GENERATION OF PROCESS MODELS FOR DYEING PROCESS USING ASSOCIATION RULE MINING

Harendra Singh*

Sitendra Tamrakar*

Abstract:

This paper uses prior domain knowledge to guide the mining of association rules in the dyeing business process atmosphere. This approach is employed so as to beat the drawbacks of data mining using rule induction such as loss of info, discover too several obvious patterns, obvious patterns, and mining of overcome association rules.. association rules. an information mining interactive rule induction association rules. an information mining interactive rule induction algorithm is introduced to mine rules at micro levels. micro levels. The strip-mined rules describe the impact of various shades of the colors, conceiver of the treatment, treatment details Existing to improve the dyeing process quality and production growth. .

Key words: Intrusion detection system, neural network, data mining, false alarm

* Asst Prof. Deptt. Of computer science & Engg., NRI Institute of Science and Technology, Bhopal

1. Introduction

Frequent Pattern Mining is most powerful problem in association mining. Most of the algorithms are based on algorithm is a classical algorithm of association rule mining [2,3, 4]. Lots of algorithms for mining association rules and their mutations are proposed on basis of Apriori Algorithm [2, 3, 4, 5, 6]. Most of the previous studies adopt Apriori-like algorithms, that generate-and-test candidates and up algorithm strategy and structure. Several modifications on apriori algorithm are focused on algorithm Strategy but no onealgorithm emphasis on representation of database. A simple approach is if we implement in Transposed database then result is very quick. Recently, totally different works projected a replacement way to mine patterns in transposed databases where a database with thousands of attributes but only tens of objects [2]. In several example attribute square measure terribly massive than objects or transaction In this case, mining the backward info runs through a smaller search area. In apriori formula every section is count the support of prune pattern candidate from database. No one algorithm filters or reduces the database in each pass of apriori algorithm to count the support of prune pattern candidate from database. We propose a new dynamic algorithm for frequent pattern mining in which database represented in transposed form. And for counting the support we find out by longest common subsequence approach and after finding pattern longest common subsequence is stored or update in database so that next time instead of whole transaction we search from these filter transaction string.

In this Synopsis the frequent pattern mining algorithms are applied in the dyeing process due to difficulties of doing the coloring process in an efficient way. Apriori algorithm can significantly reduce mining time by generating pattern candidates that had successfully brought many researchers' attention [1]. Then, Frequent Pattern (FP) Growth algorithm [2] for dense dataset and HMine algorithm [3] for sparse dataset that scans dataset only twice by utilizing memory data structures had changed the main stream of frequent pattern mining algorithm. The parallel version [4] of FPGrowth algorithm, work efficiently for dense dataset. However, Hyper structure Mining (HMine) algorithm that is known to work efficiently on sparse dataset has no parallel version up to now due to its inherent problem, i.e. HStruct link adjustment during the middle of mining process. In this research we have developed a modified version of HMine algorithm called LinkRuleMiner (LRM) algorithm which can be easily extended to parallel version.

This article analyzes the coloring process of dyeing unit using newly proposed association rule mining algorithm based on apriori algorithm using frequent patterns. These frequent patterns have a confidence for different treatments of the dyeing method. These confidences facilitate the colouring unit knowledgeable called dyer to predict better combination or association of treatments. This text conjointly proposes to change the APriori algorithm to the dyeing process of colouring unit, which can have a serious impact on the coloring process of dyeing industry to process their colors effectively without any dyeing issues, like pales, dark spots on the coloured yarn.

2. Related Work

Process mining provides a new means to improve processes in a variety of application domains. However, process mining is much more than an amalgamation of existing approaches. as an example, existing data processing techniques are too data centric to provide a comprehensive understanding of the end to- end processes in an organization. These method mining techniques facilitate organizations to uncover their actual business processes. method mining isn't restricted to method process discovery. By tightly coupling event information and method models, it is possible to process models, it's potential to envision conformity, check conformity, notice deviations, predict delays, support process redesigns. method mining breathes life into otherwise static process models and puts today's massive data volumes in a process context. The frequent pattern mining plays an essential role in many data mining tasks and applications, like mining association rules, correlations [5], sequential patterns [6], episodes [7], multidimensional patterns [8], max patterns and frequent closed patterns [9], partial periodicity [10], emerging patterns [11], classification [12] and clustering [13]. The numerous studies on the fast mining of frequent patterns can be classified into two categories.

The first category, candidate generation and test approaches, such as Apriori and many subsequent studies, are directly based on an anti-monotone Apriori property [1] if a pattern with k items is not frequent, any of its super-patterns with (k + 1) or more items can never be frequent. A candidate generation and take a look at approach iteratively generates a set of candidate patterns of length (k + 1) from a set of frequent patterns of length k (k ≥ 1), and checks their corresponding occurrence frequencies in the database. The Apriori algorithm achieves good reductions in the size of candidate sets. However, once there exist an outsized range of frequent patterns and/or long patterns, candidate generation and take a look at strategies might still suffer from generating a large number of candidates and taking many scans of large databases.

The second category of methods, pattern-growth methods, such as FPGrowth [14] and Tree Projection have been planned. A pattern-growth technique uses the Apriori property. However, rather than generating candidate sets, it recursively partitions the info into sub-databases according to the frequent patterns found and searches for local frequent patterns to assemble longer global ones. However, these algorithms may still encounter some difficulties in different cases. First, a huge memory space is required for mining, so the main memory consumption is usually hard to predict correctly. An Apriori-like algorithm generates a huge number of candidates for long patterns. To find a frequent pattern of size 100, such as {a1, . . . , a100}, up to 5 × 1030 units of main memory space is needed to store candidates.

In 2001 J. Pei [3] proposed an algorithm called HMine to mine frequent patterns efficiently on a sparse dataset. This algorithm utilizes H-Struct data structure, which has very limited and predictable space overhead, and runs very fast in memory setting, hence modified this algorithm with link structure and reverse order processing. Therefore, the proposed algorithm can be implemented in dyeing process to create frequent patterns. These generated frequent patterns were also compared with existing algorithms to analyze the performance related issues.

3. Problem Identification

Association rule mining is a popular and well researched area for discovering interesting relations between variables in large databases. We have to analyze the coloring process of dyeing unit using association rule mining algorithms using frequent patterns. These frequent patterns have a confidence for different treatments of the dyeing method. These confidences facilitate the coloring unit skilled called dyer to predict better combination or association of treatments.

Various algorithms are used for the coloring process of dyeing unit using association rules. For example. LRM,FP Growth Method., H-Mine and Aprori algorithm But these algorithm significantly reduces the size of candidate sets . However, it can suffer from three-nontrivial costs:

(1) Generating a huge number of candidate sets, and

(2) Repeatedly scanning the database and checking the candidates by pattern matching.

(3) It take more time for generate frequent item set.

(4) The large databases can not be executed efficiently in H-Mine and LRM algorithms, FP Growth

We have to proposed such that algorithm that it has a very limited and precisely predictable main memory cost and runs very quickly in memory-based settings. it can be scaled up to very large databases using database partitioning. and to identify the better dyeing process of dyeing unit.

In this thesis, previous work is based on apriori algorithm in dyeing process of dyeing unit. So basic apriori algorithm are following

3.1 Apriori algorithm:

Apriori is a classic algorithm for learning association rules. association rules. Apriori is intended to work on databases containing transactions (for example, collections of things bought by customers, or details of an online web site frequentation). different algorithms square measure designed for locating association rules in data having no transactions, or having no time stamps. The purpose of the Apriori Algorithm is to find associations between different sets of data. It is sometimes referred to as "Market Basket Analysis". each set of data options a spread of things and is called a transaction

The Apriori algorithm is an unsupervised learning technique that figures out the likelihood of A taking place if B will. it's terribly helpful in prediction relating to supermarkets and businesses.It requires numeric values of each tuple to be presented for the formationof relationships.Minimum support and minimum confidence are found and then support for each attribute is compared with it.

This algorithmic rule is especially divided into 2 processes:

1. Scanning of the database to calculate each attribute's support count and then rejecting the attributes with lesser values.
2. Pruning of candidate sets to remove the candidates without frequent subset

Algorithm:-

The pseudo code for the algorithm is given below for a transaction info ,T and a support threshold of €. Usual set theoretic tation is used, the' note that's T a multiset. is that the C_k is the candidate set for level k. Generate() algorithm is assumed to generate the candidate sets from the large item sets of the preceding level, heeding the downward closure lemma accesses $count[c]$ a field of the information structure that represents candidate set C , that is at first assumed to be zero. Several details square measure omitted below, sometimes the fore most vital a part of the implementation is the data structure used for storing the candidate sets, and investigating their frequencies

Apriori(T, ϵ)

$L_1 \leftarrow \{$ large 1-itemsets $\}$

$k \leftarrow 2$

While $L_{k-1} \neq \emptyset$

$C_k \leftarrow \{c \mid c = a \cup \{b\} \wedge a \in L_{k-1} \wedge b \in \bigcup L_{k-1} \wedge b \notin a\}$

for transactions $t \in T$

$C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq t\}$

For candidates $c \in C_t$

$count[c] \leftarrow count[c] + 1$

$L_k \leftarrow \{c \mid c \in C_k \wedge count[c] \geq \epsilon\}$

$k \leftarrow k + 1$

$\bigcup L_k$

return k

3.2 Disadvantage of Apriori algorithm

Apriori algorithm, in spite of being simple and clear, has some limitation. It is costly to handle a huge number of candidate sets. For example, if there are104 frequent 1-item sets, the Apriori algorithm will need to generate more than107 length-2 candidates and accumulate and test their occurrence frequencies. Moreover, to find a frequent pattern of size one hundred, like , it should generate2100 -2 ˜ 1010 candidates in total. this is often the inherent value of candidate generation, despite what implementation Technique is applied. it's tedious to repeatedly scan the information and check a large set of candidates by pattern matching, that is particularly true for mining long patterns. Apriori Algorithm Scans the database too repeatedly, once the information storing an outsized range of data services, the restricted memory capability, the system I/O load, goodish time scanning the info will be a very long time, so efficiency is very low. In order to overcome the drawback inherited in Apriori.

4. Proposed Algorithm

(Modified Apriori Algorithm)

The Apriori algorithm had a major problem of multiple scans through the entire data. It required a lot of space and time. The modification in our paper suggests that we do not scan the whole database to count the support for every attribute. This is possible by keeping the count of minimum support and then comparing it with the support of every attribute. The support of an attribute is counted only till the time it reaches the minimum support worth. on the far side that the support for associate degree attribute need not be known. This provision is possible by using a variable named flag in the algorithm the value for support is noted. The pseudo code for the proposed algorithm is as follows:

Input : Database, D, of transactions;
Minimum support threshold,
min_ sup

Output : L, frequent item sets in D

Method :

1) L(1)= find_frequent _1-itemsets(D);
2) For each transaction t belongs to D
3) count_items= count_items(t);
4) For (k=2; L(k-1)!=null; k++)
5) {
6) C(k)= apriori_gen(L(k-1, min_sup);
7) flag=1;
8) For each transaction t belonging to D
Where count_items>=k
9) {
10) If (flag==1)
11) {
12) c=subset(C(k),t);
13) c.count++;
14) if (c.count==min_sup)
15) flag=0;
16) }

17) if (flag==0)

18) Exit from loop

19) }

20) L(k)={c.count=min_sup}

21) }

22) return L=U(k) L(k);

5. Conclusion

In this paper, we experiment the dyeing process of the Emerald and Jayabala dyeing unit's using the association rule mining algorithms and Weka Library.

Hence, it is analyzed the association rules and understood the strengths of these rules indicated by confidence and predictive accuracy for the association mining algorithms such as Apriori, FPGrowth, H-Mine and LRM were analyzed in the view of performance. Therefore the outcome of these algorithms generates process models. The generated process models were analyzed and the importance of f these algorithms were discussed. In spite of the limitations of the association rule algorithms, the clustering approach can be conducted through Weka Library. The various dyeing processes were analyzed to generate the process model even it has the complex and less-structured processes. Also these algorithms were used to determine homogeneous shades, color mix processes with various treatments. Therefore, this paper contributes more on by implementing the dyeing process using association rule mining algorithms with Weka Library.

REFRENCES: -

1. A new approach for sensitive association rules hiding by Mohammad Naderi Dehkordi, Kambiz Badie, Ahmad Khadem Zadeh in International Journal of Rapid Manufacturing 2009 - Vol. 1, No.2 pp. 128 - 149
2. Privacy Preserving Fuzzy Association Rules Hiding in Quantative Data by Manoj Gupta and R C Joshi in International Journal of Computer Theory and Engineering, Vol. 1, No. 4, October, 2009
3. Agarwal CC. and Yu PS., “Privacy-preserving data mining: Modeland Algorithms, (editors)CharuC.Aggarwal and Philip S. Yu, ISBN: 0-387- 70991-8, 2008
4. Vassilios S. Verykios,, Ahmed K. Elmagarmid , Elina Bertino, Yucel Saygin, Elena Dasseni. “Association Rule Hiding”, IEEE Transactions on knowledge and data engineering, Vol.6, NO.4, April 2004
5. Shyue-Liang Wang, Yu-Huei Lee, Billis S., Jafari, A. “Hiding sensitive items in privacy preserving association rule mining”, IEEE International Conference on Systems, Man and Cybernetics, Volume 4, 10-13 Oct. 2004 Page(s): 3239 – 3244
6. Brin, S., Motwani, R. and Silverstein, C. “Beyond market basket: Generalizing association rules to correlations”, in the Proceedings of the 1997 ACM-SIGMOD International Conference on the Management of Data (SIGMOD’97), Tucson, AZ, 1997, pp. 265–276.
7. Agrawal, R. and Srikant, R. “Mining sequential patterns”, in the Proceedings of the 1995 International Conference on the Data Engineering (ICDE’95), 1995. pp. 3–14.
8. Mannila, H., Toivonen, H. and Verkamo, A.I. Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery, 1997, pp. 259–289.
9. Kamber, M., Han, J. and Chiang, J.Y. “Meta rule-guided mining of multidimensional association rules using data cubes”, in the Proceedings of the 1997 International Conference on Knowledge Discovery and Data Mining (KDD’97), Newport Beach, CA, 1997, pp. 207–210.
10. Bayardo, R.J. “Efficiently mining long patterns from databases”, in the Proceedings of the 1998 ACM- SIGMOD International Conference on the Management of Data (SIGMOD’98), Seattle, WA, 1998, pp. 85–93.

11. Han, J., Dong, G. and Yin, Y. Efficient mining of partial periodic patterns in time series database, in Proceedings of the 1999 International Conference on Data Engineering (ICDE'99), Sydney, Australia, 1999, pp. 106–115.

12. Dong, G. and Li, J. "Efficient mining of emerging patterns: discovering trends and differences", in Proceedings of the 1999 International Conference on Knowledge Discovery and Data Mining (KDD'99), San Diego, CA, 1999, pp. 43–52.

13. Liu, B., Hsu, W. and Ma, Y. "Integrating classification and association rule mining", in the Proceedings of the 1998 International Conference on Knowledge Discovery and Data Mining (KDD'98), 1998, New York, NY, pp. 80–86.

14. Agrawal, R., Gehrke, J., Gunopulos, D. and Raghavan, P. "Automatic subspace clustering of high dimensional data for data mining applications", in the Proceedings of the 1998 ACM-SIGMOD