Constructing a User Preference Ontology for Anti-spam Mail Systems

Jongwan Kim[1,*], Dejing Dou[2], Haishan Liu[2], and Donghwi Kwak[2]

[1] School of Computer and Information Technology, Daegu University
Gyeonsan, Gyeongbuk. 712-714 South Korea
jwkim@daegu.ac.kr, jongwan@cs.uoregon.edu
[2] Department of Computer and Information Science, University of Oregon
Eugene, Oregon 97403, USA
{dou,ahoyleo,dkwak}@cs.uoregon.edu

Abstract. The judgment that whether an email is spam or non-spam may vary from person to person. Different individuals can have totally different responses to the same email based on their preferences. This paper presents an innovative approach that incorporates user preferences to construct an anti-spam mail system, which is different from the conventional content-based approaches. We build a user preference ontology to formally represent the important concepts and rules derived from a data mining process. Then we use an inference engine that utilizes the knowledge to predict the user's action on new incoming emails. We also suggest a new rule optimization procedure inspired from logic synthesis to improve comprehensibility and exclude redundant rules. Experimental results showed that our user preference based architecture achieved good performance and the rules derived from the architecture and the optimization method have better quality in terms of comprehensibility.

Keywords: user preference ontology, anti-spam system, data mining.

1 Introduction

Spam mail is unsolicited, unwanted email sent indiscriminately, directly or indirectly, by a sender having no current relationship with the recipient [1]. Most software for email clients 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 thwart the keyword filters [2]. A variety of machine learning algorithms such as naïve Bayesian classifier (NBC) and support vector machine (SVM) have been used for email categorization task on different metadata. While these anti-spam filters achieve statistically impressive accuracies, they remain prone to false positives (i.e., non-spam or legitimate email tagged as spam, which is called "ham") and false negatives (spam in your mailbox). More to the point, some email is spam to someone but ham to others in many real situations. For

[*] Currently a visiting scholar at the University of Oregon, USA.

Z. Kobti and D. Wu (Eds.): Canadian AI 2007, LNAI 4509, pp. 272 – 283, 2007.

example, it is possible that the a customer service staff at a credit card company may send many business emails to customers and get feedbacks from some of them. In this case, some customers consider the mail as spam but others find it useful. Users' behaviors to emails vary from one another according to the different personal preferences. Therefore it is meaningful to provide user-oriented anti-spam services based on the preferences. In this work, we have collected user preference information and email responses from a group of college students to train an association and classification mining system. Then we have used the generated rules to define a user preference ontology in a formal language. To show how the ontology can help anti-spam systems, we designed a concept of user preference based anti-spam mail system and did a proof-of-concept implementation. The experimental results indicate that the proposed approach is promising.

The paper is organized as follows. Section 2 introduces some related work. Section 3 describes the data collection and preprocessing. Section 4 covers our ontology construction methodology using association and classification mining. Section 5 presents the proposed architecture and experimental results. Conclusions are given in Section 6.

2 Related Research

Recently some works about personal email management system have been proposed. Gray and Haahr suggested personalized, collaborative spam filtering [3]. Personalized collaborative filters deliver the most relevant spam notices to each user from the collection of all spam messages that are reported by members of the network. To implement this personalized collaborative filter, whenever a new spam is classified, a signature should be computed and propagated to those users likely to receive a similar mail in order to consider it spam. This requires that information is compiled and maintained for each user and other users who are in similar groups. The P2P architecture lends itself nicely to such a system. Since this approach depends on other user's signature in the P2P network, it is not stand alone. On the other hand, Ravi et al. proposed personalized email management at network edges [4]. Their artificial neural network based spam filter performs spam and virus filtering at the server's origin and therefore saves network bandwidth. The system has two filters: 1st filter recognizes text pattern in email and learns from the pattern, and 2nd filter learns images in email. This system is very close to human spam identification. But this methodology has a centralized spam filter which just identifies the spam in one email account and helps it learn and then deletes such spam mails from all other accounts of the same user.

Several anti-spam mail systems considering user preferences are currently operating. Most of them require users' selection about what they accept or not based on the recommendation of the anti-spam system. In these systems, an email that could be a potentially spam but cannot be classified as definitely spam will be stored in a specific area. The intended recipient of the email will receive a web link to the specific area, where any emails held there can be classified. Within the specific area the individual user will be able to classify the email as required. The system will remember the individual user's preference and in the future always transmit or block emails from that particular source to the user's inbox [5]. Another setting included in user preferences specifies the languages in which the ham emails are expected to be

written. Current anti-spam systems considering user preference are summarized; they are mainly based on other users' advice in the same group or other email account information of the same user or user's judgment to accept or not based on the recommendation of the system. However, our goal is to develop a user preference ontology based anti-spam management system which is purely based on user preferences and user responses.

There are two main methods to design any domain ontology: top-down and bottom-up. In the top-down approach, ontology experts determine the concepts and their relationships based on their domain knowledge and intuition. In the bottom-up approach, ontology experts select the important concepts by analyzing data coverage and patterns related to them. Both top-down and bottom up approaches need human involvement, although some automatic tools can reduce manually efforts, e.g., the tools which can acquire ontological knowledge from natural language texts [6]. Text-based learning also can be useful for selecting the keywords for the vocabulary in domain thesauri. In this paper, we focus on finding the relationships between user preferences and their behaviors (i.e., responses to emails). The relationships can be represented as rules (axioms) in the domain ontology.

Data mining is useful in discovering the classification rules but it is hard to tell which rules are useful in terms of comprehensibility and accuracy from probably a great number of rules derived through the mining process. Some rules may be too long or too specific to be useful in ontology construction. There are several works on multi-valued logic in machine learning that can be used for rule minimization and optimization. Files and Perkowski explored the multi-valued logic synthesis (MVLS) method [7]. They described how some concepts of machine learning matched nicely with MVLS and showed how MVLS outperformed both C4.5, the widely used classification algorithm and Espresso, an industry standard logic minimization tool. Also, iterative mining for rules with constrained antecedents was reported [8]. This approach was an iterative algorithm that could exploit mining information gained in previous steps to efficiently answer subsequent queries. In this paper, we suggest a simple rule minimization approach based on logic synthesis inspired from Karnaugh map [9] and data mining to exclude lengthy and redundant rules which are not easy for humans to understand.

3 Data Preparation

We collected data for user preferences and responses to the emails from the undergraduate students majoring in computer science and information technology at Daegu University in Korea. The first author has conducted several experiments on content-based email filtering [10]. From his previous experience, we aimed at developing an anti-spam mail system based on personal interests and behaviors from a specific user or user group instead of mere analysis of the contents of the email header and body.

First, we designed a user profile format to represent the user preference and their different types of responses to the emails. Many web mail systems such as Yahoo and Comcast provide registration forms to collect personal information about the users' interests. Similar to these forms, we chose {Age, Gender, RequiredHits, News, Finance, Sports, Adults, TvMovieMusic, Kids, Games, Travel, Shopping,

Jobs, RealEstates} as attributes to be included in our user profile. The Age attribute has 2 options -FS (= freshman and sophomore) and JS (= junior and senior) - as all the participants were college students. The RequiredHits (from now on, RHit) was originally adopted in Spam Assassin [2]. It is defined as how many hits are required before a mail is considered as spam, which indicates the strength of the spam filter that the user expects. However not like Spam Assassin which uses numbers in this option, we use linguistic terms such as Very Weak (VW), Weak (W), Neutral (N), Strong (S), and Very Strong (VS) because we do not consider email contents and linguistic representation is more comfortable to people. However, since weak (W) was never chosen by users participated in this work, we eliminated the W value from the RHit attribute set in the experiment. If the users want a very strong spam filter, for example, they can choose VW for RHit, or S if a relatively weak one is preferred. We include the information of email category (henceafter, ECat) labeled by human expert together with the users preferences and their responses in the data mining process to study how the preferences affects responses.

Second, the feature selection was performed before we mined the data. 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 number of features are information gain, mutual information, chi squared test and so on. According to previous works on data mining [11], information gain is a good solution to this problem. We calculated information gains for all 15 attributes in a user profile plus the ECat attribute, and chose several attributes from them. The detail of this procedure is described in Section 5.3.

Third, we conclude that email recipients typically have four kinds of response to incoming emails: Reply, Delete, Store, and Spam. When they have no interest on a mail, they just delete it. If they think a mail is important and valuable to respond, then they reply to the sender. Regardless of replying it or not, they sometimes just hold some mails in mail box because the emails might be useful in the future. Finally when a mail is concerned as spam, most of the users move it to a spam dump box to explicitly mark it. Surely, someone can do both actions such as deletion and moving into a spam box but the other either deletes or moves it. The important thing is that most of users would like to maintain only ham in their mail boxes.

Thus we collected some sample emails, personal information and preferences of participating users, and their responses to the samples. Among all the attributes in the profile, most of them have binary values. For example, {FS, JS} are used for the Age attribute and {male (M), female (F)} are given for the Gender attribute, respectively. And all the attributes concerning the user's interests are given in the true-or-false form. If a user is interested in News, he or she checks true for the attribute. However, multi-valued attributes (VW, N, S, and VS) are used for RHit. And ECat attribute uses 12 values to indicate different email category labels; besides the original 11 labels of category, the Etc label is used if no category is labeled for a specific email. The target variable or the output variable of classification mining, Response, has four categories {Reply, Delete, Store, Spam}.

4 Ontology Construction

Ontologies, which can be defined as formal specification of vocabulary of concepts and their relationships, play a key role to define the semantics of information for intelligent systems [12]. To achieve user preference based anti-spam system, it will be helpful to construct a domain ontology which can formally define user behaviors based on their preferences. To do that, we mainly perform three steps. The first step is to use association and classification mining to find relationships (rules) between several users' preferences and their email responses. In the next step, we apply a new rule-pruning procedure eliminating redundant rules and preserving highly comprehensible ones. Translation from optimized rules to axioms in a domain ontology is performed during the final step. The details are described as following.

First, we try 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 [11], to train sample email preference data. Our sample data are composed of mostly binary features and some are nominal features as described in Section 3. 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 describing 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 apply a new rule minimization procedure in order to exclude redundant rules and select highly comprehensible ones. Thus we suggest a rule pruning approach inspired from logic synthesis. Karnaugh map (K-map) is well known as a simple and easy method to understand Boolean logic simplification [9]. It is possible to find two or more simplified logic expressions in a K-map. For example, a function F (A, B, C) = Σ(1, 3, 4, 5, 6, 7) is composed 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 are possible as shown in Figure 1(a) and (b).

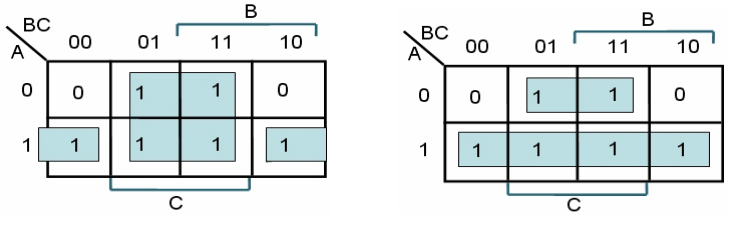

Fig. 1. K-map method examples for 3 variables

As shown in Figure 1, even though the function F is fixed, it is possible to get two different logic representations. Both expressions are equivalent with respect to logic. Since we got the idea about rule minimization from this K-map example, our rule minimization approach is called logic synthesis based rule pruning (LSRP). There are several variables in a rule set derived from data mining. Most variables are Boolean or binary but some of them are multi-valued such as RHit and ECat. In LSRP, if two or more corresponding logic of a specific variable are distinct, the rules with the specific variable are merged into one and then the corresponding antecedent condition of the variable will be omitted in a merged rule. Thus simpler rules can be derived. The following example with six artificial rules shows the idea well.

R1: if Age = JS and ECat = Finance and RHit = S and Adults = F and Games = T and Jobs = T then Response = Spam.
R2: if Age = JS and ECat = Finance and RHit = S and Adults = F and Games = T and Jobs = F then Response = Spam.
R3: if Age = FS and ECat = Adults and RHit = VW and Adults = T then Response = Store.
R4: if Age = FS and ECat = Adults and RHit = N and Adults = T then Response = Store.
R5: if Age = FS and ECat = Adults and RHit = S and Adults = T then Response = Store.
R6: if Age = FS and ECat = Adults and RHit = VS and Adults = T then Response = Store.

There are two similar rules R1 and R2 with only one distinct antecedent condition in Jobs attribute. Therefore the two rules are merged into R1 ·R2 where the antecedent conditions T and F of the variable Jobs are merged into Null because Jobs = T is in conjunction with Jobs = F by logic synthesis operation. So a new rule R7 is derived by excluding the Null condition in R1·R2.

R7: if Age = JS and ECat = Finance and RHit = S and Adults = F and Games = T then Response = Spam.

Similar operation can be also performed for nominal variables. As we mentioned, a variable RHit has four categories. If RHit values of four rules being compared are distinct, then the four rules are merged into one rule. In the above example, R3 has VW value as RHit and the values of RHit in the other three rules are N, S, and VS, respectively. From the synthesis of rules 3, 4, 5, and 6, the antecedent condition of RHit should have Null and hence the condition is excluded to construct a new rule R8.

R8: if Age = FS and ECat = Adults and Adults = T then Response = Store.

It is possible that two or more rules sometimes compete to merge other rules with only one different antecedent condition. 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 should choose a variable with lowest IG and then merge two rules with distinct binary attribute values or several rules with distinctive multiple attribute values in the variable. It is fair that the attributes with higher IGs should

survive in a rule set. As we expected, experimental results (see Section 5.3) are a little different from the ID3 mining.

Third, we interpret the derived classification rules to an ontology using a formal language, Web-PDDL [13], a strongly typed first order language especially for representing ontologies and mappings between them. Based on previous results, we first define a user preference ontology in Web-PDDL and it can be automatically translated to other popular ontology and rule languages, such as OWL [14] and SWRL [15]. We selected the following concepts as classes and properties:

Classes (Types): Preference, Event, Email, Action, Client, Gender, ECat, RHit, Response

Properties (Predicates): name, sex, age, prefer, category, respond

In Web-PDDL, it looks like

```
(define (domain spam_email)
  (:extends (uri "http://orlando.drc.com/daml/ontology/Person/G3/Person-ont-g3r1"
                 :prefix pdt)
            (uri "http://www.w3.org/2000/10/XMLSchema"    :prefix xsd))
  (types: Event Email Preference - Object Action - Event  Client - @pdt:Person
      Gender RHit ECat Age - @xsd:string Response - Action)
  (:Objects Reply Store Delete Spam - Response
            Adults Games Jobs …. - Preference)
  (:predicates (name c - Client n - @xsd:string)
      (sex c - Client s - Gender)
      (age c - Client a - Age)
      (prefer c - Client e - Preference)
      (category e - Email e - ECat)
      (respond c- Client e - Email r - Response)))
```

Where user responses (e.g., Reply Store Delete Spam) and preferences (e.g., Adults Games Jobs) can be defined as objects (instances) of "Response" class and "Preference" class. Then the rules we got from data mining can be put into the user preference ontology as axioms. For example, R8 can be represented in Web-PDDL as axioms:

```
(axioms:
  (forall (c - Client e - Email)
      (if  (and (age c "FS") (prefer c Adults) (category e "Adults")
       (respond c e Store)))
```

In the following section, we will show how to use this ontology in an ontology-based anti-spam system.

5 Architecture, Experiments, Results, and Discussion

In this section, we will present the architecture for our ontology-based system and several measures for evaluation of the system performance and the quality of rules derived from the LSRP process. Experimental results and some discussion are also described.

5.1 Architecture

Each user can respond differently to a mail with even identical mail header and content. This situation is mainly caused by personal preferences and potential modes of behaviors. However, we do not consider users' unpredictable behaviors in this paper, because the work is out of our research scope. We started from this assumption and decided to show that it was valid in real situations. Thus, we collected preferences for a group of users. To analyze potential responses of users to various emails, we provided sample emails to a user group and asked them to respond of the predefined (Reply, Delete, Store, Spam) actions. In this work, "Reply", "Delete", and "Store" responses are considered ham emails but only "Spam" response is considered as spam. Thus, our research is different from conventional anti-spam mail works because we classify user's specific responses into 4 categories instead of spam and ham.

The proposed architecture is given in Figure 2: user profile was collected from several participant users, user log file was also built with their responses to sample emails, and ECat values were given to individual mails by a human expert. We used a popular data mining tool, WEKA [11] developed by Witten and Frank to find some association and classification rules between preferences and responses. User preference ontology was constructed after data mining and rule minimization. Using the ontology, especially those axioms which we constructed, our inference engine OntoEngine [16], which is a first order logic reasoner using generalized modus ponens, can classify the emails into four categories - Spam, Reply, Delete, and Store - based on user preferences, email category, and personal information by forward chaining.

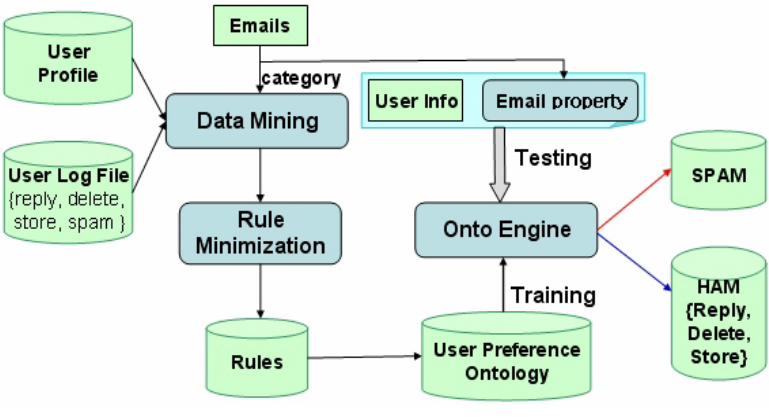

Fig. 2. Architecture of the proposed ontology-based anti-spam mail system

Conventional anti-spam software provides content-oriented filtering service. However, the proposed system can give user oriented anti-spam mail service, because our system not only gives information about spam mail but also estimates user's response to an incoming mail. This approach is new and can be an essential service for email clients suffering from lots of spam messages.

5.2 Measures

Performance measures are required to evaluate inference results of OntoEngine using axioms in user preference ontology. There are several measures such as misclassification, accuracy, prediction, recall, and so on in anti-spam mail system field [1]. Since these measures are calculated from email contents, they can be called email-oriented measures. However, we aim at user preference oriented service and hence different measures are needed to show that the proposed user preference ontology is meaningful to the anti-spam system. In this work, we suggest three measures to achieve this goal.

First, axiom accuracy or axiom confidence is useful to calculate correctness of each axiom in the user preference ontology, which is defined as the ratio of the number of instances that match the rule only in the antecedent part to the number of instances that match the whole rule. Consider the antecedent part and conclusion of each axiom when the input attributes of a test instance exactly match the antecedent conditions of i-th axiom, we increment the match count of the axiom, axiom[i].match. If the response of the test instance is also the same as the conclusion part of the axiom, the correct count of the axiom, axiom[i].correct increments too. Then axiom confidence is calculated by dividing axiom[i].correct over axiom[i].match. The ontology with higher axiom accuracy is more preferable.

Second, conventional classification accuracy is not appropriate in this context. Consider the following scenario. Two users, Bill and John, have almost the same preferences and their responses to training instances are also very similar. They can be grouped into one small user group and their responses to any email are highly possible to be inferred as the same according to the ontology. In this situation, we also suppose that the response of the user group in which Bill and John are included has been inferred as spam for specific antecedent portion in an axiom. All possible instances, including Bill's spam response and John's delete one for the specific antecedent portion in the axiom, have been already reflected during the data mining process. After that, the system judges that Bill's response is correct and John's one is wrong for the test instance matched with corresponding antecedent portion in the axiom. It is not certain to determine which one is correct. It is not guaranteed that users' responses are equivalent to every kind of email. So we introduce axiom capacity as a measure of how many instances can be accommodated by each axiom. Therefore, if the summation of match scores of each axiom (Σ_i axiom[i].match) is equal to the total number of test instances, then the axiom set can accommodate all instances and there is no capacity problem. However, it is not easy because we have mined several thousands of instances with binary as well as nominal attributes to derive tens of rules. We should pass outside instances away from axioms in the ontology to conventional content-based email filters such as NBC or SVM and let them process the instances.

Third, we provide a simple quantified measure to evaluate comprehensibility of a rule derived from the proposed rule minimization method (LSRP). We defined the matched term ratio in a rule, *mt*, for each rule as the number of attributes in each instance over the number of antecedent conditions in each rule. The greater the average value of all matched term ratio in a rule set is, the simpler and more easily interpretable the rule set is to humans. For example, a test instance, (Age = JS and ECat = Adults and RHit = S and Adults = F and Games = T and Jobs = T) and

(Response = Spam) is presented. Two rules (R9: if Age = JS and ECat = Adults and RHit = S and Adults = F and Games = T then Response = Spam) and (R10: if Age = JS and ECat = Adults and RHit= S and Adults = F then Response = Spam) are given. Then *mt*[9] = 6/5 = 1.2 and *mt*[10] = 6/4 = 1.5. Therefore, the R10 rule has a greater matched term ratio than R9 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 [8] but they did not consider the number of input attributes. The above three measures help performance evaluation in terms of axiom confidence, capacity, and comprehensibility.

5.3 Experiments, Results and Discussion

To evaluate the proposed approach based on the user preference ontology, we collected 40 sample emails with labelled categories. We also collected responses to those emails from 90 college students together with their respective preference over several options. And thus we used a dataset with 3,600 records; each record consists of the email category, user preferences, and the corresponding response. We used 2,400 instances of the dataset in the training process, and the rest for testing by performing the rule evaluation methods described in the previous section. Before carrying out the ID3 mining, we selected 6 out of 15 attributes with the highest information gain, namely, ECat, Age, RHit, Adults, Games, and Jobs.

We got 89 rules with accuracies greater than 0% from the ID3 data mining. To evaluate the performance of derived rules, we applied them to the 1,200 test instances and carried out the proposed rule minimization approach (LSRP), which then generated a reduced amount of 77 rules. We interpreted the rules in logic axiom form and selected some representative ones where one rule was chosen for each email category, as shown in Table 1 in descending order of accuracy. Each rule in the table explains the way that a user responds to a certain email with respect to the specific preference. For example, the first axiom rule shows that 85% of users with Adults=False and RHit=Neutral preferences responded as “Spam” when they got adult-related emails. In fact, all adult emails were definitely classified to spam in content-based filters, while this experiment shows that a few users did not think of those kinds of emails as spam. This point convinces us that the proposed user preference ontology based approach can be a customized solution for individual users.

Table 2 shows a comparison of the rule set generated by using rule minimization method and the one without using it. As shown in the table, average axiom rule accuracy was degraded a little by 3.3%. However the number of axiom rules is reduced by 13.5%, and the average matched term ratio and the total capacity are also improved by 15.2% and 1.6% respectively. This reduced rule set with shorter rules is desirable to construct a user preference ontology because we pass the rules to each user and the user feedback personal preference ontology to the system by easily modifying the rule set according to his or her personal interest.

The correlation of the user’s response to a certain kind of email with respect to his/her preference shown in the rules derived from the current experiment data set is not significant enough as expected. It is because the data set is not sufficient to supply samples for all pre-defined email categories and the distribution of which is unbalanced as well. And the lack of sample email data makes it hard to illustrate a clear

pattern of the user's response to a certain email category. To increase the accuracy and practicality of the rules, more data should be collected in the future. However our work points out an innovative anti-spam approach which incorporates the user preference rather than analyzing the email content alone. The result under the current experiment setup also demonstrates a good performance of the proposed rule minimization method.

Table 1. Axioms derived by logic synthesis based rule pruning and their performance

No	Axiom rules	Matched term ratio	Accuracy
1	ECat=Adults ∧ Adults=F ∧ RHit=N => Response=Spam	2	85.0%
2	ECat=Etc ∧ Age=FS ∧ Jobs=T ∧ Games=T ∧ RHit=S => Response=Spam	1.2	82.4%
3	ECat=Finance ∧ Age=JS ∧ Adults=F ∧ Games=F ∧ RHit=N => Response=Spam	1.2	81.5%
4	ECat=News ∧ Age=JS ∧ Jobs=F ∧ Adults=F ∧ RHit=N => Response=Delete	1.2	75.0%
5	ECat=TVMovieMusic ∧ Age=JS ∧ Jobs=T ∧ RHit=N => Response=Reply	1.5	71.4%
6	ECat=IT ∧ Age=JS ∧ Jobs=T ∧ Games=T ∧ RHit=N => Response=Delete	1.2	65.9%
7	ECat=Shopping ∧ Age=JS ∧ Adults=F ∧ Games=F ∧ RHit=N => Response=Spam	1.2	50.0%
8	ECat=Travel ∧ Age=JS ∧ Jobs=T ∧ Games=F ∧ RHit=N => Response=Store	1.2	45. 5%
9	ECat=Jobs ∧ Age=JS ∧ Jobs=T ∧ Adults=F ∧ RHit=N => Response=Spam	1.2	34.6%

Table 2. Comparison on experimental results for two axiom rule sets derived by the original ID3 mining and the proposed rule pruning

Method	Number of axiom rules				Axiom accuracy	Capacity	Matched term ratio
	Reply	Delete	Store	Spam			
ID3	10	18	17	44	60.1%	1111/1200	1.25
LSRP	8	16	15	38	58.1%	1129/1200	1.44
Improv.	20%	11.1%	11.8%	13.6%	-3.3%	1.6%	15.2%

6 Conclusion

We proposed a method to construct user preference ontology for anti-spam mail systems. The important feature of our approach is to allow users to give different response to the same email based on their preferences. It is different from conventional systems that normally judge which mail is spam based on the email content and expect every user to equally respond to the same email. It is a big step forward

personalized anti-spam mail service considering user preference and previous response history as well as email content. The most important contribution of this work is that a user preference ontology can explain why a mail is decided to be spam or ham in a meaningful way. Also, the logic rules found by data mining contribute to this purpose and a rule pruning method improves human comprehensibility. For the future work, we need to extend the proposed system to process real-time and larger number of emails to compare our system's performance with conventional content-based filters. We expect that users have consistent responses when the volume of experiment corpus is bigger and thus the testing result can reflect the impact of the preferences over their decision.

Acknowledgement. The first author was supported by the Korea Research Foundation Grant. (KRF-2006-013-D00285) He has worked as a visiting scholar at the AIM Lab during his sabbatical year 2006 and thanks to the Department of Computer and Information Science at the University of Oregon. Also we appreciate 90 participant students to give their preferences and feedback their responses to sample emails.

References

1. Cormack, G. V., Overview of the TREC 2005 Spam Track, http://plg.uwaterloo.ca/~gvcormac/trecspamtrack05
2. Wolfe, P., Scott, C., and Erwin, M., Anti-Spam Tool Kit, McGraw Hill (2004)
3. Gray, A. and Haahr, M., "Personalized, Collaborative Spam Filtering," in Proc. of the First Conference on Email and Anti-Spam (2004)
4. Ravi, J., Shi, W., and Xu, C., "Personalized Email Management at Network Edges," IEEE Internet Computing, Vol.9(2) (2005) 54-60
5. Anti-Spam Firewall, http://www.barracudanetworks.com/ns/products/anti_spam_tech.php
6. Maedche, A., "Ontology Learning for the Semantic Web," The Kluwer International Series in Engineering and Computer Science, Volume 665 (2003)
7. Files, C. M. and Perkowski, M. A., "Multi-Valued Functional Decomposition as a Machine Learning Method," in Proc. of ISMVL '98 (1998) 173-178
8. Chan, A. and Freitas, A., "A New Classification-Rule Pruning Procedure for an Ant Colony Algorithm," LNCS 3871 (2005) 25-36
9. Sasao T., "Switching Theory for Logic Synthesis," Kluwer Academic Publishers (1999)
10. Kim, J. and Kang, S., "Feature Selection by Fuzzy Inference and Its Application to Spam-Mail Filtering," LNAI 3801 (2005) 361-366
11. Witten, I. H. and Frank, E., Data Mining: practical machine learning tools and techniques, 2nd ed, Morgan Kaufmann (2005)
12. Gruber, T. R., "Toward Principles for the Design of Ontologies Used for Knowledge Sharing," Int. Journal of Human-Computer Studies, Vol.43 (1995) 907-928
13. McDermott, D. and Dou D., "Representing disjunction and quantifiers in RDF," in Proc. Int'l Semantic Web Conference (2002) 250-263
14. OWL Web Ontology Language. http://www.w3.org/TR/owl-ref/
15. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. http://www.w3.org/Submission/SWRL/
16. Dou, D,, McDermott, V., and Qi, P., "Ontology translation on the semantic web," Journal of Data Semantics, Vol.2 (2004) 35-57