Available online at www.HEFJournal.org

Journal of
Human, Earth, and Future

ISSN: 2785-2997

Vol. 3, No. 1, March, 2022

Decision-making Improvement in Dynamic Environments Using Machine Learning

Roua Jabla [1, 2*], Maha Khemaja [2], Sami Faiz [3]

[1] *Department of Computer Engineering, Universitat Politècnica Valencia, Camino de Vera S/N, 46022 Valencia, Spain*

[2] *Institute of Computer Sciences and Communication Techniques (ISITCom), University of Sousse, Sousse 4011, Tunisia*

[2] *LTSIRS Laboratory, University of Tunis El Manar, Tunis 5020, Tunisia*

Received 27 November 2021; Revised 24 January 2022; Accepted 11 February 2022; Published 01 March 2022

Abstract

The proliferation of ubiquitous computing with smartphones makes context models and information extremely rich and dynamic on account of highly dynamic environments. However, defining rules at design time may impair their efficiency and the decision-making process at runtime. Therefore, it is important to address decision-making problems leveraged by dynamic environments and context-model evolution. In this sense, a solution that could emerge is the continuous rule knowledge base evolution at runtime. In this paper, we propose a decision adaptation component that relies on generating new rules due to changes occurring around users at runtime. This component aims to support decision-making in dynamic environments and to alleviate the human effort needed to infer new rules. A case study was conducted to illustrate the implementation of the proposed component for the rule knowledge database enrichment and the decision-making improvement. Moreover, an experimental evaluation is provided to assess the effectiveness of the proposed component. The results show that this component exhibits better effectiveness than other well-known algorithms and state-of-the-art approaches.

Keywords: Ubiquitous Computing; Context-Awareness; Rule Generation; Decision-Making; Human–computer Interaction.

1. Introduction

Ubiquitous computing is the upward trend in computer technology in which computation occurs anywhere and everywhere using any device [1]. All current technology may switch to ubiquitous computing environments in the coming years since ubiquitous computing integrates new types of computing such as mobile computing, pervasive computing, context-awareness, and adaptation while hiding the technology and making it meld into our daily lives. In the era of ubiquitous computing, the smartphone is the most popular device thanks to the large number of sensors that it can incorporate [2]. In light of moving from a single context to multi-contexts with the use of smartphones and the exploitation of their advanced sensing capabilities, it is pertinent to deal with users' dynamic environments and infer relevant knowledge, i.e., rules.

Due to the rapid development of ubiquitous computing and context-aware technology in mobile devices, ubiquitous computing is leading to the advent of mobile device-oriented context-aware applications because they seek to be

* Corresponding author: jabla.roua@gmail.com

http://dx.doi.org/10. 28991/HEF-2022-03-01-04

everywhere. In existing mobile device-oriented context-aware applications, rules can only react to changes in environment attributes and context information [3]. As a result, these rules cannot respond to dynamic environments or context model evolution during runtime [3]. This means that rules cannot be static and need to be constantly evolved to remain relevant [4]. Therefore, an important challenge is raised related to the potential of context-aware applications to adjust their behaviors to the dynamics entailed in their surrounding environments and provide appropriate rules at runtime. To tackle this challenge, the supporting middleware of such a context-aware application should carry out dynamic changes in the application behavior at runtime through the evolution of its context model and, subsequently, its rule knowledge base with new rules to improve decision-making in dynamic and changing environments. Hence, a considerable need for a decision-making process, which aims to continuously enrich a rule knowledge base over time, has emerged to make applications more resilient to dynamic environments as well as to context model evolution at runtime.

To fill this need, we propose a component that augments an existing distributed middleware that enables context-aware applications to support dynamic pervasive computing environments. The main feature of the proposed component is to offer context-aware applications, where their rule knowledge bases are fluid and evaluative at runtime. The novelty and contribution of this paper could be drawn from three-fold. First, we provide a component for the purpose of enriching a rule knowledge base following an ontology-based context model evolution to work in dynamic environments. Second, we propose a novel hybrid learning approach towards effectively generating a complete set of non-redundant rules, in automated fashion, by adopting two decision tree machine learning algorithms. Third, we extend a *Genetic Algorithm* (GA) [5] with a multi analysis technique targeting the rule optimization. Then, we propose as a case study the application that is intended to assist engineers in the rule generation process at runtime. Moreover, we have conducted a range of experiments to assess the effectiveness of the proposed component through measuring the number of rules as well as the accuracy on different data sources from the UCI Machine Learning Repository [6] compared to other well-known algorithms and to other state-of-the-art approaches. The proposed component proves the effectiveness by achieving the best result in terms of number of rules and the highest accuracy among all algorithms and state-of-the-art approaches.

The rest of this paper is organized as follows. Section 2 discusses related works about rule generation. In section 3, a detailed presentation of the proposed component is presented. In section 4, a case study and an example of rule generation are described. Section 5 states the research questions and the experimental setup and then discusses the obtained results. Finally, in Section 6, we draw conclusions and highlight directions for future work.

2. Literature Review

Rule generation is one of most popular topics in data mining that allows the efficiency of context-aware applications adaptation according to context and environment changes occurring at runtime. Recently, this topic has been extensively studied in the literature because of its wide applicability and usefulness. By going through the literature, some approaches have supported rule generation to target ontology enrichment while certain other approaches have targeted the decision-making improvement. In the following, we provide an overview of the most recent approaches on this topic and we outline a brief discussion that highlights the research gap that motivates us to propose our component.

A couple of approaches for rule generation have been proposed in the scope of ontology enrichment process. In this sense, Paiva et al. [7] introduced a semi-automatic approach that focuses on enriching an ontology with relations between concepts using association rules. In this approach, they used the FP-Growth algorithm [8] to discover frequent items from unstructured data sources and then generate a set of association rules. After that, they exploited generated association rules for learning useful relations in the ontological model. Another work considered ontology enrichment based on learning association rules is the work of Idoudi et al. [9]. This work described a new approach for ontology content evolution through incorporating knowledge derived from medical records. For this purpose, Apriori algorithm [10] is applied to generate association rules. Next, domain experts are invited to validate generated rules. Then, validated rules are used in the enrichment process of the knowledge base with new association between existing concepts. Later, the authors in [11] presented a work, in which an association rule mining algorithm is employed to find recurrent patterns and discover association rules. The discovered rules can be automatically exploited to enrich an existing ontology with formal definitions of concepts. Apart from these approaches, novel approaches, in the scope of decision-making improvement, are emerging. These novel approaches fall in two categories. The first category includes approaches using traditional algorithms. For example, Gabroveanu et al. [12] proposed a recommender system in the field of distance learning in higher education. They used data obtained from learning management systems database and Apriori algorithm in order to identify association rules related to courses followed by students. Kaliappan et al. [13] proposed a new modified Apriori algorithm for finding the association rules among large datasets to promote sales and user interaction. They showed that the proposed algorithm improved the efficiency of generating association rules. Davagdorj and Ryu [14] offered an association rule mining method to discover useful patterns, which include medical knowledge, from a medical dataset. They applied the FP-Growth algorithm to extract a set of association rules. Then, the obtained rules are used to support medical decision-making for interpreting diagnosing patient information. Moreover, Asadianfam et al. [15] provided a new approach to improve recommendations in recommender systems. One of the challenges considered in this approach is that it could not provide appropriate recommendations to users who have different profiles from the existing users' profiles. To deal with this challenge, authors used the Apriori algorithm to generate association rules from users' behaviors and then made appropriate recommendations. They showed that the generated association rules could increase the overall efficiency of the recommender system. More recently, Miswan et al. [16] proposed a framework of association rule mining in readmission tasks. The proposed framework consisted of two processes, namely data pre-processing and rule mining extraction. Apriori algorithm is used to extract the hidden input variable patterns and relationships among admitted patients by generating supervised learning rules. The mined rules are discussed and validated by the domain expert, which is a valuable guide in making decisions on targeted patients' clinical resources based on various readmission durations.

In addition, Islam et al. [17] applied an association rule mining approach to help the insurance industry reduce the risk of adverse claims. To do so, they used Apriori-based frequent itemset algorithm to generate new association rules for supporting decision-making process. Finally, Sánchez-de-Madariaga et al. [18] proposed a new semi-supervised data mining model to discover new valuable scientific medical knowledge. They applied the FP-Growth algorithm to extract medical association rules. Moreover, they showed that FP-Growth is an efficient algorithm for calculating frequently co-occurring items in a dataset. The second category includes approaches using machine learning algorithms. In this context, Zulkernain et al. [19] introduced an intelligent mobile interruption management system. The main idea of their proposed system is to intelligently assist users in their daily activities. To this end, a Decision Tree algorithm is used to make intelligent decisions. Moreover, Sarker et al. [20] presented an association rule learning approach that can be used to discover a set of non-redundant and useful rules. In their approach, they considered, first, the Naïve Bayes (NB) algorithm to eliminate noise from data and, second, the Decision Tree algorithm to generate a set of association rules. These algorithms are used to build a robust prediction model that could improve the prediction accuracy. Finally, Basha [21] provided a cardiovascular prediction system that combines the K-Nearest Neighbour (KNN) algorithm with a GA to extract strong association rules for facilitating the decision-making process. First, the proposed system extracts association rules using KNN algorithm. Then, the output rules become the population of the GA to remove redundant and irrelevant rules.

Nevertheless, a common weakness that can be found in the most of above-described approaches [7-18] stands on the use of traditional algorithms, such as Apriori and FP-Growth. This drawback is due to the narrow applicability of these algorithms and the huge number of generated association rules [22]. As these traditional algorithms generate numerous association rules, many uninteresting or redundant rules are generated along with the interesting rules. This redundancy makes not only the rule-set unnecessarily large but also makes the decision-making process more complex and ineffective. In contrast, approaches using machine learning algorithms [19-21] can overcome this drawback to avoid rule redundancy. Furthermore, certain approaches derive the rule generation process at design time by offline mining. In this case, a context-aware application cannot not stand the dynamic environments at runtime. A further drawback is that a human intervention is needed in certain approaches to validate and manage generated rules. None of the above-described approaches, apart from [21], support the rule optimization with a target of getting the well-performed rules. Along the similar line, we aim to close gaps within the related works by proposing a component that relies on automatically enriching a rule knowledge base with new rules discovered at runtime to deal with dynamic environments. This component aims to discover and generate a set of non-redundant and well-performed rules represented as an IF-THEN logical statement, which are reliable for decision-making. To this end, the proposed component firstly, derives association rules from training models of decision tree machine learning algorithms, and secondly, retrieves the well-performed rules using an extension of a Genetic Algorithm, which can be applied in a straightforward manner for rule optimization.

3. Decision Adaptation Component

In this section, a global approach that aims to extend an existing middleware with a new composite component, is presented. The overall objective of the global approach is to support context evolution and dynamic decision-making process in dynamic environments at runtime without developer's intervention or system's disruption. This objective led to the proposal of a composite middleware component based on a four-component as depicted in Figure 1.

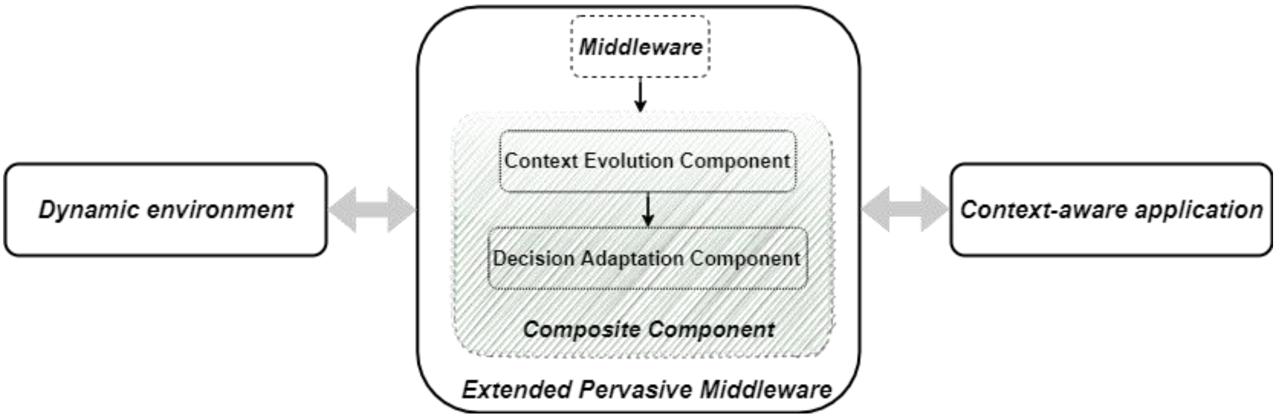

Figure 1. Architecture overview of the global approach

Within this composite middleware component, only the decision adaptation component, which deals with the enrichment of a rule knowledge base by generating non-redundant rules to go along with context evolution [23] in dynamic environments at runtime, is considered in this paper. The proposed component is designed to generate well-performed rules needed to automatically cover dynamic environments that change frequently at runtime. This component runs only once a context model evolution is achieved and a priori rules are deemed not to be relevant anymore for the new context model. The overall architecture of the component is illustrated in Figure 2. As illustrated in Figure 2, the decision adaptation component includes two key modules, namely "rule learning" and "rule optimization", that are described in more detail in the following subsections.

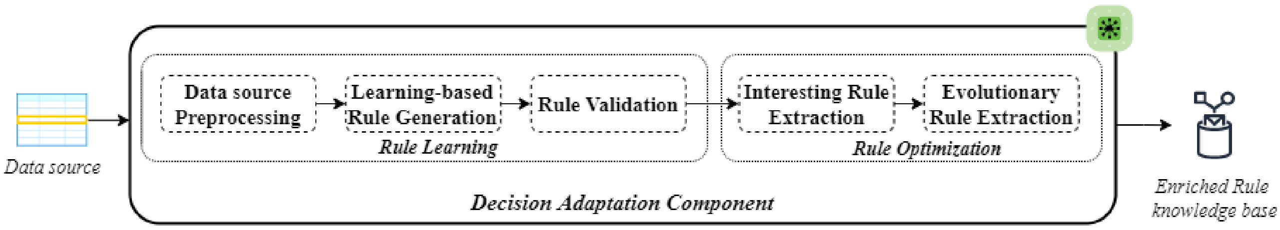

Figure 2. Architecture overview of the decision adaptation component.

3.1. Rule Learning

Rule learning module is responsible for learning and deriving non-redundant rules of IF-THEN statements from a candidate data source previously considered in the context model evolution process. Three key phases leading, ultimately, to the generation of IFTHEN rules, are previously highlighted in Figure 2. In the first phase, called data pre-processing, the candidate data source is preprocessed with the purpose of improving its quality to ensure the generation of consistent rules. Hence, two major steps are involved:

- Data cleaning that attempts to fill missing data and smooth out noisy data in the candidate data source. For filling missing values, we replace identified missing data with the average of existing data. And for smoothing noisy data, we carry out techniques, such as clustering, regression and binning, to eliminate noise in the candidate data source.
- Data reduction that is used to simplify the data source representation without any loss of useful information. To reach it in the easiest manner, we remove redundant and inconsistent data.

In the second phase, called learning-base rule generation, the candidate data source is processed to uncover relationships between seemingly unrelated elements, in a tree structure. This step is carried out through two steps as in Figure 3. First, the hybrid supervised learning step is applied to train the candidate data source targeting the creation of tree-structured training models. Towards effectively generating a complete set of non-redundant rules, we explore the idea of hybridizing machine learning algorithms with the use of two decision tree machine learning algorithms, Decision Tree [24] and Random Tree [25]. The main reason behind the choice of these algorithms is that they could represent the best compromise between accuracy and computational complexity to address the decision-making accuracy question compared with the different decision tree machine learning algorithms such as BF Tree, REP Tree, Decision Stump and Simple Cart. The resulting training models consist of a set of decisions in a tree structure, which could be utilized to generate rules from each leaf node [26]. Then, the rule extraction step is performed to automatically extract rules as IF-THEN statements. Each rule has two parts, an antecedent (IF) and a consequent (THEN) that are correlated to each other. An antecedent can be constituted by a set of atoms while a consequent contains only one atom. In the third phase, called rule validation, the following methods are applied:

- Rule structure verification, which deals with verifying that rules are preserving the inherent IF-THEN structure.
- Rule consistency verification, which has to check the consistency of rules and enumerate all inconsistent rules, where the consequent part does not refer to the antecedent part.

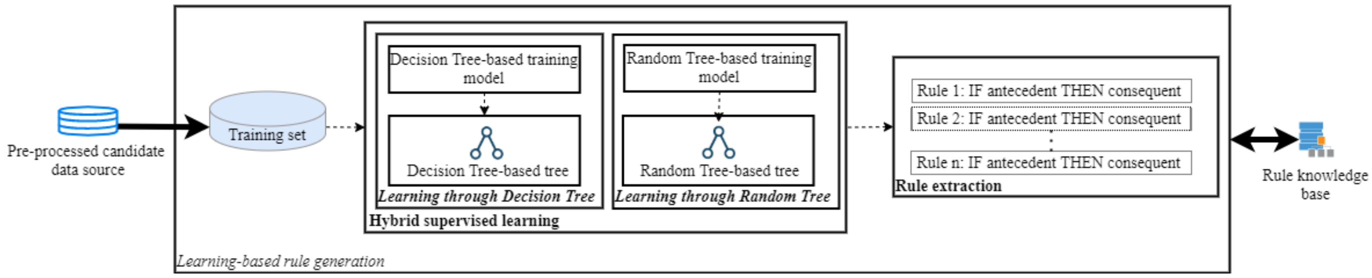

Figure 3. Learning-based rule generation phase

3.2. Rule Optimization

The rule optimization module is used to identify the well-performed rules to avoid irrelevant rules from the set of earlier validated rules. To this end, GA is extended to support a multi-analysis technique in order to jointly optimize the set of validated rules. Two main phases in the rule optimization module are illustrated in Figure 4. The first phase, namely the interesting rule extraction, is aimed to select the interesting rules and the second phase, namely the evolutionary rule extraction, is used to re-optimize the interesting rules selected in the first phase. The two phases are briefly summarized in Algorithm 1. According to Figure 4 and Algorithm 1, the interesting rule extraction phase identifies interesting rules on the basis of the candidate data source using GA. This algorithm starts with the construction of an initial population as a collection of chromosomes and lets them evolve over multiple generations to reach more and more interesting rules. In this case, each chromosome represents a candidate rule composed of the consequent part and the antecedent part. Next, the applied GA measures the performance of each chromosome using the fitness function to find out rules that have a support value above a certain threshold value. Then, it proceeds through three genetic operators, including, selection, crossover and mutation, to evolve a new generation and determine again the fitness function of each chromosome. At the end, it stops when it converges to an optimal set of interesting rules that could satisfy the fitness function. Due to the fact that fitness functions, defined in GA, are in conflict, the evolutionary rule extraction phase proposes a multi-analysis technique to retrieve from the obtained interesting rules, the well-performed rules that could lead to achieve the appropriate decision-making accuracy. On the grounds of this, we propose rule ranking and refinement steps as shown in Figure 4. The multi-analysis technique starts with the rule ranking step that is in charge of automatically ranking rules derived from decision tree machine learning algorithms and the interesting rules regarding the frequency of occurrence and the fitness function weight. The rule occurrence frequency is considered as the support degree to classify rules, followed by the fitness function weight. Then, the rule refinement step is performed to retrieve the set of well-performed rules with better accuracy performance on foundation of the ranked rules in order to enhance the decision-making accuracy. This step begins with finding a user who is related to runtime changes that occurred in the dynamic environment and loading the user profile of the corresponding user. The user profile is integrated in the refinement step to infer the well-performed rules.

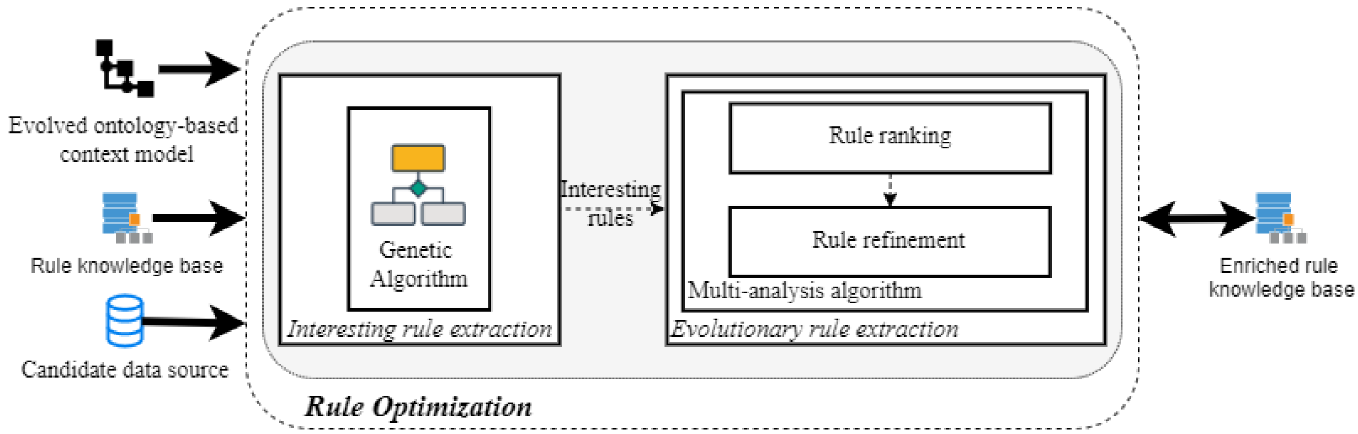

Figure 4. Learning-based rule generation phase

Algorithm 1. *Rule optimization*

```
    Input: DS pre-processed candidate data source
        FT fitness threshold value
    Output: Well-performed IF-THEN rules
Begin
Initialize population
Repeat
    Fitness evaluation
    Selection
    Crossover
    Mutation
Until fitness of new population > FT
For each interesting rule
    Calculate occurrence frequency
    If (! existFrequency(frequency))
      Check fitness function weight
    Else
      Calculate rank
    End if
End for
For each ranked rule
    If (! isRelatedToUserContext(ranked rule))
      Delete rule from the rule knowledge base
    End if
End for
End
```

4. Case Study

In this section, we present an example of a case study that allows us to illustrate the above-described component through providing the generation of well-performed rules from a candidate data source. For this, we consider an application for assisting engineers to enrich a rule knowledge base regarding an ontology-based context model evolution due to arisen changes in the surrounding environment at runtime. The presented application consists of two distinct parts, called frontend and backend. The backend part, which deals with the automatic rule generation, is implemented as REST web services [27] in Java. The frontend is created with Angular to deal with engineers' interactions and interfaces with the backend. For the present case study, we chose the weather data source [6], containing four condition attributes, such as outlook, temperature, humidity, windy, and one decision attribute 'play'. As noted previously, the candidate data source is already used for the automatic ontology-based context model evolution at runtime. In contrast to previous use, we consider the candidate data source to generate rules that help in making decisions regarding whether a user could go outside for playing or not. After the data pre-processing step, a hybrid supervised learning is performed on the candidate data source for creating tree-structured training models through training both Decision Tree and Random Tree algorithms as depicted in Figure 5 and 6, respectively.

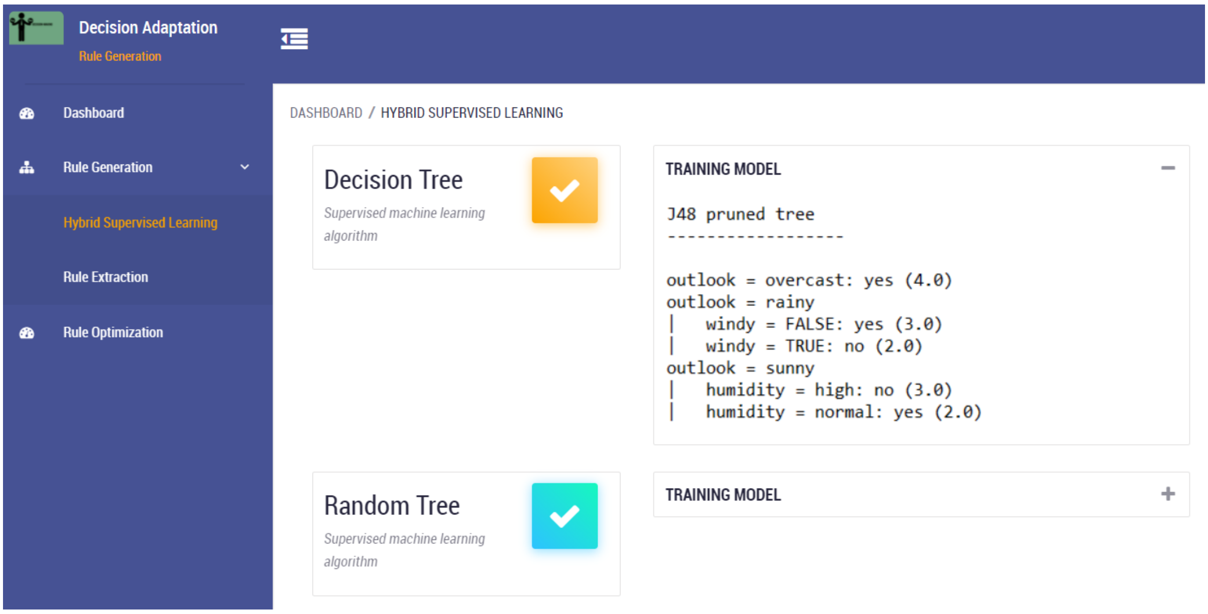

Figure 5. Tree-structured training model of the Decision Tree algorithm

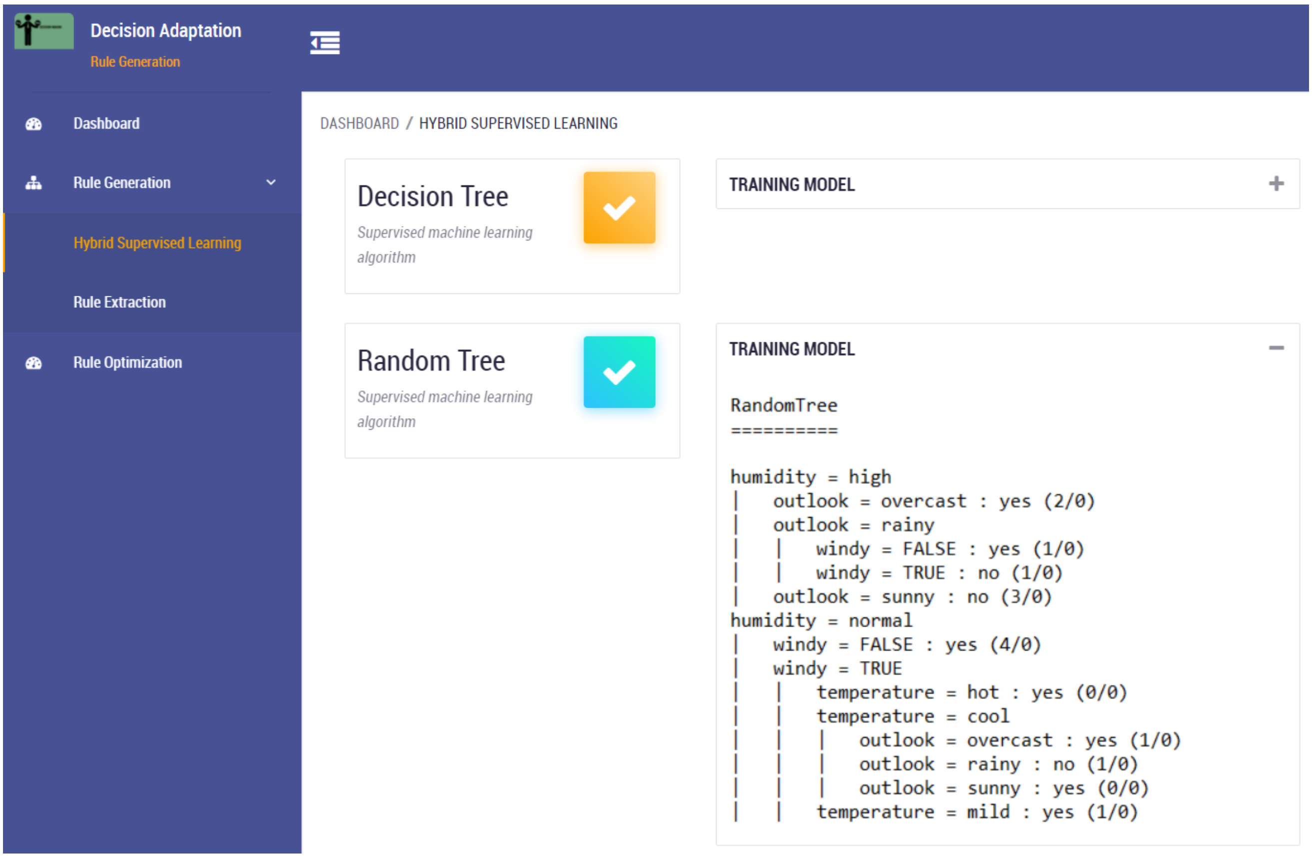

Figure 6. Tree-structured training model of the Random Tree algorithm

Next, a rule extraction is applied for automatically analyzing the obtained trees and extracting IF-THEN rules that are thoroughly validated via the different validation methods. Figure 7 and 8 present the validated IF-THEN rules from the Decision Tree training model and Random Tree training model, respectively.

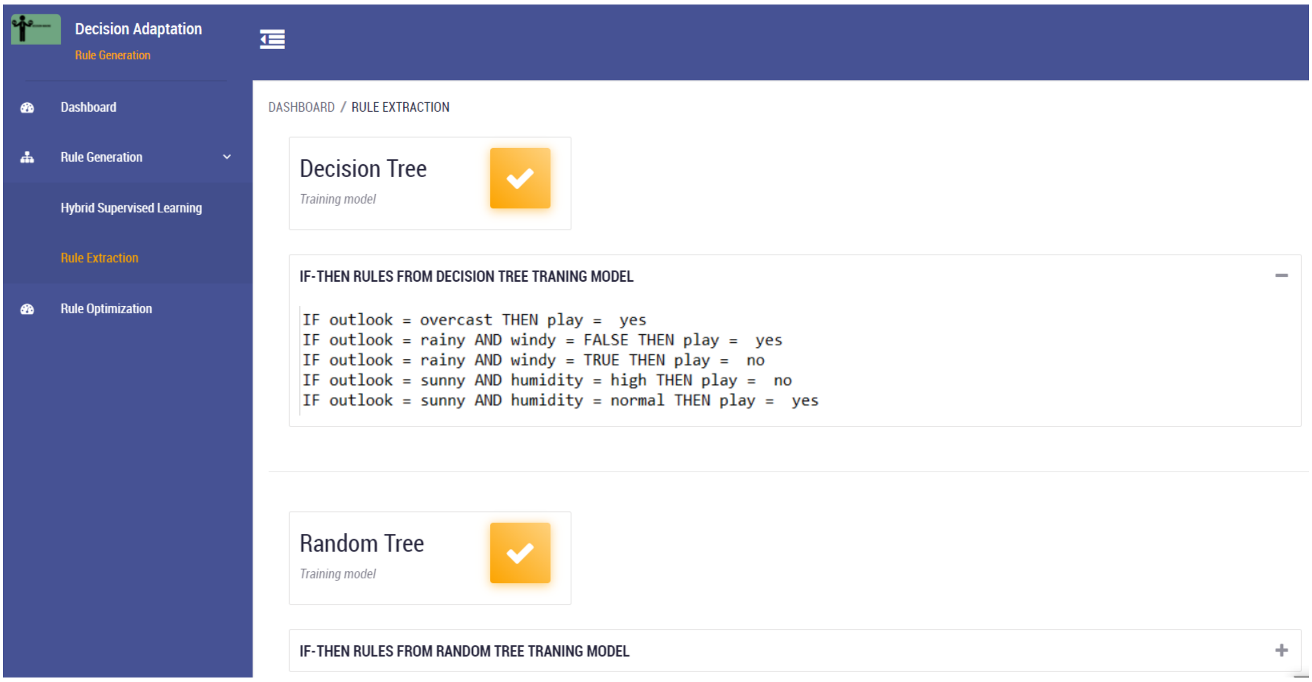

Figure 7. Validated IF-THEN rules from the Decision Tree-based training model

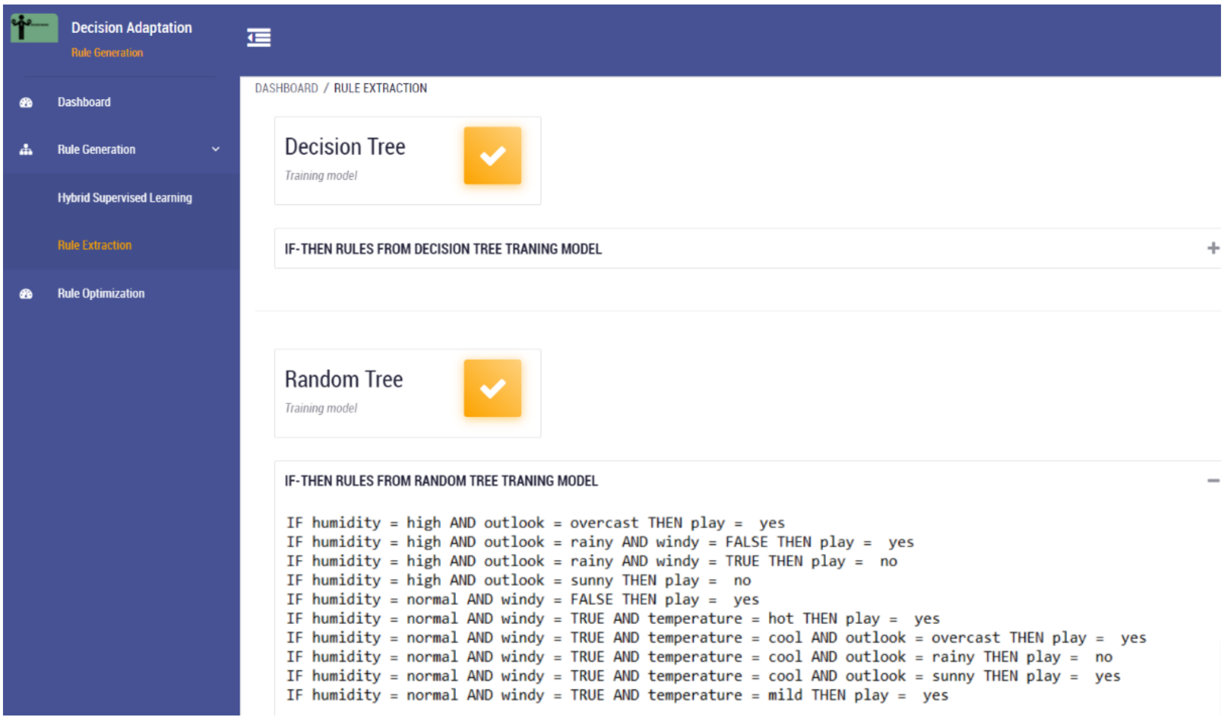

Figure 8. Validated IF-THEN rules from the Random Tree-based training model

Then, the GA is employed to find out the interesting rules. Afterward, the multi-analysis technique is performed to retrieve the well-performed rules. Figure 9 illustrates an excerpt of the well-performed rules.

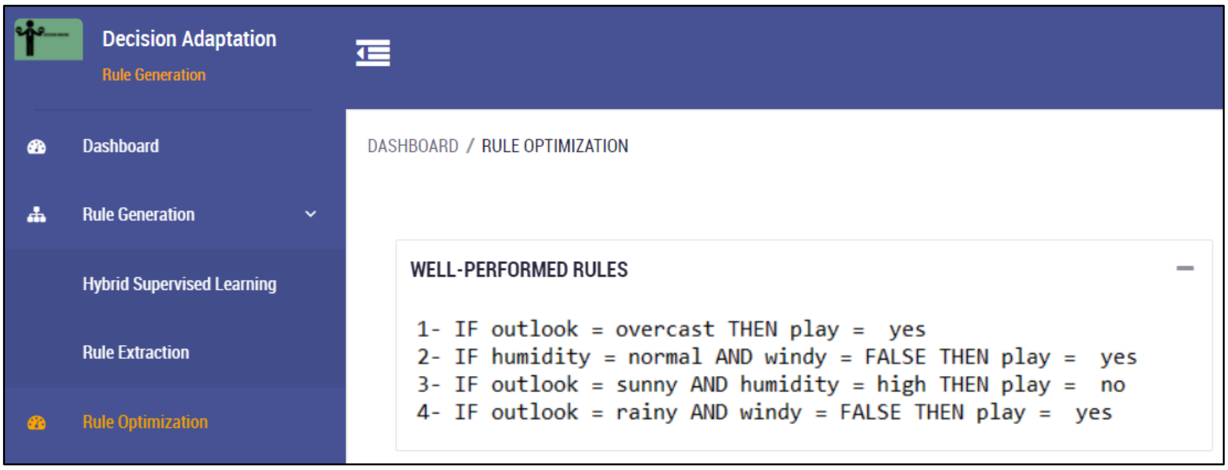

Figure 9. An excerpt of the well-performed rules

5. Experimental Evaluation and Results

In order to investigate the effectiveness of our proposed component, we have conducted a couple of experiments on a set of different candidate data sources. For this purpose, we first describe the experimental setup through a couple of Research Questions (RQs) that we aim to answer by the experiments. Then, we briefly introduce the candidate data sources. Finally, we present the experimental results with some discussion.

5.1. Experimental Setup

To measure the effectiveness, the following RQs have been framed:

- RQ.1. Is the proposed component able to eliminate the redundancy generated by existing traditional association rule mining algorithms and to generate a concise set of non-redundant and well-performed association rules?
- RQ.2. How effective is the proposed component relative to well-known decision tree machine learning algorithms as baselines?
- RQ.3. How effective is the proposed component relative to state-of-the-art approaches?

5.2. Candidate Data Sources

Our experiments are based on five different data sources taken from the UCI Machine Learning Repository [6]. The candidate data sources, namely “Weather”, “Lymphography”, “Breast Cancer”, “Restaurant” and “Adult”, are analyzed by diverse characteristics as illustrated in Table 1.

Table 1. Characteristics of candidate data sources used in experiments [6]

Data source	Attribute	Class	Instance
Weather	5	3	14
Lymphography	19	5	148
Breast Cancer	10	2	286
Restaurant	74	3	138
Adult	15	2	48842

5.3. Experimental Metric

For measuring the effectiveness of the proposed component, we discover and compute the total number of generated rules as well as the accuracy that represents both the conciseness as well as the rules’ quality. The accuracy is measured based on the relative number of correct rules and the total number of rules.

5.4. Experimental Results

Generated Rules Number Comparison

To answer the RQ.1, in this experiment, we show a relative comparison for the generated association rules using both our component and the traditional algorithms of association rule mining such as Apriori and FP-Growth, discussed in Section 2. Figures 10 to 14, show the relative comparison of the generated number of rules for different candidate data sources respectively. For this, we have explored different confidence thresholds in the range from 0.6 to 1. Since confidence is associated with rules’ accuracy, we are not interested in taking into account results that come out below 0.6 as confidence preference in our experimental analysis.

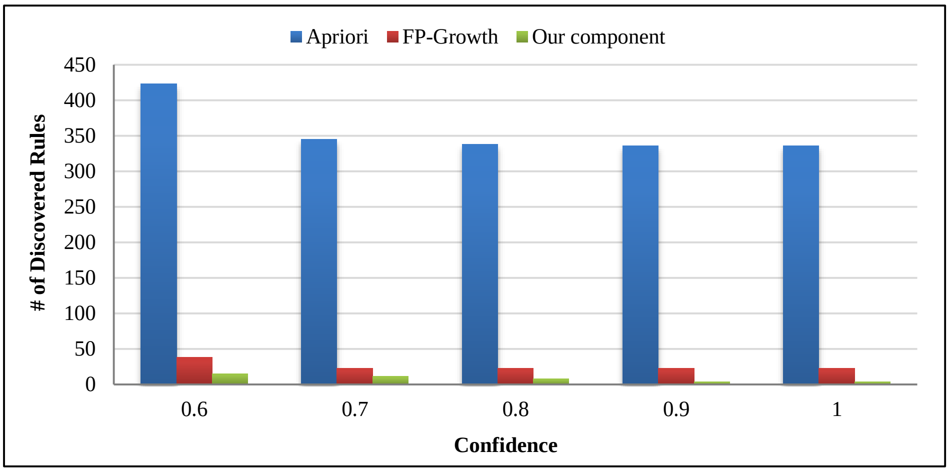

Figure 10. Rule number comparison for Weather data source

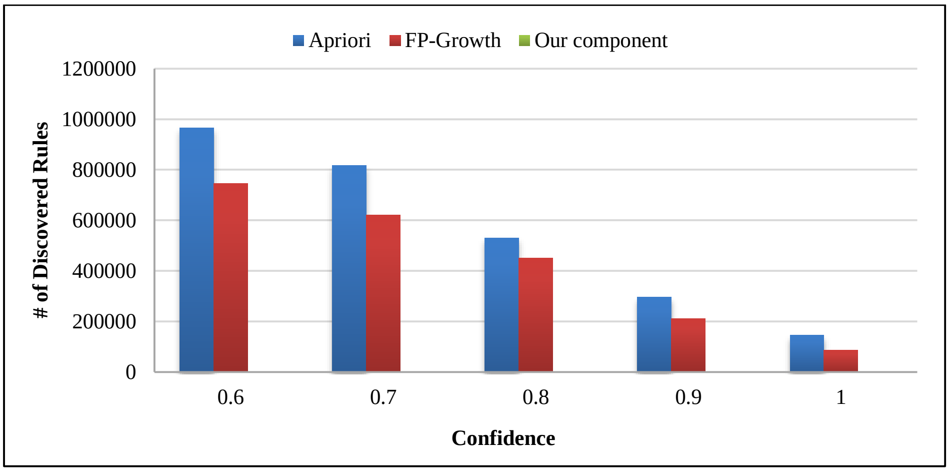

Figure 11. Rule number comparison for Lymphography data source

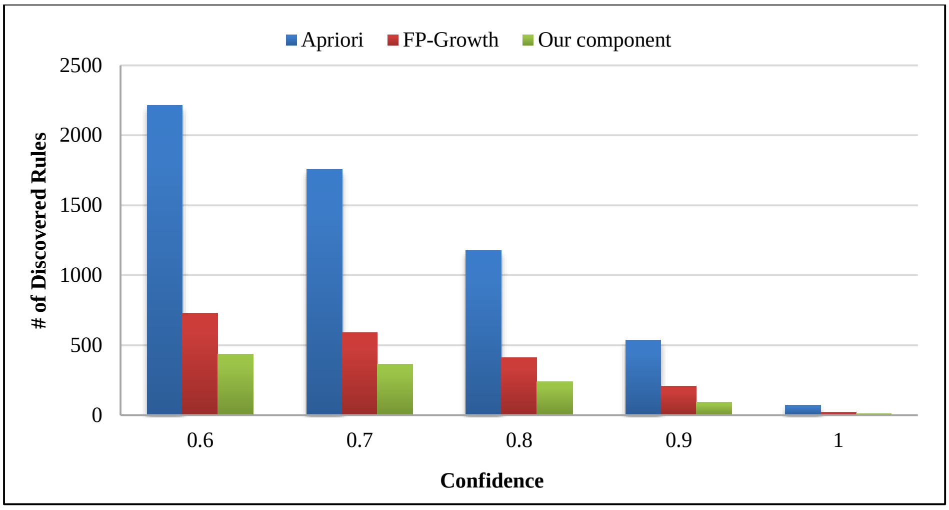

Figure 12. Rule number comparison for Breast Cancer data source

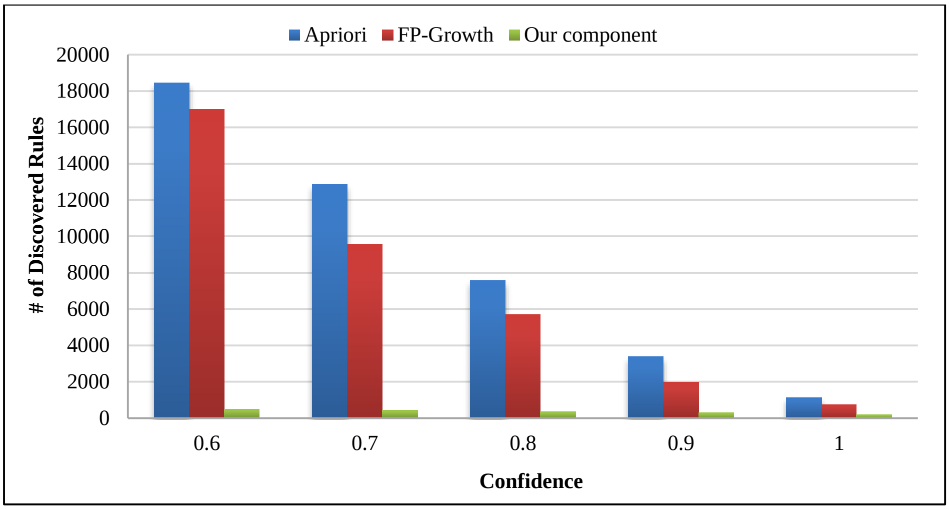

Figure 13. Rule number comparison for Restaurant data source

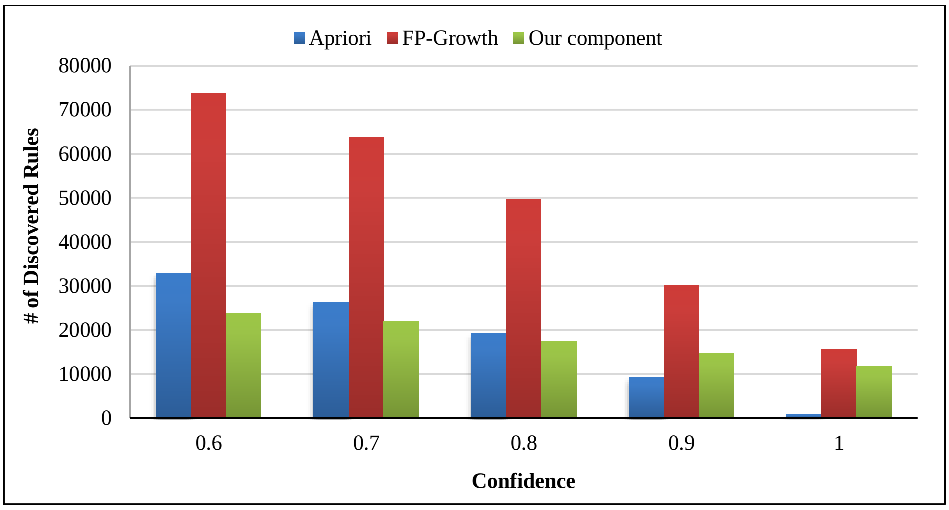

Figure 14. Rule number comparison for Adult data source

If we observe Figures 10 to 14, we see that the number of discovered association rules using traditional algorithms significantly increases with the decrease of confidence threshold. The reason beyond this result is that traditional algorithms simply take into account all combinations of attributes while generating rules. Thus, for a lower confidence value, they satisfy more associations. As a result, the generated set of rules becomes larger. Differently, the results show that, when the confidence thresholds are decreased, the number of discovered rules, in our component, is slightly increased. For higher confidence threshold, our component generates the minimum number of association rules comparing the Apriori and FP-Growth. In terms of non-redundant rule discovery, if we further observe Figures 10 to

14, we can see that the Apriori algorithm generates the highest number of rules, while our component generates the lowest. Moreover, the FPGrowth algorithm is better than Apriori in some cases, but it achieves a higher number of rules than our component. Therefore, our proposed component generates a reasonably smaller number of association rules and significantly reduces the total number of discovered rules compared with traditional association rule mining algorithms for the different confidence thresholds previously mentioned. The main reason is that the traditional association rule mining algorithms do not take into account the redundancy and the optimization analysis while generating rules. Therefore, they make the set of unnecessarily generated rules larger for a particular confidence preference.

Accuracy Comparison with Baselines

To answer the RQ.2, in this experiment, we show the effectiveness of our component based on the discovered association rules in terms of accuracy. For this purpose, well-known decision tree machine learning algorithms, namely BF Tree, REP Tree, Decision Stump and Simple Cart, are applied. We select these baselines as they are able to generate rules from candidate data sources. Table 2 shows the accuracy of these baselines and our component on the candidate data sources.

Table 2. Accuracy comparison with baselines on five datasets.

Accuracy (%)	Dataset				
	Weather	**Lymphography**	**Breast Cancer**	**Restaurant**	**Adult**
BF Tree	64.28	38.51	76.22	51.60	75.92
REP Tree	57.14	35.81	77.62	50.15	85.03
Decision Stump	78.57	37.83	74.12	50.56	84.54
Simple Cart	50	40	79	49	85
Our component	**86.71**	**56.15**	**82.70**	**64.87**	**91.42**

From Table 2, we find that our component consistently outperforms BF Tree, REP Tree, Decision Stump and Simple Cart algorithms for generating association rules by maximizing the accuracy. The reason is that we capture association rules from both more performant algorithms that improve the results. In addition, our observations reveal that our proposed component achieved better accuracy than BF Tree, REP Tree, Decision Stump and Simple Cart on “Weather”, “Lymphography”, “Breast Cancer”, “Restaurant” and “Adult” data sources. For instance, the accuracy for BF Tree, REP Tree, Decision Stump and Simple Cart on the “Adult” data source are 75.92%, 85.03%, 84.54% and, 85%, respectively, whereas for the proposed component the accuracy is 91.42%. Thus, these results proved that our proposed component tends to get reasonably high accuracy on all data sources. Therefore, we can conclude that our component is more effective relative to the well-known decision tree machine learning algorithms while generating rules.

Accuracy Comparison with State-of-the-art

To answer the RQ.3, in this experiment, we show the effectiveness of our component against three state-of-the-art approaches, including Zulkernain et al. [19], Sarker et al. [20] and Basha et al. [21]. For this purpose, the machine learning algorithms proposed in selected state-of-the-art approaches were implemented using the candidate datasets. A detailed accuracy comparison on the different candidate datasets is presented in Table 3.

Table 3. Accuracy comparison with state-of-the-art approaches on five datasets.

Accuracy (%)	Dataset				
	Weather	**Lymphography**	**Breast Cancer**	**Restaurant**	**Adult**
Zulkernain et al. [19]	74.36	73.85	79	77.53	85.80
Sarker et al. [20]	74.70	70.24	79.20	75.13	86.20
Basha et al. [21]	57.15	74.76	73.77	82.21	79.41
Our component	**78.57**	**77.60**	**79.25**	**86.80**	**87**

The comparative results, which are illustrated in Table 3, suggest that the accuracy of the proposed component is better than the results of other state-of-the-art approaches on five candidate datasets. Especially in the experiments on the Restaurant and Adult datasets, our component performs well and the accuracies achieved are 86.80% and 87%, respectively, which is higher than previous approaches. Moreover, due to the hybridization of decision tree machine learning algorithms, our component achieves a 5.65% higher accuracy compared to state-of-the-art approaches on the different candidate datasets. From these results, it is worth noting that we can safely draw the conclusion that our component can reach a better accuracy compared with state-of-the-art.

5.5. Discussion

We will discuss the encouraging results obtained from the present component. Firstly, our component outperformed compared to the traditional association rule mining algorithms in terms of reducing the number of rules by eliminating the redundant generation for each data source. It generated, on average, far fewer association rules than those generated by the traditional algorithms included in the comparison. Thus, it provided a reasonably smaller number of rules on smaller and bigger data sources compared to the well-known traditional algorithms. This is due to the fact that our component is based on hybrid supervised learning that takes advantage of machine learning and pattern recognition to learn about data, to extract relevant relationships, and to eliminate redundancy while generating association rules. In brief, our component significantly reduces the total number of discovered rules and outputs a well-performed set of association rules using the GA. Such non-redundant and well-performed rule generation makes our component more effective. Secondly, based on the second experiment, we can conclude that the effectiveness of the discovered rules is also improved compared to the well-known decision tree machine learning algorithms. In particular, our component achieved the best accuracy on "Weather", "Lymphography", "Breast Cancer", "Restaurant" and "Adult" data sources among all selected decision tree machine learning algorithms. However, it got slightly worse results in terms of accuracy on "Restaurant" and "Lymphography" data sources compared to other candidate data sources, since imbalanced data sources may lead to slightly worse accuracy results. Even though not achieving the best accuracy results on "Restaurant" and "Lymphography" data sources, our component achieved almost higher results with all selected decision tree machine learning algorithms on the remaining data sources. Finally, through the third experiment, we confirm the effectiveness of our component over state-of-the-art approaches. The obtained results proved that our component leads to approximately 5.65% accuracy gain when compared to the state-of-the-art approaches, mainly due to the hybridization of decision tree machine learning algorithms. Overall, the presented component allows us to answer the three previously mentioned research questions, RQ1, RQ2 and RQ3. More specifically, the findings of the experimental study reveal that our component (i) can effectively minimize the issues of redundant rule generation and give high accuracy by extracting a concise set of well-performed association rules and (ii) is applicable to more accurately generating rules in a context-aware application at runtime.

6. Conclusion

In this paper, we proposed a component that aims to generate rules without engineers' interventions in reaction to dynamic, ubiquitous computing environments at runtime. This component first starts with hybrid supervised learning and then a GA with multi-analysis technique is applied to support rule optimization. Furthermore, we briefly presented a case study-based application for engineers targeting first the automatic IF-THEN rule generation from a data source and then the rule optimization to improve decision-making at runtime. As part of this work, we present an experimental evaluation to assess the effectiveness of the proposed component. The results showed that the proposed component has the potential for generating association rules to further enrich a rule knowledge base and for achieving the best results in terms of number of rules and accuracy among selected algorithms and state-of-the-art approaches. However, there is still a long way to go before using our component in a real-life scenario since there are some limitations in terms of rule applicability. In the future, we plan to extend our component with the possibility to automatically transform the IF-THEN rules into rules in the syntax of Jena since we are adopting ontology-based context models that are managed within a Jena-based platform equipped with an embedded reasoner.

7. Declarations

7.1. Author Contributions

Conceptualization, R.J.; methodology, R.J.; software, R.J.; validation, R.J.; formal analysis, R.J.; investigation, R.J.; writing—original draft preparation, R.J.; writing—review and editing, R.J., M.K. and F.B.; visualization, S.F.; supervision, M.K. and F.B. All authors have read and agreed to the published version of the manuscript.

7.2. Data Availability Statement

The data presented in this study are available in article.

7.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

7.4. Institutional Review Board Statement

Not applicable.

7.5. Informed Consent Statement

Not applicable.

7.6. Declaration of Competing Interest

The authors declare that there is no conflict of interests regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

8. References

[1] Singh, S. (2021). Applications in Ubiquitous Computing. In R. Kumar & S. Paiva (Eds.), EAI/Springer Innovations in Communication and Computing. Springer, Cham, Switzerland. doi:10.1007/978-3-030-35280-6.

[2] Castro Garrido, P., Luque Ruiz, I., & Gómez-Nieto, M. Á. (2014). OBCAS: An agent-based system and ontology for mobile context aware interactions. Journal of Intelligent Information Systems, 43(1), 33–57. doi:10.1007/s10844-014-0305-8.

[3] Zhao, T. (2016). The Generation and Evolution of Adaptation Rules in Requirements Driven Self-Adaptive Systems. Proceedings - 2016 IEEE 24th International Requirements Engineering Conference, RE 2016, 456–461. doi:10.1109/RE.2016.18.

[4] Liu, Y., Zhang, W., & Jiao, W. (2016). A generative genetic algorithm for evolving adaptation rules of software systems. ACM International Conference Proceeding Series, 18-September-2016, 103–107. doi:10.1145/2993717.2993731.

[5] Goldberg, D. (1989). Genetic algorithms in search. In Optimization and machine learning. Addison-Wesley Longman Publishing Co., Boston, MA, United States.

[6] Asuncion, A., & Newman, D. (2007). UCI Machine Learning Repository. Irvine University of California, Irvine, CA, United States.

[7] Paiva, L., Costa, R., Figueiras, P., & Lima, C. (2014). Discovering semantic relations from unstructured data for ontology enrichment: Asssociation rules based approach. Iberian Conference on Information Systems and Technologies, CISTI, 1 6. doi:10.1109/CISTI.2014.6877008.

[8] Han, J., Pei, J., & Yin, Y. (2000). Mining frequent patterns without candidate generation. ACM SIGMOD Record, 29(2), 1–12. doi:10.1145/335191.335372

[9] Idoudi, R., Ettabaa, K. S., Solaiman, B., & Mnif, N. Association rules-based ontology enrichment. International Journal of Web Applications, 8(1), 16–25.

[10] Agrawal, R., & Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. Proc. of the 20th International Conference on Very Large Data Bases (VLDB'94), 1215, 487–499.

[11] Chang, M., D'Aniello, G., Gaeta, M., Orciuoli, F., Sampson, D., & Simonelli, C. (2020). Building Ontology-Driven Tutoring Models for Intelligent Tutoring Systems Using Data Mining. IEEE Access, 8, 48151–48162. doi:10.1109/ACCESS.2020.2979281.

[12] Gabroveanu, M., & Diaconescu, I. M. (2008). Extracting semantic annotations from moodle data. CEUR Workshop Proceedings, 428, 1–5.

[13] Kaliappan, J., Sai, S. M., & Shaily Preetham, K. (2019). Weblog and retail industries analysis using a robust modified Apriori algorithm. In International Journal of Innovative Technology and Exploring Engineering 8, (6), 1727–1733.

[14] Davagdorj, K., & Ryu, K. H. (2018). Association Rule Mining on Head and Neck Squamous Cell Carcinoma Cancer using FP Growth algorithm. Proceedings of the International Conference on Information, System and Convergence Applications, 31 Jan – 2 Feb 2018, Bangkok, Thailand.

[15] Asadianfam, S., Kolivand, H., & Asadianfam, S. (2020). A new approach for web usage mining using case based reasoning. SN Applied Sciences, 2(7), 1–11. doi:10.1007/s42452-020-3046-z.

[16] Miswan, N. H., Sulaiman, I. M., Chan, C. S., & Ng, C. G. (2021). Association rules mining for hospital readmission: A case study. Mathematics, 9(21), 2706. doi:10.3390/math9212706.

[17] Islam, M. R., Liu, S., Biddle, R., Razzak, I., Wang, X., Tilocca, P., & Xu, G. (2021). Discovering dynamic adverse behavior of policyholders in the life insurance industry. Technological Forecasting and Social Change, 163, 120486. doi:10.1016/j.techfore.2020.120486.

[18] Sánchez-de-Madariaga, R., Martinez-Romo, J., Escribano, J. M. C., & Araujo, L. (2022). Semi-supervised incremental learning with few examples for discovering medical association rules. BMC Medical Informatics and Decision Making, 22(1), 1–11. doi:10.1186/s12911-022-01755-3.

[19] Zulkernain, S., Madiraju, P., & Ahamed, S. I. (2010). A mobile intelligent interruption management system. Journal of Universal Computer Science 16(15), 2060-2080.

[20] Sarker, I. H. (2019). A machine learning based robust prediction model for real-life mobile phone data. Internet of Things (Netherlands), 5, 180–193. doi:10.1016/j.iot.2019.01.007.

[21] Basha, M.S. (2021). Early Prediction of Cardio Vascular Disease by Performing Associative Classification on Medical Datasets and Using Genetic Algorithm. In Lecture Notes in Networks and Systems 248, 393–402. doi:10.1007/978-981-16-3153-5_42.

[22] Mahmood, A., Shi, K., Khatoon, S., & Xiao, M. (2013). Data mining techniques for wireless sensor networks: A survey. International Journal of Distributed Sensor Networks, 2013(7), 406316. doi:10.1155/2013/406316.

[23] Jabla, R., Khemaja, M., Buendia, F., & Faiz, S. (2021). Automatic ontology-based model evolution for learning changes in dynamic environments. Applied Sciences (Switzerland), 11(22), 10770. doi:10.3390/app112210770.

[24] Quinlan, J. R. (1986). Induction of Decision Trees. Machine Learning, 1(1), 81–106. doi:10.1023/A:1022643204877.

[25] Witten, I. H., Frank, E., & Geller, J. (2002). Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. SIGMOD Record, 31(1), 76–77. doi:10.1145/507338.507355.

[26] Suresh, A., Udendhran, R., & Balamurgan, M. (2020). Hybridized neural network and decision tree based classifier for prognostic decision making in breast cancers. Soft Computing, 24(11), 7947–7953. doi:10.1007/s00500-019-04066-4.

[27] Fielding, R. T., & Taylor, R. N. (2002). Principled Design of the Modern Web Architecture. ACM Transactions on Internet Technology, 2(2), 115–150. doi:10.1145/514183.514185.