GAJA: A New Consistent, Concise and Precise Data Mining Algorithm

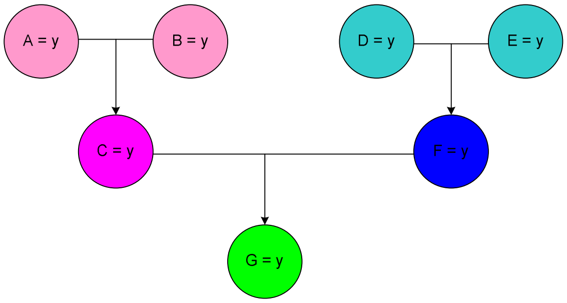
Suwimon Kooptiwoot

School of Information Technologies, University of Sydney
University of Sydney, NSW, 2006, Australia
suwimon@it.usyd.edu.au

Abstract. In this paper we present a new data mining algorithm which can give us the relationship among attributes within data set concisely and precisely. The relationships found from this algorithm shown in the rule form are consistent by not depending on the total number of record within the data set or the different proportion of all of the possible cases within the data set. This algorithm is neither classification algorithm nor association rules algorithm. The number of rules generated from this algorithm can be comparable with classification algorithm and the precise of the rules generated can be comparable with the association rules algorithms. We also show the results from testing this algorithm against two classification algorithms and two association rules algorithms with the synthetic data set with known relationship among attributes within the data set.

1 Problem

First we want to find a data mining algorithm which can be used to tell us the relationships among attributes within the data set in our hand. So we explore the data mining algorithms and see how they work. Then we test 4 data mining programs, two classification algorithms, KnowledgeSeekerIV trial version [1] and See5 demo version [2], and two association rules algorithms, Apriori algorithm [3, 4], and the derived version of Apriori algorithm which generate only one item in the consequence part of the rules. We test all of these algorithms with synthetic data sets with known relationship as shown in Figure1



Fgiure1: Relationships among A, B, C, D, E, F and G

The value of every attribute can be only *y* or *n*. We synthesize a data set from these relationships. The data set consists of all of the possible cases. KnowlegdeSeekerIV trial version gives us total 19 rules; only 6 rules get 100 % accuracy are

RULE 1: IF C = y THEN A = y
RULE 2: IF C = y THEN B = y
RULE 3: IF A = n THEN C = n
RULE 4: IF F = y THEN D = y
RULE 5: IF F = y THEN E = y
RULE 6: IF D = n THEN F = n

The question comes up to our mind that is, where is the relationship about G attribute? We see that the relationship shown cannot give us the whole picture of all of the relationships among these attributes.

Then we try with See5 demo version. We have to set class attribute for See5, so we try setting every attribute as class attribute at a time. See5 gives us the results as shown follow.

Setting G as class attribute: no rule generated, there is only default class is n
Setting A as class attribute
Rule 1: C = y -> class y
Rule 2: C = n -> class n
Setting B as class attribute
Rule 1: C = y -> class y
Rule 2: C = n -> class n
Setting C as class attribute: no rule generated, there is only default class is n
Setting D as class attribute
Rule 1: F = y -> class y

Rule 2: F = n -> class n
Setting E as class attribute
Rule 1: F = y -> class y
Rule 2: F = n -> class n
Setting F as class attribute: no rule generated, there is only default class is n

The same question comes up to our mind, that is, where is the relationship of G attribute? The result from See5 cannot give us the overall picture of the relationships among all of these attributes as well. We know that these classification algorithms try to deal with noise data and don't want to get the over fit problem or under fit problem [5, 6] that make them give the rules like this. And the rules generated from classification algorithms depend on the total records in the data set and also the different of the proportion of all of the possible cases in data set. So they cannot be used to find all real relationships among attributes within complete simple data set which consists of all possible cases.

Then we try with the association rules. As we know that the association rules algorithms basically generate all of the combination of the relationships among all attributes within the data set according to the confidence threshold and minimum support threshold setting. So everyone in the data mining community knows well that the number of rules generated from association rules algorithms is very huge. We test with Apriori algorithm and a derived version of Apriori algorithm which generate only rules with one attribute in the consequence part. We set the confidence threshold as 100 % and minimum support threshold 1 %. From this synthetic data set, Apriori algorithm gives us 2,311 rules, and the derived version of Apriori algorithm gives us 1,170 rules. Sure that the rules generate from the association rules algorithms will cover all of the real relationships if we set the minimum support threshold is very least to make it consider even one record of possible case. But the problem is how we can know from the sea of the rules generated that which rules the real relationships are. We found that there are so many works, for example in [7-13]come up to deal with this problem to find the interestingness rules or something else. They tried to use many criteria to prune some rules out. The problem is when they prune some rules out that mean it is possible that some real relationships can be pruned out as well. So we decide to develop our own algorithm which can be used to mine the relationships among all attributes within the simple complete data set. Our new algorithm is presented in the next section.

2 New Algorithm: GAJA (Generating All Rules by Joining All Attributes)

From the problem mentioned above, with even simple complete data set, we cannot find the classification algorithms or association rules algorithms which can be used to give us all real relationships among attributes within data set by not giving us the sea of the rules.

We then develop a new algorithm which is neither classification algorithm nor association rules algorithm to deal with this problem.

2.1 Aim of Developing Algorithm

We set three criteria of developing our algorithm as follow.

1. Give us all of the real relationships by not just only for class attribute.
2. The number of generated rules is not much like association rules algorithm
3. Easy to see the relationship

Our algorithm has to have the capacity that meets all of these criteria.

2.2 GAJA Algorithm (Generating All Rules by Joining All Attributes)

```
For all attributes
        For all values of current attribute
        Group data cases which have the same value of current attribute
        Select the attributes whose values are the same as all cases in current group
        Put selected attributes and their values in THEN table
        Put the name and value of current attribute used for grouping in IF table
End

Group rules which have the same THEN part together
End

Generate rules from IF and THEN table
```

3 Experiments

We test GAJA algorithm with the same synthetic data set as we test with the other existing algorithms mentioned above.

3.1 Rules from GAJA

RULE 1: IF A = n   THEN C = n G = n
RULE 2: IF B = n   THEN C = n G = n
RULE 3: IF C = y   THEN A = y B = y
RULE 4: IF C = n   THEN G = n

RULE 5: IF D = n THEN F = n G = n
RULE 6: IF E = n THEN F = n G = n
RULE 7: IF F = y THEN D = y E = y
RULE 8: IF F = n THEN G = n
RULE 9: IF G = y THEN A = y B = y C = y D = y E = y F = y

From the result, we see that GAJA algorithm can give us all of the real relationships among the attributes, not just only for class attribute. So our first aim we get from GAJA. And we can see the overall relationships among all attributes from RUEL3, RULE7 and RULE 9. From RULE 3 we can see that if we see the value of C is y, we can know that the value of A has to be y and the value of B has to be y as well. From RULE 7, we can see that if we get the value of F is y then we know that the value of D is y and the value of E is y. From RULE 9 make us know that if we see the value of G is y, we will see the values of all of the rest of attributes are y. From this result, our third aim meets. Then we compare the number of rules generated from GAJA with the other algorithms. The number of all of the rules generated from GAJA and the other algorithms is shown in Table 1

Table 1. The number of generated rules from each algorithm

Algorithm/software	Number of rules generated
Apriori	2,311
Derived Apriori	1,170
KnowledgeSeekerIv	19
See5	8
GAJA	9

We see that the number of the rules generated from GAJA is just 9. It is not as huge as the association rules algorithms. So we get the second aim. Moreover, we see that from GAJA, we can get all of the real relationships precisely over the See5 demo version and KnowledgeSeekerIV trial version. And the number of rules generated is not much. And now we get the new algorithm, GAJA, which meet our three criteria. We also test GAJA with the bigger data sets and with the difference of the proportion of all of the possible cases. GAJA still gives us the same number of rules and the same rules. So GAJA give us the relationships precisely and concisely by not depending on the total number of rules or the difference of the proportion of all of the possible cases.

4 Conclusion

In this paper, we present a new data mining algorithm, GAJA, which is not classification algorithm or association rules algorithm. From our experiments, GAJA can give us the number of generated rules comparable with classification algorithms, and the precise of the rules generated comparable with association rules algorithms. GAJA algorithm gives us all of the relationships, not only just for class attribute. GAJA make us easy to see all of the relationships among attributes. And the number of rules generated from GAJA is not much like association rules algorithm.

References

1. Groth, R., *Data Mining : Building Competitive Advantage*. 2000, New Jersey, USA: Prentice-Hall.
2. Quinlan, J.R., *Data Mining Tools See5 and C5.0.* 2001, RuleQuest.
3. Agrawal, r., T. Imielinski, and A. Swami. *Mining Association Rules between sets of items in large databases*. in *In Proc. of the ACM SIGMOD Conference on Management of Data*. 1993. Washington, D.C.
4. Agrawal, R. and R. Srikant. *Fast Algorithms for Mining Association Rules*. in *In Prpceddings of the 20th International Conference Very Large Data Bases*. 1994.
5. Kantardzic, M., *DATA MINING : Concepts, Models, Mthods, and Algorithms*, ed. E.i.C. Stamatios V Kartalopoulos. 2003, USA: IEEE press.
6. Berry, M.J.A. and G.S. Linoff, *Data Mining Techniques and Algorithms*, in *Mastering Data Mining*, R.M. Elliott, Editor. 2000, John Wiley & Sons, Inc.: USA.
7. Freitas, A.A. *On Obejective Measures of Rule Surprisingness*. in *In Porceedings of the Second European Conference on the Principles of Data Mining and Knowledge Discovery (PKDD'98)*. 1998. Nantes, France.
8. Silberschatz, A. and A. Tuzhilin. *On Subjective Measures of Interestingness in Knowledge Discovey*. in *in Proceedings of the First International Conference on Knowledge Discovery and Data Mining*. 1995. Montreal, Canada.
9. Padmanabhan, B. and A. Tuzhilin. *Small is Beautiful: Discovering the Minimal Set of Unexpected Patterns*. in *Proceedinmgs of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD00)*. 2000.
10. Liu, B., et al., *Analyzing Subjective Interestingness of Association Rules.* IEEE Intelligent Systems, 2000. **15**(5): p. 47-55.
11. Tseng, S.-M., *Mining Association Rules with Interestingness Constraints in Large Databases.* International Journal of Fuzzy Systems, 2001. **3**(2): p. 415-421.
12. Silberschatz, A. and A. Tuzhilin, *What makes Patterns Interesting in Knowledge Discovery Systems.* IEEE Transactions on Knowledge and Data Engineering (TKDE), 1996. **8**(6): p. 970-974.

13. Dong, G. and J. Li. *Interestingness of Discovered Association Rules in terms of Neighborhood-Based Unexpectedness*. in *Proceedings of Pacific Asia Conference on Knowledge Discovery in Databases (PAKDD)*. 1998. Melbourne.