Journal of Korea Multimedia Society Vol. 11, No. 12, DECEMBER 2008(pp. 1739-1748)

A Method to Minimize Classification Rules Based on Data Mining and Logic Synthesis

Jongwan Kim†

ABSTRACT

When we conduct a data mining procedure on sample data sources, several rules are generated. But some rules are redundant or logically disjoint and therefore they can be removed. We suggest a new rule minimization algorithm inspired from logic synthesis to improve comprehensibility and eliminate redundant rules. The method can merge several relevant rules into one based on data mining and logic synthesis without high loss of accuracy. In case of two or more rules are candidates to be merged, we merge the rules with the attribute having the lowest information gain. To show the proposed method could be a reasonable solution, we applied the proposed approach to a problem domain constructing user preferred ontology in anti-spam systems.

Key words: data mining, logic synthesis, rule minimization

1. INTRODUCTION

There are several works on multi-valued logic in machine learning that can be used for rule minimization and optimization. Files and Perkowski explore any multi-valued logic synthesis (MVLS) method [1]. They described some concepts of machine learning matched nicely with MVLS and showed MVLS outperformed C4.5, the widely used classification algorithm and Espresso, an industry standard logic minimization tool distributed by the UC Berkeley. A minimal rule generation algorithm called R-MINI that was an adaptation of a well established heuristic switching function minimization technique, MINI, was proposed [2]. The main mechanism of R-MINI is the process reducing the number of rules was repeated application of generalization and specialization operations to the rule set while maintaining completeness and consistency. Also, an iterative mining for rules with constrained antecedents were reported [3]. This approach was an iterative algorithm that could exploit mining information gained in previous steps to efficiently answer subsequent queries. Zaki and Ramakrishnan presented a method to reason about a collection of sets using redescription mining [4]. Redescription mining is a newly introduced data mining topic that seeks to find subsets of data that afford multiple definitions. This work used Karnaugh map as a conceptual tool to understand redescription spaces. The input to redescription mining is a vocabulary of sets or binary propositions over a domain and the goal is to construct two distinct expressions from this vocabulary that induce the same subset over the domain. It can be viewed as a generalization of association rule mining. In this work, we suggest a simple rule minimization approach based on logic synthesis and data mining.

The proposed rule minimization approach will be applied to anti-spam mail systems. Spam or junk mail is unsolicited, unwanted email sent indiscriminately by a spammer having no current re-

※ Corresponding Author : Jongwan Kim, Address : (712-714) 15 Nairi Jillyang Gycongsan Gyeongbuk, TEL : +82-53-850-6575, FAX : +82-53-850-6589, E-mail : jwkim@daegu.ac.kr.

Receipt date : May 7, 2008, Approval date : Aug. 4, 2008

† Member, Professor, School of Computer and Information Technology, Daegu University

※ This work was supported by the Daegu University Research Grant, 2007.

lationship with the recipient. Most email software provides some automatic spam mail filtering mechanism, typically in the form of blacklists or keyword-based filters. This filtering technique was somewhat effective in the beginning, but it gradually declined with accuracy over time because spammers started using personal sounding subjects to avoid the keyword filters [5]. A variety of machine learning algorithms such as naïve Bayesian classifier (NBC) [6] and support vector machine (SVM) [7] have been used for email categorization task. While these anti-spam filters achieve statistically impressive accuracy rates, they still have two problems. First, non-spam called ham or legitimate email are tagged as spam. And spam mails are stacked in a mailbox. Especially many people sometimes experience important emails are classified to the spam folder by mail server filters due to drawback of simple keyword matching. Also, users' behavior for emails could be different according to their preferences. Therefore it is desirable to give a user-oriented anti-spam service based on user preferences. We will show that the proposed rule minimization is useful to the user-oriented anti-spam system.

The paper is organized as the following. Section 2 describes the data collection and preprocessing. Section 3 presents a new rule minimization methodology to reduce rules generated by a decision tree algorithm. Experimental results are presented in Section 4 and Section 5 concludes this work.

2. DATA COLLECTION AND PREPROCESSING

Data preparation should be done first to construct some ontology from a specific domain data. Some email contents and user's responses were collected from some undergraduates studying computer science at a college. We have conducted several experiments on content-based anti-spam filtering and already held many sample spam and ham mails mostly in Korean. From the previous experience, we started to this work and wanted to develop anti-spam mail system based on personal interests and responses among similar users instead of simple email contents. To prepare the data, we designed user profile format and user response categories to emails.

Many web mail services such as Yahoo and Hotmail request us to register user's personal information. Like this registration format, we chose {Age, Gender, Required_Hits, News, Finance, Sports, Adults, TvMovieMusic, Kids, Games, Travel, Shopping, Jobs, RealEstates} attributes to be included in a user profile. Since every participant is twenties and college students, Age attribute is classified into two groups, FS (= freshman and sophomore) and JS (= junior and senior). Required_Hits (from now on, RHit) was originally adopted in Spam Assassin [5], which means how many hits are required before a mail is considered spam. Differently from Spam Assassin using numbers, we use linguistic terms such as Very Weak, Weak, Neutral, Strong, and Very Strong because we do not consider email contents. If user wants a strong spam filter, then each one chooses Weak value. On the contrary, people would choose Strong required hits when they prefer a weak filter.

Feature selection involves searching through all possible combination of features in the candidate feature set to find which subset of features works best for prediction. A few of the mechanisms designed to find the optimum numbers of features are information gain, mutual information, and chi squared test. In comparing learning algorithms, Yang and Pedersen found that, except for mutual information, all these feature selection methods had similar performance and characteristics in text categorization [8]. To select features, we calculated information gains (IG) for all attributes and then chose several top attributes from them.

In also, email recipients usually respond to an email in mail box in four ways. When they have

no interest on the mail, they just delete it. If they think the mail is important and valuable to respond, then they reply to the sender. Regardless of reply or not, they sometimes just hold some mails in their inbox because the emails might be needed in the future. Finally when a spam mail is given to them, most users move the kind of mails into spam folder because they dislike receiving the kind of spam mails again. Surely, some users can delete and move it simultaneously, but others either delete or just move it.

Thus we collected some sample mails, personal preferences of participating users, and their responses to the samples. Most attributes have binary format. For example, if user has an interest on Sports, then he or she checks true, but false is chosen when user has no interest on each attribute. Other attributes including Finance, Adults, and so on have equivalent true/false values as well. Age has {FS, JS} binary format. Male or female is given for Gender. But {VeryWeak, Weak, Neutral, Strong, VeryStrong} are used in case of the RHit attribute. Responses have also four categories {Reply, Delete, Store, Spam} in this work.

3. RULE MINIMIZATION BASED ON DATA MINING AND LOGIC SYNTHESIS

We will describe a procedure to generate rules from sample domain data and reduce the rules through logic synthesis process. To construct an anti-spam rule set, we performed two step processes. The first step is to find good rules representing well their preferences and email responses collected for several users. In the next step, we applied two new rule pruning procedures excluding redundant rules and selecting highly comprehensible ones. These steps will be described in detail.

First, we tried to discover association rules between various groups of users and their responses for sample email data. For example, we expected that women usually like shopping and students have strong interests in job recruiting. This intuition was realized after we applied association mining to user preference data set. In the same way, we wanted to find unknown correlations between user profiles and user log files, which include user responses to sample emails. Thus, we chose the typical decision tree algorithm, ID3 [9], to train sample email preference data. Our sample data are composed of mostly binary features and some are nominal features as described in Section 2. So ID3 is suitable to discover representative rules from the data set. After ID3 mining was performed, a decision tree is generated. We can convert the decision tree into rules by ascribing each path of the tree with a rule. From a root node to internal nodes in each path are considered as antecedent conditions of each rule and the leaf node as a conclusion of each rule. To evaluate which rule is good, we count the accuracy by calculating the proportion of testing instances which match the rules.

Second, we propose two new rule-pruning procedures in order to exclude redundant rules and select highly comprehensible ones. The procedures are kinds of rule minimization approaches inspired from logic synthesis. Two representative logic minimization methods are algebraic minimization and Karnaugh map (K-map) [10]. For very complicated problems the former method can be done using special software analysis programs. While K-map is also limited to problems with up to 4 binary inputs, it is well known as simple and easy method to understand Boolean logic simplification. It is possible to find two or more simplified logic expressions in a K-map. For example, a function f (A, B, C) = (1, 3, 4, 5, 6, 7) composing of three input variables A, B, and C. This function f has 6 min terms, {001, 011, 100, 101, 110, 111}. Two kinds of logic minimizations f = C + AC'and f = A + A'C are possible, even if the function f is fixed.

Surely, two expressions are the same with respect to logic. We got an idea about rule minimization from this K-map example. Thus, we tried to fill up empty antecedent conditions in a rule with every possible combination of variables or attributes in order to find another unknown rule and then mine again by using ID3. As we expected, an experimental result was a little different from the original ID3 mining (ORG). We call this method re-mining (REM) approach and got a different result from the experiments in Section 4.3. In other words, a little reduced rule set was derived for the same training data as some similar rules were merged into one. However, this kind of re-mining is somewhat weird because we have conducted a mining again for a rule set found by ID3.

So we present another rule minimization approach based on logic synthesis to prune rules generated by data mining. Our rule minimization approach is called hybrid rule pruning (HRP). There are several variables in a rule set derived from data mining. Most of variables are Boolean but some of them are multi-valued such as RHit = {Very Weak (VW), Neutral (N), Strong (S), Very Strong (VS)}. In hybrid logic synthesis, if two corresponding logic of any variable are distinct, two rules are merged into one and then the antecedent condition of the variable will be omitted. Therefore more simple rules can be constructed. The following example explains the idea well. There are two similar rules R1 and R2 with only one different antecedent condition for Adult attribute.

R1: if Age = FS and RHit = S and News = F and Adult = F and Game = T then Response = Spam

R2: if Age = FS and RHit = S and News = F and Adult = T and Game = T then Response = Spam

R3: if RHit = VW and News = T and Adult = T then Response = Store

R4: if RHit = S and News = T and Adult = T then Response = Store

So the two rules are merged into R1 • R2 where the antecedent conditions F (False) and T (True) of a variable Adult are merged into X (= Null) because Adult = T is in conjunction with Adult = F by logic synthesis operation. The Null condition is omitted to construct a new rule R5.

R5: if Age = FS and RHit = S and News = F and Game = T then Response = Spam

For categorical variables, similar operation can be applied. We represented Boolean and categorical variables as {0/1} binary format and multiple binary bit format, respectively. For example, binary attributes True and False are represented 1 and 0, respectively. Another binary attributes Male/Female are represented 1/0 too. For categorical attributes, we followed Hong's m-bit representation [2]. RHit has four categories, VW=1000, N=0100, S=0010, and VS=0001. Applying R3 and R4 in the above rule example, the hybrid logic synthesis operation will be done as follows:

R3: 1000 1 1 Store

R4: 0010 1 1 Store

R3 • R4: X0X0 1 1 Store

From the conjunction of rule 3 and 4, the antecedent condition of a variable RHit should have two NULL conditions X0X0 and hence the condition is omitted to construct a new rule R6.

R6: if News = T and Adult = T then Response = Store

However, two or more rules sometimes might compete to get a chance merging other rules with different antecedent conditions. To resolve this situation, we consider information gain (IG) of each attribute in a rule set. When two or more candidates are found to be merged, we allow a variable having the attribute with the lowest IG merge the two rules. Intuitively we think the attributes with higher IGs should survive in a rule set. The proposed rule minimization algorithm is described in Figure 1 according to the aforementioned.

```
1. Get rules through ID3 mining
2. Do the following loop for each response case
3. Do while (all of given rules in each response group
   are checked out)
   3.1 find a pivot rule with higher accuracy
   3.2 check every other rule to find rules with only
       one distinct antecedent condition for the pivot
       rule
       if (check = true) {
          if (only one pair is found) then merge two
             rules into one new rule removing the
             condition;
          else if (two or more pairs are found) then {
                sort the candidate attributes in as-
                cending order in terms of IG;
                select the attribute with the lowest
                IG to be merged;
                merge the rules into one new rule
                excluding the attribute;
          }
       }
   3.3 change the pivot rule to the one with next
       higher accuracy than now
4. Convert classification rules into a domain ontology
```

Fig. 1. Algorithm: minimizing rules derived by ID3 mining

4. EXPERIMENTS

4.1 System Architecture

We will describe a user-oriented anti-spam mail system with classification rules through data mining and logic synthesis [11]. As we mentioned, each user's response could be different to the same mails. This response is mainly caused by their personal preferences and potential actions. We started from this assumption and decided to show that it is valid in real situation. Thus, we collected preferences for a group of users who are from freshman to senior students at a computer science department. To analyze potential response to various emails, we provided sample emails to a user group and asked them to respond as one of (Reply, Delete, Store, Spam) actions. Reply and Delete actions mean user replies to and deletes this kind of mail, respectively. Store means user hold the mail into mail box. On the contrary, user selects Spam when he or she thinks the mail is a spam. In this research, Reply, Delete, and Store responses are considered to ham mails but only Spam response is considered to spam. Thus, this work is different from conventional anti-spam mail works because we consider user's specific responses into 4 categories.

The architecture of the ontology-based anti-spam mail system is given in Figure 2 [11]. In Figure 2, user profiles were collected from several participant users and user log files were also built from their responses to sample emails. We used ID3 data mining algorithm [9] to find some classification rules between preferences and responses. User preferred ontology was constructed after data mining and rule minimization [11]. We interpret the derived classification rules to an ontology using a formal language, Web-PDDL [12], a strongly typed first order language especially for representing ontologies and mappings between them.

If an email is given to the system, it will be finally judged that the mail is spam or ham (reply, store, delete) by OntoEngine inference module [12]. At this time, our system considers not only email itself but also user information of the email recipient. In Figure 2, user information is the same as the content in the user profile. In fact, user's behavior depends on his/her preference and unpredictable behavior as well as the specific email content. Thus, it is not sufficient to decide whether an email is spam or not with only email content

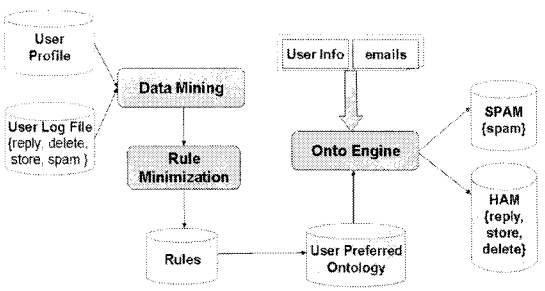


Fig. 2. Architecture of the proposed ontology-based anti-spam mail system

[In training mode]
1. Provide user profile and user log file into the system
2. Do ID3 data mining to find relationship between preferences and responses
3. Perform our rule minimization to find generalized rules by excluding redundant ones
4. Transform the chosen rules into user preference ontology in Web-PDDL.

[In test mode]
1. Give user's preference and an email into the system
2. Examine the user preference ontology to find highly matched one to the given input
3. Decide user's response according to matched score with the ontology by an inference engine.

Fig. 3. Training and testing procedure of the proposed system

and user preference information. However, it is almost impossible to estimate user's whim and is out of this work. Also, it is meaningful to consider user's preference as basic information for a personalized anti-spam mail service. Thus we follow a user preferred ontology approach based on user information and email content [11]. The training procedure and testing scenario of the proposed system are briefly described in Figure 3.

4.2 Performance Measures

Performance measures are required to evaluate a user preference ontology represented in rules in a user-oriented anti-spam system [11]. There are several measures such as misclassification, accuracy, prediction, recall, and so on in anti-spam mail system field [13]. Since these measures are calculated from email contents, they can be called email-oriented measures. However, we aim at user oriented service and hence different measures are needed to show that the proposed user preferred ontology is meaningful to the anti-spam system. In this work, we suggest three measures to achieve this goal [11].

First, rule accuracy or rule confidence is useful to calculate correctness of each rule. Let us consider the antecedent portion and conclusion of each rule. When the input attributes of a test instance exactly match the antecedent conditions of the i-th rule, we increment the match count of the rule, rule[i].match. At this time, if the response of test instance is also the same as the conclusion of the rule, the correct count of the rule, rule[i].correct increments too. Then rule confidence is calculated by dividing rule[i].correct over rule[i].match. Naturally we prefer the rule set with higher rule confidence.

Second, conventional classification accuracy is not appropriate to this anti-spam application. Because current anti-spam filters assume that the response of each user for an email is totally same. However, it is not guaranteed that users'responses are equivalent to every kind of email. So we introduce rule capacity as a measure of how many instances can be accommodated by each rule. If the summation of matching scores of each rule (i rule[i].match) is equal to the total number of test instances, then the rule set can accommodate all instances. Therefore no capacity problem exists. However, it is not easy because we have mined several thousands of instances to tens of rules. We should pass outside instances away from rules in the ontology to conventional content-based email filters such as NBC or SVM and let them process the instances.

Third, we have introduced a rule minimization method and suggest a simple quantified measure for user comprehensibility instead of qualitative statements. We define a matched term ratio, *mt*, for each rule in equation (1).

$$mt[i] = \frac{\text{number of attributes in each instance}}{\text{number of antecedent conditions in rule}[i]} \quad (1)$$

Where i is the index of the i-th rule and the number of antecedent conditions in each rule is the number of attributes compared to test instances. The greater the average value of all matched term ratios is, the simpler and more easily interpretable

the rule set is to humans. For example, a test instance, (Age = FS and RHit = S and News = F and Adults = T and Games = F) and (Response = Spam) is presented. Two rules (R7: if Age = FS and News = F then Response = Spam) and (R8: if Age = FS and RHit = S and News = F and Games = F then Response = Spam) are given. Then $mt[7] = 5/2 = 2.5$ and $mt[8] = 5/4 = 1.25$. Therefore, R7 rule has greater matched term ratio than R8 in terms of quantified comprehensibility. This measure is simple and quantified. Chan and Freitas also measured rule comprehensibility by the average number of terms in the discovered rules [3] but they did not utilize a measure considering the number of input attributes.

4.3 Experimental Results

The above three measures help performance evaluation in terms of rule confidence, capacity, and comprehensibility. To evaluate the proposed rule minimization approaches applied to anti-spam mail systems, we collected a total of 40 sample emails and 3600 records from 90 college students. Since every user responded to 40 sample emails, 3600 (= 40×90) email responses were prepared. Each user gave his or her preference and therefore 90 preferences were collected too. Splitting randomly 3600 records into 2400 training and 1200 testing instances, we performed experiments to observe the three measures aforementioned. Before ID3 mining, we calculated IG for all 14 attributes described in Section 3 and chose the top 5 attributes - Age, RHit, News, Adults, and Games. Surely, Response has been used as a target variable to generate classification rules from a decision tree constructed by ID3.

We got initially 18 rules of which accuracies are greater than 0% through ID3 data mining for 2400 training instances and then we removed 2 rules because they have correctly matched instances less than 5. The reason why we excluded the two rules was to support minimum correct match or coverage and to preserve at least one or more "Reply" rules in the final rule set. As you may know, users' actions are in a broad range of responses and especially reply option of response tends to be different. Therefore it is not easy to find any common rules for the reply response from users having various interests. Finally 16 rules were derived during ID3 mining process (ORG). To evaluate the perform-

Table1. Experimental results for a rule set derived by original ID3 mining (ORG)

No	Rule	Matched term ratio	capacity	rule accuracy
1	Age=JS & R_Hit=N & News=F & Adults=F & Games=F => Response=Delete	1.0	17	82.4%
2	Age=FS & R_Hit=S & News=F & Adults=F & Games=T => Response=Spam	1.0	31	61.3%
3	Age=FS & R_Hit=VS & News=F & Games=T => Response=Spam	1.25	68	57.4%
4	Age=FS & R_Hit=S & News=F & Games=F => Response=Spam	1.25	85	54.1%
5	Age=FS & R_Hit=VW & Adults=T => Response=Spam	1.67	66	53.0%
6	Age=FS & R_Hit=S & News=F & Adults=T & Games=T => Response=Spam	1.0	185	51.9%
7	Age=JS & R_Hit=N & News=T & Adults=T => Response=Store	1.25	32	43.8%
8	Age=FS & R_Hit=N & News=T & Adults=T => Response=Delete	1.25	12	41.7%
9	Age=JS & R_Hit=VS & News=T & Adults=T => Response=Reply	1.25	15	40%
10	Age=JS & R_Hit=S & News=F & Adults=F & Games=F => Response=Spam	1.0	15	40%
11	Age=FS & R_Hit=VS & Games=F => Response=Spam	1.67	22	36.4%
12	Age=FS & R_Hit=N & News=F => Response=Spam	1.67	58	34.5%
13	Age=JS & News=F & Adults=T => Response=Delete	1.67	480	33.1%
14	Age=JS & R_Hit=S & News=F & Adults=F & Games=T => Response=Delete	1.0	25	28%
15	Age=JS & R_Hit=N & News=F & Adults=F & Games=T => Response=Reply	1.0	41	22.0%
16	Age=JS & R_Hit=N & News=T & Adults=F & Games=T => Response=Store	1.0	28	21.4%
average		1.25	1180	43.81%

ance of derived rules, we applied 1200 test instances to the 16 rules. The experimental results such as rule accuracy, capacity, and matched term ratio for the 16 rule set in descending order of accuracy are shown in Table 1. Symbol "&" represents conjunctive "and" in each rule. The proposed rule minimization approaches REM and HRP generated 12 rules, respectively. A little difference between REM and HRP exists in the survived rule set; one less rule about Store response was gotten in REM than HRP, but one more Spam response rule was excluded in HRP compared to REM. So, we presented two 12 rule sets derived by REM and HRP and their experimental results in Table 2 and Table 3, respectively. From the Table 2 and Table 3, we found that users with Age=JS and RHit=Neutral and News=False and Adults=False and Games=False preferences usually responded "Delete" with the about 80% probability when they got emails. The rest of rules explain why each user chooses his or her response option for incoming emails in the same way.

We think there are two reasons why low accuracies for each rule are gotten. The first one is that sample emails are randomly chosen without considering all possible 11 categories including News, Finance, and so on. The distribution of email properties is skewed; some of 11 categories cover most

Table 2. Experimental results for a rule set derived by re-mining (REM)

No	Rule	matched term ratio	capacity	rule accuracy
1	Age=JS & RHit=N & News=F & Adults=F & Games=F => Response=Delete	1.0	17	82.4%
2	Age=FS & RHit=S & News=F => Response=Spam	2.5	301	53.5%
3	Age=FS & RHit=VW => Response=Spam	1.67	66	53.0%
4	Age=FS & RHit=VS => Response=Spam	1.25	90	52.2%
5	Age=FS & RHit=N & News=T => Response=Delete	1.25	12	41.7%
6	Age=JS & RHit=VS & News=T => Response=Reply	1.0	15	40%
7	Age=JS & RHit=S & News=F & Adults=F & Games=F => Response=Spam	1.0	15	40%
8	Age=FS & RHit=N & News=F => Response=Spam	1.67	58	34.5%
9	Age=JS & RHit=N & News=F => Response=Store	1.67	60	33.3%
10	Age=JS & News=F & Adults=T => Response=Delete	1.0	480	33.1%
11	Age=JS & RHit=S & News=F & Adults=F & Games=T => Response=Delete	1.0	25	28.0%
12	Age=JS & RHit=N & News=F & Adults=F & Games=T => Response=Reply	1.0	41	22.0%
average		1.33	1180	42.81%

Table 3. Experimental results for a rule set derived by hybrid rule pruning (HRP)

No	Rule	matched term ratio	capacity	rule accuracy
1	Age=JS & RHit=N & News=F & Adults=F & Games=F => Response=Delete	1.0	17	82.4%
2	Age=FS & RHit=VW & Adults=T => Response=Spam	1.67	501	53.0%
3	Age=FS & News=F => Response=Spam	2.5	66	51.9%
4	Age=JS & RHit=N & News=T & Adults=T => Response=Store	1.25	32	43.8%
5	Age=FS & RHit=N & News=T & Adults=T => Response=Delete	1.25	12	41.7%
6	Age=JS & RHit=S & News=F & Adults=F & Games=F => Response=Spam	1.0	15	40%
7	Age=JS & RHit=VS & News=T & Adults=T => Response=Reply	1.25	15	40%
8	Age=FS & RHit=VS & Games=F => Response=Spam	1.67	22	36.4%
9	Age=JS & News=F & Adults=T => Response=Delete	1.67	480	33.1%
10	Age=JS & RHit=S & News=F & Adults=F & Games=T => Response=Delete	1.0	25	28%
11	Age=JS & RHit=N & News=F & Adults=F & Games=T => Response=Reply	1.0	41	22.0%
12	Age=JS & RHit=N & News=T & Adults=F & Games=T => Response=Store	1.0	28	21.4%
average		1.36	1254	41.14%

Table 4. Summary on experimental results of the rule set derived by the ORG and of the two rule sets by the REM and the HRP.

Method	Number of rules				Rule accuracy	Capacity	Matched term ratio
	Reply	Delete	Store	Spam			
ORG	2	4	2	8	43.81%	1180/1200	1.25
REM	2	4	1	5	42.81%	1180/1200	1.33
Improvement	0	0	50%	37.5%	-2.3%	0%	6.4%
HRP	2	4	2	4	41.14%	1254/1200	1.36
Improvement	0	0	0	50%	-6.1%	6.3%	8.8%

of sample emails and the others cover a few portion. The second one is because user's response can be different to even the same kinds of emails. Thus we could not get high rule accuracies for testing instances. However this kind of user preferred ontology constructed from the logic rules found by data mining contribute to estimate user's response to various kinds of emails and explain why a mail is classified to spam or ham.

Table 4 shows the summary of the experimental results comparing two rule sets derived by the proposed rule pruning approaches, REM and HRP with those derived by the original ID3 mining (ORG). As shown in the table, average rule accuracy was degraded from 43.81% in ORG to 42.81% in REM and 41.14% in HRP, respectively. However 25% of rules were reduced and the average matched term ratio also improved by 6.4% and 8.8% in REM and HRP, respectively. Also, the proposed HRP accommodated over 1200 test instances. It shows that some instances are overlapped in several rules and our system can process almost all instances itself resulting from rule merging. We do not need to pass any instance to content-based filters. Thus we believe that the proposed rule pruning approaches can merge two or more relevant rules into one without significant loss of accuracy. This shorter rule is desirable to construct user preferred ontology because we pass the rules to each user and then the user feedbacks personal preference ontology to the system by easily modifying the rule set according to his or her personal interest.

5. CONCLUSION

We proposed a new rule minimization algorithm inspired from logic synthesis to improve comprehensibility and reduce redundant rules. The method can merge several relevant rules into one based on data mining and logic synthesis without high loss of accuracy. From the experiments applied to user preference based anti-spam systems, we found that the proposed rule minimization approaches such as REM and HRP improved the capacity of each rule set and the matched term ratio compared to the original ID3 mining. Also 25% of rules in the ID3 mining results were eliminated by the proposed rule merging strategy. There is no big difference in rules derived by the REM and the HRP. Only the HRP is more convincible to people. In contrast to our previous result [11], we gave some detailed comparison of REM and two other works such as the original ID3 and the HRP, and indicated the proposed rule pruning itself was competitive.

We need to extend the proposed user preferred ontology to process real-time emails provided to each user as our further research. As reviewers indicated, there were some non-textual emails in our data collection and they caused different rule accuracies according to email category. Moreover, they could make the reliability of this work lower such as most content-based filters. However, our system is less sensitive to the problem. Because users

consider email header and body together for their decision on spam or not in this work, this could alleviate the limit with insufficient data set due to mails with images and hyperlinks. In also, this work can be applied to many other areas where small sized and human friendly rules are needed.

ACKNOWLEDGEMENTS

The author appreciates Prof. Dou and others at the Computer and Information Science Department at University of Oregon to support my sabbatical year. I also thank to all students who read sample emails and gave their responses.

REFERENCES

[1] C. M. Files and M. A. Perkowski, "Multi-Valued Functional Decomposition as a Machine Learning Method," Proc. of ISMVL '98, pp. 173-178, 1998.

[2] S. J. Hong, "R-MINI: An Iterative Approach for Generating Minimal Rules from Examples," *IEEE Transaction on Knowledge and Data Engineering*, Vol.9, No.5, pp. 709-717, 1997.

[3] A. Chan and A. Freitas, "A New Classification-Rule Pruning Procedure for an Ant Colony Algorithm," *Lecture Notes in Computer Science*, Vol.3871, pp. 25-36, 2005.

[4] M. J. Zaki and N. Ramakrishnan, "Reasoning about Sets using Redescription Mining," Proc. of KDD'05, pp. 364-373, 2005.

[5] P. Wolfe, C. Scott, and M. Erwin, *Anti-Spam Tool Kit*, McGraw Hill, 2004.

[6] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A bayesian approach to filtering junk e-mail," Proc. of AAAI-98 Workshop on Learning for Text Categorization, pp. 55-62, 1998.

[7] H. Drucker, D. Wu, and V. Vapnik, "Support Vector Machines for Spam Categorization," *IEEE Trans. on Neural Networks*, Vol.10, No.5, pp. 1048-1054, 1999.

[8] Y. Yang and J. P. Pedersen, "A comparative study on feature selection in text categorization," Proc. of the 14th Int'l Conference on Machine Learning, pp. 412-420, 1997.

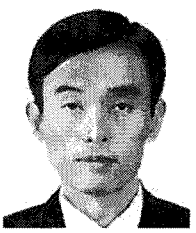
[9] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and Techniques with java implementations*, Morgan Kaufmann, San Francisco, CA, 2000.

[10] M. M. Mano, *Digital Design, 3rd Edition*, Prentice Hall, Englewood Cliffs, NJ, 2001.

[11] J. Kim, D. Dou, H. Liu, and D. Kwak, "Constructing A User Preference Ontology for Anti-spam Mail Systems," *Lecture Notes in Artificial Intelligence*, Vol.4509, pp. 272-283, 2007.

[12] D. McDermott and D. Dou, "Representing disjunction and quantifiers in RDF," Proc. of International Semantic Web Conference, pp. 250-263, 2002.

[13] P. J. Resnick, D. L. Hansen, and C. R. Richardson, "Calculating Error Rates for Filtering Software," *Communications of ACM*, Vol.47, No.9, pp. 67-71, 2004.



Jongwan Kim

received the BS, the MS, and the PhD degree in Dept. of Computer Engineering from Seoul National University, Korea, in 1987, 1989, and 1994, respectively. He has been with Daegu University since 1995 and is currently a professor. From 2006 - 2007, he was a visiting professor at Computer and Information Science Department of University of Oregon, working on the user preference ontology based anti-spam systems with the partial support of KRF. He has written several papers in the areas of information filtering, fuzzy systems, and anti-spam systems. His current research areas include artificial intelligence, data mining and IT service for the information disabled.