A Framework for Mining RFID Data From Schedule-Based Systems

Gürdal Ertek, Xu Chi, *Member, IEEE*, and Allan N. Zhang, *Member, IEEE*

***Abstract*—A schedule-based system is a system that operates on or contains within a schedule of events and breaks at particular time intervals. Entities within the system show presence or absence in these events by entering or exiting the locations of the events. Given radio frequency identification (RFID) data from a schedule-based system, what can we learn about the system (the events and entities) through data mining? Which data mining methods can be applied so that one can obtain rich actionable insights regarding the system and the domain? The research goal of this paper is to answer these posed research questions, through the development of a framework that systematically produces actionable insights for a given schedule-based system. We show that through integrating appropriate data mining methodologies as a unified framework, one can obtain many insights from even a very simple RFID dataset, which contains only very few fields. The developed framework is general, and is applicable to any schedule-based system, as long as it operates under certain basic assumptions. The types of insights are also general, and are formulated in this paper in the most abstract way. The applicability of the developed framework is illustrated through a case study, where real world data from a schedule-based system is analyzed using the introduced framework. Insights obtained include the profiling of entities and events, the interactions between entity and events, and the relations between events.**

***Index Terms*—Data mining, decision support systems, information systems.**

I. Introduction

THE topic of this paper is the mining of data collected through radio frequency identification (RFID) from schedule-based systems. A schedule is "a list of planned activities or things to be done, showing the times or dates when they are intended to happen or be done" (Cambridge Dictionary). A schedule-based system is a system that operates on (or contains within) a schedule of events and breaks at particular time intervals [1], [2]. Fig. 1 illustrates a schedule-based system, which is characterized by a set of entities (or resources) $\mathcal{I}$ entering and exiting a particular set of locations that have events $\mathcal{J}^1$ taking place in them according to a schedule.

Manuscript received July 24, 2014; revised December 9, 2014, October 24, 2015, and February 11, 2016; accepted February 21, 2016. Date of publication May 19, 2016; date of current version October 16, 2017. This paper was recommended by Associate Editor D. Akopian.

G. Ertek is with the Rochester Institute of Technology at Dubai, Dubai Silicon Oasis, 341055, UAE (e-mail: gurdalertek@gmail.com).

X. Chi and A. N. Zhang are with the Singapore Institute of Manufacturing Technology, Singapore 638075 (e-mail: cxu@simtech.a-star.edu.sg; nzhang@simtech.a-star.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2016.2557762

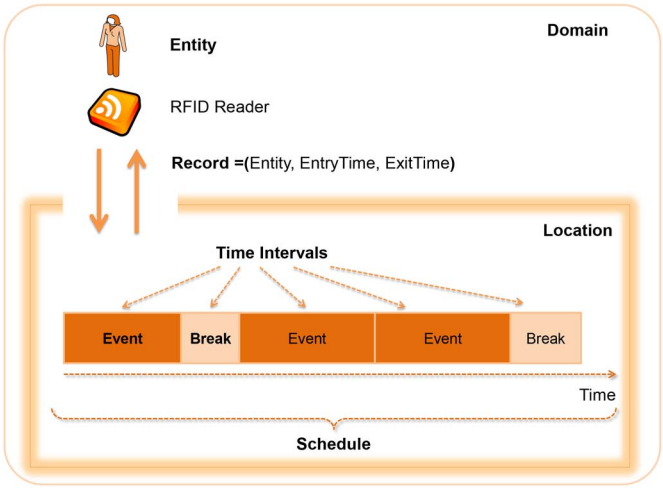

Fig. 1. Schedule-based system where entities entering and exiting the system are tracked with RFID.

An entity is a distinct, independent, or self-contained being. In this paper, an entity is considered as a unit that has an independent movement. So, for example, in the manufacturing context, the entities would be the batches of products being moved in the facility (rather than individual pieces of items). An event is something that occurs in a certain place/location during a particular interval of time. The events may take place successively or may be separated by breaks, in other words, time intervals of no events. The set of breaks is denoted by $\mathcal{J}^0$. Events and breaks constitute the set of time intervals $\mathcal{J}$. The schedule-based systems that we are particularly interested in are systems where the entry and exits of entities to location(s) are recorded through a data collection system, typically barcode, RFID, global positioning system (GPS), or sensors. Since RFID systems are gaining increasing importance in industry, we have illustrated a schedule-based system with RFID.

Schedule-based systems are extensively encountered in a variety of domains, ranging from manufacturing to social event management. However, the basic elements of the system are the same. The basic elements are shown in bold in Fig. 1. Table I lists some of the domains where schedule-based systems are present, and maps the key elements of a schedule-based system to domain-specific terminology.

An RFID system consists of tags (also known as transponders) and readers (also known as interrogators), typically also linked to an information system [3], [4]. In passive

RFID, the information on the chip of the tag is read by the reader through radio waves, and the tag cannot transmit radio waves by itself. In active RFID, the tag has its own internal power source and the capability of actively transmitting information to the reader. Passive tags have the advantage of being significantly cheaper, whereas active tags possess larger memory capacity and can be used in more sophisticated scenarios. Zhu *et al.* [3] provided an extensive review of RFID technology and its application in various industries, including logistics, retailing, travel and tourism, library science, food services and health care. A recent study reveals that only 3% of the companies in Europe have adopted RFID technology [3]. Thus, only a small percentage of companies have adopted RFID technology in their operations so far. However, the commitment of leading institutions (such as the U.S. Department of Defense) and companies (such as Walmart, JC Penney, and P&G) is expected to eventually spread the use of RFID, just as the barcode technology has gained acceptance over time. References [3], [5], and [6] provide a detailed discussion of RFID application domains, as well as a detailed literature review of RFID. Cinicioglu *et al.* [7] provided a highly useful list of potential benefits of RFID systems on operations management activities, in a multitude of domains. These benefits include preventing theft and shrinkage, identifying causes of spoilage, and evaluating employees.

RFID systems are used to basically produce data that can be mined through data mining methods for knowledge discovery and obtaining actionable insights. Data mining is the growing field of computer science where the goal is to uncover hidden information in-typically large and complex-piles of data [8]. There exist a multitude of data mining methods that can be applied depending on the size and structure of the data at hand. Data mining can thus be considered as a field which encompasses a collection of interrelated and interacting tools, including clustering, classification, association mining, network analysis, data visualization, as well as others. A significant challenge then is the selection of the appropriate set of methodologies and the way they are applied in analyzing a particular dataset.

The research questions to be answered in this paper are the following.

"Given RFID data from a schedule-based system in any domain (such as social event management, manufacturing, healthcare, etc.), what can we learn about the system (the events and entities) through data mining? Which data mining methods can be applied so that one can obtain rich actionable insights regarding the system and the domain?"

The research goal to answer the above research question is the development of a framework, that takes RFID data and basic event schedule data and information, and produces actionable insights regarding the system and entities within the system. Our first main motivation was to show that, through appropriate data analysis methodologies, one can obtain many insights from even a very simple RFID dataset, which contains only very few fields. Our second main motivation was that such a framework would be applicable in a wide range of domains. Our third motivation was observing from our survey of the literature that there is a significant gap regarding this type of research.

TABLE I
APPLICATION AREAS OF THE DEVELOPED METHODOLOGY, WITH A MAPPING OF THE VARIOUS ASPECTS OF THE MODEL

Domain	Entity	Location	Event	Break
Manufacturing	Worker, Cart	Workshop, Cell	Production, Setup	No production, Break
Warehousing	Worker, Pallet, Forklift	Warehouse, Zone	Order Pick Wave	No picking, Break
Transportation	Pallet	Truck, Warehouse	Transportation, (Un)Loading	Stalling, No operation
Social Event	Attendee, Organizer	Room, Hall	Session	Break
Healthcare	Doctor, Nurse, Patient, Equipment	Surgery Room, Hospital rooms	Surgery, Test Appointment	Break
Education	Instructor, Student	Classroom, Laboratory	Course hours Labs, Recitations	Break
Tourism	Tourist, Visitor	Museum, Art Gallery	Guided tour, Activity	No tour

The contributions of this paper are multifold: first, we introduce an analysis framework, including its mathematical representation, for mining RFID data coming from a schedule-based system. The framework developed is general, and is applicable to any schedule-based system that operates as described. While the framework is developed assuming a single location, it can also be extended to the case of multiple locations by introducing a set of locations $\mathcal{L}$ and a new dimension in the relevant sets and parameters. Second, we enumerate the different types of insights that can be obtained through the introduced framework. These insights are also general, and are formulated in the most abstract way possible. Third, we develop and present the corresponding algorithms that are needed in the analysis framework. The framework depends on these algorithms to do the required data processing, database augmentation, and other computations. Finally, we demonstrate the applicability of the developed framework through a case study, where real world data from a schedule-based system is analyzed using the introduced framework. The case study illustrates how the framework can be applied in the real world for a given domain.

The novelty of the research is the introduction of a data mining framework for the first time for this type of a system. The existing research in schedule-based systems mainly focuses on obtaining good, and if possible optimal, schedules, or event processing. However, the interaction of the entities in the system, given the obtained schedule, has not been analyzed in depth in earlier research. The importance of the research lies in its general applicability in a wide range of domains. Table I lists some of the application areas of the developed framework, with a mapping to the domain-specific terminology. Thus the developed framework is applicable in its current form in all the listed domains, because the fundamental aspects of the model are the same across domains.

The remainder of this paper is organized as follows. Section II provides a brief review of some relevant literature as the background. Section III discusses the framework developed and proposed. Section IV is devoted to the results and analysis of the case study, where new insights are obtained. Finally, Section VI presents some conclusive remarks.

II. Literature

A. Schedule-Based Systems

The primary line of existing research regarding schedule-based systems involves the derivation of good, and if possible optimal, schedules. The primary modeling approach for this line of research is optimization, and typically mixed-integer programming. Pinedo [9] is the classic reference for scheduling theory, and Pinedo [10] contains a detailed discussion of practice and application of scheduling, in addition to theory and algorithms. The scheduling research focuses on whether problems are polynomially solvable and optimal under certain conditions [11]. Typical contribution in such research also includes optimization or approximation algorithms and analysis of worst case error bound. Scheduling can be at any resolution, ranging from single-machine machine scheduling [11] to the scheduling of supply chains [12]. One line of scheduling research develops or applies machine learning and data mining methods and algorithms for generating the schedules [13]. Some of these studies also analyze generated schedules using data mining techniques for coming up with new schedules [14]–[16]. However, while very extensive research exists on scheduling, the interaction of the entities in the system, given the obtained schedule, has not been analyzed from a data mining perspective in earlier research. In our research, we provide the possible practical benefits of such a perspective in Section V. One final stream of research regarding schedule-based systems is regarding the processing of the events data [17].

B. Mining RFID Data

There exists a large body of literature on the mining of RFID data. However, an extensive survey performed during this paper revealed that none of the existing research studies have developed a comprehensive framework for mining RFID data coming from a schedule-based system. One approach could be modifying knowledge discovery and data mining (KDDM) process models [18], [19] for this particular domain.

The literature on mining RFID data is summarized in this section through the discussion of the following topics: Cleaning of RFID data, developing underlying data structures for efficient data mining, supply chain and logistics applications, retail applications, applications in other domains, outlier detection research, and finally mining RFID data from social events.

The most time consuming step in data mining is typically data cleaning. Ku *et al.* [20] developed a framework for RFID data cleaning. Baba *et al.* [21] presented a data cleaning methodology for indoor RFID data, eliminating temporal redundancy and spatial ambiguity, by building a distance-aware graph. The authors test and illustrate the methodology with real data from the baggage handling system of an airport.

The success of a data mining process is highly dependent on the underlying data structure. To this end, Han *et al.* [6] developed a data mining infrastructure that allows the efficient data mining of RFID data. Specifically, the authors introduce two new data models, namely path cube and workflow cube. They explain and illustrate their approach using examples and data from supply chain management.

Based on our literature review, the domains where one can find the mining of RFID data are supply chain management and logistics, as well as retail.

Poon *et al.* [22] presented a data processing and mining framework for logistics using RFID data. Ilic *et al.* [23] performed a rule-based analysis and GIS-based visualization of RFID data for managing items in a supply chain. For example, consistency of velocity and waiting time has to be ensured for an item throughout the supply chain, and any anomalies have to be detected. In a similar study, Shuping and Wright [24] applied 3-D visualization for tracking and understanding object movements through time, again enabling the discovery of irregularities.

The following three studies are examples of data mining for retail RFID data: Miyazaki *et al.* [25] developed a framework for the analysis of residence time in shopping, based on the mining of RFID data. Cinicioglu *et al.* [7] and Fang *et al.* [26] used RFID data for targeted advertising inside a retail store. Sakurai *et al.* [27] used RFID data for predicting retail store sales.

Studies on the mining of RFID data for other domains include the following: Lyu *et al.* [28] presented a framework for quality assurance, as well as two industry applications. Lee *et al.* [29] mined RFID data through the integration of fuzzy logic for resource allocation in garment manufacturing, and illustrates the applicability of this approach at a company. Wen [30] presented a framework that uses RFID data for intelligent traffic management. Tsai *et al.* [31] performed sequential pattern mining of RFID data for generating tourist path suggestions. Meiller *et al.* [32] presented a knowledge-based system framework for healthcare using RFID data.

A multitude of studies investigate the outlier detection problem with RFID data. Hsu *et al.* [33] carried out behavior modeling using RFID data, using clustering to detect abnormal events. Delgado *et al.* [34] used RFID data for behavior identification and anomaly detection. Masciari [35] presented an data mining framework that detects outlier observations in RFID data.

Other related papers do not necessarily use data collected through RFID, but illustrate methods and case studies that can be adopted to the analysis of RFID based data. For example, Gao and Liu [36] presented a very detailed analysis of data on location-based social networks. Some of the research questions investigated in [36] include how social connection is affected by geographical distance, how users can be clustered based on their activities, how user mobility is influenced by various factors, and how home locations of users can be predicted. Chen *et al.* [37] mined matching behavioral patterns based on joining various kinds of entity characteristics in mobile communication. One final related line of research builds social recommender systems with various benefits, such as supporting the creation of new social relations [38].

RFID technology has a great potential for facilitating and enhancing the management of social events, where humans interact with each other over time and across different

locations. The case study in our paper presents the application of RFID in the context of a social event, specifically a scientific conference. References [39]–[42] provide information system architectures for collecting data in a conference through RFID. Bravo *et al.* [42] also described how this data can be used in real time for informing conference attendees and illustrates, through a detailed scenario, how the system operates.

RFID systems, when used in social events, generate time-stamped location data for each of the attendees/participants of the event. This data, when combined with other data regarding the attributes of the attendees, locations and the event schedule, can generate significant insights regarding the attendees, the structure and the nature of the social network, and the event. Furthermore, the methods employed for mining social network data [43], [44] can be fused to obtain hybrid data analysis frameworks. These insights and the information systems designed around them can be used to improve the social event in better serving its intended goals [41]. Improved conference management information systems and managerial practices can enable the attendees find sessions and other people that they would be interested in, minimize schedule conflicts, increase participation in the sessions, and improve the overall quality of the event.

References [39], [40], and [45] are the most related studies in the literature to our case study, because these papers carry out posterior visualization and analysis of RFID enriched event data, and furthermore give examples of insight-generating questions whose answers can be obtained through querying the data.

Szomszor *et al.* [39] developed an infrastructure and a scalable information system for tracking and analyzing human face-to-face (f2f) contact networks, such as people in a scientific conference. The authors employ RFID technology and data reporting and analysis methods for enhancing social interactions between event attendees and industry exhibitors.

Chin *et al.* [40] developed an RFID based system for connecting conference attendees based on their locations, the sessions that they have attended, and the attendees they have interacted with. Posterior analysis of attendee behavior suggested that earlier physical encounter during the conference (proximity), as well as commonality of attributes (homophily) were the most important factors affecting the selection of new contacts.

Atzmueller *et al.* [45] also presented an information system for conference management and detailed analysis of the obtained f2f data, as in [39] and [40]. The main contribution of [45] is the comprehensive evaluation of the behavioral patterns in a conference setting, developing analysis techniques for revealing roles of the attendees and attendee communities. Explicit and organizing roles are discovered through the analysis of classic centrality measures used in graph theory, such as degree, strength, betweenness, closeness, and eigenvalue centrality.

While [39], [40], and [45] bring fresh perspectives to the mining of RFID data, this paper has several additional aspects in comparison to these studies: First, we develop a complete framework that exhaustively explores and exhibits all the possible types of insights, rather than a set of selected few insights. Second, our framework requires a very basic data, with very few attributes, collected by almost every RFID system by default. Third, our framework is described not only conceptually, but also through rigorous mathematical formalism. The algorithms used for data processing are also included in the work. Fourth, rather than discussing a single domain, we generalize the analysis to schedule-based systems, which can include a very rich collection of application domains. Fifth, we discuss the practical implications of our research for not only a single domain (ex: social event management), but for a multitude of domains.

III. Framework

In this section, we describe the framework that we introduce for mining RFID data from schedule-based systems. First we outline the research steps followed in this paper. Then we list our assumptions regarding the analyzed system. Third, we introduce the mathematical notation and the database structures in the various stages of the analysis framework. Fourth, we describe the computational algorithms for augmenting the RFID data obtained from a schedule-based system. Finally, we present the novel analysis framework that we have developed, and list the types of insights that can be obtained through this framework.

A. Research Steps

This paper consists of the steps listed below, and resulted in the framework and case study presented in this paper. We thus suggest the application of similar steps in analyzing the RFID data coming from a system with particular characteristics.

1) Understanding of the data mining research goal, as well as the research question and the domain.
2) Development of a mathematical notation (example: sets, parameters, ...).
3) Description of the RFID data and the domain-related data in terms of the developed mathematical notation (example: entities, entity entrance times to events).
4) Identification of the metrics to be computed (example: whether an attendee has attended a particular session or not, as well as the time s/he spent in each session), and the database structures needed.
5) Development of formulas for obtaining the desired performance metrics and insights.
6) Identification some of the possible types of data analysis that can be implemented, as well as some of the possible types of insights that can be obtained through each type of data analysis.
7) Execution of the data analysis process, and the discovery of various types of insights.
8) Elicitation of the obtained results and insights, and the subsequent filtering of the most essential and actionable insights among those obtained. The importance and actionability of insights were decided upon through discussion sessions with conference organizers from academia.

9) Integration of the executed data mining processes in a single unified framework, and proposing it as a general methodology for the analysis of RFID data from schedule-based systems, that can be applied to systems other than schedule-based ones.

B. Assumptions

Our assumptions regarding the RFID data collection are as follows.

1) The gateway where the RFID reader is located is an in-out-gateway [46].
2) RFID tags are read throughout the event schedule, not missing any of the events, nor people passing through the doors.
3) All passes (entries and exits) made with an RFID tag are read, with the RFID receiver not missing any passes.
4) RFID readings and the final data are accurate.
5) Every entity wears RFID during passes, except when the RFID tag is left in the location, never to be worn again. Our framework currently assumes and considers a single location for all the events. The framework can be extended to accommodate multilocation settings, which would enable the discovery of many new types of insights.
6) All events happen in one location.

These assumptions (except the last) are required so that the data is accurate and complete. The last assumption is assumed so that the concepts and the developed framework can be easily demonstrated.

C. Mathematical Notation

We now introduce the mathematical notation that will be used throughout the description of the framework. While the indices are always provided in the notation, for convenience, sometimes the indices are dropped (for example, u). In that case, the symbol refers to the symbol with the default indices that were specified when the notation was initially introduced (for example, u refers to u_{ir}, because that is how it is defined initially). The database structures and algorithms will also be introduced in this section.

1) Sets:

$\mathcal{R}$ Set of unique record IDs; $r : 1 \cdots R$.
$\mathcal{I}$ Set of entities; $i : 1 \cdots I$.
$\mathcal{J}$ Set of time intervals; $j : 0, 1 \cdots J$ (The time intervals correspond to actual events and the breaks between these events); $\mathcal{J} = .\mathcal{J}^0 \cup \mathcal{J}^1$.
$\mathcal{J}^0$ Set of breaks.
$\mathcal{J}^1$ Set of events.

2) Given Data:

u_{ir} Entry time of entity i in record r.
U_{ir} Exit time of entity i in record r.
$\mathcal{D}^0$ The database of RFID logs; $\mathcal{D}^0 = \{d^0\langle r, i, u_{ir}, U_{ir}\rangle\}$.

3) Event Schedule Data:

s_j Start time of time interval j.
f_j Finish time of time interval j.
d_j Duration of time interval j; $d_j = f_j - s_j$.
$\mathcal{D}'$ The database of time intervals; $\mathcal{D}' = \{d' : \langle j, \text{intervalType}(j), s_j, f_j, d_j\rangle\}$.

where intervalType(j) is a lookup function (defined next) that returns whether the time interval corresponds to an event or a break.

4) Lookup Functions:

$$\text{intervalType}(j) = \begin{cases} \text{break}, & \text{if } j \in \mathcal{J}^0 \\ \text{event}, & \text{if } j \in \mathcal{J}^1 \\ \text{null}, & \text{otherwise} \end{cases}$$

$$\text{intervalOf}(t) = \left\{j \in \mathcal{J} : s_j \le t \le f_j\right\}$$

where t is the time of occurrence of an event.

5) Intermediary Data:

$u = u_{ir}$ Entry time of an entity in a record.
$U = U_{ir}$ Exit time of an entity in a record.
$e = e_{irj}$ Entry time of an entity to an event in a record.
$x = x_{irj}$ Exit time of an entity from an event in a record.
$T = T_{irj} = x - e$ Time spent by entity at an event (in a single record).
$p = p_{irj}$ Start time of an entity present at the location for an event in a record (the entity may wait for the event).
$q = q_{irj}$ End time of an entity present at the location for an event in a record (the entity may be spending additional time at the location after the event is completed).

6) Computed Metrics:

$\underline{p}_{ij}$ Earliest start time of an entity present at the location for an event; $\underline{p}_{ij} = \min_{r\in\mathcal{R}} p_{irj}$.
$\overline{q}_{ij}$ Latest end time of an entity present at the location for an event; $\overline{q}_{ij} = \max_{r\in\mathcal{R}} q_{irj}$.
$\alpha' = \alpha_{irj}$ Earliness; how early entity i entered event j in a given record r; takes positive value if entity entered early, and takes negative value if entity entered late; $\alpha_{irj} = s_j - p_{irj}$.
$\beta' = \beta_{irj}$ Lateness; how late entity i exited from event j in a given record r; takes positive value if entity exited late, and takes negative value if entity exited early; $\beta_{irj} = q_{irj} - f_j$.
$\alpha = \alpha_{ij}$ Earliness; how early entity i entered event j; takes positive value if entity entered early, and takes negative value if entity entered late; $\alpha_{ij} = \max_{r\in\mathcal{R}} \alpha_{irj}$.
$\beta = \beta_{ij}$ Lateness; how late entity i exited event j; takes positive value if entity exited late, and takes negative value if entity exited early; $\beta_{ij} = \max_{r\in\mathcal{R}} \beta_{irj}$.

$$\text{entryStatus} = \begin{cases} \text{NoEntry}, & \text{if } \alpha = \text{null and not an entry from previous event} \\ \text{EarlyEntry}, & \text{if } \alpha \ge 0 \text{ and not an entry from previous event} \\ \text{LateEntry}, & \text{if } \alpha < 0 \text{ and not an entry from previous event} \\ \text{EntryFromPreviousEvent}, & \text{if entry from previous event} \end{cases}$$

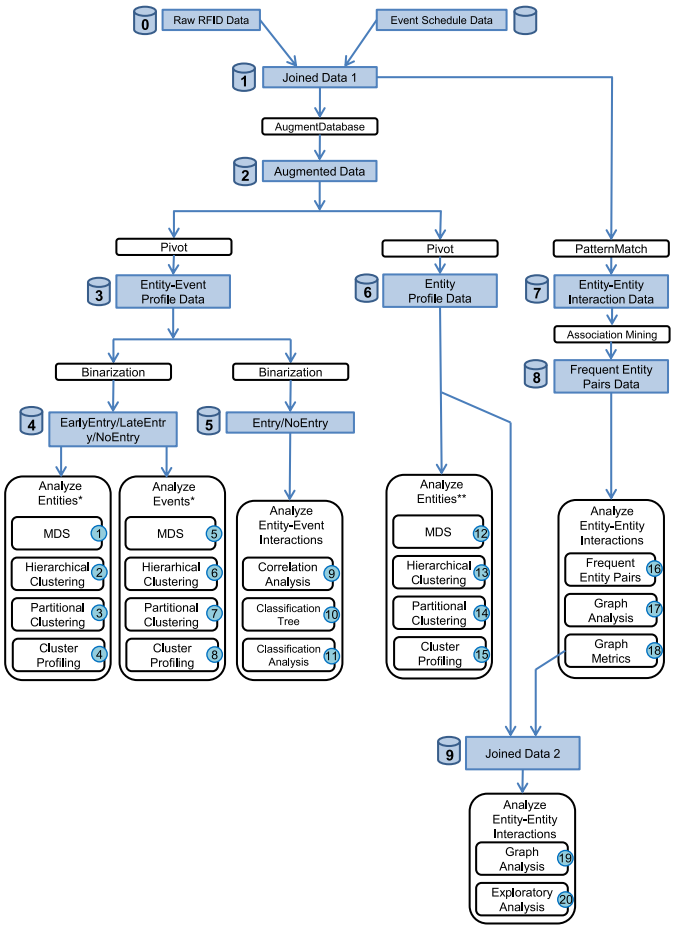

Fig. 2. Developed analysis framework for mining RFID data from a schedule-based system.

$$\text{exitStatus} = \begin{cases} \text{NoExit}, & \text{if } \beta = \text{null} \text{ and not an exit into next event} \\ \text{EarlyExit}, & \text{if } \beta \geq 0 \text{ and not an exit into next event} \\ \text{LateExit}, & \text{if } \beta < 0 \text{ and not an exit into next event} \\ \text{ExitIntoNextEvent}, & \text{if exit into next event} \end{cases}$$

$$Z_{ij} = \begin{cases} 1, & \text{if attendee } i \text{ attended event } j,\ i \in \mathcal{I},\ j \in \mathcal{J} \\ 0, & \text{otherwise} \end{cases}$$

n_{ij} Number of times that entity i has entered and/or exited event j.

T_{ij} Total time (stay duration) that entity i spent in event j; $T_{ij} = \sum_{r \in \mathcal{R}} T_{irj}$.

7) Databases: The databases whose structures are given here are shown as cylinders in Fig. 2. For example, cylinder with the label 0 refers to $\mathcal{D}^0$ and the cylinder with the label 1 refers to $\mathcal{D}^1$.

The database structures for the raw RFID database and the joined database 1 are

$$\mathcal{D}^0 = \left\{ d^0 : \langle r, i, u_{ir}, U_{ir} \rangle \right\}$$

$$\mathcal{D}^1 = \left\{ d^1 : \langle r, i, u_{ir}, U_{ir}, j_1, j_2, \varepsilon_1, \varepsilon_2, s_{j_1}, s_{j_2}, f_{j_1}, f_{j_2} \rangle \right\}$$

where $\varepsilon_1 = \text{intervalType}(j_1)$ and $\varepsilon_2 = \text{intervalType}(j_2)$. In the data we used, two successive actions (first being an entry and second being an exit) of an entity were given in the same line.

The augmented database is

$$\mathcal{D}^2 = \Big\{ d^2 : \langle i, j, r, u, U, e, x, t, p, q, \alpha', \beta', \text{entryStatus}, \text{exitStatus} \rangle, j \in \mathcal{J}^1 \Big\}.$$

Entity-event profile database and the databases derived from that database are

$$\mathcal{D}^3 = \left\{ d^3 : \langle i, j, \alpha \rangle, j \in \mathcal{J}^1 \right\}$$

$$\mathcal{D}^4 = \left\{ d^4 : \langle i, j, \Gamma_1(\alpha) \rangle, j \in \mathcal{J}^1 \right\}$$

where

$$\Gamma_1(\alpha) = \begin{cases} \text{NoEntry}, & \text{if } \alpha = \text{null} \\ \text{EarlyEntry}, & \text{if } \alpha \geq 0 \\ \text{LateEntry}, & \text{if } \alpha < 0 \end{cases}$$

$$\mathcal{D}^5 = \left\{ d^5 : \langle i, j, \Gamma_2(\alpha) \rangle, j \in \mathcal{J}^1 \right\}$$

where

$$\Gamma_2(\alpha) = \begin{cases} \text{NoEntry}, & \text{if } \alpha = \text{null} \\ \text{Entry}, & \text{otherwise.} \end{cases}$$

Entity profile database is

$$\mathcal{D}^6 = \Big\{ d^6 : \langle i, \text{avg}(t), \text{avg}_j(\alpha), \min_j(\alpha), \max_j(\alpha), \text{stdev}_j(\alpha), \text{avg}_j(\beta), \min_j(\beta), \max_j(\beta), \text{stdev}_j(\beta), \text{count}_j(i) \rangle, j \in \mathcal{J}^1 \Big\}$$

where the computations for $\mathcal{D}^6$ are done using $\mathcal{D}^2$. In obtaining the entity profile data in $\mathcal{D}^6$, the function $\text{avg}_j(x)$ and other statistical functions are used. These functions calculate, for a given entity i, the average (or other statistics) values of a metric x over all the events.

The remaining databases are

$$\mathcal{D}^7 = \left\{ d^7 : \langle i_1, i_2 \rangle, i_1, i_2 \in \mathcal{I} \right\}$$

$$\mathcal{D}^8 = \left\{ d^8 : \langle i_1, i_2, \text{count}(i_1, i_2) \rangle, i_1, i_2 \in \mathcal{I} \right\}$$

$$\mathcal{D}^9 = \left\{ d^6 \cup \text{metrics}(i), i \in \mathcal{I}', d^6 \in \mathcal{D}^6 \right\}$$

where $\mathcal{I}'$ is the set of entities in $\mathcal{D}^8$ which have a support count greater than the minimum support count threshold, and metrics(i) is a function that returns the array of computed graph metrics for an item i.

D. Computational Algorithms for Data Augmentation

The computational algorithms for augmenting the RFID data are given in Appendix A. The first of these algorithms takes the raw RFID data $\mathcal{D}^0$ and the schedule data, and joins these two tables to form a new data table, namely $\mathcal{D}^1$. The second algorithm is more complicated, and is focused only in what is happening with respect to events (rather than breaks). This second algorithm transforms the data which is in the form of entry/exit records into a database $\mathcal{D}^2$ that contains information only on entities and events. The augmented database $\mathcal{D}^2$

TABLE II
INSIGHTS THAT CAN BE OBTAINED THROUGH THE INTRODUCED ANALYSIS FRAMEWORK

Insight No	Question Answered	Example in
	Behavior Analytics 1 Based on the behavioral patterns of the entities *with respect to specific events*...	
1	· Which entities are positioned close to each other?	Figure 3
2	· Which groups of entities behave most similar?	
3	· Which entities can be clustered into which clusters?	
4	· What are the profiles of these entity clusters?	
	Event Analytics Based on the behavioral patterns of the entities *with respect to specific events*...	
5	· Which events are similar to each other?	
6	· Which groups of events are most similar?	Figure 4
7	· Which events can be clustered into which clusters?	
8	· What are the profiles of these event clusters?	
9	· What is the correlation between different events?	Table III
10	· Which earlier events affect a particular event, and how?	Figure 5
11	· Can the entry of specific entities to an event be predicted?	Table IV
	Behavior Analytics 2 Based on the *general behavioral patterns* of the entities...	
12	· Which entities are related to each other?	Figure 6
13	· Which groups of entities behave most similar?	Figure 7
14	· Which entities can be clustered into which clusters?	
15	· What are the profiles of these entity clusters?	Figure 8
	Relationship Network Analysis Based on the joint actions of the entities...	
16	· Which entity pairs enter/exit many events together?	Table V
17	· How are the entities related to each other?	Figure 9
18	· Which entities are *social* and which are *not social*?	Table VI
19	· How can the behavioral attributes be analyzed together with the relationship network?	Figure 10
20	· How can the behavioral attributes be analyzed together with the graph metrics?	

contains the entry and exit times of entities to events, as well as their earliness α (positive value if early entry), lateness β (positive value if late exit), as well as other data. $\mathcal{D}^2$ is critical, because it is used in later stages of the analysis framework to extract new databases and to obtain insights.

E. Analysis Framework

The developed analysis framework is given as a flowchart in Fig. 2. The framework starts with raw data coming from RFID system, as well as data regarding the schedule of events in the system. The data are then brought to a richness so that it can be analyzed to obtain insights. The analysis centers around three lines; shown with the numbers 3, 6, and 7 in the figure. The insights are obtained through analyzing entities, events, entity-event interactions, and entity-entity interactions.

Fig. 2 shows that the analysis begins by joining the raw RFID data $\mathcal{D}^0$ (shown with the cylinder with the label 0) with the event schedule data to form $\mathcal{D}^1$, and then augmenting $\mathcal{D}^1$ to generate $\mathcal{D}^2$. Next, three basic types of data are obtained as follows.

$\mathcal{D}^3$, entity-event profile data is obtained through pivoting on $\mathcal{D}^2$, and shows the earliness of each entity for each event. Some of the values are missing, indicating that the entity did not enter the system at all during a particular event.

$\mathcal{D}^6$, entity profile data shows the metric statistics for each entity as computed over all the sessions.

$\mathcal{D}^7$, entity-entity interaction data lists the entity pairs that have entered or exited the system simultaneously. The data is obtained through running a pattern matching algorithm (Appendix B).

Having obtained these three basic types of data, further data transformations and/or algorithms are applied to obtain insights into the system. These insight types are numbered from 1 to 20 in Fig. 2, inside the circles. These 20 insight types are then listed in Table II. In Table II, the insights that can be obtained using our proposed framework have been classified into categories of behavior analytics, event analytics, and relationship network analysis. The behavior analytics category has been further labeled as 1 or 2 based on the analytics quest and the data source.

Table II also lists (in its last column) the figure and/or tables which illustrate the insight type in the case study. For example, consider the line corresponding to Insight 2 in Table II. This insight aims at answering the question "Which entities are positioned close to each other?" The same line in the table tells that an example of the analysis that leads to this type of insight is illustrated in Fig. 3. Due to space limitations in this paper, we are able to provide examples for only some of the insights, hence the empty cells under the last column of the table.

IV. CASE STUDY

In this section, we first describe the data used in the case study, and then illustrate the various insights that can be obtained through the presented framework. The insights that will be presented are listed in Table II, along with the figure

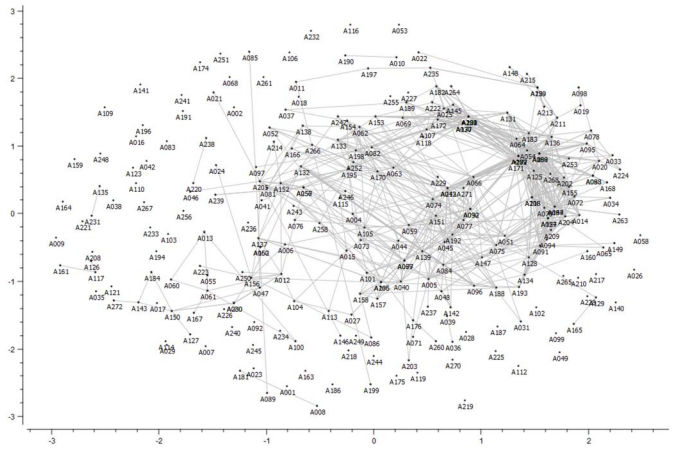

Fig. 3. MDS results for entity analysis based on entity-event profile data.

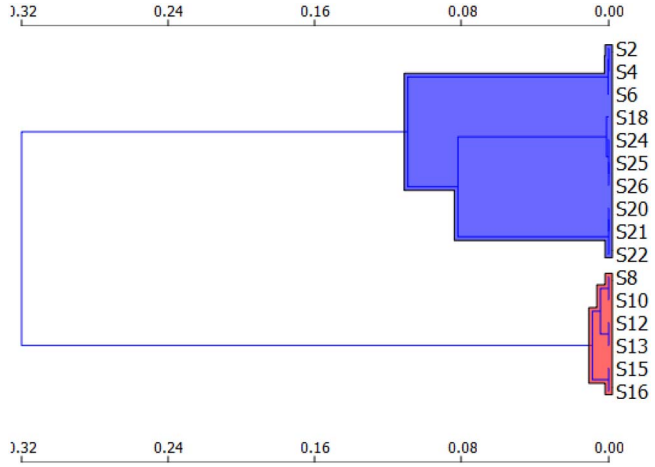

Fig. 4. Attribute (session) clustering results for case 2.

and table numbers. The data mining processes applied are described in Appendix C, and the software tools used are described in Appendix D.

A. Data

The data used in the study belongs to the domain of social event management, and comes from a four-day medical conference. Each attendee of the conference was provided with a unique RFID tag and their entry and exit times to the single conference hall were recorded. The raw RFID data $\mathcal{D}^0$ consists of 9624 rows (entry-exit combinations), and four columns, where the columns are the record ID, attendee name (masked), entry date and time, and exit date and time. The total number of attendees (total number of entities in the schedule-based system) is 272. The schedule consists of 17 events and 11 breaks (including the time interval before the first event and the time interval after the last event) giving a total of 28 time intervals.

B. Behavior Analytics 1: Based on Entity-Event Data

The first illustration is for Insight 1, and is given in Fig. 3. This insight answers the question "Which entities are positioned close to each other?" The analysis here is based on temporal proximity [47] of entities. The data mining method used for this purpose is multidimensional scaling (MDS) (as read from the box that corresponds to Insight 1 circle in Fig. 2), which maps multidimensional data onto two dimensions, based on how close the data points are [48]. Fig. 3 shows the mapping of attendees (entities) on a 2-D plane. The most significant associations are shown with lines between the points. Since Insight 1 is using the database $\mathcal{D}^4$ (EarlyEntry/LateEntry/NoEntry data), the closeness of the points, as well as the links between them, are based on the Hamming distance in between. Hamming distance is a distance measure that computes the number of bits two strings are different from each other [49]. As an example, if two entities entered all events early, but differed only in their behavior with respect to one event (for example one entered early, and the other entered late into the last event), the Hamming distance between them would be 1. The Hamming distance measure has been selected, rather than other distance measures, because it is a very popular distance measure when the data points are binary vectors. One can observe a highly dense region in Fig. 3, to the right of the figure, as well as a less dense region in the middle of the figure, and some sparse links. This shows that the entities in the dense cluster are very much related to each other, whereas there are other related entities among the remaining entities. Furthermore, given a particular entity, one can find the other entities closely positioned to this entity from the figure.

C. Event Analytics Based on Entity-Event Data

The next illustration is for Insight 6, and is given in Fig. 4. This insight answers the question "Which groups of events are most similar?" The data mining method used for this purpose (as read from Fig. 2) is hierarchical clustering, which hierarchically builds clusters from data, starting from individual points ([50], p. 44). The length of each horizontal U-shape represents the distance between the two data points connected. An interesting point here is that the data points are not the entities, but rather the events (sessions in the case study). So the goal is to see which events are similar to each other, based on the entry-exit patterns of the entities. This analysis also uses database $\mathcal{D}^4$ (EarlyEntry/LateEntry/NoEntry data), however, carries hierarchical clustering of events (rather than the entities), based on the Hamming distances between the events. The dendrogram in Fig. 4 shows that events {S2, S4, S6} are similar to each other, based on the entity-event profile data. Similarly, {S18, S24, S25, S26} are very similar, since they form a cluster together. Other groups of similar events are {S20, S21, S22}, {S8, S10}, {S12, S13}, and {S15, S16}.

The next illustration is for Insight 9, and is given in Table III. This insight answers the question "What is the correlation between different events?" The data mining / statistics method used for this purpose is correlation analysis, which computes the linear association between pairs of observations [51]. Again, the observations are events (rather than entities). Table III shows the correlation values above 0.50 and below −0.50. High correlation between successive events

TABLE III
HIGHEST AND LOWEST CORRELATIONS BETWEEN SESSIONS

Session1	Session2	Are Successive?	Correlation
S25	S26	Yes	0.88
S20	S21	Yes	0.68
S2	S4	Yes	0.59
S25	S27		0.53
S26	S27	Yes	0.60
S21	S22	Yes	0.53
S24	S27		-0.50
S8	S10	Yes	-0.55
S15	S16	Yes	-0.66
S24	S26		-0.72
S24	S25	Yes	-0.77

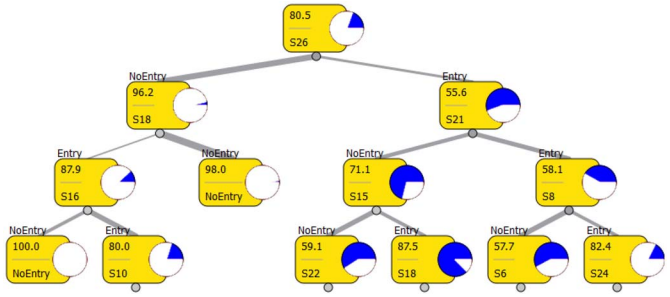

Fig. 5. Classification tree for the case where class attribute is entry into session S27.

indicates that the entities which entered the former of those successive events also mostly entered the latter, or those who did not enter the former did not enter the latter. This is the case for session pairs (S25, S26), (S20, S21), (S2, S4), (S26, S27), and (S21, S22). This high correlation can indicate that the former event encouraged entry to the latter, that the two events catered to the same set of entities, or both of these. Negative correlation between two successive events is also an important observation, and may be due to one or combination of several reasons: First, it may be that the former event was (not) successful, en(dis)couraging entry to the latter. Second, the two events may be catering to different set of entities. Third, there may be another reason, such as the latter event being the last event of the day, and entities exiting the system early. The successive event pairs with high negative correlation are (S8, S10), (S15, S16), and (S24, S25).

The following illustration is for Insight 10, and is given in Fig. 5. This insight answers the question "Which earlier events affect a particular event, and how?" The data mining method used for this purpose is classification tree analysis [52]. Classification trees summarize rule-based information about classification as trees. In classification tree models, each node is split (branched) according to a criterion. Then, a tree is constructed with a depth until all the rules are displayed on the graph under a stopping criterion. The percentages of the slices represent the percentages of data that have the different class labels. At each level, the attribute that creates the most increase in percentage compared with the previous level is observed. The algorithms for classification tree analysis are explained in [52]. In the implementation that we utilized in our analysis, selecting the attributes for the splits is based on information gain. In classification trees, identifying the nodes that

TABLE IV
CLASSIFICATION RESULTS FOR PREDICTING ATTENDEES TO SESSION S27 (LAST EVENT IN THE SCHEDULE)

Classifier	**CA**	**AUC**
CN2 rules	0.8347	0.7925
kNN	0.8186	0.7763
Classification Tree	0.8327	0.7267
SVM	0.8274	0.8386
Naive Bayes	0.8243	0.8573
Neural Network	0.8315	0.8432

differ noticeably from the root node are important, because the path that leads to those nodes tells us how significant changes are observed in the sub-sample compared with the complete data. By observing the shares of slices and comparing with the parent and root nodes, one can discover interesting classification rules and insights. Fig. 5 shows the classification tree where the Entry/NoEntry into event S27 (last session in the case study) is the predicted attribute. The very first split, based on the value of S26 provides the most information. In the complete data, 80.5% of the entities did not participate in event S27 (white-colored slice, denoting NoEntry). However, among those entities that did not enter S26 at all (NoEntry), this percentage is 96.2%. On the other hand, among the entities that did enter S26, the percentage of Entry into S27 is higher (55.6%). So, approximately half (55.6%) of those entities that entered S26 entered S27, whereas almost all (96.2%) of the entities that skipped S26 also skipped S27. This connectedness between S26 and S27 could also be hypothesized based on Table III, which shows a correlation of 0.60 between these sessions. However, the classification tree analysis provides us with specific percentages of Entry/NoEntry for S27 based on the values of S26.

The following illustration is for Insight 11, and is given in Table IV. This insight answers the question "Can the entry of specific entities to an event be predicted?". The data mining method used for this purpose is classification analysis. In classification analysis, the dataset is divided into two groups, namely, learning dataset and test dataset. Classification algorithms, also called classifiers (or learners), use the learning dataset to learn from data and predict the class attributes in the test dataset ([53], p. 17). The prediction success of each learner is measured through classification accuracy (CA) [54], the percentage of correct predictions among all, as well as receiver operating characteristic (ROC) curves [55]. Classifiers which result in higher CA and a greater area under the ROC curve (AUC) correspond to better predictive models. The following classification algorithms are among the best-known classifiers in the machine learning field, and have been used in our analysis: CN2, *k*-nearest neighbor, classification tree, support vector machines, naive Bayes, and neural networks [53]. We used a subset of the population attendance records to estimate the attendance of the rest of the population for the same event. The data of the known half of entities includes past behavior. First, the entries of entities into event S27 are predicted with a very small learning dataset of 50% (around 130 observations), with 100 experimental repeats (using percentage split of the full dataset into learning and testing datasets). The CA and AUC values are displayed in Table IV, showing that

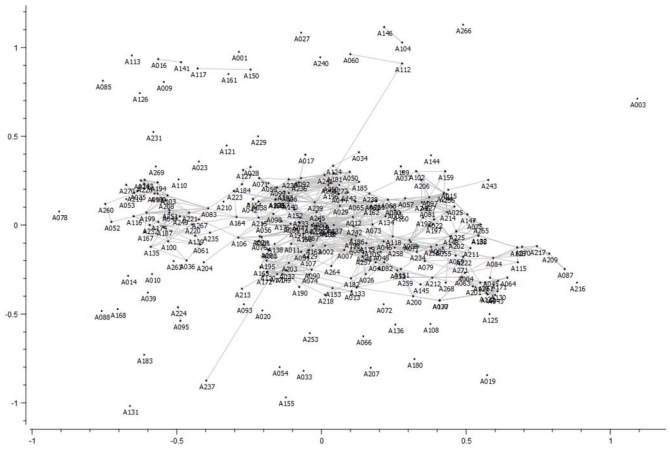

Fig. 6. MDS results for entity analysis based on entity profile data.

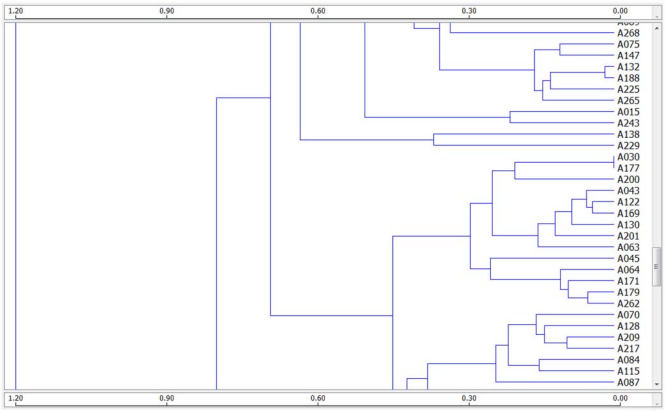

Fig. 7. Hierarchical clustering results for entity analysis based on entity profile data.

Cluster	Avg_StayDuration	Avg_Earliness	Avg_Lateness	NumberOfEntities
C1	56.31	0.24	-1.45	45
C8	45.97	-15.45	-1.32	23
C4	44.68	-5.65	-5.64	34
C10	43.65	-2.72	28.71	9
C5	41.96	-3.92	-13.05	29
C6	36.76	-12.99	-0.12	12
C3	36.06	-11.68	-15.76	41
C7	32.75	-20.04	-10.70	41
C9	32.18	-6.20	-21.80	17
C2	29.94	-29.59	-5.36	16

Fig. 8. Cluster profiles for entity analysis based on entity profile data.

if the behavior of half of the entities for S27 are known, the remaining entry or no entries can be predicted with a very high accuracy, up to 83.15%, with neural network classifier. Besides the black box neural networks technique, which does not tell the reasoning behind classification, CN2 and classification tree might be considered, since they provide the classification rules openly.

D. Behavior Analytics 2: Based on Entity Profile Data

The next illustration is for Insight 12, and is given in Fig. 6. This insight answers the question "Which entities are related to each other?" While the question answered is the same as that of Insight 1, the way it is answered is different. In Insight 1, the answer was computed based on entity-event data, whereas this time it is computed based on entity profile data. The data mining method used for this purpose is again MDS. Fig. 6 shows the mapping of attendees (entities) on a 2-D plane. The most significant associations are shown with lines between the points. Insight 12 is using the database $\mathcal{D}^6$, which contains only numerical values. Hence, the closeness of the points, as well as the links between them, are based on the Euclidean distance in between. One can observe a highly dense region in Fig. 6, to the middle of the figure, as well as a less dense region to the left of the figure, and some sparse links. This means that the entities in the dense cluster are very much close to each other, whereas there are other closely positioned entities among the remaining entities. The results of Insight 12 are different than that of Insight 1, since both the values in the database and the distance measure used are different. This illustrates that one should use the appropriate dataset (and the associated distance measure) that is aligned with the goals of the analysis.

The next illustration is for Insight 13, and is given in Fig. 7. This insight answers the question "Which groups of entities behave most similar?" The data mining method used for this purpose is hierarchical clustering, just as in Insight 6. The data points this time are entities, rather than events. So the goal is to see which entities are similar to each other, based on their overall behavior patterns, particularly their entry and exit timings. This analysis uses database $\mathcal{D}^6$, and carries hierarchical clustering of entities based on the Euclidean distance between them. The dendrogram in Fig. 7 shows that entity groups {A132, A188}, {A030, A177}, {A043, A122, A169}, {A179, A262} are similar to each other, based on the entity profile data.

When partitional clustering is carried out, the entities are partitioned into distinct clusters. One of the analysis to be done given these clusters is to profile the clusters using exploratory data visualization. This cluster profiling constitutes Insight 15, and an illustration of this insight is given in Fig. 8. This insight answers the question "What are the profiles of these entity clusters?" Here, the clusters are again based on numerical data coming from the database $\mathcal{D}^6$. Fig. 8 profiles the clusters based on three attributes, namely average stay duration, average earliness, and average lateness, and also provides the number of entities in each cluster. The 45 entities in the first cluster C1 have the highest average stay duration (56.31 min), and have the enter and exit events (sessions in the case study) almost with a perfect timing, neither early nor late. The 23 entities in the next cluster, C8, also stay in the events for a long time (average of 45.97 min), and exit with almost no earliness or lateness, but arrive an average of 15.45 min late. Each cluster has a profile, that can be similarly read from the figure. For example, at the other extreme, the last cluster, C2, consists of 16 entities who stay the least in the events (average of 29.94 min) and enter the events very late (average of 29.59 min). A

TABLE V
LIST OF SELECTED ENTITY PAIRS AND THEIR SUPPORT COUNT (ABSOLUTE SUPPORT) VALUES

Attendee1	Attendee2	Support Count
A150	A161	39
A164	A150	31
A009	A150	29
...	...	...
A055	A230	6
A249	A230	6
A249	A126	6

particularly interesting cluster is C10. The nine entities in cluster C110 stay for a long time in the events, and arrive almost on time, but they stay for a long time (average of 28.71 min) after the event is over. In the case study, this can be referring to the after-session discussions participated by these entities.

E. Relationship Network Analysis Based on Entity-Entity Interaction Data

The next illustration is for Insight 16, and is given in Table V. This insight answers the question "Which entity pairs enter/exit many events together?" The data mining method used for this purpose is association mining [56], [57]. The database $\mathcal{D}^1$ is scanned by a pattern matching algorithm (given in Appendix B), and all the entity pairs appearing together are populated into database $\mathcal{D}^7$ (where an entity pair appears as many times as they are seen together). Then, association mining is carried out to compute the entity pairs that appear together frequently, and this information is populated into database $\mathcal{D}^8$. Association mining provides us with the frequent itemsets, namely itemsets that appear together frequently. Only the itemsets that appear at least "minimum support (count) threshold" times are mined and listed. Table V gives a snapshot of $\mathcal{D}^8$ for our case study, where the minimum support threshold is given in terms of support count (absolute threshold) as 6. So, only the entity pairs that appear together at least six times are selected in the association mining analysis and for further analysis. From Table V we can observe that entities {A150, A161} have entered and/or exited together 39 times, which is more than the number of events. This means that they entered and exited together many times during the events, as well, revealing a social connection between these two entities. Other entity pairs with the "strongest" social connection include {A164, A150} and {A009, A150}. It should be noticed here that A150 appears frequently with A161, A164, and A009. So A150 is among the most social entities. The analysis of "social" entities will be extended later in the illustration of Insight 18.

The next illustration is for Insight 17, and is given in Fig. 9. This insight answers the question "How are the entities related to each other?" The identified relationship networks can be used for personalization and generating recommendations for human entities [45]. The data mining methods used for this purpose are network visualization and analysis [58]–[60]. Fig. 9 provides a grid visualization of the 163 entities that appear in $\mathcal{D}^8$. Each circular node represents an entity; the area of each circle represents the support count (number of times the entity is observed in $\mathcal{D}^8$); arcs between

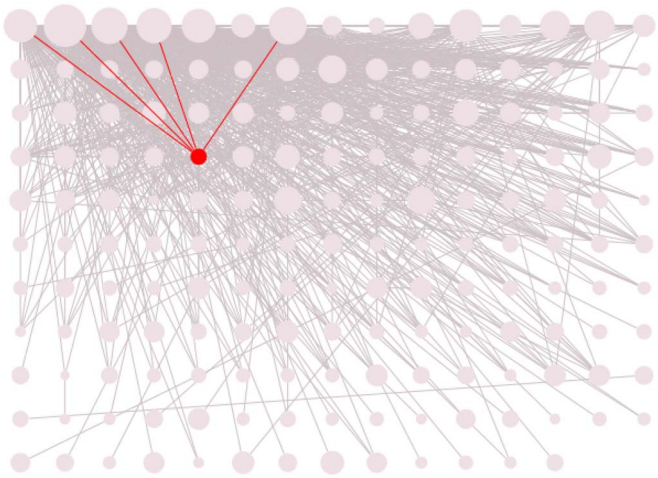

Fig. 9. Results for association mining analysis with grid visualization.

TABLE VI
LIST OF TOP 10 "SOCIAL" AND 5 OF THE "NOT SOCIAL"

Node (Attendee)	Degree	Betweenness Centrality
A161	118	2925.52
A150	110	2637.69
A164	99	2470.40
A127	87	1567.21
A009	83	1379.32
A221	71	1012.62
A126	36	445.83
A240	63	359.46
A119	43	350.30
...	...	...
A234	1	0.00
A242	1	0.00
A251	1	0.00
A255	1	0.00
A263	1	0.00

nodes represent an association between two entities. The visualization is constructed so as to minimize arc crossings. The entities in the upper region of the visualization are those that appear in many interactions. Among those that appear in many interactions, those with smaller area are even more interesting, since we can be inclined in thinking that their interactions were not due to frequent entry-exits, but rather due to interactions with other entities. Furthermore, by visual querying, it is possible to observe how each entity is related to each other entity. In Fig. 9, a particular node (entity) is selected and all the associations that it has are highlighted.

Another way of characterizing the nodes (entities) is to compute their graph metrics. One of these metrics is degree, which denotes the number of connections for each node. It is an integer value, and it is the summation of in degrees and out degrees of the node. Another such metric is betweenness centrality, which represents the total number of shortest paths for each pair of nodes, if the node is on that path (it can take values between 0 and 1). Detailed information on this and other graph metrics can be found in [59] and [60].

The next illustration is for Insight 18, and is given in Table VI. This insight answers the question "Which entities are social and which are not social?" The data mining method used for this purpose is network analysis, and specifically network

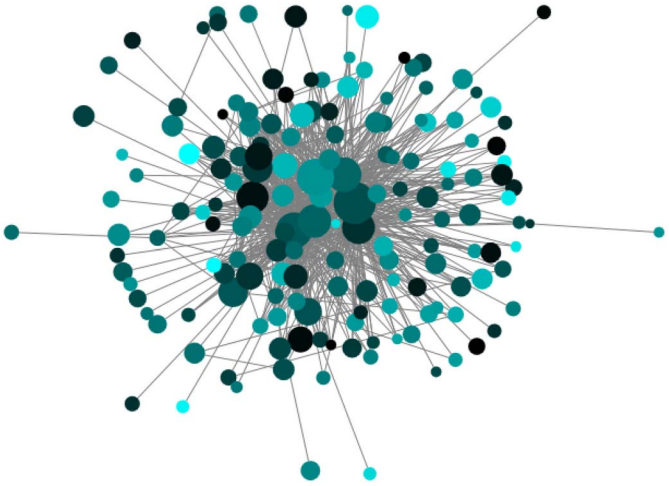

Fig. 10. Results for association mining analysis with Harel-Koren layout algorithm.

characterization through computing node metrics [59], [60]. Table VI lists the most active and social entities (A161 through A119), as well as some of the least associated ones (which appear only six times with other entities). The most social entities have high value for betweenness centrality, indicating that they are at the "crossroads" of social networks.

The next illustration is for Insight 19, and is given in Fig. 10. This insight answers the question "How can the behavioral attributes be analyzed together with the social network?" This analysis is specifically aimed at the scenarios where the entities are humans. The data mining method used for this purpose is network visualization, where some behavioral attributes are mapped onto the nodes. The network in Fig. 10 is exactly the same as that in Fig. 9 with respect to node-link structure. However, the selected layout algorithm is different (Harel-Koren algorithm; [61]), and nodes are colored according to average stay duration (a behavioral attribute). Lighter colors denote longer average stay durations. Size again denotes the support count of the node. The visualization in Fig. 10 is constructed using $\mathcal{D}^9$, which is obtained by joining the two different databases of $\mathcal{D}^6$ (entity profile data) and $\mathcal{D}^8$ (association graph). The nodes in the center are social and the ones on the outside are not social. However, we are now also able to see which social attendees stay in the events for long, and which stay less. Thus, we are able to see the social attendees that enhance our desired goals (longer stay durations in sessions with less frequent entry-exits, in our case study) versus the social attendees that disrupt the system (by staying very short in sessions and entering and exiting many times). So we are not only able to identify the social network and the socialization the entities have on the network, but also the direction of the effects (positive or negative) that they have.

V. Practical Implications

In this section, we discuss how the insights and information obtained through our analysis can be used in a multitude of ways, for improving the system and achieving various goals.

First, information regarding similar-behaving entities in a schedule-based system can be used in several ways.

1) In the context of social event management, ubiquitous information systems can use this information to suggest new people for professional social networks. For example, in a conference, when two attendees are identified as entering and exiting similar events, the conference mobile application can recommend them each other to add into LinkedIn and other professional social networks. Another use of the information is the suggestion of events to social event attendees in which similar-behaving attendees have already entered.
2) In the context of manufacturing, if one of the entities has entered the production system, *a priori* planning can be done for similar-behaving entities. Furthermore, the production system can be set up to accommodate not only the already entered entity but also those that may potentially enter, in a way to reduce total setup time.
3) In the context of warehousing, an example scenario where the information on similar-behaving entities can be used in the following: Consider pallets of similar-behaving products entering the warehouse. There is a high chance that they will also exit together. Therefore, the warehouse management system software can be programmed so as to allocate neighboring locations for these two pallets, so that they can be put away and picked on the same route, saving time and cost.
4) In the context of healthcare, similar-behaving entities can be equipment used for surgical operations. In this scenario, these equipment can be stored in the same storage room when not in use. This way, they can be accessed in the least possible time when an urgent surgical operation is to be conducted.
5) In the context of education, similar-behaving students can be identified automatically based on their entries and exits to classrooms. Then this information can be populated into the school's information system databases. In case a student cannot be reached by phone, the school management or the instructors can try to reach him through contacting his friend.
6) Finally, in the context of tourism, visitors in a museum can be offered special places of interest in the museum (visited by similar-behaving visitors) through the smart mobile devices that are guiding them.

Information regarding groups of similar events in a schedule-based system can also be used in a multitude of ways for improving the system and achieving various goals. One example use case from social event management is the joint design and improvement of similar sessions in the social event. The session managers can come together and discuss possible opportunities of improving the similar sessions in future conferences.

Information regarding the correlation between events can be used for improving schedules. For example, in the context of manufacturing, successive production periods with negative correlation may experience great changes in the product mix entering the production. These can also be sources of long

setup times and costs. Therefore, schedule can be adjusted based on the results of data mining.

Information regarding earlier events affecting later events, as well the predictability of entry of specific entities to an event, can also be used for benefiting the system. Consider a warehousing scenario where the entities are the various types of products loaded on pallets. Based on the past behaviors of these products, one can estimate for each product the probability of entering a particular warehouse zone during a particular time interval (event). This can be used to predict whether capacity will be exceeded in that zone of the warehouse during that time interval. Then, if necessary, additional capacity can be created, for example, through establishing temporary additions to that zone using pallet stacking frames.

Finally, the insights regarding social attendees in the system can be utilized in many ways. For example, in social event management, once these social attendees are determined, they can be consulted for help in promoting newly established sessions or for increasing membership to the organizing society.

VI. Conclusion

The importance of RFID systems for data collection and processing is ever increasing. RFID systems find applications in a very wide range of domains, including in schedule-based systems, which operate based on (or contain within) a schedule of events. In this paper, we have presented a comprehensive framework for mining of RFID data coming from schedule-based systems, for the first time in the literature. Our framework is generic, and can be applied to any schedule-based system that operates as described.

There exists two very fundamental future research avenues for extending this paper.

1) Our framework currently accommodates only a single location for all the events. This limitation was assumed because of the limitations of the data that motivated our research. So the existing framework can be extended to accommodate multilocation settings, and the discovery of new types of insights. Furthermore, one could explore which group of entities are related to which group of events, when the location data is available.
2) RFID tags can collect and/or carry not only location and time information, but other information, as well. Such information typically includes entity type, entity affiliation, physical attributes, and assigned attributes. In a logistics context, examples of these attributes are product type, manufacturer, weight, and price [46]. The additional information may also be collected through various sensors (e.g., temperature and GPS) integrated within or mounted on the tags. While the framework that we have presented here considers only time data in relation to schedule data, it can be highly enriched with the incorporation of analysis of these additional attributes. For example, scheduling and plans are important in manufacturing context, and are very much dependent on quality level achieved in the production process. Quality-related attributes can be analyzed together with product attributes and schedule data to improve the production process in the dimensions of quality, time, and cost. To this end, the remining framework of [62] can be integrated with the framework here to augment the data and to discover further insights.
3) The other fundamental research avenue is extending the framework from the temporal domain to the spatio-temporal domain, by extending it to handle multiple locations.
4) While the analysis of serial events is fundamental, consideration can be made in future research for concurrent events (events that can independently take place at the same time) in the system. The consideration for concurrent events would require significant changes in the augmentation algorithm, as it would require complex event processing [17]. However, the applicability of the current framework, as well as the types of analysis and the insights obtained, would still be relevant and useful.
5) One of the important challenges in industrial applications is the challenge of big data [63]. A possible future research can involve the development of the framework to accommodate for big data applications. To support the large volumes of input data, when the proposed framework is implemented, the data processing of this framework should be split into independent tasks to support parallel processing systems such as MapReduce. As indicated by Fig. 2, our framework has very few interactions between different branches of data flows and thus splitting the overall data processing into multiple tasks is possible and can be highly feasible. The methods for MapReduce implementation of the individual data mining and data visualization algorithms used in this paper, such as hierarchical clustering, can be found in [64]. Hence, the proposed framework can support the big data environment if its implementation is properly designed.

Other possible future research avenues include the following.

1) The concepts and methods used for the analysis of behavior in electronic games and virtual worlds [44], [65], [66] can be used in the analysis of RFID data, and vice versa.
2) The methods used for analyzing animal societies based on RFID data [67] can be adopted to analyzing the movement of entities in schedule-based systems in general.
3) Data from RFID (and other types of sensors) have been used in [4], [68], and [69] to (optimally) allocate the RFID readers. Data mining frameworks can be integrated with such methods to come up with better allocation of reader within an environment.
4) The study can be extended such that it encompasses more of the available data mining algorithms and techniques. For example, besides using k-Means Clustering in the unsupervised learning process, one can use k-Means++ [70], to reduce both clustering errors and running times.

Algorithm 1: Generate $\mathcal{D}^1$

Input: $\mathcal{D}^0, \mathcal{D}'$
Output: $\mathcal{D}^1$
foreach $d^0 \in \mathcal{D}^0 = \langle r, i, u_{ir}, U_{ir} \rangle$ **do**
  $d^1 = \langle r, i, u_{ir}, U_{ir}, j_1 = intervalOf(u_{ir}),$
  $j_2 = intervalOf(U_{ir}), \varepsilon_1 = intervalType(j_1),$
  $\varepsilon_2 = intervalType(j_2), s_{j_1}, s_{j_2}, f_{j_1}, f_{j_2} \rangle;$
  $\mathcal{D}^1 \bigsqcup d^1;$
end

5) Last but not least, mining of RFID data can be used in the general context of ambient intelligence applications, which are surveyed and discussed in [71]–[74].

APPENDIX A
AUGMENTATION ALGORITHMS

The first augmentation algorithm is shown Algorithm 1.

By using the schedule data $\mathcal{D}'$, this algorithm augments each record of the RFID database $\mathcal{D}^0$ with the information of the intervals covering the entry time and exit time. This information includes the type of interval, start time, and finish time. The augmented records forms a new database $\mathcal{D}^1$ for further analysis. Algorithm 1 includes a single loop that requires the initial construction of the lookup tables for the lookup functions. Each lookup has to scan through the J intervals for each of the R records. Running time of this initialization stage is $O(RJ)$. After this, each record is augmented, taking $O(R)$ time. So the running time of Algorithm 1 is $O(RJ)$.

The second augmentation algorithm is shown Algorithm 2.

For each record of database $\mathcal{D}^1$, this algorithm first identifies the types of the sequence of intervals that partially or completely falls between the entry time u and exit time U of that particular record. If an interval j is an event, its entry time scenario is analyzed to derive e and p. This is followed by the analysis of exit time scenario to determine x and q. Finally, the time T spent on that event, the α' and the β' are computed for that event based on e, p, x, and q. The first four fields in this record are then augmented with intermediary data e, p, x, q as well as the computed t, α', and β'. The augmented new record is added to a new database $\mathcal{D}^2$. This procedure is repeated for all records in database $\mathcal{D}^1$. Algorithm 2 has two interleaved loops, and runs for each record and for each interval. So the running time of Algorithm 2 is $O(RJ)$.

In the below algorithm, by noting that two successive break intervals are impossible and at least part of the event falls within the time span bounded by u and U, the following possible entry time scenarios for the event are considered.

1) Entry time u falls within the event.
2) Entry time u is before the start time of the event.
   a) The interval $j-1$ preceding the event and in the sequence is a break.
      i) The second interval $j-2$ preceding the event and in the sequence is an event.
      ii) The second interval $j-2$ preceding the event and in the sequence does not exist (Type of interval is null).
   b) The interval $j-1$ preceding the event and in the sequence is an event.

Similarly, the possible exit time scenarios under consideration are as follows.

1) Exit time U falls within the event.
2) Exit time U is after the finish time of the event.
   a) The interval $j+1$ following the event and in the sequence is a break.
      i) The second interval $j+2$ following the event and in the sequence is an event.

Algorithm 2: Generate $\mathcal{D}^2$

```
Input: D^1
Output: D^2
foreach d^1 ∈ D^1 do
    u = d^1 · u_ir;
    U = d^1 · U_ir;
    foreach j ∈ {j : j_1 ≤ j ≤ j_2} do
        if intervalType(j) = Event then
            /* Analyze different entry time scenarios */
            if u ≥ s_j then
                e = u;
                p = u;
                entryStatus = LateEntry;
            else
                e = s_j;
                if intervalType(j − 1) = Break then
                    entryStatus = EarlyEntry;
                    if intervalType(j − 2) = Event then
                        p = s_j − (f_{j−1} − s_{j−1})/2;
                    end
                    if intervalType(j − 2) = null then
                        p = u;
                    end
                end
                if intervalType(j − 1) = Event then
                    entryStatus = EntryFromPreEvent;
                    p = s_j;
                end
            end
            /* Analyze different exit time scenarios */
            if U ≤ f_j then
                x = U;
                q = U;
                exitStatus = EarlyExit;
            else
                x = f_j;
                if intervalType(j + 1) = Break then
                    exitStatus = LateStatus;
                    if intervalType(j + 2) = Event then
                        q = f_j + (f_{j+1} − s_{j+1})/2;
                    end
                    if intervalType(j + 2) = null then
                        q = U;
                    end
                end
                if intervalType(j + 1) = Event then
                    exitStatus = ExitIntoNextEvent;
                    q = f_j;
                end
            end
            /* Compute the metrics */
            T = x − e;
            α' = s_j − p;
            β' = q − f_j;
        end
        d ← ⟨i, j, r, u, U, e, x, T, p, q, α', β', entryStatus, exitStatus⟩;
        D^2 ⨆ d;
    end
end
```

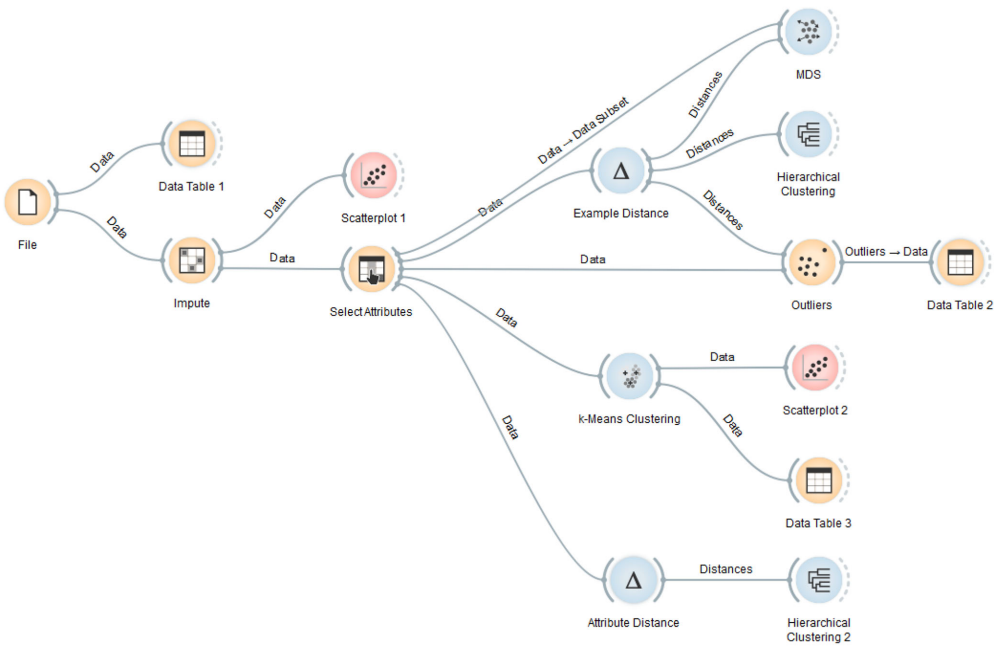

Fig. 11. Unsupervised data mining process conducted in this paper.

Algorithm 3: Generate $\mathcal{D}^{7a}$ and $\mathcal{D}^{7b}$ Whose Union Forms $\mathcal{D}^7$

Input: $\mathcal{D}^1$
Output: $\mathcal{D}^{7a}$ *and* $\mathcal{D}^{7b}$
$\mathcal{N}^1$:*Number of records in* $\mathcal{D}^1$
$\mathcal{T}_{RD}$:*Time condition for relationship detection*
for $m \leftarrow 1$ **to** $\mathcal{N}^1 - 1$ **do**
  d_m^1=*the* m^{th} *record in* $\mathcal{D}^1$;
  for $k \leftarrow m+1$ **to** $\mathcal{N}^1$ **do**
    d_k^1=*the* k^{th} *record in* $\mathcal{D}^1$;
    if $d_m^1 \cdot i \neq d_k^1 \cdot i$ **then**
      if $\left|d_m^1 \cdot u_{ir} - d_k^1 \cdot u_{ir}\right| \leq \mathcal{T}_{RD}$ **then**
        $\mathcal{D}^{7a} \bigsqcup \left\langle d_m^1 \cdot i, d_k^1 \cdot i \right\rangle$;
      end
      if $\left|d_m^1 \cdot U_{ir} - d_k^1 \cdot U_{ir}\right| \leq \mathcal{T}_{RD}$ **then**
        $\mathcal{D}^{7b} \bigsqcup \left\langle d_m^1 \cdot i, d_k^1 \cdot i \right\rangle$;
      end
    end
  end
end

ii) The second interval $j+2$ following the event and in the sequence does not exist (Type of interval is *null*).

b) The interval $j+1$ following the event and in the sequence is an event.

Different scenarios of entry time and exit time have different expressions for e, x, p, q, respectively as presented in the algorithm. The computation for the metrics, however, is unified in the sense that it is independent of the types of scenarios.

Appendix B
Pattern Matching Algorithm

The pattern matching algorithm is shown Algorithm 3.

This algorithm first generates all possible combinations of two arbitrary records from $\mathcal{D}^1$ that have different entities. For each combination, if the entry time difference between the two records are less than or equal to a predefined length $\mathcal{T}_{RD}$, the transaction ID for entry time is first updated. Then two new records with updated transaction ID, entity, entry time, and record ID are generated and added to a database $\mathcal{D}^{7a}$. Similarly, if the exit time difference is less than or equal to $\mathcal{T}_{RD}$, the transaction ID for exit time is updated and two new records are inserted into a database $\mathcal{D}^{7b}$. The union of $\mathcal{D}^{7a}$ and $\mathcal{D}^{7b}$ form a new database $\mathcal{D}^7$ containing all detected pairs showing the close relationship between entry or exit time of two entities for a particular event. Algorithm 3 has two interleaved loops, each executed for up to R records. So the running time of Algorithm 3 is $O(R^2)$.

Appendix C
Data Mining Processes

Figs. 11 and 12 display the data mining processes carried out. The first process in Fig. 11 shows an unsupervised machine learning model, whereas the second process shown in Fig. 12 shows a supervised machine learning model.

The unsupervised data mining process (Fig. 11) starts with reading data from file (File block), verifying that the data is read correctly (Data Table 1 block), and handling any missing values (Impute block). Next, data is again verified, this time visually, using a scatter plot (Scatterplot 1 block). The attributes are selected and specified (Select Attributes) and the unsupervised learning is initiated. The first type of analysis uses entity-entity distances (Example Distance) and conducts MDS (MDS), as well as Hierarchical Clustering, and detects any Outliers. The next analysis is k-Means Clustering, whose results are visually inspected (Scatterplot 2) and exported into a data table (Data Table 2). The final analysis is the computation of distances between

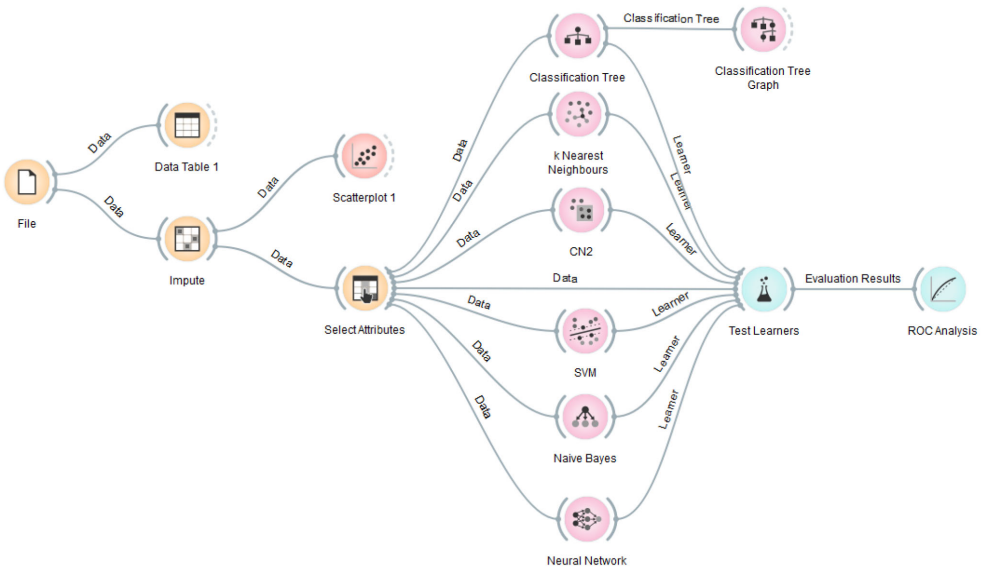

Fig. 12. Supervised data mining process conducted in this paper.

events (Attribute Distance) and the conduct of hierarchical clustering (Hierarchical Clustering 2).

The supervised data mining process (Fig. 12) also starts with the same steps. However, the attribute selection is different, because -unlike the previous process- one categorical attribute (S27, in our case study) has to be selected as the class attribute to be predicted. Next, multiple classifiers are applied, and their performances are tested and compared (Test Learners and ROC Analysis blocks). One of the classifiers has an additional benefit. Besides being used in classification analysis, the Classification Tree classifier is used in constructing the Classification Tree Graph.

APPENDIX D
SOFTWARE TOOLS USED

There exist a multitude of data analysis and data mining software tools, and we have used different tools for different purposes. MATLAB[1] was used for coding the developed and presented algorithms. Orange[2] data mining software [75] was used for clustering, classification, and classification tree analysis. RapidMiner[3] [76] was used to compute the correlation matrix for the sessions. Borgelt's implementation of the Apriori algorithm[4] [77]–[79] was used to compute frequent itemsets (attendees frequently appearing together). Finally, NodeXL[5] [80] was used to visualize association mining results and to compute graph metrics, enabling association-based social network analysis.

ACKNOWLEDGMENT

The authors thank A. Altunbaş and A. E. Altunbaş from Borda Technology and E. Eryarsoy from İstanbul Şehir University for introducing the problem to the research group and providing the data for the case study. The authors also thank U. Kaymaz, B. Dönmez, and Ç. Başel from Sabancı University for their assistance in the editing of this paper.

[1] http://www.mathworks.com
[2] http://orange.biolab.si/
[3] http://rapidminer.com/
[4] http://www.borgelt.net/apriori.html
[5] http://nodexl.codeplex.com/

REFERENCES

[1] R. J. Schonberger, "Applications of single-card and dual-card Kanban," *Interfaces*, vol. 13, no. 4, pp. 56–67, 1983.
[2] W. C. Benton, Jr., "Push and pull production systems," in *Wiley Encyclopedia of Operations Research and Management Science*. Hoboken, NJ, USA: Wiley, 2011.
[3] X. Zhu, S. K. Mukhopadhyay, and H. Kurata, "A review of RFID technology and its managerial applications in different industries," *J. Eng. Technol. Manag.*, vol. 29, no. 1, pp. 152–167, 2012.
[4] A. Oztekin, F. M. Pajouh, D. Delen, and L. K. Swim, "An RFID network design methodology for asset tracking in healthcare," *Decis. Support Syst.*, vol. 49, no. 1, pp. 100–109, 2010.
[5] W.-P. Liao, T. M. Y. Lin, and S.-H. Liao, "Contributions to radio frequency identification (RFID) research: An assessment of SCI-, SSCI-indexed papers from 2004 to 2008," *Decis. Support Syst.*, vol. 50, no. 2, pp. 548–556, 2011.
[6] J. Han, H. Gonzalez, X. Li, and D. Klabjan, "Warehousing and mining massive RFID data sets," in *Advanced Data Mining and Applications*. Heidelberg, Germany: Springer, 2006, pp. 1–18.
[7] E. N. Cinicioglu, P. P. Shenoy, and C. Kocabasoglu, "Use of radio frequency identification for targeted advertising: A collaborative filtering approach using Bayesian networks," in *Symbolic and Quantitative Approaches to Reasoning With Uncertainty*. Heidelberg, Germany: Springer, 2007, pp. 889–900.
[8] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Burlington, MA, USA: Morgan Kaufmann, 2011.
[9] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, 1st ed. Prentice Hall College Div., Upper Saddle River, NJ, USA, 1994.
[10] M. L. Pinedo, *Planning and Scheduling in Manufacturing and Services*, 2nd ed. Dordrecht, The Netherlands: Springer, 2009.
[11] Y. Yin, M. Liu, J. Hao, and M. Zhou, "Single-machine scheduling with job-position-dependent learning and time-dependent deterioration," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 1, pp. 192–200, Jan. 2012.
[12] J. S. K. Lau, G. Q. Huang, K. L. Mak, and L. Liang, "Agent-based modeling of supply chains for distributed scheduling," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 5, pp. 847–861, Sep. 2006.
[13] X. Qiu and H. Y. K. Lau, "An AIS-based hybrid algorithm for static job shop scheduling problem," *J. Intell. Manuf.*, vol. 25, no. 3, pp. 489–503, 2014.

[14] R. Balasundaram, N. Baskar, and R. S. Sankar, *Discovering Dispatching Rules for Job Shop Scheduling Using Data Mining* (Advances in Intelligent Systems and Computing), vol. 178. Heidelberg, Germany: Springer, 2013, pp. 63–72.

[15] C. Rainer, "Data mining as technique to generate planning rules for manufacturing control in a complex production system: A case study from a manufacturer of aluminum products," in *Robust Manufacturing Control* (Lecture Notes in Production Engineering), K. Windt, Ed. Heidelberg, Germany: Springer, 2013.

[16] C. L. Wang, G. Rong, W. Weng, and Y. P. Feng, "Mining scheduling knowledge for job shop scheduling problem," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 800–805, 2015.

[17] R. Helaoui, M. Niepert, and H. Stuckenschmidt, "Recognizing interleaved and concurrent activities using qualitative and quantitative temporal relationships," *Pervasive Mobile Comput.*, vol. 7, no. 6, pp. 660–670, 2011.

[18] G. Mariscal, O. Marbán, and C. Fernández, "A survey of data mining and knowledge discovery process models and methodologies," *Knowl. Eng. Rev.*, vol. 25, no. 2, pp. 137–166, 2010.

[19] S. Sharma, K.-M. Osei-Bryson, and G. M. Kasper, "Evaluation of an integrated knowledge discovery and data mining process model," *Expert Syst. Appl.*, vol. 39, no. 13, pp. 11335–11348, 2012.

[20] W.-S. Ku, H. Chen, H. Wang, and M.-T. Sun, "A Bayesian inference-based framework for RFID data cleansing," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 10, pp. 2177–2191, Oct. 2013, doi: 10.1109/TKDE.2012.116.

[21] A. I. Baba, H. Lu, X. Xie, and T. B. Pedersen, "Spatiotemporal data cleansing for indoor RFID tracking data," in *Proc. IEEE 14th Int. Conf. Mobile Data Manag. (MDM)*, vol. 1. Milan, Italy, 2013, pp. 187–196.

[22] T. C. Poon *et al.*, "A RFID case-based logistics resource management system for managing order-picking operations in warehouses," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 8277–8301, 2009.

[23] A. Ilic, T. Andersen, and F. Michahelles, "Increasing supply-chain visibility with rule-based RFID data analysis," *IEEE Internet Comput.*, vol. 13, no. 1, pp. 31–38, Jan./Feb. 2009.

[24] D. Shuping and W. Wright, "Geotime visualization of RFID supply chain data," *RFID J.*, Mar./Apr. 2005, pp. 1–6.

[25] S. Miyazaki, T. Washio, and K. Yada, "Analysis of residence time in shopping using RFID data—An application of the kernel density estimation to RFID," in *Proc. IEEE 11th Int. Conf. Data Min. Workshops (ICDMW)*, Vancouver, BC, Canada, 2011, pp. 1170–1176.

[26] B. Fang *et al.*, "A novel mobile recommender system for indoor shopping," *Expert Syst. Appl.*, vol. 39, no. 15, pp. 11992–12000, 2012.

[27] S. Sakurai, M. Sanbe, and K. Watanabe, "Application of the RFID data mining to an apparel field," in *Proc. 13th Int. Conf. Netw. Based Inf. Syst. (NBiS)*, Takayama, Japan, 2010, pp. 28–35.

[28] J. Lyu, Jr., S.-Y. Chang, and T.-L. Chen, "Integrating RFID with quality assurance system—Framework and applications," *Expert Syst. Appl.*, vol. 36, no. 8, pp. 10877–10882, 2009.

[29] C. K. H. Lee, K. L. Choy, G. T. S. Ho, and K. M. Y. Law, "A RFID-based resource allocation system for garment manufacturing," *Expert Syst. Appl.*, vol. 40, no. 2, pp. 784–799, 2013.

[30] W. Wen, "An intelligent traffic management expert system with RFID technology," *Expert Syst. Appl.*, vol. 37, no. 4, pp. 3024–3035, 2010.

[31] C.-Y. Tsai, J. J. H. Liou, C.-J. Chen, and C.-C. Hsiao, "Generating touring path suggestions using time-interval sequential pattern mining," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3593–3602, 2012.

[32] Y. Meiller, S. Bureau, W. Zhou, and S. Piramuthu, "Adaptive knowledge-based system for health care applications with RFID-generated information," *Decis. Support Syst.*, vol. 51, no. 1, pp. 198–207, 2011.

[33] H.-H. Hsu, Z. Cheng, T. K. Shih, and C.-C. Chen, "RFID-based personalized behavior modeling," in *Proc. IEEE Symp. Workshops Ubiquitous Auton. Trusted Comput. (UIC-ATC)*, Brisbane, QLD, Australia, 2009, pp. 350–355.

[34] M. Delgado, M. Ros, and M. A. Vila, "Correct behavior identification system in a tagged world," *Expert Syst. Appl.* vol. 36, no. 6, pp. 9899–9906, 2009.

[35] E. Masciari, "A framework for outlier mining in RFID data," in *Proc. 11th Int. Database Eng. Appl. Symp. (IDEAS)*, Banff, AB, Canada, 2007, pp. 263–267.

[36] H. Gao and H. Liu, "Data analysis on location-based social networks," in *Mobile Social Networking: An Innovative Approach*, A. Chin and D. Zhang, Eds. New York, NY, USA: Springer, 2014.

[37] T.-S. Chen, Y.-S. Chou, and T.-C. Chen, "Mining user movement behavior patterns in a mobile service environment," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 42, no. 1, pp. 87–101, Jan. 2012.

[38] P. Kazienko, K. Musial, and T. Kajdanowicz, "Multidimensional social network in the social recommender system," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 4, pp. 746–759, Jul. 2011.

[39] M. Szomszor *et al.*, "Providing enhanced social interaction services for industry exhibitors at large medical conferences," in *Proc. Develop. E-Syst. Eng. (DeSE)*, Dubai, United Arab Emirates, 2011, pp. 42–45.

[40] A. Chin *et al.*, "Using proximity and homophily to connect conference attendees in a mobile social network," in *Proc. 32nd Int. Conf. Distrib. Comput. Syst. Workshops (ICDCSW)*, Macau, China, 2012, pp. 79–87.

[41] W. Reinhardt, T. Messerschmidt, and T. Nelkner, "Awareness support in scientific events with SETapp," in *Proc. 1st Eur. Workshop Awareness Reflect. Learn. Netw.*, Palermo, Italy, 2011, pp. 100–115.

[42] J. Bravo, R. Hervás, I. Sánchez, G. Chavira, and S. Nava, "Visualization services in a conference context: An approach by RFID technology," *J. Univers. Comput. Sci.*, vol. 12, no. 3, pp. 270–283, 2006.

[43] M. Atzmueller, "Mining social media: Key players, sentiments, and communities," *Wiley Interdiscipl. Rev. Data Min. Knowl. Disc.*, vol. 2, no. 5, pp. 411–419, 2012.

[44] D. A. Huffaker, C.-Y. Teng, M. P. Simmons, L. Gong, and L. A. Adamic, "Group membership and diffusion in virtual worlds," in *Proc. IEEE 3rd Int. Conf. Soc. Comput. Privacy Security Risk Trust (SOCIALCOM)*, Boston, MA, USA, 2011, pp. 331–338.

[45] M. Atzmueller, S. Doerfel, A. Hotho, F. Mitzlaff, and G. Stumme, "Face-to-face contacts during a conference: Communities, roles, and key players," in *Proc. 2nd Int. Workshop Min. Ubiquitous Soc. Environ.*, Athens, Greece, 2011, p. 25.

[46] H. Gonzalez *et al.*, "Modeling massive RFID data sets: A gateway-based movement graph approach," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 1, pp. 90–104, Jan. 2010.

[47] Y. Wang, E.-P. Lim, and S.-Y. Hwang, "Efficient mining of group patterns from user movement data," *Data Knowl. Eng.*, vol. 57, no. 3, pp. 240–282, 2006.

[48] I. Borg, P. J. F. Groenen, and P. Mair, *Applied Multidimensional Scaling*. Heidelberg, Germany: Springer, 2012.

[49] R. Bose, *Information Theory, Coding and Cryptography*. New Delhi, India: Tata McGraw-Hill, 2008.

[50] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis* (Wiley Series in Probability and Statistics), 1st ed. Hoboken, NJ, USA: Wiley, 2005.

[51] J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *Amer. Stat.*, vol. 42, no. 1, pp. 59–66, 1988.

[52] L. Rokach and O. Maimon, *Data Mining With Decision Trees: Theory and Applications*. Singapore: World Sci., 2008.

[53] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA, USA: MIT Press, 2010.

[54] C. E. Brodley and M. A. Friedl, "Identifying mislabeled training data," *J. Artif. Intell. Res.*, vol. 11, pp. 131–167, 1999.

[55] T. Fawcett, "ROC graphs: Notes and practical considerations for researchers," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 882–891, 2004.

[56] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *ACM SIGMOD Rec.*, vol. 22, no. 2, pp. 207–216, 1993.

[57] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th Int. Conf. Very Large Data Bases (VLDB)*, vol. 1215. Santiago, Chile, 1994, pp. 487–499.

[58] I. Herman, G. Melancon, and M. S. Marshall, "Graph visualization and navigation in information visualization: A survey," *IEEE Trans. Vis. Comput. Graphics*, vol. 6, no. 1, pp. 24–43, Jan./Mar. 2000.

[59] C. Christensen and R. Albert, "Using graph concepts to understand the organization of complex systems," *Int. J. Bifurcat. Chaos*, vol. 17, no. 7, pp. 2201–2214, 2007.

[60] T. Opsahl, F. Agneessens, and J. Skvoretz, "Node centrality in weighted networks: Generalizing degree and shortest paths," *Soc. Netw.*, vol. 32, no. 3, pp. 245–251, 2010.

[61] D. Harel and Y. Koren, "Graph drawing by high-dimensional embedding," in *Graph Drawing*. Heidelberg, Germany: Springer, 2002, pp. 207–219.

[62] A. Demiriz, G. Ertek, T. Atan, and U. Kula, "Re-mining item associations: Methodology and a case study in apparel retailing," *Decis. Support Syst.*, vol. 52, no. 1, pp. 284–293, 2011.

[63] V. Mayer-Schönberger and K. Cukier, *Big Data: A Revolution That Will Transform How we Live, Work, and Think*. Boston, MA, USA: Mariner Books, 2014.

[64] T. Sun *et al.*, "An efficient hierarchical clustering method for large datasets with Map-Reduce," in *Proc. Int. Conf. PDCAT*, Higashihiroshima, Japan, 2009, pp. 494–499.
[65] K. Börner and S. Penumarthy, "Social diffusion patterns in three-dimensional virtual worlds," *Inf. Vis.*, vol. 2, no. 3, pp. 182–198, 2003.
[66] N. Hoobler, G. Humphreys, and M. Agrawala, "Visualizing competitive behaviors in multi-user virtual environments," in *Proc. Conf. Vis.*, Austin, TX, USA, 2004, pp. 163–170.
[67] G. Cabanes, Y. Bennani, and D. Fresneau, "Mining RFID behavior data using unsupervised learning," *Int. J. Appl. Logist.*, vol. 1, no. 1, pp. 28–47, 2010.
[68] W. Chang, D. Zeng, and H. Chen, "A stack-based prospective spatio-temporal data analysis approach," *Decis. Support Syst.*, vol. 45, no. 4, pp. 697–713, 2008.
[69] B. C. F. Cheung, S. L. Ting, A. H. C. Tsang, and W. B. Lee, "A methodological approach to optimizing RFID deployment," *Inf. Syst. Frontiers*, vol. 16, no. 5, pp. 923–937, 2012.
[70] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proc. 8th Annu. ACM SIAM Symp. Discr. Algorithms*, New Orleans, LA, USA, 2007, pp. 1027–1035.
[71] C. Ramos, J. C. Augusto, and D. Shapiro, "Ambient intelligence—The next step for artificial intelligence," *IEEE Intell. Syst.*, vol. 23, no. 2, pp. 15–18, Mar./Apr. 2008.
[72] D. J. Cook, J. C. Augusto, and V. R. Jakkula, "Ambient intelligence: Technologies, applications, and opportunities," *Pervasive Mobile Comput.*, vol. 5, no. 4, pp. 277–298, 2009.
[73] F. Sadri, "Ambient intelligence: A survey," *ACM Comput. Surveys*, vol. 43, no. 4, 2011, Art. no. 36.
[74] J. C. Augusto, H. Nakashima, and H. Aghajan, "Ambient intelligence and smart environments: A state of the art," in *Handbook of Ambient Intelligence and Smart Environments*. New York, NY, USA: Springer, 2010, pp. 3–31.
[75] J. Demšar *et al.*, "Orange: Data mining toolbox in Python," *J. Mach. Learn. Res.*, vol. 14, pp. 2349–2353, Aug. 2013.
[76] M. Hofmann and R. Klinkenberg, *RapidMiner: Data Mining Use Cases and Business Analytics Applications*. Boca Raton, FL, USA: CRC Press, 2013.
[77] C. Borgelt and R. Kruse, "Induction of association rules: Apriori implementation," in *Proc. 15th Conf. Comput. Stat. (Compstat)*. Berlin, Germany, 2002, pp. 395–400.
[78] C. Borgelt, "Efficient implementations of apriori and eclat," in *Proc. IEEE ICDM Workshop Frequent Itemset Min. Implement. (FIMI)*, Melbourne, FL, USA, 2003.
[79] C. Borgelt, "Recursion pruning for the apriori algorithm," in *Proc. 2nd IEEE ICDM Workshop Frequent Item Set Min. Implement. (FIMI)*, Brighton, U.K., 2004.
[80] D. L. Hansen, B. Shneiderman, and M. A. Smith, *Analyzing Social Media Networks With NodeXL: Insights From a Connected World*. Amsterdam, The Netherlands: Morgan Kaufmann, 2010.

Gürdal Ertek received the B.S. degree from the Bogazici University, Istanbul, Turkey, in 1994, and the Ph.D. degree from the School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA, in 2001.

He was with Sabanci University, Istanbul, and a Visiting Scientist with the Singapore Institute of Manufacturing Technology, Singapore. He is an Assistant Professor with the Rochester Institute of Technology at Dubai, Dubai, UAE. His current research interests include knowledge-based systems, warehousing and material handling, and data visualization and mining.

Dr. Ertek was a recipient of the Bogazici University Alumni Scholarship, the Haci Omer Sabanci Scholarship, and the Fulbright Scholarship throughout his education. He has served as a Reviewer for 50+ research and development projects submitted to TUBITAK (Turkish National Science Foundation), mostly on the topics of information technology and data analytics.

Xu Chi (M'09) received the bachelor's (Hons.) and Ph.D. degrees with the Electrical and Electronic School, Nanyang Technological University, Singapore, in 2010 and 2003, respectively.

He was a Researcher with the Positioning and Wireless Technology Center, Nanyang Technological University, Singapore, researching on RFID ranging and positioning using ultrawideband signal. He is currently a Research Scientist with the Planning and Operations Management Group, Singapore Institute of Manufacturing Technology, Singapore. His current research interests include information management for track and trace system and unstructured data mining.

Allan N. Zhang (M'10) received the B.S., M.S., and Ph.D. degrees from Wuhan University, Wuhan, China, in 1986, 1989, and 1992, respectively.

He is a Senior Scientist with the Singapore Institute of Manufacturing Technology, A*STAR, Singapore. He has over 20 years experience in knowledge-based systems and enterprise information systems development. His current research interests include knowledge management, data mining, machine learning, artificial intelligence, computer security, software engineering, software development methodology and standard, and enterprise information systems. He and his team are currently researching in manufacturing system analyses including data mining, supply chain information management, supply chain risk management using complex systems approach, multi-objective vehicle routing problems, and urban last mile logistics.