Computers in Human Behavior 30 (2014) 583–594

Contents lists available at ScienceDirect

Computers in Human Behavior

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

Understanding users' behavior with software operation data mining

Stella Pachidi[a,b,*], Marco Spruit[a], Inge van de Weerd[b]

[a] Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands
[b] Information, Logistics and Innovation Department, VU University Amsterdam, De Boelelaan 1105, 1081HV Amsterdam, The Netherlands

ARTICLE INFO

Article history:
Available online 21 August 2013

Keywords:
Software usage
User behavior
Software operation knowledge
Software analytics
Log data
Data mining

ABSTRACT

Software usage concerns knowledge about how end-users use the software in the field, and how the software itself responds to their actions. In this paper, we present the Usage Mining Method to guide the analysis of data collected during software operation, in order to extract knowledge about how a software product is used by the end-users. Our method suggests three analysis tasks which employ data mining techniques for extracting usage knowledge from software operation data: users profiling, clickstream analysis and classification analysis. The Usage Mining Method was evaluated through a prototype that was executed in the case of Exact Online, the main online financial management application in the Netherlands. The evaluation confirmed the supportive role of the Usage Mining Method in software product management and development processes, as well as the applicability of the suggested data mining algorithms to carry out the usage analysis tasks.

1. Introduction

Software usage concerns the utilization of a software product by the end-users. Software usage data may be collected while the end-users are using the software in the field (El-Ramly & Stroulia, 2004). Simmons (2006) points out the possibility to extract system requirements from usage, rendering the beneficial role of user experience in product innovation and differentiation. Software usage knowledge includes the awareness of how end-users use the software in the field, and how the software itself responds to their actions (Van der Schuur, Jansen, & Brinkkemper, 2010).

By tracking software usage, we can monitor which applications are most often used, which features are underutilized, and which functionalities could be expanded (Junco, 2013). This information could for example be used to highlight changes in the requirements engineering process. We may also gain insights on how users browse themselves through the user interface in order to perform an operation, with the goal to improve software usability or to reengineer processes. Furthermore, by observing the usage behavior of different customer profiles, the software vendor can implement more directed marketing or customized licensing (Germanakos, Tsianos, Lekkas, Mourlas, & Samaras, 2008; van der Schuur et al., 2010). Improved customer satisfaction, and consequently customer retention and increase in sales, are some of the business advantages that could be gained through an automated usage analysis, based on real execution data.

Software usage knowledge may be extracted from *software operation data*, i.e. data that are collected during software operation in the field (van der Schuur et al., 2010). A noticeable amount of research has already been performed in the process of recording software operation data (Bowring, Orso, & Harrold, 2002; Nusayr & Cook, 2009). In practice, most vendors tend to handle the acquired data manually, or use general statistics and simple visualization techniques (Kristjansson & Van der Schuur, 2009). However, such analysis cannot yield interesting patterns that are hidden in large datasets (Kantardzic, 2002).

On the other hand, a lot of development has been seen in the web usage mining field (Cooley, Mobasher, & Srivastava, 1997). Although many lessons can be learned from there, the approach for analyzing web usage by website visitors has significant differences, compared to analyzing how software products are used by the users. The techniques that are used in web usage mining (and other related domains) need to be revised for their application in mining usage on software operation data.

While usage knowledge is highly important for making good software products, the rise of cloud computing and Software-as-a-Service (SaaS) applications (Park & Ryoo, 2013) creates an opportunity to mine the easily acquired data. Even though there are algorithms for doing such data analysis, they are hardly ever used for analyzing software usage. Following a meta-algorithmic approach, we will try to answer the research question:

How should we inspect software operation data, in order to gain knowledge about how the software is used by the end-users?

* Corresponding author at: Information, Logistics and Innovation Department, VU University Amsterdam, De Boelelaan 1105, 1081HV Amsterdam, The Netherlands. Tel.: +31 644478898.

E-mail addresses: s.pachidi@vu.nl (S. Pachidi), m.r.spruit@uu.nl (M. Spruit), i.vande.weerd@vu.nl (I. van de Weerd).

0747-5632/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.chb.2013.07.049

This research suggests how data mining techniques can be integrated to analyze software operation data in a uniform and automated way. Hence, it contributes to the domain of software usage analysis as well as to the software operation knowledge and its use in software product management, development and maintenance processes (Van der Schuur et al., 2010). From a practical perspective, the method that we suggest for usage mining constitutes a reference process model that can be followed by software vendors, to analyze how their customers use their products.

The remainder of this paper has the following structure: In Section 2 we review the research that has been performed on the area of extracting usage knowledge from the system utilization. We shortly present our research design in Section 3. In Section 4 we present the method that has been constructed to extract usage knowledge. In Section 5 we describe the usage knowledge subjects that we suggest to extract, and the variables that should be inspected in software operation data, in order to derive conclusions about how software operates in the field. Section 6 describes the data mining techniques that are suggested for mining software usage knowledge. In Section 7 we present the prototype that was constructed as an instantiation of the usage mining method. We evaluate the two artifacts in a case study in Section 8. Finally, in Section 9 we discuss the insights from this research and provide some general conclusions.

2. Related work

As far as specific research on software usage analysis is concerned, extraction of in-the-field usage knowledge remains an area that needs a lot of enrichment. Data analysis techniques have been previously applied to this field: for software reengineering purposes (El-Ramly, Stroulia, & Samir, 2009; Lefngwell & Widrig, 2003), for program comprehension (Zaidman, Calders, Demeyer, & Paredaens, 2005), for re-documentation of use cases (Smit, Stroulia, & Wong, 2008), or for user interface learning agents (Ruvini & Dony, 2001). However, these approaches are not directly related to analyzing how the end-users are utilizing the software in the field. Also, they do not provide any holistic approach to the various usage knowledge types (e.g. user profiles or most frequent navigation paths). Some of them are very old, so they do not use state of the art data mining techniques.

Several techniques have been developed for deriving models based on analysis of log files (Petruch, Tamm, & Stantchev, 2012). For example, analyzing the audit trails through sequence analysis techniques can prove to be quite useful for evaluators who are curious to compare the designers' expectations of use with the actual usage patterns followed by the users (Judd & Kennedy, 2004). This approach is similar to the field of Process Mining (Van der Aalst & Weijters, 2004), which involves analysis of event logs with the goal to monitor and/or redesign operational business processes that take place in an information system (Maruster & van Beest, 2009). Process Mining has also been applied on web services workflows mining (Dustdar & Gombotz, 2007).

Practical examples that include usage analysis of logged events can be found in literature (Lin & Tsai, 2011). Transaction logs analysis techniques are used in the usage analysis of a digital library (Jones, Cunningham, & McNab, 1998). Shen, Fitzhenry, and Dietterich (2009) use a subgraph mining algorithm, in order to partially automate the user's workflows or to create to-do lists, in a desktop assistant application. Sartipi and Safyallah (2009) developed a data mining algorithm for sequential pattern discovery on traces that are generated from the execution of task scenarios.

A closely related area to software usage analysis is web usage mining, one of the subfields in web mining. Web usage mining is the process of automatically discovering and analyzing behavioral patterns and users profiles in clickstream and other associated data, which are generated or collected when users interact with web resources found on one or more websites (Liu, 2006). The most common pattern discovery and analysis tasks include: session and visitor analysis (Liu, 2006), visitor segmentation and profiling (Xie & Phoha, 2001), association analysis (Meo, Lanzi, Matera, & Esposito, 2006), navigation analysis or path analysis (Cooley, Mobasher, & Srivastava, 1999), and prediction based on web user transactions (Liu, 2006).

3. Research design

The users' shift to cloud computing applications (Park & Ryoo, 2013) creates the opportunity for software vendors to automatically collect vast amounts of usage data. Although several algorithms have been developed to analyze the behavior of website visitors, they are hardly ever used in the software products domain. This research aims to follow a meta-algorithmic approach, by incorporating the state-of-the-art data mining techniques in a method. Our goal is to show how the appropriate technique can be used for analyzing each aspect of software users' behavior.

In Fig. 1 we display a diagram of our research design. We follow the *design science research* (*DSR*) approach (Hevner, March, Jinsoo, & Ram, 2004), as we develop a method and a prototype for software usage mining. We follow the General Design Cycle (Vaishnavi & Kuechler, 2007), which includes the phases: problem awareness, suggestion of a tentative design, development of the artifact, evaluation, and conclusion.

We construct a method for usage mining using the Method Engineering approach provided by van de Weerd and Brinkkemper (2008). To evaluate the effectiveness and applicability of the method, we perform a case study in an international software company, for analyzing the usage of an online financial application by trial customers.

In order to structure our data mining research, but also to assemble our Usage Mining Method, we follow the CRISP-DM Reference Model (Chapman et al., 2000), which includes six phases of data mining activities: business understanding, data understanding, modeling, evaluation and deployment. In Section 4 we show how these activities were incorporated in the method.

For the evaluation of this research we use Case Study Research (Runeson & Höst, 2009) and follow a positivist approach. The case study takes place in the context of Exact Online, an online financial management application, and consists of four phases: (1) design and preparation, (2) conducting, (3) analyzing and (4) reporting. In Section 8 we describe these phases in detail.

4. Usage Mining Method

In this section we present the first design artifact that we constructed in this research. The Usage Mining Method suggests an ordered set of activities that should be followed to extract relevant usage knowledge from software operation data.

In order to provide guidance in analyzing software product users' usage behavior, we propose the Usage Mining Method (Fig. 2). The method has been constructed with the Method Engineering approach, provided by van de Weerd and Brinkkemper (2008). The method is designed for mining usage, user and corporate data of software-as-a-service applications, which are collected in a central point on the software vendor's side.

Fig. 2 includes an overview of the method's activities and sub-activities. The activities that are connected through an arrow are sequential, i.e. they need to be carried out in a pre-defined order, for the reason that the outcomes of the former activities are