Development of a Causality Least Association Rules Algorithm Tool Using Rational Unified Process Methodology

Zailani Abdullah, Fatihah Mohd, Amir Ngah, Ang Bee Choo, Nabilah Huda Zailani, and Wan Aezwani Wan Abu Bakar

Abstract Among the most crucial research areas in data mining is association rule mining (ARM). Rules are classified into two types: frequent rules and least frequent rules. Extracting the least association rules is more difficult and always leads to the "rare item problem" quandary. The rules with the fewest items are known as the "least association rules." However, most data mining tools favour frequent association rules over the least frequent association rules. Furthermore, the process of extracting the least association rules is more difficult. Therefore, this paper proposes and develops Causality Least Association Rules Algorithm Tool (CLART) using the Rational Unified Process (RUP) methodology and the C# programming language. The results showed that CLART is workable, and the proposed algorithm also outperformed the existing benchmark algorithm. In addition, CLART is a dedicated tool that is freely available and can be used to extract the causality least association rules from the benchmarked datasets.

Z. Abdullah (✉) · F. Mohd
Faculty of Entrepreneurship and Business, Universiti Malaysia Kelantan, 16100 Kota Bharu, Kelantan, Malaysia
e-mail: zailania@umk.edu.my

F. Mohd
e-mail: fatihah.m@umk.edu.my

A. Ngah · A. B. Choo
Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia
e-mail: amirnma@umt.edu.my

A. B. Choo
e-mail: abcais_07@yahoo.com.my

N. H. Zailani
Faculty of Social Sciences and Humanities, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
e-mail: hudaazailani@gmail.com

W. A. Wan Abu Bakar
Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, 21030 Kuala Nerus, Terengganu, Malaysia
e-mail: wanaezwani@unisza.edu.my

M. N. H. Yusoff (ed.), *Industry Forward and Technology Transformation in Business and Entrepreneurship*, https://doi.org/10.1007/978-981-99-2337-3_50

Keywords Least · Algorithm · Association rules · Rational unified process

1 Introduction

Rational Unified Process (RUP) [1] is a kind of agile software development methodology. It was invented by the Rational Software Corporation, one of IBM's divisions, in 2003 for an iterative software development process framework [2]. RUP is also referred to as a software engineering process that relies on a web-enabled, searchable knowledge base [3, 4].

Association Rule Mining (ARM) is among the most widely used algorithms in data mining. Today, ARM is still active and has attracted a lot of attention from researchers in the field of data mining [4–13]. ARM is typically used to reveal all association rules [14] with support and confidence values greater than predefined minimum support and confidence [15]. However, most of the algorithms indirectly ignore the occurrence of the least association rules. In other words, by using the typical minimum support in the algorithm, it will accidentally exclude the least association rules.

The least association rule refers to an association rule forming between the least frequent items or among the least items. The presence of these items in relation to least association rules in some disciplines is extremely significant and necessitates close attention. For example, to identify relatively rare diseases, predict telecommunication equipment failure, find abnormal reactions in nuclear plants, and find associations between the least purchased items [16].

The Bayesian Network (BN) [17] is a graphical model that encapsulates probabilistic correlations between the relevant variables. BN comprises statistical techniques that blend together domain knowledge and data. Causal semantics in BN makes the encoding of causal prior knowledge extremely simple. Additionally, BN also uses probabilities to express the strength of causal linkages [18].

However, most of the existing data mining tools, such as WEKA [19], RapidMiner [20], H20 [21], etc., are focusing more on extracting the frequent association rules. In reality, the extraction of the fewest association rules is not simple and is frequently plagued by the problem of a computer's memory overflow. This paper proposes an improved algorithm called Causality Least Association Rules, which is based on Apriori [22] and .Net Framework 3.5 [23]. Various ranges of predefined minimum support thresholds are employed to discover these rules. RUP methodology was employed rather than the typical Waterfall Methodology for developing the Causality Least Association Rules Algorithm Tool (CLART).

The organisation of the paper is as follows. Section 2 explains work done by others. Section 3 highlights the basic concepts and terminology. Section 4 focuses on the methodology for developing the algorithm and its tool. Section 5 elaborates on the result and discussion, particularly the performance of the developed tool. Finally, the paper is concluded with a short summary in Sect. 6.

2 Related Works

As noted in the introduction, the Apriori algorithm is a fundamental and popular algorithm for ARM. This algorithm was proposed by Agrawal et al. [22]. It mines frequent item sets using prior knowledge of frequent item properties. In terms of execution, it uses a level-wise search, an iterative method that uses k-itemsets to examine $(k + 1)$ items.

Koh and Rountree [24] suggested a method to find the least rules with candidate itemsets that are between minimum and maximum support values. They developed an algorithm known as Apriori-Inverse to quickly discover sporadic rules with a number of variations, such as the rare connection of two frequent symptoms pointing to a rare disease.

Koh et al. [25] proposed the RSAA algorithm to construct rules in which significant rare itemsets take part without any "magic numbers" given by the user. In this method, relative support (RSup) is used in place of minimal support. The support threshold is lowered by this method for items with a very low frequency and raised for things with a high frequency.

WEKA (the Waikato Environment for Knowledge Analysis) is one of the most prominent open-source machine learning and data mining algorithms. It used Java programming by the University of Waikato, New Zealand. Another free and open-source tool for data and text mining is called RapidMiner. It is one of the most popular data science tools. H_2O is a piece of open-source software developed by the H_2O.ai Company. It provides heterogeneous, conventional analytics mechanisms.

3 Basic Concepts and Terminology

3.1 Association Rules

The association rule is a statement of the form {X1, X2, …, Xn} => Y, meaning that if X1, X2, …, Xn are all found in the market basket, then there is a good chance of finding Y. The likelihood of discovering Y is called the confidence of the rules. Typically, only rules with confidence levels greater than a specific threshold are maintained.

The association rule is a statement of the form {X1, X2, …, Xn} => Y, indicating that if X1, X2, …, Xn are all discovered in the market basket, there is a strong likelihood of finding Y.

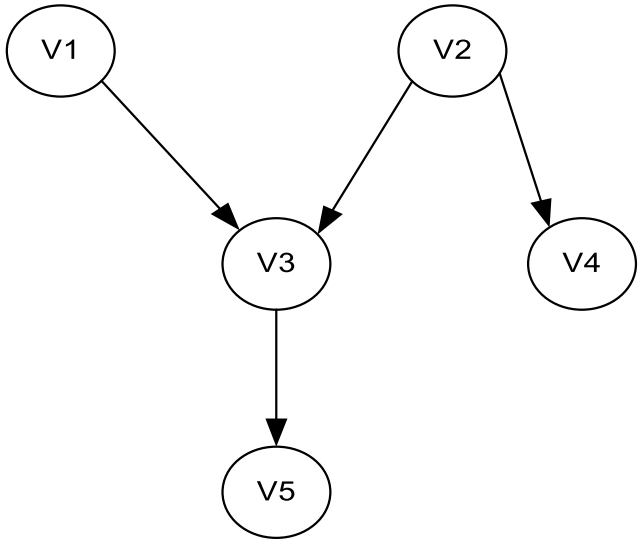

Fig. 1 The five attributes of the Bayesian network

3.2 Apriori Algorithm

Apriori [22] employed the large itemset property, whereby any subset of a large itemset must also be large. The large itemsets are also referred to as "downward closed," because if an itemset satisfies the minimum support requirements, all of its subsets are also applied. The fundamental principle of the Apriori algorithm is the construction of candidate itemsets based on a specific size (n), followed by a database search to count them and evaluate whether they are large or not.

3.3 Bayesian Network

The Bayesian Network (BN) [26] is a model that makes use of conditional probabilities among several variables. It is generally impossible to generate all conditional probabilities from a given dataset. Informally, the vertex set and directed edge set of BN serve as representations of an enhanced directed acyclic graph. An example of the five attributes of BN is shown in Fig. 1.

3.4 Least Association Rules

In some cases, it may be very interesting to look for the fewest itemsets, that is, itemsets that do not appear frequently in the data. To extract the least association rules, they should at least comply with two requirements; first, it should be the minimal and simplest association rule set, and second, its predictive power should not be weaker than the complete association rules.

4 Methodology

The Relational Unified Process (RUP) [1] approach has been adopted for creating CLART. RUP is a well-defined system process, often used to develop software based on object-based and/or component-based technologies. RUP is among the modern process models derived from the Unified Modeling Language (UML) and Associated Software Development Process [27]. RUP divides software development into four stages: inception, elaboration, building, and transition. Each stage includes a single or more executable iterations of the software at that level of development.

4.1 Inception Phase

The business case is established at the end of this phase. To demonstrate the primary functionalities provided by the causality least association rules, a basic use case diagram is created. Figure 2 depicts a use case diagram with one (1) actor named "User" and five (5) main use-cases: "load datasets", "generate association rules using the Apriori algorithm", "generate least association rules using RSAA", and "analyse performance". First, open the datasets, which are in text file format, as an input to generate association rules. Three algorithms are developed: Apriori, CLART, and RSAA.

4.2 Elaboration Phase

The elaboration phase is where the developers examine the project more thoroughly. It involves an analysis of how the development of CLART workflow is performed. The basic data characteristics and restrictions that may occur during the development of an algorithm will be taken into consideration. All related diagrams, such as relational entity classes, diagrams, activity diagrams, basic GUIs, and Gantt charts, are produced to provide a clear perspective on the project.

Figure 3 shows the activity diagram for generating association rules using a standard algorithm use case. An association rule is generated when it fulfils the minimum support and minimum confidence values. Users are required to determine what the minimum support and minimum confidence values are. The user needs to select the dataset for generating the association rules. All single items in the dataset are counted and then compared to the minimum support value threshold. The items that satisfy this threshold will be used in generating the association rules. Then, these association rules will be filtered by a minimum confidence threshold value.

Figure 4 shows the activity diagram for generating the least association rules using the least Apriori algorithm Least items are generated once they satisfy the interval support, which includes minimum and maximum support. Items that fall within the

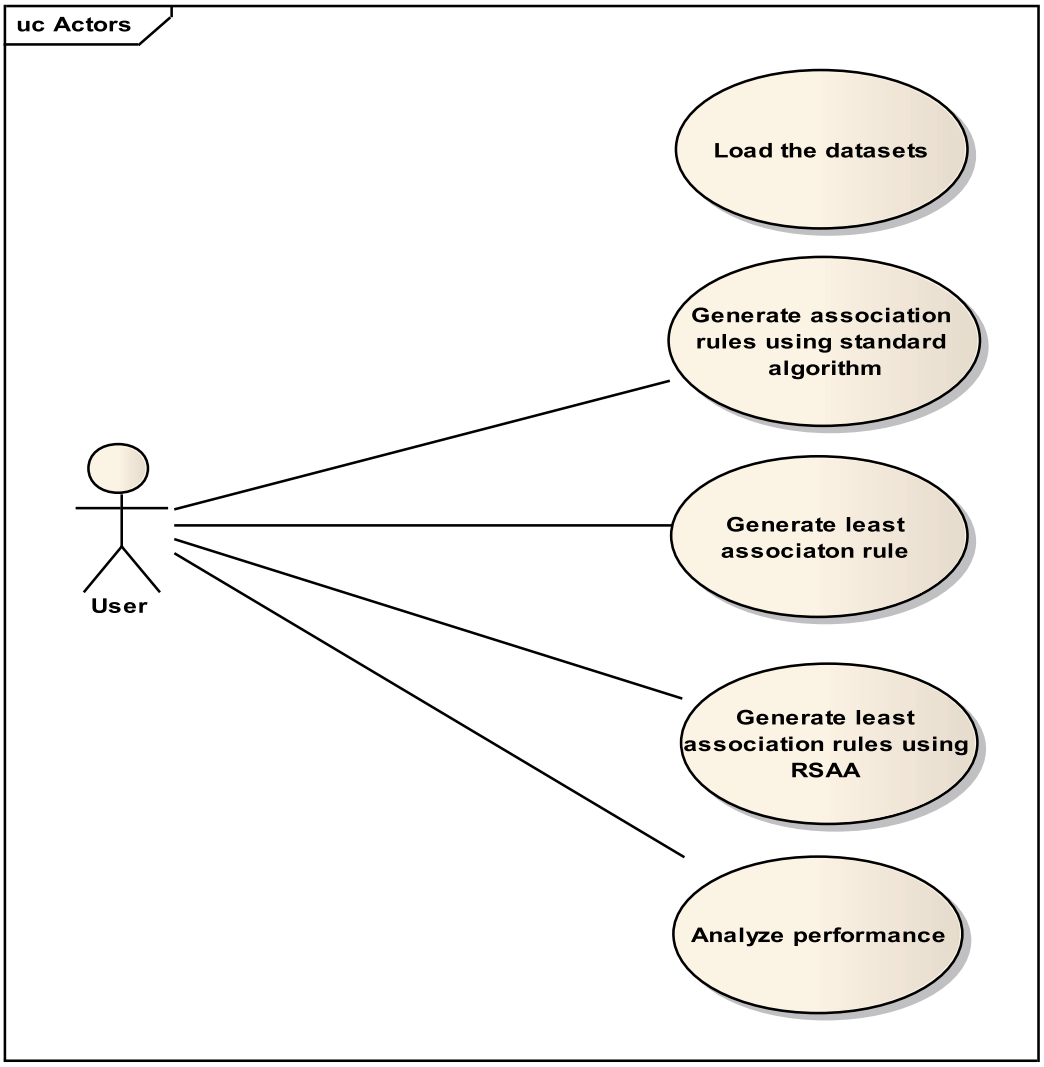

Fig. 2 Use case diagram

interval support range are considered the least. Least frequent items and frequent items are then joined together to form the desired association rules. Any association rule that fails to fulfil the minimum support and Bayesian network values will be pruned out.

Figure 5 shows the activity diagram for generating the least association rules using RSAA. RSAA uses two supports: the first is to find the frequent item and the second is to find the least frequent item. An item's support is compared to its first and second supports. The joining step is used to generate the least association rules, which are then compared to minimum and relative support.

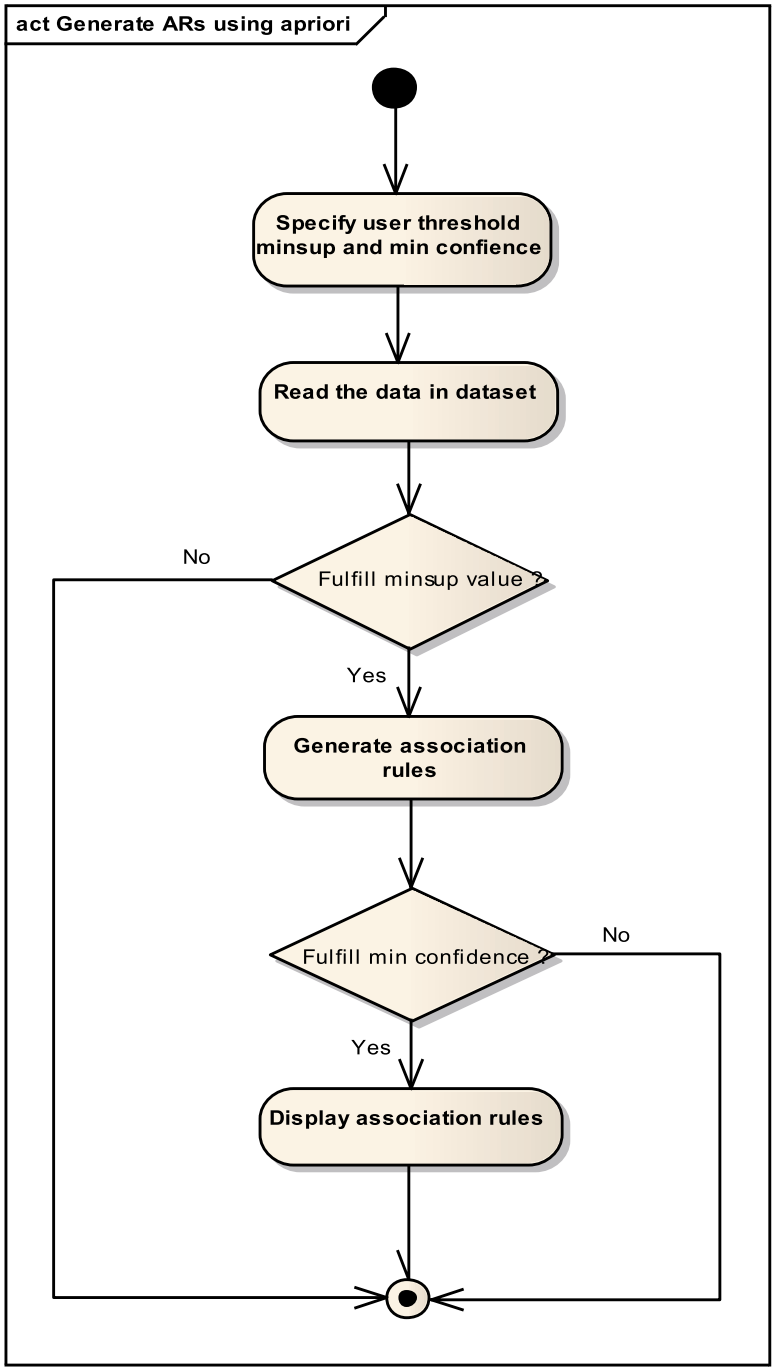

Fig. 3 Activity diagram to generate association rules using standard algorithm

4.3 Construction Phase

All codes are created in Microsoft C# at this point. A thorough model illustrating the fundamental workflow is produced. The model makes it very obvious how the Microsoft C# programming language is used to link and interact with the data and

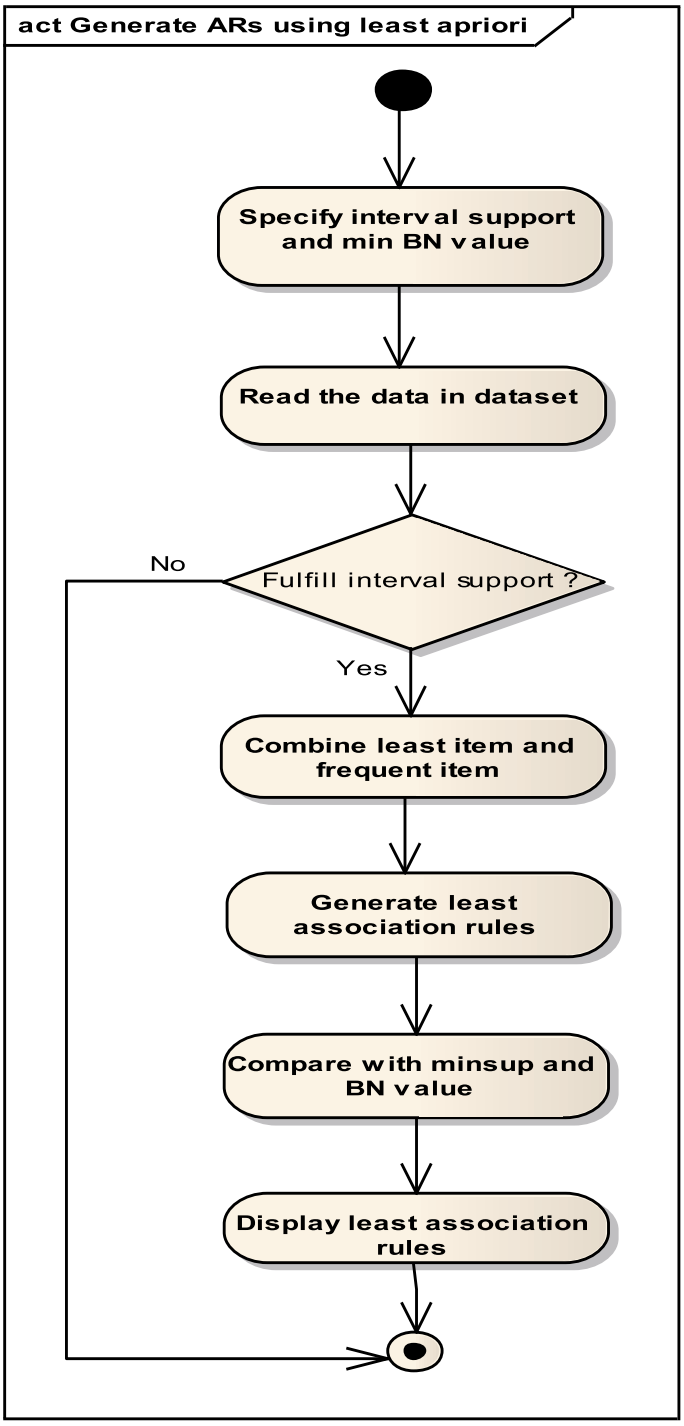

Fig. 4 Activity diagram to generate least association rules using Least Apriori

procedures in the causality least association rules algorithm. Figure 6 depicts the overall model of causality development using the least association rules together with the least Apriori algorithm.

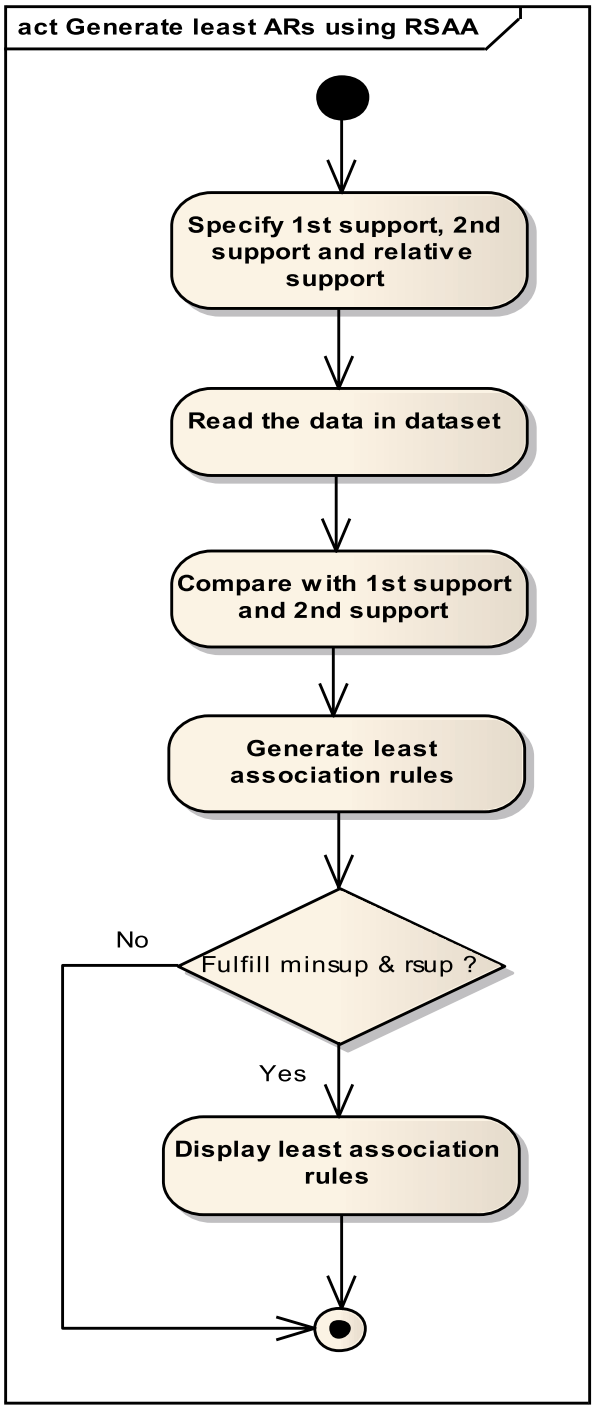

Fig. 5 Activity diagram to generate least association rules using RSAA

There are three (3) phases involved in the development of causality least association rules. Basically, it covers the generation of association rules based on three algorithms: the standard algorithm (Apriori), the Least Apriori algorithm, and the RSAA. The association rules are generated by comparing the first support, second support, and relative support. After the phases are completed, the performance (processing time) of three (3) algorithms is compared.

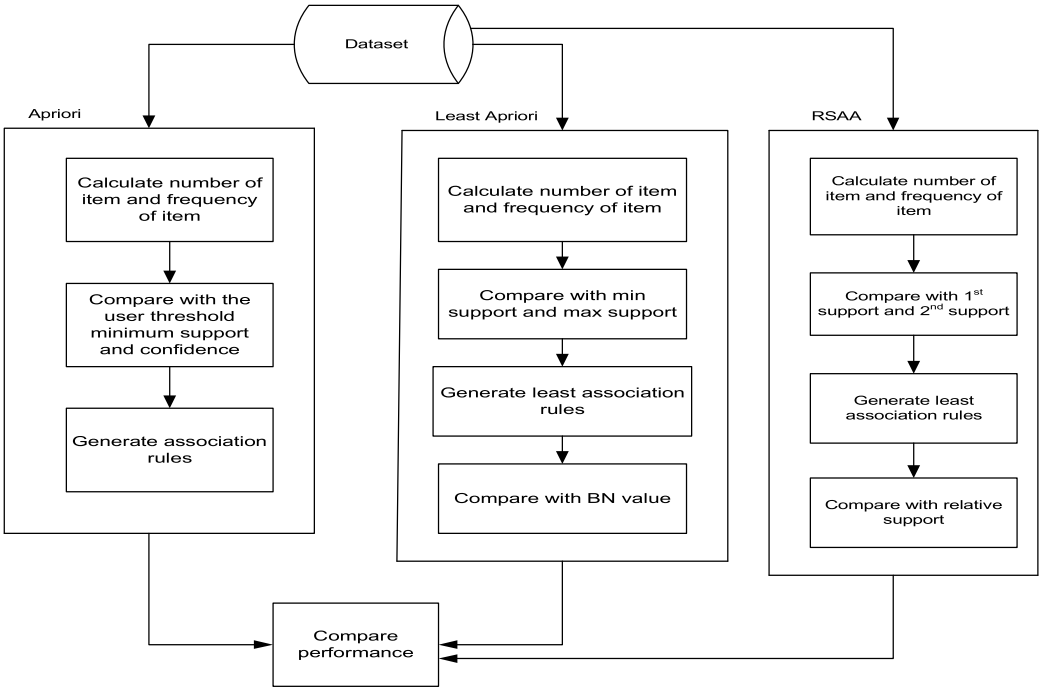

Fig. 6 An overview model of CLART

Phase 1 is to generate association rules using the standard Apriori algorithm. The association rule is generated by comparing the minimum support and minimum confidence, which are predefined by the user. Phase 2 is the development of causality least association rules using the least Apriori algorithm. Less frequently occurring but highly associated itemsets are generated. However, phase 3 is to generate the least association rules using RSAA. The association rules are generated by comparing the first support, second support, and relative support. After the phases are completed, the performance of two (3) algorithms is compared. An overview of the model is presented in Fig. 6.

4.4 Transition Phase

In the final phase, the tool is developed. To make sure there are no defects or mistakes, the testing process will be run many times. Coding fixes will be made until all problems and errors have been fixed, if any are found during testing. Not only will the function be tested, but the user interfaces will also be considered and may be commented on and changed to meet the needs.

5 Results and Discussion

The experiments were run on a PC with the specifications of a Core i7-8565U processor, 12 GB of RAM, a 512 GB SSD, and the Windows 10 operating system. The characteristics of the Mushroom datasets used in experiments are shown in Table 1.

Figure 7 shows association rules generated using the standard algorithm (Apriori). The processing time and number of rules generated are also shown.

Figure 8 depicts the RSAA algorithm's least association rules. The processing time and number of rules generated are also shown. The causality least association rules generated by the Least Apriori algorithm are shown in Fig. 9. The processing time and the number of rules generated are also shown.

Figure 10 shows the overall result of the performance analysis according to the three (3) algorithms. The processing time and number of generated rules are also displayed. The processing time for both the Least Apriori algorithm and RSAA to generate association rules is significantly faster than that of the classical or standard algorithm (Apriori). Both the Least Apriori algorithm and the RSAA have the same

Table 1 Characteristics of the mushroom datasets

Size	No. of transaction	No. of items	Avg item per transaction	Min item in transaction	Max item in transaction
558 kb	8124	119	68	23	23

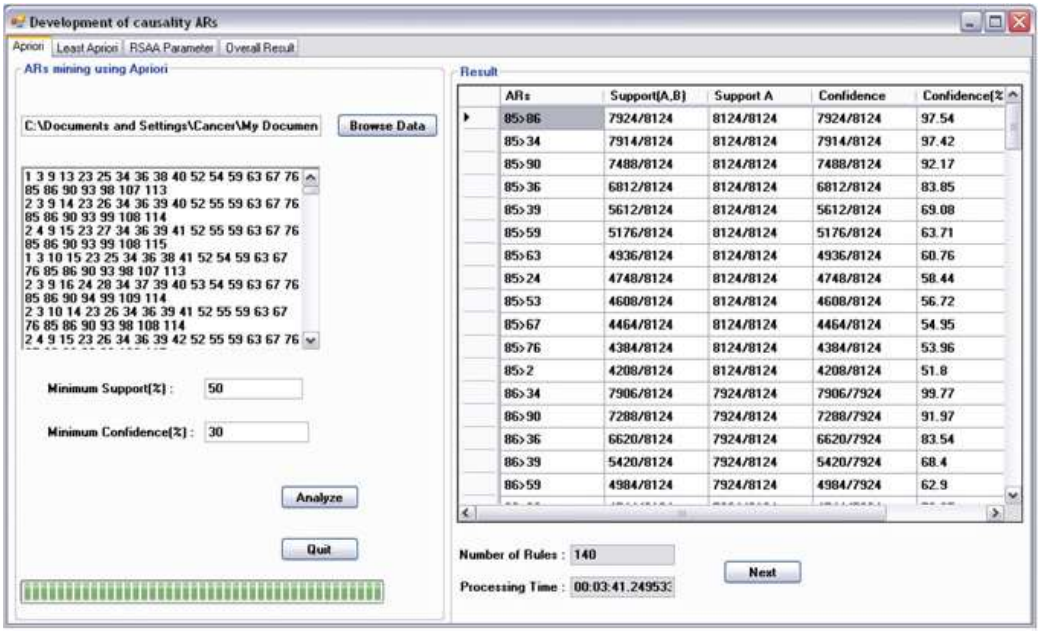

Fig. 7 Association rules generated by Apriori algorithm

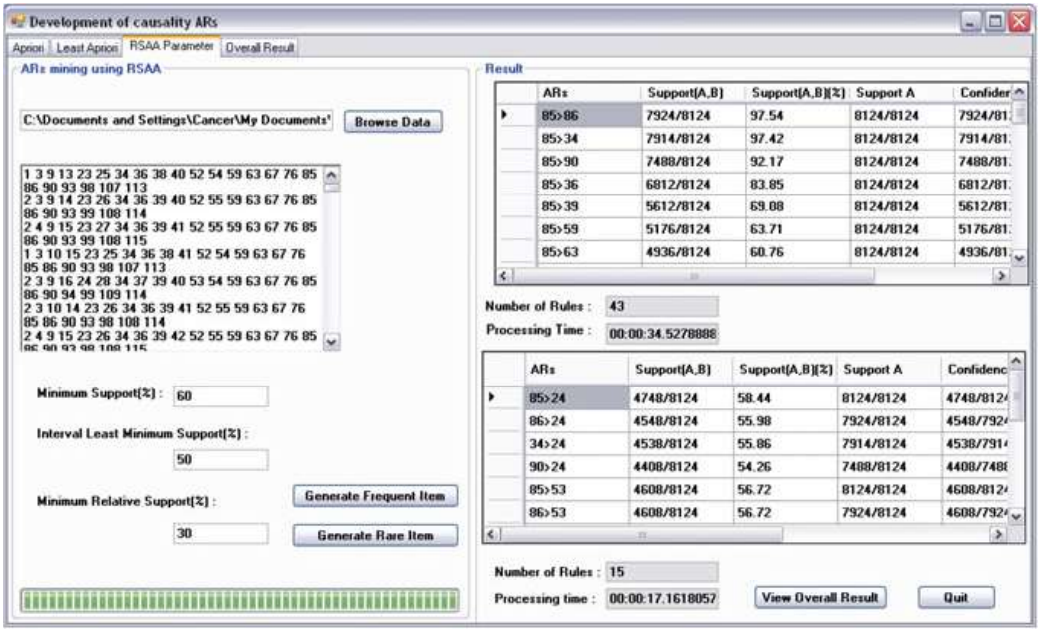

Fig. 8 Least association rules generated by RSAA

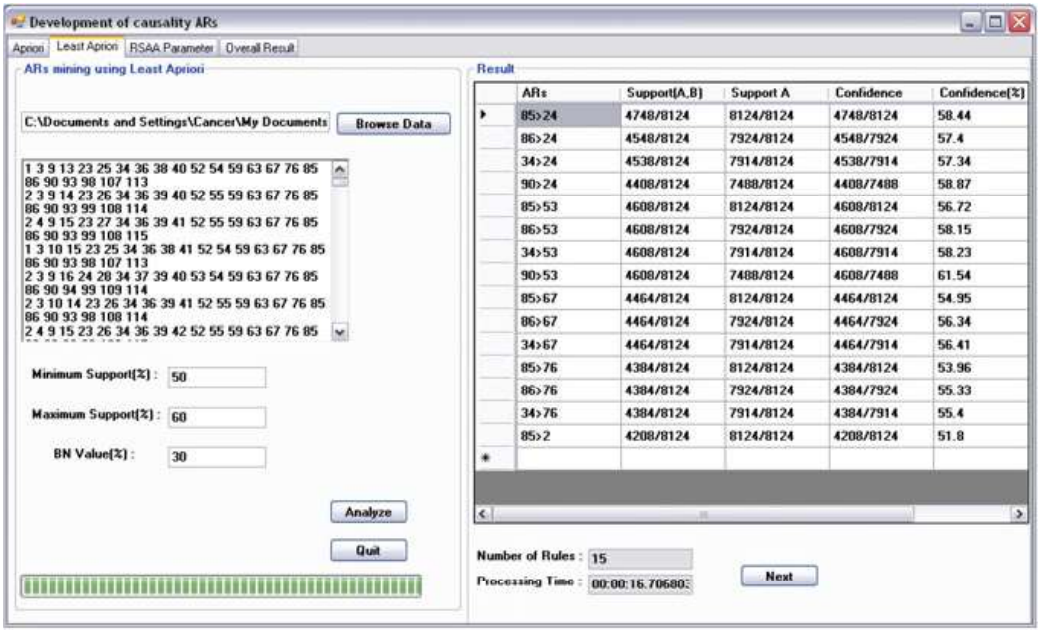

Fig. 9 Causality least association rules generated by the Least Apriori algorithm

number of rules as the standard algorithm (Apriori). Although both algorithms share the same number of generated rules, the Least Apriori algorithm is faster at 3.23% than RSAA.

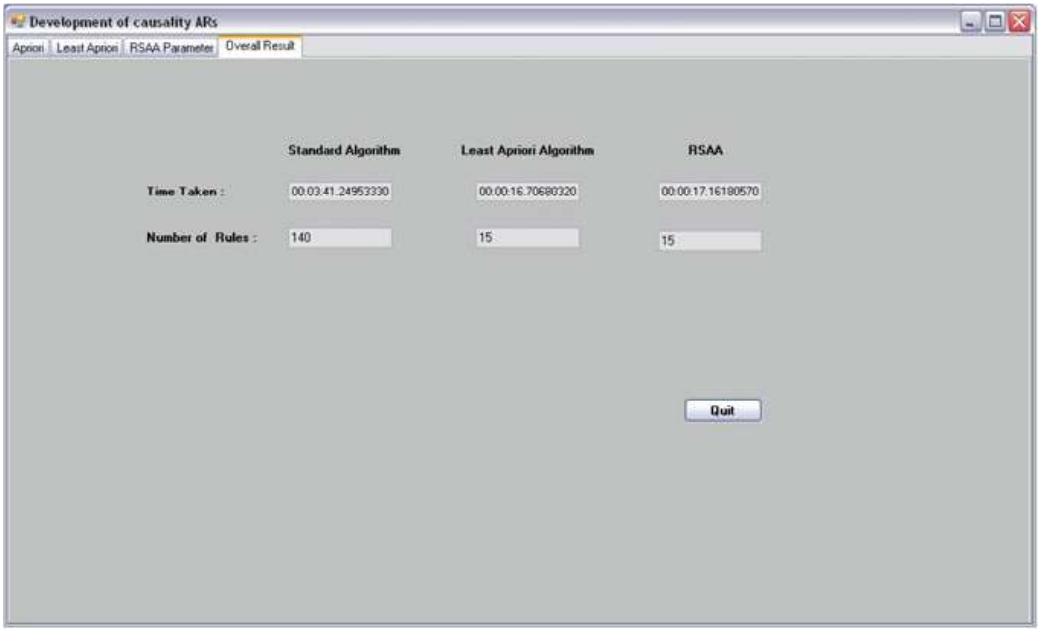

Fig. 10 Overall performance result analysis

6 Conclusion

This study proposed and developed the Causality Least Association Rules Algorithm Tool (CLART) using the Rational Unified Process (RUP) methodology. A comparative analysis has been carried out on these three algorithms' performance. The results show that CLART can be used to extract the desired least association rules, and it also outperformed the existing benchmark algorithms, i.e., the standard algorithm (Apriori) and the RSAA. CLART is an open-source tool to extract the causality least association rules from benchmark datasets such as the UCI Machine Learning Repository, Kaggle, Google Dataset Search, etc.

References

1. Shuja, A.K, & Krebs, J.: IBM Rational Unified Process References and Certification Guide. Solution Designer, Pearson plc (2007). https://www.oreilly.com/library/view/ibm-rational-unified/9780131562929/.
2. Taft, D.K.: IBM Acquires Rational (2002). https://www.eweek.com/pc-hardware/ibm-acquires-rational/.
3. Anwar, A.: A Review of RUP (Rational Unified Process). International Journal of Software Engineering 5(2), 8–24 (2014). https://www.cscjournals.org/manuscript/Journals/IJSE/Volume5/Issue2/IJSE-142.pdf.
4. Kruchten, P.: The Rational Unified Process: An Introduction, Addison-Wesley Longman Publishing Co., Inc. (2003). https://books.google.com.my/books/about/The_Rational_Unified_Process.html?id=RYCMx6o47pMC&redir_esc=y.

5. Malik, M.M., & Haouassi, H.: Efficient sequential covering strategy for classification rules mining using a discrete equilibrium optimization algorithm. Journal of King Saud University-Computer and Information Sciences, 34(9), 7559–7569 (2022). https://doi.org/10.1016/j.jksuci.2021.08.032.
6. Bashir, S., Lai, D.T.C.: Mining Approximate Frequent Itemsets Using Pattern Growth Approach. Information Technology and Control, 50(4), 627–644 (2021). https://doi.org/10.5755/j01.itc.50.4.29060.
7. Elhady, N.E., Jonas, S., Provost, J., & Senner, V.: Sensor failure detection in ambient assisted living using association rule mining Sensors (Switzerland), 20(23), 6760, 1–21 (2020). https://doi.org/10.3390/s20236760.
8. Abdullah, Z., Saman, M.Y.M., Karim, B., Deris, M.M. & Hamdan, A.R. FCA-ARMM: A model for mining association rules from formal concept analysis. Advances in Intelligent Systems and Computing, 549 AISC, 213–223 (2017). https://doi.org/10.1007/978-3-319-51281-5_22.
9. Abdullah, Z., Ngah, A., Herawan, T., Mohamad, S.Z., & Hamdan, A.R.: ELP-M2: An Efficient Model for Mining Least Patterns from Data Repository Advances in Intelligent Systems and Computing, 549 AISC, 224–232 (2017). https://doi.org/10.1007/978-3-319-51281-5_23.
10. Jibril, A.B., Kwarteng, M.A., Appiah-Nimo, C., & Pilik, M.: Association rule mining approach: Evaluating pre-purchase risk intentions in the online second-hand goods market. Oeconomia Copernicana, 10(4), 669–688 (2019). http://economic-research.pl/Journals/index.php/oc/article/view/1737/1628.
11. Abdullah, Z., Adam, O., Herawan, T., Saman, M.Y.M., & Hamdan, A.R.: 2M-SELAR: A model for mining sequential least association rules. Lecture Notes in Electrical Engineering, 2019, 520, 91–99 (2019). https://doi.org/10.1007/978-981-13-1799-6_10.
12. Shao, J. & Tziatzios, A.: Mining range associations for classification and characterization. Data and Knowledge Engineering, 118, 92–106 (2018). https://doi.org/10.1016/j.datak.2018.10.001.
13. Telikani, A., Gandomi, A.H., Shahbahrami, A.: A survey of evolutionary computation for association rule mining. ACM Computing Surveys, 524, 318–352 (2022). https://doi.org/10.1016/j.ins.2020.02.073.
14. De, S., Dey, S., Bhatia, S., Bhattacharyya, S.: An introduction to data mining in social networks, In Hybrid computational intelligence for pattern analysis. Advanced Data Mining Tools and Methods for Social Computing, Academic Press, 1–25 (2022). https://doi.org/10.1016/B978-0-32-385708-6.00008-4.
15. Hu, Y-H., & Chen, Y-L.: Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism, Decis Support Syst., 42(1), 1–24 (2004). https://doi.org/10.1016/j.dss.2004.09.007.
16. Koh, Y.S and Pears, R.: Rare association rule mining via transaction clustering. In Proceedings of the 7th Australasian Data Mining Conference (AusDM '08). Australian Computer Society, Inc., AUS, 87–94 (2008). https://doi.org/10.5555/2449288.2449304.
17. Yang, X-S. 2019. Mathematical foundations, Xin-She Yang, Introduction to Algorithms for Data Mining and Machine Learning, Academic Press, 19–43 (2019). https://books.google.com.my/books/about/Introduction_to_Algorithms_for_Data_Mini.html?id=QtKdDwAAQBAJ&redir_esc=y.
18. Heckerman, D.: A tutorial on learning with Bayesian networks (Tech. Rep. MS-TR-95–06). Redmon, WA: Microsoft Research (1995). https://doi.org/10.48550/arXiv.2002.00269.
19. Ian, W., Mark, H., Eibe, F., Holmes, G., Phafringer, B., & Reutemann, P.: The WEKA data mining software: An update, ACM SIGKDD Explorations, Newsletter, 11(1), 10–18 (2009). https://doi.org/10.1145/1656274.1656278.
20. Ristoski, P., Bizer, C., & Paulheim, H.: Mining the web of linked data with RapidMiner. Journal of Web Semantics, 35 (3), 142–151 (2015). https://doi.org/10.1016/j.websem.2015.06.004.
21. Dulhare, U.N., Ahmad, K., & Ahmad, K.A.: Machine Learning and Big Data: Concepts, Algorithms, Tools and Applications, Scrivener Publishing LLC (2020). https://www.wiley.com/enie/Machine+Learning+and+Big+Data:+Concepts,+Algorithms,+Tools+and+Applications-p-9781119654742.

22. Agrawal, R. & Srikant, R.: Fast algorithms for mining association rules. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, 487–499 (1994). https://doi.org/10.5555/645920.672836.
23. Atanasova, M., Cleeton, L. & Flasko, M., & Paka, A.: Networking, Get Connected With The .NET Framework 3.5. MSDN Magazine (September 2007). https://learn.microsoft.com/en-us/archive/msdn-magazine/2007/september/networking-get-connected-with-the-net-framework-3-5.
24. Koh, Y.S & Rountree, N. Finding sporadic rules using apriori-inverse. In Proceedings of the 9th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining (PAKDD'05). Springer-Verlag, Berlin, Heidelberg, 97–106 (2005). https://doi.org/10.1007/11430919_13.
25. Koh, Y.S., Rountree, N. & O'Keefe, R.A.: Finding Non-Coincidental Sporadic Rules Using Apriori-Inverse, International Journal of Data Warehousing and Mining, 2(2), 8–54 (2006). https://doi.org/10.4018/jdwm.2006040102.
26. Yang, X-S.: Mathematical foundations. Introduction to Algorithms for Data Mining and Machine Learning, Academic Press, 19–43 (2019). https://doi.org/10.1016/B978-0-12-817216-2.00010-7.
27. Jacobson, I., Booch, G. & Rumbaugh, J.: Unified Software Development Process. Addison-Wesley (1999). https://doi.org/10.5555/309683.