Adaptive and Personalized User Behavior Modeling in Complex Event Processing Platforms for Remote Health Monitoring Systems

Mohammad Mehdi Naseri[a] , Shima Tabibian[a ,*], Elaheh Homayounvala[b]

[a] *Cyberspace Research Institute, Shahid Beheshti University, Tehran, Iran*
[b] *School of Computing and Digital Media, London Metropolitan University, London, United Kingdom*
moha.naseri@mail.sbu.ac.ir, sh_tabibian@sbu.ac.ir, e.homayounvala@londonmet.ac.uk

Abstract

Taking care of people who need constant care is essential and its cost is rising every day. Many intelligent remote health monitoring systems have been developed from the past till now. Intelligent systems explainability has become a necessity after the worldwide adoption of such systems, especially in the health domain to explain and justify decisions made by intelligent systems. Rule-based techniques are among the best in terms of explainability. However, there are several challenges associated with remote health monitoring systems in general and rule-based techniques, specifically. In this research, an adaptive platform based on Complex Event Processing (CEP) has been proposed for user behavior modeling to provide adaptive and personalized remote health monitoring. This system can manage a massive amount of data in real-time utilizing the CEP engine. It can also avoid human errors in setting rules thresholds by extracting thresholds from previous data using JRip rule-based classifier. Moreover, a feature selection method is proposed to decrease the high number of features while maintaining accuracy. Additionally, a rule adaption method has been proposed to cope with changes over time. Additionally, a personalized rule adaption method is proposed to address the need for responsiveness of the system to the special requirements of each user. The experimental results on both hospital and activity data sets showed that the proposed rule adaption method improves the accuracy by about 15% compared to non-adaptive systems. Additionally, the proposed personalized rule adaption method has an accuracy improvement of about 3% to 6% on both mentioned datasets.
Keywords:
Remote health monitoring
User-behavior modeling
Complex event processing
Rule-based learning
Explainability
Personalised rule adaption

1. Introduction

Taking care of patients or people with disabilities (the elderly, children, etc.) in any environment (home, outdoors, or anywhere else) is important and necessary. Some people may have problems or disorders that require remote care. Due to the increase in the number of such people and health costs, it is necessary to develop systems that monitor users (patients) in different environments and send alarms to them or other people in charge, when necessary. These systems can act as a complement to the treatment staff or completely independent of them [1].

Due to the high sensitivity of intelligent health systems, the correct, on-time, and accurate operation of such systems are necessary. However, there are several challenges which these systems are facing. Some of them are briefly described in the following paragraphs:

- Intelligent health monitoring systems naturally generate a lot of data per time unit due to a large number of sensors and the need for their explainability to domain experts. On the other hand, according to the high sensitivity in the field of health, any delay in data processing or producing alarm messages can have irreparable consequences. Therefore, it is necessary to design an explainable system that can model users' behavior, process a large amount of data, and produce a suitable output in real-time [2]. Exploiting the tools to provide this processing model, such as rule-based methods like complex events processing (CEP) engine, can be useful in the design and development phase of these systems. Besides, it has been experienced that these sensors and devices do not have the same effect on the

*Corresponding Author
Address: Shahid Beheshti University, Shahid Shahriari Square, Daneshjou Boulevard, Shahid Chamran Highway, Tehran, Iran.
Postal code: 1983969411
Tel: +982129905482
This Research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

system output. Therefore, using methods or approaches to optimize the number of these sensors and devices, is one of the necessities of intelligent health monitoring systems. The artificial intelligence-based feature selection methods can help select more important and effective features among data collected from sensors.

- In the processing of intelligent health monitoring systems development (especially in rule-based systems like CEP), unavoidable human errors in rules threshold settings can have a significant impact on the final output. Sometimes the effects of human error in these systems will be more destructive and irreparable than other systems. Because patients' lives are directly related to them. As a result, using artificial intelligence-based techniques, especially machine learning algorithms to make these systems more intelligent and to reduce human interferences can help decrease human errors in setting rules thresholds [3].
- The requirements and approaches used in the development process of an intelligent health monitoring system lose their efficiency and effectiveness over time. For example, at the discretion of clients who use the system, several sensors or input data generators may be removed or added either in the environment or in connection with the patients' bodies. It is even possible to add another output to the previous output of the system as a new situation by changing the users' conditions/requirements. Moreover, it is possible that in a new critical situation such as new disease, thresholds might change for all patients. In this case, it is necessary to update the intelligent health monitoring system to cope with these conditions and defining an analytical approach that automatically calculates health rules thresholds [3].
- In the health monitoring context, the health data usually is analyzed based on comparing the extracted health measures with some predefined thresholds. Diseases and abnormal situations can be detected when the mentioned measures are above or below the threshold (due to the corresponding conditions). Early detection of these abnormalities supports the prediction of critical events and therefore prevents their occurrence. Hence, defining accurate thresholds is really important. Domain experts and healthcare staff usually define and update thresholds based on patient measurements as well as patient interviews. In fact, experts confirm that thresholds are not the same for all patients and can even change for patient from time to time. Moreover, two patients may cope with a same critical situation with different medical parameters [3]. On the other hand, intelligent health monitoring systems are generally developed based on a wide range of patient conditions. However, each patient may need a personalized system tailored to his/her specific circumstances. For instance, a young athlete person faces with a heart attack in a different range of heart rate in comparison to an elder with experience of heart surgery. Therefore, with continuous and regular adaptation, it is possible to consider the addition of new people to the users' domain and adapting the system with different users' behavior. As a result, the process of adapting and updating intelligent health monitoring systems is essential.

In the present work, an adaptive and personalized CEP Platform for remote health monitoring is presented. CEP method has been used to model, process, and manage the large volume of data generated in real-time in a health monitoring system. In addition, a machine learning method has been used to extract the rules and rules' thresholds for a CEP engine from the existing data [4]. Rule-based machine learning algorithms are chosen in this research due to the similarity of their extracted rules and rules which are used by CEP and their high explainability for domain experts in comparison to other machine learning methods.

Moreover, in this paper, two datasets called hospital and activity datasets, have been used, however the proposed platform has the potential to be used in other domains such as stock market. We decided to choose health monitoring context for evaluating our platform according to two reasons; first, the necessity for processing large amount of data in real time in health domain and second, the need of adaption and personalization for better monitoring. The activity dataset is large in terms of the dimension of each record of the data. Thus, the data dimensions have to be reduced to increase the training speed and to achieve better accuracy. For this purpose, a feature selection method is presented which is a combination of two feature selection approaches namely attribute evaluation and subset evaluation.

Another achievement of this work is to provide the possibility of updating the rules in order to do rule adaption. This is made possible by a built-in memory associated with the CEP engine and applying the JRip rule-based classifier to the previously stored data. After classifying data and extracting new rules, the current engine rule-set is rewritten in such a way that new rules that did not exist before, are added and the thresholds of the existed rules are readjusted. Another advantage of updating the rules is to personalize rule adaption that can make the system more robust and personalized to cope with different user behaviors.

In summary, the main contributions of this work are as follows:

- Using the extracted rules by a rule-based classifier in a CEP engine ruleset in order to process input data flows.
- Representing a new feature selection method by combining two feature selection approaches namely attribute evaluation and subset evaluation. (The results of seven attribute evaluation methods are utilized in our proposed approach and a sub set is extracted from them.)
- Rule adaptation by updating rules periodically in order to adapt system to new environmental conditions.
- Personalization by updating rules periodically in order to adapt system to recent user's behaviour and conditions.

The structure of the paper is as follows. In Section 2, the literature review of the health monitoring systems will be presented. Then, CEP, rule-based learning, and the relation

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

between them will be discussed in Section 3. Section 4 contains a discussion of the proposed method and its details. The experimental results will be evaluated in Section 5. Finally, Section 6 will conclude the paper.

2. Related Work

Remote health monitoring systems are one of the popular research topics for scientific collaboration between medical and computer science researchers. They aim to monitor the medical sign and situations of a patient remotely. As an example, the project proposed by Tsiourti et al. [5], is in fact a virtual assistant for the elderly that helps them take care of themselves on their own. This system, which is based on smart TVs, records and analyzes the behavior and the environmental conditions received via sensors connected in the environment and on the elderly person. Then, it generates different responses according to the corresponding conditions. In [6] Verma et al. presented a remote health monitoring system in smart homes by using fog computing. Embedded data mining, distributed storage and notification services at the edge of network were used for developing this system. A transmission based on event triggering-based data was used to process the data in real-time at fog layer. For evaluating this system, health data of 67 patients in an IoT system based on smart home were generated in a 30-day period. Results showed that the proposed model based on Bayesian Belief Network classifier had high accuracy and response time in detecting the state of an event in comparison with other classification algorithms.

Besides of home appliances like smart TVs, sensors are another useful device for remote monitoring. Radio-frequency identification (RFID) is one of the useful devices that can be connected to a person or every environment like his/her home or every critical thing that he/she uses like medicines. The system was proposed by Becker et al. [7], which is a drug-reminder based on (RFID) and can be considered as a system, in the form of a smart drug box. RFID tags are attached to the bottom of the medicine bottles and placed in a drawer. Below this drawer is an RFID reader that communicates with the attached labels to the bottles via an antenna and exchanges information with them. This information contains bottle exit from the drawer, bottle entry into the drawer, opening and closing the drawer, time of drug use, etc. Other kind of sensors used in some healthcare systems in hospital or homes. Albahari et al. [8] proposed a real-time health monitoring structure for hospital distributors based on the data of wearable health sensors. The number of health care services showing the condition of the hospital was collected from 12 hospitals in Baghdad. Hospitals were ranked using Multi-Criteria Decision-Making (MCDM) technique, the analytic Hierarchy Process (AHP) and VlseKriterijumska Optimizacija I Kompromisno Resenje (VIKOR). The integration of AHP and VIKOR solved the problems of choosing hospital. In the target validation, considerable differences were detected between the scores of the groups, which indicates that the ranking results were the same. In the evaluation, the proposed framework illustrated an advantage over the standard with 56.25%. Moreover, hospitals with fewer health care services received the lowest rankings while hospitals with multiple health care services received the highest rankings.

Internet of things (IoT) can manage and control devices like home goods and sensors like RFIDs and vital signs sensors in health monitoring systems. It can also provide an ability for a health monitoring system that works remotely. Kharel et al. [9] have proposed a structure for an intelligent health monitoring system. This system is an IoT application developed by LoRa wireless network and fog computing. The test results show that the proposed system can promise to convert the clinic-based health system to a patient-centered health system and provide integrated health services for all patients. The focus of Basu et al in [10], is to implement an IoT-based health monitoring system. This system uses non-invasive sensors to read various health parameters (temperature, blood oxygen saturation, and heart rate) and instantly shows it on a screen. In addition, the user has the option of sending data to the cloud so that it can store information, securely. The system can be easily accessed from anywhere in the world. Low cost, portability, user-friendly interface, and ease of data visualization are the main properties of this system. In [11] the authors distributed the complex and energy intensive computations of machine learning between the edge, fog, and cloud, based on the concept of self-awareness, which considers the complexity and reliability of the algorithm. The problem of epileptic seizures was considered as a real-world case study to demonstrate the importance of the proposed conscious approach.

Artificial intelligence and machine learning have been used in some health monitoring systems. Subasi et al. [12] have developed an intelligent health care system to use machine learning techniques to achieve modeling and cognition of daily life activities. The system performs this operation automatically, accurately, and efficiently by recognizing Extensive Human Activities (HAR). The manner of exploiting this approach using two different datasets, mobile and wearable body sensors, is shown for strong and accurate HAR. This paper has shown that identifying human activities based on sensor data is very challenging. In [13], an acceptable solution to overcome data overload and delay processing was provided by Gupta et al. which used in real-time sensory data collected from wearable devices for mental health monitoring. A modified k-medoid data clustering technique were presented based on similarity calculations of restricted clusters over a time frame to obtain a summary version of the data set for which the degree of lost information is minimal. A Convolutional Neural Network (CNN) was trained on this summarized dataset to classify mental conditions into basic categories, stress and entertainment. The results demonstrated a remarkable reduction in average runtime of 34% with a comparable accuracy to the original data set, thus providing immediate real-time healthcare analysis.

One of the major challenges of researches in this field is that they have paid less attention to some issues like

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

processing a large amount of continuous and heterogeneous data, users' behavior modeling, and producing suitable output in an acceptable time. Hence a platform like CEP can be useful for developing a system that solves these challenges by modeling, processing, and responding to users' massive data in an acceptable time.

CEP can manage and respond to the massive amount of data that is produced by sensors like vital signs sensors, GPS, RFID and, etc. The system introduced by Pathak et al. [14] is the sophisticated event-based health monitoring system (CRHMS). This system uses several health sensors (heart rate, respiration rate, blood pressure, and ECG) to store vital signs and a number of environmental sensors (accelerometer, GPS) to understand the environmental conditions of the elderly person inside the home. In this system, data streams are sent to a CEP-based system to detect abnormal conditions and generate alerts, after storing sensors' data by an Android phone. Another study by Yao et al. [2] suggested designing a smart hospital. The hospital provides a complex event-based framework for modeling surgical events and critical situations in a hospital equipped with RFID equipment.

As it was mentioned, in addition to sensors, devices like a smart TV or a smartphone can be used in health monitoring systems to monitor people remotely. Processing and responding in real-time to all data that is generated by devices require to utilize a real-time processing system like CEP. Pérez-Veredaet al. in [15], proposed to use a centralized mobile architecture that allows elderly people to have a device that shows a virtual health profile to them. This profile is extracted from the data collected from connected sensors and other smart devices such as heart rate monitor bands. At this stage, a CEP method is used to combine input data from different sources and analyze them to extract meaningful information for physicians.

In an IoT-based system also massive data flows are generated and a method that can process all of these flows in real-time (like CEP) is necessary. Dautov et al. in [16] propose an integrated hierarchical distributed data architecture in which different data sources at each level of IoT taxonomy are combined to produce on-time and accurate results. In this way, important decisions are made as soon as the necessary information is produced and collected with the least time delay. The proposed method in [16] uses the CEP technology. This technology specifically emphasizes streaming data processing, which is a key requirement for IoT devices with limited storage and time-consuming applications. Experimental results show that the proposed approach allows accurate decision-making at different levels of data integration. Therefore, it improves the overall performance and response time of public health services. In [17] Rahmani et al. presented an IoT-based event architecture to analyze data from trusted healthcare applications, including contexts, events, and service layers. Reliability parameters are considered at each layer, and the CEP method is applied as a new solution and automated intelligence at the event layer. The implementation results showed that the CEP method increases reliability, reduces costs, and improves the quality of health care. Caballero et al. [18] proposed a collaborative context-aware health mobile application supported by a software architecture integrated with technologies like IoT and CEP. People can collaborate with the mentioned application in a simple way and it can process and warn in real time with corresponding text alerts. In addition, this application is efficient while it uses low resource conservation. According to a user study of 42 people with different age ranges and abilities, this application and its key features could be very successful and practical. Researchers also have demonstrated that the system is not limited to the health context can adapt the used architecture to other specific areas, by considering the capacity to process the data provided by citizen collaboration in real time, an effective mechanism for adapting notifications to the user context and low resource consumption in the program.

According to these researches, CEP can be useful in coping and managing a large amount of date/data flow. Nevertheless, CEP engine rules should have developed by an expert. It means that human errors in developing rules are unavoidable. In [19] Manashty et al. provide a health protection platform (called HEAL) for detecting and predicting abnormal conditions. This system consists of three layers including the service layer, control layer, and data layer. It also uses a CEP engine to process analytics rules at runtime and to predict abnormal conditions. This system performs the analysis based on comparing vital signs with an interval. They suggested updating the intervals, but they did not provide a way to do it.

Research studies have tackled this issue using various approaches. Mdhaffar et al. in [3] proposed a health analysis approach to predict heart failure. This research is also based on the CEP combined with a statistical method. In this system, the patient's vital signs-received through various sensors- are sent to a CEP engine that uses interval-based rules. In this research, a statistical method has been used to update the intervals for better diagnosis. In [20], Teng et al. describe a method based on a meta-heuristic method used as a classifier to obtain optimal parametric rules for the CEP engine. In particular, this approach optimizes CEP rules by improving their behavior control parameters, based on improved alarm detection criteria. As a result, the proposed scheme retrieves the parameterized rules optimally. Olabarrieta et al. In [21] have provided a method for updating the adaptive rules to maintain the changing environment. The first step is to eliminate unnecessary errors and data to improve processing efficiency. Then, it will be completed by abnormal rule detection which is generated by monitoring changes in Page Hinckley Test (PHT). In the second step, updating the rule condition part is done by limiting the compatible cluster rules and calculating the effective value. Furthermore, some studies which try to correct human interferences in description making could be a considerable approach for extracting rules for a CEP-based system. In one of these kinds of study [22] a new complex evidential quantum dynamical (CEQD) model is proposed to predict interference effects on human decision-making behaviors. Moreover, they proposed some functions based on Pignistic belief transformation which can be used in CEQD model for

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

removing interference effects. Experimental results show the effectiveness of the proposed method. This CEQD method presents a new perspective to study and explain the effects of involved interference in human decision-making behaviors.

Using artificial intelligence (AI) methods is one of the best approaches for setting rules of the CEP engine. Bayesian networks are one of the famous AI methods that can be used aligned with CEP. In [23], the authors proposed a complex predictive event processing method based on the evolution of the Bayesian networks. The Bayesian model is designed by an inference method based on the Gaussian mixed model and EM algorithm and based on event type and time. The evolving Bayesian network structure is supported by the mountaineering method. This system can continuously monitor the Bayesian network model and corrects it if new input information is not appropriate. The proposed method is evaluated in the field of road traffic. The total error is 12.12% and 7.78% for real and simulated data, respectively. While the least error rate among the other methods is 11.79% and 14.59% for real and simulated data, respectively. Sequence clustering is another AI approach that is used for rule generation in a CEP-based system. In [24] Lee et al. provide an automated sequential clustering-based rule generation method (SCARG) that can extract rules automatically by extracting the decision history of experts. It generates graphs based on sequential clustering and modeling. In addition, the proposed method is able to support automatic updating of the rules, regularly or occasionally. This system enables a self-compatible event processing system by combining the existing rule generation methods and the existing dynamic event processing systems. The proposed results show an increase of 19.32% in performance in comparison to the existing complex dynamic event processing methods. In [25] Roldán et al. proposed and implemented a smart architecture combining CEP and ML that is able to easily manage dynamic patterns to detect IoT security attacks. Dynamic patterns have some event characteristics that depend on values automatically calculated by a linear regression and support vector regression (SVR) prediction [26]. In this paper, by using MEdit4CEP-based tools [27], domain experts can graphically model these patterns, which are then automatically converted to Esper EPL code and deployed in the CEP engine at runtime. In order to validate the architecture, it is applied to a prototype IoT network built in a hospital to detect attacks by a malicious device. The results confirmed that this architecture works effectively when using a model that can be adapted to the context. Moreover, linear regression showed better results against SVR in terms of precision, recall and F1-score.

Fuzzy systems are one of the recent topics in AI researches. In some situations, it can merge with some processing systems like CEP for achieving better results. The main purpose of another research presented by Mehdiyev et al. [28] is to replace the manual identification of event patterns. Due to the uncertainty related to the sensor data, the fuzzy disordered rule induction (FURIA) algorithm has been used to identify the event patterns. In this paper, the method was implemented after selecting the relevant feature subset using the Elitist Pareto multipurpose evolution algorithm (ENORA) to enhance diversity. The results were compared with alternative machine learning methods and showed that the FURIA classifier combined with ENORA methodology classifies 98.39% of instances correctly. In [29] a method is proposed called a cost-aware, fault-tolerant and reliable strategy (CaFtR). This method used network resources for obtaining continuous and highly available CEP regardless of dynamic operator migrations under fuzzy environment. Moreover, a case study was provided to demonstrate the proposed method efficiency, and it was exploited in a StreamBase system application.

One of the approaches that can be more useful in terms of rule-setting and rule adaption in CEP-based systems, is rule-based learning. Because of the likelihood of rule-based learning and CEP systems, this AI method can be widely used for rule defining and adapting in a CEP system. In the research of Mehdiyev et al. [30], the aim is to provide a machine learning model to replace the manual identification of rule patterns. After a preprocessing step (dealing with missing values, data-related information, etc.), various rule-based machine learning methods were used to detect complex events. Promising results were obtained with high accuracy. Among the evaluated seven algorithms, the PART algorithm had the highest accuracy (about 93%).

Moreover, rule-based learning is more explainable than other machine learning methods like neural networks and, etc. In some cases, like health monitoring, it is necessary that the system has appropriate transparency and acceptable explainability. This makes domain experts aware of machine decisions, predictions, and functionality in order to investigate the reliability and supervise the system. Therefore, explainability is a significant issue in AI especially in contexts in which experts should be aware of the system. In [31] a survey is presented in order to test existing researches which are done in the context of explainable AI. The authors classify previous approaches in the field of explainability in machine learning and representing a definition from explainable machine learning. Furthermore, they presented a taxonomy from machine learning models which are explainable. Tjoa et al. [32] reviews the interpretability methods presented in various papers and categorizes them. These categories range from approaches that provide clearly explainable information to studies that provide complex methods. Moreover, the authors hope to provide a situation for exploiting these methods more than before using interpretable classifications in the field of health monitoring and making an acceptable vision of explainable approaches. They believe this would happen by considering these methods more than before in the medical context and using mathematics in advanced medical education. A comprehensive review about human-computer interaction (HCI) and explainability in intelligence health care was presented in [33, 34]. The main purpose of this paper [33] was to discover the point of intersection between HCI and AI. The understanding of Explainable Artificial Intelligence (XAI), which is the link between HCI and XAI, was obtained by reviewing the previous works in this study. The literature

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

review included topics identified in the previous studies (such as XAI and its domains, XAI main objectives, and XAI problems and challenges). Another major focus of this study was on the use of AI, HCI and XAI in healthcare. The survey also showed the XAI's shortcomings in healthcare, as well as its future potential. As a result, the literature shows that XAI is still a new topic in healthcare that needs to be further explored in the future.

Eventually, it seems that CEP accompanied by an artificial intelligence technique like rule-based learning is a reasonable package for an adaptive user modeling system for remote health monitoring. As it was reviewed in this section, machine learning methods can be a fair solution for coping with human errors in rule-setting in a CEP system, and especially for rules that are produced by rule-based learning are more similar to a CEP rule set and more explainable for domain expert.

3. CEP and Rule-based learning

3.1. Complex Event Processing

Complex event processing (CEP) [35] is a novel method to process a flow of incoming events in real-time. This method has the ability to process large volumes of generated data (such as data generated from a health monitoring system that includes multiple sensors and multiple users) compared to the traditional data processing methods. This operation is done by mapping data in the form of events [36].

3.1.1. Event

Different definitions have been found for "event". An event is a record of activity within the system. The processing is performed on this event for a specific purpose [35]. In other words, an event is a desirable and expected occurrence in a time unit [37]. In fact, a set of data at a specific time that represents a situation, especially in a period of time, is called an event. In [38] some attributes such as being meaningful per time unit, being real-time, and being atomic are mentioned for each event. "Being meaningful per time unit" expresses that every event which occurs in a specific time shows a particular concept. "Being real-time" means that the event instantly happens in a time unit. Ultimately, "being atomic" means that event either happened or does not happen in a time unit and there is no intermediate state. Events are divided into simple and complex events according to their characteristics.

3.1.1.1. Simple event

A micro event that is indistinguishable into smaller components and occurs in a unit of time is called a simple event [39]. For example, information about each sensor that is connected to the patients produces a simple event.

3.1.1.2. Complex events

A complex event is made from the aggregation of simple or complex events that are interconnected using special operators such as concatenation, aggregation, abstraction, or combination (disjunction, conjunction, and sequential combinations) [40]. A complex event uses other events to describe a situation and the action is required by the system in that situation according to the system description. Complex events are semantically more useful for making decisions in a system [2].

3.1.2. Rules

In the CEP method, special rules are used to detect complex events. A rule is a logical inference or pattern for recognizing complex events [41] consisting of events interconnected by operators. In [42] and [37] different types of syntax for defining a rule have been identified. Fig. 1 shows an example of these patterns known as "event-condition-action":

```
Rule rule_id, rule_name, rule_group, priority
ON event
IF condition
THEN action1, action 2, ..., action n
END
```

Fig. 1. An example of "event-condition-action" syntax [2]

In Fig. 1, "rule_id" and "rule_name" uniquely identify the information of the rule. "rule_group" indicates a group on which the rule is semantically dependent on. "Condition" is a specific combination of conditions defined by the user. "Action" shows the user's intended action which happens when a complex event occurs, for example, sending a warning.

3.2. Rule-based Learning

In methods that exploit the CEP, all the rules are defined by the domain expert, which would lead to human errors in setting rules [30]. An approach is suggested based on using a rule-based classifier instead of the domain expert to extract rules from the previous data. Also, this AI approach can be used along with domain expert's opinions due to the explainability of the rule-based learning approaches. In some studies [43, 44] a chart was presented that illustrated the tradeoff between explainability and performance of some famous machine learning methods. This chart was presented in Fig. 2.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

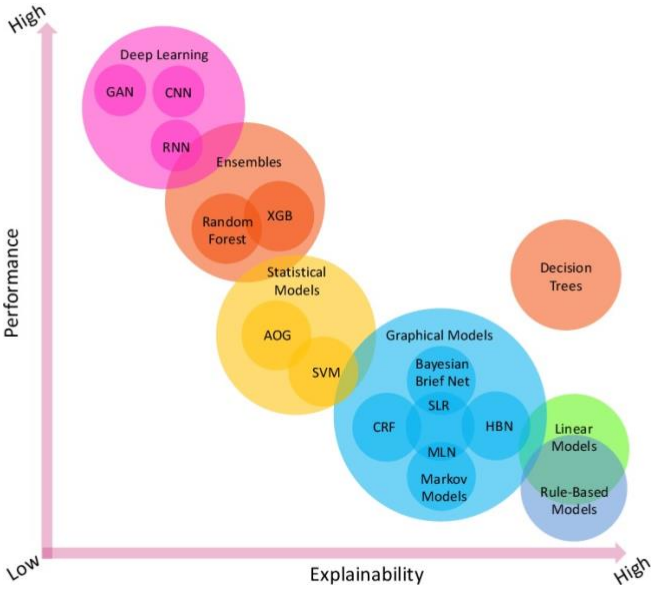


Fig. 2. Model explainability vs. model performance for widely used machine learning and deep leaning algorithms [43, 44]

As it is shown in Fig. 2, the rule-based learning and decision tree algorithm (which is the base of rule-based learning) are more explainable than other learning methods.

Rule-based classification refers to any form of classification method that uses an if-then rule-based structure to predict the final class. In general, rule-based learning refers to any method of learning that performs actions such as storage, manipulation, or practice to identify, learn, or evolve rules [30]. The main property of rule-based machine learning is the identification and application of a set of rules that represent knowledge within a system. This is in contrast to other machine learning methods, which usually specify a single model that can be used in general to predict the class of each instance.

As mentioned earlier, the nature of the CEP method is based on the rules defined by the domain expert, which leads to the problem of cold start. Rule-based classifiers also eventually provide a set of if-then rules to identify classes. As a result, it would be very practical and reasonable that the set of rules identified by these classifiers are used as rules in the rules repository of a CEP system.

In addition, in some areas, such as intelligent health monitoring systems, it is necessary to examine the opinions of the domain expert about the designed system, its diagnosis, and how it works. Thus, rule-based classifiers that use a set of rules to identify a class, can easily be presented to domain experts. Moreover, the functionality of the rule's conditions can be checked by domain experts.

Although some classifiers such as neural networks, support vector machines, etc., have presented good results in data classification, they are not appropriate in the present work from two perspectives. Firstly, these classification algorithms detect new sample classes based on a trained model. This model cannot be used in extracting CEP engine required rules. Secondly, in some context we cannot ignore domain experts easily. Thus, it is difficult for domain expert to understand the main knowledge of model and we cannot seek their opinions. In the following, we will introduce some of the rule-based classifiers.

3.2.1. OneR classifier

The OneR rule-based classifier, abbreviated to single-rule [45], is one of the most widely used rule-based classifiers and is a viable alternative to more complex classifiers. This classifier uses a single-level decision tree to classify samples based on the value of individual properties. Unlike other classifiers that use entropy criteria to classify samples, the OneR classifier uses an optimized error rate from training data. The final algorithm develops a rule for each individual predictor in the training data and specifies a rule with the lowest error rate.

3.2.2. PART classifier

In [46], Frank and Witten proposed an algorithm based on partial decision trees (PART). PART is actually a decision list generation algorithm and contains a set of C4.5 decision trees [47]. In the PART algorithm, a combination of the C4.5 decision tree and the RIPPER algorithm is used. Unlike other rule-based classification algorithms, the PART algorithm does not use a general rule adaption when extracting rules. Instead, it uses a separate and deceptive approach. This makes the algorithm simple and fast. An important feature of this algorithm is its simplicity and efficiency. In addition, PART generates more rules than most rule-based algorithms.

3.2.3. RIPPER classifier

The JRip algorithm, known as repeated incremental pruning to produce error reduction (RIPPER), was proposed by Cohen [48]. The JRip algorithm uses a special method in preparing rules. It is a combination of dependency rules with reduction error pruning (REP), which is a common and effective method in decision tree algorithms. In the REP method, for rule-based algorithms, during the rule extraction operation, the training data is divided into two parts. One of these two parts is called the growth set and the other part is called the pruning set. First of all, a basic set of rules is created that is empty. This set of rules is formed over time, using instances of growth sets and with the help of heuristic methods. As the ruleset grows using the growth set, the instances in the pruning dataset are applied to optimize the rule set's functional progress. This is done by pruning (selecting items with the largest error reduction).

4. User behavior modeling in adaptive and personalized CEP-based platform

In this section, first, the novelties of the platform are presented briefly. After that, a schema of the proposed system will be provided. Then, we will review the dataset that used in the system implementation process. In the following, we will discuss the methods and tools used in designing the innovations considered in the system design process. Finally, we present the results obtained from each section and the analysis of these results.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

4.1. The novelties of the proposed adaptive CEP platform

In this research the proposed CEP platform is an adaptive and personalized system which are novelties of this research. The system is updated periodically in pre-set period of the time. The predefined time period is set by experts according to the context. The update has two benefits for the platform. Firstly, it provides adaptivity. It means that if conditions in the context change, the rules will be adjusted accordingly to new conditions and the platform can adapt itself to them. Additionally, in some contexts, users create various types of data. The platform should be able to trigger personalized answer considering single user requirements. For example, increasing heart rate to a specific level could be dangerous for a patient but not for another one. Hence, it would be very beneficial to personalize rules or rule thresholds for each user. We have implemented this idea by saving the input data in a database and then applying a rule-based learning for extracting initial rules for CEP on the saved (historical) data. In the following, the proposed adaptive CEP platform and the novelties in developing of this platform, are presented in detail.

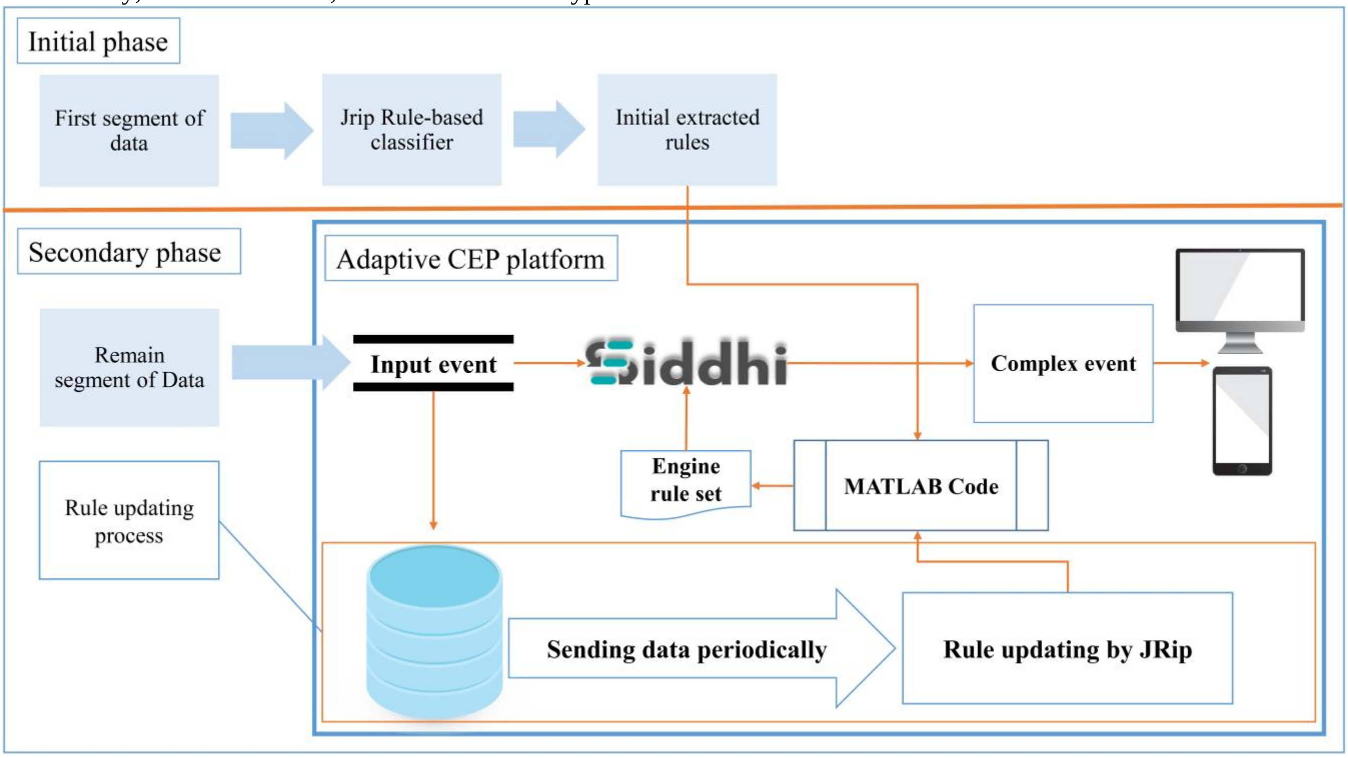


Fig. 3. The proposed adaptive CEP platform

4.2. Schema of the proposed adaptive CEP platform

The proposed adaptive CEP platform has been shown in Fig. 3. In general, this platform consists of two phases: initial and secondary phases. In the initial phase rule-based classifier is applied on a segment of data and initial rules are extracted. After that, extracted rules are changed to a CEP engine rule defining language and CEP is ready to use. In secondary phase the rest of data is imported to platform and complex events are detected by initial rules and these rules are updated periodically.

According to Fig. 3, in adaptive CEP platform part, the user's data is recorded by existing sensors and is entered into the system as the input data. These sensors generally are connected to the users and can produce a wide range of data depending on the context, such as activity data (usually mounted on limbs and contain movement in three axes), sensors that receive vital data (including blood pressure, respiration rate, body temperature, heart rate, brain signals). Time in the form of different hours of the day is always recorded as input. The notable point in this section is that historical and pre-recorded data (gender, history of pre-existing diseases or use of a particular drug, etc.) are not considered.

In the processing part of the system, there is a CEP engine that uses a set of rules to distinguish the complex event pattern from the input data. In this process, which is performed in real-time, if a complex event is detected from the input data, an appropriate response will be generated. The rest of the data, which does not make a specific meaning, passes through the system without making any changes to the system.

Furthermore, we can use the received data from the engine for rule adaption in accordance with the new conditions and providing personalized rule adaption for different users, and updating the rules set. For this reason, one of the steps could be including a memory in the path of the event processing engine. After that, this adaption is done by using a rule-based classifier method and setting the previous rules with the new

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

intervals. Thus, the rules can be adjusted to the new operating conditions of the system or the new conditions of each user. In the mentioned memory, the data entered the system and the occurred time of the event were stored. Moreover, the data was labelled based on the condition at the time it was occurred. This data was later used for evaluating the proposed system. After a while (based on predefined duration of time that was determined by experts) a rule-based classifier was applied automatically on the stored historical data (data from the time that systems started working until now) which was stored in the database and rules were modified based on new conditions and users' behaviors. Fig. 4 illustrates a schema of CEP engine rule set before and after rule updating.

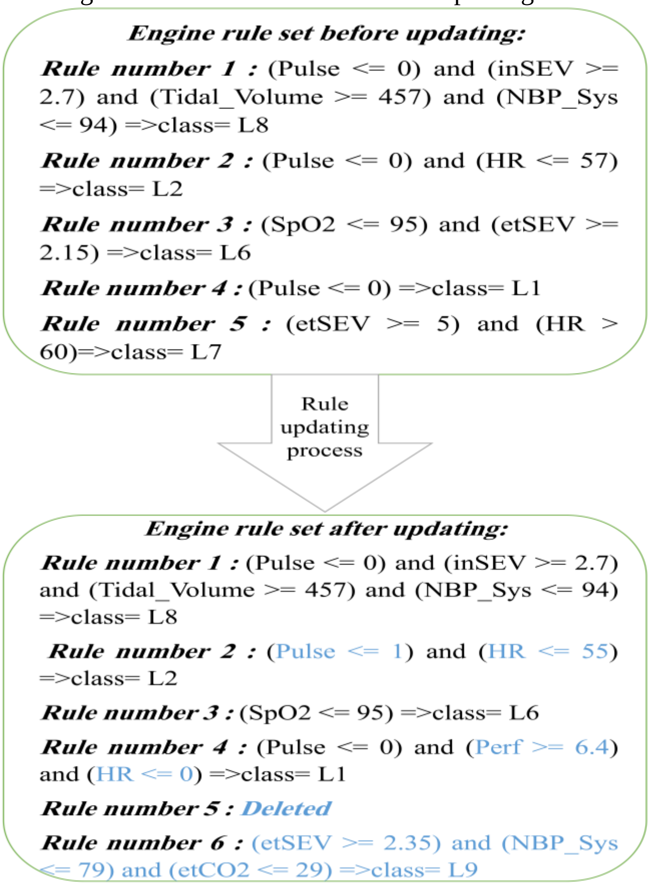


Fig. 4. A schema of CEP engine rule set before and after rule updating.

As it is illustrated in Fig. 4, the engine rule set updating process consists of these parts: the threshold(s) of existed rule(s) is(are) updated (like Rule number 2), the extra attribute(s) of existed rule(s) is(are) deleted (like Rule number 3), the new attribute(s) is added to the existed rule(s) (like Rule number 4), the non-functional rule(s) is(are) deleted (like Rule number 5) and new rule(s) is(are) added (like Rule number 6).

In our platform, the adaption or personalization methods are applied periodically and as a background parallel process to the real-time event processing. It means that the CEP processes the input events while the adaption or personalization process are running (according to Figure 3). During the engine rule-set updating, the system may be down for a while just like other kinds of system. This interruption is related to the number of rules. However, the mentioned interruption is not unbearable at all and since it is happened periodically, it does not have any significant effects on the whole platform latency. The system downtime can also be totally avoided by using two servers which the platform is existed in both of them and the output of the rule updating process from a server can update the engine rule set of another server and while the rule set in one of the servers is updating, the other one can response to the requests; then, we can switch between them.

In the output section, if a complex event is detected by the engine, the corresponding output is generated according to the rule. This output can be in different data formats that can be reopened and displayed in different ways. The system also has the ability to send output to various sources. It can be a warning for the patient or the patient's relatives or medical staff. If this system is connected to the other networks, the warning message can be sent in the form of SMS, e-mail, etc.

In Siddhi engine, all the input/output flow and the data type of each flow should be defined at the beginning. In the experiment part of this study, we will define every single data set feature as an input flow and every target class as an output flow. After that, we will add every row of data in every one second to the CEP engine. It means that every row is considered as an event happened at a second. Therefore, there is not any necessity to consider a specific data transforming process in this step. In the real world and in the health monitoring context, the produced data of every real-time sensor like blood pressure or heart rate screen devices are connected to the CEP engine as a flow. If a complex event is detected based on an existing rule, an appropriate action which is already defined in the mentioned rule, will be triggered.

In addition, the pseudocode of the proposed adaptive CEP platform is presented in Fig. 5. All the process of adaptive CEP platform is shown in this figure in detail. Moreover, there are two functions in this code for rule adaption for the whole platform and rule personalization for each user. These two functions will be introduced in the following with more detail.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

```
PROGRAM adaptiveCEP
    CREATE inputData
    IMPORT inputData as flow
    INSERT INTO database
    APPLY JRip on inputData
    EXTRACT rules from JRip
    SET every rule threshold interval as [a, b]
    SET CEPEngine
    ADD rules INTO CEPEngineRuleSet
    START CEPEngine
    CREATE dataFlow
    IMPORT dataFlow as flow
    FOR all rules in CEPEngineRuleSet
        COMPARE flow with every rule in CEPEngineRuleSet
        IF flow parameters in rule threshold interval [a, b]
            TRIGGER ruleResponse
        ENDIF
    ENDFOR
    CREATE variable adaptionTime
    ASK the expert for a number
    READ input into adaptionTime
    IF today equal time
        CALL ruleAdaption ()
    ENDIF
    CREATE variable personalizationTime
    ASK the expert for a number
    READ input into personalizationTime
    IF today equal time
        CALL rulePersonalization ()
    ENDIF

END
```

Fig. 5. The pseudocode of the proposed adaptive CEP platform

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

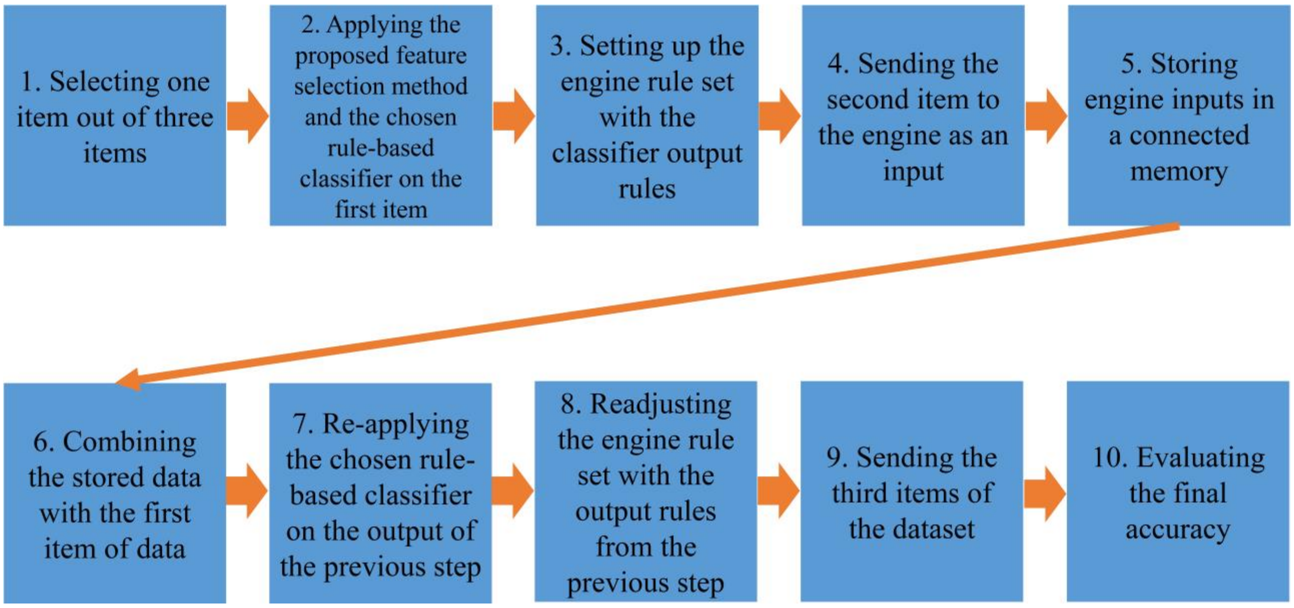


Fig. 6. The flowchart of the rule adaption using the previous data

As it is explained in WSO2 documentation about the input mapping data [49], if an event with null or empty attributes is occurred and the type of these null or empty attributes are not string, an error will be raised and the event will be ignored. However, in this study, we used one of the Siddhi's features that detects null or empty attributes of an event using "is null" operator [50]. In this platform, if a null or empty attribute in an event is detected, the platform uses the last existed value of this attribute until the sensor send new value again. This practice prevents platform efficiency reduction. Moreover, the last user condition is considered. Additionally, in our experiment we handled the missing values of both data sets with the mentioned process and if any feature of data set was null or empty, the platform filled it with the last existing value. Please note this practice does not prevent us to seek domain experts advise whether filling missing values with most recent values is the best choice in each domain or not.

4.3. Rule adaption by using previous data for the whole system

In this part, an approach for rule adaption is proposed. It happens in order to adapt whole rules in a rule set for coping with new situations or new conditions in context by using historical data. The flow chart of the rule adaption process with previous data is shown in Fig. 6.

According to the flowchart in Fig. 6, first of all, three items of data are selected (each representing an operation or a person's movements). On one of these items, the proposed feature selection method is applied (which will be explained in the next section). The proposed feature selection method is applied when the size of data is large and the processing time is too long. Then, on the new data set, the rule-based learning algorithm is implemented and the output rules set is defined along with the definition of input and output of the CEP engine.

After the engine is ready, another selected item from the three items is entered into the engine and its samples are saved in the database. In the second step, the rule adaption process (creating new rules and improving the intervals of the previous rules) is proposed. Hence, the first item that is under the rule-based learning algorithm is combined with the second item entered into the engine and stored in the defined database. Then, the learning algorithm is re-applied for adjusting the rules set with the new rules and entered the final selected data into the engine. This process can run continually in a certain period of time in order to adapt the rules based on the system conditions.

This process is applied once again without the rule adaption process (eliminating the second to eighth steps of the flow chart in Fig. 6), and the basic rules set of the engine is used for the last data item.

As it is mentioned before, this rule adaption in our proposed platform is a function. The pseudocode of this function is illustrated in Fig. 7. As it is shown in Fig. 4, this function is called periodically based on the domain expert determined time for the whole rules in CEP rule repository and updates them and boosts their thresholds based on the previous data stored in database.

```
FUNCTION ruleAdaption ()
    1.  SELECT allData from database
    2.  APPLY JRip on allData
    3.  EXTRACT newAdaptedRules from JRip
    4.  SET every newAdaptedRule threshold interval as [a, b]
    5.  UPDATE rules threshold interval [a, b] in CEPEngineRuleSet
        based on newAdaptedRule threshold interval [a, b]
    6.  END
```


AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

Fig. 7. The pseudocode of the rule adaption function

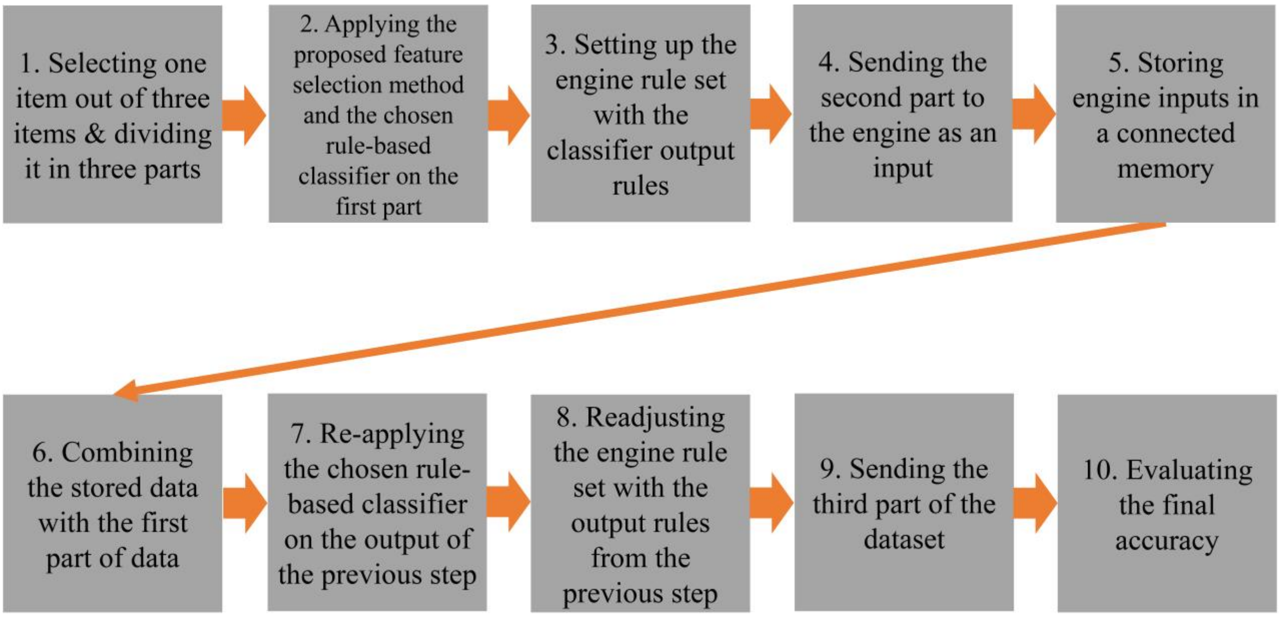


Fig. 8. The flowchart of the personalized rule adaption using the previous data

4.4. Personalized rule adaption by using each person's previous data

In the following part, a personalized rule adaption is proposed in order to personalize rules based on each user condition. For instance, some conditions like age or surgery experience can affect some medical situation that a user may cope with it. Hence, personalized rule adaption is a necessary part of our system. The flowchart of the rules' personalization using each person's previous data has been shown in Fig. 8.

According to Fig. 8, in this process, a personalization rule adaption method on the data is tested, similar to the previous method. For this purpose, this time, each data item that is related to a user or surgery, is divided into three equal parts. After applying the feature selection method (if the data is large), a subset of the appropriate features is chosen. Now, the rule-based classifier is applied to the first part of the data. The output rules are entered into the CEP engine and the engine rules set is modified with rule-based classifier rules. Then, the second part is entered into the engine and the data is stored in the database. After that, the first data part (which was used to make the rules) and the second data part (which is in the database embedded in the engine) are merged and the classification algorithm is applied to it again.

In the end, with the help of the output rules, the rules are updated in the rules set and the last part is applied. After that, the third part is applied once without the personalized rule adaption approach (by eliminating the second to eighth steps of the flow chart in Fig. 8). Then, the above process is repeated for the other two cases. The personalized rule adaption method can apply continuously in order to adapt system rules based on the new conditions of each user.

As it is mentioned before, this rule personalization in our proposed platform is a function. The pseudocode of this function is illustrated in Fig. 9. As it is shown in Fig. 4, this function is called periodically based on the domain expert determined time for the whole rules which is used for a specific user in CEP rule repository and updates them and boosts their thresholds based on the previous data of the user which is stored in database.

```
FUNCTION rulePersonalization ()
    1.  SELECT userData from database WHERE every user
    2.  APPLY JRip on userData
    3.  EXTRACT newUserRules from JRip
    4.  SET every newUserRule threshold interval as   [a, b]
    5.  UPDATE threshold interval rules [a, b] in CEPEngineRuleSet
        based on newUserRule threshold interval [a, b]
    6.  END
```

Fig. 9. The pseudocode of the rule personalization function

5. Evaluation results

In this section we are going to explain and evaluate the results of implementing the proposed approaches on two datasets. The datasets will be discussed in the following sub-sections. Then, the results of applying the chosen rule-based classifiers on the mentioned datasets will be presented and evaluated based on their accuracy and the produced rules number. After choosing the best rule-based classifier, we explain the results of exploiting the best rules set in the CEP engine according to our proposed adaptation and personalization ideas. Moreover, we will present and

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

compare the results of two case studies. One of them exploited our proposed method and another one did not. At the end, we will present a table for comparing our proposed method with similar methods in this field.

It should be notable that for evaluating our proposed platform, we designed a simulated experiment with two pre-stored data sets (Hospital dataset and Activity dataset). We did not develop the platform for a real-world case due to two reasons. First, the cost of collecting data flow from real users via devices such as sensors is very high and second, it needs lots of recording devices and a group of participants. However, the platform can be utilized for real-world scenarios. It should be noted that both exploited data sets are based on real-world scenarios.

5.1. Datasets properties

As it is explained before, in this study, we are feeding our CEP engine with existing data sets, due to the high cost and high complexity of designing and preparing sensors. We found two datasets for this case study. One of them is hospital dataset and the other one is activity dataset. The exploited datasets will be introduced in flowing.

5.1.1. Hospital dataset

The present data set [51] includes clinical anesthesia monitoring data from all surgical cases in which patients underwent anesthesia and was recorded at the Royal Hospital of Adelaide, Australia. This dataset includes a wide range of vital signs variables that are commonly examined during surgery.

This dataset includes vital signs of anesthetized and under surgical procedure patients, which includes 32 surgeries, of which 25 are general and 7 are special. The duration of these surgeries is between 13 minutes to 5 hours and the average length of each operation is about 105 minutes. Most fields include data from an electrocardiogram (for ECG), pulse oximeter (for arterial oxygen saturation), capnography monitor (for monitoring and instantaneous measurement of end-expiratory carbon dioxide), non-invasive blood pressure monitor, airflow, and pressure monitor. In some cases, a spirometer (to measure the volume of inhaled and exhaled air), an electroencephalogram (to record an electroencephalogram), and an arterial blood pressure monitor have been used. We use those files with specified numerical values of sensors with intervals of one second. The reason for this choice is that the one-second interval between samples is a good deadline to show the real-time nature of the proposed system. Moreover, this file is numeric which is suitable for the monitoring process.

This dataset contains many sensors data that make multiple target classes. However, the surgery cases chosen for evaluating our systems, consist of the following classes. “The electrocardiograph does not work”, “normal heart rate is high”, “blood pressure is not correct”, “normal conditions”, “blood oxygen sensor is off”, “AGM (related to breathing sensors) alarm is not supported”, cannot analyze ST index “blood oxygen concentration is incorrect”, “final CO_2 concentration is high”, “N_2O (nitrous oxide) concentration is high”, “high heart rate”, “high blood pressure”, “high concentration of anesthetic” and etc.

5.1.2. Activity data set

This dataset was provided by Banos et al. [52, 53] in Spain. In this data set, activity data and vital signs have been recorded and stored using an Android-based platform and a number of sensors. This dataset contains movement information and vital signs of 10 users recorded during 12 different movements. These classes consist of “standing”, “sitting and resting”, “drawing”, “walking”, “climbing stairs”, “stretching”, “hand exercise”, “bending”, “cycling”, “jogging” and “jumping forward and backward”.

Wearable sensors have been used to record this data set. These sensors measure the acceleration, rotation speed, and orientation of the magnetic field. The chest sensor also offers two ECG metrics. This data can be used for basic heart monitoring, screening for various arthritis, or examining the effects of exercise on the ECG.

In both datasets, a sample of the user’s activity/patients’ conditions is recorded per second which is a proper deadline for a real-time system. Therefore, both datasets have the ability to be used in a real-time intelligent health monitoring system and have been exploited for the evaluation experiments in this work.

It should be noted that we used two datasets, separately. In fact, we designed two different case studies with two different datasets to evaluate our proposed platform. Firstly, we defined rules with one part of the hospital dataset. After that, we evaluated our proposed methods for adaptation and personalization (as described in Section 4) with another part of the dataset. The above-mentioned method was repeated for the activity dataset, too. This time, one part of the activity data was used to define the rules and the other part was used to evaluate the proposed methods for adaptation and personalization (as mentioned in Section 4). All steps of the above process can be applied to any other compatible datasets such as stock trading and etc.

5.2. Evaluating the system on the hospital data

In this section we present the evaluation on the hospital data.

5.2.1. Implementing process

In this section, we present the experimental results of evaluating the proposed health monitoring system on the hospital data. In summary, we first selected some cases, each of which represents a surgical procedure, from the dataset. Then, we applied the rule-based classifiers to one of them and compared their results with each other. We entered the output rules of the selected rule-based algorithm into the event processing engine and import the next selected item as an input to the engine. Once we have combined the new inputs to the engine with the previous ones, and after applying the rule-based algorithm, we update the rules set of the engine

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

based on the new output of the algorithm. This time, we enter the last item into the engine. This process is done one time again without doing the mentioned updating process. Therefore, the situation with updating process can be compared to the situation without updating process. In the following, we will discuss the above case in more detail.

In this paper for implementation the experiments, we used a computer with an Intel CORE i7 CPU and a 6GB Memory and windows 10 operating system.

5.2.2. Choosing some cases of data

As mentioned previously in the description of the hospital dataset, this dataset included 25 general surgeries and 7 special surgeries. In these surgeries, in some cases, a number of sensors are turned off and no data is stored for them. This has resulted in different outputs. Moreover, each case may differ from the other cases in the number of feature values. Therefore, we have been trying to select similar samples. Therefore, we selected three similar surgeries out of 25 general surgeries. Each of these three items includes about 15,000 data samples. Each of these three items contains 13 different labels that make our problem a 13-class classification problem.

5.2.3. Rule-based learning algorithms

As mentioned earlier, the rule-based learning algorithms which are used to generate a ruleset, are useful in two parts. They are useful firstly, inputting the rules that are extracted from the algorithm in the CEP engine and secondly, in showing a proper abstraction of how the system works. This abstraction can be provided to medical professionals and received objective feedback from them.

In this paper, some of the most popular rule-based learning algorithms available in Weka [54] have been exploited. As mentioned, in [30] a number of these algorithms are compared with each other in the terms of accuracy, error rate, and so on. One of the disadvantages of the work proposed in [30] is the lack of reporting on training time and the number of hardware resources required for these algorithms. In the present work, by re-comparing these algorithms, it was concluded that basically some of these algorithms, do not have a significant difference in accuracy with faster algorithms. However, some of them require a lot of hardware resources and have a very time-consuming learning process. Finally, after examining different rule-based algorithms, three algorithms were selected that performed better in terms of speed and hardware resources. These three algorithms include PART, JRip, and OneR algorithms. In [30] it was also confirmed that PART and JRip had the highest accuracy.

5.2.3.1. Applying rule-based algorithms and their results

After selecting these three algorithms and 15,000 samples, the selected dataset samples were uploaded in Weka software and all three mentioned rule-based classifiers were applied to them. The CPU and memory consumption of rule-based classifiers for Hospital dataset are presented in Table 1. In addition, Fig. 10 shows the difference between these three classifiers based on accuracy in classifying data.

Table 1. CPU and memory consumption in rule-based classifiers for Hospital dataset

Classifier	Memory consumption	CPU consumption
PART	52 MB	18%
JRip	99 MB	15.23%
OneR	55 MB	20.25%

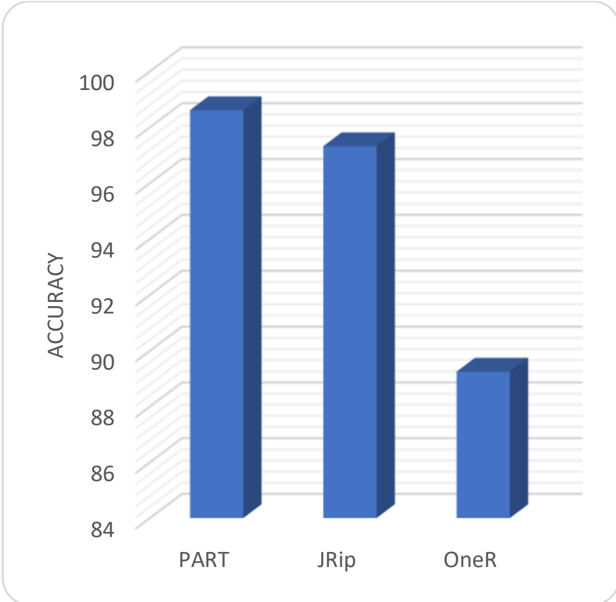


Fig. 10. Comparing the accuracy of the three rule-based classifiers on the hospital data set

5.2.3.2. Accuracy Comparison of the three rule-based algorithms for hospital data

As expected, the OneR algorithm is less accurate than the PART and JRip classifiers. This is because of using only one feature for producing rules in this classifier. But the other two algorithms do not differ much in terms of accuracy. However, the PART classifier performs slightly more accurately than JRip. This may be due to the high similarity of the two algorithms in structure. Since the PART classifier is actually a combination of the JRip classifier and the partial decision trees, the final algorithm performs a little better than JRip in terms of accuracy. As a result, due to the high sensitivity of the medical field on the true detection rate, we select the PART and the JRip algorithms due to their better accuracy.

Another criterion that has been considered in our work but not in [30] is the number of extracted rules from the classifier. In this work, we compared two classifiers with the highest accuracy in terms of the number of rules. In [30], the authors only intended to compare the classification methods with each other and they did not intend to use the output rules in an event processing engine. Thus, the number of rules is not important for them. But in our work, due to our intention to use the output rules of the classifier in the CEP engine, the number of rules is also a significant criterion for us. Fig. 11 shows the comparison results of the two algorithms in terms of the number of output rules.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

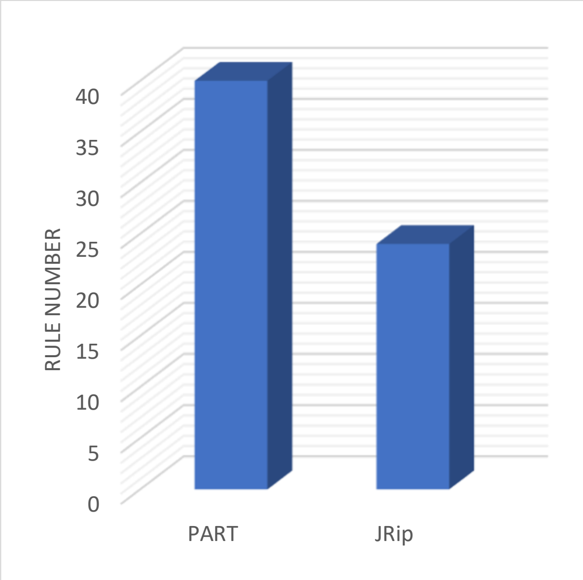


Fig. 11. Comparing the number of rules generated by two rule-based learning algorithms

According to Fig. 11, the JRip algorithm produces fewer rules (about 40%) in comparison to the PART algorithm. One of the innovations in designing the PART algorithm is the elimination of the optimization step. Of course, it has enhanced the final accuracy and at the same time increases the speed of the learning process. However, the number of the generated rules is not optimal. This difference in the number of rules is more profound in larger cases, and sometimes the PART algorithm generates about three times more rules than the JRip algorithm. Yao et al. in [2] showed that fewer rules number can reduce the processing time. Therefore, the JRip algorithm is our final choice in this work.

5.2.4. Implementing and setting up the CEP engine

CEP systems typically consist of a user-defined processing engine and a repository of rules. These rules are used to identify complex events from input sources. CEP engine associates with the rules repository. After detecting a complex event based on rules which are in the repository, the CEP engine triggers an appropriate response according to the selected rule. These include WSO2 from Red Hat [55], Esper from EsperTech [56], and Drools Fusion engine from Drools [57].

Most of CEP-based platforms use a server-client system. The CEP engine, most of the time, is set on a server and not a gateway device. Client devices would be used for sending data to the server for processing and getting the response. This is the case for many companies such as Uber, eBay, PayPal to name a few. Processing a huge flow of input data and triggering actions need powerful hardware that may not be accessible for most of the users. In addition, for monitoring the system and using data for purposes such as rule adaption and personalization, it would be practical to store data in the server.

Meanwhile, a CEP engine and information flow processing called Siddhi [58], which uses as core engine in the WSO2 platform, has become widely used in recent years. Moreover, businesses such as Uber use this platform. The advantages of this platform include easy and effective communication with various input and output ports such as files, emails, data streaming services such as Spark in various formats of JSON, CSV, etc. In addition, the platform can use non-relational databases (NoSQL) such as MongoDB to store previous data and use it for various purposes, including updating rules.

CEP engines differ from each other based on differences in their functional features. In this paper, we used "Siddhi" which is one of the strong processing engines, according to its high usability and user-friendly interfaces.

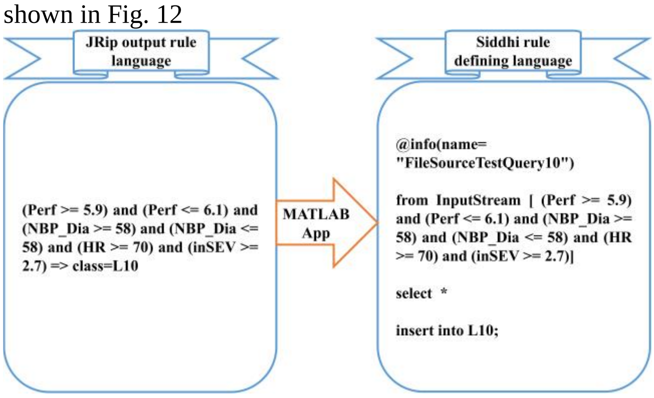
In these cases, our only action is to warn the hospital staff that an unusual situation has occurred. Additionally, the system has to inform them of the type of abnormal situation. In fact, in the hospital data, we have a normal situation and a number of abnormal situations. The engine detects these normal and abnormal conditions based on the input data model in the form of events. It notifies an abnormality to the hospital staff in the form of an alert message.

In this work, the Siddhi platform is used. To use this platform, we entered the output rules of the classification algorithm into the platform engine rules set, and also used a database to store the previous data to take advantage of them.

It should be noted that CEP is just a processing system and does not classify input data or predict a target class. It only distinguishes complex events based on predefined rules. Therefore, the accuracy of CEP directly depends on the accuracy of rules. Since, a rule-based learning (JRip) is used in this paper for rule extraction, the accuracy of CEP is equal to the accuracy of JRip algorithm.

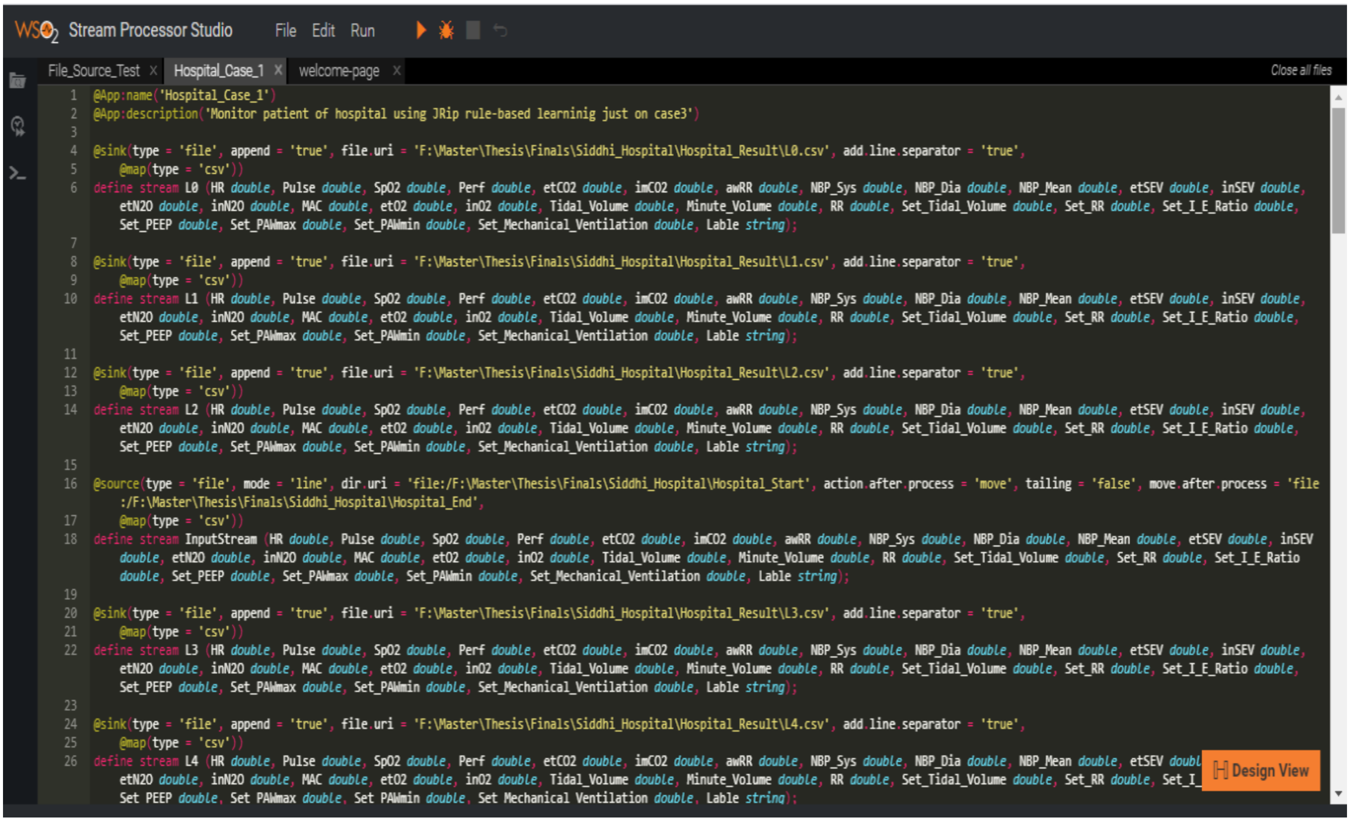
Additionally, we did not use any extension or function from Siddhi for implementing. We implemented all parts, separately, because we decided to examine our novelties and evaluate the efficiency of the updating process. However, we think Siddhi has many facilities for implementing the whole platform altogether.

The grammar of the output rules of the JRip algorithm is different from the grammar of the engine rules. Thus, we prepared an application under MATLAB to convert automatically the output rules to the standard rules defining language of the engine. An example of the rules mapping is shown in Fig. 12



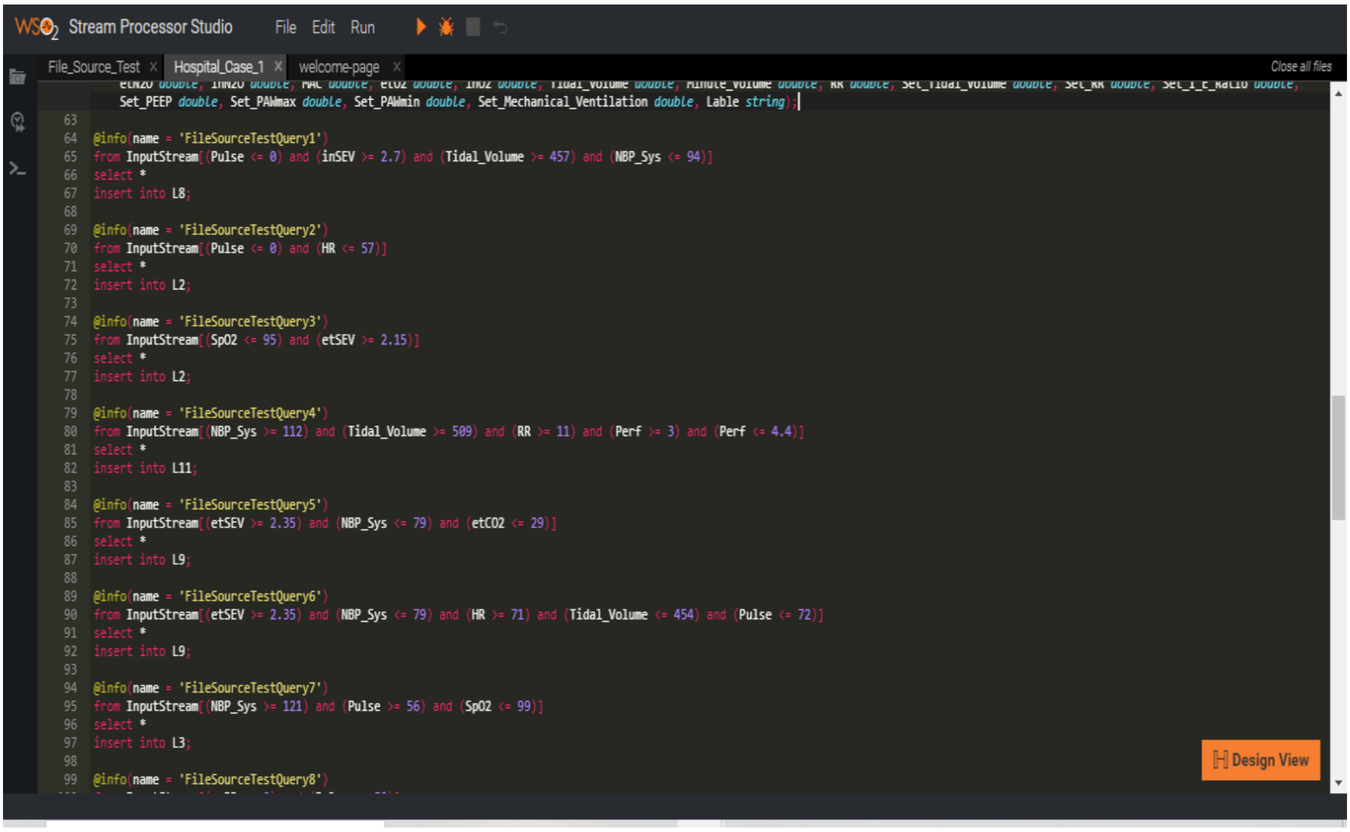
AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

Fig. 12. The output rule of JRip in comparison to the accepted rule format of the same rule in the engine



(a) Some samples of data flow on hospital dataset

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421



(b) Some samples of rules and their action on hospital dataset

Fig. 13. Siddhi environment for defining data flows and rules

In order to make our proposed system ready to use, we first defined the input stream which includes all the available features of the data set. After that, we defined the output streams (one output stream for each label). Then, we import the rules obtained from the rule-based learning algorithm to the MATLAB-based program for converting them into the engine acceptable rules. In the following, we set up the rule repository of the engine, and finally, our CEP-based system is ready to use.

Fig. 13 illustrate the Siddhi environment for defining rules and data flows. In Fig. 13 (a) some samples of defined data flows and in Fig. 13 (b) some samples of defined rules and their action that were extracted from JRip and added into Siddhi engine rule set are shown. These rules are related to hospital dataset. In top of figure needed data flows are defined. Then in bottom of picture the queries that represent the rules are defined. These rules and their threshold are extracted from JRip rule-based learning. As mentioned before these extracted rules are changed to event processing language (EPL) by a MATLAB program that engine can understand.

5.2.5. *Results of implementing of general rule using previous data*

The results of this experiment are illustrated in Fig. 14.

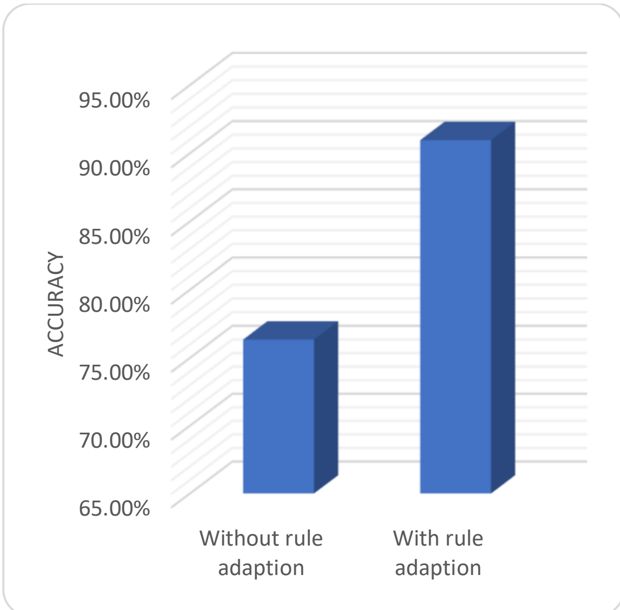


Fig. 14. Comparison of system output accuracy, with and without proposed rule adaption method

As Fig. 14 shows, updating the rules has improved the accuracy by about 15%. The updating process in all areas of computer science generally increases the performance, and

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

complex events processing is no exception. Especially, in the areas such as health in which the situation is continuously changing. In conclusion, the best way to use event processing engines in various fields is to constantly update the rules and create new rules or improve the intervals of previous rules.

5.2.6. Results of implementing personalization of rules using the previous data of each person

The results of this experiment can be seen in Fig. 15.

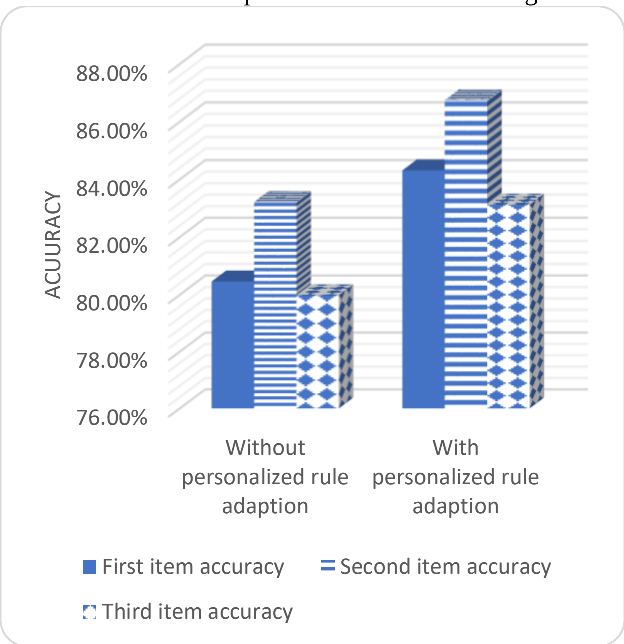


Fig. 15. Comparison of output accuracy of the system, with and without the proposed personalized rule adaption

As Fig. 15 shows, about three to five percent accuracy improvement has been achieved. However, according to the sample size in each case, there is a possibility of enhancing this improvement. Most likely it depends on the application and context of the data that is being processed and monitored. Also, the timing of this improvement (updating rules) can be changed by the user (in cases where the user has the authority, such as activity detection) or by the medical staff (in cases such as a hospital or home monitoring).

5.3. Evaluating the system on the activity data

In this section we present the evaluation on the activity data.

5.3.1. Implementing process

In this section, we present the experimental results of implementing the proposed health monitoring system on the activity data. This implementation is very similar to the implementation of the proposed system on the hospital data and differs in only one aspect. In brief, we first selected a number of items from the data set. Each item of the data set represents the activity information of a person and contains about 100,000 samples. In this section, due to the large volume of data, we first applied a feature selection method to the selected items, and then, we applied the rule-based algorithms. Again here, the whole process is done one time again without doing the mentioned updating process. Therefore, the situation with updating process can be compared to the situation without updating process. In the following, we will discuss the above case in more detail.

As we mentioned before, for implementation, we used from a computer with an Intel CORE i7 CPU and a 6GB Memory and windows 10 operating system.

5.3.2. Selecting items from the data set

As mentioned earlier, the activity data set include the activity test and heart rate of 10 people who have done different movements. These 10 items are not different in the terms of features and labels, and in all 10 cases, all sensors are used, and there are 12 activity labels in every 10 cases. Also, the number of recorded samples from each person is between 100,000 to 160,000 on average.

The same as the hospital data, in this data, we decided to select some of the 10 cases in order to evaluate our proposed system. We selected three cases of these 10 cases. Each case contains about 100,000 to 105,000 data samples and 12 different labels that makes our problem a 12-class classification problem.

5.3.3. Rule-based learning

In this section, as in the previous data, rule-based algorithms in Weka are used. We decided to compare these algorithms in terms of their accuracy and the OneR, PART, and JRip algorithms as our final choices,

However, due to a large number of features and samples in the present data (which is about 100 to 105 thousand samples with 24 features), compared with the previous data, we have to use feature selection methods. Ultimately, this feature selection process is one of the benefits of our system that users can use or ignore, due to the application field.

We used three approaches for choosing feature selection methods. One of them is using one of the well-known feature selection methods. The second one is proposing a feature selection method that is appropriate according to the type of the data and the last one is using a combination of different feature selection methods. After obtaining the selected features using these approaches, we applied the mentioned rule-based algorithms to them and evaluated them in terms of accuracy, speed, and the number of rules. The evaluation results of each of these feature selection approaches will be described in the following sub-section.

5.3.3.1. Methods based on evaluation of subsets results

These methods evaluate subsets of features. Our two chosen methods are correlation-based feature subset evaluator [59] and consistency subset evaluator [60]. At first, to compare these two methods, we evaluated each of these two algorithms in Weka with the best first search method [54] on one case of activity data. The result showed that the feature selection method based on a correlation-based subset evaluator and was accompanied by the best first search method provided us a subset with 18 features out of 24 features. The consistency evaluator selected a subset of 7

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

features out of 24 features as output. After applying three rule-based learning algorithms, the evaluation results are illustrated in Table 2.

Table 2. Comparison of two subset evaluations in terms of subset size and accuracy of rule-based algorithms

subset evaluations	Subset Size	OneR accuracy	PART accuracy	JRip accuracy
Correlation-based Subset Evaluator	18	73.32%	90.11%	90.23%
Consistency Subset Evaluator	7	76.12%	91.25%	89.96%

As it is shown in Table 2, there is no specific difference between these subset evaluating methods in terms of accuracy. However, the significant difference between them is in the size of the selected subset. The consistency subset evaluator method has produced the smallest and the most optimal subset. In addition, after selecting the consistency subset evaluator as a final method at this stage, we evaluated this method with different search methods in terms of subset size, the ruleset size, and final accuracy after applying the rule-based methods to each of them. The experimental results are shown in Table 3.

Table 3. Comparing output accuracies of Consistency Subset Evaluator with different search methods

Search methods	Subset size	OneR acc	PART acc	PART rules number	JRip acc	JRip rules number
Tabo search [61]	8	74.3%	88.1%	916	80%	380
Step by step greedy search [54]	9	75%	90.3%	890	88.6%	340
Best first search [54]	7	76.1%	90.5%	921	89.2%	376
Exhaustive search [54]	7	79.4%	90.3%	964	89%	389
Random search [60]	8	75.1%	89.3%	925	88.1%	378

According to Table 3, no significant difference was observed between different search methods in terms of any comparison criteria, and the best first search method was selected for the rest of the process.

5.3.3.2. The proposed feature selection method based on data in the activity data set

In the activity dataset, as described, seven activity sensors and two heart rate sensors are used. Each of these activity sensors records position in three axes: height, width, and depth. One of our proposed solutions to reduce the size of these data is using Equation (1) (Pythagorean Theorem).

$$F = \sqrt{X^2 + Y^2 + Z^2} \tag{1}$$

Where X, Y, and Z represent the activity vector sizes in 3-dimension and F represents Pythagoras formula in 3-D space. Hence, instead of three attribute values for each sensor, one data was computed for each of them by averaging between the two heart sensors, two feature values were replaced by one feature value. Thus, instead of 23 attributes, there are eight features in the data. The experimental results of applying the proposed feature selection method are presented in Table 4.

Table 4. Final accuracy and number of output rules of the proposed method in two rule-based classifiers

feature selection method	OneR accuracy	PART accuracy	PART rules number	JRip accuracy	JRip rules number
The proposed feature selection method (Eq. (1))	71.72%	85.61%	1218	86.23%	281

As it is illustrated, this method has a lower accuracy than the consistency subset evaluator method.

5.3.3.3. Combined subsets evaluation and individual evaluation

In this approach, we first selected seven individual evaluators (chi-square feature evaluation [62], correlation feature evaluation [63], OneR feature evaluation [54], filter feature evaluation [54], feature evaluation by gain ratio [64], feature evaluation information gain [62] and feature evaluation with symmetric uncertainty [65]) and applied each of them to one of the selected items from the activity data set along with the ranker search method.

From the output of these methods, which was a list of 23 ranked features, we selected the first ten features and put them together (number ten was selected for the number of features because it is close to the number of features available in the consistency subset evaluation method). We selected the repetition features from these seven batches and placed them in a separate category (we chose a feature with at least three repetitions in each category.). Eventually, we applied the three rule-based classification algorithms to the selected subset and presented the evaluation results in Table 5.

Table 5. Final accuracy and number of output rules of the proposed combined method in rule-based classifiers

feature selection method	OneR accuracy	PART accuracy	PART rules number	JRip accuracy	JRip rules number

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

Combined method	71.70%	90.18%	700	89.93%	298

In the end, from these three feature selection methods, the proposed combined method generated was selected due to its accuracy and number of rules.

As well as the hospital data set, it was observed that the JRip algorithm produces a much smaller number of rules while its accuracy is close to the accuracy of the PART algorithm. This difference in the number of rules is more evident in this data set which is significantly larger than the hospital data set. Finally, the proposed feature selection method and JRip rule-based classification was our achievement in this section. Fig. 16 and 17 compare the three feature selection methods besides using rule-based classifications methods in terms of accuracy and number of generated rules. As we mentioned before in [2] it is indicated that fewer rules number leads to lower processing time.

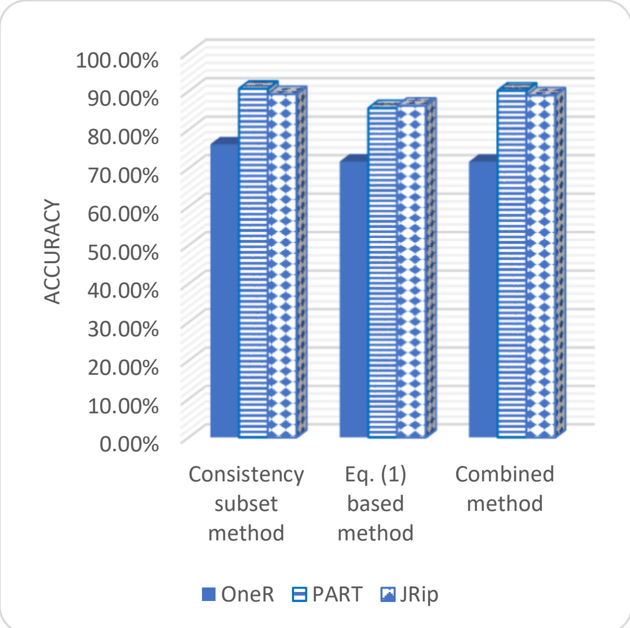


Fig. 16. Comparison of three feature selection methods in terms of accuracy in rule-based classification algorithms

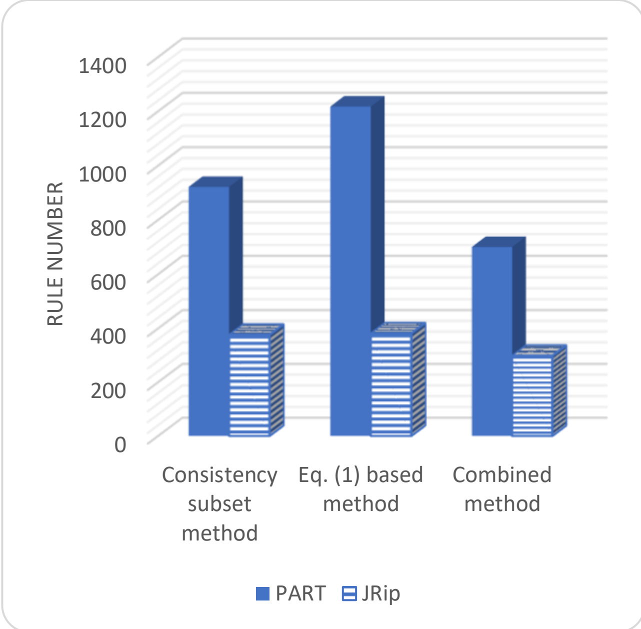


Fig. 17. Comparison of three feature selection methods in terms of the number of output rules in rule-based classification algorithms

In addition, the CPU and memory consumption of the proposed combined method in rule-based classifiers for Activity dataset are presented in Table 6.

Table 6. CPU and memory consumption of the proposed combined method in rule-based classifiers for activity dataset

Classifier	Memory consumption	CPU consumption
PART	315 MB	20.15%
JRip	400 MB	15.21%
OneR	340 MB	21.51%

5.3.4. Implementing and setting up CEP engine

As said in the previous section, we use the complex event method in this section. In this case, our goal is to detect the movement of the users to notify them or the rest of the people who need to be informed. In fact, in this data set, we have 13 different activities that the engine recognizes them based on the modeling of the input data in the form of events. By identifying an activity, it is announced to the user or the interested people in a suitable format.

As previously explained, in this work, the Siddhi platform [58] is the best choice for these kinds of cases. Moreover, an application under MATLAB was used to convert automatically the rules from the output format of the rule-based algorithm to the accepted format of the engine.

Additionally, as we mentioned before, CEP is a processing system and does not have ability to classify input data or predict a target class. It just detects complex events based on predefined rules. Thus, CEP's accuracy directly relies on the accuracy of rules. Because of using a rule-based learning (JRip) in this paper for rule extraction, the accuracy of CEP is equal to the accuracy of JRip algorithm.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

5.3.5. Results of rule adaption method by using previous data

The results of this experiment can be seen in Fig. 18. As it has been shown, this update process improves the accuracy by 13%. As mentioned in the previous section, updating process in all areas of computer science causes improvements and complex events processing is no exception. Especially, in the areas such as movement monitoring of people where the situation is constantly changing. Therefore, it is concluded that the best way to use event processing engines in various fields is to constantly update the rules and to create new rules or to improve the intervals of the previous ones.

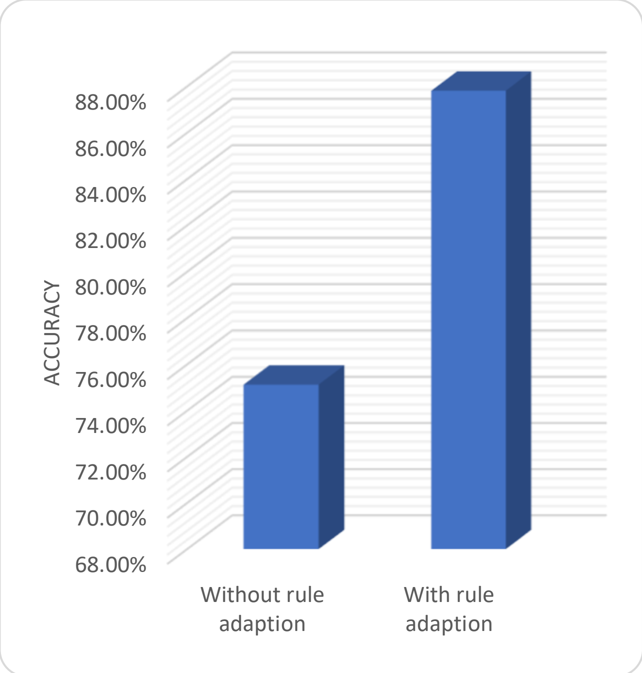


Fig. 18. Comparison of the system accuracy, with and without the proposed rule adaption method

5.3.6. Results of implementing personalization of rules using the previous data of each person

The output accuracy in experimental results is presented in Fig. 19.

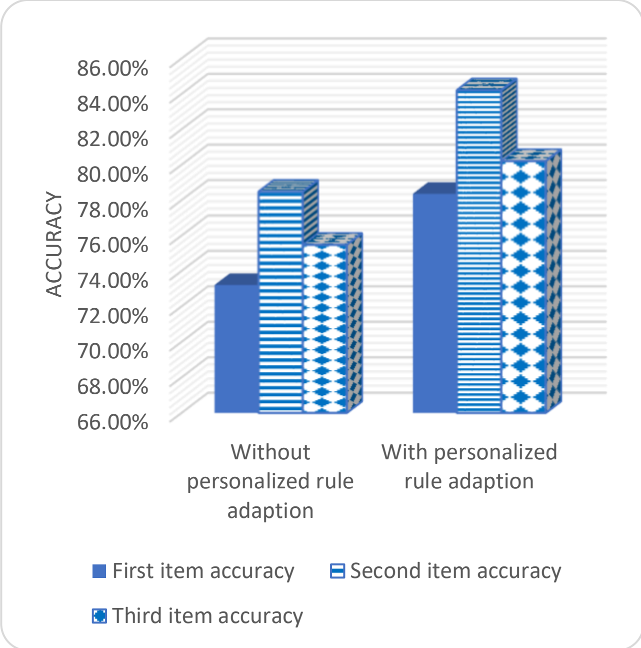


Fig. 19. Compare system output accuracy, with and without proposed personalized rule adaption method

As it is clear from Fig. 19, about 4 to 6 percent improvement has been achieved in this process. However, more volumes of samples for each user, make more difference in the behavior of users than their previous behavior. Thus, there is a possibility of changing this improvement value. It depends on the application and the user whose data is being processed and monitored.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

Table 7. Comparison of similar research studies using CEP in healthcare with our research

Research	Main goal	The proposed approach	Dataset	Evaluation results
Our research	Adaptive and personalized remote health monitoring based on CEP platform	Extracting rules from the selected rule-based classifier and setting the rule set of the CEP engine by them. Updating rules by applying JRip algorithm frequently due to the rule adaption and personalization.	The university of Queensland vital signs dataset. Mhealthdroid activity dataset	JRip has shown the best accuracy with respect to the lowest number of rules. Rule adaption improve accuracy of event detection between 13% up to 15% and rule personalization improve accuracy of event detection between 4% up to 6%.
Yao et al. [2]	Tracking surgeries and hospital equipment	Developing surgical procedure system in RFID enabled hospital using CEP	Recorded RFID data from various parts of the hospital	By increasing the size of the rule set, processing time and accuracy will be increased
Caballero et al. [18]	Collaborative context-aware health mobile application supported by a software architecture integrated with technologies like IoT and CEP	Using a system based on IoT and CEP that citizens can collaborate in a simple way and it can process and warn in real time with corresponding text alerts.	User study on 42 people of different ages and abilities with new technologies	This application and its key features could be very successful and practical
Mdhaffar et al. [3]	Predicting heart failure. Based on health analysis approach t	The patient's signs sent to a CEP engine that uses interval-based rules. A statistical method has been used to update the intervals for better diagnosis	MIMIC II waveform dataset	CEP4HFP achieves 84.75% precision, 100% recall, and the F1 score is around 91.74%.
Teng et al. [20]	Using a meta-heuristic method used as a classifier to obtain optimal parametric rules for the CEP engine.	Optimizes CEP rules by improving their behavior control parameters, based on improved alarm detection criteria.	-	The proposed scheme retrieves the parameterized rules optimally.
Olabarrieta et al. [21]	Providing a method for updating the adaptive rules to maintain the changing environment.	Eliminating unnecessary errors and data to improve processing efficiency. Abnormal rule is detected. Updating the rule condition part is done by limiting the compatible cluster rules and calculating the effective value.	Fuel level dataset	The proposed scheme retrieves the rules parameterized in a detection-optimal fashion. The performance gains provided by this method.
Wang et al. [23]	Proposing a complex predictive event processing method based on Bayesian networks.	The Bayesian model is designed by an inference method based on the Gaussian mixed model and EM algorithm and based on event type and time. The evolving Bayesian network structure is supported by the mountaineering method	Road traffic data set.	The total error is 12.12% and 7.78% for real and simulated data.
Lee et al. [24]	Proposing a sequential clustering-based rule generation method that extract rules from the experts' decision history	It generates graphs based on sequential clustering and modelling. In addition, the proposed method is able to support automatic updating of the rules, regularly or occasionally.	Stock market dataset	Increasing of 19.32% in performance in comparison to the existing complex dynamic event processing methods.
Mehdyev et al. [28]	Replacing the manual identification of event patterns	The fuzzy disordered rule induction (FURIA) algorithm has been used to identify the event patterns. The method was implemented after selecting the relevant feature subset using the Elitist Pareto multipurpose evolution algorithm (ENORA) to enhance diversity.	Dataset conducted by gathering accelerometer sensor data from 29 users and indicates their daily activities.	FURIA classifier combined with ENORA methodology classifies 98.39% of instances correctly.
Xiao et al. [29]	Cost-aware, fault-tolerant and reliable strategy (CaFtR).	Using network resources for obtaining continuous and available CEP regardless of dynamic operator migrations under fuzzy environment.	A case study by using stream base system application.	CaFtR can process the input events faster, and can stand the operator failure at 55 s.
Roldán et al. [25]	Proposing an architecture combining CEP and ML that is able to manage dynamic patterns to detect IoT security attacks.	A combination of a dynamic patterns that event characteristics depend on values are automatically calculated by a linear regression and support vector regression (SVR) prediction and a domain expert pattern graphical model by using MEdit4CEP - based tools.	A prototype IoT network built in a hospital that attacked by malicious device.	The results confirm that this architecture works effectively when using a model that can be adapted to the context and linear regression shows better results against SVR in terms of precision, recall and F1-score.
Mehdyev et al. [30]	Providing a machine learning model for CEP rule extracting	Rule-based learning comparing for creating the rule set of the CEP	Daily activity of 29 persons	Part has the best performance in terms of accuracy.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

Personalized rule adaption is more useful in monitoring user's movements than the previous data set because different users may move at different speeds and positions (running, climbing stairs, etc.). Also, the timing of the personalized rule adaption process can be different depending on the users themselves (in cases where the user has the authority, such as activity detection) or depending on the medical staff (in cases such as a hospital or home monitoring). All the implementation materials can be downloaded from: https://???[2].

5.4. Compassion proposed model with similar works

In Table 7, our research is compared with some papers that used CEP for health monitoring. Although the researches in TABLE 7 are different with our research in terms of the elevation goal, but in this table the main papers in the field of health monitoring based on CEP has been gathered.

There are some differences between the proposed work in [25] and our proposed work. In Roldán et al. the machine learning method is used for predicting upcoming events and makes a model based on this prediction. Thus, the system can be ready for unknown network attacks that might happen in future. In addition, a domain expert defines and determines some patterns for security attack. Then, the defined patterns are validated and converted to rules by using a predefined model. After that these rules are sent to CEP engine repository. However, in our proposed work, first the system stores user behavior data and then uses it to determine the rules that can be used as a rule set of CEP engine.

As we explained above, the output of the rule-based classification methods is a set of rules that can be used as the rules used in the CEP engine rule set with some modifications. Therefore, the outputs of the typical machine learning methods which act as a black box, cannot be directly converted into the form of the compatible rules for CEP engine. Thus, in our opinion it is not suitable for this platform. In addition, the explainability of the rule-based algorithms is higher than most of machine learning algorithms and more understandable. If necessary, with the advice and discretion of a domain expert, those rules that are outlier or inappropriate can be removed after obtaining the rules. As a result, we found that the machine learning methods which are used in suggested work are not suitable for rule extraction. Additionally, although these machine learning methods might be practical in threshold updating, but in case that a new situation happen or a new feature is added to the system or a new user with different behavior join to the system, threshold updating is not enough and updating method should be able to extract new rules too.

As it is mentioned above, evaluating the rule-based classifiers on a real time data is presented in [30]. Therefore, we illustrate the comparison of the results of our research and study in [30] Table 8. It is necessary to mention that the evaluation datasets in our work and the work presented in are not the same but we use the same classifiers in Weka.

Table 8. Comparison between our research and the last research in the field of rule-based classifiers

	OneR accuracy	PART accuracy	JRip accuracy	PART rule number	JRip rule number
Mehdiyev et al. [30]	79.89%	93.14%	92.13%	-	-
Our research	89.25%	98.61%	97.32%	40	24

As it is illustrated in Table 8, the PART method has the best accuracy in both researches. Furthermore, in, the rule-based classifiers have been just compared in terms of accuracy, error rate and etc. However, since, in our research. We want to use the extracted rules in a CEP engine, we compared the rule-based classifiers in terms of rule number in addition to classifier accuracy. Due to the results represented in TABLE II, although the highest accuracy belongs to the PART and JRip classifiers in both works, they were different in the number of extracted rules. Decreasing the number of rules leads to lower processing time as it shown in [2] as it shown in Fig. 20.

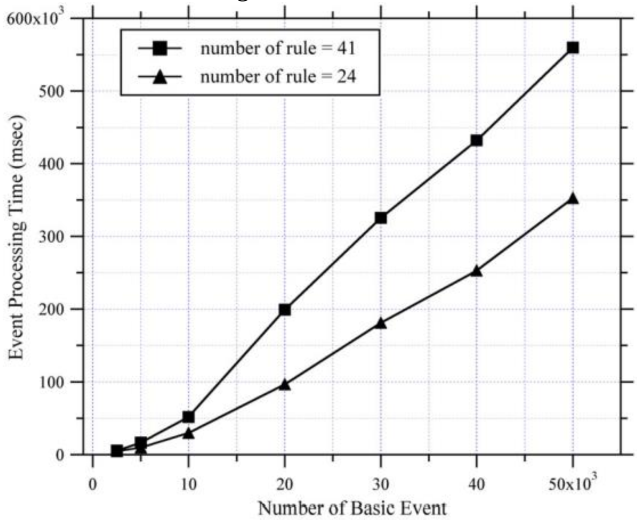


Fig. 20. Performance evaluation based on event processing time [2]

Based on Fig. 20, the system processing time decreases according to the number of rules decreasing. Hence, we chose JRip in our work according to its smaller rules number compared with the PART algorithm. As a result, Emergency Action Plan (EAP) in our platform is faster because it can detect complex events much faster than a platform which uses the PART algorithm for rules extraction. Therefore, in emergency situations, suitable actions can be triggered earlier.

[2] The link to the GitHub page will be updated in the camera-ready version of our paper after acceptance.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

6. Conclusion and future work

In this paper, an adaptive and personalized user behavior modeling platform based on CEP for remote health monitoring systems was presented, which provided solutions for each of its challenges.

The important challenges of these kinds of systems include massive generated data streams, human errors and cold start in the design of health systems, getting out-of-date and degradation of the system efficiency, and personalization in monitoring users according to the specific conditions of each of them. For each challenge, a solution has been introduced in this paper.

To solve the problem of massive generated data streams, the CEP method is a suitable solution because it uses a set of rules to process and model the input streams. This method performs the real-time processing of data streams using modeling of patient data in the form of complex events.

Moreover, to solve the problem of human errors and cold start in setting rules for a CEP system, exploiting machine learning approaches were suggested as a suitable solution. Among the various methods in this field, rule-based classification methods were selected to construct rules similar to those used in CEP. Also, rule-based methods were more explainable than the other machine learning approaches because domain experts can supervise the whole system. Among the rule-based classification methods, the JRip algorithm was the most appropriate method both in terms of accuracy and the number of generated rules in comparison with other rule-based classification methods. After the feature extraction phase and before classification, depending on the feature type and size appropriate feature selection methods can be used. In this research, a combined method was presented which uses a combination of individual feature evaluation methods and the logic of the feature subset evaluation method. This method is a combination of seven feature selection methods with an individual evaluation method. We selected the repetition features in the first ten features of these seven evaluators and presented them as a subset. In terms of accuracy, the output of this method in combination with the JRip algorithm was almost identical to the consistency subset evaluation, but it was able to produce a smaller number of rules without any degradation in terms of accuracy.

Another problem was the decreasing the system efficiency due to the out-of-date rules. To solve this problem, a rule adaption approach was presented that uses a database along with the CEP engine and the data passing through the system was stored in this memory with their labels, and by helping it, the engine rules were updated. This approach was evaluated one time with the original rules and another time with the updated rules. The results of the evaluation indicate that in the case of updating process the accuracy is improved about 13% to 15% compared to the situation without updating process.

The last challenge in the health monitoring systems, considered in this work, is personalization in monitoring users according to the specific conditions of each of them. A personalized rule adaption method was proposed that personalized rules with the data of one user individually. The data was divided into three segments. The first segment was used for initial rule extraction. Then, the second segment was imported to the CEP engine and stored in a database. After that, the first and second segments were combined and new rules were extracted from them and CEP engine rule set was updated by new rules. At the end, the third segment was used for evaluating and comparing the efficiency of initial and updated rules. The Evaluation results demonstrate that the accuracy in the personalized rules has been improved by about 3% to 6% compared to the situation without updating the rules.

Eventually, we can conclude that CEP is an appropriate tool for user behavior modelling in order to remote health monitoring in real-time manner. Furthermore, rule-based machine learnings such as JRip can be useful for addressing the cold start problem and defining rules in a CEP-based system. Moreover, JRip is able to make fewer rules and it leads to decrease the processing time. As a result, Emergency Action Plan (EAP) in the proposed adaptive CEP platform is faster than CEP platforms with higher number of rules. Besides, rule-based machine learnings are more explainable than other learning methods. Thus, the domain experts will have a better sense of the system functionality. At the end, we showed in this paper that rule adaption and personalized rule adaption can increase the performance of the CEP-based health monitoring system and personalizes the system for each user by utilizing previous data.

It should be noted that this research would have benefited from real data collected live from sensor devices involving real participants, but we decided to use datasets instead because of ethical implications of collecting data from real users and secondly, the high cost associated with recruiting a big group of participants and large number recording devices. This, of course, can be considered as a limitation of the current study.

In the future work, we will try to examine the system in another context like stock market. Moreover, we can adjust rule-based learning in order to achieve better performance in a CEP-based system.

Furthermore, we decide to study more methods for extracting rules such as artificial neural network-based and deep learning-based and even fuzzy methods due to cost-aware, fault-tolerant and reliable strategy (CaFtR).

Additionally, we plan to develop a specific method based on a model like complex evidential quantum dynamical (CEQD) which can improve defined rules by domain expert and combine intelligent approaches with expert decision-making.

Reference

[1] Azimi I, Rahmani AM, Liljeberg P, Tenhunen H. Internet of things for remote elderly monitoring: a study from user-centered perspective. Journal of Ambient Intelligence and Humanized Computing. 2017;8:273-89.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

[2] Yao W, Chu C-H, Li Z. Leveraging complex event processing for smart hospitals using RFID. Journal of Network and Computer Applications. 2011;34:799-810.
[3] Mdhaffar A, Rodriguez IB, Charfi K, Abid L, Freisleben B. CEP4HFP: Complex event processing for heart failure prediction. IEEE transactions on nanobioscience. 2017;16:708-17.
[4] Naseri MM, Tabibian S, Homayounvala E. Intelligent Rule Extraction in Complex Event Processing Platform for Health Monitoring Systems. 2021 11th International Conference on Computer Engineering and Knowledge (ICCKE): IEEE; 2021. p. 163-8.
[5] Tsiourti C, Joly E, Wings C, Moussa MB, Wac K. Virtual assistive companions for older adults: qualitative field study and design implications. Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare: ICST (Institute for Computer Sciences, Social-Informatics and …; 2014. p. 57-64.
[6] Verma P, Sood SK. Fog assisted-IoT enabled patient health monitoring in smart homes. IEEE Internet of Things Journal. 2018;5:1789-96.
[7] Becker E, Metsis V, Arora R, Vinjumur J, Xu Y, Makedon F. SmartDrawer: RFID-based smart medicine drawer for assistive environments. Proceedings of the 2nd international Conference on Pervasive Technologies Related to Assistive environments2009. p. 1-8.
[8] Albahri AS, Albahri OS, Zaidan A, Zaidan B, Hashim M, Alsalem M, et al. Based multiple heterogeneous wearable sensors: A smart real-time health monitoring structured for hospitals distributor. IEEE access. 2019;7:37269-323.
[9] Kharel J, Reda HT, Shin SY. Fog computing-based smart health monitoring system deploying lora wireless communication. IETE Technical Review. 2019;36:69-82.
[10] Basu S, Saha S, Pandit S, Barman S. Smart Health Monitoring System for Temperature, Blood Oxygen Saturation, and Heart Rate Sensing with Embedded Processing and Transmission Using IoT Platform. Computational Intelligence in Pattern Recognition: Springer; 2020. p. 81-91.
[11] Forooghifar F, Aminifar A, Atienza D. Resource-aware distributed epilepsy monitoring using self-awareness from edge to cloud. IEEE transactions on biomedical circuits and systems. 2019;13:1338-50.
[12] Subasi A, Khateeb K, Brahimi T, Sarirete A. Human activity recognition using machine learning methods in a smart healthcare environment. Innovation in Health Informatics: Elsevier; 2020. p. 123-44.
[13] Gupta D, Bhatia M, Kumar A. Resolving Data Overload and Latency Issues in Multivariate Time-Series IoMT Data for Mental Health Monitoring. IEEE Sensors Journal. 2021;21:25421-8.
[14] Pathak R, Vaidehi V. Complex event processing based remote health monitoring system. 2014 3rd International Conference on Eco-friendly Computing and Communication Systems: IEEE; 2014. p. 61-6.
[15] Pérez-Vereda A, Flores-Martín D, Canal C, Murillo JM. Complex Event Processing for health monitoring. International Workshop on Gerontechnology: Springer; 2018. p. 3-14.
[16] Dautov R, Distefano S, Buyya R. Hierarchical data fusion for Smart Healthcare. Journal of Big Data. 2019;6:19.
[17] Rahmani AM, Babaei Z, Souri A. Event-driven IoT architecture for data analysis of reliable healthcare application using complex event processing. Cluster Computing. 2020:1-14.
[18] Caballero P, Ortiz G, Garcia-de-Prado A, Boubeta-Puig J. Paving the way to collaborative context-aware mobile applications: a case study on preventing worsening of allergy symptoms. Multimedia Tools and Applications. 2021;80:21101-21133.
[19] Manashty A, Light J, Yadav U. Healthcare event aggregation lab (HEAL), a knowledge sharing platform for anomaly detection and prediction. 2015 17th International Conference on E-health Networking, Application & Services (HealthCom): IEEE; 2015. p. 648-52.
[20] Teng P, Li G, Su L, Chen X. Adaptive Rule Update Method in Complex Event Process. 2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC): IEEE; 2017. p. 270-5.
[21] Olabarrieta II, Torre-Bastida AI, Laña I, Campos-Cordobes S, Del Ser J. A Heuristically Optimized Complex Event Processing Engine for Big Data Stream Analytics. International Conference on Harmony Search Algorithm: Springer; 2017. p. 101-11.
[22] Xiao F. CEQD: A complex mass function to predict interference effects. IEEE Transactions on Cybernetics. 2021.
[23] Wang Y, Gao H, Chen G. Predictive complex event processing based on evolving Bayesian networks. Pattern Recognition Letters. 2018;105:207-16.
[24] Lee O-J, Jung JE. Sequence clustering-based automated rule generation for adaptive complex event processing. Future Generation Computer Systems. 2017;66:100-9.
[25] Roldán J, Boubeta-Puig J, Martínez JL, Ortiz G. Integrating complex event processing and machine learning: An intelligent architecture for detecting IoT security attacks. Expert Systems with Applications. 2020;149:113251.
[26] Boubeta-Puig J, Díaz G, Macià H, Valero V, Ortiz G. MEdit4CEP-CPN: An approach for complex event processing modeling by prioritized colored Petri nets. Information Systems. 2019;81:267-89.
[27] Boubeta-Puig J, Ortiz G, Medina-Bulo I. MEdit4CEP: A model-driven solution for real-time decision making in SOA 2.0. Knowledge-Based Systems. 2015;89:97-112.
[28] Mehdiyev N, Krumeich J, Werth D, Loos P. Determination of event patterns for complex event processing using fuzzy unordered rule induction algorithm with multi-objective evolutionary feature subset selection. 2016 49th Hawaii International Conference on System Sciences (HICSS): IEEE; 2016. p. 1719-28.
[29] Xiao F. CaFtR: A Fuzzy Complex Event Processing Method. International Journal of Fuzzy Systems. 2021:1-14.
[30] Mehdiyev N, Krumeich J, Enke D, Werth D, Loos P. Determination of rule patterns in complex event processing using machine learning techniques. Procedia Computer Science. 2015;61:395-401.
[31] Arrieta AB, Díaz-Rodríguez N, Del Ser J, Bennetot A, Tabik S, Barbado A, et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information Fusion. 2020;58:82-115.
[32] Tjoa E, Guan C. A survey on explainable artificial intelligence (xai): Toward medical xai. IEEE Transactions on Neural Networks and Learning Systems. 2020.
[33] Nazar M, Alam MM, Yafi E, Mazliham M. A Systematic Review of Human-Computer Interaction and Explainable Artificial Intelligence in Healthcare with Artificial Intelligence Techniques. IEEE Access. 2021.
[34] Cutillo CM, Sharma KR, Foschini L, Kundu S, Mackintosh M, Mandl KD. Machine intelligence in healthcare—perspectives on trustworthiness, explainability, usability, and transparency. NPJ digital medicine. 2020;3:1-5.
[35] Luckham D. The power of events: Addison-Wesley Reading; 2002.
[36] Etzion O, Niblett P, Luckham D. Event processing in action: Manning Greenwich; 2011.
[37] Wang F, Liu S, Liu P. Complex RFID event processing. The VLDB Journal. 2009;18:913-31.
[38] Adi A, Etzion O. Amit-the situation manager. The VLDB journal. 2004;13:177-203.
[39] Luckham D, Schulte R. Epts event processing glossary v2. 0. Event Processing Technical Society. 2011.
[40] Liu L, Özsu MT. Encyclopedia of database systems: Springer New York, NY, USA:; 2009.
[41] Margara A, Salvaneschi G. Ways to react: Comparing reactive languages and complex event processing. REM. 2013;14.
[42] Zang C, Fan Y, Liu R. Architecture, implementation and application of complex event processing in enterprise information systems based on RFID. Information Systems Frontiers. 2008;10:543-53.
[43] Hacker P, Krestel R, Grundmann S, Naumann F. Explainable AI under contract and tort law: legal incentives and technical challenges. Artificial Intelligence and Law. 2020;28:415-39.
[44] Yang G, Ye Q, Xia J. Unbox the black-box for the medical explainable ai via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. Information Fusion. 2022;77:29-52.
[45] Holte RC. Very simple classification rules perform well on most commonly used datasets. Machine learning. 1993;11:63-90.
[46] Frank E, Witten IH. Generating accurate rule sets without global optimization. 1998.
[47] Quinlan JR. C4. 5: programs for machine learning: Elsevier; 2014.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421

[48] Cohen WW. Fast effective rule induction. Machine learning proceedings 1995: Elsevier; 1995. p. 115-23.
[49] document W. WSO2 document. [Online]. Accessed 18 March 2022
[50] guide Sq. Siddhi query guide. [Online]. Accessed 18 March 2022
[51] Liu D, Görges M, Jenkins SA. University of Queensland vital signs dataset: development of an accessible repository of anesthesia patient monitoring data for research. Anesthesia & Analgesia. 2012;114:584-9.
[52] Banos O, Garcia R, Holgado-Terriza JA, Damas M, Pomares H, Rojas I, et al. mHealthDroid: a novel framework for agile development of mobile health applications. International workshop on ambient assisted living: Springer; 2014. p. 91-8.
[53] Banos O, Villalonga C, Garcia R, Saez A, Damas M, Holgado-Terriza JA, et al. Design, implementation and validation of a novel open framework for agile development of mobile health applications. Biomedical engineering online. 2015;14:1-20.
[54] Witten IH, Frank E, Hall MA, Pal CJ. Data Mining: Practical Machine Learning Tools and Techniques: Elsevier Science; 2016.
[55] WSO2. "WSO2".[Online]. Accessed 18 March 2022
[56] EsperTech. “Esper.”[Online]. Accessed 18 March 2022
[57] Drools. “Drools – Business Rules Management System.” [Online]. Accessed 18 March 2022
[58] Siddhi. "Siddhi".[Online]. Accessed 18 March 2022
[59] Hall MA. Correlation-based feature subset selection for machine learning. Thesis submitted in partial fulfillment of the requirements of the degree of Doctor of Philosophy at the University of Waikato. 1998.
[60] Liu H, Setiono R. A probabilistic approach to feature selection-a filter solution. ICML: Citeseer; 1996. p. 319-27.
[61] Hedar A-R, Wang J, Fukushima M. Tabu search for attribute reduction in rough set theory. Soft Computing. 2008;12:909-18.
[62] Rachburee N, Punlumjeak W. A comparison of feature selection approach between greedy, IG-ratio, Chi-square, and mRMR in educational mining. 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE): IEEE; 2015. p. 420-4.
[63] Karegowda AG, Manjunath A, Jayaram M. Comparative study of attribute selection using gain ratio and correlation based feature selection. International Journal of Information Technology and Knowledge Management. 2010;2:271-7.
[64] Quinlan JR. Simplifying decision trees. 1986.
[65] Ali SI, Shahzad W. A feature subset selection method based on symmetric uncertainty and ant colony optimization. 2012 International Conference on Emerging Technologies: IEEE; 2012. p. 1-6.

AMM copy, Published in Artificial Intelligence in Medicine Journal 2022, https://doi.org/10.1016/j.artmed.2022.102421