User Behavior Analysis of the Open-Ended Document Classification System*

Yang Sok Kim[1], Byeong Ho Kang[1], Young Ju Choi[1], SungSik Park[1], Gil Cheol Park[2], and Seok Soo Kim[2]

[1] School of Computing, University of Tasmania
Sandy Bay, Tasmania 7001, Australia
{yangsokk, bhkang, Y.J.Choi, sspark}@utas.edu.au
[2] School of Information & Multimedia, Hannam University
133 Ojung-Dong, Daeduk-Gu, Daejeon 306-791, Korea
{gcpark, sskim}@hannam.ac.kr

Abstract. Real-world document classification is an open-ended problem, rather than a close-ended problem, because the document classification domain continually evolves as the time passes. Unlike the close-ended document classification, the participants in the open-ended problem actively take part in the problem solving process. For this reason, it is important to understand the problem solver's behavioral characteristics. This paper proposes a thorough analysis of them. We found that the problem solving strategies are significantly different among participants because of individual differences in cognition among participants.

1 Introduction

Automated document classification has been a significant research topic. In the 1980's a common approach to document classification was rule-based, which involved a human in the construction of classifier. Though such a method provides accurate rules and has the additional benefit of being human understandable, the construction of such rules requires significant human input and the human needs some knowledge concerning the details of rule construction as well as domain knowledge, which become a bottleneck of this approach [1]. As an alternative, the machine learning (ML) approach has become the dominant one since 1990's. The ML system requires a set of pre-classified training documents and automatically produces a classifier from documents and the domain expert is needed only to classify a set of existing documents.

Although ML method produced accurate classifiers, there are a number of drawbacks as compared to a rule-based one [2]. Some limitations of the ML approach come from its assumptions about the document classification problem. Generally problems can be classified either close-ended or open-ended. Whereas the former usually has specified problem solving goals, correct answers and clearly defined criteria for success at problem solving, the latter typically has a lack of clearly defined

* This work is supported by the Asian Office of Aerospace Research and Development (AOARD) (Contract Number:FA5209-05-P-0253).

A. Sattar and B.H. Kang (Eds.): AI 2006, LNAI 4304, pp. 1089 – 1095, 2006.

goals, no single correct solution to a problem, and no immediately obvious criteria for making a judgment as to what constitutes a correct solution [3, 4].

The ML approach is basically based on the close-ended assumption, because it uses finite and known data set and its significant goal is to find a method that successfully classifies documents into the predefined classes. In addition, there are lots of clearly defined success criteria of the classification method. Though this assumption has made considerable contributions, it is beneficial to look into the document classification problem from the open-ended viewpoint. First of all, sometimes the document classification problem, especially in the real-world document classification, becomes a kind of the open ended problem. For example, let's assume that we want to classify news articles from the web-based news provider. In this case, documents that should be classified are not known until they are presented and classes continually evolve over time, not pre-defined. Furthermore, most people share the intuition that it is not *a priori* clear that the results from the close-ended research will be generalized in the open-ended domains [3].

2 System Requirements and Implementation

We reviewed the research results of the open-ended education or intelligent tutoring systems and elicited the following three requirements for the open-ended document classification system:

Firstly, the system user should actively take part in the problem solving process. We select the rule-base approach because whereas the rule-based approach assumes the **active role of the problem solvers**, the ML approach excludes the system users from the learning process [5]. The problem solver can be either experts or novices in the domain. This approach is philosophically similar to that of the constructivist approach of knowledge acquisition [6]. Secondly, the open-ended system can provide multiple solutions for the same classification problem, because each user has individual differences in cognition [7]. Therefore, supporting **multiple classifications** is an essential requirement of the classification system. In the classification problem, this has a two-fold meaning. On one hand, it means that ***(1) a case can be classified into multiple classes*** without causing dissatisfactions among the problem solvers. On the other hand, ***(2) multiple cases can be classified into one class*** because of different reasons. Lastly, the system should support **negotiating interactions** between the system and the users. In the education, the more the problem solver knows about the domain and the learners, the more he/she can transfer more of his/her knowledge to the learner [4, 5, 8]. Likewise, the system and its users can interact to improve the system knowledge base. The system can provide current classification recommendation based on the current knowledge base and the user can either accept them or not. If the user does not want to accept the current recommendation, he/she can create another rule(s) to fix the current error. In this way, the system incrementally constructs its knowledge base over the time. Especially this is an essential requirement when the system user is a novice in a domain because he/she needs to refine the system knowledge base as he/she learns the domain knowledge.

An open-ended document classification system was developed with the MCRDR (Multiple Classification Ripple-Down Rules) algorithm, C++ program language, and

MySQL database to fulfill these requirements. MCRDR is based on the traditional rule-based system, but it overcomes traditional knowledge acquisition bottleneck problem by employing the exception-based knowledge representation and case-based validation [9, 10]. In this paper, we will focus not on the system itself, but on the user's behaviors because the system users have different roles in the problem solving process and it is important to understand how they act while constructing knowledge base. More detailed explanation about the classifier in [11, 12].

3 Inference and Knowledge Acquisition

A case is defined by attributes as follows:

$$CASE = T \cup B$$

where T is a distinct word set of hyperlink text and B is a distinct word set of the main content of the linked document. T and B are respectively defined as $T = \{t_1, t_2, \ldots, t_N\}$ and $B = \{b_1, b_2, \ldots, b_M\}$, where N and M are the number of distinct word and are greater that 0 (N, M $\geq$ 0). t_i and b_j are a word in the hyperlink text, which is text between <a> tags, and the hyperlinked document.

A rule structure is defined as follows:

IF

$$(TC \subset T)\ AND\ (BC \subset B)\ AND\ (AC \subset T\ OR\ AC \subset B)$$

THEN

$$Classify\ into\ folder\ F_i$$

where TC is a condition set for the hyperlink text, BC is a condition set for the hyperlinked document, and AC is a condition set for the hyperlink text or the hyperlinked document. Each set is defined as $TC = \{tc_1, tc_2, \ldots, tc_X\}$, $BC = \{bc_1, bc_2, \ldots, bc_Y\}$, and $AC = \{ac_1, ac_2, \ldots, ac_Z\}$, where tc_i is the word in the hyperlink text, bc_j is the word in the hyperlinked document, and ac_k is the word either in the hyperlink text or in the hyperlinked document. The number of words in each condition is greater than 0 (X, Y, Z $\geq$ 0).

In the inference process, the MCRDR-Classifier evaluates each rule node of the knowledge base (KB). If a case is selected from the case list (CL), the system evaluates rules from the root node and the inference result is provided by the last rule satisfied in a pathway. All children of the satisfied parent rule are evaluated, allowing for multiple conclusions. The conclusion of the parent rule is only given if none of the children are satisfied [10, 13].

The knowledge acquisition process begins when a case has been classified incorrectly or is missing a classification. Fig.1 illustrates knowledge acquisition algorithm. In the MCRDR-Classifier, the knowledge base (KB) is automatically maintained by the system. If a new knowledge acquisition process begins, the MCRDR-Classifier decides the location of a new rule according to the rule type. If there is no classification recommendation, the new rule is located under the root rule. If there is some inference result, but the user thinks it is wrong, a refining rule is located under the current firing rule. The stopping rule is a specific refining rule that has NULL conclusion.

```
begin
1.  User selects New Conclusion (NC)
2.  System generates Attribute List (AL) of Current Case
    If the new rule (NR) is refining rule,
        AL is attributes of current case that are not in the current firing rule's corner-
        stone case.
    Else
        AL is all attributes of current case
3.  User selects condition(s) from the AL
4.  System generates Tentative Firing Cases (TFC)
    If user agrees all TFC are also fired by NR
        End the KA process
    Else
        User selects Exclusive Case (EC) from TFC
        Recursively do again step 2 ~ 4.
end
```

Fig. 1. Knowledge Algorithm in the MCRDR-Classifier

4 Experiment and Results

The experiment is designed to analyze knowledge acquisition behaviors of the open-ended document classification users. This experiment was conducted by 20 Master and Hounours course students at the University of Tasmania for four months from August, 2005 ~ November, 2005. The Web monitoring system, called WebMon, continually collected newly updated documents from 9 well known information technology news Websites. Each participant could read the collected documents in real-time and train their own MCRDR-Classifiers. The classification structure (folder structure) of 86 folders was predefined for the experimental purpose, but each participant might use any number of folders for the classification, which totally depended on each participant's intention. In addition, the participants used a different number of documents based on his/her document filtering level.

4.1 Overall Classification Results

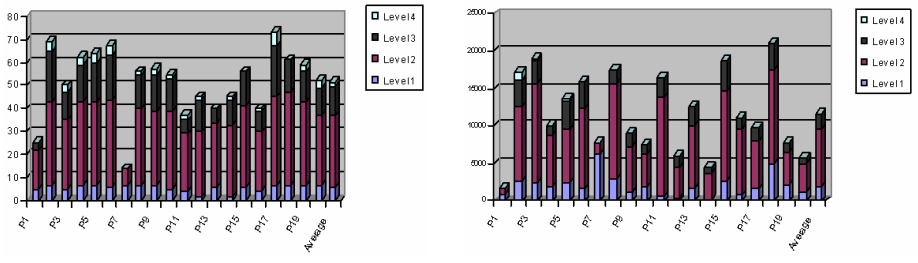
In total 12,784 articles were collected during the experiment period. In overall 95.6% documents (12,304) were used by the participants. The lowest ratio was 87.6% (BBC) and the highest ratio was 99.8% (ITNews). 13.0% documents were commonly used by all participants, which varied from 5.1% to 22.4%.

4.2 Classification Results by the Participants

Used Documents. Each participant classified documents into multiple folders by using the MCRDR-Classifier. Though the same monitored documents were provided to all the participants, the document usage results are very different. The smallest number is 1,775 and the largest number 21,045. The mean number is 11,693. The differences

caused by two factors. Firstly, ***the document filtering levels*** are different among the participants. That is, the documents that each participant felt sufficiently important to classify were different among participants. Secondly, the ***multiple classifications of each participant*** were also different among the participants because some participants tended to classify document multiple classifications whereas others did not.

Used Folders. A total 86 hierarchical folders were used for this experiment: level one (8), level two (40), level three (31), and level four (8). The numbers of folders that were used for classification were different among participants. They varied from 14 to 73, with the mean number of used folders being 51.35. The most used folders were in level 2, and the least used folders in level 4. There is no evidence of the symmetric relationship between the number of used folder and the number of classified documents. For example, though P8, P9, and P10 used a very similar number of folders their correspondent document use is very different.



(a) Folder Usage (b) Documents in Folders

Fig. 2. Folder Usage by Participants

4.3 Knowledge Acquisition

Rules. On average, the participants created 254 rules for 51 folders with 579 conditions to classify 11,693 documents. The minimum number of rules created was 59 (P13) and the maximum (P18, P19) 597. Documents per rule were 62, rules per folder numbered 5.3, with conditions per rule being 2.3. To examine relationships between document classification and rule creation, and between folder creation and rule creation, correlation values were calculated. The correlation between document classification and rule creation ($CR_{d,r}$) was 0.27 and folder (class) creation and rule creation ($CR_{f,r}$) 0.49.

Conditions. Participants were able to use three different types of condition words, which were seen in title (Type 1), seen in body (Type 2), and seen in both title and body (Type 3). Fig. 3. illustrates each participant's condition usage ratio of the three types of rules. Condition selecting depends on each participant's rule construction strategy. Whereas some participants mainly used title condition words (P5, P20), others frequently employed all conditions words (P7, P10, P17).

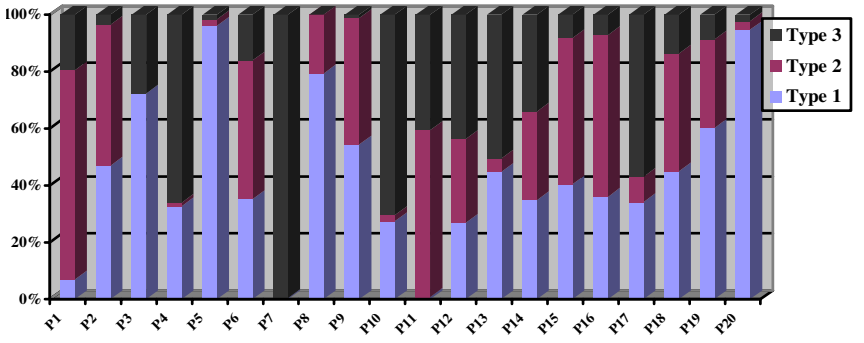


Fig. 3. Participants' Condition Usage Comparison

5 Conclusions

A new problem solving approach is required for open-end problems since they differ from close-ended problems. Real-world document classification is a kind of open-ended problem, because there are no pre-defined classes and cases, and it is possible to classify cases with various coexisting document classifications. We firstly examined the open-ended education experiences to obtain insights into this matter. Active roles of the problem solver, multiple solutions for the same problem and negotiating interactions between the problem solver and the learner were extracted. The MCRDR knowledge acquisition method was employed to implement an open-ended document classification system.

We conducted experiments to analyze knowledge acquisition behaviors. Twenty participants used the MCRDR-Classifier to classify real-world documents. The experiment results show that the participants have different problem solving behaviors while using the open-ended document classification problem.

References

1. Apte, C., F. Damerau, and S.M. Weiss, *Automated learning of decision rules for text categorization.* ACM Transactions on Information Systems (TOIS), 1994. **12**(3): p. 233 - 251.
2. Hirsch, L., M. Saeedi, and R. Hirsch. *Evolving Rules for Document Classification*. in *8th European Conference, EuroGP 2005*. 2005. Lausanne, Switzerland.
3. Goel, V. *Comparison of Well-Structured & Ill-Structured Task Environments and Problem Spaces*. in *Fourteenth Annual Conference of the Cognitive Science Society*. 1992. Hillsdale, NJ: Erlbaum.
4. Cook, J., *Bridging the Gap Between Empirical Data on Open-Ended Tutorial Interactions and Computational Models.* International Journal of Artificial Intelligence in Education, 2001. **12**: p. 85-99.
5. Andriessen, J. and J. Sandberg, *Where is Education Heading and How About AI?* International Journal of Artificial Intelligence in Education, 1999. **10**: p. 130-150.
6. Shaw, M.L.G. and J.B. Woodward, *Modeling expert knowledge.* Knowledge Acquisition, 1990. **2**(3): p. 179 - 206.

7. Dillon, R.F. and R.R. Schmeck, *Individual Differences in Cognition*. Vol. 1. 1983, New York, USA: Academic Press, Inc.
8. Hong, N.S., *The Relationship Between Well-Structured and Ill-Structured Problem Solving in Multimedia Simulation*, in *The Graduate School, College of Education*. 1998, The Pennsylvania State University.
9. Byeong Ho, K., *Validating Knowledge Acquisition: Multiple Classification Ripple Down Rules*, in *School of Computer Science and Engineering*. 1995, University of New South Wales.
10. Kang, B., P. Compton, and P. Preston. *Multiple Classification Ripple Down Rules : Evaluation and Possibilities.* in *9th AAAI-Sponsored Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*. 1995. Banff, Canada, University of Calgary.
11. Park, S.S., Y.S. Kim, and B.H. Kang. *Web Document Classification: Managing Context Change*. in *IADIS International Conference WWW/Internet 2004*. 2004. Madrid, Spain.
12. Kim, Y.S., et al. *Adaptive Web Document Classification with MCRDR*. in *International Conference on Information Technology: Coding and Computing ITCC 2004*. 2004. Orleans, Las Vegas, Nevada, USA.
13. Compton, P. and D. Richards, *Generalising ripple-down rules.* Knowledge Engineering and Knowledge Management Methods, Models, and Tools. 12th International Conference, EKAW 2000. Proceedings (Lecture Notes in Artificial Intelligence Vol.1937), 2000: p. 380-386.