

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# Mining and Adaptivity in Automated Teller Machines

---

Ghulam Mujtaba Shaikh and Tariq Mahmood

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/48317>

---

## 1. Introduction

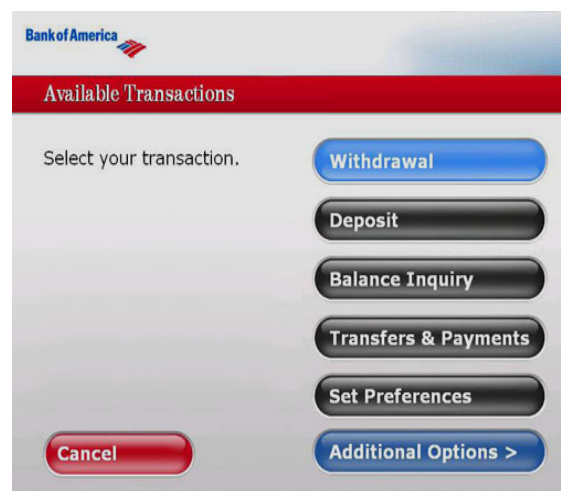
Since the past few years, the banking sector has seen a considerable application of diverse technologies for its daily operations. The most significant of such technologies has been the introduction of Automated Teller Machines (ATMs). A typical ATM is shown in Figure 1. Initially, ATMs were used only for dispensing cash, but now offer round-the-clock services for a diverse number of operations, e.g., electronic transfer of funds, paying bills, viewing past transactions of bank accounts, changing the ATM sign-in credentials etc [9]. For using ATM services, the bank issues its customer an ATM card and a PIN code. The customer inserts the card into the ATM terminal, and enters the PIN code. If the bank authenticates the PIN code, then the customer can use the ATM services. The first ATM was installed in 1967 by Barclay's Bank in the USA, and now there is hardly any bank in the world which operates without an ATM. Till March 2012, the current number of ATMs is estimated around



**Figure 1.** A typical Automated Teller Machine (ATM)

2.3 million<sup>1</sup>. Also, both the number of ATM terminals and the ATM transactions are expected to grow exponentially in the near future [8].

Notwithstanding their popularity, a major limitation of ATMs is that customers often have to wait in queue, due to a large ATM usage time of other customers [22]. This trend frequently occurs with old people, who can spend considerable time at ATMs understanding the content of the interfaces, due to their cognitive and visual disabilities [17]. Such a situation is likely to frustrate the other customers in the queue, considering that the typical ATM usage time can be considered between 20 seconds to 60 seconds [10, 17]<sup>2</sup>. Moreover, a study conducted on the elderly ATM users of Israel [22] found that these users require ATM interfaces which present content in a simpler manner with easy-to-understand jargon. This calls for designing usable (simple) ATM interfaces which can assist customers in completing their transactions more efficiently. In fact, ATMs currently support a *fixed* interaction process with their customers. Specifically, each bank selects a set of services which will be offered by its ATMs. Each time the user enters her PIN, these same services are displayed to her. The user selects her desired service, with each service being typically offered on a separate interface. For instance, Figure 2 shows the ATM interface of the Bank of America<sup>3</sup> offering the available services, and Figure 3 shows the ATM interface for the “Balance Inquiry” service.



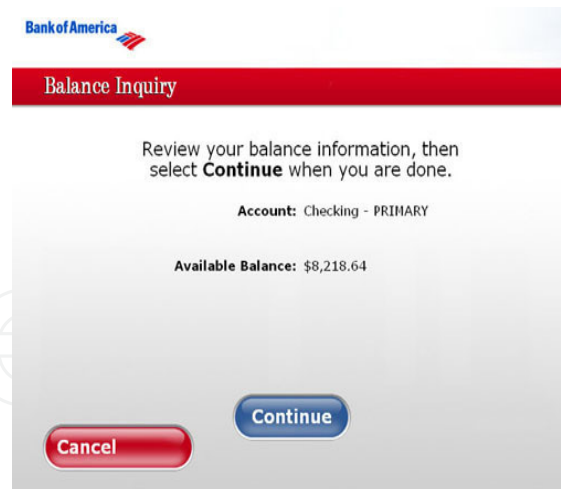
**Figure 2.** An ATM interface of Bank of America showing some of the offered services

Each time any customer enters her PIN, she is always shown the screen in Figure 2. However, suppose that a majority of customers use only the “Balance Inquiry” and “Deposit” operations, and we show a screen with only these operations, along with the “Additional Options” button for the remaining operations. Then, viewing lesser options could reduce the usage time for a majority of customers, along with making the interfaces more usable. Similarly, if many customers inquire their current balance, showing this balance autonomously to the user can reduce their usage time (as they won’t have to request for it explicitly). Finally, the rapid accumulation of ATM transaction data makes it possible to *mine* this data, in order to extract interesting information regarding the usage patterns of ATM customers [4]. This information can be used to make the ATM usage process more

<sup>1</sup> <https://www.atmia.com/mig/globalatmclock/>

<sup>2</sup> [http://www.me.utexas.edu/~jensen/or\\_site/computation/unit/stoch\\_anal/marp\\_add/marp\\_add.html](http://www.me.utexas.edu/~jensen/or_site/computation/unit/stoch_anal/marp_add/marp_add.html)

<sup>3</sup> <https://www.bankofamerica.com/>



**Figure 3.** An ATM interface of Bank of America for the “Balance Inquiry” service

*adaptive*, i.e., one that caters for the customers’ behavior. These requirements call for the design of *adaptive* ATM interfaces, which are more usable, attempt to reduce the usage time of customers, hence enhancing the customer satisfaction along with having a positive impact on customer relationship management [4]. Generating efficient transactions implies lesser waiting times (and frustration) for customers in the queue. We note that currently, no concrete work exists that involves the application of data mining techniques to ATM transactions. In fact, only simple statistics are being computed to answer minor queries of the bank managers, e.g., number of customers who used a particular ATM on some day, total amount of withdrawal on a given day etc.

In this work, we propose a set of five adaptive ATM interfaces, which are adapted to the behavior of an ATM customer population. For this, we obtain an ATM transaction dataset for an international bank in Kuwait, a Middle-Eastern country<sup>4</sup>. We initially pre-process this dataset to ensure data quality for the mining activity. Then, we mine it using the technique of “process mining” [1], i.e., mining of the ATM usage process, through the Process Miner (ProM) tool. Our results show that the customers frequently use only some ATM operations. Specifically, withdrawal is performed most frequently, followed by purchase, and balance inquiry options. A majority of customers perform these operations repeatedly, and also one after the other. We also obtain the distribution of the withdrawn amount, with respect to individual customers, the location (ATM terminal) of withdrawal, and the time of the withdrawal. These data reveal that individual customers withdraw specific amounts at specific timestamps. We also identify heavy traffic ATMs, as well as usage times of peak customer activity, on which our five adaptive interfaces can be applied to reduce the usage time.

In the first interface, we show only the ATM operations that are frequently used by the customers. In the second interface, we show only those amounts that are frequently withdrawn by the customers. In the third interface, we query the customer explicitly about performing another withdrawal. In the fourth interface, we display the customer’s current balance on several screens autonomously. Finally, in the fifth interface, we query the customer explicitly about viewing her purchase history. In order to acquire the opinion of the customers regarding these interfaces, we conducted an online questionnaire (survey), that was filled by

<sup>4</sup> We will not mention the name of this bank for the sake of privacy.

216 users who were representative ATM customers. A large majority of these users believed that the first, second and fourth interfaces could reduce their usage time, and were willing to evaluate these interfaces in real-time. Moreover, our work has been approved by the State Bank of Pakistan, which is Pakistan's banking authority<sup>5</sup>. We are currently implementing four of our interfaces for a Pakistani bank. We note that a part of this work was published in two papers [14, 15] in the International Conference of Information and Communication Technologies in 2011<sup>6</sup>.

This chapter is structured as follows. In Section 2, we discuss the state-of-the-art related to our work. In Section 3, we describe the background knowledge related to the ProM tool, i.e., the description of process mining, and how it can be carried out in ProM through a specific XML format. Then, in Section 4, we describe the data pre-processing tasks that we applied on our ATM transaction dataset, and in Section 5, we present the results of mining this dataset. In Section 6, we illustrate and describe our five adaptive ATM interfaces, and in Section 7, we present the results of our real user ATM survey. Finally, in Section 8, we conclude our work, and present the limitations of our adaptive interfaces along with the future work.

## 2. Related work

Perhaps the work most related to our approach is [12, 13]. Here, the author proposes the design and implementation of a software agent called the Personal Bank Teller (PBT). The PBT is associated with an ATM card, and personalizes (adapts) the ATM interaction for the customer, irrespective of the time and location of ATM access. It attempts to show the customer's frequently-used ATM operations towards the top of the screen, so that they are more in focus. Also, those amounts are shown towards the top which are withdrawn more frequently. These are similar behaviors to our first and second adaptive interfaces, respectively (Section 1). However, PBT operates by simply calculating the probability of access of a given operation, or amount. Our adaptive interfaces are based on more robust mining methods, and focus also on a more usable (simpler) display by removing less frequently used operations or amounts. Moreover, PBT has a smaller scope because it doesn't employ the roles of our third, fourth and fifth adaptive interfaces.

Moreover, in [8], the authors perform a questionnaire-based survey to acquire the opinions of customers regarding ATM usage in Pakistan. They show that more customers prefer using ATM services when they are provided in the national language, and also, when they are available near the customers' residences. This survey provides a motivation for international banks to open their branches in sub-urban (rural) locations in Pakistan, which can provide ATM terminals connected both locally (within Pakistan) and internationally. The scope of our work is expansive as compared to [8], because we are mining the actual ATM transactions to extract usage patterns of customers, along with a questionnaire-based survey. Also, our goal is more critical: we are not catering for the language requests of the customers, but rather, attempting to minimize their ATM usage time.

On a related note, in [19], the authors have employed data mining techniques to discover useful information concerning Customer Relationship Management (CRM) in the banking

<sup>5</sup> <http://www.sbp.org.pk/>

<sup>6</sup> <http://iciict.iba.edu.pk/>



sector. The results show that retaining old customers is very important because it costs 7 to 10 times more to make a new customer rather than to retain an old one. In comparison, although we are not aiming to directly optimize CRM, we believe that building adaptive ATM interfaces will positively influence CRM, e.g., customers will use ATM services more often if these services are adapted to their behaviors and preferences. Another work related to our approach is a research survey [7], which mentions that building and maintaining profiles of customers will increase banking transactions. It also concludes that the usage of data mining tools is essential for efficient banking functionality, which provides a strong support and motivation for our work. Moreover, we believe that we are also (indirectly) profiling different customers according to their behaviors, e.g., those who make frequent withdrawals, those who check their current balance regularly, at a given location and time.

Along with this, several enterprises are conducting simple statistical analysis on ATM transactions to increase customer satisfaction. For instance, the enterprise Prognosis [20] has implemented the ATM Transaction Manager, which provides business intelligence-related transaction statistics to enhance the transaction throughput for ATM customers. By using data warehousing techniques, it can uncover reasons for fallacies such as excessive denials of ATM services, transaction failures and slow response times. Also, the enterprise ESQ Business Services [6] has implemented the Automated Trailer Machine Transaction Analyzer (ATMTA), which provides continuous real-time monitoring of ATM transactions to perform important tasks such as dynamic calculation of transaction time, generation of user activity reports, identifying user traffic across diverse ATMs etc. Although these solutions are important, they do not involve mining of ATM transactions to extract relevant information regarding the customers' behavior. In other words, they do not adapt to what the customers are doing, in order to provide them with a more usable ATM interaction experience.

The work done in the domain of process mining [1] is also related to our approach. Given a particular domain, process mining is the branch of data mining that discovers models of processes from the event logs of this domain. In our work, the event logs are the ATM transactions, and the process is that of ATM usage by the customers. We process mine our transaction dataset through the ProM tool (Section 3). In this context, the following works will assist the reader in understanding that ProM has been applied in diverse domains, and its outputs have been extremely beneficial for these domains. In [21], the authors apply process mining to the domain of software development, in order to determine whether the evolution process of a software proceeds according to its documentation (or deviates from it), as the project progresses. The authors hypothesized that such deviations could occur towards the end of the project, when the time constraints become more severe. However, the results of mining reject this hypothesis, and showed that the throughput time of software development is longer when a particular process of development is not followed. Furthermore, process mining has been used to bridge the gap between customers' expectations about commercial products and the actual performance of the products [5]. Here, the process of usability of a particular product is mined, in order to acquire useful information about the product's usage. For instance, if the users have previous knowledge about the product, then their usability performance will be typically good for a first use of this product. Finally, in [11], the authors apply process mining to the domain of health care in order to obtain meaningful information about the patients' care flows, i.e., the process of treatment (health care) imparted to the patients. The authors apply ProM to a real-life oncology process (of a metropolitan

hospital), and show that the mining outputs can be used to improve the efficiency and efficacy of existing care flows.

### 3. Process mining and ProM tool

In this section, we will describe the techniques of process mining, and the method through which it can be carried out within the ProM tool.

#### 3.1. Background on process mining

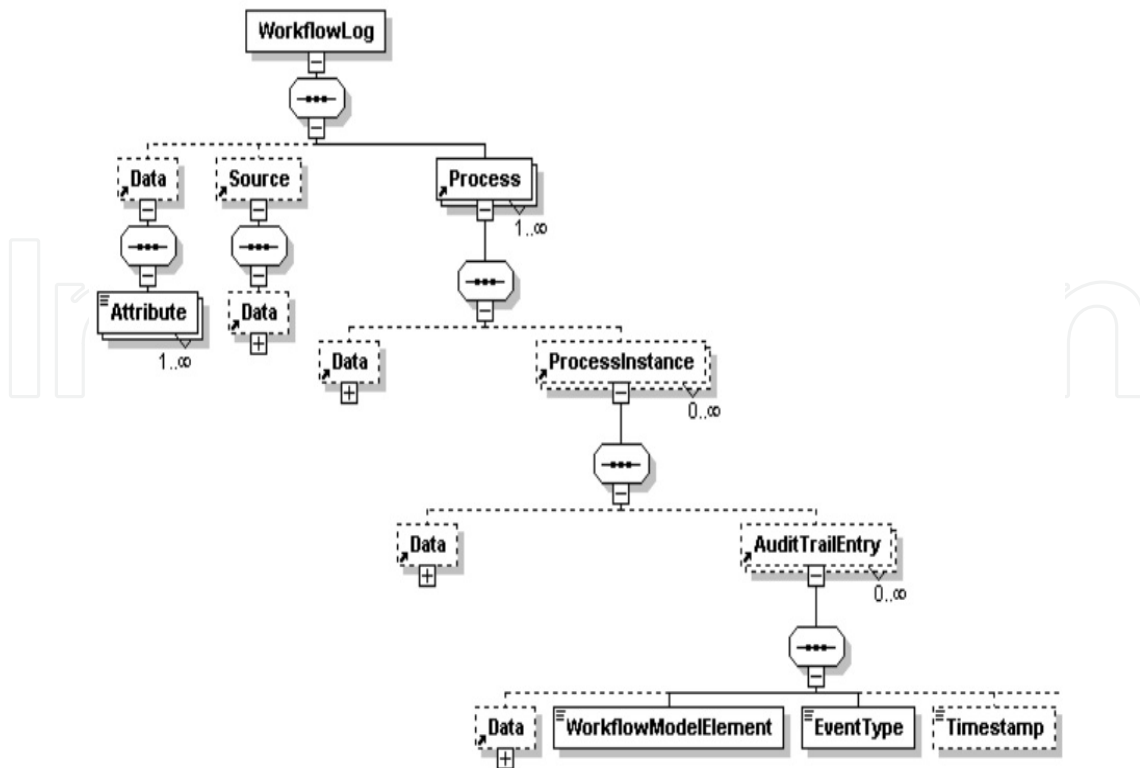
Process mining [1, 23] is a data mining technique for extracting useful knowledge from event logs in information systems. An event log logs different events as they occur within a particular process of the system. For instance, consider a computer system that keeps track of the actions of user on an E-Commerce portal. Then, it could log events such as “the start of user session”, “user buys a product”, “user views top-10 products”, “end of user session” etc. Process mining discovers useful information regarding the process available in the event logs. For instance, in the car repair example, process mining can reveal that a particular part of a car engine is repaired most frequently, and mostly through two specific repair operations. The following points describe process mining in detail:

- Each event refers to an instance of a process, also called a *process instance*,
- Each event can have an originator (the actor initiating the activity),
- Each event is identified by a particular activity; activities are associated with cases, and
- Each event can have a timestamp.

Process mining discovers causalities amongst events by using three state-of-the-art methodologies, i.e., finite state machines, Markovian framework, and neural networks [2].

#### 3.2. Background on ProM tool

ProM is an open-source tool for process mining [24]. It provides a vast variety of process mining plug-ins. A given plug-in implements one or more data mining algorithms. ProM reads files in a particular XML format, called MXML (described below). For creating an MXML file, we need an *audit trail entry*, which is an event that occurs at a particular time stamp. Each audit trail entry should refer to one unique activity. It also contains the description of the event, and refers to a process instance or a case. Also, each process instance belongs to one specific process. The MXML file format is described in Figure 4. At the root (top), there exists a WorkflowLog element, which can contain optional Data and Source elements, along with a number of Process elements. A Data element can log textual data. It comprises a set of Attribute elements. A Source element records information about the system which originated the event log (under analysis). The Process element depicts a particular process in the system, and a ProcessInstance depicts a process instance. An AuditTrailEntry element may refer to an activity (WorkflowModelElement), a type of the event (labeled as Eventtype), a timestamp (Timestamp), or a person that started (or managed) the activity (Originator). In order to convert our event logs into MXML format, we employ the Nitro tool [3]. Through Nitro, we load the event data, select the cases, activities, time stamps and the originator entities in this data, and convert it to MXML.



**Figure 4.** MXML File Format (adapted from [23])

#### 4. Data collection and data cleaning

As mentioned in Section 1, we employ an ATM transaction dataset of an international bank based in Kuwait. This dataset originally consisted of approximately 10 million transactions conducted by 5000 customers. In order to minimize the computational cost involved in data mining, we sampled this dataset using random sampling technique [4]). Our sample dataset consisted of 2 million transactions for 676 customers. These transactions are recorded from 17th September 2009 (17-09-2009) to 1st December 2010 (01-12-2010). They are represented by 45 attributes (columns) and are stored in Microsoft Excel format. In order to select a reduced (more reasonable) set of attributes, we applied *data cleaning* techniques to the sampled transaction dataset [4]. Specifically, we deleted the redundant attributes, e.g., there is both a name and a unique ID available for the customer, from which we selected only the ID. Moreover, some attributes are irrelevant for our analysis. For instance, each transaction is assigned a tracking number for data archival; this is irrelevant because we are focusing on the customers' usage patterns. Also, there were several attributes which contained missing values, i.e., data for these attributes was not logged completely, e.g., an audit number (assigned to each transaction) was not available for each transaction. We also performed the Pearson's and Chi-squared co-relation test to detect co-related numerical and categorical attributes, respectively. If two attributes were positively co-related, we considered only one of them [4]. In summary, we ignored all columns that were redundant, irrelevant, contained missing values, or were (pair-wise) positively co-related.

After data cleaning, we were left with 9018 transactions of 276 different customers, represented over 10 attributes. We describe these attributes as follows (for a given performed transaction):



1. *PAN*: Stores the ATM card number,
2. *Cust\_Code*: Stores a unique code of the customer,
3. *Proc\_Code*: Stores the type of transaction, e.g., withdrawal, deposit, balance inquiry etc.,
4. *DateTime*: Stores the time stamp,
5. *Amount*: Stores the amount being withdrawn or deposited,
6. *Response\_Code*: Stores the possible responses to the transaction, e.g., approved, denied, invalid pin code entry etc.,
7. *Terminal\_ID*: Stores the ID of the ATM cabin at which the transaction was performed,
8. *Terminal\_Loc*: Stores the name of the location where the ATM cabin is installed,
9. *Acquiring\_ID*: Stores the ID of the bank which has installed the ATM cabin, and
10. *Authorizer\_ID*: Stores the ID of the bank whose card is being used for transactions (since card holders of different banks can use the ATM).

We converted our cleaned transaction dataset into MXML using Nitro. We chose the *PAN* attribute as the Case attribute of the MXML format, *Cust\_Code* as the Originator attribute, *Proc\_Code* as the Activity attribute, and *DateTime* as the Timestamp attribute (Section 3.2). The ProcessInstance tag identifies a particular transaction, i.e., a process instance. Moreover, the AuditTrailEntry tag specifies the detail of one particular action being performed by the customer, e.g., a “withdrawal”, which is specified by the WorkflowModelElement tag. Besides this, the EventType tag represents the type of an event, which we have set as default to “start”, i.e., a transaction has started. When a transaction finishes, i.e., when the customer takes a final action, the EventType is “complete”, i.e., the transaction has finished. Finally, the Originator tag represents the *Cust\_Code*, i.e., the particular customer doing the transaction.

## 5. Process mining results

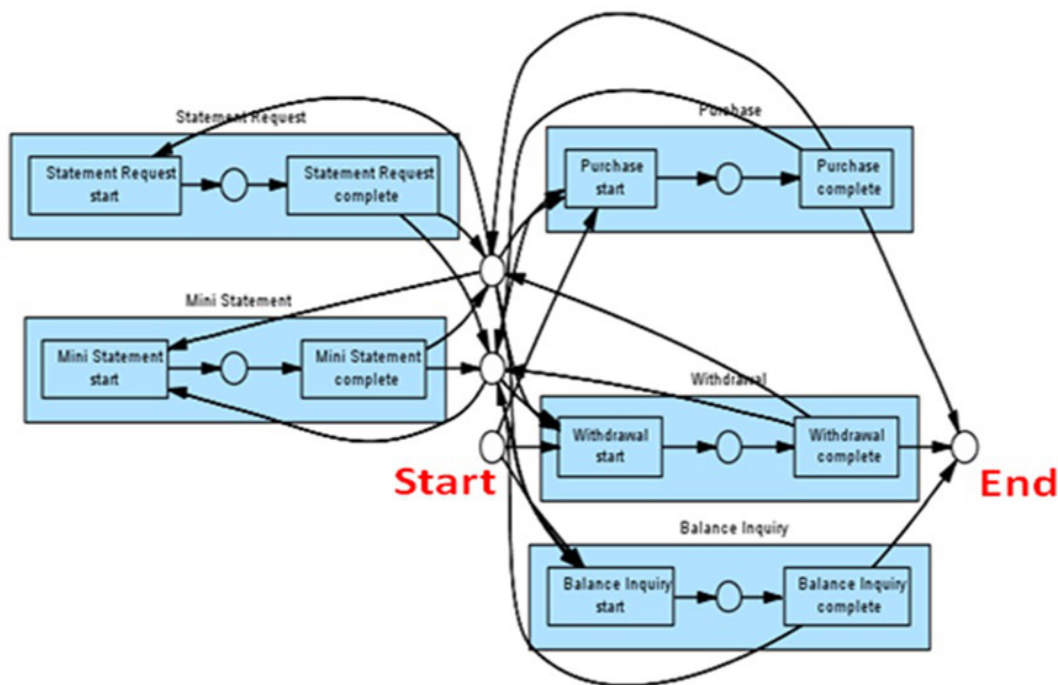
In this section, we employ eight ProM plug-ins to mine useful knowledge from our ATM MXML file. We recall from Section 1 that our aim is to use this knowledge to develop adaptive ATM interfaces, which have a tendency to minimize the ATM usage time for a population of customers, particularly at heavy traffic ATMs. We discuss our results in the following eight sub-sections.

### 5.1. Generic transaction distribution

We initially acquired the frequency distribution of the different ATM transactions, i.e., the number of transactions conducted for a given ATM operation. We note that a pattern or an event that occurs frequently is deemed interesting for data mining purposes [4]. The generic transaction distribution of our dataset is shown in Table 1. We see that customers perform 10 different ATM operations. In order of decreasing frequency, these include the withdrawal of money, purchase of products, balance inquiry (query for the current account balance), open-ended credit (acquiring loan as a credit on account), cash deposit (in the customer’s own account), PIN validation (on inserting the ATM card), mini-statement request (a short version of the bank statement), open-ended cash (acquiring loan as cash), statement request, and PIN change (request to change PIN code). Also, the frequencies of withdrawal (5213), purchase

S. No.	Transaction	Frequency
1	Withdrawl	5213
2	Purchase	1916
3	Current Balance Inquiry	1064
4	Open-Ended Credit	338
5	Own-Account Cash Deposit	201
6	PIN Validation	131
7	Mini-Statement Request	101
8	Open-Ended Cash	34
9	Statement Request	10
10	PIN Change	10

**Table 1.** Generic Distribution of ATM Transaction Dataset (adapted from [15])



**Figure 5.** Generic process of ATM transactions mined by the Alpha++ algorithm (adapted from [15])

(1916) and balance inquiry (1064) are quite large as compared to those of other operations. Due to these large frequencies, we conclude that the withdrawal, purchase and balance inquiry operations are most important, and we will learn an adaptive ATM behavior regarding these operations only.

## 5.2. Alpha++ algorithm

In order to acquire further information regarding the ATM usage, we employ the Alpha++ algorithm plug-in. This plug-in mines the generic process that is implicitly present in the event logs (MXML file). This process is represented as dependencies (directed arcs) between events [1]. Figure 5 shows the output of the Alpha++ algorithm for our ATM transaction dataset. The generic ATM process comprises 5 (out of 10) transactions, i.e., withdrawal, balance inquiry,

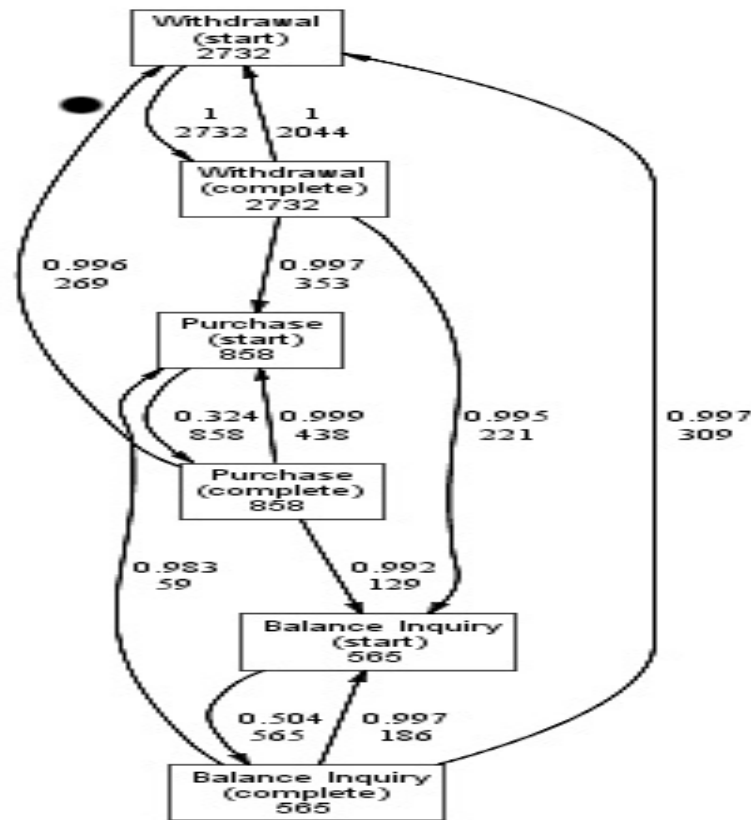
purchase, mini-statement request, and statement request. Each transaction has a “start” and “complete” event, signaling the start and end of this transaction. These events are represented collectively as a blue rectangle. The circles outside the rectangles represent junctions of process flow for modeling dependencies between events. We have labeled the starting junction as “Start”, which represents the start of a transaction (after PIN authorization). Also, we have labeled the terminating junction as “End”, which represents the end of a transaction. From “Start”, customers either performed a purchase, withdrawal or a balance inquiry. Moreover, customers have requested for the bank statement and the mini-statement, after withdrawal and after a purchase. After this, customers can again perform withdrawal, balance inquiry or purchase. Finally, the “End” junction shows that customers terminate their transactions only after performing one of these three transactions. In summary, customers generally perform withdrawal, balance inquiry and purchase operations, sometimes repeatedly, and follow these up by sometimes viewing their bank statements. This result confirms our statement that withdrawal, balance inquiry and purchase are the more important ATM operations. In order to acquire further details about the sequence of usage (dependencies) of these operations, we employ the Heuristic Miner plug-in (next section).

### 5.3. Heuristic miner

The Heuristic Miner is a ProM plug-in which uses heuristics to provide mathematical details about the dependencies illustrated by the Alpha++ algorithm [1]. This information is displayed in a graphical structure called the heuristic net, which is very similar to a directed cyclic graph. The heuristic net for our ATM transaction dataset is shown in Figure 6. Here, each box represents a particular event, and the number within the box represents the frequency of this event. Also, a directed arc from event A to event B indicates that B occurred after A. Each arc is labeled with two numbers (one above the other). The upper number indicates the likelihood (probability) that B occurs *immediately* after A, i.e., the user performs no other transaction between A and B. The lower number indicates the frequency with which B occurred after A. As compared to Alpha++, the heuristic net shows the “Start” and “Complete” events for withdrawal, purchase and balance inquiry only. In other words, it ignores the transactions concerning the bank statement requests, possibly due to the smaller frequency of these operations.

From Figure 6, we see that a total of 2732 withdrawals have been performed, out of which 2044 have been performed one after another, 309 have been performed after balance inquiry, and 269 after a purchase. The likelihood of these transactions is very high, i.e., 1, 0.997 and 0.996 respectively. Moreover, balance inquiry was performed 565 times, out of which 186 were performed one after another, 221 after a withdrawal, and 129 after a purchase. These transactions also have a high likelihood, i.e., 0.997, 0.995 and 0.992 respectively. Finally, purchase was performed 858 times, out of which 438 were performed after one another, 353 after a withdrawal, and 59 after a balance inquiry. The likelihood of these transactions is high, i.e., 0.999, 0.997 and 0.983 respectively. These statistics reveal that customers have a strong tendency to repeat their previous operation, particularly withdrawal and purchase. The largest repetition occurs for the withdrawal operation, followed by purchase and balance inquiry.

Also, customers tend to perform different operations in a sequence, albeit with a reduced frequency as compared to their repetitive behavior. More notably, customers tend to withdraw



**Figure 6.** Heuristic net for the ATM transaction dataset (adapted from [15])

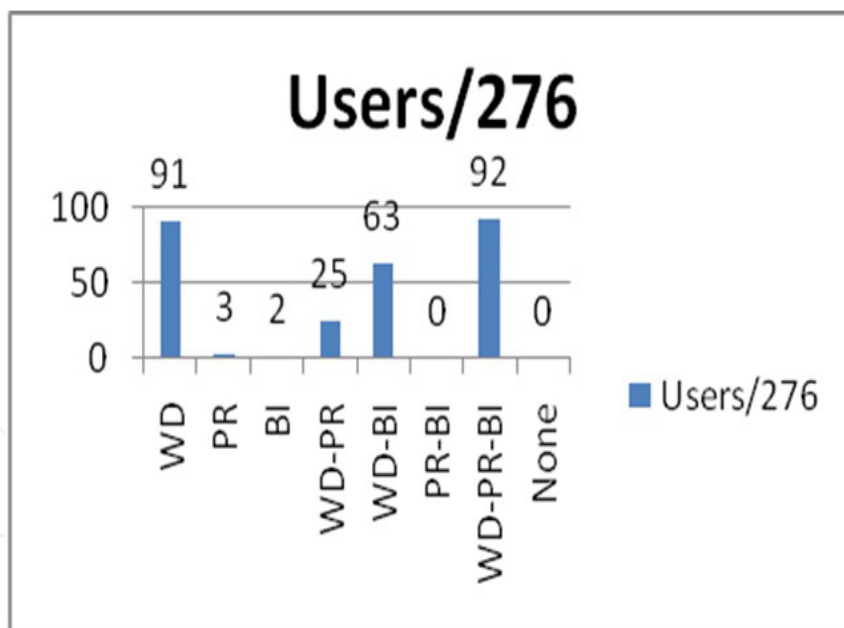
money after a balance inquiry or a purchase, and to inquire the balance after a withdrawal or a purchase. Finally, we note that users performed several transactions between starting and completing a purchase (likelihood=0.324), or the balance inquiry (likelihood=0.504). However, users completed their withdrawal operations immediately after starting them (likelihood=1). These results confirm that withdrawal, balance inquiry and purchase are the most important operations. Customers perform them repeatedly, as well as one after the other. If only these operations are shown to the customer, she might spend lesser time browsing through the ATM options, and on a simpler interface. We now employ a series of ProM plug-ins to further investigate the actions of the customers regarding these operations.

**5.4. Originator by task matrix**

ProM allows us to obtain the distribution of transactions for each individual customer, through the “Originator by Task Matrix” plug-in. A snapshot of this distribution is shown in Figure 7. The left-most column is the ID of the customer followed by the frequencies of balance inquiry, purchase and withdrawal respectively. These figures indicate the customers who have carried out a particular transaction maximum, or minimum, number of times. For instance, the first customer has performed balance inquiry and purchase the maximum number of times (126 and 22 time respectively), while the second one has performed withdrawal the maximum number of times (22). Although everyone has made a withdrawal, there are customers who haven’t performed either balance inquiry and/or purchase.

originator	Balance Inquiry	Purchase	Withdrawal
521175999004306	126	22	34
5211759990020302	18	10	78
5211759990020310	0	4	4
5211759990119302	0	0	6
5211759990127305	16	10	8
5211759990135308	4	0	2
5211759990274305	6	0	48
5211759990282308	2	0	44
5211759990346301	0	0	28
5211759990389012	0	0	12
5211759990442308	0	12	28
5211759990477304	0	0	28
5211759990493301	0	0	12

**Figure 7.** Distribution of withdrawal, balance inquiry, and purchase for several customers (adapted from [15])



**Figure 8.** Transaction-based Customer Distribution (adapted from [15])

### 5.5. Transaction-based customer distribution

To acquire further details about this behavior, we determine the frequency of customers who have performed one or more operations, either individually or collectively (together). These statistics are shown in Figure 8. The y-axis represents the number of customers. On the x-axis, WD, PR and BI represent customers who have made a withdrawal, purchase,



Timestamp	Count	Location	Min_Amnt	Max_Amnt
12:00 AM TO 01:00 AM	120	L1	1	500
01:00 AM TO 02:00 AM	81	L1	1	300
02:00 AM TO 03:00 AM	64	L2	1	185
03:00 AM TO 04:00 AM	34	L2	2	350
04:00 AM TO 05:00 AM	9	L3	1	615
05:00 AM TO 06:00 AM	38	L3	1	820
06:00 AM TO 07:00 AM	36	L4	1	1000
07:00 AM TO 08:00 AM	77	L4	1	500

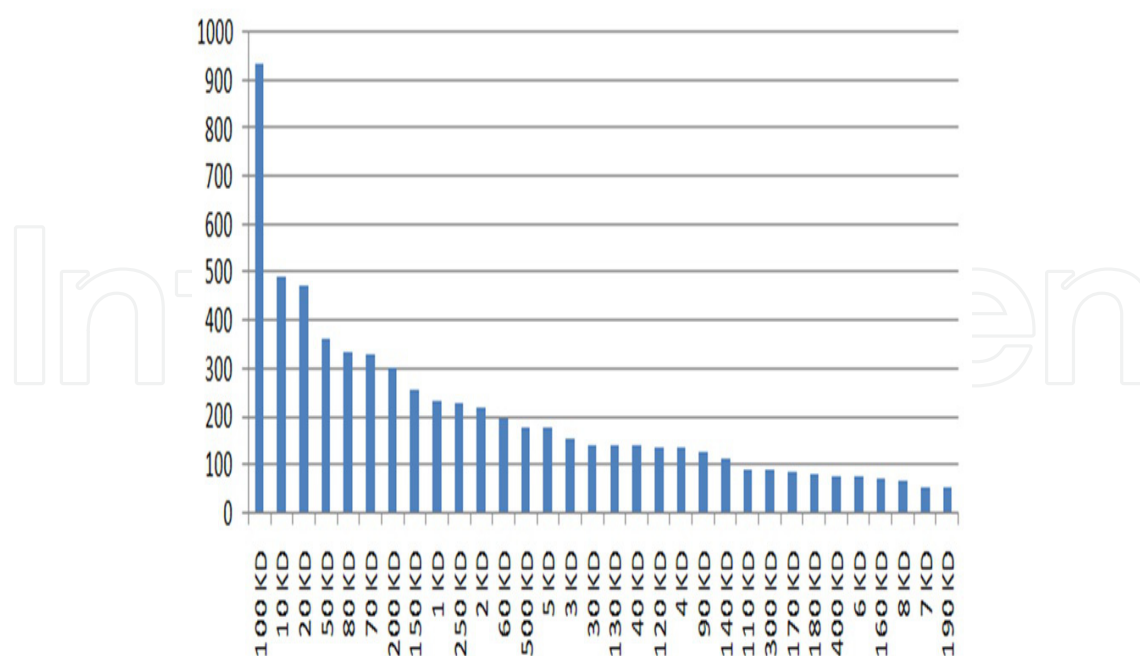
**Table 2.** Location-based withdrawl distribution (adapted from [15])

and balance inquiry respectively. Also, WD-PR, WD-BI, PR-BI, and WD-PR-BI represent customers who have made a withdrawl and purchase together, a withdrawl and balance inquiry together, a purchase and balance inquiry together, and all three operations together, respectively. Finally, the label “None” represents those customers who didn’t perform any transaction. From Figure 8, we see that every customer performs one or more transactions. Also, withdrawl was performed by 91 customers, purchase by 3 customers and balance inquiry by 2 customers. Viewing the operations collectively, 92 customers perform all three operations, 25 make withdrawl and purchase together, while 63 make withdrawl and balance inquiry together. Also, no one performs purchase and balance inquiry together. These results show that a majority of the customers either perform all three operations collectively, or only the withdrawl operation. Based on this, we deem the withdrawl operation to be the most important one for the sake of designing an adaptive ATM behavior. To this end, we will now mine three separate distributions regarding withdrawl.

## 5.6. Location-based withdrawl distribution

We also acquired the withdrawl distribution based on the location of ATM terminals, shown in Table 2. In our transaction dataset, we have withdrawl information of 16 ATM locations in Kuwait. In order to avoid a length analysis, we show a snapshot of the withdrawl distribution for four locations only, i.e., L1, L2, L3 and L4 (column ‘Location’)<sup>7</sup>. We have grouped the withdrawls according to one-hourly time periods (column ‘Timestamp’). For each time period, we show the number of customers who have made a withdrawl in this period (column ‘Count’), and the minimum and maximum amounts withdrawn in this period (columns ‘Min\_Amnt’ and ‘Max\_Amnt’). We can see that, at L1, around 200 customers access the ATM between 12 AM and 2 AM, and withdraw between 1 KD - 500 KD. At L2, around 100 customers access the ATM between 2 AM and 4 AM, and withdraw between 1 KD - 350 KD. Similarly at L3, around 50 customers access the ATM between 4 AM and 6 AM, and withdraw between 1 KD - 820 KD. Finally, at L4, around 100 customers access the ATM between 6AM and 8AM, and withdraw between 1 KD - 1000 KD. These statistics reveal the overall withdrawl behavior of customers at a given ATM location. Using them, we can also calculate the *usage rate* of an ATM, i.e., the number of withdrawls (operations) performed in a given timestamp. For instance, there might be heavy usage traffic at some ATM from 1300-1400 (during lunch

<sup>7</sup> The actual names of the location are kept anonymous.



**Figure 9.** Distribution of the amount withdrawn by customers, KD = Kuwaiti Dinar (adapted from [15])

break), because users collectively perform 80 withdrawals in this interval. These statistics can help us in identifying heavy traffic ATMs, at which a large number of withdrawals are being performed daily, at various timestamps of peak activities. Then, an adaptive ATM behavior can be employed at these timestamps in order to minimize the ATM usage time.

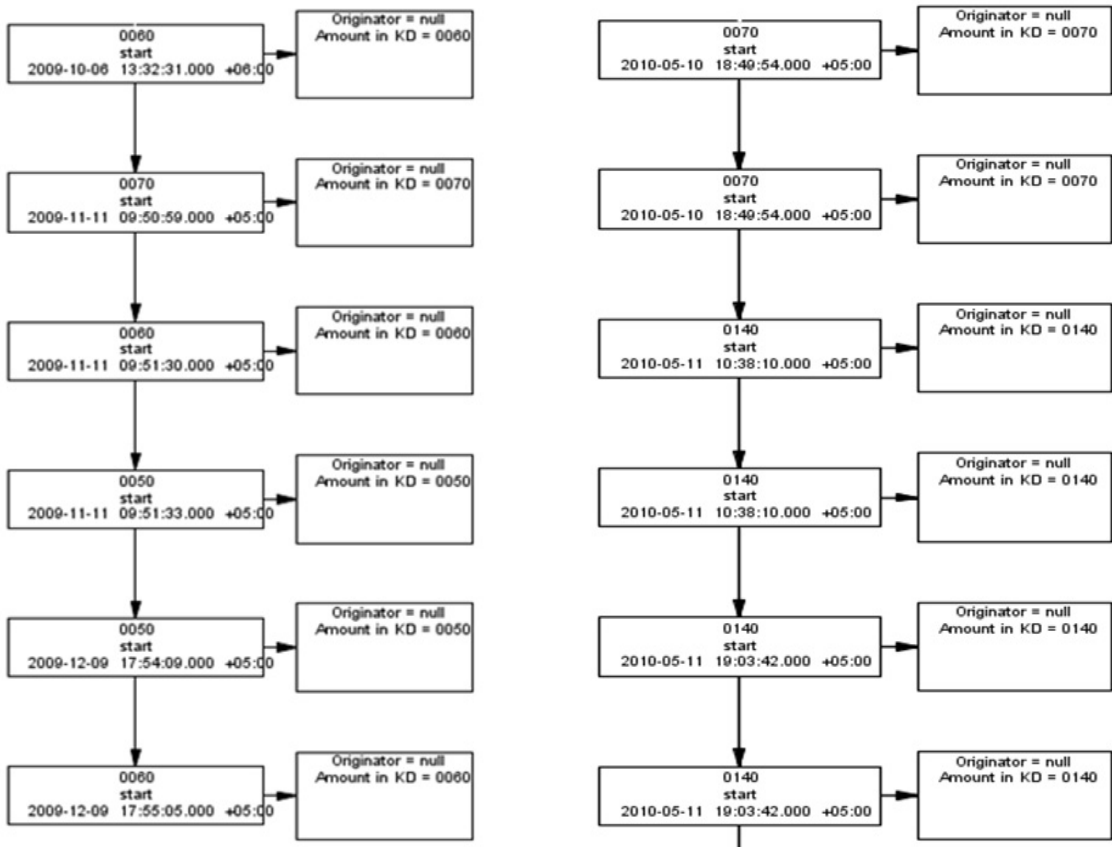
### 5.7. Amount-based withdrawal distribution

Using another ProM plug-in, we acquired the distribution of the amount withdrawn by the customers. The results are shown in Figure 9. Here, the X-axis shows the amount in Kuwaiti Dinar (KD)<sup>8</sup>, and the Y-axis shows the frequency of the withdrawn amount. The distribution shows that 100 KD has been withdrawn a maximum number of times (around 900), while amounts of 10 KD and 20 KD have been withdrawn around 500 times. Moreover, the amounts of 50 KD, 70 KD, 80 KD and 200 KD have been withdrawn between 300 - 350 times. Customers have also withdrawn more than 100 KD, and very small amounts, e.g., 1KD and 2KD, as well. In other words, we can acquire the top-N frequent amounts being withdrawn by customers. If only these amounts are shown to a customer at peak activity times, she will have to browse lesser options on a simpler interface, possibly reducing her usage time.

### 5.8. Customer-based withdrawal distribution

Finally, we acquired the withdrawal distribution for each individual customer. This provides more detail, as compared to the generic distribution shown in Figure 8. For instance, Figure 10 shows the withdrawal distribution for two customers including timestamps of withdrawals. Let's assume that the distribution on the left and right belongs to Customer A and Customer

<sup>8</sup> Dinar is the currency of Kuwait



**Figure 10.** Withdrawal distribution for two individual customers

B respectively <sup>9</sup>. We can see that, from October 2009 to December 2009, Customer A has withdrawn 60 KD, 70 KD, 60 KD, 50 KD, 50 KD and finally 60 KD, in this order. These withdrawals have been made in the mornings, afternoons and evenings. Also Customer B has withdrawn only two different amounts, i.e., 70 KD and 140 KD, over a period of two days in May 2010. These withdrawals have been made in the mornings and evenings. Until now, we have talked about developing adaptive interfaces for a customer population. However, using the customer-based withdrawal statistics, we can develop adaptive interfaces for each customer individually. This, however, is part of our future work (Section 8).

## 6. Adaptive ATM interfaces

In this section, we use our ProM results to describe and illustrate five adaptive ATM interfaces. In doing so, we will use the Pakistani currency PKR (Pakistani Rupees) rather than Kuwaiti Dinar (KD). Let us initially summarize our ProM results (abbreviated with 'R') as follows:

- **R1:** Withdrawal is the most frequent operation performed by customers, followed by purchase and balance inquiry,
- **R2:** These operations are frequently repeated, with withdrawal having the highest repetition frequency,

<sup>9</sup> Customers are identified by their ATM card number, which we keep confidential.

- **R3:** These operations are also performed in a sequence (one after another); sequences containing withdrawal have the highest frequency,
- **R4:** A majority of customers perform only the withdrawal operation,
- **R5:** A majority of customers withdraw only specific amounts,
- **R6:** It is possible to obtain the rate of the withdrawal operation at a given location (ATM terminal).
- **R7:** A majority of customers perform withdrawal, balance inquiry and purchase collectively.

Recall that our primary goal is to reduce the ATM usage time for the customers (Section 1). This can be achieved particularly at locations with heavy usage traffic. Due to R6, we can easily identify such ATMs. For instance, ATMs installed near a large corporate environment, or a busy thoroughfare, can witness heavy traffic at peak activity times, e.g., from 0900 to 1000, 1330-1530, 1900-2100 etc. We now define five adaptive ATM interfaces which can be employed at heavy traffic ATM's:

1. **Interface 1:** This shows only the most frequently used ATM operations to the customer,
2. **Interface 2:** This shows only the most frequently withdrawn amounts to the customer,
3. **Interface 3:** This queries the customers regarding their willingness to perform another withdrawal (after a withdrawal)?
4. **Interface 4:** This shows the customer's current balance on several screens autonomously,
5. **Interface 5:** This queries the customers regarding their consent to view the purchase history (or not)?

Let us discuss these interfaces separately.

### 6.1. Interface 1

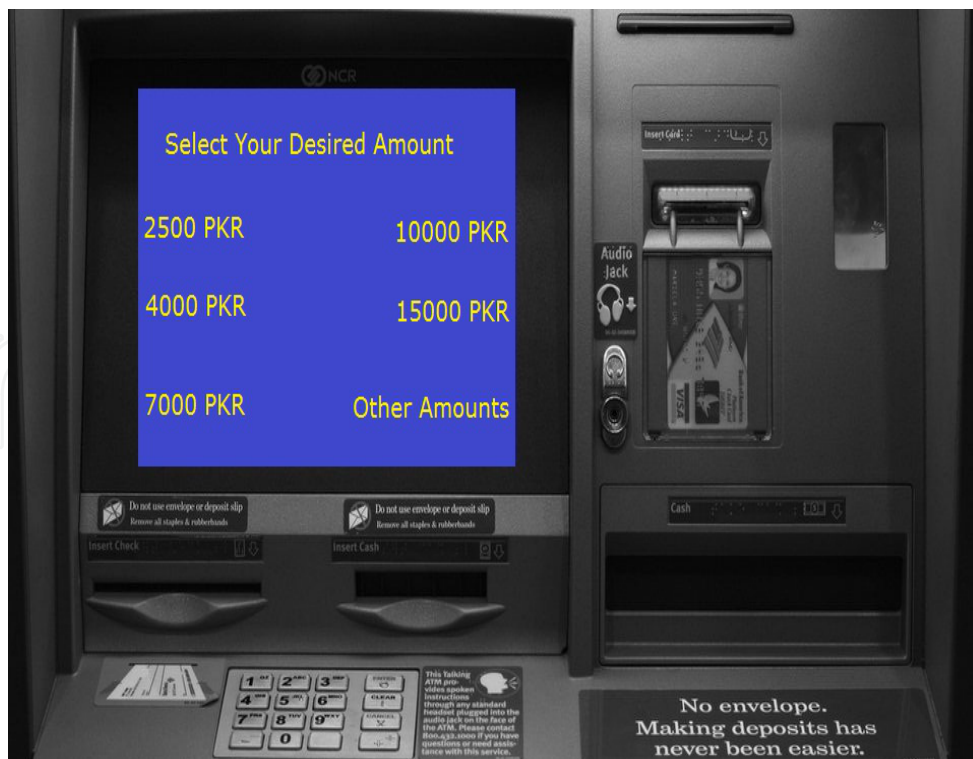
As can be seen from R1, R2, R3 and R4, customers frequently use only a subset of the ATM operations. At a heavy traffic ATM, we can only show these specific operations to the user, once she logs on to her account. The remaining operations can then be accessed through some other option. Such an interface is shown in Figure 11. Here, the screen only shows the "Cash Withdrawal" and "Balance Inquiry" operations, while "Other Operations" allows access to the remaining ATM operations. Showing lesser options leads to more usable interfaces, as customers can avoid the cognitive effort required in browsing through all ATM operations, which are typically 5-10 in number (Figure 2). Also, in relationship with the principles of usability, online customers prefer to view a smaller number of text options [16]. We believe that both these factors will lead to a reduction in the ATM usage time.

### 6.2. Interface 2

As can be seen from R4 and R5, customers frequently use the withdrawal operation, and withdraw only specific amounts. Typically a pre-defined set of amounts is displayed to every customer by default, which could be 5-10 in number. On the other hand, if we show only the frequently-withdrawn amounts, it is possible that the overall usage time at a heavy



**Figure 11.** Adaptive Interface 1 showing only the frequently used options to the customer



**Figure 12.** Adaptive Interface 2 showing only the frequently withdrawn amounts to the customer



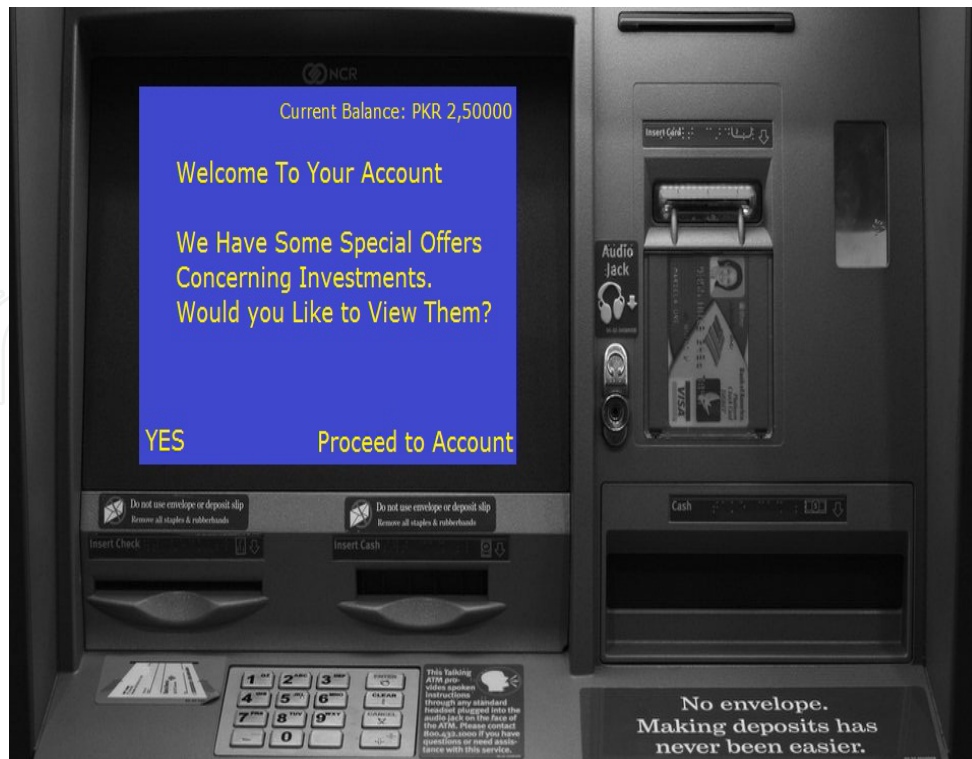


**Figure 13.** Adaptive Interface 3 querying the customer directly about performing another withdrawal

traffic ATM will be reduced, in a given time frame. Such an interface is shown in Figure 12. Here, the screen shows amounts of PKR (Pakistani Rupees) 2500, 4000, 7000, 10000 and 15000, which could be the top-5 frequently withdrawn amounts. Note that we have the flexibility of showing top-N amounts. In order to withdraw other amounts, customers can select “Other Amounts”. We believe that such an interface can assist a majority of users in quickly selecting their desired amount. For instance, in Pakistan, there are not many ATMs which display the withdrawal amount of PKR 7000. At a given ATM, if 80% of customers want to withdraw this amount, they have to select “Other Amounts” and then type this amount. This process could consume from 10-30 seconds, because the response time of ATMs could be slow [18]. So, showing PKR 7000 in the initial withdrawal amounts could minimize this time delay considerably. Similarly to Interface 1, Interface 2 is also a usable interface showing lesser amount of text (amounts).

### 6.3. Interface 3

As can be seen from R2 and Section 5.3, a large number of customers who withdraw money, follow it up by making another withdrawal. In essence, customers can perform withdrawals in a sequence. In a typical scenario, customers have to return to the main ATM menu (showing all the operations) after a withdrawal, and then re-select withdrawal from this menu. This process could consume from several seconds to half a minute, depending on the ATM’s response time [18]. If we can autonomously query the customer about performing another withdrawal, she can avoid wasting this time. Such an interface is shown in Figure 13. Once the user has completed a withdrawal, we can show this interface to her. If she selects “YES”, the withdrawal amounts will be displayed to her (as in Figure 12). Otherwise, she can select “Other



**Figure 14.** Adaptive Interface 4 showing the current balance before a withdrawal

Operations” (the balance being shown towards the top-right will be explained in next section). Interface 3 is another example of a usable interface: it contains a small amount of text which can be read clearly.

#### 6.4. Interface 4

As can be seen from R3 and Section 5.3, customers tend to perform a balance inquiry after a withdrawal operation and also after a purchase. This requires accessing the main menu and selecting “Balance Inquiry”. We have seen that this could consume up to half a minute, depending on the ATM response times. This time could be saved if the ATM displays the current balance autonomously for the customer. This could be done on several screens. For instance, Figure 14 shows a balance of PKR 2,50000 (at the top-right corner) when the user logs on to her account. Also, Figure 15 shows a balance of PKR 1,20000 just before a withdrawal operation, and Figure 13 shows a balance of PKR 15,000 just after a withdrawal operation. In all these interfaces, we cater for the privacy of customers by displaying the balance in smaller fonts size, and in a corner of the screen. This would prevent other customers from viewing the balance (clearly) over the shoulder of the customer. This situation could occur in open-space ATMs (Figure 16), but not in ATMs installed within cabins (Figure 17).

#### 6.5. Interface 5

Our final adaptive interface (Interface 5) is regarding the purchase behavior of customers. Note that a purchase is made by an ATM card in some market, but not at an ATM machine. From R1, R2, and R3, we can see that purchasing is a frequent operation, occurs repeatedly, as well as after a withdrawal and a balance inquiry. After a purchase is made, customers



**Figure 15.** Adaptive Interface 4 showing the current balance after a withdrawal



**Figure 16.** An ATM installed in an open space

could be interested in viewing their purchase history, which is detailed in the bank statement. Customers typically access the bank statement from the main menu, which could consume time. In order to minimize this time, we can autonomously query the customer to view her purchase history when she logs on (Figure 18), and when she logs out (Figure 19). Similarly to Interface 3, Interface 5 is also a usable interface containing a small amount of clearly-legible text. One final comment is concerning R7; there exists a user majority which performs withdrawal, balance inquiry and purchase collectively. To cater for this trend, we have proposed Interface 4 and Interface 5, which attempt to reduce usage time regarding



**Figure 17.** An ATM Cabin



**Figure 18.** Adaptive Interface 5 autonomously querying the customer about viewing her purchase history, at logout

balance inquiry and purchase respectively. Note that we have proposed Interface 2 and Interface 3 for the majority of users who perform only the withdrawal operation.

## 7. ATM usage questionnaire and results

We have described five different types of adaptive ATM interfaces. We believe that these interfaces could considerably reduce the ATM usage time at heavy traffic ATMs. We also have the flexibility to apply these interfaces at specific timestamps related to heavy traffic, for instance, during lunch breaks, or in the evenings. In order to acquire the opinions of the ATM customers regarding these interfaces, we conducted an online survey (questionnaire)





**Figure 19.** Adaptive Interface 5 autonomously querying the customer about viewing her purchase history, at login

comprising 15 statements<sup>10</sup>. In all, 216 users completed the questionnaire, over a period of two months (from 1st December 2011 to 24 January 2012). They provided three types of data regarding ATM usage, which we analyze in the following three sub-sections.

### 7.1. Basic analysis

Some basic analysis about the users is listed below:

- 81% of the users were males and 19% were females,
- 71% of the users were employed (either permanently or part-time),
- 80% of the users were Pakistani residents with the remaining 20% residing internationally,
- 94% of the users owned an ATM card; half of the users owned one card, and 45% owned two or more cards, and
- 63% of the users use ATM for 30 seconds to 1 minute, while 21% take 30 seconds or less. Around 15% of the users take more than a minute.

Based on these statistics, our users can be considered representative of the typical ATM users: they comprise both males and females, are both employed and non-employed, and are located across different countries. They own one or more ATM cards, and use ATM services for typical ATM usage times [10, 17].

<sup>10</sup> Available online at <https://docs.google.com/a/nu.edu.pk/spreadsheet/viewform?formkey=dEtpa0FvRTRxTEZZOFNnRTZaOHdjB1E6MQ#gid=0>



## 7.2. Analysis regarding ATM usage

We acquired further details from the users regarding their ATM usage, which are listed below:

- 75% of the users are satisfied with their ATM usage, while only 3% are dissatisfied,
- Users are satisfied due to several reasons, i.e., ATMs are much faster as compared to manual bank transactions (82%), avoid interaction with bank personnel (64%), are more accessible (64%), and facilitate the usage of the same ATM card at ATMs of different banks (58%),
- Users are dissatisfied due to several reasons, i.e., ATMs are mostly out of service (67%), are insecure (44%), and charge high fees on the usage of ATMs of other banks (30%). More importantly, around 30% of users believe that old people take too much time using ATMs, while another 30% get frustrated by waiting in queues to use ATM.

Generally speaking, around 60% of the users are dissatisfied due to more time taken by other customers in using ATMs. *This provides a concrete motivation for the application of our work, i.e., applying adaptive approaches to ATM usage processes to minimize the ATM usage time.* Concerning the issue of insecurity, users feel more insecure in using open-space ATMs; 70% of the users believe that they will feel more secure using ATM's installed in cabins. Based on this, *we believe that our adaptive approach can serve the users in a better way if it were applied inside ATM cabins only.*

## 7.3. Analysis regarding adaptive ATM interfaces

Finally, we acquired opinions of the users regarding our adaptive ATM interfaces, which are listed below.

- We inquired about Interface 4, i.e., whether (or not) users would prefer the system to show the current balance on several screen simultaneously. 75% of the users want to view the balance in this way, but under secure conditions, in which people in the queue won't get a chance to view the balance over the customer's shoulder; this condition can be easily satisfied by using ATM cabins.
- We inquired about Interface 1, i.e., whether (or not) users would prefer to view only frequently used ATM operations. Around 38% of users are willing to view frequent operations (because it will save time), while a similar percentage of users is not. However, the remaining 25% of users, although indecisive, are willing to give it a try. Overall, around 65% of users are willing to view frequent operations.
- We inquired about Interface 2, i.e., whether (or not) users would prefer to view only frequently-withdrawn amounts. 35% of users are willing to view these amounts but around 50% believe that showing all amounts is the better option. However, around 20% of (indecisive) users are willing to give it a try. In essence, more than half of the users believe that showing Interface 2 could be a good option.
- We inquired about the need of heavy traffic ATMs to assist users in completing their ATM transactions more efficiently. Around 70% of users believe that this need should be satisfied, particularly in hours of peak activities.

In conclusion, a majority of users are willing to adopt Interface 1, Interface 2, and Interface 4. Also, users think these interfaces should be shown in ATM cabins witnessing typical heavy traffic. Note that we deliberately didn't query the users regarding Interface 3 and Interface 5. We didn't query regarding Interface 5 because, in Pakistan, ATM customers do not typically make many purchases using ATM cards<sup>11</sup>. Regarding Interface 3, we didn't consider the opinion of the users as important; as system designers, we decided in advance to implement Interface 3 whenever a majority of withdrawals are found to occur repeatedly (one after another).

## 8. Conclusions and future work

In this paper, we have applied process (data) mining techniques on a transaction dataset of Automated Teller Machines (ATMs). Based on the results, we have proposed a set of five *adaptive* ATM interfaces, which can reduce the ATM usage time at heavy traffic ATMs and particularly at peak activity hours. Our interfaces have the potential to reduce the frustration of the users waiting at ATM queues, display much simpler (usable) screens (e.g., for old people), and can have a positive impact on customer relationship management. The first interface shows only the most frequently-used ATM operations, and the second one shown only the most frequently-withdrawn amounts. The third interface queries the user explicitly for another withdrawal (after the current withdrawal is completed), and the fourth one autonomously outputs the balance inquiry on several screens. Finally, the fifth interface queries the user explicitly for showing her purchase history. Interfaces 1, 2, 3 and 5 are examples of usable (simpler) interfaces containing lesser amounts of text. They can reduce the cognitive effort required by old people in using ATMs, possibly reducing their ATM usage time. In order to acquire the opinions of the customers regarding our interfaces, we conducted a real-user survey. The survey consisted of 15 statements which were filled by 216 respondents. The results show that a large majority of representative ATM users (more than 70%) are willing to employ the first, second and fourth interfaces<sup>12</sup>.

Our work has been approved by the State Bank of Pakistan, the primary authority which manages all banking operations in Pakistan. We are currently applying process mining on the dataset of a Pakistani bank. Our preliminary results show that several of our interfaces can be applied on the ATMs of this bank in heavy traffic hours. We will test these interfaces on an ATM testbed, which is provided by the multi-national company Transaction Processing Systems<sup>13</sup>. Moreover, our interfaces are currently adapted to the behavior of a customer population. Specifically, we mined the transaction dataset of around 250 customers, and suggested interfaces which are applicable for a majority of these customers. As part of the future work, we plan to propose a set of adaptive interfaces for each customer *individually*, i.e., we will mine the sequence of operations for each user separately, and use the results to propose the adaptive interfaces. For instance, for one customer, our second adaptive interface can show withdrawal amount of PKR 5000, 10000, and 20,000. For another customer, these amounts could be PKR 6000, 8000 and 10000.

<sup>11</sup> We have observed this trend from our personal experience lasting over several years; we have no statistics to prove this claim.

<sup>12</sup> We didn't query the users regarding the other two interfaces.

<sup>13</sup> <http://www.tpsonline.com/>

## Acknowledgements

We acknowledge the assistance provided by Transaction Processing Systems in helping us to acquire the ATM transaction data sets, and also for providing the testbed for testing our interfaces.

## Author details

Ghulam Mujtaba Shaikh

*Sukkur Institute of Business Administration, Department of Computer Science, Airport Road, Sukkur, Pakistan*

Tariq Mahmood

*National University of Computer and Emerging Sciences, Department of Computer Science, Shah Lateef Town, National Highway, Karachi, Pakistan*

## 9. References

- [1] Aalst, W. [1998]. The application of petri nets to workflow management, *The Journal of Circuits, Systems and Computers*, Vol. 8, pp. 21–66.
- [2] Cook, J. E. & Wolf, A. L. [1998]. Discovering Models of Software Processes from Event-Based Data, *ACM Transactions on Software Engineering and Methodology* 7: 215–249. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.46.5456>
- [3] Fluxicon [2011]. Nitro flux capacitor, <http://fluxicon.com/blog/2010/09/nitro/>.
- [4] Han, J. & Kamber, M. [2006]. *Data Mining: Concepts and Techniques*, The Morgan Kaufmann Series in Data Management Systems, 2 edn, Morgan Kaufmann.
- [5] Hofstra, P. [2009]. *Analysing the effect of consumer knowledge on product usability using process mining techniques*, Master's thesis, Technische Universiteit Eindhoven.
- [6] Inc., E. B. S. [2011]. Atmta, <http://www.esq.com/products-a-solutions/category/4-atm-transaction-analyzer>.
- [7] Keane, J. [1997]. High performance banking, *Seventh International Workshop on Research Issues in Data Engineering*, 1997., pp. 66–69.
- [8] Khawaja, K. & Manarvi, I. [2009]. Evaluating customer perceptions towards atm services in financial institutions; a case study of pakistani banks, *International Conference on Computers and Industrial Engineering*, CIE 2009, pp. 1440–1445.
- [9] LevelFour [2011]. Is a new business and technology model required for atm channel.
- [10] Luca, A. D., Langheinrich, M. & Hussmann, H. [2010]. Towards understanding atm security: a field study of real world atm use, *SOUPS*.
- [11] Mans, R., Schonenberg, M., Song, M., van der Aalst, W. & Bakker, P. [2008]. Process mining in health care, *International Conference on Health Informatics (HEALTHINF'08)*, Funchal, Maldeira, Portugal, pp. 118–125.
- [12] Morgan, J. [2012a]. A conceptual model for personalising an automated teller machine, <http://usabilityetc.com/articles/atm-conceptual-model/>.
- [13] Morgan, J. [2012b]. Learning user preferences to personalise smart card applications, <http://usabilityetc.com/articles/learning-user-preferences/>.

- [14] Mujtaba, G. & Mahmood, T. [2011a]. Adaptive automated teller machines - part i, *Proceedings of the 4th International Conference on Information and Communication Technologies*, pp. 1–6.
- [15] Mujtaba, G. & Mahmood, T. [2011b]. Adaptive automated teller machines - part ii, *Proceedings of the 4th International Conference on Information and Communication Technologies*, pp. 6–12.
- [16] Nielsen, J., Molich, R., Snyder, C. & Farrell, S. [2001]. *E-Commerce User Experience*, Nielsen Norman Group.
- [17] Noonan, T. [2000]. Barriers to using automatic teller machines: A review of the useability of self-service banking facilities for australians with disabilities, *Australian Human Rights and Equal Opportunity Commission* pp. 61–101.
- [18] Perakh, A. V. [2010]. *ATM (Automated Teller Machine) Business Basics*, 2 edn, Cashflow ATM, Inc.
- [19] Ping, Z. L. & Liang, S. Q. [2010]. Data mining application in banking-customer relationship management, *Computer Application and System Modeling (ICCASM)*.
- [20] Prognosis [2011]. Atm/pos products, integrated research limited, [http://www.prognosis.com/products\\_and\\_solutions/atm/\\_pos\\_monitoring/products/page\\_1550.aspx](http://www.prognosis.com/products_and_solutions/atm/_pos_monitoring/products/page_1550.aspx).
- [21] Rigat, J. [2009]. *Data mining analysis of defect data in software development process*, Master's thesis, Technische Universiteit Eindhoven.
- [22] Tarakanov-Plax, A. [2004]. Use and non-use of automatic teller machines by older people in israel, *Gerontechnology* 3(2): 107–110.
- [23] van Dongen, B. F., de Medeiros, A. K. A., Verbeek, H. M. W., Weijters, A. J. M. M. & van der Aalst, W. M. P. [2005]. The prom framework: A new era in process mining tool support, *ICATPN*, pp. 444–454.
- [24] van Dongen, B. F. & van der Aalst, W. M. P. [2005]. A meta model for process mining data, *CAiSE WORKSHOPS*, pp. 309–320.