Journal of Advanced Computer Science & Technology, 4 (1) (2015) 156-162
www.sciencepubco.com/index.php/JACST

doi: 10.14419/jacst.v4i1.4398
Research Paper

Using the interestingness measure lift to generate association rules

Nada Hussein, Abdallah Alashqur*, Bilal Sowan

Faculty of Information Technology Applied Science University Amman, Jordan
**Corresponding author E-mail: alashqur@asu.edu.jo*

Abstract

In this digital age, organizations have to deal with huge amounts of data, sometimes called Big Data. In recent years, the volume of data has increased substantially. Consequently, finding efficient and automated techniques for discovering useful patterns and relationships in the data becomes very important. In data mining, patterns and relationships can be represented in the form of association rules. Current techniques for discovering association rules rely on measures such as support for finding frequent patterns and confidence for finding association rules. A shortcoming of confidence is that it does not capture the correlation that exists between the left-hand side (LHS) and the right-hand side (RHS) of an association rule. On the other hand, the interestingness measure lift captures such as correlation in the sense that it tells us whether the LHS influences the RHS positively or negatively. Therefore, using Lift instead of confidence as a criteria for discovering association rules can be more effective. It also gives the user more choices in determining the kind of association rules to be discovered. This in turn helps to narrow down the search space and consequently, improves performance. In this paper, we describe a new approach for discovering association rules that is based on Lift and not based on confidence.

Keywords: *Data Mining; Knowledge Discovery in Database (KDD); Association Rule Mining; Interestingness Measures.*

1. Introduction

Data mining is a computational and analytic process used to discover hidden patterns in large data sets and summarize those patterns in a manner useful to the user. Data Mining is widely used in many fields such as marketing, business, and medical applications [1,2,3]. Association rule mining is a popular data mining method used to discover relationships between data items in databases and then represent those relationships in the form of association rules. It uses two measures, namely, *support* and *confidence* that are usually provided by the user, as a criteria for discovering association rules [4,5]. A shortcoming of confidence is that it does not capture the correlation that exists between the left hand sides (LHS) and the right hand side (RHS) of an association rule. In other words, confidence does not capture whether the LHS influences the RHS positively by increasing its likelihood or negatively by decreasing its likelihood. Also it does not capture if the RHS is totally independent from the LHS. Because of this shortcoming, a need arises in many situations to run a post process that identifies the rules of interest based on how the LHS influences the RHS in each rule. The post process normally uses one of different interestingness measures such as Laplace, Conviction, or Lift. These measures are well-known and widely used in association rule mining. They help in filtering out association rules that are not interesting. Lift is one of the simplest of these measures, yet it is very powerful. It can capture and represent the type of correlation that exists between the LHS and RHS of an association rule.

To alleviate the above-mentioned shortcoming of *confidence*, in this paper a new association rule mining algorithm is described. In this algorithm, *lift* is integrated with the algorithm and used instead of confidence as a criteria for discovering association rules, hence this algorithm is called Lift-Based Algorithm (LBA). Using LBA there is no need to apply a post process to identify rules of interest since this capability is integrated in LBA itself. This approach has major advantages over existing approaches. Basically, instead of returning a huge number of association rules, and then filtering out the ones that are not interesting in a post process, the filtration is performed in the main process. This

reduces the number of association rules that the system extracts, therefore, it does not overwhelm the user with a huge number of association rules. In addition, LBA gives the user more options to choose from at the very beginning. Based on the choices collected from the user, LBA can narrow down the search space in order to extract *only* the association rules of interest to the user. Furthermore, a new pruning technique is introduced in LBA, which helps to improve the performance.

2. Background information

In this section, we give some useful background information before presenting the details of our work in the following sections.

Support and confidence are two measures used to find frequent patterns and association rules. Support is an indication of the number of times in which items appear in the database. If an item set {milk, bread} exist in the database, then support (Milk, bread) means the proportion of transactions in the data set, which contain the item set, all transactions which contain milk and bread with others are in data.

Whereas confidence is used to discover an association rule of the form: {Milk, bread} → {butter}. Confidence is the ratio of the number of transactions that include all items in the association rule {milk, bread, butter} to the number of transactions that include all items in the left-hand side of the rule {milk, bread}.

The rule means specific association relationships among a set of objects ("occur together" or "one implies the other") in a database (data warehouse) [6]. Association rules depend on finding frequent patterns of item sets by determining minimum-support and minimum-confidence. When using confidence to find association rules, we will face a particular problem, such as confidence, which does not give the correlation between item sets on each, is it positive or negative? For that, this paper will try to resolve this weakness through Lift to find the relationship between the item sets and specific algorithm applied.

Lift has been used instead of confidence to extract the association rules because the confidence does not give what the effect between LHS (left-hand side hand side) and RHS (right-hand side hand side) in association rule is. Since the algorithm we are developing is based on Lift, we call it Lift-Based Algorithm (LBA). The work presented in this paper is part of an on-going wider scope project named Probabilistic Data Management and Mining (PDMM) that we are conducting. The LBA algorithm will use only support with Lift to extract an association rule. The Lift is simpler than other measures because it depends on support only. LBA algorithm will solve the problem of impact of LHS on RHS in association rules. In other words, it will work to determine the type of correlation between LHS and RHS in association rules because we are dealing with huge and structured size of data. Finally, the LBA algorithm will give data that are less dependent on what user required. Another technique will be introduced, and use calculation of the association rules with the Lift to increase the speed of the algorithm and to generate association rules.

This paper is organized as follows; Section 2 will contain some previous algorithm in related work, section 3; we will introduce our algorithm to generate an association rule, while section 4 we discuss the result for LBA algorithm, finally in section 5 the conclusion for our algorithm.

3. Related work

Data Mining has become the most important areas which help decision-maker in various fields. Several algorithms have appeared to extract association rules.

The Apriori algorithm to extract association rules was proposed in [7] this algorithm based on finding frequent support and generation association rule by confidence. Furthermore, it uses pruning technique to find a frequent and association rules.

Through finding frequent itemsets using candidate generation, Apriori first scans the database and finds frequent itemsets of size 1 by accumulating counts for each item and collecting those which meet the minimum support requirement [8].

The FP-Growth algorithm without candidate generation was proposed in [9]. FP-Growth or frequent pattern tree FP-Tree algorithm depends on trees, each tree consists of root and sub-tree item (child), as each child node is a different item. Each node also stores support information for itemset including items on the path from the root to node. This algorithm works as follows: first, it compresses input database by creating an FP-tree instance to represent frequent items. Then, it divides the compressed database into a set of conditional databases; each one is associated with one frequent pattern. At the end, each such database will be mined separately. Using this strategy, the FP-Growth reduces the search costs looking for short patterns repeatedly and then linking them in long frequent patterns, offering good selectivity [10].

The AIS Algorithm was proposed in [11]. AIS focuses on improving the quality of databases along with necessary functionality to process decision support queries. In this algorithm, only one item that is consequent association rules is generated. This means that the consequence of those rules only contains one item. For example, A U B→ C but not rule A→B U C [12], where U denote set UNION. The algorithm works as follows: Candidate item sets are generated and

counted on-the-fly as the database is scanned. After reading a transaction, it is determined which of the item sets that is large in the previous pass which is included in this transaction [8].
TERTIUS algorithm was introduced in [13]. This algorithm finds the rule according to the confirmation measures. It uses first-order logic representation. It includes different options like class index, classification, confirmation threshold, confirmation values, frequency threshold, horn clauses, missing values, negation, noise threshold, number literals, repeat literals, values Output, etc. [14].

RElim (Recursive Elimination) algorithm [15] represents a (conditional) database through storing one transaction list for each item (partially vertical representation). The SaM algorithm employs only a single transaction list (purely horizontal representation), which is stored as an array [16].
There are some previous algorithms which have been developed over the years, for example, Apriori algorithm has been developed to AprioriTid and AprioriHybrid.
We will introduce the algorithm by using lift as an interestingness measure instead of confidence to extract an association rule. We know that the association rule depends on support and confidence with algorithms that use support and confidence such as Apriori, but LBA algorithm will be used for Lift instead of confidence because confidence lacks an important feature which is that some of the association rule extracted after applied lift on them may be of a negative impact, this is the most important difference between confidence and Lift.
Support and confidence are usually used to extract association rules. It is well known, that even rules with a strong support and confidence may indeed be uninteresting. This is the reason, once the association rules X→ Y has been extracted; it is wise to double-check that how much X and Y are related.
Association technique does not need only adequate thresholds that can be chosen for the two standard parameters of support and confidence. However, also, that appropriate measures of interestingness can be considered to keep the meaningful rules which filter un interesting rules out [17].
Lift plays an important role in data mining, regardless the kind of patterns being mined. It is a proposal for selecting and classifying patterns according to their potential interest to the user.
Lift also allows the time and space costs of the mining process to be reduced [18]. But how this is done using the Lift?

The Lift equation is: $\text{Lift} = \frac{P(X \cup Y)}{P(X)P(Y)}$ (1)

When we calculate the Lift, we will have three possibilities:
1) If Lift is greater than 1, the correlation is positive.
2) When it's less than 1, the correlation is negative.
3) When it equals 1, the correlation is independent.

Sometimes, the added value is used with Lift. The added value of the rules X→ Y is denoted by AV (X → Y) and measures, whether proportion of transactions containing Y among the transactions containing X is greater than proportion of transactions containing Y among all transactions. Then, only if the probability of finding item Y when item X has been found is greater than the probability of finding item Y at all, we can say that X and Y are associated, and X implies Y [19]. A Lift which is greater than 1 indicates a strong correlation between X and Y. A Lift, which is around 1, says that.

$P(X, Y) = P(X)*P(Y)$ (2)

In terms of probability, it means that the occurrence of X and occurrence of Y in the same transaction are independent events; hence X and Y are not correlated. It is easy to show that the Lift is 1, when added value is 0; the Lift is greater than 1 exactly, when it is positive and Lift is below 1 exactly, when it is negative [12]. The Figure 1 explains the Lift. So, Lift solves the problem of the impact of LHS on RHS in association rules; it will work to determine the type of data impact on each other. Thus, it classifies the correlations.

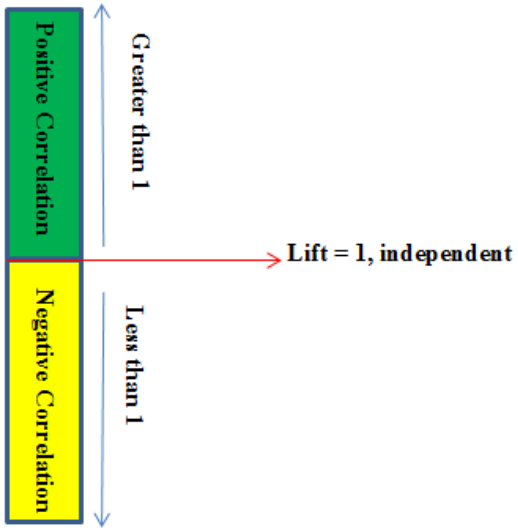

Fig. 1: Lift Probabilities

4. Our method

Usually, when we use the confidence and support to discover the association rules, sometimes a number of patterns is discovered beyond human possibilities of determining the required results. Using the confidence is not much effective to determine the association rules; which means the confidence does not give the type of correlation between LHS and RHS in association rules. The LBA algorithm has been used instead of confidence to extract the association rules because the confidence did not give the effect between X and Y in association rule. The LBA algorithm will use only support with Lift to extract an association rule; the Lift is simpler than other measures because it depends only on support. The LBA algorithm will use two values with the Lift to generate association rules (alpha and Beta)

α (Alpha): This value is used to identify positive correlation (if lift > 1 + α)

β (Beta): This value is used to identify negative correlation (if lift < 1 - β)

The correlation between the LHS and RHS of a rule does not exist (i.e., LHS and RHS are independent of each other) if: (Lift < 1 + α) AND ((lift > 1 - β). Figure 2 shows this case.

α And β values that are determined by the user give algorithm more flexibility with any data (marketing, medicine, education…etc.).

The introduced algorithm system is for extracting all the possibilities and classifications of correlation when using lift.

In addition to provide value for minsup, in a system based on our approach, user can also select one or more of the following options for Lift depending on the application needs.

1) Lift > (1 + α): positively correlation of association rule
2) Lift < (1 – β): negatively correlation of association rule
3) Lift is in the range between (1 – β) and (1 + α): independent correlation of association rule. Figure 2 shows that.

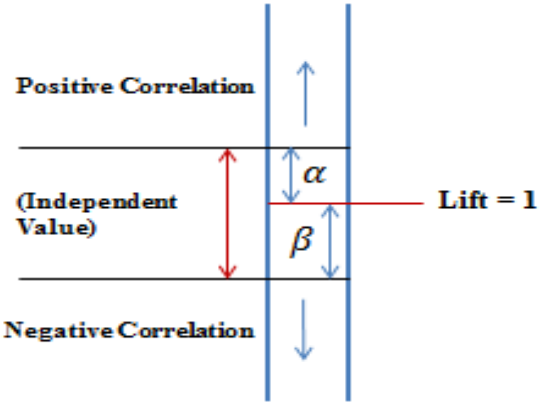

Fig. 2: Meaning of Alpha and Beta Used In LBA.

In this paper, the algorithm used the data provided by the Applied Science University (ASU). Data contains information such as High School GPA, High school majors (Science, Literature, and Information Technology, etc.), current college GPA and college majors among other data. These data used as a test data to apply the ideas and concepts introduced in this paper. Table 1 illustrates a sample of data obtained from the university's Computer Center.
Data have been extracted in this proposal from six tables:

1) Student-info: It contains some information about student (name, gender, phone no.... etc.), it contains 62 columns.
2) Term: Description of each student's term (term No, Hours No, subjects, etc.), it contains 18 columns.
3) Collage: Description of each collage (Collage name, Collage No ... etc.), it contains five columns.
4) Major: Description of each major collage (Major-id, Major name ... etc.), it contains eight columns.
5) Branch-School: Description of school branch for each student (Branch-id, Branch description, School average), it contains 4 columns.
6) Degree: Description of degree for each subject (First-mark, Second-mark, Final-mark ... etc.), it contains 9 columns.

The ASU data were on 7334 students, but we extracted 2171 rows (students).

Table 1: Students Data

Students	College	Major	GPA	School branch	School Avg.
1	Engineer	Industrial	56 fail	Scientific	93.3
2	Pharmaceutical	Pharmaceutical	63 passable	Scientific	86
3	Literature	Political	59 fail	Scientific	62.2
4	Law	Law	75 good	Scientific	69.9
5	Engineer	Civil	64 passable	Industrial	79.76
6	Arts	Graphics	90 excellent	Literary	57.8
7	Engineer	Computer	59 fail	Scientific	92.72
8	Arts	Interior Design	78 very good	Literary	53.7
9	Engineer	Civil	56 fail	Scientific	88.66
10	Pharmaceutical	Pharmaceutical	61 passable	Scientific	62.4
11	Law	Law	56 fail	Scientific	71.03
12	Economy	Finance and banking	60 passable	Scientific	56.4
13	Economy	Finance and banking	58 fail	Literary	56
.					
.					
2171	IT	CS	74 good	Scientific	64.6

Our algorithm works on a user basis of providing value of Lift. The algorithm returns required results. Lift which may be equal to 1 or less or greater than 1. The algorithm can be used with any structure data. In our data, we assume: minsup = 20% and minconfidence = 30% to calculate the following association rules:

B AVG, Scientific → Fail where B AVG = School average between (61- 71.99)

$$\text{Confidence} = \frac{P(A \cup B)}{P(A)} = 0.305 = 30\%$$

However, already support of fail > 30%
Let's use Lift:

$$\text{Lift} = \frac{P(A \cup B)}{P(A)P(B)} = 0.94 < 1$$ so The LHS negatively effects the RHS.

It can be noted that the result above compares the effect of the right lift on the left lift. An algorithm is based on determining two factors; these factors are asymmetric values, which means they are not identical in terms of the increase and decrease in the rate of the value of each of them in the same condition.
An algorithm is based on determining two factors; these factors are asymmetric values, which means they are not identical in terms of the increase and decrease in the rate of the value of each of them in the same condition. This algorithm works as follows:

1) Insert MinSupport.

2) Choose Correlation of Association rule:
   a) If choice Positive correlation of Association rules, Insert α. (In this choice lift value > 1+ α)
   b) If choice Negative correlation of Association rules, Insert β. (In this choice lift value < 1- β)
   c) If choice Independent correlation of Association rules, Insert α and β. (In this choice lift value is between (1- β and 1+ α).
   d) Else insert α and β.
3) Press accepts to execute operation (to scan database).
4) Get a frequent item for 1,2 and three item by Sequentially

Check for per value in item:

a) if >= MinSupport added to frequent items
b) Else Ignore.
   1) Generate candidate association rules from frequent items.
   2) Calculate the lift value for each candidate association rules to classify as like:

$$\text{Lift} = \frac{\text{Support (LHS} \cup \text{RHS)}}{(\text{Support LHS}) * (\text{Support RHS})}$$

Where A, B, C are the items to generate an association rule.

5) Generate Association Rule for the choosing Correlation of Association rule.
6) If not found result (not found association rules), go to step 1 to edit MinSupport or edit the type of correlation).

Furthermore, we rely on the principle of pruning that depends on association rule aggregates. This principle means any two opposite association rules. It has the same lift value: Lift (A→B) = lift (B→A). If we have the association rule: Lift (A→B), by taking the opposite association rule: lift (B→A) we reduced the time needed to calculate the association rule to half and there are no need to calculate the lift value for each rule.

5. Result

LBA algorithm solved the problem of the confidence which has been mentioned above. It helps the user to make important decisions during extracting the association rules and their impact on each other and in all aspects of life. Furthermore, it explained the correlation between LHS and RHS in association rule: positive, negative or independent correlation with each other's. LBA algorithms directly extract association rule and type of correlation without using confidence. After implemented LBA algorithm, we found that the performance to generate an association rule has been improved and it generated only what the user wants, based on the type of correlation selected by the user.

6. Conclusion

Many techniques in association rule mining used confidence and support, but those techniques are not effective completely. Confidence discovers association rules without taking into consideration, whether the left-hand side of the rule influences the right-hand side positively or negatively. Lift is playing an important role in data mining, whatever the kind of patterns being mined is. Furthermore, it allows the selection and classification of patterns according to their potential interest to the user by determining the required criteria for organization or user. The LBA algorithm uses lift to extract association rules.

Acknowledgement

The authors are grateful to the Applied Science Private University, Amman, Jordan, for the financial support granted to cover the publication fee of this research article.

References

[1] R. Elmasri, S. B. Navathe, Fundamentals of database systems, sixth edition, Adeson-wesley publish, New York, 2011.
[2] D. Hand, H. Mannila and P. Smyth, Principles of Data Mining, 2001, publish USA.
[3] D. T. Larose, Discovering knowledge in data an introduction to data mining, first edition, Adeson-wesley publishes, USA, 2005.
[4] Ayobami S. A., Rabi'u S., Knowledge Discovery in Database: A Knowledge Management Strategic Approach, Knowledge Management International Conference (KMICe) 2012, Johor Bahru, Malaysia, 4 – 6 July 2012.
[5] Lou, Q., Advancing Knowledge Discovery and Data Mining. School of Electrical and Information engineering, WITN, China. IEEE Computer Society. 0-7695-3090-7/08, 2008.
[6] M. D. Khatri, S. Dhande, History and Current and Future trends of Data mining Techniques, IJARCSMS International Journal of Advance Research in Computer Science and Management Studies, Vol.2, Issue 3, March 2014.
[7] M. Ingle, N. Suravanshi, "Review: Apriori Algorithms and Association Rule Generation and Mining", AIJRSTEM American International Journal of Research in Science, Technology, Engineering & Mathematics, Published by IASIR International Association of Scientific Innovation and Research, USA, December 2013-Febryary 2014, pp. 180-183.

[8] R. Agrawal, R. Srikant, Fast Algorithms for Mining Association Rules. In Proc. 1994 Int. Conf. Very Large Data Bases (VLDB'94), Santiago, Chile, pp. 487–499.
[9] J. Han, J. Pei, and Y. Yin. Mining Frequent Patterns without candidate Generation. In Proc. 2000 ACM-SIGMOD Int. Conf. Managenment of Data (SIGMOD'00), PAGE 1 12, Dallas, TX, May 2000. http://dx.doi.org/10.1145/342009.335372.
[10] C. Gyorödi, R. Gyorödi, and S. Holban "A Comparative Study of Association Rules Mining Algorithms", In: Proceeding SACI 2004, 1st Romanian-Hungarian Joint Symposium on Applied Computational Intelligence , Timisoara, Romania, May 25-26, 2004, pp. 213-222.
[11] R. Agrawal, T. Imielinski, and A. Swami, Database mining: A performance perspective. IEEE Transactions on Knowledge and Data Engineering, 5(6): 914 925, December 1993.
[12] C. S.Deora, S. Arora, and Z. Makani, Comparison of Interestingness Measures: Support-Confidence Framework versus Lift-Irule Framework, IJERA International Journal of Engineering Research and Applications, Vol. 3, Issue 2, March-April 2013.
[13] Peter a. Flach and et al , Confirmation-Guided Discovery of First-Order Rules with Tertius, Machine Learning, 42, 61–95, 2001 °c 2001 Kluwer Academic Publishers. Manufactured in The Netherlands.
[14] J. Arora, N. Bhalla, S. Rao, A Review on Association Rule Mining Algorithms, IJIRCCE International Journal of Innovative Research in Computer and Communication Engineering, Vol. 1, Issue 5, July 2013.
[15] B. D. Dubey, M. Sharman, and R. Shah, "Comparative Study of Frequent Item Set in Data Mining", IJPLA International Journal of Programming Languages and Applications, Indora, India,Vol. 5, No. 1, January 2015.
[16] S. Pramod ., O.P. Vyas, Survey on Frequent Item set Mining Algorithms, IJCA International Journal of Computer Applications, Vol. 1, No. 15, 2010.
[17] U. K. Pandey, S. Pal, A Data Mining view on Class Room Teaching Language, IJCSI International Journal of Computer Science, Vol.8, Issue 2, March 2011.
[18] L. Geng, H. J. Hamilton, Interestingness Measures for Data Mining: A Survey, ACM Computing Surveys (CSUR), Vol.38, Issue 3, No. 9, 2006.