In this paper, methods for collecting and analyzing log data are not addressed because the methods have been studied in previous researches and there are many forensic tools that automate the methods, such as EnCase, AXIOM, FTK, etc.

The second phase is to create *activity statements* and organize *frames* for each application based on Entity Profiling with Binary Predicates (EPBP) model. The EPBP model that aims to create a profile for the user is newly proposed in this paper. In the EPBP model, the user's activity is described based on an activity statement that can be explained with 6 parameters such as subject, predicate, direct object, indirect object, process, and timestamp. The information of application usage history, acquired in the first phase, is used to create the activity statement. Activity statements constitute frames that are the unit of measurement for profiling. The definition of the activity statement and frame will be presented in Section 3.1.

The third phase is to configure a user profile. In this phase, two features considered to represent the user profile are presented: *tendency* and *impact*. The *tendency* indicates whether a user mainly generates or consumes information when using an application. The *impact* indicates how much information is generated or consumed by the user. As the EPBP model assumes that a user has different usage patterns for different applications, the two features are calculated for each application. The definition of *tendency* and *impact* will be presented in Sections 3.2 and 3.3. Then, we induce a user's distinct and unique patterns with computer simulations in Sections 3.4 and 3.5.

3.1. Representation of components used in the EPBP model

This section proposes a model named Entity Profiling with Binary Predicates (EPBP) that aims to create a profile for identifying an entity. This model assumes that the user's usage pattern differs between applications. To identify the distinct and unique patterns for each application, the EPBP model conducts probability and periodic analysis for the usage history.

In the EPBP model, an entity is defined as a process that satisfies the following requirements.

1. The entity is a unique unit.
2. One entity gives and receives data, such as text or image, to other entities through a communication channel called service.

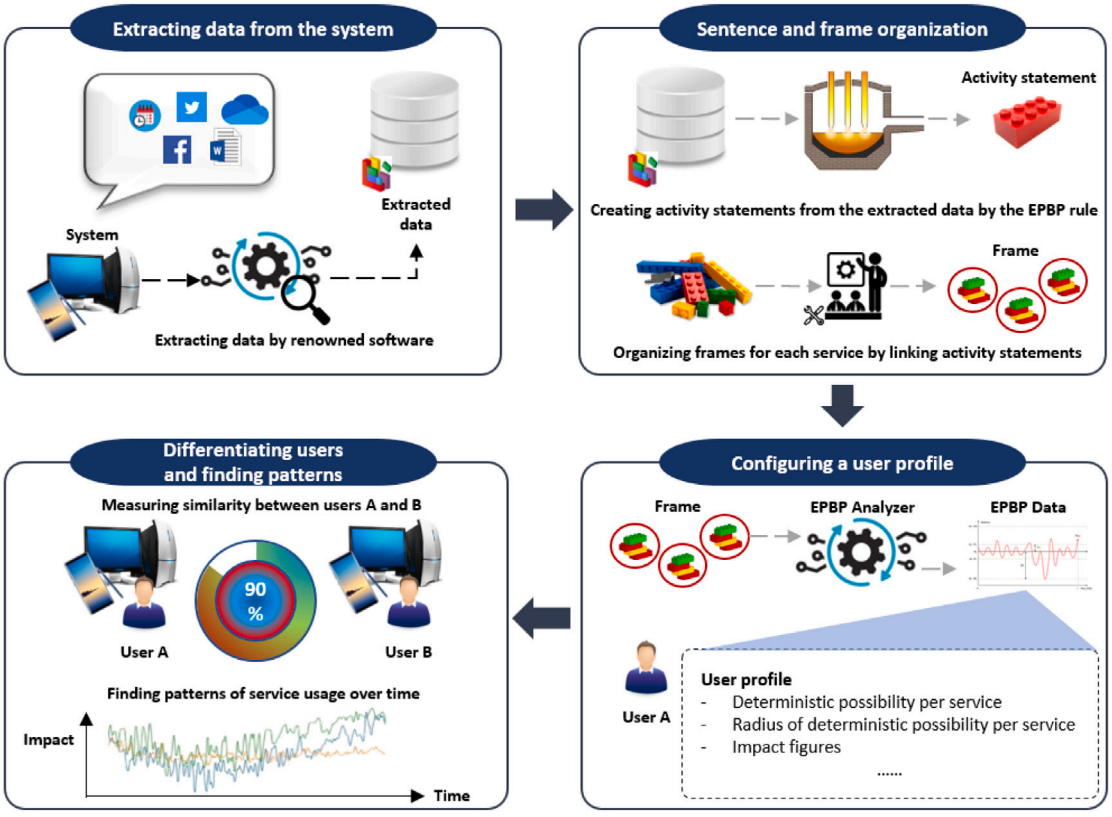

Fig. 1. An overview of our proposed profiling system.

3. The activity of the entity is described in terms of two predicates that have the antonym relationship; for example, 'send A to B' or 'receive A from B'.

The EPBP model defines a nonempty set of entities as a user. For example, as seen in Fig. 2, user A (E_A) uses a certain e-mail service to communicate with user B (E_B) and user C (E_C). e_{ij} means an entity j of user i, therefore, e_{a1} indicates entity 1 of user A. Because the user can use the certain e-mail service with multiple accounts, it is possible that several processes communicate with other processes through the same service. Fig. 2 shows that e_{a1} and e_{a2} communicate with e_{b0} and e_{c0}, respectively, through the same e-mail service.

The EPBP model creates a profile of the user based on activity statements with 6 parameters: subject, predicate, indirect object, direct object, process, and timestamp. The activity statement is expressed as follow:

subject predicate direct object indirect object via process at timestamp.

For instance, when user A sends interesting news to user B using a process of e-mail service on 1 November 2020, the activity statement is described as follow:

A sends interesting news to B via email process at 1 *November* 2020.

Frame F is composed of activity statements sorted by timestamp. F has a time axis, so it can be divided by time intervals. The divided frames are defined as interval frames. Fig. 3 describes the relationship with the interval frames and activity statements. $F^{(t)}$ denotes tth interval frame and each $F^{(t)}$ has several activity statements. The $I_{ij}(k,l)$ means that user i sends data to user j via process e_{ik} at sequence number l. F_k is the selection of the activity statements related to the entity k. $F_{(k,t)}$ means that tth interval frame of F_k.

3.2. Entity's attribute 1: Impact

The *impact* is a complex value combining three factors: the degree of entanglement between users, the number of statements in F, and the utilization rates of processes at time t. The degree of entanglement indicates how actively data exchanged between users. The high degree of entanglement expresses that the two sets of entities are interdependent in terms of data flow. The number of statements in F represents the amount of data entering and leaving the service through the system. The utilization rate is the frequency of entity usage. The high utilization rate is expected to be used frequently in sharing data. The section describes a method for estimating the *impact* based on the three factors.

As described in Section 3.1, F_i is a frame obtained by extracting only the activity statements related to entity i from F, and $F_{(i,t)}$ is the interval frame at t in F_i. $P_{(i,t)}$ is the number of activity statements that the entity i sends data to other users within $F_{(i,t)}$, and $S_{(i,t)}$ is the number of activity statements that the entity i receives data from other users within $F_{(i,t)}$. Two symbol $P_{(i,t)}$ and $S_{(i,t)}$ are derived from the predicates of 'publish' and 'subscribe' for each. At time t, the ratio of the P-S value of i is $r_{(i,t)}$ which is expressed as

$$r_{(i,t)} \equiv \frac{P_{(i,t)} + S_{(i,t)}}{\sum_{i=1}^{n} (P_{(i,t)} + S_{(i,t)})}. \tag{1}$$

where $r_{(i,t)}$ is always inside the closed interval $[0,1]$ on the $\mathbb{R}$.

In order to reflect entanglement between users, consider the coefficient representing the correlation between P-S at service i at t.

$$c_{(i,t)} \equiv \left| \frac{1}{P_{(i,t)} + 1} - \frac{1}{S_{(i,t)} + 1} \right| \tag{2}$$

where $c_{(i,t)}$ is always less than 1. When $P_{(i,t)}$ is close to $S_{(i,t)}$, $c_{(i,t)}$ approaches 0, and $c_{(i,t)}$ is 0 if and only if $P_{(i,t)}$ and $S_{(i,t)}$ are the same by its definition. Also, if one element goes to the infinity, $c_{(i,t)}$ converges to a specific value. The graph of $c_{(i,t)}$ is Fig. 4 where $P_{(i,t)} > 0$ *and* $S_{(i,t)} > 0$.

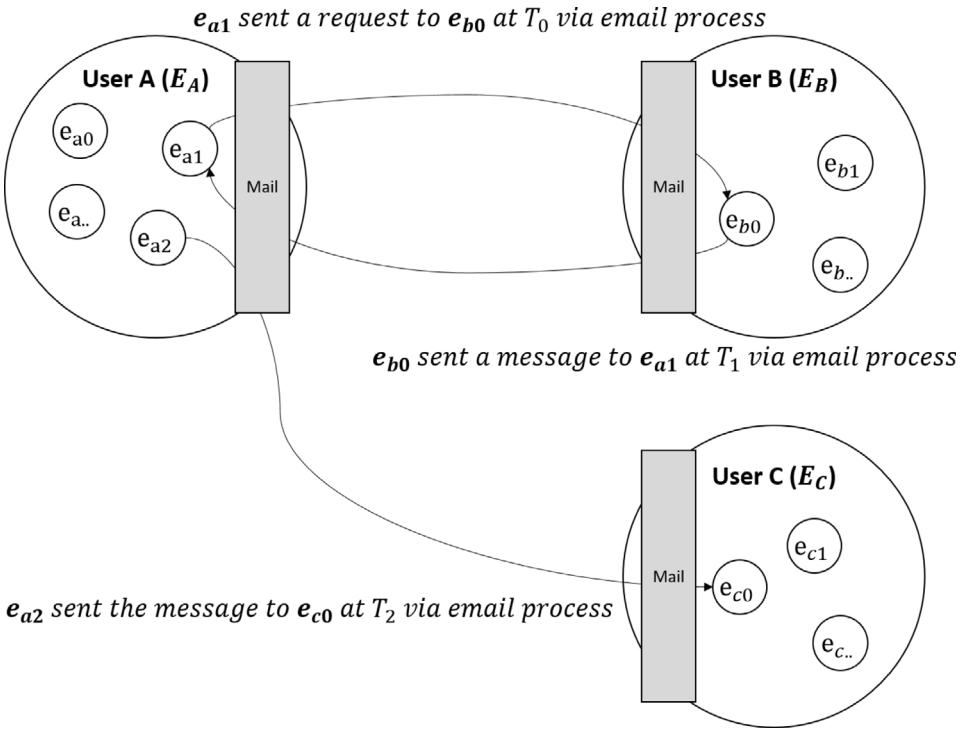

Fig. 2. A diagram for the EPBP model. The diagram shows the exchanging message and file at T via Mail between users.

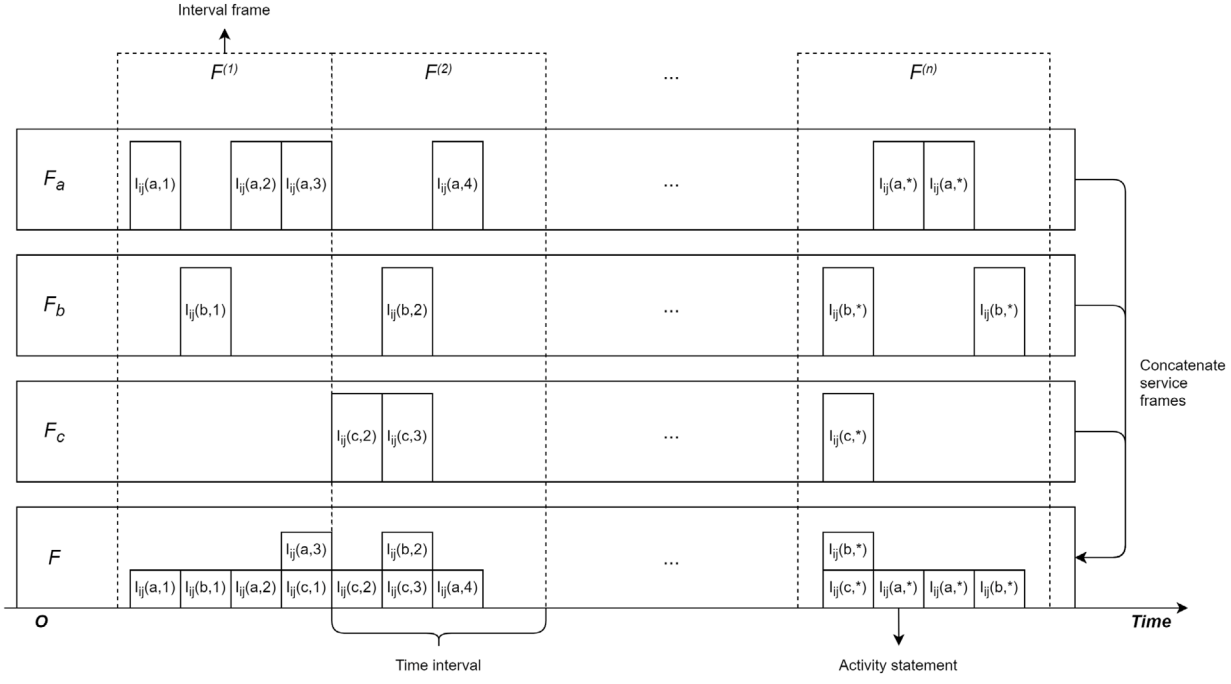

Fig. 3. The relationship between frames and activity statements.

The utilization rate for entity i at t is represented as $u_{(i,t)}$, which can be identified by analyzing activity logs. The utilization rate cannot exceed 100%, thus

$$\sum_{i=1}^{n} u_{(i,t)} \leq 1 \tag{3}$$

In the EPBP model, the *impact* of the entity i is computed using the following formula:

$$impact_{(i,t)} = \ln\left(\frac{u_{(i,t)} r_{(i,t)}}{c_{(i,t)}} + 1\right). \tag{4}$$

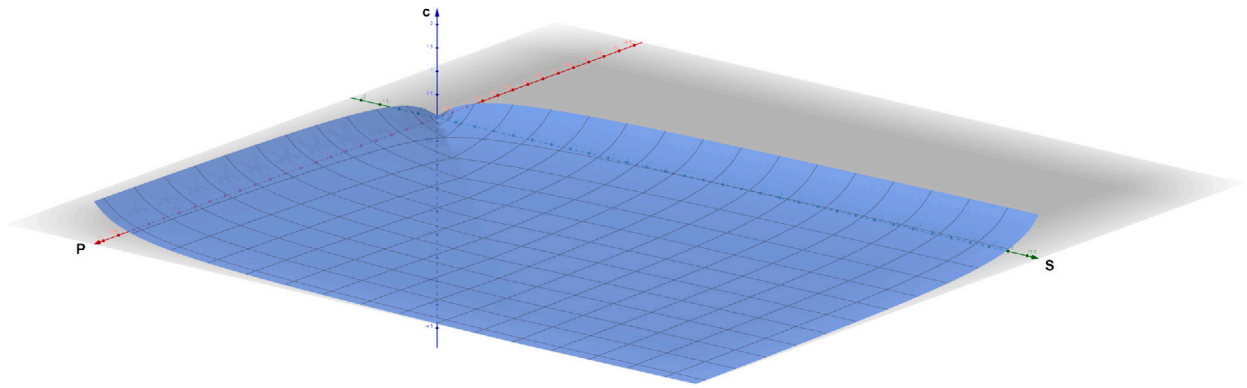

Fig. 4. The graph of $c_{(i,t)}$ where $P_{(i,t)} > 0$ *and* $S_{(i,t)} > 0$.

3.3. Entity's attribute 2: Tendency

An activity log such as event logs, database records, etc., designed to record who gave what to whom, can be converted into an activity statement. By analyzing the activity statements, it can be estimated how a user has used applications. We identify the user's tendency for each entity, in terms of production or consumption of information. Our model assumes that a unique usage pattern for each process (p_i) is observed from the tendency (td_i). This section describes the process of calculating the td_i and then estimating the p_i.

The p_i is the deterministic probability that the user publish data when using process i. It is based on the user's unique characteristics, so it cannot be directly measured. On the other hand, it is possible to estimate the probability that the user publish data at $F_{(i,t)}$, denoted by $p_{(i,t)}$. The $p_{(i,t)}$ meets only on the $F_{(i,t)}$, so the unmeasurable and stochastic properties of p inevitably cause the difference between p_i and $p_{(i,t)}$. To examine how close between $p_{(i,t)}$ and p_i, assume that there is a probability radius R_i where $\{R_i \mid 0 < R_i < 1, R_i \in \mathbb{R}\}$ over i, and $p_{(i,t)}$ lies in the open interval $(p_i - R_i, p_i + R_i)$.

To make these values nondimensionalized and normalized, a new value $td_{(i,t)}$ representing a *tendency* to generate information within $F_{(i,t)}$ is defined as follow:

$$td_{(i,t)} \equiv \frac{S_{(i,t)} - P_{(i,t)}}{P_{(i,t)} + S_{(i,t)}} \tag{5}$$

where two states $P_{(i,t)}$ and $S_{(i,t)}$ are simple count values classified by predicates. The $td_{(i,t)}$ always satisfies $\left|td_{(i,t)}\right| \leq 1$. The *tendency* of 'publish' and 'subscribe' is reflected by the sign. When $P_{(i,t)}$ is as close to $S_{(i,t)}$, $td_{(i,t)}$ is close to 0. $F_{(i,t)}$ is the sum of $P_{(i,t)}$ and $S_{(i,t)}$ by its definition. That is

$$F_{(i,t)} \equiv P_{(i,t)} + S_{(i,t)}. \tag{6}$$

The $td_{(i,t)}$ induces $p_{(i,t)}$ from (5) and (6) in the form of

$$td_{(i,t)} = \frac{(F_{(i,t)} - P_{(i,t)}) - P_{(i,t)}}{F_{(i,t)}} = \frac{F_{(i,t)} - 2P_{(i,t)}}{F_{(i,t)}} = 1 - \frac{2P_{(i,t)}}{F_{(i,t)}} = 1 - 2p_{(i,t)}. \tag{7}$$

The td_i can be obtained from (7) with p_i. The relation of the td_i and p_i is

$$td_i = 1 - 2p_i. \tag{8}$$

At the time point of t, let the difference between p_i and $p_{(i,t)}$ as r. The absolute value of r is inside the probability radius R_i by its definition.

$$\left|td_i - td_{(i,t)}\right| = \left|(1 - 2p_i) - (1 - 2p_{(i,t)})\right| = \left|2(p_{(i,t)} - p_i)\right| = |2r| \leq 2R_i \tag{9}$$

Thus, the *tendency* for entity i can be estimated as td_i within $2R_i$ of maximum error, thus p_i within R_i of maximum error.

The method to estimate the p_i and R_i, which are user's unique characteristics, is described experimentally in the following sections. First, we generate virtual dataset based on p_i and R_i, which are determined arbitrarily. Then, we study how to estimate the p_i and R_i from the generated virtual dataset.

3.4. Creating the virtual entity for validation

To construct a virtual frame F, we develop an algorithm that generates interval frames rather than generating activity statements one by one, inasmuch as p is the deterministic probability and Bernoulli trial. One service frame F_i consists of interval frames arranged in chronological order. The process of constructing a frame F is to concatenate these service frames in interval frame units (See Fig. 3).

There are six main parameters to configure F for a new virtual user A.

- *entity*: The entity e_i is defined in Section 3.1.
- $profile$: The profile of entities used by user A comprises two attributes p and R.
- $total\_days$: The $total\_days$ is the size of the time domain that user A has.
- *period*: When constructing the frame F, the *period* in the process of creating interval frames based on the EPBP model means sufficient interval time to reflect p from activity statements.
- *generator*: The *generator* yields variance to depict people. All generators must return a real number less than the absolute value 1.
- $impulse$: The $impulse$ is to emulate the behavior of data distortion caused by a sudden accident, intentional deletion, etc.

To construct interval frames with these parameters, we use two generators f and g comprising random functions of which output range is limited by a given radius. f yields a random real number for every *period* in the closed interval $[-R_i, R_i]$. R_m is the maximum rate at which $impulse$ affects data generation. g yields a random real number in the closed interval $[-R_m, R_m]$. The scope of the $impulse$ and the period to which the $impulse$ applies to vary from service to service. The Algorithm 1 is that we used to create interval frames for a virtual user. If the value of the $impulse\_cycle$ is less than 1, the algorithm does not interfere with the decision flow.

3.5. Inducing the entity's attributes

To find a method to estimate entity's attributes, we create virtual dataset based on virtual user A's profile. To generate virtual dataset, we set parameters such as p_i, *initial_PS Count*, R_i, etc. Table 1 shows the parameters used for dataset.

To obtain p_i from the created activity statements, we consider to use two numerical interpolation methods: Least Squares Method (LSM) and Huber Regression (HBR) (Huber, 1964). A linear function $W_i(t)$ obtained through the first-order interpolation method is calculated for each t of each data point. An average of the interval values is considered to be p'_i. $p'_{i,avg}$ is the average of p'_i that is obtained by configuring user A 100 times using the same parameters. As shown in Table 2,

Algorithm 1 Constructing frames per service

procedure GENERATEFRAME($e_i, p_i, R_i, R_m, f, g, initial\_PSCount, total\_days, period, impulse\_cycle$)
  $F_i \leftarrow []$
  $PSCount \leftarrow initial\_PSCount$
  $days \leftarrow 0$
  $cycle\_step \leftarrow 0$
  $weeks \leftarrow \lceil total\_days/period \rceil$
  while $days < weeks$ **do**
    $variance \leftarrow f(R_i)$ ▷ $|variance| \leq 1$
    $P \leftarrow (p_i + variance) * PSCount$
    $S \leftarrow (p_i - variance) * PSCount$
    $F_i[days] \leftarrow [P, S]$
    if $cycle\_step < impulse\_cycle$ **then**
      $PSCount \leftarrow P + S$
      $cycle\_step \leftarrow cycle\_step + 1$
    if $cycle\_step = impulse\_cycle$ **and** $impulse\_cycle \geq 1$ **then**
      $variance \leftarrow g(R_m)$ ▷ $|variance| \leq 1$
      $PSCount \leftarrow (1 + variance) * PSCount$
      $cycle\_step \leftarrow 0$
    $days \leftarrow days + 1$
  return F_i

Table 1
E_c's EPBP profile and model parameters.

App name	App1	App2	App3	App4
p_i	0.2	0.3	0.3	0.6
initial_PS Count	480	681	764	200
R_i	0.12	0.02	0.04	0.20
$R_{(m,i)}$	0.07	0.09	0.07	0.07
period	7	7	7	7
impulse_cycle	14	28	14	28
usage (%)	0.9	0.3	0.1	4.2

Table 2
The estimation result of deterministic possibility with linear interpolation methods.

App name	App1	App2	App3	App4
p_i	0.2	0.3	0.3	0.6
$p'_{i,avg}$ with LSM	0.209	0.299	0.302	0.604
$p'_{i,avg}$ with HBR	0.207	0.299	0.303	0.603
Diff of LSM	0.009	0.001	0.002	0.004
Diff of HBR	0.007	0.001	0.003	0.003

the difference between $p'_{i,avg}$ which is computed by the LSM and HBR method and the parameter value p_i of the model can be considered to be sufficiently close to values of less than 2 decimal places.

We measure the amplitude to estimate the probability radius R from F. The probability radius R'_i is the effective amplitude centered at p_i. R'_i is designed as a conservative value to avoid the excessive size of the interval. td_i is obtained by using (5) from p_i which is estimated through linear interpolation methods. R'_i is measured as the RMS (Root Mean Square) centered at td_i.

$$R'_i = \frac{1}{\sqrt{2}} \times \sqrt{\frac{\sum_{t=0}^{n-1}(td_i - td_{(i,t)})^2}{n}} \tag{10}$$

The reason for dividing by $\sqrt{2}$ is that R'_is are experimentally close to given parameter R_i when R_i is under 0.050. Table 3 shows the difference between R_i and R'_i.

The relationship between td_i, $td_{(i,t)}$, R_i, and R'_i is described in Fig. 5 based on time-delta (TD) and *tendency* (T). A user's *tendency*, determined by the user's unique characteristic p, is estimated by the measured values $td_{(i,t)}$. R_i, which represents the variation, is also estimated by the measurement R'_i.

Table 3
The difference of probability radius.

App name	App1	App2	App3	App4
R_i	0.120	0.020	0.040	0.200
R'_i	0.091	0.021	0.041	0.176
Diff of RMS	0.039	0.001	0.001	0.024

Table 4
EPBP profile table of user A, B, and virtual user C (*period* = 7).

User	Profile	App1	App2	App3	App4
A	Usage (%)	14.561	4.445	0.115	14.105
	PS-Count	1020	630	5	264
	p_i with LSM	0.280	0.032	0.000	0.443
	p_i with HBR	0.275	0.025	0.000	0.439
	R_i with LSM-RMS	0.180	0.053	0.000	0.170
	R_i with HBR-RMS	0.180	0.054	0.000	0.170
B	Usage (%)	14.561	2.763	0.114	12.423
	PS-Count	527	4182	9	375
	p_i with LSM	0.313	0.087	1.000	0.431
	p_i with HBR	0.302	0.086	1.000	0.431
	R_i with LSM-RMS	0.345	0.079	0.000	0.104
	R_i with HBR-RMS	0.345	0.079	0.000	0.104
C	Usage (%)	14.561	4.445	0.115	14.105
	PS-Count	858	512	5	210
	p_i with LSM	0.273	0.023	0.000	0.450
	p_i with HBR	0.261	0.020	0.000	0.427
	R_i with LSM-RMS	0.192	0.037	0.000	0.157
	R_i with HBR-RMS	0.192	0.037	0.000	0.160

4. Forensic analysis with the EPBP model

In this section, we present an investigation method to distinguish users with users' profiles by using the EPBP model. To describe our method, we analyze real data as an example. We extract 4 application data from 2 smartphones used by different users A and B. The 4 applications are used for social networks (App1), SMS/MMS (App2), email (App3), and contact (App4). Then, to introduce a method for similarity detection, we create virtual data by delete user A's 1∼2 week(s) data in the middle of each month. We regard the generated data as being created by a virtual user C.

4.1. Interpreting EPBP parameters

Tables 4 and 5 are based on EPBP profile data which are *period* = 7 and *period* = 28 respectively. There is no dramatic change in p_i, although the *period* is different. As seen in App3 case, the EPBP profile between two users can be as extreme as App3, if there are a small number of activity statements (PS-Count). The value of p_3 is 0.000 for A and 1.000 for B, indicating that they are on the anode.

It is observed that R_i is in inverse proportion to the *period*. Because the R stands for a degree of variability in the p, it is natural that the greater the *period*, the smaller the R. On the other hand, increasing the *period* results in data loss about fluctuations and convergence errors; it may lead to the over-fitting problem or lack of information. As seen in Table 5, p and R of App4 cannot be identified, because the *period* is too long to get enough information to calculate the entity's attributes.

In terms of application usage, we report the applications as follows:

- App1: It is a service that accounts for about 15% of the utilization rate in the entire system, and is a highly important service to both users. p_1 values of users are close to 0.3. That is, users have **often** consumed information with a high utilization rate by using App1.
- App2: It is a service that accounts for about 4% of the utilization rate in the entire system. p_2 of users are less than 0.1. That is, users have **almost always** consumed information with a high utilization rate. In the aspect that App2 leaves enormous records for user B that is about 8 times more data than App1, it is determined that user B has used App2 mainly to consume information.

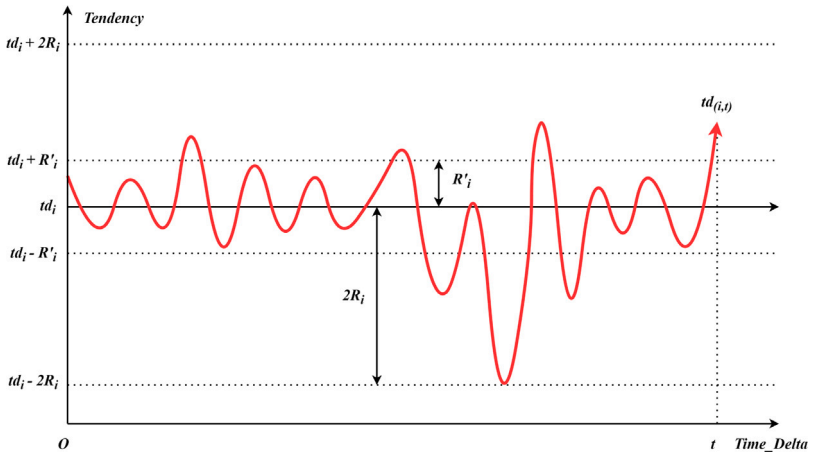

Fig. 5. TD-T projection graph of td_i, $td_{(i,t)}$, R_i, and R'_i.

Table 5
EPBP profile table of user A, B, and virtual user C (*period* = 28).

User	Profile	App1	App2	App3	App4
A	Usage (%)	14.561	4.445	0.115	14.105
	PS-Count	1020	630	5	264
	p_i with LSM	0.285	0.044	0.000	0.437
	p_i with HBR	0.245	0.049	0.000	0.437
	R_i with LSM-RMS	0.081	0.016	0.000	0.011
	R_i with HBR-RMS	0.099	0.017	0.000	0.011
B	Usage (%)	14.561	2.763	0.114	12.423
	PS-Count	527	4182	9	375
	p_i with LSM	0.315	0.094	1.000	0.419
	p_i with HBR	0.325	0.096	1.000	N.A.
	R_i with LSM-RMS	0.119	0.079	0.000	0.000
	R_i with HBR-RMS	0.119	0.078	0.000	N.A.
C	Usage (%)	14.561	4.445	0.115	14.105
	PS-Count	628	512	5	210
	p_i with LSM	0.280	0.027	0.000	0.411
	p_i with HBR	0.236	0.019	0.000	0.411
	R_i with LSM-RMS	0.090	0.020	0.000	0.000
	R_i with HBR-RMS	0.109	0.023	0.000	0.000

- App3: It is a service that accounts for about 0.1% of the regular utilization rate. User A has **always** produced information using App3. On the other hand, user B has **always** used to consume information.
- App4: It is a service that accounts for about 15% of the utilization rate, and is a highly important service to users. p_4 of users are in the open interval (0.410, 0.450). That is, users have consumed or produced information with **almost half** probability.

4.2. Profiling with p and R

To calculate the similarity between users, we use a method of calculating the degree of overlap between regions of users' profiles. Let A as an open interval with a radius of A's $R_{(i)}$ centered at A's $p_{(i)}$, and let B as an open interval with a radius of B's $R_{(i)}$ centered at B's $p_{(i)}$. Then, the overlapped section is $A \cap B$ in the i. The degree of overlap in the probability domain, which is a one-dimensional real space, is obtained by

$$D.O. = \frac{len(A \cap B)}{len(A \cup B)}. \tag{11}$$

The D.O. indicates a degree of similarity between users. Table 6 shows that the similarity between the user A and B is 54% when *period* = 7. D.O. in Table 7 exceeds 85%, because user C is a virtual user based on user A, as mentioned above. However, if *period* = 28, D.O. between the user A and C is about 42%. It is due to App4's lack of information described in Section 4.1. On the other hand, it is observed that D.O. of App1 and App2 can be used for distinguishing users. This analysis implies that user profiling is possible if the period is properly set to calculate the p and R values.

Table 6
Degree of overlap between A and B.

Period	App1	App2	App3	App4	D.O.
7	0.547	0.463	N.A.	0.612	0.541
28	0.680	0.203	N.A.	0.000	0.441

Table 7
Degree of overlap between A and C.

Period	App1	App2	App3	App4	Avg (D.O).
7	0.938	0.706	N.A.	0.924	0.856
28	0.900	0.358	N.A.	0.000	0.419

4.3. Tendency-impact analysis

4.3.1. TD-I projection

Entity modeled by the EPBP also has an *impact* attribute, which includes the intensity of use and the degree of entanglement with other entities. We generate 3D plots, named TD-I projection, with 3 coordinate axes: time-delta (TD), *tendency* (T) and *impact* (I). TD-I projection is used to identify the user's dependence on the application.

TD-I projections of the user A and B are shown in Fig. 6. The user A and B are clearly distinguished in terms of *impact*. User A is more consistent than user B in that user B has more intersections than user A between App1 and App2. User B tends to collect and distribute data using multiple channels than user A. We discuss the TD-I projections as follow:

- App1: Both user A and B generally remain high *impact* above 5. Especially for user B, the abnormal interval is detected. TD-I projection describes that valid data plunges between 100 and 125 days. It is assumed that user B did not use App1 during the period or deleted the data intentionally.
- App2: User A has used applications consistently, compared to user B.
- App3: The *impact* of both users is less than 1. User B shows a very short duration when using App3. It is estimated that both users rarely use the email application on their smartphone.
- App4: The *impact* of both users has remained high above 5. On the other hand, the *impact* graph of user B shows only the last short section. It means that there is no previous application data;

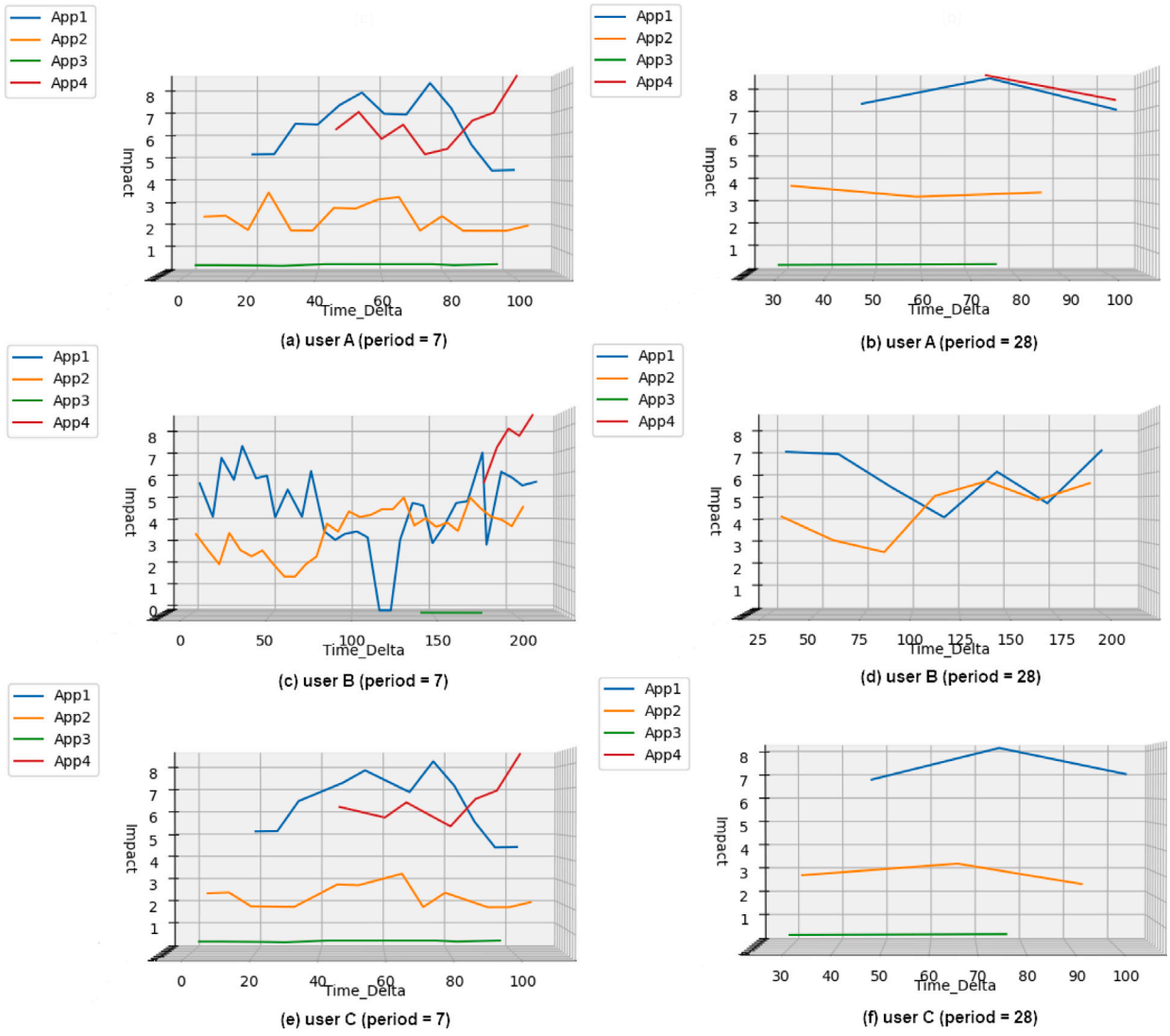

Fig. 6. TD-I projections.

it was revealed that user B had intentionally deleted past data from App4 for anti-forensics.

On the other hand, as seen in Fig. 6, the overall usage trends of user A and C are similar. Although the data of virtual user C is generated by deleting some of the data of user A, the TD-I projection, which indicates the intensity of use, is not much different because *impact* is calculated by frame-by-frame. However, *period* can affect TD-I projection. Remind that the D.O. score between user A and C is 0.856 when $period = 7$, and 0.419 when $period = 28$, as described in Section 4.2.

4.3.2. T-I triangle

The *tendency-impact* triangle (T-I triangle) is used to graphically visualize the degree of overlap between users. In the T-I triangle, 3 points are marked: $T_{(i,center)}$, $T_{(i,left)}$, and $T_{(i,right)}$. The equation for the points are as follows:

$$T_{(i,center)} = (td_i, impact_i) \tag{12}$$

$$T_{(i,left)} = (td_i - \frac{P_i}{P_i + S_i}, 0) \tag{13}$$

$$T_{(i,right)} = (td_i + \frac{S_i}{P_i + S_i}, 0) \tag{14}$$

Table 8
Distance from user A.

User	App1	App2	App3	App4	Average distance
B	1.674	0.069	1.277	2.152	1.293
C	0.246	0.111	0.0	0.218	0.144

The sign of td_i indicates the tendency of the user to generate and consume information. If td_i is positive, it means that the user mainly receives data using the entity *i*. The ratio of left and right widths around the center axis in the triangle means $P_i : S_i$. As seen in Fig. 7, App1 and App2 triangles of user A partially overlaps and those of user B. On the other hand, triangles of user A and C are quite similar; it means that their usage patterns are also similar (See Fig. 8).

To quantify the similarity, we calculate Euclidean distance between the points $T_{(i,center)}$ of each user. Table 8 shows the distance between the users. The average distance indicates that user A and C have a similar usage pattern, compared to user B. Based on the distance, multiple users can be clustered; it is represented in Section 5.

5. Case study

This section represents two case studies: user clustering and anomaly detection. We collected the application data from 20 devices

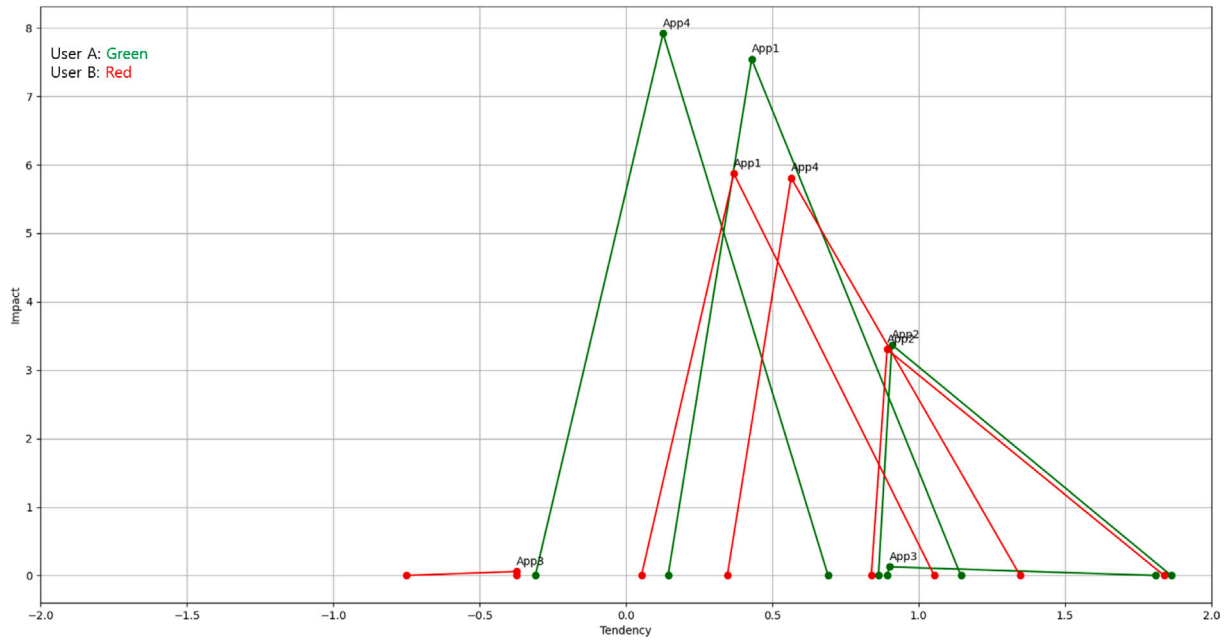

Fig. 7. T-I triangles of user A and B.

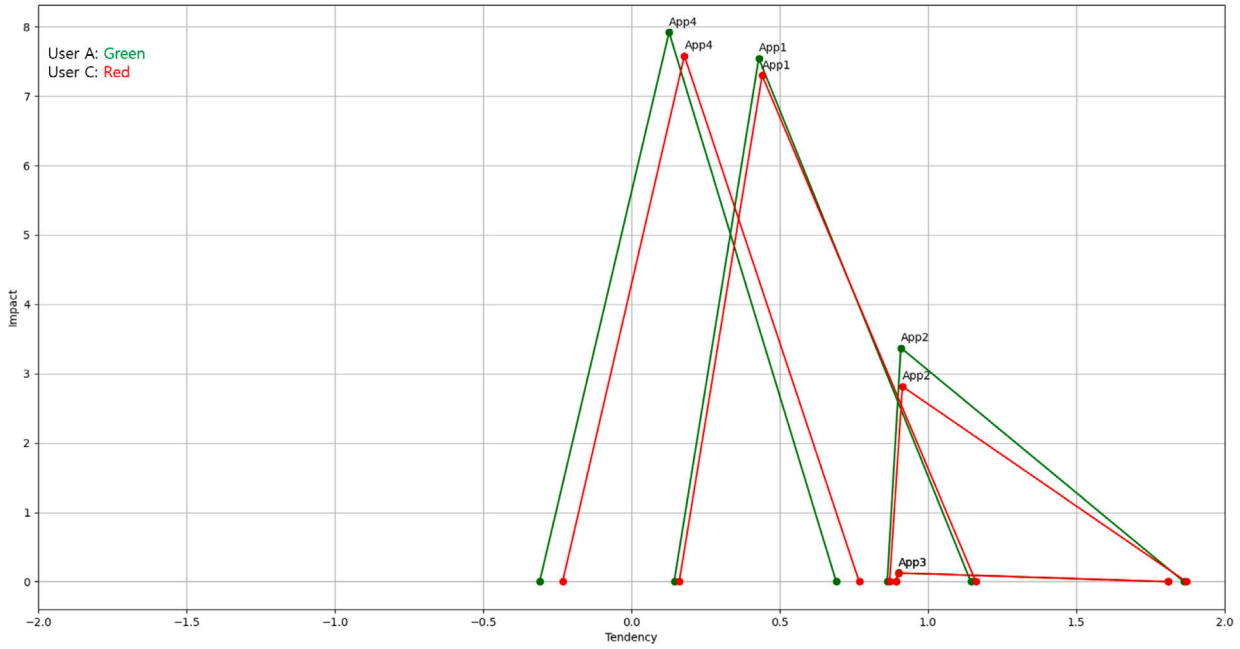

Fig. 8. T-I triangles of user A and C.

that were investigated during an internal audit. There were 12 Samsung smartphones, 3 LG smartphones, 3 Apple iPhones, and 2 Apple tablet PCs. The application data was extracted by MD-NEXT.[1] Then, we parsed the application data using AXIOM.[2] Based on the parsed data, we configured activity statements for two applications mainly used in the company; in this paper, we name the applications App1 and App2.

5.1. Clustering

In the digital forensics field, clustering is used to find accomplices or persons involved in the case (Kebande et al., 2018). We clustered the 20 users through the Affinity Propagation (Brendan J. Frey, 2007) based on the Euclidean distances between the users, as introduced in Section 4.3.2. Table 9 shows the result of the clustering. Through the investigation, it was observed that 4 users (01, 17, 19, and 20) had deleted App2, so it was impossible to measure the users' *td* and *impact* of App2. Profile label of Table 9 is the result of clustering considering both App1 and App2. The users that the distance is 0 are cluster representatives.

[1] http://www.hancomgmd.com/product/ (last accessed 6 November 2020).
[2] https://www.magnetforensics.com/products/magnet-axiom/ (last accessed 6 November 2020).

Table 9
The result of clustering for 20 users.

User no	Profile		App1		App2	
	Label	Distance	*td*	*impact*	*td*	*impact*
01	–	–	0.757	2.635	(null)	(null)
02	0	0.603	0.167	0.283	−0.299	8.573
03	2	0.074	0.728	0.384	0.779	1.730
04	3	0.548	0.008	5.511	0.974	2.544
05	2	0.795	0.941	2.956	0.333	0.909
06	1	0.326	0.429	2.600	0.603	5.026
07	2	0.105	0.330	3.410	0.977	2.243
08	1	0.391	−0.111	2.348	0.780	3.005
09	0	0	0.600	1.549	0.189	8.446
10	1	0	−0.217	2.990	0.908	2.974
11	2	0.924	0.068	2.000	0.778	1.638
12	1	0.129	0.900	2.668	0.424	3.526
13	2	0	0.310	3.477	0.879	1.983
14	3	0.434	0.200	6.056	0.837	3.173
15	3	0.911	0.210	8.408	0.827	2.382
16	1	0.756	−0.114	4.012	0.852	3.503
17	–	–	0.260	3.324	(null)	(null)
18	3	0	0.665	6.260	0.953	2.640
19	–	–	0.429	6.015	(null)	(null)
20	–	–	0.292	4.892	(null)	(null)

The usage patterns are divided into 4 labels. To understand the characteristics of each label, the application data of the users included in the label was identified. Each label has the following characteristics:

- Label 0: Users deleted a lot of data on App2. The *impact* is relatively large because the utilization rate of App2 is high but the number of the remaining data is small.
- Label 1: Users generally received shorter messages than the sent messages. They used the applications only for company business. Some of them were responsible for delivering notices to other employees using App1, so their *td*s of App1 were negative.
- Label 2: Users also exchanged task-oriented messages. Compared to Label 1, they received long messages from others including users of Label 1.
- Label 3: Most users received data from App2. There is an unusually large number of notice messages (e.g. credit card usage, shopping mall promotion).

The reason why many users' *td* is larger than zero is that they received a lot of notification messages from others. A *td* value close to zero indicates that the user actively communicated with others or intentionally deleted past messages. In fact, it was found that the users in Label 0 intentionally deleted some messages on App2 to hide their corruption, as a result of the actual audit. It is forensically significant to identify a group of users that conducted anti-forensics behaviors. A case for analyzing a user's pattern and detecting the anti-forensics is described in the following sections.

5.2. Anomaly detection

The TD-T projection graph, introduced in Section 4.3.2, is used for anomaly detection. The *tendency* implies a distinct usage pattern, so its abrupt change indicates that abnormal behavior has occurred. For an experiment, we drew a TD-T projection graph for a user who is suspected of deleting application data intentionally.

As seen in Fig. 9, *tendency* sharply drops in time 600~700. Based on the observation, we identified the application data showing an anomaly pattern. As a result, it was identified that some data had been deleted. Also, some messages containing phrases that the original user does not usually use were found. The experimental result shows that abnormal behavior, such as data destruction, unauthorized use, or suspicious behavior of the user, can be detected by using the EPBP model.

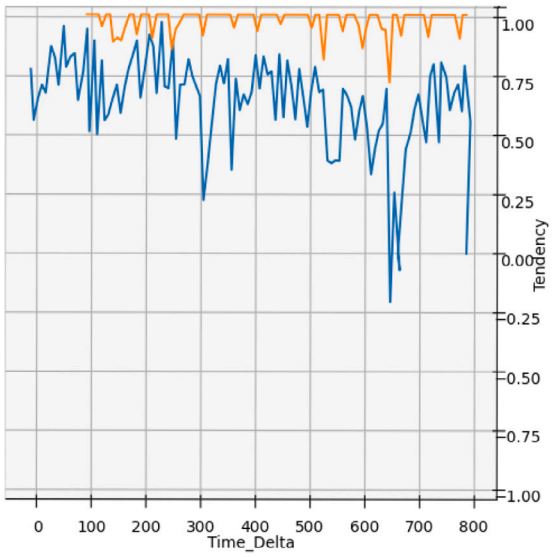

Fig. 9. TD-T projection graph for anomaly detection.

6. Conclusions and future work

In this paper, we presented the Entity Profiling with Binary Predicate (EPBP) model that creates the EPBP profile by finding the properties of the entity in given datasets. In the EPBP model, *tendency* and *impact* were proposed to extract application usage patterns of users. Based on the features, TD-T projection, TD-I projection, and T-I triangle for each application were drawn to distinguish users. Methods to identify similar users and abnormal behaviors were also introduced with real datasets. Through case studies, we verified that proposed methods can be applied to practical forensic investigations.

We focused not on the content of messages obtained from digital devices, but on the fact that the messages were exchanged. Thus, future studies will the development of more comprehensive methods that consider the intention of the user who exchanged the messages. In addition to the analysis of digital evidence, we also will develop methods to apply the EPBP model to real-time monitoring for anomaly detection.

CRediT authorship contribution statement

Hongkyun Kwon: Conceptualization, Methodology, Validation, Investigation, Data curation, Writing - original draft, Writing - review & editing, Visualization. **Sangjin Lee:** Conceptualization, Supervision, Project administration, Writing - original draft, Writing - review & editing. **Doowon Jeong:** Conceptualization, Methodology, Validation, Investigation, Data curation, Writing - original draft, Writing - review & editing, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Ahn, H., Wijaya, M. E., & Esmero, B. C. (2014). A systemic smartphone usage pattern analysis: focusing on smartphone addiction issue. *International Journal of Multimedia Ubiquitous Engineering, 9*, 9–14.

Al Mutawa, N., Bryce, J., Franqueira, V. N., & Marrington, A. (2015). Behavioural evidence analysis applied to digital forensics: an empirical analysis of child pornography cases using p2p networks. In *2015 10th international conference on availability, reliability and security* (pp. 293–302). IEEE.

Al Mutawa, N., Bryce, J., Franqueira, V. N., & Marrington, A. (2016). Forensic investigation of cyberstalking cases using behavioural evidence analysis. *Digital Investigation*, *16*, S96–S103.

Baddar, S. A.-H., Merlo, A., & Migliardi, M. (2019). Behavioral-anomaly detection in forensics analysis. *IEEE Security & Privacy*, *17*(1), 55–62.

Brendan J. Frey, D. D. (2007). Clustering by passing messages between data points. *Science*, 972–976.

Choi, J., & Lee, S. (2016). A study of user relationships in smartphone forensics. *Multimedia Tools and Applications*, *75*(22), 14971–14983.

Colombini, C., & Colella, A. (2011). Digital profiling: A computer forensics approach. In *International conference on availability, reliability, and security* (pp. 330–343). Springer.

van Dam, J.-W., & Van De Velden, M. (2015). Online profiling and clustering of facebook users. *Decision Support Systems*, *70*, 60–72.

Dhelim, S., Aung, N., & Ning, H. (2020). Mining user interest based on personality-aware hybrid filtering in social networks. *Knowledge-Based Systems*, *206*, Article 106227.

Eke, C. I., Norman, A. A., Shuib, L., & Nweke, H. F. (2019). A survey of user profiling: state-of-the-art, challenges, and solutions. *IEEE Access*, *7*, 144907–144924.

Huber, P. J. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, *35*(1), 73–101.

Kebande, V. R., Karie, N. M., Wario, R. D., & Venter, H. (2018). Forensic profiling of cyber-security adversaries based on incident similarity measures interaction index. In *2018 international conference on intelligent and innovative computing applications (ICONIC)* (pp. 1–6). IEEE.

Khatri, Y. (2015). Forensic implications of system resource usage monitor (srum) data in windows 8. *Digital Investigation*, *12*, 53–65.

Lee, Y., Park, I., Cho, S., & Choi, J. (2018). Smartphone user segmentation based on app usage sequence with neural networks. *Telematics and Informatics*, *35*(2), 329–339.

Lu, E. H.-C., Lin, Y.-W., & Ciou, J.-B. (2014). Mining mobile application sequential patterns for usage prediction. In *2014 IEEE international conference on granular computing (GrC)* (pp. 185–190). IEEE.

Maggi, F., Zanero, S., & Iozzo, V. (2008). Seeing the invisible: Forensic uses of anomaly detection and machine learning. *SIGOPS Operating Systems Review*, *42*(3), 51–58.

Mahbub, U., Komulainen, J., Ferreira, D., & Chellappa, R. (2019). Continuous authentication of smartphones based on application usage. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, *1*(3), 165–180.

Matias Nicoletti, & Silvia Schiaffino, D. G. (2013). Mining interests for user profiling in electronic conversations. *Expert Systems with Applications*, *40*, 638–645.

Mizumura, N., Nkurikiyeyezu, K., Ishizuka, H., Lopez, G., & Tobe, Y. (2018). Smartphone application usage prediction using cellular network traffic. In *2018 IEEE international conference on pervasive computing and communications workshops (PerCom Workshops)* (pp. 753–758). IEEE.

Mondal, S., & Bours, P. (2017). Person identification by keystroke dynamics using pairwise user coupling. *IEEE Transactions on Information Forensics and Security*, *12*(6), 1319–1329.

Pieterse, H., Olivier, M., & Van Heerden, R. (2018). Smartphone data evaluation model: Identifying authentic smartphone data. *Digital Investigation*, *24*, 11–24.

Qi, Z., Xiang, C., Ma, R., Li, J., Guan, H., & Wei, D. S. (2016). Forenvisor: A tool for acquiring and preserving reliable data in cloud live forensics. *IEEE Transactions on Cloud Computing*, *5*(3), 443–456.

Shen, J., & Shafiq, M. (2019). Learning mobile application usage-a deep learning approach. In *2019 18th IEEE international conference on machine learning and applications (ICMLA)* (pp. 287–292). IEEE.

Shi, L.-L., Liu, L., Wu, Y., Jiang, L., & Hardy, J. (2017). Event detection and user interest discovering in social media data streams. *IEEE Access*, *5*, 20953–20964.

Singh, B., & Singh, U. (2018). Program execution analysis in windows: A study of data sources, their format and comparison of forensic capability. *Computers & Security*, *74*, 94–114.

Steel, C. M. (2014). Idiographic digital profiling: Behavioral analysis based on digital forensics. *Journal of Digital Forensics, Security and Law*, *9*(1), 1.

Warikoo, A. (2014). Proposed methodology for cyber criminal profiling. *Information Security Journal: A Global Perspective*, *23*(4–6), 172–178.

Yang, J., Ma, J., & Howard, S. K. (2020). Usage profiling from mobile applications: A case study of online activity for Australian primary schools. *Knowledge-Based Systems*, *191*, Article 105214.

Yu, Z., Du, H., Yi, F., Wang, Z., & Guo, B. (2019). Ten scientific problems in human behavior understanding. *CCF Transactions on Pervasive Computing and Interaction*, *1*(1), 3–9.

Zhao, S., Li, S., Ramos, J., Luo, Z., Jiang, Z., Dey, A. K., & Pan, G. (2019). User profiling from their use of smartphone applications: A survey. *Pervasive and Mobile Computing*, Article 101052.