Studying Netconf in Hybrid Rule Ordering Strategies for Associative Classification

Raudel Hernández-León[1], José Hernández-Palancar[1], J.A. Carrasco-Ochoa[2], and José Fco. Martínez-Trinidad[2]

[1] Centro de Aplicaciones de Tecnologías de Avanzada (CENATAV), 7a ♯ 21406 e/ 214 and 216, Rpto. Siboney, Playa, C.P. 12200, La Habana, Cuba
{rhernandez,jpalancar}@cenatav.co.cu

[2] Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) Luis Enrique Erro No. 1, Sta. María Tonantzintla, Puebla, C.P. 72840, México
{ariel,fmartine}@ccc.inaoep.mx

Abstract. In Associative Classification, building a classifier based on Class Association Rules (CARs) consists in finding an ordered CAR list by applying a rule ordering strategy. Since this CAR list will be used to build a classifier, it is important to develop a good rule ordering strategy. In this paper, we introduce four novel hybrid rule ordering strategies; the first three combine the Netconf measure with Support-Confidence based rule ordering strategies. The fourth strategy, called Hybrid Specific Rules/Netconf (SR/NF), combines the Netconf measure with a rule ordering strategy based on the CAR's size. The experiments show that the proposed strategies obtain better classification accuracy than the best ordering strategies reported in the literature.

Keywords: Classification, class association rules, rule ordering strategy.

1 Introduction

Associative Classification, also known as Classification Association Rule Mining (CARM) is a well-known Data Mining technique for the extraction of Class Association Rules (CARs) from a given class-transaction dataset. The aim of Associative Classification is to build a classifier to predict the class of "unseen" transactions. In general, a CAR describes an implicative co-occurring relationship between a set of items (itemset) and a pre-defined class, expressed as "$\langle item_1, \ldots, item_n \rangle \Rightarrow class$".

Associative classification has been applied to many tasks including text classification [6], automatic image annotation [19], automatic error detection [16], determination of cotton yarn quality [2], among others.

In CARM, it is assumed that a set of items $I = \{i_1, i_2, \ldots, i_n\}$, a set of predefined classes $C = \{c_1, c_2, \ldots, c_m\}$ and a class-transaction dataset T are given. Each transaction $T_j \in T$ comprises a set of items $I_j \subseteq I$ and a class $c \in C$. A CAR has the form $X \Rightarrow c$, with $X \subseteq I$ and $c \in C$. The rule $X \Rightarrow c$ holds in the dataset T with certain Support s and Confidence α, where s is the fraction

J.F. Martínez-Trinidad et al. (Eds.): MCPR 2014, LNCS 8495, pp. 51–60, 2014.

of transactions in T that contain $X \cup \{c\}$, and α is the ratio of the number of transactions in T containing $X \cup \{c\}$ from the total number of transactions containing X; "alpha" represents how "strongly" the antecedent X implies the consequent c.

Several classifiers based on CARs have been developed [15, 14, 9, 22, 12, 13]. Regardless of the CARM approach, a CAR based classifier is usually presented as an ordered list of CARs based on a rule ordering strategy. Therefore, we could obtain a more accurate classifier by developing a better rule ordering strategy.

In this paper, we propose four novel hybrid rule ordering strategies, which combine the Netconf measure with four rule ordering strategies based on Support and Confidence, as well as with a rule ordering strategy based on the CAR's size. Our experiments were conducted using several datasets taken from the UCI Machine Learning Repository [3], all of them used in other representative works [20–22]. The results obtained using the proposed hybrid rule ordering strategies show good performance, regarding the accuracy of classification, compared against the best hybrid rule ordering strategies reported in the literature.

This paper is organized as follows: The next section describes the related work. In Section 3, the new hybrid rule ordering strategies are introduced. In Section 4, experimental results comparing our hybrid rule ordering strategies against the best ones reported in the literature are shown. Finally, our conclusions are given in Section 5.

2 Related Work

Broadly, the CAR-based classifiers can be categorized into two groups according to the way the CARs are generated: (1) **Two stage classifiers** where all CARs satisfying the Support and Confidence thresholds are mined in a first stage; and later, in a second stage, a classifier is built by selecting an ordered subset of CARs (CBA [15], CMAR [14]), and (2) **Integrated classifiers** where a reduced set of CARs is built in a single step (TFPC [9]), these algorithms avoid the coverage process by directly generating a subset of CARs.

2.1 Case Satisfaction Mechanisms

In [22], the authors summarize three case satisfaction mechanisms that have been employed for classifying "unseen" transactions in CAR based classifiers.

1. **Best Rule:** This mechanism selects, according to an ordering imposed on the list of CARs [22], the first (the best) rule that satisfies the transaction to be classified, and it assigns to this transaction the class associated to the best rule.
2. **Best K Rules:** For each pre-defined class, this mechanism selects the first (top) K rules satisfying the transaction to be classified and it assigns the class for this transaction applying an averaging process as those used in [20].
3. **All Rules:** This mechanism selects all rules satisfying the given transaction and it assigns the class applying an averaging process [14].

2.2 Rule Ordering Strategies

Once a subset of CARs has been obtained, regardless the way they were generated, the CARs are ordered. In [4, 20–22] the authors established six main CAR ordering strategies divided in two groups:

Group 1: Based on Support-Confidence

a) CSA (Confidence - Support - Antecedent size): The CSA rule ordering strategy sorts the rules in descending order according to their Confidence. Those CARs with the same Confidence value are sorted in descending order according to their Supports, and if the tie persists, CSA sorts the rules in ascending order according to their antecedent sizes [15, 14].
b) ACS (Antecedent size - Confidence - Support): The ACS rule ordering strategy is a variation of CSA, but it takes into account the antecedent size as first ordering criterion followed by Confidence and Support criteria [9].
c) L^3: This rule ordering strategy was proposed in [4] and it is also a variation of CSA, but it sorts the rules in descending order according to their antecedent size, as third ordering criterion.

Group 2: Based on Weighting

e) WRA (Weighted Relative Accuracy): The WRA rule ordering strategy, proposed in [9], assigns to each CAR a weight (based on their Support and Confidence) and then sorts the set of CARs in descending order according to these weights. The weight for a CAR $X \Rightarrow c$ is computed as $Sup(X)(Conf(X \Rightarrow c) - Sup(X))$.
f) LAP (Laplace Expected Error Estimate): The LAP rule ordering strategy was introduced in [8] and it has been used to sort the CARs in some classifiers [21]. LAP rule ordering strategy is similar to WRA but LAP defines the weight of the CARs in a different way, also based on their Support and Confidence. Given a rule $X \Rightarrow c$, LAP is defined as $\frac{Sup(X \Rightarrow c)+1}{Sup(X)+|C|}$, where C is the set of predefined classes.
g) χ^2 (Chi-Square): The χ^2 rule ordering strategy is a well-known technique in statistics, which can be used to determine whether two variables are independent or related. After computing an additive χ^2 value for each CAR (also based on their Support and Confidence), these values are used to sort the rules in descending order [14].

Additionally, hybrid rule ordering strategies have been proposed in [20–22] by combining one rule ordering strategy taken from the Support-Confidence based group (group 1) and another one taken from the Weighting based group (group 2). In general, let A, B be rule ordering strategies from groups 1 and 2 respectively. The rule ordering strategy "Hybrid A/B" is defined as follows:

– For each predefined class, this ordering strategy selects, from the original list, the Best K Rules in B manner; the remainder of the original list is sorted in A manner; after, this strategy reorders the selected K Rules in A manner. Finally, these K Rules are put at the front of the remainder of the original rule list which has been already sorted in A manner.

The experiments in [20–22] showed that the obtained classification accuracies using these hybrid rule ordering strategies were better than the ones obtained using their "parents", e.g., the obtained accuracy using the Hybrid CSA/WRA rule ordering strategy was better than the accuracies obtained using CSA and WRA separately.

In [7], the authors proposed a hybrid rule ordering strategy, called MLRP (Multi-level Rule Priority), which sorts the CARs using rule priority to reduce the influence of rule dependence. The rule dependence problem occurs when (during the database coverage analysis) a training transaction O is covered by several CARs concurrently and one of them is used first for classifying O [7]. In this case, because the transaction O has been classified, the confidences of all other CARs covering O are recalculated excluding the transaction O, implying possible changes in the CARs order.

In a first step, MLPR adopts the L^3 rule ordering strategy as initial order. The L^3 strategy gives higher priority to higher confidence and longer rules. In a second step, rule dependencies are computed and stored in a matrix EWM (Effective Weight Matrix), and later, these dependencies are used to obtain the final order of the CARs [7].

In [13], a novel measure called Netconf [1] (see Eq. 1) was used to compute and sort the set of CARs; in this paper we will call Netconf rule ordering strategy (NF) to this approach. Some useful properties of the Netconf measure are the following:

- Netconf holds the statistical independence property, therefore $Netconf(X \Rightarrow c) = 0 \Leftrightarrow Sup(X \Rightarrow c) = Sup(X)Sup(c)$.
- $Netconf(X \Rightarrow c) \neq Netconf(c \Rightarrow X)$ if $Sup(X) \neq Sup(c)$, which means that Netconf is not symmetric, therefore, it can indicate the strength of implication in both directions.
- Netconf takes values in $[-1, 1]$, positive values represent positive dependencies, negative values represent negative dependencies and a zero value represents independence.
- Netconf satisfies some properties which, according to Shapiro [17], should be satisfied by every good quality measure used for separating strong rules from weak rules.

$$Netconf(X \Rightarrow c) = \frac{Sup(X \Rightarrow c) - Sup(X)Sup(c)}{Sup(X)(1 - Sup(X))} \tag{1}$$

In addition, in [13] the authors showed that Netconf solves the drawbacks of the Support and Confidence measures, which have been mentioned in several works [5, 18]. According to the above defined groups, the Netconf measure can be considered as a rule ordering strategy belonging to the Weighting based group. Therefore, in this paper, we investigate the use of the Netconf rule ordering strategy combined with rule ordering strategies from the Support-Confidence group in order to determine whether it is possible to obtain better hybrid rule ordering strategies than those proposed in the literature.

3 Novel Hybrid Rule Ordering Strategies

In this section, we present four novel hybrid rule ordering strategies. Following the results reported in [13], we propose to combine the Netconf rule ordering strategy (NF) with the well known Support-Confidence based rule ordering strategies (CSA, ACS and L^3), which gives as results the next three hybrid rule ordering strategies: (1) Hybrid CSA/NF, (2) Hybrid ACS/NF and (3) Hybrid L^3/NF.

Our hypothesis is that Netconf combined with CSA, ACS and L^3 can outperform the best hybrid strategies reported in the literature [20–22, 7].

A fourth hybrid rule ordering strategy based on Netconf (NF strategy) and the CAR's size (Specific Rules strategy) is also introduced in this paper. As it was shown in [12, 13], the Specific Rule strategy (SR) considers the CAR's size in descending order, favoring specific (large) rules since specific rules would involve more items from the unseen transactions than general (short) rules [13]. Notice that all rule ordering strategies of group 1 are also based on the CAR's size but this criterion is considered in ascending order (CSA and ACS), favoring general rules. In the case of L^3, the CAR's size is considered in descending order but it is applied after the Confidence and Support criteria (both applied in descending order), therefore, the general rules are favored because of the Support download closure [7]. Unlike the L^3 strategy, the SR strategy considers the CAR's size as first (and unique) criterion order, favoring specific rules. Thus, this new hybrid rule ordering strategy is called Hybrid Specific Rules/Netconf (Hybrid SR/NF).

In this case, as in the former proposed hybrid rule ordering strategies, our hypothesis is that Netconf combined with the CAR's size can outperform the best hybrid strategies reported in the literature [20–22, 7]. The overall procedure of the above fifth hybrid rule ordering strategies is shown in Algorithm 1.

In lines $2-5$, for all CARs belonging to list L, their Netconf are calculated and stored in the list L^{NF}. In line 6, the list L^{NF} is sorted in descending order according to the Netconf value. In line 7, the top K CARs of L^{NF} are selected and stored in L^{topK}. Later, the remainder of the L^{NF} list is stored in the L^X list (line 8). In lines $9-10$ both lists L^{topK} and L^X, are sorted in X manner. Finally, in line 11, the list L^{topK} is placed at front of the list L^X.

4 Experimental Results

In this section, we aim to evaluate the novel hybrid rule ordering strategies introduced in this paper; comparing their accuracy of classification against the best hybrid rule ordering strategies reported in the literature [20–22, 7]. The evaluation of all strategies was obtained using the TFPC classifier [10] coupled with the "Best K Rules" case satisfaction mechanism (with K equal to 5 as proposed in

Algorithm 1. Hybrid X/NF

Input: A list of CARs L
Output: A re-ordered list of CARs L^{Hybrid} ordered in a hybrid manner

$L^{NF} = L^{topK} = L^{X} = L^{Hybrid} \leftarrow \emptyset$
forall $r \in L$ **do**
    calculate the *Netconf* Γ of r
    add r jointly with its Γ value to L^{NF}
end
sort L^{NF} in a descending order according to Γ
$L^{topK} \leftarrow$ **select** the top K CARs $\in L^{NF}$
$L^{X} \leftarrow L^{NF} - L^{topK}$
sort L^{topK} in X manner
sort L^{X} in X manner
$L^{Hybrid} \leftarrow$ **put** L^{topK} at front of L^{X}
return L^{Hybrid}

[20]), although any classifier coupled with the "Best K Rules" case satisfaction mechanism could be used. In our experiments, for CSA/NF, ACS/NF, L^3/NF and SR/NF strategies, we also compute the Netconf value of the CARs in order to apply the proposed ordering strategies. Our tests were run on a 1.86 GHz Intel(R) Core(TM)2 CPU with 1.00 GB DDR2 of RAM, running Windows XP SP2.

The experiments were conducted using the datasets reported in [20–22], all of them taken from the UCI Machine Learning Repository [3]. It should be noticed that our experiments were done using ten-fold cross-validation, reporting the average over the ten folds. The same folds were used to evaluate all the rule ordering strategies. Additionally, in order to perform a fair comparison as it was reported in other works [14, 10, 22, 13], we set the Confidence threshold to 0.5, the Support threshold to 0.01 and the Netconf threshold to 0.5. Finally, for Algorithm 1 (see Section 3), we set K equal to 5 (the same value used in other works [20–22]).

In the first four columns of Table 1, we show the average accuracy that we get at applying the best no-hybrids rule ordering strategies, based on Support-Confidence [22]; additionally, in the last two columns we show the average accuracy that we get at applying NF and SR rule ordering strategies, respectively. From this set of results, it can be seen that the NF rule ordering strategy worked better than all other evaluated ordering strategies since as we can see from Table 1, NF gets the maximum accuracy in 13 out of the 19 datasets.

As we announced at section 3 of this paper, our hypothesis is that Netconf combined with CSA, ACS, L^3 and SR can overcome the best hybrid strategies reported in the literature. Thus, in order to test our hypothesis, we performed an experiment comparing the results of our novel hybrid rule ordering strategies against the best rule ordering strategies reported in the literature [20–22, 7].

Table 1. Classification accuracy using the main non-hybrid rule ordering strategies from group 1 as well as using the Netconf (NF) and Specific Rule (SR) ordering strategies

Dataset	CSA	ACS	L^3	NF	SR
adult	80.80	74.70	80.80	**81.68**	81.43
anneal	88.29	75.58	88.51	**92.21**	**92.21**
breast	**89.99**	**89.99**	**89.99**	84.12	83.76
connect4	**65.83**	65.18	**65.83**	62.05	62.05
flare	84.30	84.30	84.52	85.87	**86.21**
glass	64.97	50.74	64.97	**67.49**	67.12
heart	51.42	39.76	51.42	**54.45**	**54.45**
hepatitis	81.83	48.50	82.03	**84.16**	**84.16**
horseColic	79.07	41.11	79.39	**82.56**	81.73
ionosphere	**86.34**	64.67	**86.34**	83.92	84.09
iris	95.33	95.33	95.33	**95.76**	95.21
led7	68.72	64.22	68.72	**73.32**	**73.32**
mushroom	99.04	64.92	99.04	**99.40**	98.36
pageBlocks	89.99	89.99	89.99	**91.83**	**91.83**
pima	74.37	73.85	75.06	**76.79**	76.01
nursery	77.75	55.08	77.75	**78.12**	77.96
soybean-large	88.01	86.10	88.01	**89.35**	88.73
ticTacToe	67.10	39.03	**67.45**	66.69	66.12
wine	71.51	50.28	71.51	71.23	**71.72**
Average	79.19	65.96	79.30	**80.05**	79.81

In the last four columns of Table 2, we show the accuracies obtained by our proposed Hybrid strategies CSA/NF, ACS/NF, L^3/NF and SR/NF (see columns 5–8). In columns 2–4 of the same table we show the results of the best rule ordering strategies reported in the literature (CSA/χ^2, ACS/LA and MLRP). The first interesting thing that we can observe from our experiments is that, as it was established in [20–22], an Hybrid A/B strategy obtains better classification accuracy results than its "parents" A or B separately. For example, we can see in Table 2 that the classification accuracy obtained using Hybrid SR/NF (81.55) is greater than the accuracies obtained by its "parents" (see Table 1) SR (79.81) and NF (80.05). Analogously, the classification accuracies obtained by using Hybrid CSA/NF, Hybrid ACS/NF and Hybrid L^3/NF are greater than the accuracies obtained by their "parents". From our results reported in Table 2, we can see that all the rule ordering strategies proposed in this paper outperform the best rule ordering strategies reported in the literature. It means that our hypothesis is true for all the proposed strategies except for ACS/NF. This occurs because the ACS strategy, different from CSA, L^3 and SR strategies, favors short rules.

As we mentioned above, we evaluate all hybrid rule ordering strategies applying the "Best K Rules" case satisfaction mechanism (with K equal to 5 as in [20]). However, in order to show that these strategies are independent of the case satisfaction mechanism used, we performed the same experiments of

Table 2. Classification accuracy using hybrid rule ordering strategies

Dataset	CSA/χ^2	ACS/LA	MLRP	CSA/NF	ACS/NF	L^3/NF	SR/NF
adult	80.08	83.86	82.56	82.78	**84.25**	82.78	83.37
anneal	89.92	80.73	**93.77**	93.48	88.51	93.64	93.72
breast	**91.00**	89.59	83.66	84.12	89.60	84.12	85.26
connect4	65.88	65.28	62.06	62.13	65.41	62.15	62.37
flare	84.51	84.49	86.29	86.26	85.85	86.35	**86.52**
glass	65.59	60.82	67.80	67.71	65.40	67.73	**68.02**
heart	51.14	50.66	54.09	54.52	53.31	54.54	**56.51**
hepatitis	80.50	76.83	84.47	84.38	83.94	84.65	**86.43**
horseColic	81.24	81.01	82.90	82.56	81.85	82.88	**83.25**
ionosphere	84.05	84.90	83.75	84.10	**85.23**	84.10	84.31
iris	95.33	95.33	95.82	95.87	95.33	95.91	**96.93**
led7	68.92	64.85	73.41	73.50	73.06	73.47	**74.69**
mushroom	98.52	98.82	98.96	99.40	98.52	99.40	**99.52**
pageBlocks	90.72	90.16	92.22	91.91	91.47	91.88	**92.61**
pima	74.63	74.50	77.18	76.84	75.87	77.23	**77.65**
nursery	78.52	66.83	79.50	79.25	78.41	79.19	**80.12**
soybean-large	88.23	77.66	89.93	89.98	89.03	89.94	90.56
ticTacToe	67.94	63.16	66.74	66.69	66.31	66.86	**72.43**
wine	74.52	72.31	71.21	71.44	71.95	71.44	**75.14**
Average	79.54	76.94	80.33	80.36	80.17	80.43	**81.55**

Table 2 applying the other two case satisfaction mechanisms ("Best Rule" and "All Rules"); all these results appear in Table 3. Something interesting that can be seen from these results is that if we rank the strategies based on the accuracy they reach for the three case satisfaction mechanisms, we get the same ranking (see the last row in Table 3). However, the best average accuracies are obtained when the "Best K Rules" mechanism is used.

Table 3. Average classification accuracy throughout the 19 datasets using other Case Satisfaction Mechanisms (CSM)

CSM	CSA/χ^2	ACS/LA	MLRP	CSA/NF	ACS/NF	L^3/NF	SR/NF
Best Rule	79.46	76.86	80.26	80.29	80.12	80.37	81.48
All Rules	78.08	74.63	78.82	78.90	78.66	79.11	80.16
Best K Rules	**79.54**	**76.94**	**80.33**	**80.36**	**80.17**	**80.43**	**81.55**
Ranking	6	7	4	3	5	2	1

From our experiments we can conclude that in general our hybrid rule ordering strategies outperform the best hybrid strategies reported in the literature. We can see in Table 2 that the Hybrid SR/NF strategy gets an average accuracy, throughout the 19 datasets, of 81.55; it represents an improvement of 1.12 percentage points in accuracy with respect to the second place (Hybrid L^3/NF).

Finally, in order to determine if the results shown in Table 2 are statistically significant, we performed a pairwise comparison between all tested hybrid rule ordering strategies. Each cell (i, j), in Table 4, contains the number of datasets where the strategy of row i significantly Win/Lose to the strategy of column j. We detected ties using a one-tailed T-Test [11] with significance level of 0.05. The results in the pairwise comparison reveal that the proposed Hybrid SR/NF rule ordering strategy beats in accuracy all other evaluated rule ordering strategies, over most of the tested datasets.

Table 4. Pairwise comparison between all evaluated hybrid rule ordering strategies. Each cell (i, j) contains the number of datasets where the strategy of row i significantly Win/Lose to the strategy of column j, over the 19 selected datasets.

	CSA/χ^2	ACS/LA	MLRP	CSA/NF	ACS/NF	L^3/NF	SR/NF
CSA/χ^2		8/1	4/11	4/10	6/4	3/9	2/14
ACS/LA	1/8		5/12	4/12	0/3	4/12	1/14
MLRP	11/4	12/5		0/0	4/4	0/0	0/7
CSA/NF	10/4	12/4	0/0		7/4	0/0	0/9
ACS/NF	4/6	3/0	4/4	4/7		3/3	3/13
L^3/NF	9/3	12/4	0/0	0/0	3/3		0/5
SR/NF	14/2	14/1	7/0	9/0	13/3	5/0	

5 Conclusions

In this paper, we have proposed four novel hybrid rule ordering strategies based on the Netconf measure. Our experimental results show the proposed hybrid strategies always reached better classification accuracies than those obtained by their parents rule ordering strategies, as occurs with other hybrid strategies. From the experiments, we can conclude that our novel hybrid rule ordering strategies (with exception of ACS/NF) have better performance than the best hybrid rule ordering approaches reported in the literature. In particular, our Hybrid SR/NF strategy reached the best classification accuracy.

Acknowledgements. This work was partly supported by the National Council of Science and Technology of Mexico (CONACyT) through the project grants CB2008-106443 and CB2008-106366.

References

1. Ahn, K.I., Kim, J.Y.: Efficient Mining of Frequent Itemsets and a Measure of Interest for Association Rule Mining. Information and Knowledge Management 3(3), 245–257 (2004)
2. Amin, A.E.: A novel Classification Model for Cotton Yarn Quality based on Trained Neural Network using Genetic Algorithm. Knowledge-Based Systems 360, 124–132 (2013)

3. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository (2007), http://www.ics.uci.edu/~mlearn/MLRepository.html
4. Baralis, E., Garza, P.: A Lazy Approach to Pruning Classification Rules. In: Proc. of the ICDM, pp. 35–42 (2002)
5. Berzal, F., Blanco, I., Sánchez, D., Vila, M.A.: Measuring the Accuracy and Interest of Association Rules: A new framework. Intelligent Data Analysis 6(3), 221–235 (2002)
6. Buddeewong, S., Kreesuradej, W.: A new Association Rule-Based Text Classifier Algorithm. In: Proc. of the 17th IEEE International Conference on Tools with Artificial Intelligence, pp. 684–685 (2005)
7. Chun-Hao, C., Rui-Dong, C., Cho-Ming, L., Chih-Yang, C.: Improving the Performance of Association Classifiers by Rule Prioritization. Knowledge-Based Systems 36, 59–67 (2012)
8. Clark, P., Boswell, R.: Rule Induction with CN2: Some Recent Improvments. In: Kodratoff, Y. (ed.) EWSL 1991. LNCS, vol. 482, pp. 151–163. Springer, Heidelberg (1991)
9. Coenen, F., Leng, P.: An Evaluation of Approaches to Classification Rule Selection. In: Proc. of the ICDM, pp. 359–362 (2004)
10. Coenen, F., Leng, P., Zhang, L.: Threshold Tuning for Improved Classification Association Rule Mining. In: Ho, T.-B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 216–225. Springer, Heidelberg (2005)
11. Demšar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. J. Mach. Learn. Res. 7, 1–30 (2006)
12. Hernández, R., Carrasco, J.A., Martínez, F.J., Hernández, J.: Classifying using Specific Rules with High Confidence. In: Proc. of the MICAI, pp. 75–80 (2010)
13. Hernández, R., Carrasco, J.A., Martínez, F.J., Hernández, J.: CAR-NF: A Classifier based on Specific Rules with High Netconf. Intelligent Data Analysis 16(1), 49–68 (2012)
14. Li, W., Han, J., Pei, J.: CMAR: Accurate and Efficient Classification based on Multiple Class-Association Rules. In: Proc. of the ICDM, pp. 369–376 (2001)
15. Liu, B., Hsu, W. and Ma, Y.: Integrating Classification and Association Rule Mining. In: Proc. of the KDD, pp. 80–86, 1998.
16. Malik, W.A., Unwin, A.: Automated Error Detection using Association Rules. Intelligent Data Analysis 15(5), 749–761 (2011)
17. Shapiro, G.: Discovery, Analysis, and Presentation of Strong Rules. In: Knowledge Discovery in Databases. LNCS, pp. 229–238 (1991)
18. Steinbach, M., Kumar, V.: Generalizing the notion of Confidence. Knowl. Inf. Syst. 12(3), 279–299 (2007)
19. Teredesai, A.M., Ahmad, M.A., Kanodia, J., Gaborski, R.S.: CoMMA: a Framework for Integrated Multimedia Mining using Multi-Relational Associations. Knowl. Inf. Syst. 10(2), 135–162 (2006)
20. Wang, Y.J., Xin, Q., Coenen, F.: A Novel Rule Weighting Approach in Classification Association Rule Mining. In: International Conference on Data Mining Workshops, pp. 271–276 (2007)
21. Wang, Y.J., Xin, Q., Coenen, F.: A Novel Rule Ordering Approach in Classification Association Rule Mining. In: Perner, P. (ed.) MLDM 2007. LNCS (LNAI), vol. 4571, pp. 339–348. Springer, Heidelberg (2007)
22. Wang, Y.J., Xin, Q., Coenen, F.: Hybrid Rule Ordering in Classification Association Rule Mining. Trans. MLDM 1(1), 1–15 (2008)