Business Process Mining from E-Commerce Web Logs

Nicolas Poggi[1,2], Vinod Muthusamy[3], David Carrera[1,2], and Rania Khalaf[3]

[1] Technical University of Catalonia (UPC) Barcelona, Spain
[2] Barcelona Supercomputing Center (BSC) Barcelona, Spain
[3] IBM T. J. Watson Research Center Yorktown, New York, USA

Abstract. The dynamic nature of the Web and its increasing importance as an economic platform create the need of new methods and tools for business efficiency. Current Web analytic tools do not provide the necessary abstracted view of the underlying customer processes and critical paths of site visitor behavior. Such information can offer insights for businesses to react effectively and efficiently. We propose applying Business Process Management (BPM) methodologies to e-commerce Website logs, and present the challenges, results and potential benefits of such an approach.

We use the Business Process Insight (BPI) platform, a collaborative process intelligence toolset that implements the discovery of loosely-coupled processes, and includes novel process mining techniques suitable for the Web. Experiments are performed on custom click-stream logs from a large online travel and booking agency. We first compare Web clicks and BPM events, and then present a methodology to classify and transform URLs into events. We evaluate traditional and custom process mining algorithms to extract business models from real-life Web data. The resulting models present an abstracted view of the relation between pages, exit points, and critical paths taken by customers. Such models show important improvements and aid high-level decision making and optimization of e-commerce sites compared to current state-of-art Web analytics.

1 Introduction

To remain competitive, online retailers need to adapt in an agile, non-structured way, resulting in large, unstructured websites and rapidly changing server resource demands [14]. Moreover, Conversion Rates (CR), the fraction of users that reach a certain goal, such as buying a product on the site, are decreasing: less than 2% of visits result in a purchase on most sites [14]. A low CR is influenced by factors including affiliation programs, changes in user habits such as comparing different sites at the same time [15], and meta-crawling. For example, *Kayak.com* and similar meta-crawlers present the user the best results gathered from several sites, thereby lowering the visits to each site and the CR.

Most online businesses rely on free Web analytic tools to inform their Web marketing campaigns and strategic business decisions. However these tools currently do not provide the necessary abstracted view of the customer's actual

F. Daniel, J. Wang, and B. Weber (Eds.): BPM 2013, LNCS 8094, pp. 65–80, 2013.

behavior on the site. Without the proper tools and abstractions, site owners have a simplified and incorrect understanding of their users' real interaction patterns on the site, and how they evolve.

In this paper we apply Business Process Management (BPM) methodologies to e-commerce Website logs. Structured formal models of user behavior can provide insights on potential improvements to the site. In particular, providing a high-level abstracted view of the workflows leading to purchases and most common exit pages in order to make decisions on site optimization. BPM concerns the management of business processes including the modeling, design, execution, monitoring, and optimization of processes [8]. While loosely-structured to completely ad-hoc processes have not traditionally not been considered by BPM, we (and others [7]) see this is part of a spectrum [19].

Unlike Web analytics [9], process analytics is concerned with correlating events [20], mining for process models [24,26,18], and predicting behavior [25]. We propose treating a user's web clicks as an unstructured process, and use process mining algorithms to discover user behavior. The mined process model captures the causality and paths of user interactions that lead to certain outcomes of interest, such as buying a product. Such insights can be difficult to extract from traditional Web analytic tools.

We use the Business Process Insight (BPI) platform, a collaborative process intelligence toolset [19]. BPI includes the knowledge-based process miner, which differs from traditional process mining in its initial search structure and the set of activities considered for edge operations.

We use a real data set from Atrapalo, an online travel and booking agency (OTA) that includes popular services such as flight and hotel reservation systems. The data set includes the HTTP requests made by customers to the site over a three month period, captured using Real User Monitoring techniques. We apply process analytics to this dataset, and make three main contributions:

1. We outline how to transform web clicks into tasks suitable for analysis and modeling with BPM tools. In particular, we classify the URLs that correspond to web clicks into high level tasks. We compare both a manual classification approach with knowledge from a domain expert, and an automatic classification algorithm. The tasks are then grouped into web sessions representing a particular customer's interaction with the site.
2. We describe how to mine business processes that includes how regular web visitors and customers behave. A challenge here is that, by design, most process mining algorithms capture only the most common behavior in order to keep the resulting mined process model simple enough for a human to understand. However, in web commerce data, the behaviors of interest, such as a customer buying a product, are infrequent. We address this issue with techniques such as saturating the dataset with low frequency behavior we wish to observe, clustering the process instances to extract patterns of behavior, and using a knowledge-based processing mining algorithm.
3. We evaluate the use of the knowledge-based mining algorithm under a variety of conditions, and explain its suitability to extract process models that

abstract a complete over-view of user navigation from real, noisy data. Our evaluation is notable for using real web logs, and unique in applying BPM techniques to an e-commerce site.

2 Background and Related Work

Business Process Management. Business processes can be strongly structured (as in BPEL), loosely-structured (as in Case Management tools), or entirely unstructured. The latter are common with ad-hoc human tasks. For example, a party planning process may be carried out by phone, e-mail, and faxes by people not following any predefined process. Such unstructured processes are an important part of the spectrum of processes in the wild [7,19]. Process mining automates the discovery of process models from event logs, and we propose treating e-commerce web interactions as business processes.

Business Process Insight. The Business Process Insight (BPI) system [19] detects relationships among events, and outputs a set of correlation rules. The correlation engine applies these rules to create *process traces* that group related events that belong to the same process instance. The process traces can then be used to discover different process models or to train predictive models for making live predictions on future behavior. Similar to Process Spaceship [11], BPI is a process intelligence solution that simplifies the understanding of business process executions across heterogeneous systems, as well as provide a foundation for process-aware analytics for both historical and live events. The BPI architecture supports plugging in different process mining algorithms, such as the alpha and heuristic mining algorithms in the ProM process mining tool [26].

Process Mining. Process mining aims to extract a business process model from a set of execution logs [1,26,23,4,13,17,25]. Process mining algorithms typically find all activities (nodes) in the process model, constructing a dependency graph with no edges, and then search through the space of process models by adding, deleting, and reversing edges. However, many algorithms restrict the activities that can be considered for these edge operations by observing activity adjacency in the execution log. For instance, only if activities A and B are adjacent in the log will they be considered for an edge operation. The *knowledge-based miner* developed at IBM Research can leverage domain knowledge by initializing its search with a predefined process model. The algorithm also considers a larger search space of process model structures by considering edge operations on both log-adjacent and log non-adjacent activities. This larger search space enables the discovery of process models that more accurately represent the process execution log. The knowledge-based miner constructs an activity precedence graph that encodes statistically significant activity dependencies among the execution logs, as well as dependency and independency graphs [16] augmented with the confidence levels of edges specified by a domain expert. The algorithm first extracts activity dependencies and independencies from the process logs and expert knowledge, partially using some of the techniques developed by Agrawal *et al.* [2]. It then discovers the split/join semantics based on some of the ideas in [16].

ProM is a prominent process mining tool that serves as a front-end for various process mining techniques [26]. How data attributes influence the choices made in a process based on past process executions by leveraging decision trees has been investigated in [18]. The focus of [18] is to correctly identify decision points in the presence of duplicate and invisible tasks in a log. There are also a number of probabilistic models proposed for modeling business processes [13,17,25]. In [5] Ferreira et al. proposes a probabilistic approach implementing Expectation-Maximization for discovering process models from unlabeled event logs. In Section 4 we propose a similar approach to cluster clicks into events, both by manual and automatic methods.

Web Analytics. Web analytics deals with the collection, measurement, and analysis of user navigational data. One way to classify the analytics techniques is by the method of data collection: page tagging through Javascript, web server log analysis, beaconing by inserting a remote object on the page, packet sniffing, and hybrid approaches [27]. The main metrics analyzed include the number of unique and returning visits, URL access frequency, geolocation, client web browser and version, and statistics around these metrics. Newer tools from Google and Yahoo also support tracking of marketing campaigns and conversion goals, such as users subscribing to the site's newsletter or purchasing a product. The latter platforms are Javascript-based implementations of page tagging. Page tagging can be manually tuned to group different tasks on the website logically; by default tools follow the traditional URL analysis. The objective of web analytics it to provide feedback for website owners on user behavior in order to improve site navigation and conversion goals [9]. However, improvements are only possible when there is a clear understanding on the underlying site structure and user needs.

Web Mining. There are few published studies on real e-commerce data, mainly because web logs are considered sensitive data. In [21] web mining is classified into usage, content, and structure web mining. The main purpose for structure mining is to extract previously unknown relationships between Web pages. While this paper falls within the scope of structure web mining, most of the literature in this topic focus on recommendation systems and web personalization [3]. In [22] authors presented a comparative study of the navigation behavior of customers and non-customers to assess and improve the quality of a commercial web site; while in this work we aim to build a process model that shows the complete interactions of most users in the site that includes customer sessions.

Customer Behavior Model Graphs (CBMG) can be used to provide an abstracted view on web navigation [10]. The CBMG is built using the k-means clustering algorithm that creates a probability matrix for the possible path transitions from a state. In this paper, we do not focus on predicting the user's next click, but seek to extract the most relevant critical paths occurring in the site and build the process model. In particular, we are interested in the important events and workflows that lead to a user buying a product.

Web analytics has evolved from those that analyzed web server access logs to generate reports and evaluations based on URL, IP address, and browser agent grouping and frequency analysis, to newer tools such as Google's or Yahoo's analytics. These tools, however, do not extract the user behavior at an abstraction level that is appropriate to understand the actual critical paths taken by consumers. The experimental results in this paper lead us to believe that web analytics can benefit from BPM modeling. We are not aware of any other literature on applying BPM techniques to an e-commerce site.

3 Application Scenario and Dataset

Online Travel E-commerce Market. Online travel agencies (OTAs) are a prominent sector in the online services market. A Nielsen report on global online shopping found airline ticket reservations represented 24% of online shopping purchases, hotel reservations 16%, and event tickets 15%, for a combined 55% of global online sales [12]. Conversion Rates (CR) are usually not made public as they reveal the success of a business strategy, but we have confirmed that for the OTA industry CR of less than 2% is a common figure when taking into account all web requests [14]. Our study considers Atrapalo, an international online travel agency and booking site representative of the OTA industry. It features popular e-commerce applications found in the Web and over twelve years of online presence. We have been given access to a three month dataset from 2012 featuring several million HTTP requests of site visits.

Atrapalo's Application. Atrapalo's online application follows a typical travel site structure, offering the following products: flights, hotels, cars, restaurants, activities, cruises, vacation packages, and ticket bookings. Some product inventories are maintained internally, such as restaurant bookings, some are completely external, such as flights, and some products such as hotels contain a mix of internal and external providers. The company's main presence and clientele include Spain and Italy from Europe; Peru, Colombia, Brazil and Chile in South America; and a few visitors from elsewhere. Each country is served by a separate top level domain and has differentiated products enabled. It is important to remark that the site has over 12 years of online presence, and its structure has been in constant update and optimization including a recently added mobile version, but it retains an important legacy code base.

Dataset Used for the Experiments. The dataset provided by Atrapalo contains click-stream information from visitors and customers of the different products offered in their domains presented in the previous section. The dataset contains more than four million user clicks representing about 850 000 full user sessions. The average navigation time per user is four minutes and eight seconds, and there are 4.36 clicks per session. The dataset was collected by sampling over a period of three months from June to September 2012.

The novelty of the presented dataset is that it was produced using Real User Monitoring (RUM) techniques, in contrast to typical server logs collected by the web server. Every time a page is loaded in Atrapalo—for a sample of web sessions—an asynchronous AJAX request is sent from the user browser to the server. This information is used by Atrapalo to monitor and optimize the performance of web pages from the user's perspective.

RUM log files are useful in this study. First, the dataset is cleaner, as it only contains data from web browsers that can process Javascript, thereby avoiding most crawler and bot traffic. Crawler behavior is particularly different from user traffic and can account for over 20% of total requests [15] and distort results. Second, it only contains information about pages that the user actually clicks. In our previous work [15] we have performed workload characterization of web server generated datasets and among other findings found that less than 50% of requests corresponded to user clicks. The rest of the request traffic was composed of automatic AJAX requests for autocomplete controls or the RUM request, dynamically generated Javascript and CSSs, HTTP redirections, and backend requests for the user session. Third, cached web pages, either in the customer browser or any intermediate proxy are present in the RUM log. With this, the complete user navigation can be reconstructed. Having complete data sets is important for any mining or prediction algorithms, as most are susceptible to noise to different degrees. The next section presents our approach to convert web sessions into process models.

4 Web Sessions as Process Models

Among the characteristics and challenges in process mining [24] is having to deal with noisy event data. Noise is common in web logs as web proxies and caches can alter content. Moreover, web browsers behave differently, and browser plugins can affect navigation patterns. Furthermore some requests can get lost, due to dropped connections, users roaming over a mobile network, and users altering the normal flow with refresh, back, and forward browser buttons. Also, a user's web session can expire. We have observed in our preliminary work that weblogs are indeed noisier than typical event logs for BPM systems.

Another set of important characteristics is the presence of *loops*, *duplicate activities* and *parallel* tasks. Web sessions also exhibit these properties to different degrees. For example, when a user is searching for hotels, he might try different dates, looping over the search page, or he might click on a hotel deal, see the details, go back to the search page, click on another deal and so forth. The user might have also opened different hotel deals in different tabs of his browser, creating *parallel* tasks. He might have also been searching for flights to the same destination, or to rent a car from the airport in parallel. Parallel tasks, duplicate activities and loops are present in most web navigations of more than a couple of clicks. Current research detecting loops and having loop aware algorithms can be substantially beneficial for mining web navigation and performing predictions on the user's navigation.

While Web sessions are also time constrained as typical BPM activities, *time* is also major difference. As mentioned in the previous section, the average web navigation is only of about four minutes, while BPM processes, such as supply chain management, can span days. As BPM processes can require human intervention, in the web the process is completely automatic. This difference has several implications as there is no time for manual interpretation and modification of an executing process. BPM tools, if applied to web navigation need to be automatic, free of human intervention and deployed in real-time.

While on this study we only target web content, we advocate that user navigation be included in process models of companies that involve both web interaction and traditional processes. The next section looks at how to abstract web clicks into logical tasks to be consumed by a BPM system.

4.1 Classifying URLs into Logical Tasks

The first challenge analyzing web logs is to classify the URLs of the site. For the dataset used in the experimentation several URL rewriting techniques were implemented for security, dynamic page generation, search engine optimization, and localization. There were 949 532 unique URL in the dataset, if we take the query string out of the URL, the number of distinct pages reduces to 375 245.

In order to extract the *action* —type of process and output of a page— from a URL in Atrapalo's dataset, we had to implement the rewrite engine used for the page classification. Rewrite engines usually perform regular expression matching to URLs. In Atrapalo's URLs, the first element in the URL path indicates the name of the product, such as flights, hotels, cars, or events. Each product had custom implementations of the rewrite engine and how regular expressions were performed. About 20% of the URLs didn't match any regular expression, and for these URLs query string classification was performed by looking for a custom parameter "pg", which specified the page *action*. Using the query string approach we were left with 5% of unclassified URLs that were manually analyzed and classified using string search and replace.

Table 1. Classification of URLs into logical tasks

Tag	Description
Home	Main home page
ProductHome	Home page for each product
Landing	Search engine landing pages
Promo	Special promotional pages
Search	General site search
Results	Product search and results
Details	Product detailed information
Opinions	Opinions about a product
Info	Site help or general information
CartDetails	Shopping cart details
CartPurchase	Shopping cart purchase forms
Confirmation	Confirmation page of a sale
UserAdmin	User self reservation management

After the URLs where translated we were left with 533 different page actions or type of pages. However some of the page names occurred only once, a problem we attribute to noise and errors in the rewrite engine implementation. We then filtered the pages that did not have more than one occurrence, and ended with

233 page names. This means that across the products of the site there were 233 different types of pages. Some of the pages serve the same logical function, such as the search page for hotels, flights or cars, or the different home pages for each product. After a manual analysis on the site structure and URLs, we decided to classify them in 14 logical types of pages detailed in Table 1.

Although the classification in Table 1 is particular to Atrapalo's dataset, many e-commerce sites share similar structures especially for sites implementing travel and booking products. It is important to remark that through the classification of pages no data is lost. Page classification is added as extra columns to the dataset. The URL and page types are kept in the dataset, so we can later use them to filter or to extract better path predictions. The next section presents a proposal for automating page classification.

4.2 Automating Page Classification

Classification of types of pages into logical groups is necessary to map user clicks occurring in a website to abstracted logical tasks to be consumed both by BPM algorithms and final reports to humans. We noticed while reviewing the results that many page actions had similar names. There was at least a search page per product and different types of search pages, including flightsSearch, hotelsSearch, flightsCalendarSearch, hotelsSearchCity. To aid classification, we have tested the clustering of the page names using the WEKA open source machine learning framework [6]. WEKA contains several popular ready to use algorithms for classification and clustering among other tools. As we had previously decided that the classification has 14 logical types of pages, K-means clustering was our first natural choice to test, as it performs in general scenarios with known number of clusters. We have used WEKA's SimpleKMeans implementation and setting the number of clusters to 14 and the "classes to clusters" evaluation option. SimpleKMeans yielded an error of 39.90% in classifying the 233 names into 14 clusters. We have also experimented with the EM (Expectation-Maximisation) algorithm both with automated and manual numbers of clusters yielding 76.93% and 41.88% of classification errors, respectively. Table 2 summarizes the clustering results. If the number of classifications is known, K-means clustering can reduce the manual work needed to simplify page classification. The next section details our experiments with process mining.

Table 2. Classifier Evaluation

Algorithm	Clusters	Error
SimpleKmeans	14	39.90%
EM	14	41.88%
EM	Automatic	76.93%

5 Process Mining for Customers

This section details our experiments mining the business processes of customers in Atrapalo's dataset with the page classification from the previous section. Three new events were added to each web session: Start, End, and BuyerEnd.

These events are helpers to the mining algorithms and to their visualizations to show where sessions start—as there are different starting points—and exit. Exit events were marked BuyerEnd if the session ended in a purchase, to differentiate them from regular sessions. This distinction is not only used for visualization purposes, but for path prediction algorithms as well for our ongoing research.

As mentioned in Section 3, only a small fraction of visits to the site ended buying a product. The conversion rate for the site is less than 2% of the total number of visits. Having such a small percentage is a problem for most mining algorithms, as these low-frequency traces (web sessions) will be filtered out by most implementations producing an incomplete model. In our study we present three different approaches to this problem creating three new different datasets: saturating the data set (*saturated*), clustering (*clustered*), and *biasing* toward a previously set model with the knowledge-based miner. We call the original dataset the *normal* dataset.

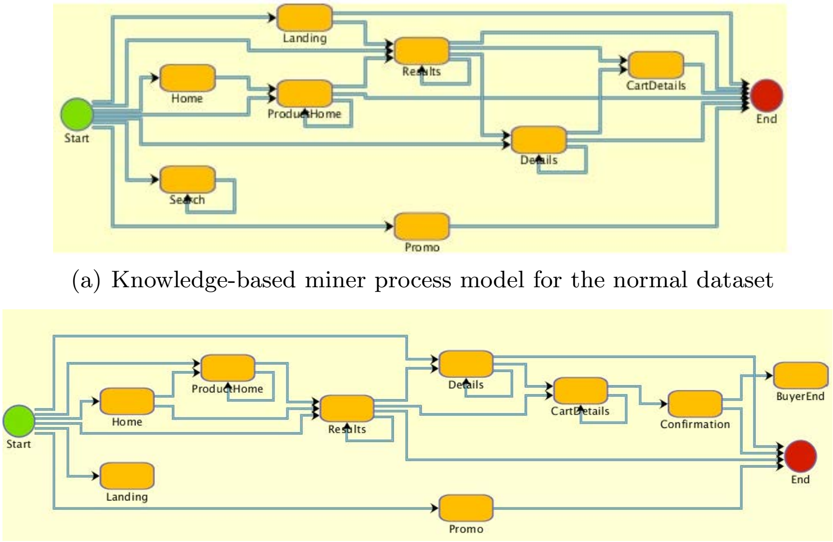
5.1 Saturating the Dataset with Customers

The first strategy to mine customer models was saturating the dataset. This entailed producing a new dataset where the percentage of buying customers is higher by removing sessions that did not purchase. We have chosen the ratio 1/3 of customers to just visitors. This ratio is choosen as customer sessions are longer in average, leaving us with and even dataset of about half of the entries belonging to customer sessions. With this ratio, we have created a new dataset including the entire customer sessions present in the normal dataset, and 2/3 more sessions from regular visits from the top of the dataset. This dataset having about 8% of the total entries of the normal dataset, but including all the purchasing sessions.

This approach was tested with the process mining algorithms implemented in BPI, and allowed us to test the different algorithm implementations. As mentioned previously, the alpha miner is not suited for event logs with noise or incompleteness, as is typical in real logs [4]. Results for alpha miner are omitted for the saturated dataset as it produced incomplete results.

Knowledge-Based Miner. Figure 1 shows the resulting models by applying our knowledge based miner with default noise and window parameters to the *normal* (Figure 1(a)) and *saturated* (Figure 1(b)) datasets. The general workflow of events can be seen from the figures, with the main distinction being that the *normal* dataset does not contain the *Confirmation* and *BuyerEnd* events and edges. The *CartDetails* event is present in both. This means that while there are many users that add a product to the shopping cart and see its details, few ultimately purchase the product. In these cases the buying events are being discarded as noise, while on the *saturated* dataset they are being kept. Loops can also be seen in both models, but the loops are from the same originating event to itself, such as users iterating over the *Results* event.

Another insight from the knowledge-based miner models is that the *Promo* event is not linked to any other event; almost all users that get to the site through a promotional page leave the site without any further navigation. On the *normal*



(a) Knowledge-based miner process model for the normal dataset

(b) Knowledge-based miner process model for the buyers saturated dataset

Fig. 1. Knowledge-based miner process models for the *normal* and *saturated* datasets

dataset, some users from the *Landing* event get to the results. In the *saturated* dataset, however, the landing page event doesn't have any outbound links. The same can be observed with the *Search* event in the *normal* dataset: it's only link is a self-loop. The *Search* event is not present in the *saturated* model, because it is a low frequency event and not used by most customers. We have verified that most results pages were directly reached from each product home pages. *Search* events represent the general site search feature that searches all products at the same time, and results show they are not very effective and were reported back for optimization. Further details about the knowledge-based miner are given later in this Section.

Heuristic Miner. Figure 2 shows the model generated by the heuristic miner. The heuristic miner model included all of the events from the *saturated* dataset, presenting the same behavior for the *Search*, *Promo*, and *Landing* events as the knowledge-based miner. One addition is the *UserAdmin* event, discarded by the knowledge-based miner as noise. There is however one main difference with the knowledge-based miner: most events are shown as independent from another, except for a few with one edge and the combination *Details*-*Info*-*ProductHome*. This is the main difference with the knowledge-based miner, and from our tests it makes it less applicable to web logs and similar datasets where an end to end path is required.

Another disadvantage is that it overfits the model. If we had more events, as we did prior to applying the classification in Section 4, the algorithm would not highlight the critical paths in the web navigation. While the heuristic miner is very well regarded [4], as mentioned in Section 2, the same study also questions

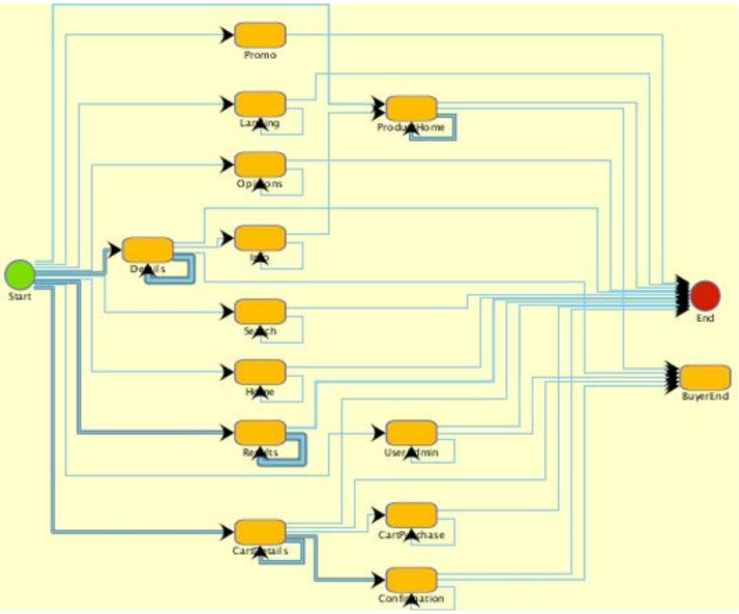


Fig. 2. Heuristic miner with saturated dataset

traditional process mining algorithms and advocates for new methods for real data. Results with the *normal* dataset were almost identical, except for the thickness (significance) of the edges between activities, as the frequency was different between both datasets.

Fuzzy Miner. The Fuzzy Miner in PRoM [26] can visually cluster events, and can be useful when working with a large number of activities and unstructured behavior. The fuzzy miner gave good results mining the *saturated* dataset. The main difference with the knowledge-based miner is that it doesn't remove noise from the dataset, but it can simplify and group the model to the desired level of abstraction. However, the generated clusters do not necessarily group events logically; the clusters included unrelated event types as compared to our manual classification. It, therefore, does not seem that fuzzy mining can be used to aid or avoid the URL classification performed in Section 4.

5.2 Clustering Sessions

The next tested approach to mine for customer sessions was clustering. BPI implements the string distance algorithm to cluster imported traces. By clustering similar sessions, we can run the process mining directly on individual clusters through the BPI interface. This feature is very helpful as clustering can help remove noise and allows the ability to mine specific customer clusters or target groups without the need to saturate the dataset. For example, with clustering, the alpha miner could be applied to small clusters if required.

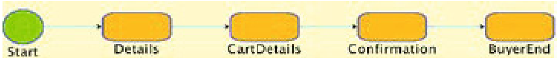


Fig. 3. Process model of a customer cluster for Heuristic and knowledge-based miners

Figure 3 shows the model produced by both Heuristic and knowledge-based miner to a specific small cluster of customers, representative of the most common buying process. It shows the critical path (the most important pages) for buyers on the website, and thus, the most important pages to keep optimized. It also shows that the most typical buying process consists of three main pages: *Details*, specific product information; *CartDetails*, final costs details and payment options; and *Confirmation*, the reservation confirmation page. This would mean that most buying sessions go strait to purchasing without much searching, probably performed at a previous time and different session.

The disadvantage of clustering, besides not having the complete process in the output model, is that models cannot be combined directly without manual work. The knowledge-based miner allows us to use prior knowledge, such as the model produced by clustering as shown in Figure 3, to assign more weight for these events and edges. This particular feature is detailed in the next subsection as a different strategy.

5.3 Prior Knowledge

The knowledge-based miner, besides being able to keep longer paths and be parameterized by the amount of noise (fitting) and window size, can use another model as prior knowledge with a tunable confidence. This feature can be used not only to mine for customer models without saturating the dataset, but also to include certain clusters or behavior, such as the effect of improving the promotional page, or a marketing campaign targeting a certain product.

Figure 4 shows both the model produced by the knowledge-based miner miner on the *normal* dataset, and the output when the model from Figure 3 is applied to the knowledge-based miner. Results are the same in both, except that when the prior knowledge is applied, the output includes the *CartPurchase*, *Confirmation*, and *BuyerEnd* events.

Figure 4 also shows the use of the knowledge miner parameters. Compared to Figure 1 it shows the *UserAdmin* event and more edges and loops between events. The reason is that both figures were executed with lower *window* and *noise* parameters. This shows how models can be abstracted and fitted using these parameters in the knowledge-based miner algorithm.

6 Discussion of Results

The previous section presented three strategies to mine process models to include customer navigation behavior besides general user behavior, as well as a comparison of the mining algorithms. As we are dealing with real web user navigation of a large site, there is no correct process model we can compare our results against.

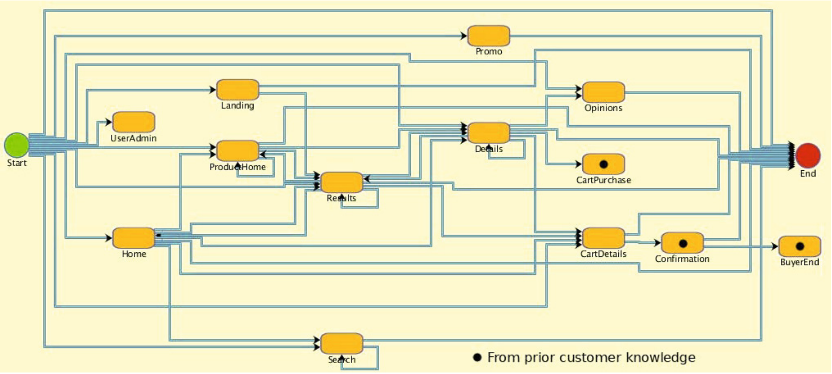


Fig. 4. Knowledge-based miner process models on the normal dataset

The main reasons for this are that there are about one million different pages, most pages can be reached from search engines or link sharing directly and the combinations of pages are too large to generate or validate manually. Rather, we rely on domain knowledge of the site and previous works characterizing the site and generating CBMGs [10].

The first strategy to generate the process models consisted in *saturating* the supplied dataset—in our case with customers—to be able to work with most mining implementations. The alpha miner gave incomplete results in the *saturated* dataset. The heuristic miner process model did not show the relation between the different pages, but displayed them as independent from one another (see Figure 2) except for one combination of pages. The second strategy consisted in *clustering* sessions before applying process mining. *Clustering* enabled mining directly on a specific group of interest, such as customer clusters, but required human intervention to select the target cluster and only represent sessions contained in the selected cluster. A benefit of *clustering*, is that it allowed the alpha miner to produce results on small clusters of similar sessions, and can be used for tools that need a model produced by the alpha miner. The heuristic miner also gave coherent results for clusters. The last strategy, particular to our implementation, was the use of *prior knowledge*, a specific feature of our knowledge-based miner implementation that can use a previously generated model, such as from a customer's clustering as taken from Figure 3. The knowledge-based miner was able to produce process models our domain experts were expecting, as it was able to filter non-critical events, bias towards a particular group, and keep longer paths between edges. Furthermore, the size of the output model can be altered through its parameters to control the noise and trace window size.

Lessons Learned. While applying techniques described in previous sections enabled us to produce meaningful process models, process mining on real logs, including the datasets presented in this study, demonstrates the need for new algorithms that can work with noisy event logs, a large number of events, and underrepresented groups, such as buyer sessions. As with [4], we also quickly found that the alpha miner was not able to produce the process models we and domain experts were expecting. We attribute noise and session incompleteness as the main reasons for the incomplete or incoherent outputs from the alpha and heuristic miners. Most web sessions in our datasets are only composed of one click, while a smaller fraction of visitors, including buyers, have longer sessions which mislead miner classification.

From the different web logs we had available: web server, application, analytics, and RUM; we choose the one produced by RUM techniques (see Section 3). The RUM dataset was free of crawlers that cannot process Javascript, it doesn't include automated background actions such as AJAX controls, and includes pages cached by intermediary proxies or the user browser. While cleaner and more complete than the other available logs, they were still not clean enough for some mining algorithms. Having a large number of activities (pages), was a problem for most of the mining algorithms which failed to produce coherent results and required excessive computational resources. We tested the fuzzy miner in ProM with success as a suitable option to process a large numbers of events and unstructured activities. The fuzzy miner can also group events into clusters for visualization though ProM's interface. However a pre-classification of events into categories is needed not only for miners, but for models to have a clear abstraction level for human consumption. As from our previous experience, we also found that only a small sample of sessions—a few thousand—are needed to produce meaningful process models. However, while this is true for a general process model, for process models that need to reflect seasonal trends, larger datasets are needed. Processing larger datasets requires the mining implementations to be efficient in computational resources as well as parallelizable.

User Feedback. The Atrapalo.com Web Analytics team provided some feedback about the results in this paper. They noted that for a data analyst to use the discovered models, BPM tools need to be integrated into their familiar day-to-day products such as their current Web analytics tools stack. Since the time our results were presented, the general site search feature, which our process models showed as not contributing to the sale process, has been redesigned. With the added features we have been reported that the conversion rate of visitors using the search feature has improved up to 46% and the bounce rate (users that leave the site after visiting this particular page) lowered by 22% for a particular product. Page classification as performed in Section 4.1 was also mentioned to be very useful, as the site contains over a million different URLs and the number of different pages keeps growing making it difficult to get a clear understanding of the site. In general we have found that while larger models are needed for automated processing such as path prediction, simpler, more abstracted models are appropriate for informing business decisions.

7 Conclusions

This paper applied process mining techniques, and in particular the Business Process Insight platform, to analyze web user behavior. We found that web navigation shares characteristics with traditional BPM activities such as loops and parallel tasks. However, sessions only span a few minutes on average and include no human intervention. We also discovered that any analysis of web logs required the classification of URLs to higher level logical tasks. Otherwise, the number of unique URLs—almost a million in our case study—is impractical for human consumption and traditional mining algorithms. Manual URL rewriting rules reduced the number of unique URLs substantially in our case study. We also showed that clustering algorithms can automatically classify URLs, requiring only that each cluster be named. The above classification of URLs allowed web logs to be mined for processes that represent the navigation behavior of users. We found that a knowledge-based process mining algorithm performed the best, generating process models that most resemble the behavior we were expecting in our dataset. We hypothesize that this mining algorithm may perform well in other real web applications, but this will require further study to validate. There are several insights from the obtained process models, such as the low conversion of the *Promo* page, and the ineffectiveness of the general site search feature. Since our first results and feedback, the company redesigned the general site search, improving the conversion rate of visitors using the search feature by up to 46%, and lowering the bounce rate by 22% for a particular product.

Process mining algorithms are designed to extract the dominant behavior observed and filter out noise to keep the resulting mined process manageable. However, in our case study the interesting behavior—those that result in a user buying a product—seldom occur. We expect this to be the case in many web applications. To avoid losing this behavior, we took the approach of saturating the dataset with more traces that result in the outcome of interest. This simple strategy worked well in producing a complete process model that includes both the most common behavior on the site, and also includes the behavior of users that buy a product. An alternate strategy is to provide an expected process model—for example from clustering—as input to the mining algorithm. However, this option is only available with the knowledge-based miner, and requires some domain knowledge. Web sites can be complex to model, but the insights derived from mining the actual behaviors were extremely valuable in our case study for site optimization. We feel that BPM tools and techniques can complement and improve current Web Analytic tools by giving them abstracted views of the most important paths taken by types of visitors. This understanding of their navigation behavior can be used to inform business and IT decisions and improve sales as from the results of this study.

Acknowledgements. We thank Atrapalo.com for the datasets, feedback, and domain knowledge for this study. We also acknowledge Aubrey Rembert who developed and offered support on the knowledge-based miner. This work is partially supported by the Ministry of Science and Technology of Spain under contract TIN2012-34557.

References

1. Aalst, W., et al.: Process mining manifesto. In: Business Process Management Workshops, vol. 99, Springer, Heidelberg (2012)
2. Agrawal, R., Gunopulos, D., Leymann, F.: Mining process models from workflow logs. In: Schek, H.-J., Saltor, F., Ramos, I., Alonso, G. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 469–483. Springer, Heidelberg (1998)

3. Bhushan, R., Nath, R.: Automatic recommendation of web pages for online users using web usage mining. In: ICCS (2012)
4. De Weerdt, J., et al.: A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. Inf. Syst. 37(7) (2012)
5. Ferreira, D.R., Gillblad, D.: Discovering process models from unlabelled event logs. In: Dayal, U., Eder, J., Koehler, J., Reijers, H.A. (eds.) BPM 2009. LNCS, vol. 5701, pp. 143–158. Springer, Heidelberg (2009)
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explorations 11(1) (2009)
7. Kemsley, S.: It's not about BPM vs. ACM, it's about a spectrum of process functionality, `http://www.column2.com/2011/03/its-not-about-bpm-vs-acm-its-about-a-spectrum-of-process-functionality/`
8. Koehler, J.: Business process modeling
9. Kumar, L., Singh, H., Kaur, R.: Web analytics and metrics: a survey. In: ACM ICACCI (2012)
10. Menascé, D.A., Almeida, V.A., Fonseca, R., Mendes, M.A.: A methodology for workload characterization of e-commerce sites. In: ACM EC (1999)
11. Nezhad, H.R.M., Saint-Paul, R., Casati, F., Benatallah, B.: Event correlation for process discovery from web service interaction logs. VLDB J. 20(3) (2011)
12. Nielsen. Trends in online shopping, a Nielsen Consumer report. Technical report, Nielsen (February 2008)
13. Pfeffer, A.: Functional specification of probabilistic process models. In: AAAI (2005)
14. Poggi, N., Carrera, D., Gavald, R., Ayguad, E., Torres, J.: A methodology for the evaluation of high response time on e-commerce users and sales. In: ISF (2012)
15. Poggi, N., et al.: Characterization of workload and resource consumption for an online travel and booking site. In: IEEE IISWC (2010)
16. Rembert, A.J., Ellis, C.S.: Learning the control-flow of a business process using icn-based process models. In: ACM ICSOC, pp. 346–351 (2009)
17. Rozinat, A., Mans, R.S., Song, M., van der Aalst, W.M.P.: Discovering colored petri nets from event logs. STTT 10(1) (2008)
18. Rozinat, A., van der Aalst, W.M.P.: Decision mining in ProM. In: Dustdar, S., Fiadeiro, J.L., Sheth, A.P. (eds.) BPM 2006. LNCS, vol. 4102, pp. 420–425. Springer, Heidelberg (2006)
19. Rozsnyai, S., et al.: Business process insight: An approach and platform for the discovery and analysis of end-to-end business processes. In: IEEE SRII (2012)
20. Rozsnyai, S., Slominski, A., Lakshmanan, G.T.: Discovering event correlation rules for semi-structured business processes. In: ACM DEBS (2011)
21. Sharma, K., Shrivastava, G., Kumar, V.: Web mining: Today and tomorrow. In: ICECT, vol. 1 (2011)
22. Spiliopoulou, M., Pohle, C., Faulstich, L.C.: Improving the effectiveness of a web site with web usage mining. In: Masand, B., Spiliopoulou, M. (eds.) WebKDD 1999. LNCS (LNAI), vol. 1836, pp. 142–162. Springer, Heidelberg (2000)
23. van der Aalst, W.M.P.: Process Mining - Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
24. van der Aalst, W.M.P.: et al. Workflow mining: a survey of issues and approaches. Data Knowl. Eng., 47(2) (November 2003)
25. van der Aalst, W.M.P., Schonenberg, M.H., Song, M.: Time prediction based on process mining. Inf. Syst. 36(2), 450–475 (2011)
26. van der Aalst, W.M.P., van Dongen, B.F., Gunther, C.W., Rozinat, A., Verbeek, E., Weijters, T.: ProM: The process mining toolkit. In: BPM (Demos) (2009)
27. Waisberg, D., et al.: Web analytics 2.0: Empowering customer centricity (2009)